TOWARDS COMPACT VISUAL DESCRIPTOR VIA DEEP FISHER NETWORK WITH BINARY EMBEDDING

Jianqiang Qian *Xianming Lin* *Hong Liu* *Youming Deng* *Rongrong Ji*⋆

Fujian Key Laboratory of Sensing and Computing for Smart City,
School of Information Science and Engineering, Xiamen University, 361005, China
{rrji, linxm}@xmu.edu.cn, {jqqian, lynnliu, youmingdeng}@stu.xmu.edu.cn

ABSTRACT

Fisher Vector (FV) has been widely used to aggregate the local descriptors of an image into a global representation in large-scale image retrieval. However, FV has limited learning capability and its parameters are mostly fixed after constructing the codebook, which is inflexible and cannot be trained jointly with deep networks. Moreover, the high dimension of FV makes it difficult to be applied in scenarios compact descriptors are needed. In this paper, we propose a novel compact image description scheme based on Fisher network with binary embedding to solve the large-scale image retrieval problem, which consists of two components: a Fisher encoder component and a binary embedding component. Concretely, the Fisher encoder is a trainable neural network functions as the traditional FV, which aggregates the local descriptors into a global representation. And the binary encoder embeds the high-dimensional FV to a binary vector, which outputs the compact global binary descriptor. To learn such a descriptor, we further introduce a novel and effective loss function, in which maximum margin criterion is exploited to minimize the distances of positive pairs, as well as maximizing the distances of negative pairs. Extensive experiments performed on MPEG-7 CDVS benchmarks and ILSVR2010 demonstrate that the proposed framework can achieve very superior performance over the state-of-the-art methods.

Index Terms— Fisher network, Binary coding, CDVS, Large-scale image retrieval

1. INTRODUCTION

Large-scale visual search has attracted extensive research attention due to its wide application prospects. Recent advances of large-scale visual search have paid more focus on the aggregation of local descriptors of an image into a discriminative global descriptor [1, 2, 3, 4, 5, 6]. Among the existing endeavors, Fisher Vector (FV) is widely regarded as among the most powerful ones, whose main advantages lie in: (1) it captures the first- and second-order statistics of the image. (2) it only relies on a small visual vocabulary and is very fast to compute. In a standard pipeline, FV is extracted via first extracting a set of local descriptors, then encoding those descriptors with a Gaussian Mixture Model (GMM) to output an image-level signature.

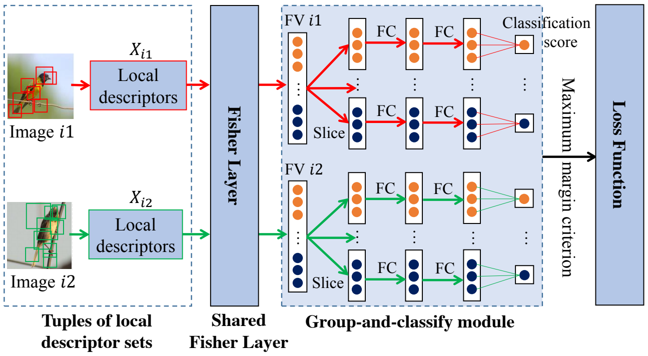


Fig. 1. The architecture of deep Fisher binary embedding neural network. The input to the proposed architecture is in the form of tuples, *i.e.* (X_{i1}, X_{i2}), X_{i1} and X_{i2} are local feature sets extracted from an image pair respectively. Through the proposed architecture, the tuples of local feature sets are first aggregated into a tuple of FV by the Fisher Layer. Then, each FV is converted to binary codes by a group-and-classify module. After that, these two binary codes are used in a loss, which aims to minimize the distances of positive pairs and maximize the distances of negative pairs.

However, FV is typically very high dimensional, which restricted its usage in resource limited application such as mobile or embedded scenarios, where the descriptor compactness is of fundamental importance. To this end, recent works mainly focus on embedding high-dimensional aggregated descriptors into low dimensional ones, which can be further divided into two categories: Quantization and Hashing. Jégou *et al.* used product quantization for large-scale approximately nearest neighbor search (ANN) [7]. And hashing, *a.k.a.* binary code learning, has also widely been studied for ANN search in large-scale datasets [8, 9, 10, 11, 12, 13, 14]. To that effect, a natural solution is to do binary code learning (embedding) over the learned high-dimensional Fisher de-

⋆ Corresponding Author

978-1-5386-1737-3/18/$31.00 ©2018 IEEE

scriptor. However, both schemes cannot be directly applied to FV model, whose parameters are fixed once the codebook is learned. As a result, the learning of FV departed from the subsequent binary code learning, which makes the entire process suboptimal.

