2014 22nd International Conference on Pattern Recognition

Label Consistent Fisher Vectors for Supervised Feature Aggregation

Quan Wang[⋆] Xin Shen[†] Meng Wang[⋆] Kim L. Boyer[⋆]
[⋆]Department of Electrical, Computer, and Systems Engineering
[†]Department of Mathematical Sciences
Rensselaer Polytechnic Institute, Troy, NY 12180, USA
{wangq10, shenx5, wangm7}@rpi.edu kim@ecse.rpi.edu

***Abstract*—In this paper, we present a simple and efficient way to add supervised information into Fisher vectors, which has become a popular image representation method for image classification and retrieval purposes in recent years. The basic idea of our approach is to improve the Fisher kernel in the training process by adding a discriminative label comparison matrix to it. The resulting new representations, which we call Label Consistent Fisher Vectors (LCFV), can be solved for both overdetermined and underdetermined cases. We show that LCFV has better classification performance than traditional Fisher vectors on three public datasets.**

Keywords*—*supervised information; Fisher kernel; image classification; feature aggregation

I. Introduction

Image classification has always been one of the key problems in computer vision, and has many applications such as scene recognition [1], digit recognition, content-based image retrieval [2] and even style categorization [3]. A common practice to represent an image is to extract low-level features such as SIFT [4] descriptors of local patches, and build statistical models to aggregate these low-level features.

A widely adopted model is the bag-of-words (BoW) model, where a "visual vocabulary" is learned by clustering a large set of local features with k-means, and an image is represented with a histogram by simply counting the occurrence or frequency of each "visual word". The optionally normalized histogram of an image will work as higher-level features for classification tasks.

In recent years, the Fisher vector method [5], [6], [7] has been proposed as an alternative of the bag-of-words model. This method is based on the work of the Fisher kernel [8], which models the generative process of a signal. The basic idea of the Fisher vector is to represent a signal by the whitened gradient of its probability density function (PDF) with respect to the parameters of the PDF. Usually, people use Gaussian mixture models (GMM) as the probability distribution for Fisher vectors. As stated by the authors of [9], while the bag-of-words method encodes only the 0-order statistics of the distribution, the Fisher vector method extends BoW by encoding up to second order statistics. The Fisher vector model has been applied to lots of image classification problems, such as large scale image classification [6] and retrieval [10], [11], scene classification [12], aesthetic quality assessment [13], and photographic style categorization [3].

There are a number of ways to add supervised information into the Fisher vector representation. The simplest way is to learn the GMM in a supervised manner. In such a case, however, the learned GMM is task-specific. Thus we need to learn a different GMM for each set of categories [5], which is computationally expensive. Another method is the Fisher kernel learning (FKL) [14], which trains the model to induce similar gradients for signals with the same class label, and "maximizes the expected number of correctly classified objects by a 1-nearest neighbor classifier". Our method, the Label Consistent Fisher Vector (LCFV), is a novel approach in which we improve the Fisher kernel in the training process by adding a discriminative label comparison matrix to it, and solve for a transformation matrix which projects the gradients to approximate the improved kernel. This approach is straightforward, easy to implement, and computationally efficient.

The rest of this paper is organized in the following way: Section II briefly reviews the Fisher kernel and the Fisher vector method that were introduced in previous literature; Section III presents our label consistent Fisher vector method; Section IV presents the experimental results on three public datasets; and finally Section V is the conclusions.

II. Fisher Kernels and Fisher Vectors

In this section, we review the basic concepts and notations of Fisher kernels and Fisher vectors. For one image, let $X = \{x_t\}_{t=1}^{T}$ be the set of T local descriptors. The generative process of X follows the probability density function $p_\theta(X)$ where θ is the set of parameters. The contribution of the parameters to the generative process can be described by the gradient of the log-likelihood:

$$\begin{aligned} g_\theta(X) &= \frac{1}{T}\nabla_\theta \log p_\theta(X) \\ &= \frac{1}{T}\sum_{t=1}^{T}\nabla_\theta \log p_\theta(x_t). \end{aligned} \tag{1}$$

Let the Fisher information matrix of $p_\theta(X)$ be

$$F_\theta = E_X[g_\theta(X)g_\theta(X)^\mathsf{T}], \tag{2}$$

then the Fisher kernel [8] on two images X_1 and X_2 is defined as

$$K(X_1, X_2) = g_\theta(X_1)^\mathsf{T} F_\theta^{-1} g_\theta(X_2). \tag{3}$$

Since the Fisher information matrix F_θ is symmetric and positive definite, we can find the Cholesky decomposition of its

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.617

inverse $F_\theta^{-1} = L_\theta^\mathsf{T} L_\theta$, in which L_θ is an upper triangular matrix. The Fisher vector of X is defined as $\mathcal{G}_\theta(X) = L_\theta g_\theta(X)$, and the kernel can be rewritten as

$$K(X_1, X_2) = \mathcal{G}_\theta(X_1)^\mathsf{T} \mathcal{G}_\theta(X_2). \tag{4}$$

Since in Eq. (4) the kernel is simply the dot-product of two Fisher vectors, a linear classifier on the Fisher vectors will be equivalent to a kernel classifier on the Fisher kernel [6].

