Image Retrieval via Canonical Correlation Analysis

Kangdi Shi, Xiaohong Liu, *Student Member, IEEE*, Muhammad Alrabeiah, Xintong Guo,
Jie Lin, *Senior Member, IEEE*, Huan Liu, Jun Chen, *Senior Member, IEEE*

***Abstract*—Canonical Correlation Analysis (CCA) is a powerful multivariate statistical method. It can be used to find, for a given dimension, a projection pair that maximally captures the correlation between two target random vectors. This work introduces a CCA-based approach for image retrieval. It capitalizes on feature maps extracted from a pre-trained Convolutional Neural Network (CNN) and leverages basis vectors identified via CCA, in conjunction with an element-wise selection method based on the Chernoff information, to generate compact transformed image features; the level of similarity between two images is determined by a hypothesis test regarding the joint distribution of transformed feature pair. The proposed approach is benchmarked against two popular statistical analysis methods, Linear Discriminant Analysis (LDA) and Principal Component Analysis with whitening (PCAw). The CCA approach is shown to achieve competitive retrieval performances on popular datasets such as Oxford5k and Paris6k.**

Keywords—Canonical Correlation Analysis, Chernoff Information, Hypothesis Testing, Image Retrieval, Multivariate Gaussian Distribution.

I. Introduction

The past few decades have seen the explosive growth of online image databases. This growth presents valuable opportunities for the development of visual-data-driven applications, but at the same time poses significant challenges to the Content-Based Image Retrieval (CBIR) technology [1].

Traditional approaches to CBIR typically rely on handcrafted scale- and orientation-invariant image features [2]–[6]. Recent advances in Deep Learning (DL) for image classification and object recognition have brought Convolutional Neural Networks (CNNs) to the spotlight as contenders for CBIR. Although CNN models are usually trained to perform different tasks than CBIR, Razavian *et al.* [7] have shown the potential of features extracted from modern deep CNNs, commonly referred to as DL features. Retrieval methods utilizing DL features can generally be differentiated by whether they train (fine-tune) the CNN model or not. The earlier use of CNN for CBIR focuses on methods using Off-The-Shelf (OTS) CNNs (i.e., popular pre-trained CNNs) for feature extraction, [8], [9], [10] and [11] to name a few. A major pro of those methods is the low implementation cost, which is largely attributed to the direct adoption of pre-trained CNNs. Their performance is, overall, on par with state-of-the-art conventional methods that rely on handcrafted features. However, in order to push it further, another group of methods, like [12], [13], and [14], has incorporated a CNN fine-tuning component to enhance the discrimination power of the extracted features. The end-to-end learning framework proposed in [15] for CBIR is by far the cream of the crop in fine-tuned CNN-based methods. It beats almost all popular conventional and OTS-CNN-based methods on all standard testing datasets. However, this performance improvement comes at a significant cost. Indeed, the method requires the training of large triple-branched CNN architecture using a large training dataset, which may not be affordable all the time.

To better leverage DL features for image retrieval, many preprocessing methods have been developed. Among them, Principal Component Analysis with whitening (PCAw) and Linear Discriminant Analysis (LDA) are most widely used. In spite of their popularity, PCA and LDA have obvious weaknesses: the dimensionality reduction in PCA usually ignores critical principal components with small contribution rate while the performance of LDA tends to degrade with decreasing differences between mismatched features. Therefore, it is desirable to have a preprocessing method that is more robust against dimensionality reduction and more sensitive to feature mismatch. This motivates us to bring Canonical Correlation Analysis (CCA) [16] into image information preprocessing.

CCA is a powerful tool to investigate the relationship between two sets of multivariate data. It can be used to identify two projection subspaces of a given dimension that capture maximally the correlation between the two sets. The applications of CCA in cross-modality matching/retrieval have been extensively studied, from those based on handcrafted features (see, e.g., [17]) to the more recent ones that rely on DL features [18]–[20]. Some related theoretical development can be found in [21], [22].

In the line of seeking computationally-efficiency and affordability, a new image retrieval method based on OTS deep CNNs is developed and presented in this work. It is built around CCA, but has several distinctive aspects as compared to the related works. For dimensionality reduction purposes (feature compression), the proposed method, named CCA-based method, employs a basis-vector selection technique using Chernoff information. It provides a ranking on how discriminative the basis vectors are. Those vectors and their ranking are both learned form a training set of features that are extracted from a pre-trained CNN – the neural network is not trained in this work. For a new pair of unknown features, the basis vectors are used to transform the features and compress them if needed. Then, a hypothesis test on the joint distribution of pairs of feature elements is carried out to obtain a matching score. Top best retrieved matches are identified using their matching score. The experimental results indicate that the proposed method can achieve competitive retrieval performances on some popular datasets such as Oxford5k and Parise6k.

This paper is organized as follows. A detailed description of

978-1-7281-0954-1/19/$31.00 ©2019 IEEE

the proposed CCA-based preprocessing method together with the associated matching procedure is presented in Section 3. The experimental results and the relevant discussions can be found in Section 4. The paper is concluded with some final remarks in Section 5.