In this paper, we propose an end-to-end trainable Fisher Network that integrates the FV learning with binary embedding in a joint optimization framework. First, we mimic the traditional FV with a network representation, *i.e.* Fisher network [1]. Then, binary embedding is conducted over the extracted deep Fisher feature to produce a very compact binary code. Notably, both components are jointly learned, enabling a reinforcement among them towards the respective optimal.

As shown in Fig. 1, the proposed architecture consists of three building blocks: (1) *the Fisher Layer*, it is a learnable module based on the traditional FV encoding process, which encodes the local descriptors to the very high-dimensional FV. Such a layer allows back-propagating error deviations as well as flexibly optimizing the Gaussian codebook. (2) *the group-and-classify module*, it splits the original high-dimensional FV into S sub-vectors. Each sub-vector performs a non-linear transformation by two fully-connected layers, and is quantized to one bit by a linear classifier layer.(3) *the loss function*, it exploits a maximum margin criterion to minimize the distances of positive pairs, as well as to maximize the distance of negative pairs.

We have conducted extensive experiments on two large-scale benchmark datasets, *i.e.* the **MPEG-7 CDVS** dataset with one million distractor images from Flickr, and the large-scale **ILSVR2010** dataset containing 1.2 million images from $1,000$ categories. The quantitative results show that the proposed method could get superior results over several state-of-the-art methods.

2. RELATED WORK

FV is a powerful local descriptor aggregation scheme for global image representation. Comparing to the widely used Vector of Locally Aggregated Descriptors (VLAD) [2], FV contains more discriminative information. FV is also faster than the widely used Bag-of-Words (BoW) descriptor [15], since it uses a much smaller visual vocabulary.

There exist several improved versions of FV [1, 16]. For instance, Tang *et al.* proposed a Fisher Net [1] to train the parameters of the FV model jointly with the other neural networks. But the local features are extracted by using CNN rather than SIFT, which is quantitatively shown to be less discriminative against photoing variations [17, 18]. Sydorov *et al.* proposed deep Fisher kernels [16] and used an iterative algorithm to separately learn the parameters of FV and SVM. Different from their method, we decompose FV into a series of neural layers and insert them to a network, which can learn the parameters of both the FV and fully-connected layers with the standard back-propagation. In addition, due to the high-dimensional FV representation, the number of parameters in deep Fisher kernels is very large, which makes the training time-consuming. To solve this problem, the group-and-classify module is proposed to divide the high dimensional vector into a series of sub-vectors, so that the parameters in each subspace can be trained easily with low dimension.

Moreover, aiming at compact descriptor for large-scale image retrieval, several recent works have been proposed to encode high-dimension data to binary codes. For instance, PQ [7] is a popular and effective method. Interactive Quantization (ITQ) is also widely used in the literature, which adopts a rotation to balance the variance of high-dimensional data. Hashing is also popular in compact feature representation [8, 9, 10, 11, 12]. However, most hashing-based approaches are less effective comparing to local descriptor aggregation schemes like FV and VLAD. Recently, Liu *et al.* used a bank of linear classifiers to project high-dimensional data to binary codes effectively [19]. It is worth to note that, the work in [19] uses kernel function to separate linear-inseparable data, which is computationally expensive.

FV has been recently adopted for the Compact Descriptor for Visual Search (CDVS) MPEG standard [20]. CDVS consists of two blocks: (1) retrieving a subset of images from the large-scale database. (2) using Geometric Consistency Checks (GCC) [21] for finding relevant images with high precision. In the first block, a scalable compressed Fisher Vector (SCFV) is proposed to aggregate the local descriptors. In particular, SCFV uses scalar quantization to quantize the resulting high-dimensional FV to binary codes, as well as selecting a subset of Gaussian components (with an average size of 304, 384, 404, 1117 bytes for different specified bit rates) according to their standard deviations [20].