Now assume $p_\theta(x_t)$ is a Gaussian mixture model of K components $p_\theta(x_t) = \sum_{i=1}^{K} w_i p_{\theta_i}(x_t)$, where $\theta_i = \{\mu_i, \Sigma_i\}$, μ_i is the mean vector, and Σ_i is the covariance matrix. To simplify, Σ_i is assumed to be diagonal and can be denoted as the variance vector σ_i^2. Let $\gamma_t(i)$ be the soft assignment of x_t to the ith component [9]:

$$\gamma_t(i) = \frac{w_i p_{\theta_i}(x_t)}{\sum_{j=1}^{K} w_j p_{\theta_j}(x_t)}, \tag{5}$$

then the gradient $g_\theta(X)$ can be mathematically derived:

$$\begin{aligned} g_{\mu_i}(X) &= \frac{1}{T}\sum_{t=1}^{T} \gamma_t(i) \left(\frac{x_t - \mu_i}{\sigma_i^2} \right), & (6)\\ g_{\sigma_i}(X) &= \frac{1}{T}\sum_{t=1}^{T} \gamma_t(i) \left(\frac{(x_t - \mu_i)^2}{\sigma_i^3} - \frac{1}{\sigma_i} \right). & (7) \end{aligned}$$

If the mean vector μ_i and the variance vector σ_i are both D-dimensional, then the gradient vector $g_\theta(X)$ or the Fisher vector $\mathcal{G}_\theta(X)$ is $2KD$-dimensional.

III. Label Consistent Fisher Vectors (LCFV)

The entire process of computing the Fisher vectors in Section II is unsupervised and makes no use of class labels of the training images. Now we consider a training set of N images $\{X_i\}_{i=1}^N$, and c_i is the class label of image X_i. In the Fisher kernel, a large value of $K(X_i, X_j)$ means that images X_i and X_j are similar, while a small value means dissimilar. If we add supervised information to the kernel, we are expected to make it better. Thus we define

$$\widetilde{K}(X_i, X_j) = K(X_i, X_j) + \alpha C_{i,j}, \tag{8}$$

where $\alpha > 0$, $C_{i,j}$ takes value 1 when $c_i = c_j$, and takes value 0 otherwise.

A. Problem Formulation

Assume there are N images, and each Fisher vector is M-dimensional. Let $\mathbf{G} = [\mathcal{G}_\theta(X_1), \ldots, \mathcal{G}_\theta(X_N)]$ be the $M \times N$ matrix of the Fisher vectors of all training images, $\mathbf{C} = [C_{i,j}]$ be the label comparison matrix, and $\mathbf{K}$ be the Fisher kernel matrix. For basic image classification tasks, the class labels c_i's are mutually exclusive, thus according to Theorem A.1 in Appendix A, $\mathbf{C}$ is positive semi-definite. The resulting new kernel $\widetilde{\mathbf{K}} = \mathbf{K} + \alpha \mathbf{C}$ is supposed to capture better similarity/dissimilarity information, and it is a valid kernel by Mercer's theorem since both $\mathbf{K}$ and $\mathbf{C}$ are positive semi-definite. Now we seek to find a transformation matrix $\mathbf{M}$, such that $(\mathbf{MG})^\mathsf{T}(\mathbf{MG})$ approximates matrix $\widetilde{\mathbf{K}}$:

$$(\mathbf{MG})^\mathsf{T}(\mathbf{MG}) = \mathbf{G}^\mathsf{T}\mathbf{G} + \alpha \mathbf{C}. \tag{9}$$

Apart from Eq. (9), we also wish the matrix $\mathbf{M}$ to approximate the identity matrix $\mathbf{I}$ because we want to preserve the good properties of Fisher vectors $\mathbf{G}$. We can solve for the matrix $\mathbf{M}$ on the training set with label information in $\mathbf{C}$. When a new image without class label comes, we first compute its Fisher vector $\mathcal{G}$, then its LCFV representation is $\mathbf{M}\mathcal{G}$. LCFV is expected to be a more discriminative representation than traditional Fisher vectors.

In the rest of this section, we discuss two solutions to the problem Eq. (9), and we name them LCVF1 and LCFV2 respectively.

