Visual Recognition of Ancient Inscriptions using Convolutional Neural Network and Fisher Vector

GIUSEPPE AMATO, FABRIZIO FALCHI, and LUCIA VADICAMO, ISTI-CNR

By bringing together the most prominent European institutions and archives in the field of Classical Latin and Greek epigraphy, the EAGLE project has collected the vast majority of the surviving Greco-Latin inscriptions into a single readily-searchable database. Text-based search engines are typically used to retrieve information about ancient inscriptions (or about other artifacts). These systems require that the users formulate a text query that contains information such as the place where the object was found or where it is currently located. Conversely, visual search systems can be used to provide information to users (like tourists and scholars) in a most intuitive and immediate way, just using an image as query. In this paper, we provide a comparison of several approaches for visual recognizing ancient inscriptions. Our experiments, conducted on 17,155 photos related to 14,560 inscriptions, show that BoW and VLAD are outperformed by both Fisher Vector (FV) and Convolutional Neural Network (CNN) features. More interestingly, combining FV and CNN features into a single image representation allows achieving very high effectiveness by correctly recognizing the query inscription in more than 90% of the cases. Our results suggest that combinations of FV and CNN can be also exploited to effectively perform visual retrieval of other types of objects related to cultural heritage such as landmarks and monuments.

CCS Concepts: •**Information systems** → **Evaluation of retrieval results; Retrieval on mobile devices; Image search;** •**Computing methodologies** → **Visual content-based indexing and retrieval;**

Additional Key Words and Phrases: Fisher Vector, Convolutional Neural Network, Epigraphy, Latin and Greek inscriptions

ACM Reference Format:

Giuseppe Amato, Fabrizio Falchi, and Lucia Vadicamo. 2015. Visual Recognition of Ancient Inscription using Convolutional Neural Network and Fisher Vector. *ACM J. Comput. Cult. Herit.* V, N, Article XXXX (2016), 25 pages.
DOI: http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

The large diffusion of powerful mobile devices equipped with digital cameras has led to the emergence of novel Content-Based Image Retrieval (CBIR) applications such as visual instance retrieval systems, i.e. systems able to find object instances, from large image collections, given an image of the object to be retrieved. In this work, we focus on visual searching for ancient inscriptions such as Greek and Latin epigraphs. The use of a visual search system allows users (like tourists or epigraphists) to retrieve information about an inscription by simply taking a photo (e.g., Figure 1). This represents a profitable and user-friendly alternative to the traditional way of retrieving information from an epigraphic database that is mainly based on submitting text queries related to the place where an item has been found, or where it is currently stored.

This work was partially supported by EAGLE, Europeana network of Ancient Greek and Latin Epigraphy, co-founded by the European Commission, CIP-ICT-PSP.2012.2.1 - Europeana and creativity, Grant Agreement n. 325122.
Author's address: G. Amato, F. Falchi, and L.Vadicamo, ISTI-CNR, Via G. Moruzzi 1, 56124, Pisa, Italy; email: {giuseppe.amato, fabrizio.falchi, lucia.vadicamo}@isti.cnr.it.
 1556-4673/2016/-ARTXXXX $15.00
DOI: http://dx.doi.org/10.1145/0000000.0000000

Fig. 1. Example of an application that enables a user to get information about a visible inscription by taking a photo with a mobile device. The application uses a visual search engine to retrieve the photographed object from a database of inscriptions and provides the related information to the user. In the depicted example, the provided information are the transcription of the inscription (*M(arcus) Vipsanius / Narcissus, / rogator ab scaena.*), the type of the inscription (*sepulcralis*), the type of the object (*tabula*), and its present location (Roma), just to cite some.

Our research was conducted in the context of the Europeana network of Ancient Greek and Latin Epigraphy (EAGLE) that is a best-practice network co-funded by the European Commission, under its Information and Communication Technologies Policy Support Programme. The EAGLE Consortium is composed of nineteen partners from thirteen European countries and connects some of the most prominent European institutions and archives in the field of ancient epigraphy. EAGLE has collected the vast majority of the surviving inscriptions of the Greco-Roman world in a single readily-searchable database. The strategic partnership with Europeana Foundation creates synergies in best practice areas such as content harmonization, multi-linguality, multi-culturality and semantic interoperability. The EAGLE main aim is to provide a single user-friendly portal to search inscriptions of the Ancient World and services that include a mobile application to enable tourists to understand inscriptions simply by taking a picture with a smartphone. The EAGLE mobile application allows visitors of a site, where one of the stored inscriptions is visible (museum, street, archaeological site, printed reproduction, etc.), to take a picture with a mobile phone, send the picture to the visual search engine and receive back the information associated with that inscription.

The epigraph visual recognition functionality of the EAGLE mobile application is an example of visual instance retrieval process.

In the last decade, the research on visual instance retrieval has focused on local features [Lowe 2004] and their encoding in a compact vector such as Bag of (Visual) Words (BoW) [Sivic and Zisserman 2003], Vector of Locally Aggregated Descriptors (VLAD) [Jégou et al. 2010] and Fisher Vector (FV) [Perronnin and Dance 2007]. Starting from 2012 [Krizhevsky et al. 2012], deep learning and Convolutional Neural Networks (CNNs) [Goodfellow et al. 2016] have become the state-of-the-art to both classify and detect objects, as demonstrated by the ImageNet Large Scale Visual Recognition Challenge results. Neural networks can be trained, for instance, to recognize speech or objects in images, by learning from a large set of examples. The results of deep learning technologies have been so relevant that some researchers have drawn a parallelism between the Cambrian explosion of life on earth half a billion years ago and the diversification and applicability of robotics due to deep learning [Pratt 2015]. Deep CNNs have been recently used for producing high-level descriptors of the visual

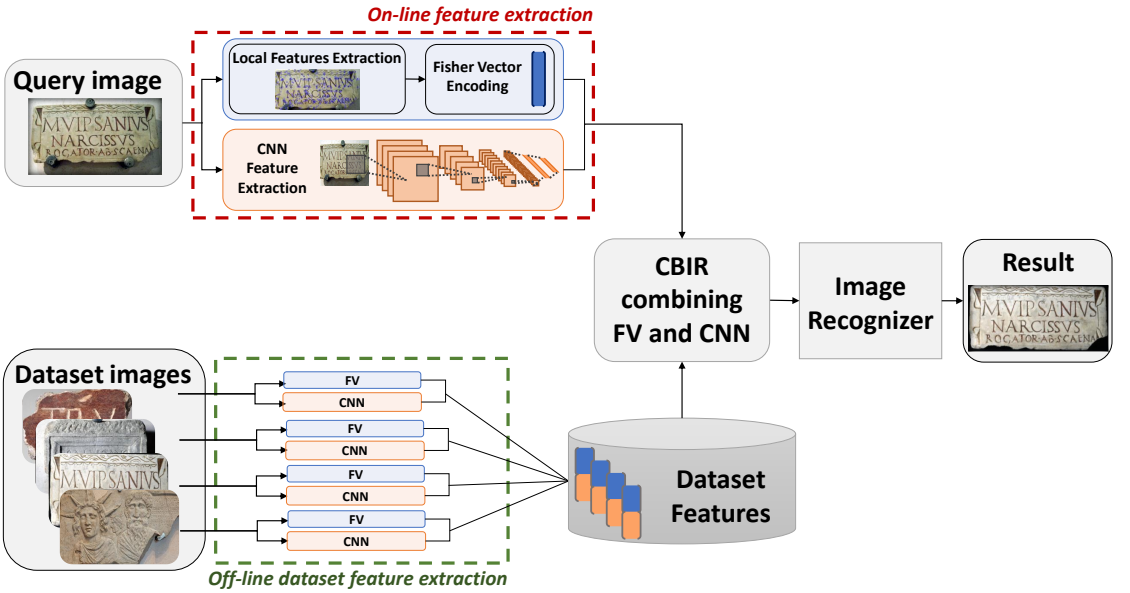


Fig. 2. Image recognition pipeline using the combination of FV and CNN

content of images [Donahue et al. 2013]. This is obtained by using the activations of the top layers of a CNN as visual features. These visual features are used to perform visual instance retrieval. A combination of CNN features and FVs has been proposed as well and achieved state-of-the-art results [Chandrasekhar et al. 2015].

Visual retrieval and recognition of epigraphs was also the subject of our previous paper [Amato et al. 2014] where BoW and VLAD approaches were tested for this task. Here, we extended our previous work by also considering: (i) Fisher Vectors for encoding local descriptors; (ii) various CNN representations; (iii) combining FVs and CNNs. We observed that both FV and CNN approaches overcome BoW and VLAD in visually recognize ancient inscriptions. Moreover, the combination of FV and CNN into a single image representation allows us to achieve very high effectiveness by correctly recognizing the query inscription in more than 90% of the cases (rather than about 70% previously achieved with BoW/VLAD). An example of visual recognition pipeline using FV and CNN is depicted in Figure 2.

All the experiments were conducted using our publicly available Visual Information Retrieval library and other open source libraries (Caffe, OpenCV), so the tested techniques can be freely adopted by anyone to visually search other databases of objects related to cultural heritage (such as monuments, landmarks, paintings, etc.). For instance, in this paper we also report some results obtained using FV and CNN features to search the Pisa Dataset [Amato et al. 2015] which contains photos of monuments and landmarks located in Pisa. Our experiments show that the combination of FV and CNN leads to improve the retrieval performance also in this case.