3. THE PROPOSED FRAMEWORK

Let $X_i = \{\mathbf{x}_{ij}\}_{j=1}^{m_i}$, to be a set of local descriptors (*e.g.* SIFT) extracted from an image, where $\mathbf{x}_{ij} \in \mathbb{R}^D$, and m_i is the number of local descriptors. Our target is to learn a mapping function $\mathcal{F} : X_i \to \{0, 1\}^S$, such that an input image can be encoded into S-bit binary codes by the stages of both FV encoding and binary embedding. The proposed framework merits in a joint learning between Fisher network based descriptor aggregation, and and the classifier based binary embedding. The proposed architecture contains three building blocks: (1) the Fisher Layer, (2) the group-and-classify module, (3) the loss function. We depict the details as below:

3.1. The Fisher Layer

We first revisit the traditional FV $\phi(X_i)$, which encodes a set of local descriptors X_i extracted from an image by fitting a K–component Gaussian Mixture Model (GMM) $u_\lambda(\mathbf{x}) = \sum_{k=1}^{K} w_k u_k(\mathbf{x})$ to the local descriptors. FV encodes the derivatives of log-likelihood with respect to its parameter set [22], denoted as $\lambda = \{w_k, \mu_k, \mathbf{\Sigma}_k, k = 1, 2, \dots, K\}$, where $w_k \in \mathbb{R}$, $\mu_k \in \mathbb{R}^{D\times 1}$, $\mathbf{\Sigma}_k \in \mathbb{R}^{D\times D}$. $\{w_k, \mu_k, \mathbf{\Sigma}_k\}$ are the mixture weight, mean vector and covariance matrix of the GMM model, and $\mathbf{\Sigma}_k = diag(\sigma_k^2)$, $\sigma_k \in \mathbb{R}^{D\times 1}$.

However, the parameters of FV are fixed once the codebook is learned, which is inflexible and suboptimal scheme, while considering the fact that the binary embedding is learned in a separated manner. In the proposed scheme, we combine the training of Fisher Layer [1] with the binary encoder, leading to an end-to-end framework that produces optimal binary codes for each image. In particular, the proposed Fisher Layer makes two simplifications to the original FV: (1) all GMM components have equivalent weights; (2) the covariance matrices of all the Gaussian components share the same determinant, so that the k-th Gaussian distribution $u_k(\mathbf{x}_{ij})$ can be written as:

$$u_k(\mathbf{x}_{ij}) = \frac{1}{(2\pi)^{D/2}} \exp\big\{-\frac{1}{2}(\mathbf{x}_{ij} - \mu_k)^T \mathbf{\Sigma}_k^{-1}(\mathbf{x}_{ij} - \mu_k)\big\}. \tag{1}$$

The gradients of a single descriptor $\mathbf{x}_{ij}$ *w.r.t* the parameters μ_k and σ_k of the simplified GMM can be written as:

$$\mathcal{G}_{\mu_k}^{\mathbf{x}_{ij}} = \gamma_j(k)\big[\mathbf{w}_k \odot (\mathbf{x}_{ij} + \mathbf{b}_k)\big], \tag{2}$$

$$\mathcal{G}_{\sigma_k}^{\mathbf{x}_{ij}} = \gamma_j(k)\big[(\mathbf{w}_k \odot (\mathbf{x}_{ij} + \mathbf{b}_k))^2 - 1\big], \tag{3}$$

where $\odot$ is an element-wise product operation, $\mathbf{w}_k = 1/\sigma_k$, $\mathbf{b}_k = -\mu_k$. $\mathbf{w}_k$ and $\mathbf{b}_k$ are sets of learnable parameters for each Gaussian component k. $\gamma_j(k)$ is the posterior probability, and can be written as follows:

$$\gamma_j(k) = \frac{u_k(\mathbf{x}_{ij})}{\sum_{n=1}^{K} u_n(\mathbf{x}_{ij})} \tag{4}$$

$$= \frac{\exp\{-\frac{1}{2}(\mathbf{w}_k \odot (\mathbf{x}_{ij} + \mathbf{b}_k))^T(\mathbf{w}_k \odot (\mathbf{x}_{ij} + \mathbf{b}_k))\}}{\sum_{n=1}^{K} \exp\{-\frac{1}{2}(\mathbf{w}_n \odot (\mathbf{x}_{ij} + \mathbf{b}_n))^T(\mathbf{w}_n \odot (\mathbf{x}_{ij} + \mathbf{b}_n))\}}.$$