B. LCFV1

Let $\mathbf{M}$ be an $M \times M$ matrix and $\mathbf{W} = \mathbf{M}^\mathsf{T}\mathbf{M} = \mathbf{I} + \mathbf{B}$. Instead of solving for $\mathbf{M}$, we solve for $\mathbf{W}$ first. Now Eq. (9) becomes:

$$\mathbf{G}^\mathsf{T}\mathbf{B}\mathbf{G} = \alpha \mathbf{C}. \tag{10}$$

1) Overdetermined cases: Since the matrix $\mathbf{G}$ is $M \times N$, if $N \geq M$, Eq. (10) is an overdetermined system and does not necessarily have an exact solution. Thus we seek to minimize the Frobenius norm of the error:

$$\min_{\mathbf{B}} ||\mathbf{G}^\mathsf{T}\mathbf{B}\mathbf{G} - \alpha\mathbf{C}||_F. \tag{11}$$

This can be simply solved by pseudo-inverse:

$$\mathbf{B} = \alpha(\mathbf{G}\mathbf{G}^\mathsf{T})^{-1}\mathbf{G}\mathbf{C}\mathbf{G}^\mathsf{T}(\mathbf{G}\mathbf{G}^\mathsf{T})^{-1}. \tag{12}$$

2) Underdetermined cases: If $N < M$, Eq. (10) is an underdetermined system, and the solution is not unique. Thus we need to add extra constraints. Since we wish $\mathbf{M}$ to approximate $\mathbf{I}$, we also want $\mathbf{B}$ to be close to the zero matrix. Now based on Eq. (10), we can minimize the Frobenius norm of the $M \times M$ matrix $\mathbf{B}$:

$$\begin{aligned} \min_{\mathbf{B}} \quad & ||\mathbf{B}||_F \\ \text{s.t.} \quad & \mathbf{G}^\mathsf{T}\mathbf{B}\mathbf{G} = \alpha\mathbf{C}. \end{aligned} \tag{13}$$

Let the singular value decomposition (SVD) of matrix $\mathbf{G}$ be $\mathbf{G} = \mathbf{U}\mathbf{S}\mathbf{V}^\mathsf{T}$. Eq. (10) becomes $\mathbf{S}^\mathsf{T}\mathbf{U}^\mathsf{T}\mathbf{B}\mathbf{U}\mathbf{S} = \alpha\mathbf{V}^\mathsf{T}\mathbf{C}\mathbf{V}$. The matrix $\mathbf{S}$ is $M \times N$ and is a non-square diagonal matrix. Let $\mathbf{S} = \begin{bmatrix} \mathbf{S}_1 \\ \mathbf{0} \end{bmatrix}$ where $\mathbf{S}_1$ is the $N \times N$ submatrix, and $\mathbf{U}^\mathsf{T}\mathbf{B}\mathbf{U} = \begin{bmatrix} \mathbf{Z}_1 & \mathbf{Z}_2 \\ \mathbf{Z}_3 & \mathbf{Z}_4 \end{bmatrix}$. Thus we only need to ensure $\mathbf{Z}_1 = \alpha\mathbf{S}_1^{-1}\mathbf{V}^\mathsf{T}\mathbf{C}\mathbf{V}\mathbf{S}_1^{-1}$, and $\mathbf{Z}_2$, $\mathbf{Z}_3$, $\mathbf{Z}_4$ can be any matrices. Since Frobenius norm is invariant under a unitary transform, we have

$$\begin{aligned} ||\mathbf{B}||_F^2 =& ||\mathbf{U}^\mathsf{T}\mathbf{B}\mathbf{U}||_F^2 \\ =& ||\mathbf{Z}_1||_F^2 + ||\mathbf{Z}_2||_F^2 + ||\mathbf{Z}_3||_F^2 + ||\mathbf{Z}_4||_F^2. \end{aligned} \tag{14}$$

Thus the solution to problem Eq. (13) is $\mathbf{Z}_2 = \mathbf{Z}_3 = \mathbf{Z}_4 = \mathbf{0}$, and

$$\mathbf{B} = \mathbf{U}\begin{bmatrix} \alpha\mathbf{S}_1^{-1}\mathbf{V}^\mathsf{T}\mathbf{C}\mathbf{V}\mathbf{S}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{U}^\mathsf{T}. \tag{15}$$

An interesting observation is that, since the rank of a matrix is also invariant under unitary transforms, we know $\mathrm{rank}(\mathbf{B}) =$

$\mathrm{rank}(\mathbf{U}^\mathsf{T}\mathbf{B}\mathbf{U})$, thus Eq. (15) is also one solution to the rank minimization problem:

$$\begin{aligned} \min_{\mathbf{B}} \quad & \mathrm{rank}(\mathbf{B}) \\ \text{s.t.} \quad & \mathbf{G}^\mathsf{T}\mathbf{B}\mathbf{G} = \alpha\mathbf{C}. \end{aligned} \tag{16}$$