II. Proposed Method

Inspired by CCA, a correlation analysis method is developed for image retrieval purposes. Using a training dataset of features extracted from a pre-trained CNN model, a set of canonical vectors are learned to serve as the basis vectors of the feature space. They are used to transform the features of a pair of images into a new subspace, in which a selection method is applied to identify the most discriminative elements of the transformed features. Then, those elements undergo a hypothesis test to determine the degree of similarity between the features and, therefore, the two images. The following four subsections will shed more light on the details of that process.

A. Image pre-processing and feature extraction

The CNN model used for feature extraction is VGG16 [23]. It takes an input image of maximum size of 1024x1024 and produces 512 feature maps from its last convolutional layer. A pooling technique is applied on those feature maps to extract a single feature element from each one. Those elements are concatenated to get a 512-dimensional vector, which undergoes decentralization and normalization to form the global feature vector representing the image.

B. Correlation analysis and canonical vectors

In the core of the proposed method lies the set of canonical vectors. They are learned in a fashion inspired by CCA and from a large training set of matching and non-matching image features. The learning process goes through the following steps: *Step 1*: The two matching matrices are shown below:

$$\mathbf{X}^{(M)} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_L],$$
$$\mathbf{Y}^{(M)} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_L],$$

where L is the total number of matching pairs, $\mathbf{x}_l$ and $\mathbf{y}_l$ for $l \in \{1, 2, ..., L\}$ are a pair of feature column vectors representing two matching images. The training data matrix of matching features $\mathbf{H}^{(M)}$ is constructed as follows:

$$\mathbf{H}^{(M)} = \begin{bmatrix} \mathbf{X}^{(M)} & \mathbf{Y}^{(M)} \\ \mathbf{Y}^{(M)} & \mathbf{X}^{(M)} \end{bmatrix}_{(1024 \times 2L)}. \tag{1}$$

The estimated covariance matrix of matching features is given by

$$\begin{aligned} \mathbf{\Phi}^{(M)} =& \frac{1}{2(L-1)} \mathbf{H}^{(M)} (\mathbf{H}^{(M)})^T \\ =& \begin{bmatrix} \mathbf{\Sigma}_{auto} & \mathbf{\Sigma}^{(M)} \\ \mathbf{\Sigma}^{(M)} & \mathbf{\Sigma}_{auto} \end{bmatrix}, \end{aligned} \tag{2}$$

where $\mathbf{\Sigma}_{auto} = \frac{\mathbf{X}^{(M)}(\mathbf{X}^{(M)})^T + \mathbf{Y}^{(M)}(\mathbf{Y}^{(M)})^T}{2(L-1)}$ and $\mathbf{\Sigma}^{(M)} = \frac{\mathbf{X}^{(M)}(\mathbf{Y}^{(M)})^T + \mathbf{Y}^{(M)}(\mathbf{X}^{(M)})^T}{2(L-1)}$.

Step 2: By randomly permuting the columns of $\mathbf{X}^{(M)}$ and $\mathbf{Y}^{(M)}$, we can get two non-matching matrices

$$\mathbf{X}^{(N)} = [\mathbf{x}_{\pi(1)}, \mathbf{x}_{\pi(2)}, ..., \mathbf{x}_{\pi(L)}],$$
$$\mathbf{Y}^{(N)} = [\mathbf{y}_{\pi'(1)}, \mathbf{y}_{\pi'(2)}, ..., \mathbf{y}_{\pi'(L)}].$$

With a similar procedure to that of *step 1*, a covariance matrix $\mathbf{\Phi}^{(N)}$ is estimated for the non-matching features $\mathbf{H}^{(N)}$:

$$\begin{aligned} \mathbf{\Phi}^{(N)} =& \frac{1}{2(L-1)} \mathbf{H}^{(N)} (\mathbf{H}^{(N)})^T \\ =& \begin{bmatrix} \mathbf{\Sigma}_{auto} & \mathbf{\Sigma}^{(N)} \\ \mathbf{\Sigma}^{(N)} & \mathbf{\Sigma}_{auto} \end{bmatrix}, \end{aligned} \tag{3}$$

where $\mathbf{\Sigma}^{(N)} = \frac{\mathbf{X}^{(N)}(\mathbf{Y}^{(N)})^T + \mathbf{Y}^{(N)}(\mathbf{X}^{(N)})^T}{2(L-1)}$.