For any single local descriptor $\mathbf{x}_{ij}$, its output of Fisher Layer can be denoted as:

$$\varphi(\mathbf{x}_{ij}) = \Big[\mathcal{G}_{\mu_1}^{\mathbf{x}_{ij}\,T}, \dots, \mathcal{G}_{\mu_K}^{\mathbf{x}_{ij}\,T}, \mathcal{G}_{\sigma_1}^{\mathbf{x}_{ij}\,T}, \dots, \mathcal{G}_{\sigma_K}^{\mathbf{x}_{ij}\,T}\Big]^T. \tag{5}$$

As a result, the final FV $\phi(X_i)$ of the local feature set X_i from an image is the mean-pooling of all local representations in X_i, *i.e.* $\phi(X_i) = \frac{1}{m_i}\sum_{j=1}^{m_i} \varphi(\mathbf{x}_{ij})$. All the operations in Fisher Layer are differentiable and the parameters $\mathbf{w}_k$ and $\mathbf{b}_k$ can be derived via back-propagation.

3.2. The Group-and-Classifier Module (GCM)

After obtaining the high-dimensional FV from the Fisher Layer, we further propose a Group-and-Classify Module (GCM) to encode the global feature into binary codes.

As shown in Fig. 2, the GCM firstly divides the output of the Fisher Layer into S sub-vectors, each of which is a continuous slice within a Gaussian component. This dividing is natural and can preserve the intrinsic data structure. Secondly, each sub-vector is non-linearly transformed by two fully-connected layers and is subsequently quantized to one bit by a linear classifier layer. The linear classifier layer adopts a linear classifier $f(\mathbf{x}_{(p)}) = \mathbf{w}_{(p)}^T \mathbf{x}_{(p)} - b_{(p)}$, where $\mathbf{w}_{(p)}$ is the hyperplane, $b_{(p)}$ is the bias, and p present the model of p-th slice. Consequently, positive pairs are positioned in the same side of hyperplane, while negative pairs are expected to be placed at different sides of the hyperplane. Note that the non-linear transformation is necessary, since it can transform the linearly-inseparable data into a space where a linear classifier works.

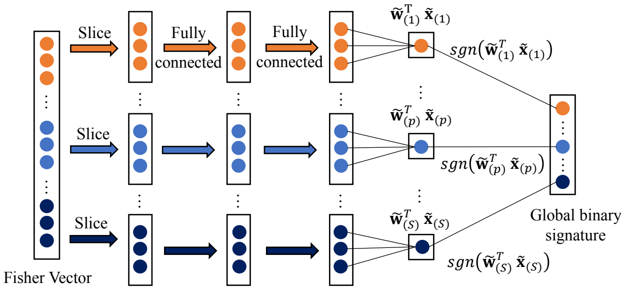

Slice
Fully connected
Fully connected
$\tilde{\mathbf{w}}_{(1)}^T\ \tilde{\mathbf{x}}_{(1)}$
$sgn(\tilde{\mathbf{w}}_{(1)}^T\ \tilde{\mathbf{x}}_{(1)})$
Slice
$\tilde{\mathbf{w}}_{(p)}^T\ \tilde{\mathbf{x}}_{(p)}$
$sgn(\tilde{\mathbf{w}}_{(p)}^T\ \tilde{\mathbf{x}}_{(p)})$
Slice
$\tilde{\mathbf{w}}_{(S)}^T\ \tilde{\mathbf{x}}_{(S)}$
$sgn(\tilde{\mathbf{w}}_{(S)}^T\ \tilde{\mathbf{x}}_{(S)})$
Global binary signature
Fisher Vector

Fig. 2. The group-and-classify module (GCM). Fisher Vector is split into S sub-vectors. Each sub-vector is non-linearly transformed by two fully connected layers, and projected to one bit. The final global binary signature is concatenation of the bits.

We denote $\tilde{\mathbf{x}}_{(p)} = [\mathbf{x}_{(p)}; -1]$, $\tilde{\mathbf{w}}_{(p)} = [\mathbf{w}_{(p)}; b_{(p)}]$, then the linear classifier can be rewritten as $f_p(\tilde{\mathbf{x}}_{(p)}) = \tilde{\mathbf{w}}_{(p)}^T \tilde{\mathbf{x}}_{(p)}$, where $\tilde{\mathbf{w}}_{(p)}$ is the coefficient of the classifier for the p-th sub-vector. And the binary code for each sub-vector can be acquired as follows:

$$h_p(\tilde{\mathbf{x}}_{(p)}) = \operatorname{sgn}(\tilde{\mathbf{w}}_{(p)}^T \tilde{\mathbf{x}}_{(p)}), \tag{6}$$

where $\operatorname{sgn}(\cdot)$ is the sign function that returns 1 if $\operatorname{sgn}(\cdot) > 0$ and -1 otherwise.