3) Solving for $\mathbf{M}$*:* Once we have determined the matrix $\mathbf{B}$, we get $\mathbf{W} = \mathbf{I} + \mathbf{B}$. Since the matrix $\mathbf{C}$ is symmetric, $\mathbf{W}$ is also symmetric. Let the eigenvalue decomposition of $\mathbf{W}$ be $\mathbf{W} = \mathbf{Q}_w\mathbf{\Lambda}_w\mathbf{Q}_w^\mathsf{T}$ where $\mathbf{Q}_w$ is a unitary matrix, then we simply have $\mathbf{M} = \mathbf{P}\mathbf{\Lambda}_w^{1/2}\mathbf{Q}_w^\mathsf{T}$ where $\mathbf{P}$ is an arbitrary unitary matrix. Since we want $\mathbf{M}$ to approximate $\mathbf{I}$, we can minimize the Frobenius norm of the difference:

$$\begin{aligned} \min_{\mathbf{P}} \quad & ||\mathbf{P}\mathbf{\Lambda}_w^{1/2}\mathbf{Q}_w^\mathsf{T} - \mathbf{I}||_F \\ \text{s.t.} \quad & \mathbf{P} \text{ is a unitary matrix.} \end{aligned} \tag{17}$$

Since Frobenius norm is invariant under unitary transforms, we know

$$||\mathbf{P}\mathbf{\Lambda}_w^{1/2}\mathbf{Q}_w^\mathsf{T} - \mathbf{I}||_F = ||\mathbf{\Lambda}_w^{1/2} - \mathbf{P}^\mathsf{T}\mathbf{Q}_w||_F. \tag{18}$$

According to Theorem A.2 in Appendix A, it can be shown that the solution to Eq. (17) is $\mathbf{P} = \mathbf{Q}_w$. Thus our final solution is $\mathbf{M} = \mathbf{Q}_w\mathbf{\Lambda}_w^{1/2}\mathbf{Q}_w^\mathsf{T}$.

C. LCFV2

Another way of solving Eq. (9) is to directly work on $\mathbf{M}$ rather than $\mathbf{W}$. Although $\mathbf{C}$ is not necessarily positive definite and may have no Cholesky decomposition, as long as the class labels are mutually exclusive, $\mathbf{C}$ is positive semi-definite (Theorem A.1). Let the eigenvalue decomposition of $\mathbf{C}$ be $\mathbf{C} = \mathbf{Q}_c\mathbf{\Lambda}_c\mathbf{Q}_c^\mathsf{T}$. Let L be the number of classes, then $\mathrm{rank}(\mathbf{\Lambda}_c) = \mathrm{rank}(\mathbf{C}) = L$. Assume the L non-zero eigenvalues of $\mathbf{\Lambda}_c$ are on the first L rows of $\mathbf{\Lambda}_c$, and let $\mathbf{A}$ be the first L rows of $\mathbf{\Lambda}_c^{1/2}\mathbf{Q}_c^\mathsf{T}$. Then the $L \times N$ matrix $\mathbf{A}$ satisfies $\mathbf{C} = \mathbf{A}^\mathsf{T}\mathbf{A}$.

Now Eq. (9) can be rewritten as:

$$\begin{aligned} (\mathbf{M}\mathbf{G})^\mathsf{T}(\mathbf{M}\mathbf{G}) =& \mathbf{G}^\mathsf{T}\mathbf{G} + \alpha\mathbf{A}^\mathsf{T}\mathbf{A} \\ =& \begin{bmatrix} \mathbf{G}^\mathsf{T} & \sqrt{\alpha}\mathbf{A}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathbf{G} \\ \sqrt{\alpha}\mathbf{A} \end{bmatrix}. \end{aligned} \tag{19}$$

This can be simplified to $\mathbf{M}\mathbf{G} = \begin{bmatrix} \mathbf{G} \\ \sqrt{\alpha}\mathbf{A} \end{bmatrix}$. If we assume $\mathbf{M} = \begin{bmatrix} \mathbf{I} \\ \mathbf{E} \end{bmatrix}$ where $\mathbf{E}$ is an $L \times M$ matrix, we just need:

$$\mathbf{E}\mathbf{G} = \sqrt{\alpha}\mathbf{A}. \tag{20}$$

1) Overdetemined cases: Similar to LCFV1, if $N > M$, Eq. (20) is an overdetermined system, and we seek to minimize the Frobenius norm of the error:

$$\min_{\mathbf{E}} ||\sqrt{\alpha}\mathbf{A} - \mathbf{E}\mathbf{G}||_F. \tag{21}$$