Step 3: Since $\mathbf{\Sigma}_{auto}$ is Positive Definite (PD), both covariances, $\mathbf{\Phi}^{(M)}$ and $\mathbf{\Phi}^{(N)}$, are multiplied left and right by $\mathbf{\Sigma}_{auto}^{-\frac{1}{2}}$ to de-correlate their diagonal blocks:

$$\hat{\mathbf{\Phi}}^{(M)} = \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} \mathbf{\Phi}^{(M)} \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} = \begin{bmatrix} \mathbf{I} & \mathbf{J}^{(M)} \\ \mathbf{J}^{(M)} & \mathbf{I} \end{bmatrix}, \tag{4}$$

$$\hat{\mathbf{\Phi}}^{(N)} = \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} \mathbf{\Phi}^{(N)} \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} = \begin{bmatrix} \mathbf{I} & \mathbf{J}^{(N)} \\ \mathbf{J}^{(N)} & \mathbf{I} \end{bmatrix}, \tag{5}$$

where $J^{(M)} = \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} \mathbf{\Sigma}^{(M)} \mathbf{\Sigma}_{auto}^{-\frac{1}{2}}$, and $J^{(N)} = \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} \mathbf{\Sigma}^{(N)} \mathbf{\Sigma}_{auto}^{-\frac{1}{2}}$.

Step 4: Eigen Decomposition (ED) [24] is applied on the $\mathbf{J}^{(M)}$:

$$\mathbf{J}^{(M)} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T. \tag{6}$$

The eigenvectors in the columns of $\mathbf{U}$ are the sought-after canonical vectors for matching image features, and the block-wise left- and right-multiplication of both $\hat{\mathbf{\Phi}}^{(M)}$ and $\hat{\mathbf{\Phi}}^{(N)}$ by $\mathbf{U}^T$ and $\mathbf{U}$, respectively, yields the following pair of matrices:

$$\begin{bmatrix} \mathbf{U}^T\mathbf{U} & \mathbf{U}^T\mathbf{J}^{(M)}\mathbf{U} \\ \mathbf{U}^T\mathbf{J}^{(M)}\mathbf{U} & \mathbf{U}^T\mathbf{U} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{\Lambda} \\ \mathbf{\Lambda} & \mathbf{I} \end{bmatrix}, \tag{7}$$

$$\begin{bmatrix} \mathbf{U}^T\mathbf{U} & \mathbf{U}^T\mathbf{J}^{(N)}\mathbf{U} \\ \mathbf{U}^T\mathbf{J}^{(N)}\mathbf{U} & \mathbf{U}^T\mathbf{U} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{\Pi} \\ \mathbf{\Pi} & \mathbf{I} \end{bmatrix}, \tag{8}$$

where $\mathbf{\Pi} = \mathbf{U}^T\mathbf{J}^{(N)}\mathbf{U}$. The off-diagonal blocks of Equ. (7) are de-correlated by the canonical vectors $\mathbf{U}$ whereas $\mathbf{\Pi}$ of Equ. (8) is not. This is due to the fact that $\mathbf{\Sigma}_{XY}^{(N)}$ and $\mathbf{\Sigma}_{YX}^{(N)}$ are the cross covariances of non-matching image features, $\mathbf{X}^{(N)}$ and $\mathbf{Y}^{(N)}$.

C. Chernoff information for feature element selection

The learned canonical vectors of matching image features are orthonormal basis vectors that span the whole $\mathbb{R}^{512}$, and it is expected that they are not all useful in the process of measuring the similarity between two feature vectors of an unknown pair of images– more on this in the next subsection. Hence, it is of great interest to identify those canonical vectors that are more *discriminative* than others. The off-diagonal blocks of the covariance matrix of non-matching image features come on handy right now. Using Chernoff information (CI) [25] with the diagonal elements of both $\mathbf{\Lambda}$ and $\mathbf{\Pi}$, a ranking of the most different diagonal element pairs

978-1-7281-0954-1/19/$31.00 ©2019 IEEE

could be established. This helps select a subset of n-canonical vectors, amounting to a selection of some elements from the feature vectors.

Starting with the diagonal elements of $\mathbf{\Lambda}$ and $\mathbf{\Pi}$:

$$\mathbf{\Lambda} = \begin{bmatrix} c_1^{(M)} & 0 & \dots & 0 \\ 0 & c_2^{(M)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & c_{512}^{(M)} \end{bmatrix} \tag{9}$$

$$\mathbf{\Pi} = \begin{bmatrix} c_1^{(N)} & \pi_{1,2} & \dots & \pi_{1,512} \\ \pi_{2,1} & c_2^{(N)} & \dots & \pi_{2,512} \\ \vdots & \vdots & \ddots & \vdots \\ \pi_{512,1} & \pi_{512,2} & \dots & c_{512}^{(N)} \end{bmatrix}. \tag{10}$$

Therefore, matching coefficient $c_t^{(M)} = [\mathbf{\Lambda}]_{tt}$, and non-matching coefficient $c_t^{(N)} = [\mathbf{\Pi}]_{tt}$ where $t \in \{1, 2, \dots, 512\}$.