The rest of the paper is organized as follows. Section 2 offers an overview of other works in literature related to image representation for instance retrieval problems. Section 3 describes various image representations built upon local features (e.g. BoW, VLAD, FV) or convolutional neural networks. Section 4 discusses the evaluation experiments and the obtained results on the Epigraphic Database Roma. Section 5 concludes.

2. RELATED WORK

The problem of visually recognizing objects of cultural heritage has received growing attention over the last few years. For example, Google presented a web-scale landmark recognition engine [Zheng et al. 2009] and [Amato et al. 2015] investigated several strategies to visually recognize monuments and heritage-related landmarks.

The creation and diffusion of digital archives have led to the development of numerous multimedia systems and applications for cultural heritage. In this context, image retrieval techniques offer a promising way for searching digital archives. Indeed, using a picture as query is perhaps the easiest way for a user to obtain information about an object of interest. From the computational point of view, the search through images is performed by transforming each image (both query and database images) into a mathematical descriptor and then searching for the dataset objects whose descriptors are the most similar to the query one. The computation of a good image descriptor is crucial to the image retrieval problem and so the research for effective image representations has been object of much interest from the research community.

Current state-of-the-art approaches are mainly based on *local features* such as SIFT [Lowe 2004] and SURF [Bay et al. 2006], which are numerical representation of local structures of images. Each image is characterized by describing the visual content of typically thousands of regions of interest that are automatically selected. Images are then compared by matching their local features and searching for a geometric transformation that can associate the regions of both images.

Local features have been widely used since they allow local structures to be efficiently matched between images. However, since each image is represented by typically thousands of local features, there is a significant amount of memory consumption and time required to compare local features within large databases. The use of the information provided by each local feature is crucial for some tasks such as image stitching and 3D reconstruction. For other tasks such as image classification and retrieval high effectiveness have been achieved using the *encoding techniques* which provide meaningful summarization of all the extracted feature of an image [Jégou et al. 2010]. One profitable outcome of using encoding techniques is that they allow to represent an image by a single descriptor rather than thousands descriptors. This reduces the cost of image comparison and leads to scale up the search to large database.

By far, the most popular encoding method has been the *Bag-of-Words* (BoW) [Sivic and Zisserman 2003], initially utilized for matching objects in videos. BoW uses a *visual vocabulary* to quantize the local descriptors extracted from images and represents each image as a histogram of occurrences of visual words. From the very beginning word reduction techniques have been used and images have been ranked using the standard *term frequency-inverse document frequency* (tf-idf) weighting [Salton and McGill 1986]. Several approaches for the reduction of visual words have been investigated to improve the efficiency of BoW [Thomee et al. 2010; Amato et al. 2013]. Search results obtained using BoW in CBIR has also been improved by exploiting additional geometrical information and applying re-ranking approaches [Philbin et al. 2007; Tolias and Jégou 2013]. To overcome the loss in information about the original descriptors due to the quantization process, more accurate representation of the original descriptors and alternative encoding techniques were proposed [Philbin et al. 2008; Van Gemert et al. 2010; Jégou et al. 2010].

Recently, other encoding schemes such as the *Fisher Vector* (FV) [Perronnin and Dance 2007] and the *Vector of Locally Aggregated Descriptors* (VLAD) [Jégou et al. 2010] have attracted attention due to their effectiveness in both image classification and large-scale image search. The FV approach transforms an incoming set of descriptors into a fixed-size vector representation that is compatible with the cosine similarity. The vector representation is built by characterizing how a sample of descriptors devi-

ates from an average distribution that might be understood as a "probabilistic visual vocabulary". The Gaussian Mixture Model (GMM) [McLachlan and Peel 2000] is usually used as average distribution.

While BoW counts the occurrences of visual words and so takes in account just 0-order statistics, the FV offers a more complete representation by encoding higher order statistics (first and optionally second order) related to the distribution of the descriptors. FV results also in a more efficient representation, since fewer visual words are required in order to achieve a given performance. However, the vector representation obtained using BoW is typically quite sparse while that obtained using the Fisher Kernel is almost dense. This leads to some storage and input/output issues that have been addressed by using techniques of dimensionality reduction such as Principal Component Analysis (PCA) [Bishop 2006], compression with product quantization [Gray and Neuhoff 1998; Jégou et al. 2011] and binary codes [Perronnin et al. 2010a].

Similarly to BoW, the VLAD method uses a visual vocabulary to quantize the local descriptors of an image. Differently from BoW, VLAD encodes the accumulated difference between the visual words and the associated descriptors and so exploits more aspects of the distribution of the descriptors assigned to each visual word. Initially [Jégou et al. 2010], VLAD descriptors were L_2-normalized. Subsequently a power normalization step was introduced for both VLAD and FV [Jégou et al. 2012; Perronnin et al. 2010a]. Furthermore, PCA dimensionality reduction and product quantization were applied and several enhancements to the basic VLAD were proposed [Arandjelovic and Zisserman 2013; Chen et al. 2011; Delhumeau et al. 2013; Zhao et al. 2013].

Recently, a new class of image descriptors computed using Deep Convolutional Neural Networks (CNNs) has been used as effective alternative to descriptors built upon local features. Starting from 2012 [Krizhevsky et al. 2012], Deep Convolutional Neural Networks have attracted enormous interest within the Computer Vision community because of the state-of-the-art results obtained by CNNs approach in image classification. Moreover, the image representations obtained using CNN have been shown to be effective also for image search and object recognition, and not only classification tasks. In [Donahue et al. 2013], semi-supervised multi-task learning of deep convolutional representations were investigated. In particular, it has been proven that activations produced by an image within the top layers of the CNN can be used as a high-level descriptor. Following this approach, in [Razavian et al. 2014] several experiments were conducted for different recognition tasks (object image classification, scene recognition, fine grained recognition, attribute detection and image retrieval). In [Tolias et al. 2015], filtering and re-ranking stages of object retrieval were revisited by employing CNN activations of convolutional layers to derive representations for image regions.

One limitation of the feature vectors built upon CNN, unlike the descriptors built upon local features, is that the CNN pipeline do not ensure robustness to transformation such as rotation or scale changes, which are crucial to instance retrieval tasks. In [Chandrasekhar et al. 2015] a fusion of FV and CNN features have been investigated to improve retrieval results and balance the lack of geometrical invariance of CNN.

In this work, we discuss and compare several image representations that can be used to perform the search of cultural heritage objects. In particular, we thoroughly test FV, CNN, and their mixture to visually recognize ancient inscriptions. Finally, we compare the obtained performances with the state-of-the-art results previously achieved in this context by using BoW and VLAD [Amato et al. 2014].

3. IMAGE REPRESENTATIONS

To compare the visual content of two or more images, in order to decide if they contain the same object, one needs to use an appropriate numerical representation of each image. This section introduces some of the most prominent approaches to transform an input image into a fixed-length representation. Specifically the Bag-of-Words (BoW), Vector of Locally Aggregated Descriptor (VLAD), Fisher Vector

(FV) and Convolutional Neural Network (CNN) techniques are presented. BoW, VLAD and FV are computed by aggregating local features extracted from images. CNN vectors are directly extracted from images by using deep convolutional neural networks.

In the following we briefly introduce local features and their encoding by using BoW and VLAD. Then we present FV and CNN features.

3.1 Local Features

Local features are numerical representation of local structures of images that have been widely used to support image retrieval and object recognition tasks. Local features are extracted in two phases: first, a set of interest points, referred to as *keypoints*, are automatically selected; then one or more descriptors are associated with each keypoint. A local feature is generally a histogram representing statistics of the pixels in the neighborhood of an interest point. In our test we used the Scale Invariant Feature Transformation (SIFT) [Lowe 2004] that is the most cited and used local feature to date thanks to its effectiveness. However, executing image retrieval and object recognition tasks relying on local features is generally resource demanding. In fact, each digital image, both queries and images in the digital archives, are typically described by thousands of local descriptors. In order to decide that two images match, since they contain the same or similar objects, local descriptors in the two images need to be pairwise compared to identify matching patterns.

Encoding techniques such as Bag-of-Words, VLAD and Fisher Vector are used to summarize the information provided by all the local features extracted from an image. The resulting image representations have been proved to be effective for image comparisons and lead to reduce the cost of image search on a very large scale.

3.2 Bag-of-Words

The Bag of (Visual) Words (BoW) was initially proposed in [Sivic and Zisserman 2003] for matching objects throughout a video database. The approach was inspired by the BoW model used in text retrieval. Thereafter, BoW has been widely used for classification and CBIR tasks [Csurka et al. 2004; Jégou et al. 2010]. BoW uses a "visual vocabulary" to represent each image as a set (bag) of visual words. The visual vocabulary is built by clustering the local descriptors of a dataset, e.g. by using k-means [Lloyd 1982]. The cluster centers, named *centroids*, act as the *visual words* of the vocabulary and they are used to quantize the local descriptors extracted from images. Specifically, each local descriptor of an image is assigned to its closest centroid and the image is represented by a histogram of occurrences of the visual words.

The retrieval phase is performed using text retrieval techniques, where visual words are used in place of text word and considering a query image as disjunctive term-query. Typically, the cosine similarity measure in conjunction with a term weighting scheme, e.g. *term frequency-inverse document frequency* (tf-idf) [Salton and McGill 1986], is adopted for evaluating the similarity between any two images.