The solution to (21) is simply:

$$\mathbf{E} = \sqrt{\alpha}\mathbf{A}\mathbf{G}^\mathsf{T}(\mathbf{G}\mathbf{G}^\mathsf{T})^{-1}. \tag{22}$$

2) Underdetermined cases: If $N < M$, Eq. (20) is an underdetermined system, and the solution is not unique. Similar to LCFV1, we can minimize the Frobenius norm of matrix $\mathbf{E}$:

$$\begin{aligned} \min_{\mathbf{E}} \quad & ||\mathbf{E}||_F \\ \text{s.t.} \quad & \mathbf{E}\mathbf{G} = \sqrt{\alpha}\mathbf{A}. \end{aligned} \tag{23}$$

Again, let the singular value decomposition of matrix $\mathbf{G}$ be $\mathbf{G} = \mathbf{U}\mathbf{S}\mathbf{V}^\mathsf{T}$ and $\mathbf{S} = \begin{bmatrix} \mathbf{S}_1 \\ \mathbf{0} \end{bmatrix}$ where $\mathbf{S}_1$ is the $N \times N$ submatrix, then the solution to Eq. (23) is:

$$\mathbf{E} = \begin{bmatrix} \sqrt{\alpha}\mathbf{A}\mathbf{V}\mathbf{S}_1^{-1} & \mathbf{0} \end{bmatrix} \mathbf{U}^\mathsf{T}. \tag{24}$$

Note that if we replace the Frobenius norm $||\mathbf{E}||_F$ in Eq. (23) by $\mathrm{rank}(\mathbf{E})$, Eq. (24) is still a solution.

D. Further Discussion

For the underdetermined cases of both LCFV1 and LCFV2, we need to compute the singular value decomposition of matrix $\mathbf{G}$. For numerical stability, it is a good practice to discard the columns of matrix $\mathbf{S}$ and $\mathbf{V}$ that correspond to very small singular values (*e.g.* smaller than 0.01) of $\mathbf{G}$.

One big difference between LCFV1 and LCFV2 is that the transformation matrix $\mathbf{M}$ of LCFV2 is not a square matrix. It adds L additional dimensions to the feature space, thus LCFV2 may seem a little "hacking" compared with LCFV1. As will be shown in Section IV, LCFV2 usually brings more performance improvement than LCFV1, which can also be expected.

IV. Experiments

In this paper, we focus on the comparison between original Fisher vectors and our LCFV method, thus we simplify the experiments instead of targeting at the state-of-the-art performance. We evaluate both Fisher vectors and LCFV on three well-known datasets: the fifteen scene categories dataset [15], the Graz-02 dataset [16], and a subset of the Corel Photo Gallery [2]. On each dataset, we take a number of images from each category to form a training set, and use a similar collection as the testing set. First, we extract SIFT features for each image, reduce the dimension of the SIFT features using PCA, learn a Gaussian mixture model on the low dimensional features, and represent each image with a Fisher vector. Then using the training set, we compute the label comparison matrix $\mathbf{C}$ and run the LCFV algorithm (using LCFV1 and LCFV2 respectively) to learn the transformation matrix $\mathbf{M}$, and train a linear SVM on the LCFV. In the testing stage, we use the learned matrix $\mathbf{M}$ to compute the LCFV for testing images, and use the learned SVM to classify these images. The classification accuracy values using traditional Fisher vectors, LCFV1 and LCFV2 are recorded. This experiment pipeline is repeated using different PCA dimensions, different number of Gaussian mixture components, different values of α (in a log scale), and different training-testing partitions of the dataset (we run 10 random partitions for each configuration).

For simplicity, when we compute Fisher vectors in our experiments, we only use the gradient with respect to the mean vectors of the GMM, without using the gradient with respect to the variance vectors. Thus the Fisher vector dimension M

is simply equal to the PCA dimension times the number of Gaussian mixture components.

Again, we emphasize that these experiments are simplified to compare LCFV with traditional Fisher vector method. There are many ways to improve the experiments to achieve state-of-the-art classification performance, including: using more features such as HOG [17] and LBP [18]; computing LCFV on a spatial pyramid instead of on the entire image; using the gradient with respect to the variance vectors when computing Fisher vectors.