Finally, we concatenate the the S binary bits together to form the final binary codes for the high-dimensional FV:

$$[\operatorname{sgn}(\tilde{\mathbf{w}}_{(1)}^T \tilde{\mathbf{x}}_{(1)}), \dots, \operatorname{sgn}(\tilde{\mathbf{w}}_{(S)}^T \tilde{\mathbf{x}}_{(S)})].$$

3.3. The Loss Function

To achieve better binary codes, we use Hinge loss to minimize the distances of positive pairs, and to maximize the distances of negative pairs in the learned subspace. Following the linear classifier layer of each branch in the network, we exploit a maximum margin criterion for positive and negative pairs of sub-vectors in the p-th slice, $p \in \{1, 2, \dots, S\}$. For a pair of sub-vectors in the p-th slice, the maximum margin criterion can be written as follows:

$$\ell_i\big(\tilde{\mathbf{w}}_{(p)}^T \tilde{\mathbf{x}}_{i1(p)} \cdot \tilde{\mathbf{w}}_{(p)}^T \tilde{\mathbf{x}}_{i2(p)}\big) > 1,\ p = 1, 2, ..., S, \tag{7}$$

where $\tilde{\mathbf{x}}_{i1(p)} = [\mathbf{x}_{i1(p)}; -1]$, $\mathbf{x}_{i1(p)}$ is the p-th sub-vector of the Fisher vector $\mathbf{x}_{i1}$, and $\mathbf{x}_{i1}$ belongs to the first image of the image pair. $\tilde{\mathbf{x}}_{i2(p)}$ follows the same criterion. ℓ_i is the pair-wise label, where $\ell_i = +1$ if $\mathbf{x}_{i1}$ and $\mathbf{x}_{i2}$ belong to the same class, $\ell_i = -1$ otherwise. S is the number of sub-vectors split from the whole vector. The maximum margin criterion means positive pairs are positioned in the same side of the hyperplane, while negative pairs are placed at different sides of the hyperplane.

For a mini-batch, all the image pairs is constrained to satisfy the maximum margin criterion in the p-th subspace. In the p-th subspace, similar to SVM, the optimization problem can be written as follows:

$$\begin{aligned} &\min_{\tilde{\mathbf{w}}_{(p)}} \ \frac{1}{2}\|\tilde{\mathbf{w}}_{(p)}\|^2, \\ &s.t. \ \ \ell_i\big(\tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i1(p)} \cdot \tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i2(p)}\big) > 1, \ i = 1, 2, \ldots, N, \end{aligned} \tag{8}$$

where N is the mini-batch size.

We further integrate the Hinge loss into Eq. 8, which can be rewritten as follows:

$$\begin{aligned} L(\tilde{\mathbf{w}}_{(p)}) &= \frac{1}{2}\|\tilde{\mathbf{w}}_{(p)}\|^2 \\ &+ \lambda \sum_{i=1}^{N} \max\big(0, 1 - \ell_i(\tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i1(p)} \cdot \tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i2(p)})\big), \end{aligned} \tag{9}$$

where λ is the balance parameter to control the importances of the two terms. We denote $L_i(\tilde{\mathbf{w}}_{(p)}) = \max(0, 1 - \ell_i(\tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i1(p)} \cdot \tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i2(p)}))$. Taking the derivative of $L_i(\tilde{\mathbf{w}}_{(p)})$ with respect to $\tilde{\mathbf{w}}_{(p)}$, the gradient of $L_i(\tilde{\mathbf{w}}_{(p)})$ is:

$$\begin{aligned} &\triangledown L_i(\tilde{\mathbf{w}}_{(p)}) = \\ &\begin{cases} 0, \ \text{if } 1 - \ell_i(\tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i1(p)} \cdot \tilde{\mathbf{w}}_{(p)}^T\tilde{\mathbf{x}}_{i2(p)}) \leq 0, \\ -\ell_i(\tilde{\mathbf{x}}_{i1(p)} \cdot \tilde{\mathbf{x}}_{i2(p)}^T + \tilde{\mathbf{x}}_{i2(p)} \cdot \tilde{\mathbf{x}}_{i1(p)}^T)\tilde{\mathbf{w}}_{(p)}, \ \text{otherwise.} \end{cases} \end{aligned} \tag{10}$$

By summing up all pair-wise loss of the sub-vectors in all subspaces, the overall loss function with respect to each mini-batch can be written as follows:

$$\hat{L}(\tilde{\mathbf{w}}_1, ..., \tilde{\mathbf{w}}_S) = \sum\nolimits_{p=1}^{S} L(\tilde{\mathbf{w}}_{(p)}). \tag{11}$$

4. EXPERIMENTS

Two experiments are performed on two large-scale image retrieval benchmarks, *i.e.* **MPEG-7 CDVS** [23], and **ILSVR2010** [24].