3.3 Vector of Locally Aggregated Descriptors

The Vector of Locally Aggregation Descriptors (VLAD) was initially proposed in [Jégou et al. 2010]. As for the BoW, a visual vocabulary $\{\mu_1, \ldots, \mu_K\}$ is first learned using a clustering algorithm (e.g. k-means). Each local descriptor x_t of a given image is then associated with its nearest centroid $NN(x_t)$ in the vocabulary. For each centroid μ_i the differences $x_t - \mu_i$ of the vectors x_t assigned to μ_i are accumulated:

$$v_i = \sum_{x_t : NN(x_t) = \mu_i} x_t - \mu_i. \tag{1}$$

Finally, the accumulated residuals v_i are concatenated into a single vector $V = [v_1^\top \dots v_K^\top]$ referred to as VLAD. All the residuals have the same size D which is equal to the dimensionality of the local features.

Thus the dimensionality of the whole vector V is fixed too and it is equal to DK. Power-law and L_2 normalization are usually applied and Euclidean distance has been proved to be effective for comparing two VLADs [Jégou et al. 2012; Arandjelovic and Zisserman 2013]. Since VLAD descriptors have high dimensionality, Principal Component Analysis (PCA) can been used to obtain a more compact representation [Jégou et al. 2010].

3.4 Fisher Vector

The Fisher Kernel is a powerful framework introduced in [Jaakkola and Haussler 1998] for classifying DNA splice site sequences and to detect homologies between protein sequences. In [Perronnin and Dance 2007], the Fisher Kernel method was adopted in the context of image classification as efficient tool to encode image local descriptors into a fixed-size vector representation.

The main idea of this method is to derive a kernel function to measure the similarity between two sets of data such as the sets of local descriptors extracted from two images. Specifically, the similarity of two sample sets X and Y is measured by analyzing the difference between the statistical properties of X and Y, rather than comparing directly X and Y. To this scope a probability distribution $p(\cdot|\lambda)$ with some parameters $\lambda \in \mathbb{R}^m$ is first estimated on a large training set and is used as "average distribution" over the space of all the possible data observations. Then each sample $X = \{x_1, \dots, x_T\}$ is represented by a vector, named *Fisher Vector*, that indicates the direction in which the parameter λ of the probability distribution $p(\cdot|\lambda)$ should be modified to best fit the data in X. In this way, two samples are considered similar if the directions given by their respective Fisher Vectors are similar. Specifically, as proposed in [Jaakkola and Haussler 1998; Perronnin and Dance 2007], the similarity between two sample sets X and Y is measured using the Fisher Kernel, defined as

$$K(X, Y) = (\mathcal{G}_\lambda^X)^\top \mathcal{G}_\lambda^Y, \tag{2}$$

where $\mathcal{G}_\lambda^X = L_\lambda \nabla_\lambda \log p(X|\lambda)$ and L_λ is the square root of the inverse of the Fisher Information Matrix (see [Sànchez et al. 2013] for more details). The vector $\mathcal{G}_\lambda^X$ is referred to us the *Fisher Vector* (FV) of X.

Note that the FV is a fixed size vector whose dimensionality only depends on the dimensionality m of the parameter λ. The FV is further divided by T in order to avoid the dependence on the sample size [Sànchez et al. 2013]. Moreover, in the context of image retrieval and classification the FV is usually L_2-normalized because this is a way to cancel-out the fact that different images contain different amounts of image-specific information (e.g. the same object at different scales)[Perronnin et al. 2010b; Sànchez et al. 2013].

In this work, as in [Perronnin and Dance 2007], we choose $p(\cdot|\lambda)$ to be a Gaussian Mixture Model (GMM) of parameter $\lambda = \{w_k, \mu_{kd}, \Sigma_k = \text{diag}(\sigma_{k1}, \dots, \sigma_{kD}),\ k = 1, \dots, K,\ d = 1, \dots, D\}$, where K is the number of Gaussian, D is the dimension of each local descriptor, and w_k, μ_k, Σ_k are respectively the mixture weight, mean vector and covariance matrix of k-th Gaussian. By using the GMM model, the FV of a set of D-dimensional local descriptor $X = \{x_1, \dots, x_T\}$ is obtained as the concatenation of the

vector $\mathcal{G}_{\alpha}^{X} \in \mathbb{R}^{K}$, $\mathcal{G}_{\mu}^{X} \in \mathbb{R}^{KD}$, $\mathcal{G}_{\sigma}^{X} \in \mathbb{R}^{KD}$, computed as

$$\mathcal{G}_{\alpha_k}^{X} = \frac{1}{T\sqrt{w_k}} \sum_{t=1}^{T} (\gamma_t(k) - w_k) \qquad k = 1, \ldots, K \tag{3}$$

$$\mathcal{G}_{\mu_{kd}}^{X} = \frac{1}{T\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) \frac{x_{td} - \mu_{kd}}{\sigma_{kd}} \qquad k = 1, \ldots, K,\ d = 1, \ldots, D \tag{4}$$

$$\mathcal{G}_{\sigma_{kd}}^{X} = \frac{1}{T\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) \frac{1}{\sqrt{2}} \left[\frac{(x_{td} - \mu_{kd})^2}{(\sigma_{kd})^2} - 1 \right] \qquad k = 1, \ldots, K,\ d = 1, \ldots, D \tag{5}$$

where $\gamma_t(k) = p(k|x_t, \lambda)$ is the probability for the observation x_t to be generated by the k-th Gaussian. The whole FV is of dimension $(2D+1)K$. However, the FV is often used considering only the sub-vector associated with the mean parameters ($\mathcal{G}_{\mu}^{X}$) whose dimensionality is KD [Perronnin et al. 2010a; Jégou et al. 2010; Jégou et al. 2012].

3.5 Convolutional Neural Network

CNNs are neural networks specialized for data that has a grid-like topology like image data. The applied discrete convolution operation results in a multiplication by a matrix which has several entries constrained to be equal to other entries. Three important ideas are behind the success CNNs: sparse connectivity, parameter sharing, and equivalent representations [Goodfellow et al. 2016].

In image retrieval, Deep CNN have been successfully adopted using the activations produced by an image within the top layers of the CNN as a high-level descriptor of the visual content of the image [Donahue et al. 2013]. In [Razavian et al. 2014] the same approach was adopted for evaluating the CNN representation in visual instance retrieval tasks. The results confirmed that the activations produced within the top layers of the CNN, compared by using the Euclidean distance, achieve state-of-the-art quality in terms of mAP.

Most of the papers reporting results obtained using features built upon CNN, maintain the REctified Linear Unit (RELU) transform [Donahue et al. 2013; Razavian et al. 2014; Chandrasekhar et al. 2015], i.e., negative activations values are discarded replacing them with 0. In our experiments, we also reported the results obtained without the RELU as in [Babenko et al. 2014]. In fact, while the RELU, being a non-linear operation, has been proved to be very effective as activation, the negative values discarded by using this operation could also be exploited in the visual feature. Values are typically L_2 normalized [Babenko et al. 2014; Razavian et al. 2014; Chandrasekhar et al. 2015] and we do the same in this work.

In Section 4.2, we describe the various CNN models, the training data and the layers used as features for our experiments.

4. EXPERIMENTAL EVALUATION

In this section we compare the performance of the techniques described in the previous section in order to identify the most effective methods to perform visual recognition of ancient inscriptions. In particular we evaluate the performance of FV and CNN features and compare them with the state-of-the-art results obtained using BoW and VLAD approaches [Amato et al. 2014]. First, we introduce the used dataset and we describe the experimental setup. Then, we report results and their comparison.

Fig. 3. Example of queries and their associated images in the ground truth.

4.1 Dataset and Ground Truth

The tests were conducted using the *Epigraphic Database Rome* (EDR)[1] that is part of the international federation of Epigraphic Databases called Electronic Archive of Greek and Latin Epigraphy (EAGLE)[2]. The EDR dataset is composed of $17,155$ photos related to $14,560$ inscriptions, so in most cases just one photo is provided for each inscription object.

To carry out the performance analysis, we selected 70 queries, i.e. images to be recognized, and we built a ground truth. For each query, the ground truth contains all the images of the dataset that represent the same object of the query. During the retrieval tests we removed the query photos from the knowledge base, so we selected as query only inscriptions that have more than one photo in the dataset. Furthermore, the queries were carefully selected in order to represent the various types of inscriptions contained in the dataset (as, for example, inscription with different state of preservation or incised on different material). Figure 3 shows four query examples together with their related images in the dataset, i.e. image that contains the same query object.

4.2 Experimental Settings

In the following we report some details on how the features for the various approaches were extracted. Moreover, we describe how we combined FV and CNN features.

[1] http://www.edr-edr.it/English/index_en.php
[2] http://www.eagle-network.eu/

Local features: We extracted SIFT [Lowe 2004] local features from each image by using OpenCV (Open Source Computer Vision Library)[3].

We obtained an average of 1,591 SIFT per image. However, the information about the scale at which the features were extracted allows us to select a subset of local features that are in principle more relevant. In fact, features detected at higher scale refer to bigger regions than others and should also be present at lower resolution versions of the same image or of the same object. Thus, the criterion that the bigger the scale the higher the importance can be used to perform feature selection [Amato et al. 2013]. In the experiments, we reduced the number of local features by selecting the 250 features detected at highest scales. We refer to the latter approach as *reduced-keypoints*. The feature selection was not applied whenever the number of features extracted from an image was already less than 250.