A. Fifteen Scene Categories Dataset

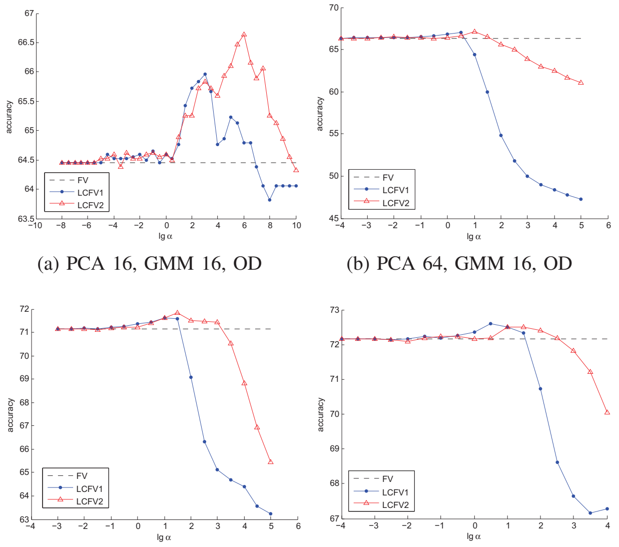
The fifteen scene categories dataset [15] has 200 to 400 gray images for each category, and the average size of an image is 300 × 250 pixels. For training, we take 100 images from each category, thus $N = 1500$ in this case. In Fig. 1, we show the classification accuracy of FV, LCFV1 and LCFV2 using different PCA dimensions and different number of Gaussian mixture components. In each plot, we show the classification accuracy of LCFV1 and LCFV2 using different values of α in a log scale. We can observe that when α is very small, there is almost no difference between LCFV and traditional Fisher vectors; when α is too large, lots of information in the traditional Fisher vectors is lost, and the performance drops. Only when α lies in a reasonable range, the performance of LCFV will be better than traditional Fisher vectors.

One way to find a good value of α is to perform cross validation on the training set, and select the best α according to the cross validation. For example, we can further divide the training set into five subsets, and perform a five-fold cross validation. We find the best α for each fold, and take their mean value as the final α to apply to testing. Following this parameter tuning practice, we report the average classification accuracy on 10 independent runs using random training-testing partitions of the dataset in Table I. This practice is also used for experiments on other datasets in this paper.

From Table I, we can see that the classification performance of LCFV is better than traditional Fisher vectors. This improvement benefits from the supervised information that we have integrated into the training process of the Fisher vectors. Such benefits are computationally inexpensive: on a Mac machine with 2.4GHz Quad-Core Intel Xeon CPU, when PCA dimension is 16 and the number of Gaussian mixture components is 16 (overdetermined), the computation of $\mathbf{M}$ takes about 0.1 second using LCFV1 and 1 second using LCFV2; when PCA dimension is 64 and the number of Gaussian mixture components is 64 (underdetermined), the computation of $\mathbf{M}$ takes about 6 seconds using LCFV1 and 4 seconds using LCFV2.

B. Graz-02 Dataset

The Graz-02 dataset [16] has four categories: bike, person, car and a negative category. Each category has 300 to 500 color images, and the typical image size is 640 × 480 pixels. Classification on this dataset is more difficult due to its high intra-class variation. We take 100 images from each category for training ($N = 400$) and 200 images from each category for testing. Classification results are shown in Fig. 2 and Table II. We can see that LCFV again has better performance than traditional Fisher vectors.



(a) PCA 16, GMM 16, OD

(b) PCA 64, GMM 16, OD

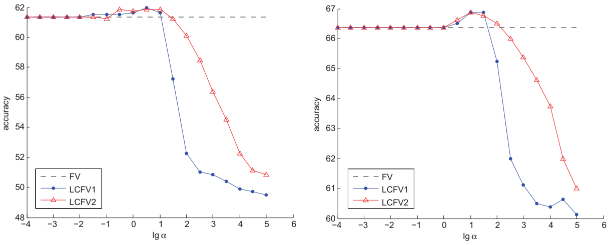
(c) PCA 64, GMM 64, UD

(d) PCA 64, GMM 128, UD

Fig. 1: Evaluation results on the fifteen scene categories dataset using different values of α. (OD: overdetermined, UD: underdetermined)

TABLE I: Average classification accuracy (%) of ten runs with tuned α on the fifteen scene categories dataset.

PCA dim.	# of GMM	FV	LCFV1	LCFV2
16	16	65.71	66.34	**66.90**
16	64	70.27	70.46	**70.52**
16	128	71.76	71.85	**71.89**
64	16	66.93	67.26	**67.37**
64	64	72.03	**72.40**	72.39
64	128	72.86	73.22	**73.22**



(a) PCA 64, GMM 16, UD

(b) PCA 64, GMM 64, UD

Fig. 2: Evaluation results on the Graz-02 dataset using different values of α.

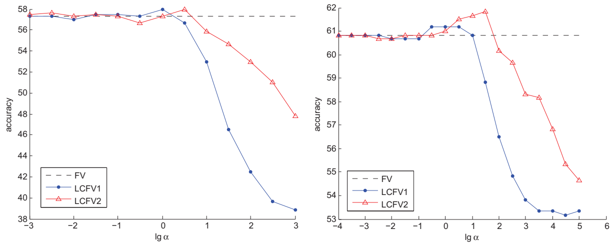
C. Corel Images

The last experiment is an evaluation using a subset of the Corel Photo Gallery [2]. We take twelve categories from the Corel dataset, where each category has 100 color images, and the typical size of an image is 120 × 80 pixels. These twelve categories are selected such that they are distinguishable by a human but do not have trivial clues such as pure background

TABLE II: Average classification accuracy (%) of ten runs with tuned α on the Graz-02 dataset.