As for the first experiment performed on the MPEG-7 CDVS dataset, we compare our method with the SCFV method used in the CDVS standard [20]. The CDVS dataset consists of 5 classes: *Graphics, Paintings, Video Frames, Common Objects and Buildings*, which includes $8,314$ query images, $18,840$ reference images and a distractor set of 1 million images from Flickr. Our experiments are performed on three largest subsets: Graphics (1500 queries), Common Objects (2550 queries), and Buildings (3499 queries). The remaining images are used as the gallery database.

Table 1. Recall@500 vs. different descriptor length over the graphic, object and building datasets, combines with the distractor set Flickr.

method	Recall@500 (%)			
	Length (bytes)	Graphics	Common Objects	Buildings
SCFV	304	88.9	85.8	66.6
	384	91.9	88.1	68.2
	404	92.5	90.2	70.1
	1117	94.1	92.5	72.3
The proposed method	304	91.5	87.3	69.3
	384	93.6	91.4	72.2
	404	94.5	92.8	74.5
	1117	96.0	94.9	76.1

In the CDVS image retrieval pipeline [20], the step of Geometric Consistency Checks (GCC) is computationally complex, which can only be performed on a small number of images. In contrast, the first step of global binary signature matching is extremely fast. Therefore, it is necessary for the relevant images to be presented in top returning after the first step. Hence, we evaluate the recall at several typical operating points, for example $R = 500$ after the first step of the retrieval pipeline.

For each image in the CDVS dataset with the 1 million distractors, we extract 128-d SIFT descriptor in patches of 16×16 around interest points detected by a Laplacian of Gaussian (LoG) detector [20]. To train the Fisher Layer, we use a random set of $100,000$ images in $1,000$ categories from ImageNet. The training set has no overlap with the query and gallery dataset used in the subsequent retrieval. The learning rate of the neural network is set to 0.001. The momentum and weight decay are set to 0.9 and 0.0005, respectively. The mini-batch size of image pairs is 32. And the number of Gaussian components of GMM is set to 512 according to the CDVS standard. To get the binary signature of the same size in the CDVS standard, we split the whole FV into S sub-vectors and quantize the FV with GCM.

Tab. 1 compares the proposed scheme to the SCFV at different descriptor length. It shows the retrieval results in terms of Recall@500 under different bit rates over three datasets. Comparing to SCFV, our method achieves very competitive results. For example, the recall is increased from 72.3% at $1,117$ bytes to 76.1% on Buildings and 85.8% at 304 bytes to 87.3% on Common Objects. Fig. 3 further shows the mAP comparison, where GCC includes a ratio test followed by a fast geometric estimation [21]. Comparing to SCFV, we achieve an mAP improvements of +3.72%, +4.61%, +4.48% on average for Graphics, Common Objects, and Buildings on different image descriptor lengths. The results have demonstrated the effectiveness of the proposed deep Fisher network.