Subsequently, the local SIFT descriptors are reduced from 128 to 64 components by using PCA. The PCA rotation matrix was learned on about 2M of local features randomly selected from the whole dataset.

GMM and Fisher Vector: The Gaussian Mixture Model and the Fisher Vector representation were computed by using our Visual Information Retrieval library that is publicly available on GitHub[4]. The parameter $\lambda = \left\{w_k, \mu_{kd}, \sigma_{kd}\right\}_{k=1,\ldots,K,\, d=1,\ldots,D}$ of the GMM (where K is the number of mixture components and D is the dimension of each local descriptor) were learned by optimizing a maximum-likelihood criterion with the Expectation Maximization (EM) algorithm [Bishop 2006]. EM is an iterative method that is deemed to have converged when the change in the likelihood function, or alternatively in the parameters λ, falls below some threshold ϵ. As stopping criterion for the GMM estimation we used the convergence in L_2-norm of the mean parameters, choosing $\epsilon = 0.05$. As suggested in [Bishop 2006], the GMM parameters used in EM algorithm were initialized with: (a) $1/K$ for the mixing coefficients w_k; (b) centroids precomputed using k-means for the GMM means μ_{kd}; (c) mean variance of the clusters found using k-means, for the diagonal elements σ_{kd} of the GMM covariance matrices. Both the k-means and the GMM estimation were performed using in order of 1M local descriptors randomly selected from the whole dataset. As a common post-processing step [Perronnin et al. 2010b; Jégou et al. 2012], the FVs were power-law normalized and subsequently L_2-normalized. The power-law normalization is parametrized by a constant β and it is defined as $x \rightarrow |x|^{\beta}\text{sign}(x)$. In our experiments we used $\beta = 0.5$.

CNN features: In this work we tested four different pre-trained CNN models, downloaded from the Caffe Model Zoo[5]:

—*OxfordNet [Simonyan and Zisserman 2014].* This is an improved version of the model used by the VGG team in the ILSVRC-2014 competition. The model was trained on $1,000$ categories of ImageNet [Deng et al. 2009] with about 1.5 million images and it contains 16 weight layers (13 convolutional + 3 fully-connected). The input image are fixed-size to 224×224 RGB.

—*AlexNet (BVLC Reference CaffeNet).* This model mimics the original AlexNet [Krizhevsky et al. 2012], with minor variations as described in [Jia et al. 2014]. The model was trained on ImageNet with about 1.5 million images and it has 8 weight layers (5 convolutional + 3 fully-connected). The input image are fixed-size to 227×227 RGB.

[3]http://opencv.org/
[4]https://github.com/ffalchi/it.cnr.isti.vir
[5]https://github.com/BVLC/caffe/wiki/Model-Zoo

—*PlacesNet [Zhou et al. 2014].* PlaceNet model shares the same architecture of BVLC Reference CaffeNet, while being trained on 205 scene categories of Places Database [Zhou et al. 2014] with about 2.4 million images.

—*HybridNet [Zhou et al. 2014].* The architecture of HybridNet is the same as the BVLC Reference CaffeNet. The model was trained on $1,183$ categories (205 scene categories from Places Database and 978 object categories from the train data of ImageNet) with about 3.6 million images.

In the test phase we used Caffe and, for each model, we extracted the output of the last convolutional layer after pooling (*pool5*) and the first two fully-connected layers (*fc6, fc7*). The only preprocessing we did is resizing the input images to the canonical resolution and then subtracting (from each pixel) the mean RGB value $(104, 117, 123)$ computed on ImageNet. All the descriptor are L_2 normalized. $pool5$ still contains spatial information from the input image, however it is very high dimensional ($25,088$ components for OxfordNet and $9,216$ components for AlexNet/PlacesNet/HybridNet). $fc6$ and $fc7$ are $4,096$-dimensional vectors.

Combination of FV and CNN features: FVs and outputs of intermediate layers of CNN have complementary behavior under some image transformations. In fact, the FVs (computed from SIFT or SIFTPCA) are robust to image rotation while the CNN features have limited level of rotation invariance. Additionally, in [Chandrasekhar et al. 2015] extensive experiments on benchmark dataset for image retrieval have showed that CNN features generally are less affected by small scale changes than FV. In order to leverage the positive aspects of both these methods, in [Chandrasekhar et al. 2015] a fusion of FV and CNN features has been proposed.

In this paper, we evaluated the combination of FV and CNN features using the following approach. Each image was represented by a couple (c, f), where c and f were respectively the CNN descriptor and the FV descriptor of the image. Then, we evaluated the distance d between two couples (c_1, f_1) and (c_2, f_2) as the convex combination of the L_2 distances of the CNN descriptors (i.e. $\|c_1 - c_2\|_2$) and the FV descriptors (i.e. $\|f_1 - f_2\|_2$). In other words we defined

$$d\big((c_1, f_1), (c_2, f_2)\big) = \alpha \|c_1 - c_2\|_2 + (1 - \alpha) \|f_1 - f_2\|_2 \tag{6}$$

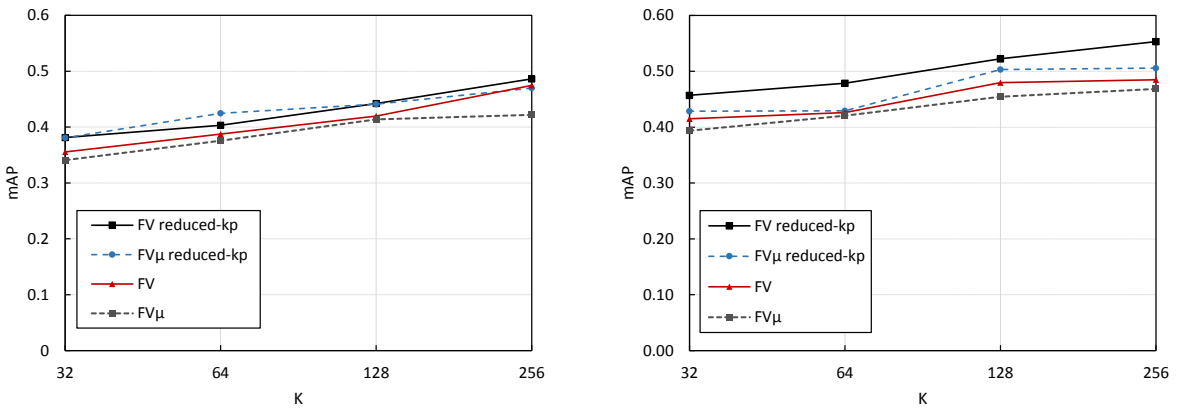
with $0 \leq \alpha \leq 1$. Choosing $\alpha = 0$ corresponds to use only FV approach, while $\alpha = 1$ corresponds to use only CNN features.

4.3 Performance Measures

As in [Amato et al. 2014], in order to recognize the actual object in a query image, we basically perform a visual similarity search between all the images in the dataset. When examining the ranked result list of a query it is evident that the greater the ranked position of a relevant image (i.e. an image of the same query object) the less valuable it is for the user, because the less likely it is that the user will examine the image. Thus, the main goal is to have one relevant image as first result. Whenever this is not the case, it is interesting to understand at which position in the result list the most visually similar photo of the query object appears. In fact, re-ranking techniques could be applied on the results list in order to achieve better effectiveness. Therefore we report, for each technique, the probability p of finding an image of the same query object within the first r results, varying r between 1 and 100. Specifically, $p(r)$ is defined as

$$p(r) = P(R \leq r), \tag{7}$$

where R is the random variable denoting the position of the first relevant image in the ranked result list of a query. For $r = 1$, p also equals the accuracy of a classifier that recognizes the query inscription as the most similar that have been found. In the experiments, for each query q_i we calculate the position



(a) SIFT (b) SIFTPCA64

Fig. 4. mAP for various Fisher Vector representations, varying the number K of mixture components of the GMM. FV indicates the full-sized Fisher Vector, while FV_μ is the Fisher Vector referred only to the mean values of the GMM. The representations are computed both using all the keypoints extracted from images and the reduced-keypoint approach. In (a) we report the results obtained using SIFT descriptors; in (b) we report the results obtained with SIFTPCA64 descriptors, i.e. SIFT reduced to 64 dimensions by means of PCA

r_{q_i} of the first relevant image, and we estimated the probability $p(r)$ as $\sum_{i=1}^{N} [\![r_{q_i} \leq r]\!]/N$, where N is total the number of queries and $[\![\cdot]\!]$ represent the Iverson bracket which equals one if the arguments is true, and zero otherwise.

The retrieval performance of each method was measured also by the mean average precision (mAP), with the query removed from the ranking list. During the mAP computation, not just the first relevant image but all the images associated with the query are considered. Therefore, while $p(r)$ measures how good each method is in reporting at least one relevant image in the first r positions, the mAP reveal how good each method is in reporting all the relevant images in the top positions of the result list.

4.4 Results

In the following, we report the results of extensive tests on FV and CNN approaches to visually recognize ancient inscriptions (Sections 4.4.1 and 4.4.2). Also the combination of FV and CNN features into a single image representation is taken into account (Section 4.4.3). Finally, we compare our best results with the state-of-the-art retrieval performances achieved in [Amato et al. 2014] using BoW and VLAD image representations (Section 4.4.4). Both FV and CNN approaches outperformed BoW and VLAD in recognizing ancient inscriptions. However, the use of a combination of FV and CNN features have led to obtain the best retrieval performances.