Define the following set of 2×2 matrices:

$$\mathbf{S}_t^{(M)} = \begin{bmatrix} 1 & c_t^{(M)} \\ c_t^{(M)} & 1 \end{bmatrix}, \mathbf{S}_t^{(N)} = \begin{bmatrix} 1 & c_t^{(N)} \\ c_t^{(N)} & 1 \end{bmatrix}.$$

Now, let $(\mathbf{S}_t^{(\lambda_t)})^{-1} = \lambda_t (\mathbf{S}_t^{(M)})^{-1} + (1-\lambda_t)(\mathbf{S}_t^{(N)})^{-1}$, $\lambda_t \in [0, 1]$ and define

$$D(\mathbf{S}_t^{(\lambda_t)} || \mathbf{S}_t^{(M)}) = \frac{1}{2} \log_e \frac{|\mathbf{S}_t^{(M)}|}{|\mathbf{S}_t^{(\lambda_t)}|} + \frac{1}{2} \operatorname{tr}((\mathbf{S}_t^{(M)})^{-1} \mathbf{S}_t^{(\lambda_t)}) - 1, \tag{11}$$

$$D(\mathbf{S}_t^{(\lambda_t)} || \mathbf{S}_t^{(N)}) = \frac{1}{2} \log_e \frac{|\mathbf{S}_t^{(N)}|}{|\mathbf{S}_t^{(\lambda_t)}|} + \frac{1}{2} \operatorname{tr}((\mathbf{S}_t^{(N)})^{-1} \mathbf{S}_t^{(\lambda_t)}) - 1, \tag{12}$$

where $\operatorname{tr}(\cdot)$ represents the trace of input matrix in linear algebra. The equation $D(\mathbf{S}_t^{(\lambda_t)} || \mathbf{S}_t^{(M)}) = D(\mathbf{S}_t^{(\lambda_t)} || \mathbf{S}_t^{(N)})$ has a unique solution $\lambda_t = \lambda_t^*$. The Chernoff information $CI(\mathbf{S}_t^{(M)} || \mathbf{S}_t^{(N)})$ is defined as

$$CI(\mathbf{S}_t^{(M)} || \mathbf{S}_t^{(N)}) = D(\mathbf{S}_t^{(\lambda_t^*)} || \mathbf{S}_t^{(M)}) = D(\mathbf{S}_t^{(\lambda_t^*)} || \mathbf{S}_t^{(N)}). \tag{13}$$

Solving for each λ_t^*, CI of all pairs $(\mathbf{S}_t^{(M)}, \mathbf{S}_t^{(N)})$ could be evaluated, leading to a ranking of the most different pairs of diagonal elements $(c_t^{(M)}, c_t^{(N)})$ and, therefore, the most discriminative k-canonical vectors of $\mathbf{U}$. Those vectors make the columns of the new canonical vector matrix $\tilde{\boldsymbol{U}}$. In additional, the top k different pairs of diagonal elements $(\tilde{c}_i^{(M)}, \tilde{c}_i^{(N)})$, and the corresponding $(\tilde{\boldsymbol{S}}_i^{(M)}, \tilde{\boldsymbol{S}}_i^{(N)})$ are selected, where $i \in \{1, 2, \dots, k\}$.

D. Similarity measurement with hypothesis testing

The selected canonical vectors could be used to measure the similarity between any new pair of images through a hypothesis test. For any pair of feature vectors $(\mathbf{x}_r, \mathbf{y}_c)$, the transformed feature column vectors are computed as follows:

$$\mathbf{w} = [w_1, w_2, \dots, w_k]^T = \tilde{\boldsymbol{U}}^T \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} \mathbf{x}_r, \tag{14}$$

$$\mathbf{v} = [v_1, v_2, \dots, v_k]^T = \tilde{\boldsymbol{U}}^T \mathbf{\Sigma}_{auto}^{-\frac{1}{2}} \mathbf{y}_c, \tag{15}$$

where $r, c \in \{1, 2, \dots, L\}$. The main assumption underlying the hypothesis test is that the pair of feature vectors $(\mathbf{x}_r, \mathbf{y}_c)$ comes from a jointly Gaussian distribution. Hence, the elements (w_i, v_i) for $i \in \{1, 2, \dots, k\}$ are also jointly Gaussian and independent from each other. All the Gaussian distributions are characterized by a zero-mean vector and a set of 2×2-covariance matrices $(\tilde{\boldsymbol{S}}_i^{(M)}, \tilde{\boldsymbol{S}}_i^{(N)})$. Since the final outcome of the test is whether the two images are matching or not, each pair has two possible Gaussian densities, one is characterized by $\tilde{\boldsymbol{S}}_i^{(M)}$ and the other is characterized by $\tilde{\boldsymbol{S}}_i^{(N)}$.