As for the second group of experiments performed on

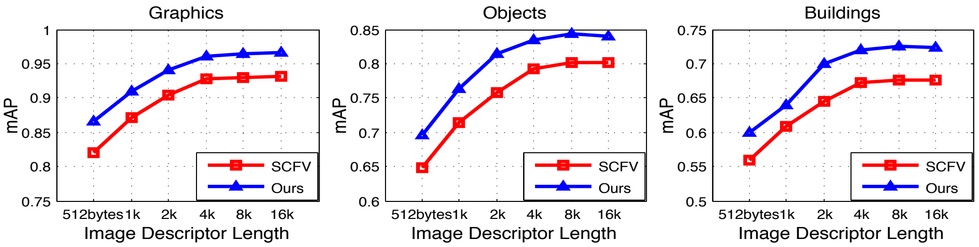
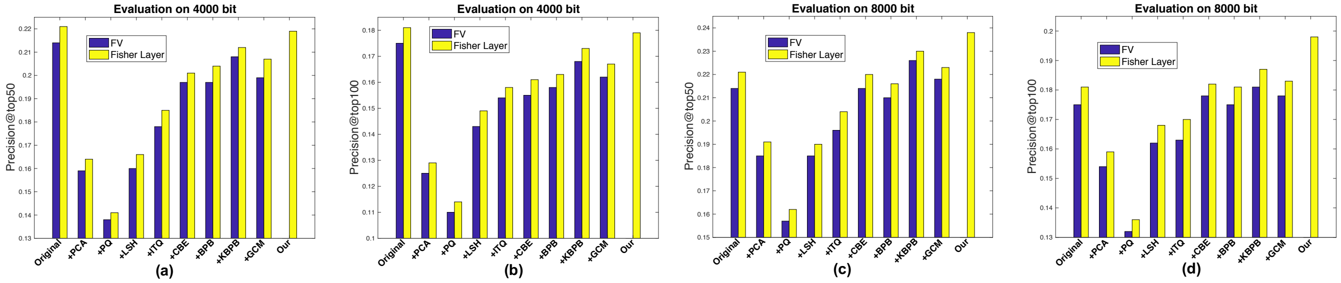


Fig. 3. Results of mAP of SCFV and our architecture over the CDVS framework on MPEG-7 CDVS dataset.



(a)

(b)

(c)

(d)

Fig. 4. Results of mAP of SCFV and our architecture over the CDVS framework on ILSVR2010 dataset.

ILSVR2010, a subset of ImageNet containing 1.2 million images from 1000 categories are used, we mainly compare the proposed end-to-end learning architecture to the typical two-stage encoding scheme. In this experiment, dense 128-d SIFT are used. We randomly select $1,000$ images as the query and the remaining as the database. Besides, from the database, 150K images are randomly selected for training and another 50K images are randomly selected for cross-validation.

We compare the proposed method with two alternative schemes: (1) replacing the Fisher Layer in the proposed framework with the traditional FV, which compares the retrieval performance of FV produced by the Fisher Layer and the traditional FV. (2) replacing the GCM with other dimension reduction methods [7, 11, 8, 12, 19] to compare the proposed end-to-end learning architecture with the traditional two-stage encoding.

First, to compare the retrieval performance of Fisher Layer and the traditional FV encoding method, we train the Fisher Layer independently using the triplet loss function. The triplets are randomly constructed based on the image labels. The learning rate of the Fisher Layer is set to 0.001. The momentum and weight decay are set to 0.9 and 0.0005 respectively. The mini-batch size of image pairs is set to 32, and the number of GMM components is set to 250.

Second, we embed these two kinds of FVs into a reduced feature space with eight coding methods, including two real-valued dimensionality reduction methods: Principal Component Analysis (PCA), Product Quantization (PQ) [7], and six binary encoding methods: Locality Sensitive Hashing (LSH) [11], Iterative Quantization (ITQ) [8], Circulant Binary Embedding (CBE) [12], Binary Projection Bank (BPB), Kernel Binary Projection Bank (KBPB) [19], and the Group-and-Classify Module (GCM). ITQ is used in the subspace to map the sub-vector to multiple bits. The GCM is trained independently to map the FV to binary codes. More specifically, for the traditional FV, it is directly fed into the GCM to train the group-and-classify network. For the FV produced by the Fisher Layer, to train the GCM independently, the learning rates of the pre-trained Fisher Layer and the GCM are set to 0 and 0.001, respectively.

Next, we train the proposed architecture end-to-end. Based on the independently trained Fisher Layer using the triplet loss function, we connect the GCM after the pre-trained Fisher Layer, and then fine-tune the whole neural network. The learning rates of the Fisher Layer and the GCM are set to 0.0001 and 0.001, respectively. In Fig. 4, we observe that the proposed method can achieve better performance for medium-dimensional binary codes compared to other methods. Besides, the FV produced by Fisher Layer also performs better than the traditional FV encoding method.

5. CONCLUSIONS

This paper has proposed a novel compact image description scheme based on Fisher network with binary embedding. The neural network maps the local descriptors of an image to medium-length compact global descriptor. In the proposed scheme, we combine the training of Fisher Layer with the binary encoder, leading to an end-to-end framework that produces optimal binary codes for each image. The proposed method has achieved better results compared with state-of-the-art methods for image retrieval. In recent years, convolutional neural networks (CNN) has achieved great success. In the future, we will try to exploit CNN to extract local features and construct an end-to-end system mapping an image to a global binary representations directly.