4.4.1 ***Fisher Vector***. We evaluated the performance of FV approach for various settings. We built the FV both using full-size SIFT local descriptor and SIFTPCA64, i.e. SIFT reduced to 64 dimensions by means of PCA. We also tested the reduced keypoint approach based on the scale selection, as described in Section 4.2. We varied the number K of Gaussian mixtures, considering $K = 32, 64, 128, 256$. Finally, we evaluate the performance of both the whole FV and the FV related just to the mean vectors (as defined in equation 4) that we indicated with FV_μ. In fact, in literature [Perronnin et al. 2010a; Jégou et al. 2010; Jégou et al. 2012] the FV is often used cosidering the componets associated with the mean parameters only since it results in a more compact vector represenation.

Table I. Performance of various Fisher Vector representations. FV is the full-size Fisher Vector, while FV_μ is the Fisher Vector referred only to the mean values of the GMM. All the FVs are computed by using SIFTPCA64 (i.e. SIFT descriptors reduced to 64 dimensional vectors by means of PCA). The *Reduced-keypoints* column indicates if the local feature selection was used. K is the number of mixture components of the GMM. *dim* and *bytes* are respectively the number of components and the average size in bytes of each vector representation. The results are ordered with respect to the mAP quality measure. Bold numbers denote maxima in the respective column.

Method	**Reduced keypoints**	**K**	**dim**	**bytes**	**mAP**	**p** $(r = 1)$	**p** $(r = 10)$	**p** $(r = 100)$
FV	×	256	33,024	132,096	**0.55**	**0.73**	0.76	0.87
FV	×	128	16,512	66,048	0.52	0.69	0.76	0.84
FV_μ	×	256	16,384	65,536	0.51	0.69	0.77	0.86
FV_μ	×	128	8,192	32,768	0.50	0.67	0.74	0.80
FV		256	33,024	132,096	0.49	0.66	**0.79**	**0.94**
FV		128	16,512	66,048	0.48	0.64	0.76	0.93
FV	×	64	8,256	33,024	0.48	0.63	0.76	0.83
FV_μ		256	16,384	65,536	0.47	0.64	0.73	0.89
FV	×	32	4,128	16,512	0.46	0.59	0.73	0.83
FV_μ		128	8,192	32,768	0.45	0.60	0.73	0.89
FV_μ	×	32	2,048	8,192	0.43	0.57	0.66	0.79
FV_μ	×	64	4,096	16,384	0.43	0.57	0.71	0.81
FV		64	8,256	33,024	0.43	0.57	0.74	0.89
FV_μ		64	4,096	16,384	0.42	0.56	0.64	0.84
FV		32	4,128	16,512	0.42	0.59	0.67	0.83
FV_μ		32	2,048	8,192	0.39	0.59	0.67	0.79

Table II. FV: performance comparison after dimensionality reduction with PCA. The FV was computed by using SIFTPCA64 with reduced-keypoints. K is the number of mixture components of the GMM. *dim* and *bytes* are respectively the number of components and the average size in bytes of each vector representation. Bold numbers denote maxima in the respective column. The fist colums is reported from Table I for reference.

Method	**Reduced keypoints**	**K**	**dim**	**bytes**	**mAP**	**p** $(r = 1)$	**p** $(r = 10)$	**p** $(r = 100)$
FV	×	256	33,024	132,096	**0.55**	**0.73**	0.76	**0.87**
			PCA→ 8,192	32,768	0.54	0.71	0.74	0.86
			PCA→ 4,096	16,384	**0.55**	0.70	**0.77**	0.86
			PCA→ 2,048	8,192	0.51	0.66	0.73	0.83
			PCA→ 1,024	4,096	0.40	0.51	0.66	0.71

Figure 4 shows the mAP for FV and FV_μ varying the number K of Gaussian and different local feature setting. As expected the bigger K the better the performance. However, we did not take into account K bigger that 256 because the dimensionality of the resulting FV would be very large and the estimation of the GMM expensive. The Fisher Vectors computed using SIFTPCA64 are more compact and more effective than the respective vectors computed from SIFT, so in the following we analyze just the results obtained using SIFTPCA64. The whole FV performs better than FV_μ and the use

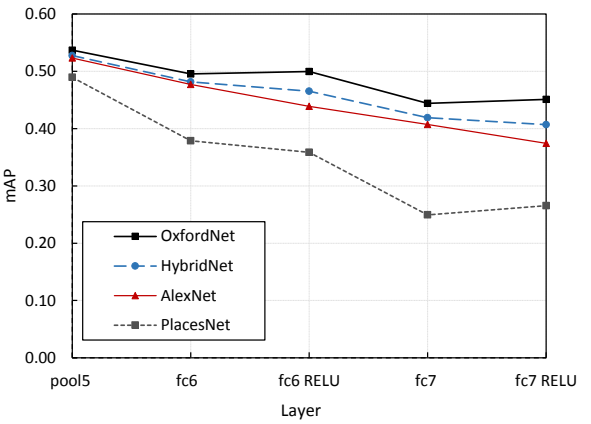


Fig. 5. mAP for last convolutional layer (*pool5*) and the first two fully-connected layers (*fc6, fc7*) of the state-of-the-art CNN: *OxfordNet*, *HybridNet*, *AlexNet* and *PlacesNet*. The outputs of *fc6* and *fc7* layers were analyzed both before and after applying the REctified Linear Unit transform (RELU).

of keypoint-reduction technique further improves the results. Thus, the overall best mAP was $\mathbf{0.55}$, obtained using FV with $K = 256$ and reduced-keypoints.

In table I, we summarize the obtained mAP and probabilities p of finding at least one relevant image between the first r results, with $r = 1, 10, 100$. The results show that the keypoint reduction is in general useful, especially according to the mAP and $p(r = 1)$ quality measures. For example, in the 73% of cases, the full-size FV with $K = 256$ and reduced-keypoints correctly recognized the query object as the first result. However, the use of all the extracted local features, respect to the keypoint reduction approach, has led to obtain better probabilities p for big value of r (e.g. in the 94% of the cases the FV with $K = 256$ recognized the query object between the top 100 positions of the result list while the FV with $K = 256$ and reduced-keypoints reached a probability of 87%).

We already observed that for the same K the FV outperforms FV_μ and that the performances increases with increasing K. However, in order to limit the size of the Fisher Vector representation, in literature, the FV_μ have been usually preferred to the full-size FV. This is not our case, because for the same size of the final vector representation and for the same used local features, the whole FV has similar performance to that of FV_μ also if the last uses bigger K. For example, FV with $K = 128$ and FV_μ with $K = 256$ have quite the same dimension (about $16,400$ components) and similar mAP ($0.52/0.51$) and probabilities p. However, the cost of learning the GMM and computing the FV increases with K, so in our case it would be advisable to use the whole FV (with smaller K) than FV_μ.

The performances obtained using 256 mixtures of Gaussian are promising, but the resulting FV is very high-dimensional. In order to reduce the cost of storing and comparing FV, we also evaluated the effect of PCA-dimensionality reduction. Table II shows the results for the PCA-reduced version of the FV computed using 256 mixtures of Gaussian and reduced-keypoints. It is worth noting that the size of the FV could be effectively reduced by 88% (from $33,024$ to $4,096$ components) maintaining performance basically unchanged. Thus, it is clearly convenient to use FV in conjunction with PCA dimensionality reduction.

4.4.2 ***CNN Features***. In figure 5 and table III we report the results obtained using the outputs of the last convolutional layer (*pool5*) and the first two fully-connected layers (*fc6, fc7*) of the following state-of-the-art pre-trained CNNs: *OxfordNet*, *HybridNet*, *AlexNet* and *PlacesNet* (see Section 4.2).

Table III. Performance comparison of different layers of *OxfordNet*, *HybridNet*, *AlexNet* and *PlacesNet*. *dim* and *bytes* are respectively the number of components and the average size in bytes of each vector representation. Results are ordered with respect to the mAP measure. Bold numbers denote maxima in the respective column.

Method	**Layer**	**dim**	**bytes**	**mAP**	**p** ($r = 1$)	**p** ($r = 10$)	**p** ($r = 100$)
OxfordNet	pool5	25,088	100,352	**0.54**	**0.66**	0.77	**0.93**
HybridNet	pool5	9,216	36,864	0.53	**0.66**	0.81	0.90
AlexNet	pool5	9,216	36,864	0.52	**0.66**	0.81	0.89
OxfordNet	fc6 RELU	4,096	16,384	0.50	0.64	**0.84**	0.91
OxfordNet	fc6	4,096	16,384	0.50	0.63	0.80	**0.93**
PlacesNet	pool5	9,216	36,864	0.49	0.64	0.77	0.90
HybridNet	fc6	4,096	16,384	0.48	0.59	0.80	**0.93**
AlexNet	fc6	4,096	16,384	0.48	0.59	0.83	0.87
HybridNet	fc6 RELU	4,096	16,384	0.47	0.56	0.76	0.90
OxfordNet	fc7 RELU	4,096	16,384	0.45	0.59	0.74	0.86
OxfordNet	fc7	4,096	16,384	0.44	0.56	0.76	0.89
AlexNet	fc6 RELU	4,096	16,384	0.44	0.54	0.79	0.86
HybridNet	fc7	4,096	16,384	0.42	0.54	0.76	0.90
AlexNet	fc7	4,096	16,384	0.41	0.50	0.71	0.84
HybridNet	fc7 RELU	4,096	16,384	0.41	0.53	0.71	0.87
PlacesNet	fc6	4,096	16,384	0.38	0.49	0.64	0.90
AlexNet	fc7 RELU	4,096	16,384	0.37	0.44	0.69	0.86
PlacesNet	fc6 RELU	4,096	16,384	0.36	0.49	0.66	0.84
PlacesNet	fc7 RELU	4,096	16,384	0.27	0.49	0.64	0.90
PlacesNet	fc7	4,096	16,384	0.25	0.49	0.64	0.90

The outputs of *fc6* and *fc7* layers are analyzed both before and after applying the REctified Linear Unit (RELU) transform.