PCA dim.	# of GMM	FV	LCFV1	LCFV2
64	16	62.61	62.96	**63.08**
64	64	67.11	67.38	**67.43**
64	128	68.10	68.28	**68.43**

color. The categories are: castle, bonsai, ship, train, flower, mushroom, forests, waterfall, butterfly, fish, wolf and woman. We take 50 images from each category for training ($N = 600$) and 50 images from each category for testing. Classification results are shown in Fig. 3 and Table III. Again, LCFV performs better than traditional Fisher vectors.



(a) PCA 64, GMM 16, UD

(b) PCA 64, GMM 64, UD

Fig. 3: Evaluation results on Corel images using different values of α.

TABLE III: Average classification accuracy (%) of ten runs with tuned α on Corel images.

PCA dim.	# of GMM	FV	LCFV1	LCFV2
64	16	56.27	56.58	**56.70**
64	64	60.53	60.77	**60.87**
64	128	61.68	62.03	**62.10**

V. Conclusions

In this paper, we have introduced the label consistent Fisher vector (LCFV) method, which is a supervised extension of the traditional Fisher vector method. LCFV is based on pairwise label comparison in the training set and solves for a transformation matrix which is applied on Fisher vectors. Our method is very straightforward and computationally efficient. Evaluated on three public datasets, we have shown that LCFV improves the classification performance of traditional Fisher vectors. One limitation of our method is that the classification performance is sensitive to the parameter α. However, the parameter can be tuned by cross validation on the training data. Although in this paper we only present experiments on scene classification and object recognition problems, it is very promising to apply our method on other problems such as digit recognition and style categorization as future work.

Appendix A
Two Useful Theorems

In order to show that the kernel $\widetilde{K}(\cdot,\cdot)$ defined in Eq. (8) is a valid kernel, we use the following theorem:

Theorem A.1. *An $N \times N$ label comparison matrix $\mathbf{C} = [C_{i,j}]$ where $C_{i,j} = \delta(c_i = c_j)$ is positive semi-definite.*

Proof: Since $\mathbf{C}$ is a label comparison matrix, we can re-order the class labels c_i such that the same labels are clustered together. This corresponds to applying a series of row-switching elementary operation matrices $\mathbf{R}_1, \dots, \mathbf{R}_r$ on $\mathbf{C}$ to make it a block-wise diagonal matrix $\widetilde{\mathbf{C}}$:

$$\begin{aligned} \mathbf{C} &= \mathbf{R}_r \cdots \mathbf{R}_1 \widetilde{\mathbf{C}} \mathbf{R}_1 \cdots \mathbf{R}_r \\ &= \mathbf{R} \widetilde{\mathbf{C}} \mathbf{R}^\mathsf{T}, \end{aligned} \tag{25}$$

where $\mathbf{R} = \mathbf{R}_r \cdots \mathbf{R}_1$. If there are L classes in total, and the size of class i is N_i, then $\widetilde{\mathbf{C}}$ has L blocks, and each block is an all-ones submatrix. Let $\mathbf{1}_{N_i} = [1, \dots, 1]^\mathsf{T}$ denote the $N_i \times 1$ all-ones column vector, then we can rewrite $\widetilde{\mathbf{C}}$ as:

$$\widetilde{\mathbf{C}} = \begin{bmatrix} \mathbf{1}_{N_1}\mathbf{1}_{N_1}^\mathsf{T} & & \\ & \ddots & \\ & & \mathbf{1}_{N_L}\mathbf{1}_{N_L}^\mathsf{T} \end{bmatrix}. \tag{26}$$

Now given an arbitrary $N \times 1$ column vector $\mathbf{x}$, let $\hat{\mathbf{x}} = \mathbf{R}^\mathsf{T}\mathbf{x}$, and we partition $\hat{\mathbf{x}}$ according to the block sizes of $\widetilde{\mathbf{C}}$:

$$\hat{\mathbf{x}} = \begin{bmatrix} \hat{\mathbf{x}}_1 \\ \vdots \\ \hat{\mathbf{x}}_L \end{bmatrix}. \tag{27}$$