6. ACKNOWLEDGEMENT

This work is supported by the National Key R&D Program (No.2017YFC0113000, and No.2016YFB1001503), Nature Science Foundation of China (No.U1705262, No.61772443, and No.61572410), Post Doctoral Innovative Talent Support Program under Grant BX201600094, China Post-Doctoral Science Foundation under Grant 2017M612134, Scientific Research Project of National Language Committee of China (Grant No. YB135-49), and Nature Science Foundation of Fujian Province, China (No. 2017J01125).

7. REFERENCES

[1] P. Tang, X.-G. Wang, B.-G. Shi, X. Bai, W.-Y. Liu, and Z.-W. Tu, "Deep fishernet for object classification," *CoRR*, vol. abs/1608.00182, 2016.

[2] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proceedings of CVPR*, 2010.

[3] H. Jégou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *International Journal of Computer Vision*, vol. 87, no. 3, pp. 316–336, 2010.

[4] Y. Liu and F. Perronnin, "A similarity measure between unordered vector sets with application to image categorization," in *Proceedings of CVPR*, 2008.

[5] Z. Zhong, L. Zheng, D.-L. Cao, and S.-Z. Li, "Re-ranking person re-identification with k-reciprocal encoding," in *Proceedings of CVPR*, 2017.

[6] Z.-M. Luo, A. K. Mishra, A. Achkar, J. A. Eichel, S.-Z. Li, and P.-M. Jodoin, "Non-local deep features for salient object detection," in *Proceedings of CVPR*, 2017.

[7] H. Jégou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 117–128, 2011.

[8] Y.-C. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2916–2929, 2013.

[9] J.-P. Heo, Y. Lee, J.-F. He, S.-F. Chang, and S.-E. Yoon, "Spherical hashing," in *Proceedings of CVPR*, 2012.

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

[11] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proceedings of STOC*, 2002.

[12] F.-X. Yu, S. Kumar, Y.-C. Gong, and S.-F. Chang, "Circulant binary embedding," in *Proceedings of ICML*, 2014.

[13] H. Liu, R.-R. Ji, Y.-J. Wu, and F.-Y. Huang, "Ordinal constrained binary code learning for nearest neighbor search," in *Proceedings of AAAI*, 2017.

[14] R.-R. Ji, H. Liu, L.-J. Cao, D. Liu, Y.-J. Wu, and F.-Y. Huang, "Towards optimal manifold hashing via discrete locally linear embedding," *IEEE Transactions on Image Processing*, vol. PP, no. 99, pp. 1–1, 2017.

[15] J. Farquhar, S. Szedmak, H.-Y. Meng, and J. Shawe-Taylor, "Improving "bag-of-keypoints" image categorisation: Generative models and pdf-kernels," *Technical Report: University of Southampton*, 2005.

[16] V. Sydorov, M. Sakurada, and C. H. Lampert, "Deep fisher kernels - end to end learning of the fisher kernel GMM parameters," in *Proceedings of CVPR*, 2014.

[17] V. Chandrasekhar, J. Lin, O. Morère, H. Goh, and A. Veillard, "A practical guide to cnns and fisher vectors for image instance retrieval," *Signal Processing*, vol. 128, pp. 426–439, 2016.

[18] K. Yan, Y.-W. Wang, D.-W. Liang, T.-J. Huang, and Y.-H. Tian, "CNN vs. SIFT for image retrieval: Alternative or complementary?," in *Proceedings of MM*, 2016.

[19] L. Liu, M.-Y. Yu, and L. Shao, "Projection bank: From high-dimensional data to medium-length binary codes," in *Proceedings of ICCV*, 2015.

[20] L.-Y. Duan, V. Chandrasekhar, J. Chen, J. Lin, Z. Wang, T.-J. Huang, B. Girod, and W. Gao, "Overview of the MPEG-CDVS standard," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 179–194, 2016.

[21] S. Lepsøy, G. Francini, G. Cordara, and P. P. B. de Gusmao, "Statistical modelling of outliers for fast visual search," in *Proceedings of ICME*, 2011.

[22] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proceedings of ECCV*, 2010.

[23] M. J. Huiskes, B. Thomee, and M. S. Lew, "New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative," in *Proceedings of MIR*, 2010.

[24] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li, "Imagenet: A large-scale hierarchical image database," in *Proceedings of CVPR*, 2009.