OxfordNet exhibits the overall best performance, followed by HybridNet and AlexNet. PlacesNet, instead, has lowest mAP results. Let us remind that HybridNet and PlacesNet share the same architecture of AlexNet, while being trained on different datasets: AlexNet is trained on 1.2 million images of ImageNet, PlacesNet is trained on 2.4 million images of Places Database, and HybridNet is trained on both the previous datasets. The ImageNet is more object-centric than Places dataset, so it could be considered more appropriate for our test data that contains a lot of photos of peculiar objects (inscriptions) rather than scenes. Thus, our results confirm the fact already pointed out in [Chandrasekhar et al. 2015], that an appropriate choice of the training dataset may improve retrieval performance significantly. In facts, in our case, the models trained on ImageNet perform better than the one trained on Places Dataset.

This suggests that results could be further improved if an epigraphic-related dataset is used for training or fine-tuning the CNN models. However, in our tests we used just pre-trained models for the following reasons. First, a large amount of labeled data is needed to train the networks and we had not access to such amount of epigraphic labeled images. The EDR dataset is not suitable for learning, nor for model fine-tuning because in most cases contains just one image for each inscription. Moreover, several research articles [Razavian et al. 2014; Girshick et al. 2014; Chandrasekhar et al. 2015] have

Table IV. OxfordNet CNN: performance comparison after PCA dimensionality reduction. Results for both *pool5* and *fc6 RELU* layers are reported. *dim* and *bytes* are respectively the number of components and the average size in bytes of each vector representation. The results related to the full-size features (i.e. the *pool5* and *fc6* features before PCA-reduction) are reported from Table III for reference. Bold numbers denote maxima in the respective column and for each approch.

Method	**Layer**	**dim**	**bytes**	**mAP**	**p** $(r = 1)$	**p** $(r = 10)$	**p** $(r = 100)$
OxfordNet	pool5	25,088	100,352	0.54	0.66	0.77	**0.93**
		PCA→ 4,096	16,384	**0.57**	**0.70**	0.80	**0.93**
		PCA→ 2,048	8,192	**0.57**	**0.70**	**0.83**	**0.93**
		PCA→ 1,024	4,096	0.55	0.67	0.81	0.90
		PCA→ 512	2,048	0.53	0.66	0.81	0.90
		PCA→ 256	1,024	0.50	0.63	0.77	0.87
OxfordNet	fc6 RELU	4,096	16,384	0.50	0.64	**0.84**	0.91
		PCA→ 2,048	8,192	0.52	0.66	**0.84**	0.91
		PCA→ 1,024	4,096	**0.53**	**0.67**	**0.84**	0.91
		PCA→ 512	2,048	0.52	0.66	0.83	**0.94**
		PCA→ 256	1,024	0.51	0.63	0.81	**0.94**

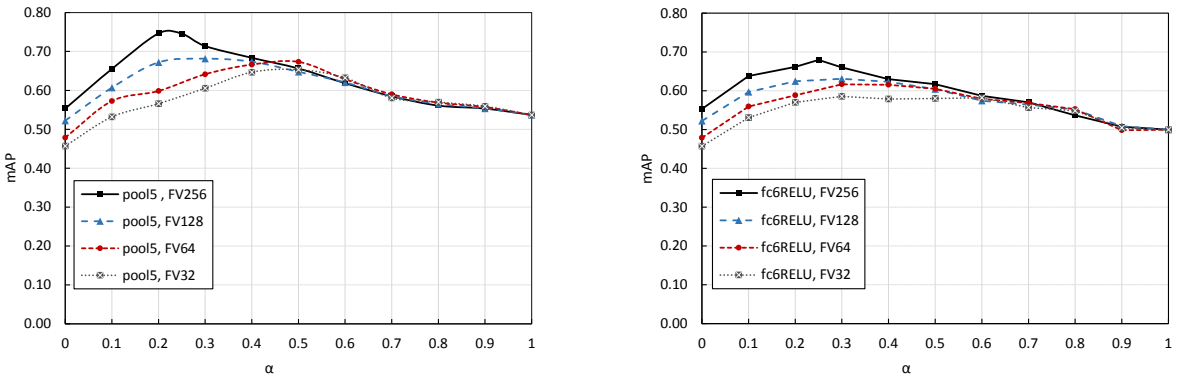
shown that CNN off-the-shelf features perform well for visual recognition tasks even without using fine-tuning on domain-specific dataset.

In our test, RELU transform improves the results just for the OxfordNet descriptors while the other CNNs have better performance by extracting the descriptor without applying the RELU transform. Layer *pool5* performs the best for all CNNs and the performance drops with increasing in depth. Deep learning methods learn representations of data with multiple levels of abstraction: the higher the level, the bigger the abstraction [Goodfellow et al. 2016]. As mentioned before, the HybridNet from which we extracted the activation was trained on the ImageNet and Places dataset. Thus, the higher layers are not only more abstract but also more specific for the tasks on which it has been trained. Our experiments show that lower level features as the ones extracted from *pool5* are more appropriate for the ancient inscription recognition task. It is worth to mention that the dimensionality of *pool5* is higher and thus the extracted feature larger.

In summary, the best results are achieved by OxfordNet *pool5* with a mAP of $\mathbf{0.54}$ and probability $p(r)$ equals to 66%, 77%, 93% respectively for $r = 1$, 10, 100. These results are similar to that of HybridNet *pool5* (mAP 0.53) and AlexNet *pool5* (mAP 0.52). However, it is worth noting that OxfordNet *pool5* is high dimensional and its size is almost triple that of AlexNet/HybridNet *pool5*.

In table IV we analyze the effect of PCA dimensionality reduction for both OxfordNet *pool5* ad *fc6*. PCA reduction results effective since it can provide very compact image signatures without loss in accuracy. Conversely, limited reduction tends to improve accuracy for both *pool5* ad *fc6*. For example, Oxford *pool5* which originally has $25,088$ components and a mAP of 0.54, reaches a mAP of 0.57 when reduced to $2,048$ components and the probabilities p are improved after the dimensionality reduction. The *fc6 RELU* also benefits of PCA reduction: we obtained a mAP of 0.53 after reducing from $4,096$ to $1,024$ components. So, as in the case of FV, we deduce that it is convenient in term of both efficiency and effectiveness to use PCA-reduced version of the CNN features.

The results related to the full dimensional image descriptors, i.e. without PCA dimensionality reduction, show that FV approach slightly outperforms CNN features, in fact FV with $K = 256$ reaches a



(a) Combination of FVs and OxfordNet pool5

(b) Combination of FVs and OxfordNet fc6 RELU

Fig. 6. mAP for various combinations of FV and OxfordNet features. $\alpha = 0$ corresponds to use only FV, while $\alpha = 1$ corresponds to use only the OxfordNet feature. The FV representations are computed varying the number K of Gaussian mixture (for $K = 32, 64, 128, 256$) and using SIFTPCA64 with reduced-keypoints. In (a) results for the combinations between FVs and OxfordNet pool5 are reported. Similarly, in (b) the results of the combinations between FVs and OxfordNet fc6 RELU are considered.

mAP of 0.55 while the best mAP obtained using CNN features is 0.54 (OxfordNet *pool5*). However, the CNN features have been revealed to be more robust to dimensionality reduction and so more suitable when very compact image descriptor are needed to reduce memory consumption. For example, OxfordNet *fc6* reduced to just 256 dimensional vector achieve a mAP of 0.51 that is 10% higher than the mAP achieved by FV256 reduced to the same dimension. In addition, it is interesting to note that CNN features take more advantage from PCA reduction, in fact OxfordNet *fc6* reduced to $2,048$ components (0.57 mAP) even outperform the overall best FV approach (0.55 mAP).

4.4.3 ***Combining FV and CNN Features***. In this section we report the results combining FV and CNN features as described in Section 4.2. Figure 6 shows the mAP obtained by combining FV with OxfordNet *pool5* and *fc6 RELU* features. The FVs were computed using SIFTPCA64 with reduced-keypoints and varying K from 32 to 256. The best results is a mAP of **0.75** obtained, as expected, by combining FV and CNN features with their respective best settings, i.e. FV with $K = 256$ (0.55 mAP) and OxfordNet *pool5* (0.54 mAP). All the combinations show an improvement with respect to the single use of CNN or FV features. It is interesting to note that, for appropriate value of α, there is valuable improvement also when CNN is combined with less effective Fisher Vector, such that obtained with small K. Since the cost for computing and storing FV is relatively low when small K are used, our results show that it is convenient to combine FV and CNN also when one want to contain costs of computing image representation. In this case, small K could be used during FV computation.