Considering the multivariate gaussian distribution [26], the density function of a pair (w_i, v_i) is defined as

$$P_M(w_i, v_i) = \frac{e^{-\frac{1}{2}\begin{bmatrix} w_i & v_i \end{bmatrix} \begin{bmatrix} 1 & \tilde{c}_i^{(M)} \\ \tilde{c}_i^{(M)} & 1 \end{bmatrix}^{-1} \begin{bmatrix} w_i \\ v_i \end{bmatrix}}}{\sqrt{(2\pi)^2 \begin{vmatrix} 1 & \tilde{c}_i^{(M)} \\ \tilde{c}_i^{(M)} & 1 \end{vmatrix}}}, \tag{16}$$

or

$$P_N(w_i, v_i) = \frac{e^{-\frac{1}{2}\begin{bmatrix} w_i & v_i \end{bmatrix} \begin{bmatrix} 1 & \tilde{c}_i^{(N)} \\ \tilde{c}_i^{(N)} & 1 \end{bmatrix}^{-1} \begin{bmatrix} w_i \\ v_i \end{bmatrix}}}{\sqrt{(2\pi)^2 \begin{vmatrix} 1 & \tilde{c}_i^{(N)} \\ \tilde{c}_i^{(N)} & 1 \end{vmatrix}}}. \tag{17}$$

Using the pair of density functions, Equ. (16) and Equ. (17), the hypothesis test is carried out to obtain a confidence score with the following expression:

$$score = \sum_{i=1}^{k} \log \frac{P_M(w_i, v_i)}{P_N(w_i, v_i)}. \tag{18}$$

Substituting Equ. (16), (17) and expanding, it becomes

$$\begin{aligned} score &= \sum_{i=1}^{k} (\log P_M(w_i, v_i) - \log P_N(w_i, v_i)) \\ &= \sum_{i=1}^{n} \Bigg(- \frac{w_i^2 - 2 w_i v_i \tilde{c}_i^{(M)} + v_i^2}{2\pi \sqrt{(1 - (\tilde{c}_i^{(M)})^2)}} + \frac{w_i^2 - 2 w_i v_i \tilde{c}_i^{(N)} + v_i^2}{2\pi \sqrt{(1 - (\tilde{c}_i^{(N)})^2)}} \\ &\qquad + \log \frac{\sqrt{1 - (\tilde{c}_i^{(N)})^2}}{\sqrt{1 - (\tilde{c}_i^{(M)})^2}} \Bigg). \end{aligned} \tag{19}$$

This $score$ determines how similar the two images are. The higher the $score$ is, the more likely the two images are a match.

III. Experimental Results

A. Training datasets

Two datasets are independently used for training, namely 120k-Structure from Motion (120k-SfM) and 30k-Structure from Motion (30k-SfM) [27]. Both are preprocessed to avoid overlapping elements with the evaluation datasets. A brief description of both datasets is given below:

978-1-7281-0954-1/19/$31.00 ©2019 IEEE

1) 120k-Structure from Motion (120k-SfM) [27]: This dataset is selected from the dataset used in the work of Schonberger *et al,* [28] which contains 713 3D models with nearly 120k images. Each image has a size of 1024 x 1024. The raw unprocessed dataset includes all image from Oxford5k and Paris6k. Thus, those subsets are both removed (98 clusters are eliminated).

2) 30k-Structure from Motion (30k-SfM) [27]: This training dataset is a subset of the 120k-SfM, which contains around 30k images and 551 classes. The maximum dimensions of each image is resized to 362x362.

Each dataset serves its own purpose; the 30k-SfM is an example of training on a small dataset while 120k-SfM is an example of training on a large dataset, exploring the pros and cons of both. Compared to the 30k-SfM set, the 120k-SfM is expected to provide richer features for all methods being tested, PCAw, LDA, the proposed method (G-CCA), and a variation of proposed method by replacing the hypothesis testing with the scalar similarity (S-CCA).

B. Training details

The feature vector of an image is extracted from the last convolutional layer of a pre-trained VGG16 using one of the following three pooling strategies: Sum-Pooled Convolutions (SPoC) [9], Maximum Activation of Convolutions (MAC) [8], and Standard Deviation (SD) pooling. Training is performed once for each one of those strategies for the purpose of performance comparison.

For benchmarking, the proposed method (G-CCA) and its variation (S-CCA) are trained along with two more feature-space analysis methods, namely PCAw [29], and LDA. The former analyzes the covariance matrix of the training image features to derive a basis matrix of the feature space. That basis matrix is used to whiten and compress new image features, which are later used with the scalar similarity measure to make a matching or non-matching decision. The details of the PCAw method and its performance have been laid out in [9]. On the other hand, due to the popularity of LDA [30] in statistical analysis, it has been applied in image-retrieval problems [31]. Here it will be used as a competing method of feature-space analysis.

LDA is trained on the classes provided with both training datasets. It develops a set of projection vectors for which the classes are best linearly separated. Those projection vectors are stored and used as a means to transform and compress (reduce the dimensionality of) new feature vectors. Scalar similarity is, then, applied to the transformed features to determine whether they are matching or not.

C. Evaluation datasets and details

Two datasets are used to evaluate the performance of each retrieval method, namely Oxford5k and Paris6k. They are part of the large raw 120k-SfM dataset, but are excluded from the training dataset. A short description of these two datasets is given below.

1) Oxford5k [32]: It is a dataset with 5063 images and 55 query images for 11 different buildings. It is annotated with bounding boxes for the main objects.