We have

$$\begin{aligned} \mathbf{x}^\mathsf{T}\mathbf{C}\mathbf{x} &= \mathbf{x}^\mathsf{T}\mathbf{R}\widetilde{\mathbf{C}}\mathbf{R}^\mathsf{T}\mathbf{x} \\ &= \hat{\mathbf{x}}^\mathsf{T}\widetilde{\mathbf{C}}\hat{\mathbf{x}} \\ &= \sum_{i=1}^{L} \hat{\mathbf{x}}_i^\mathsf{T}\mathbf{1}_{N_i}\mathbf{1}_{N_i}^\mathsf{T}\hat{\mathbf{x}}_i \\ &= \sum_{i=1}^{L} ||\mathbf{1}_{N_i}^\mathsf{T}\hat{\mathbf{x}}_i||_2^2 \\ &\geq 0. \end{aligned} \tag{28}$$

Thus $\mathbf{C}$ is positive semi-definite. ■

To find the solution to Eq. (17), we need the following theorem:

Theorem A.2. *Let $\mathbf{\Lambda}$ be an $M \times M$ diagonal matrix with diagonal entries $\lambda_1, \lambda_2 \dots, \lambda_M$, where $\lambda_i > 0$ for $i = 1, 2, \dots, M$. If $\mathbf{U}$ is the $M \times M$ unitary matrix that minimizes $||\mathbf{\Lambda} - \mathbf{U}||_F$, then $\mathbf{U} = \mathbf{I}$.*

Proof: Let e_i be the ith column of the $M \times M$ identity matrix $\mathbf{I}$, u_i be the ith column of $\mathbf{U}$, and u_{ij} be the jth entry of u_i. Now we have $u_i^\mathsf{T} u_i = 1$, and

$$\begin{aligned}
||\mathbf{\Lambda} - \mathbf{U}||_F^2 &= \sum_{i=1}^{M} ||\lambda_i e_i - u_i||_2^2 \\
&= \sum_{i=1}^{M} (\lambda_i e_i^\mathsf{T} - u_i^\mathsf{T})(\lambda_i e_i - u_i) \\
&= \sum_{i=1}^{M} (\lambda_i^2 - \lambda_i e_i^\mathsf{T} u_i - \lambda_i u_i^\mathsf{T} e_i + u_i^\mathsf{T} u_i) \\
&= \sum_{i=1}^{M} (\lambda_i^2 - 2\lambda_i u_{ii} + 1) \\
&= M + \sum_{i=1}^{M} \lambda_i^2 - 2 \sum_{i=1}^{M} \lambda_i u_{ii}.
\end{aligned}$$

Since $\lambda_i > 0$, minimizing $||\mathbf{\Lambda} - \mathbf{U}||_F$ is equivalent to maximizing each u_{ii}. Since $u_{ii} \leq 1$, the solution is simply $u_{ii} = 1$ for $i = 1, 2, \ldots, M$. Because $u_{ii} = 1$ is equivalent with $u_i = e_i$, we have $\mathbf{U} = \mathbf{I}$. ■

References

[1] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *CVPR*, vol. 2. IEEE, 2005, pp. 524–531.

[2] J. Z. Wang, J. Li, and G. Wiederhold, "Simplicity: Semantics-sensitive integrated matching for picture libraries," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947–963, 2001.

[3] N. Murray, L. Marchesotti, and F. Perronnin, "Ava: A large-scale database for aesthetic visual analysis," in *CVPR*. IEEE, 2012, pp. 2408–2415.

[4] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[5] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *CVPR*. IEEE, 2007, pp. 1–8.

[6] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *ECCV*. Springer, 2010, pp. 143–156.

[7] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the fisher vector: Theory and practice," *International Journal of Computer Vision*, pp. 1–24, 2013.

[8] T. Jaakkola, D. Haussler *et al.*, "Exploiting generative models in discriminative classifiers," *NIPS*, pp. 487–493, 1999.

[9] H. Jégou, F. Perronnin, M. Douze, C. Schmid *et al.*, "Aggregating local image descriptors into compact codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1704–1716, 2012.

[10] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier, "Large-scale image retrieval with compressed fisher vectors," in *CVPR*. IEEE, 2010, pp. 3384–3391.

[11] M. Douze, A. Ramisa, and C. Schmid, "Combining attributes and fisher vectors for efficient image retrieval," in *CVPR*. IEEE, 2011, pp. 745–752.

[12] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *CVPR*, vol. 1, 2013.

[13] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csurka, "Assessing the aesthetic quality of photographs using generic image descriptors," in *ICCV*. IEEE, 2011, pp. 1784–1791.

[14] L. Maaten, "Learning discriminative fisher kernels," in *ICML*, 2011, pp. 217–224.

[15] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, vol. 2. IEEE, 2006, pp. 2169–2178.

[16] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer, "Generic object recognition with boosting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 416–431, 2006.

[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, vol. 1. IEEE, 2005, pp. 886–893.

[18] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.