According to our results, seems that exists an optimal α for each combination of FV and CNN, i.e. a value such that mAP reach a maximum. The maximum performance was obtained for α between 0.2 and 0.3 when considering high number K of Gaussian mixtures (i.e. $K = 256, 128$). Considering that we are interested in finding the most prominent approach to perform recognition of ancient inscription, in the following we fixed $\alpha = 0.25$ and we focused on the combination of the best FV and CNN approaches, i.e. FV with $K = 256$ and OxfordNet *pool5* and *fc6 RELU*. Table V summarizes the obtained results and explores also the PCA-reduced version of FV and CNN features, since both benefit from PCA-reduction (see Section 4.4.1 and 4.4.2). However, for FV we did not consider reduction to less than

Table V. Performace of various mixtures of FV and OxfordNet CNN, by using $\alpha = 0.25$ in the convex combination of FV and CNN distances. Both full-sized and PCA-reduced features are considered. The FV representations are computed using SIFTPCA64 with reduced-keypoints. *dim* and *bytes* columns indicate respectively the number of components and the average size in bytes of each vector representation. For each approach, the bold numbers denote maxima in the respective column.

Method	**dim**	**bytes**	**mAP**	**p** ($r = 1$)	**p** ($r = 10$)	**p** ($r = 100$)
pool5, FV256	58,112	232,448	**0.75**	**0.93**	**0.97**	0.99
(pool5→PCA 2,048), FV256	35,072	140,288	0.70	0.86	0.96	**1.00**
(pool5→PCA 1,024), FV256	34,048	136,192	0.70	0.87	0.96	**1.00**
(pool5→PCA 512), FV256	33,536	134,144	0.67	0.84	0.94	**1.00**
pool5, (FV256→PCA 4,096)	29,184	116,736	0.74	0.91	0.96	0.99
(pool5→PCA 2,048), (FV256→PCA 4,096)	6,144	24,576	0.73	0.89	0.96	**1.00**
(pool5→PCA 1,024), (FV256→PCA 4,096)	5,120	20,480	0.73	0.89	0.96	**1.00**
(pool5→PCA 512), (FV256→PCA 4,096)	4,608	18,432	0.69	0.84	0.96	**1.00**
(fc6 RELU), FV256	37,120	148,480	**0.68**	**0.87**	0.94	**1.00**
(fc6 RELU→PCA 2,048), FV256	35,072	140,288	**0.68**	0.86	**0.97**	**1.00**
(fc6 RELU→PCA 1,024), FV256	34,048	136,192	**0.68**	0.86	0.94	**1.00**
(fc6 RELU→PCA 512), FV256	33,536	134,144	**0.68**	0.86	0.93	**1.00**
(fc6 RELU), (FV256→PCA 4,096)	8,192	32,768	0.67	0.84	0.94	**1.00**
(fc6 RELU→PCA 2,048), (FV256→PCA 4,096)	6,144	24,576	**0.68**	0.86	**0.97**	**1.00**
(fc6 RELU→PCA 1,024), (FV256→PCA 4,096)	5,120	20,480	**0.68**	0.84	**0.97**	**1.00**
(fc6 RELU→PCA 512), (FV256→PCA 4,096)	4,608	18,432	**0.68**	0.84	0.94	**1.00**

$4,096$ components since in these cases the mAP degraded as shown in Table II. Interestingly, for all the combinations we recognized the query as the first result between 84% and 93% of cases and almost always we correctly recognized the query at least in the 100 top positions, even if the dimension of the original descriptors is significantly reduced.

The combination of the OxfordNet *pool5*, reduced from $25,088$ to $1,024$ components, with the FV256, reduced from $33,024$ to $4,096$, could be considered as trade-off between efficiency and effectiveness since it has compact representation ($5,120$ components) and reaches very high performance (0.73 of mAP and $p(r)$ equals to 89%, 96%, 100% respectively for $r = 1$, 10, 100).

4.4.4 ***Summary and Comparison with the State-of-the-Art***. To the best of our knowledge, in literature, the topic of visual recognition of ancient inscriptions has been faced just in [Amato et al. 2014], where BoW and VLAD approaches have been analyzed. The experimental set up (dataset, ground truth, quality measures, local features extraction, etc.) used in [Amato et al. 2014] is the same as this paper, so the results are comparable.

In this section we summarize the best results obtained in this paper using FV, CNN and their combinations (top part of the Table VI) and we compare them with the results achieved in [Amato et al. 2014] using VLAD and BoW approaches (bottom part of Table VI).

In [Amato et al. 2014] the BoW approach achieved a maximum of 0.52 mAP using a visual vocabulary of $200,000$ words and performing geometric consistency check using RANSAC [Fischler and Bolles 1981]. Compared to BoW with RANSAC, the VLAD approach with 256 visual words (computed us-

Table VI. Summary of the best results obtained using FV, CNN and their combination, and comparison with the state-of-the-art results achieved using BoW and VLAD. Results related to FV and CNN are reported from Tables I, II, III, IV, and V. Results related to BoW and VLAD are reported from [Amato et al. 2014]. *dim* and *bytes* are respectively the number of components and the average size in bytes of each vector representation. The results are ordered with respect to the mAP quality measure. Bold numbers denote maxima in the respective column.

Method		**dim**	**bytes**	**mAP**	**p** $(r = 1)$	**p** $(r = 10)$	**p** $(r = 100)$
pool5, FV256	$\times$ $\diamond$	58,112	232,448	**0.75**	**0.93**	**0.97**	0.99
pool5, (FV256 $\rightarrow_{\text{PCA}}$ 4,096)	$\times$ $\diamond$	29,184	116,736	0.74	0.91	0.96	0.99
(pool5$\rightarrow_{\text{PCA}}$ 1,024), (FV256 $\rightarrow_{\text{PCA}}$ 4,096)	$\times$ $\diamond$	5,120	20,480	0.73	0.89	0.96	**1.00**
(pool5$\rightarrow_{\text{PCA}}$ 512), (FV256 $\rightarrow_{\text{PCA}}$ 4,096)	$\times$ $\diamond$	4,608	18,432	0.69	0.84	0.96	**1.00**
pool5$\rightarrow_{\text{PCA}}$ 2,048	$\diamond$	2,048	8,192	0.57	0.70	0.83	0.93
FV256	$\times$	33,024	132,096	0.55	0.73	0.76	0.87
FV256$\rightarrow_{\text{PCA}}$ 4,096	$\times$	4,096	16,384	0.55	0.70	0.77	0.86
pool5	$\diamond$	25,088	100,352	0.54	0.66	0.77	0.93
fc6 RELU$\rightarrow_{\text{PCA}}$ 1,024	$\diamond$	1,024	4,096	0.53	0.67	0.84	0.91
FV128	$\times$	16,512	66,048	0.52	0.69	0.76	0.84
VLAD256 *[Amato et al. 2014]*	$\star$	32,768	131,072	0.52	0.69	0.74	0.84
BoW 200k RANSAC *[Amato et al. 2014]*	$\bullet$	4,773	19,092	0.52	0.66	0.70	0.74
BoW 400k Cos-TFIDF *[Amato et al. 2014]*	$\star$	235	940	0.51	0.64	0.76	0.87

The combinations of FV and CNN were computed using $\alpha = 0.25$
$\bullet$ Descriptor computed by using SIFT
$\times$ Descriptor computed by using SIFTPCA64 with reduced-keypoints
$\star$ Descriptor computed by using SIFT with reduced-keypoints
$\diamond$ OxfordNet CNN

ing SIFT descriptors and reduced-keypoints) reached the same mAP and slight better probabilities $p(r)$. Quite worse results are obtained using BoW with $400,000$ words and using the cosine similarity measure in conjunction with tf-idf weighting scheme.

Interestingly, all the previous results of BoW and VLAD have been overcome by both FV and OxfordNet features, even if used individually (except for the full-size *fc6 RELU*). For example, FV256 (i.e. FV with $K = 256$, computed using SIFTPCA64 and reduced-keypoints) has quite the same dimensionality of VLAD256 but gets a 4% better mAP and slight better probabilities $p(r)$. Furthermore, OxfordNet *pool5*, reduced to a vector of dimension $2,048$, far outperforms either BoW and VLAD both in effectiveness and memory occupation.

The overall best results were obtained by combining FV256 with OxfordNet *pool5*, with a gain over BoW and VLAD of $+\mathbf{22}\%$ in mAP and $+\mathbf{24}\%$ in retrieving a relevant image as first result, i.e. $p(r = 1)$. The fusion of the FV256 and OxfordNet features is costly due to the extraction of both FV and CNN features from the query. This may be an issue when using devices with limited computational resources. To reduce the cost of the feature combination it is possible to use FV with $K = 32$ or 64 that are cheaper to compute. In facts, as proved in Section 4.4.3 (figure 6), the performance of OxfordNet features is alway improved by the combination with FV, even if less accurate (and less expensive) FV is used, such as that obtained using small K.

The mixture of FV256 with OxfordNet *pool5* is very high dimensional ($58,112$ components) so PCA-reduced version of FV and CNN features can be used to obtain more compact image representation while preserving effectiveness. However, we did not consider reduction of FV256 to less than $4,096$

Table VII. Performance of the combination of FV and OxfordNet *pool5* feature, varying the parameter α. $\alpha = 0$ corresponds to use only FV, while $\alpha = 1$ corresponds to use only the OxfordNet feature. The FV was computed using $K = 256$ Gaussian. *dim* and *bytes* are respectively the number of components and the average size in bytes of each vector representation. Bold numbers denote maxima in the respective column.