2) Paris6k [33]: It is a dataset with 6412 images and 55 query images for 11 different buildings. It is also annotated with bounding boxes.

The performance of a retrieval method is assessed using mean-Average Precision (mAP) [32]. Standard evaluation protocol is followed for the Oxford5k, Paris6k. The query images are all cropped with the provided bounding boxes before they are fed to VGG16. All methods are trained and evaluated twice. We first perform training on the small dataset, 30k-SfM, followed by evaluation. The second training is based on the larger 120k-SfM. In this way, the effect of the dataset size and diversity on all methods could be studied.

D. Performance evaluation and analysis

In Table I, the baseline performances of MAC, SPoC, and SD methods are reported without any preprocessing and dimensionality reduction (DR). In the evaluation, we use the proposed method (G-CCA), and a variation of the proposed method by replacing the multivariate Gaussian distribution with the scalar similarity (S-CCA). From Table I, we observe that the G-CCA achieves higher performance than S-CCA on most cases except for the SPoC in Paris6k, and $\mathcal{R}$Paris.

Using three different pooling strategies, two image retrieval methods are trained on the 30k-SfM dataset, and evaluated on the two test sets. Table II is a comprehensive depiction of all those experiments. It shows the results of each retrieval method using all pooling strategies and with different feature dimensionality choices (compression levels). The LDA results are not reported there, for LDA cannot be trained on the 30k-SfM dataset. It is a consequence of the fact that the difference between each class is too small for LDA training. From the table II, four observations stand out. The first is the effect of the pooling strategy on the G-CCA, SD pooling seems to boost up the performance of all methods at every

TABLE I
PERFORMANCE COMPARISON OF THE BASELINE, S-CCA, AND G-CCA ON OXFORD5K AND PARIS6K WITHOUT DIMENSION REDUCTION

Method	Oxford5k	Paris6k
MAC	0.5311	0.7455
S-CCA + MAC	0.5765	0.7287
G-CCA + MAC	**0.6229**	**0.7671**
SPoC	0.5315	0.6320
S-CCA + SPoC	0.6851	**0.7849**
G-CCA + SPoC	**0.7131**	0.7455
SD	0.6073	0.7330
S-CCA + SD	0.6900	0.7799
G-CCA + SD	**0.7393**	**0.8160**

[1] The evaluation results are based on 120k-SfM learning database.
[2] The MAC, SPoC, and SD are evaluated without any preprocessing methods.
[3] For the same type of features, the best performances are highlighted in **bold**.

978-1-7281-0954-1/19/$31.00 ©2019 IEEE

TABLE II
EVALUATION RESULTS FROM 30K-SFM LEARNING DATABASE ON OXFORD5K AND PARIS6K

Learning dataset: 30k-SfM													
	Dim	MAC				SPoC				SD			
		LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA
Oxford5k	25	—	0.3504	0.2424	**0.3901**	—	0.4796	0.2511	**0.4879**	—	0.4993	0.3355	**0.5008**
	50	—	0.4264	0.3290	**0.4690**	—	0.5153	0.3149	**0.5437**	—	0.5129	0.4542	**0.5856**
	100	—	0.4980	0.4106	**0.5064**	—	0.5217	0.4549	**0.6219**	—	0.6038	0.5292	**0.6300**
	200	—	0.5547	0.4933	**0.5592**	—	0.6072	0.5123	**0.6658**	—	0.6580	0.6838	**0.6877**
	300	—	**0.5710**	0.5400	0.5406	—	0.6433	0.5274	**0.6723**	—	0.6728	0.6610	**0.6824**
	400	—	**0.5726**	0.5614	0.5463	—	0.6516	0.5373	**0.6713**	—	0.6825	0.6699	**0.6831**
	450	—	**0.5731**	0.5618	0.5424	—	0.6549	0.5333	**0.6696**	—	**0.6869**	0.6750	0.6812
	512	—	0.5620	**0.5621**	0.5418	—	0.6535	0.6535	**0.6704**	—	0.6805	0.6786	**0.6815**
	Dim	MAC				SPoC				SD			
		LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA
Paris6k	25	—	0.5171	0.4096	**0.5306**	—	**0.5411**	0.4412	0.5223	—	0.5746	0.4467	**0.5820**
	50	—	0.6130	0.5305	**0.7118**	—	0.5680	0.5601	**0.6240**	—	0.6459	0.6207	**0.6964**
	100	—	0.6201	0.5893	**0.7118**	—	0.6374	0.6279	**0.6806**	—	0.7335	0.6934	**0.7494**
	200	—	0.7049	0.6417	**0.7118**	—	0.6887	0.6981	**0.7152**	—	0.7747	0.7561	**0.7851**
	300	—	0.7037	0.6699	**0.7065**	—	0.7080	**0.7319**	0.7208	—	0.7846	0.7769	**0.7895**
	400	—	0.7056	0.6915	**0.7099**	—	0.7345	**0.7404**	0.7213	—	**0.8037**	0.7961	0.7899
	450	—	0.7115	0.7008	**0.7046**	—	0.7447	**0.7481**	0.7217	—	**0.8120**	0.8027	0.7897
	512	—	0.7063	**0.7064**	0.7039	—	0.7530	**0.7532**	0.7218	—	0.8067	**0.8086**	0.7895

[1] The best performances in each dimension are highlighted in **bold**.

TABLE III
EVALUATION RESULTS FROM 120K-SFM LEARNING DATABASE ON OXFORD5K AND PARIS6K

Learning dataset: 120k-SfM													
	Dim	MAC				SPoC				SD			
		LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA
Oxford5k	25	0.3603	0.3830	0.2682	**0.4235**	0.4758	0.4472	0.3203	**0.4783**	0.4759	0.4779	0.3262	**0.5017**
	50	0.4760	0.4277	0.3720	**0.4780**	0.5612	0.4930	0.4085	**0.5627**	0.5375	0.5129	0.4521	**0.5688**
	100	0.5157	0.5185	0.4510	**0.5432**	0.6017	0.5675	0.5379	**0.6338**	0.6429	0.6038	0.5547	**0.6597**
	200	0.5887	0.5443	0.5516	**0.6182**	0.6571	0.6399	0.6440	**0.6947**	0.6861	0.6337	0.6485	**0.7176**
	300	0.6028	0.5619	0.5723	**0.6246**	0.6643	0.6575	0.6651	**0.7089**	0.7030	0.6638	0.6770	**0.7325**
	400	0.5974	0.5793	0.5680	**0.6251**	0.6688	0.6808	0.6699	**0.7116**	0.7020	0.6933	0.6824	**0.7381**
	450	0.5939	0.5819	0.5777	**0.6233**	0.6678	0.6862	0.6754	**0.7124**	0.6972	0.6952	0.6894	**0.7391**
	512	0.5868	0.5765	0.5765	**0.6229**	0.6613	0.6851	0.6851	**0.7131**	0.6958	0.6900	0.6900	**0.7393**
	Dim	MAC				SPoC				SD			
		LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA	LDA	PCAw	S-CCA	G-CCA
Paris6k	25	0.5553	0.4907	0.4566	**0.5828**	**0.5781**	0.4851	0.4601	0.5541	0.6204	0.5746	0.5036	**0.6328**
	50	0.6362	0.6186	0.5254	**0.6576**	**0.6384**	0.5693	0.5127	0.6139	0.6900	0.6459	0.5781	**0.7001**
	100	0.6994	0.6822	0.6160	**0.7242**	**0.6916**	0.6608	0.6375	0.6781	0.7502	0.7335	0.6972	**0.7593**
	200	0.7162	0.7095	0.6739	**0.7606**	0.7244	0.7111	0.7122	**0.7248**	0.7845	0.7877	0.7700	**0.8023**
	300	0.7299	0.7251	0.6921	**0.7678**	**0.7493**	0.7417	0.7460	0.7424	0.8030	0.8070	0.7993	**0.8159**
	400	0.7247	0.7233	0.7126	**0.7683**	0.7548	0.7679	**0.7724**	0.7451	0.8042	**0.8180**	0.8089	0.8156
	450	0.7197	0.7229	0.7191	**0.7675**	0.7540	**0.7796**	0.7786	0.7455	0.8003	**0.8169**	0.8123	0.8158
	512	0.7111	0.7274	0.7287	**0.7671**	0.7549	0.7845	**0.7849**	0.7455	0.7971	0.8178	**0.8179**	0.8160

[1] The best performances in each dimension are highlighted in **bold**.

choice of feature dimensionality while MAC makes the G-CCA somehow superior to PCAw in almost all dimensions of the all test sets. This is interesting because for the MAC, SPoC, and SD pooling strategies, the proposed method outperforms PCAw at low feature dimensionality, which is the second observation. It suggests that the proposed method is a better choice for producing compact features than PCAw regardless of the pooling strategy. The third observation is that G-CCA have higher robustness to DR than S-CCA, this is showed by all experiment results in Table II. It is worth to point out that S-CCA is useful with SPoC on Paris6k, and its performance is even better than G-CCA and PCA at high dimensions, which gives the last observation.

The moderate performance of the proposed method could be improved by the use of a larger training set like the 120k-SfM. Table III, similar to Table II, shows the evaluation results of all three methods using the three pooling strategies and different dimensionality choices. Clearly, the increased-size training set results in an improved mAP performance on all test sets and using all pooling strategies. The most interesting observation there is how the proposed method outperforms all others on Oxford5k. This superiority across all dimensionalities is only attained on the Paris6k using MAC pooling. Other pooling strategies improve the retrieval performance of all methods in general but have different effects on the proposed one; although SPoC improves the mAP, it causes the G-CCA to lose its edge at almost all dimensionalities on Paris6k. This is not the case with SD pooling, which retains the dominating performance of the proposed method at reduced feature dimensionalities.

Based on the Table II and III, there are two main advantages of G-CCA over the PCAw and LDA. The First is that the CCA-based methods can be trained from the learning dataset that the difference between each classes are small, but LDA cannot be trainable on such dataset. The second advantage is that the G-CCA usually have better performance than PCAw after DR.

978-1-7281-0954-1/19/$31.00 ©2019 IEEE

IV. Conclusion

Leveraging the good performance of OTS CNNs in image classification, CCA-based methods are proposed to analyze DL features for image retrieval applications. By avoiding CNN fine-tuning, it achieves good retrieval accuracy with as minimal computational burden as possible. Experimental results on standard evaluation datasets have shown that its performance is very competitive to that of other OTS-CNN-based methods.

References

[1] Z. Wengang, L. Houqiang, and T. Qi, "Recent advance in content-based image retrieval: A literature survey," *arXiv preprint arXiv:1706.06064*, 2017.

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

[3] 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.

[4] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[5] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *International Conference on Image and Signal Processing*. Springer, 2008, pp. 236–243.

[6] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.

[7] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 806–813.

[8] G. Tolias, R. Sicre, and H. Jégou, "Particular object retrieval with integral max-pooling of CNN activations," *arXiv preprint arXiv:1511.05879*, 2015.

[9] A. Babenko and V. Lempitsky, "Aggregating local deep features for image retrieval," in *IEEE International Conference on Computer Vision*, 2015, pp. 1269–1277.

[10] J. Lin, L.-Y. Duan, S. Wang, Y. Bai, Y. Lou, V. Chandrasekhar, T. Huang, A. Kot, and W. Gao, "Hnip: Compact deep invariant representations for video matching, localization, and retrieval," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 1968–1983, 2017.

[11] Y. Kalantidis, C. Mellina, and S. Osindero, "Cross-dimensional weighting for aggregated deep convolutional features," in *European Conference on Computer Vision*. Springer, 2016, pp. 685–701.

[12] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han, "Largescale image retrieval with attentive deep local features," in *IEEE International Conference on Computer Vision*, 2017, pp. 3456–3465.

[13] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5297–5307.

[14] F. Radenović, G. Tolias, and O. Chum, "CNN image retrieval learns from bow: Unsupervised fine-tuning with hard examples," in *European Conference on Computer Vision*. Springer, 2016, pp. 3–20.

[15] A. Gordo, J. Almazan, J. Revaud, and D. Larlus, "End-to-end learning of deep visual representations for image retrieval," *International Journal of Computer Vision*, vol. 124, no. 2, pp. 237–254, 2017.

[16] J. Richard A and D. W. Wichern, "canonical correlation analysis," in *Applied Multivariate Statistical Analysis*. Pearson, 2018, pp. 539–574.

[17] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik, "A multi-view embedding space for modeling internet images, tags, and their semantics," *International journal of computer vision*, vol. 106, no. 2, pp. 210–233, 2014.

[18] F. Yan and K. Mikolajczyk, "Deep correlation for matching images and text," in *IEEE conference on computer vision and pattern recognition*, 2015, pp. 3441–3450.

[19] M. Dorfer, J. Schlüter, A. Vall, F. Korzeniowski, and G. Widmer, "End-to-end cross-modality retrieval with cca projections and pairwise ranking loss," *International Journal of Multimedia Information Retrieval*, vol. 7, no. 2, pp. 117–128, 2018.

[20] Y. Yu, S. Tang, K. Aizawa, and A. Aizawa, "Category-based deep cca for fine-grained venue discovery from multimodal data," *arXiv preprint arXiv:1805.02997*, 2018.

[21] Z. Lin and J. Peltonen, "An information retrieval approach for finding dependent subspaces of multiple views," in *International Conference on Machine Learning and Data Mining in Pattern Recognition*. Springer, 2017, pp. 1–16.

[22] O. Yair and R. Talmon, "Local canonical correlation analysis for nonlinear common variables discovery," *IEEE Transactions on Signal Processing*, vol. 65, no. 5, pp. 1101–1115, 2017.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[24] H. Abdi, "The eigen-decomposition: eigenvalues and eigenvectors," *Encyclopedia of Measurement and Statistics*, pp. 304–308, 2007.

[25] F. Nielsen, "An information-geometric characterization of chernoff information," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 269–272, 2013.

[26] S. J. Prince, "Common probability distribution," in *Computer Vision: Models, Learning and Inference*. Cambridge University Press, 2012, pp. 35–42.

[27] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[28] J. L. Schonberger, F. Radenovic, O. Chum, and J.-M. Frahm, "From single image query to detailed 3d reconstruction," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5126–5134.

[29] H. Jégou and O. Chum, "Negative evidences and co-occurences in image retrieval: The benefit of pca and whitening," in *European Conference on Computer Vision*. Springer, 2012, pp. 774–787.

[30] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[31] D. L. Swets and J. J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 8, pp. 831–836, 1996.

[32] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[33] ——, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

978-1-7281-0954-1/19/$31.00 ©2019 IEEE