(a) Full-sized descriptors

Method	$\boldsymbol{\alpha}$	**dim**	**bytes**	**mAP**	**p** ($r = 1$)	**p** ($r = 10$)	**p** ($r = 100$)
FV256	0	33,024	132,096	0.56	0.85	0.97	**1.00**
pool5, FV256	0.25	58,112	232,448	**0.65**	**0.93**	**0.99**	**1.00**
pool5, FV256	0.50	58,112	232,448	0.61	0.91	0.98	**1.00**
pool5, FV256	0.75	58,112	232,448	0.58	0.89	0.98	**1.00**
pool5	1	25,088	100,352	0.56	0.88	0.97	**1.00**

(b) PCA-reduced descriptors

Method	$\boldsymbol{\alpha}$	**dim**	**bytes**	**mAP**	**p** ($r = 1$)	**p** ($r = 10$)	**p** ($r = 100$)
FV256$\rightarrow_{\text{PCA}}$ 4,096	0	4,096	16,384	0.51	0.82	0.97	**1.00**
(pool5$\rightarrow_{\text{PCA}}$ 1,024), (FV256$\rightarrow_{\text{PCA}}$ 4,096)	0.25	5,120	20,480	**0.64**	**0.92**	**0.99**	**1.00**
(pool5$\rightarrow_{\text{PCA}}$ 1,024), (FV256$\rightarrow_{\text{PCA}}$ 4,096)	0.50	5,120	20,480	0.61	0.90	0.98	**1.00**
(pool5$\rightarrow_{\text{PCA}}$ 1,024), (FV256$\rightarrow_{\text{PCA}}$ 4,096)	0.75	5,120	20,480	0.58	0.88	0.98	**1.00**
pool5$\rightarrow_{\text{PCA}}$ 1,024	1	1,024	4,096	0.56	0.87	0.98	**1.00**

components since in these cases the mAP decreases and the retrieval gain due to the features combination do not balance the extra cost of FV256 extraction with respect to the single use of PCA-reduced version of OxfordNet *pool5*.

In conclusion our results show that combinations of FV and CNN achieve very high effectiveness in recognizes ancient inscription and can be profitably used when efficiency is not the main concern. The memory occupation can be downsized using PCA, and the cost of FV-CNN combination can be reduced using cheaper FV (i.e., FV with small K). Finally, the single OxfordNet *pool5* reduced to $2,048$ dimensions can be used as trade-off between efficiency ad effectiveness. In Figure 7 we report some examples of top results obtained using different image representations. Please note that while our objective is to recognize a specific inscription (namely to have a correct answer in the first positions of the result list), the use of CNN-features (both pure and combined with FV) allow us to retrieve images that, even if are not a correct answer, represent objects very similar to the query one.

4.5 Experimental evaluation on Pisa Dataset

The results on the Epigraphic Database Roma reported in the previous sections show that the use of a combination of FV a CNN features leads to improve the retrieval performance with respect to use the FV or the CNN feature alone. In this section, we further analyse the retrieval performance of FV-CNN combination in a cultural heritage context that is different from the one of ancient inscriptions. To this scope, we used the publicly available *Pisa Dataset*[6] composed of $1,227$ photos of 12 monuments and

[6]http://www.nmis.isti.cnr.it/falchi/pisaDataset/

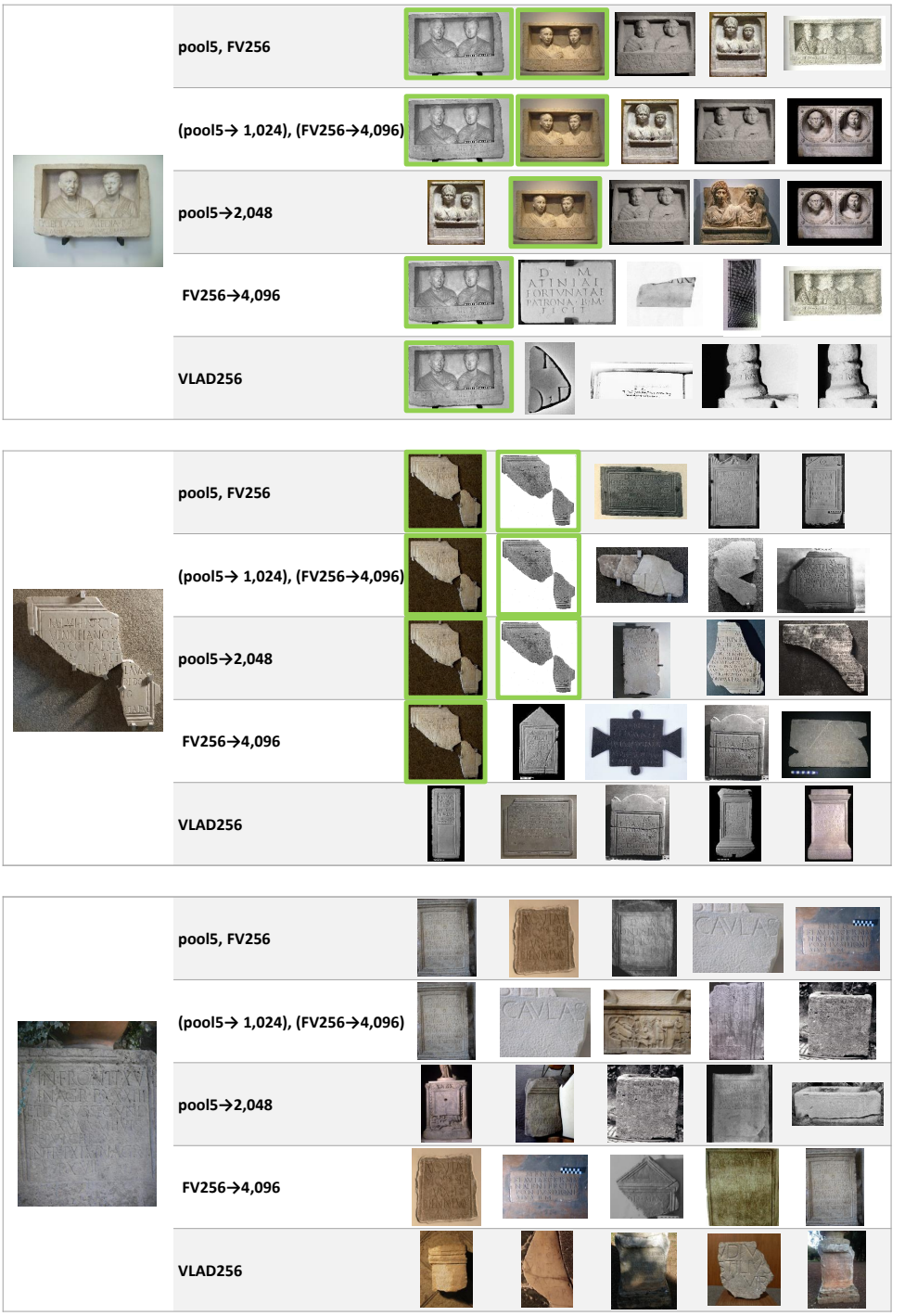


Fig. 7. Examples of top retrieved images for different image representation approaches. On the left, we show the query image; on the right, for each method, we report the top five results. Please note that the correct answers are outlined in green and that we removed the query image from the result list.

landmarks located in Pisa. This dataset was created during the VISITO Tuscany project[7] and has also been used in [Amato et al. 2015; Kamahara and Nagamatsu 2012; Barrios et al. 2014] for tasks such as classification and indexing. The dataset is divided in a training set consisting of 226 photos (20% of the dataset) and a test set consisting of 921 photos (80% of the dataset).

In Table VII (a) we report the results obtained using FV with $K = 256$, OxfordNet *pool5* and their combinations for various values of α. Please remember that α is the parameter used in the convex combination defined in Eq. (6). $\alpha = 0$ corresponds to use only the FV, while $\alpha = 1$ corresponds to use the CNN feature alone.The experimental setting used for the feature extraction is the same described in Section 4.2. We observed that also in this scenario the combination of FV and CNN allows to improve the retrieval performance with respect to use just the FV or the CNN feature. As happened in the experiments on the Epigraphic Database Rome, we observed that exists an optimal value for the parameter α where the combination achieved the maximum mAP. The optimal α values was obtained around 0.25. In facts, the use of either FV or CNN leads to have a mAP of 0.56, while the FV-CNN combination with $\alpha = 0.25$ reaches a maximum of 0.65 mAP. For this value of α we also correctly recognize the query as first results in the 93% of the cases.

In Table VII (b) we also evaluated the case in which the PCA is used to reduce the dimensionality of FV and CNN feature before the combination. We reduced the OxfordNet *pool5* from $25,088$ to $1,024$ components and the FV representation from $33,024$ to $4,096$. We obtain that the maximum mAP was achieved using $\alpha = 0.25$ as well. The retrieval results obtained using the PCA-reduced version of FV and CNN feature were in line with that obtained using the full-sized descriptors. Thus also in this case the use of a combination of the PCA-reduced FV and CNN descriptors can be considered as trade-off between efficiency and effectiveness.

5. CONCLUSIONS

This paper has investigated the problem of visually recognize ancient inscriptions (such as Roman and Greek epigraphs) by testing the most prominent visual instance retrieval approaches (VLAD, BoW, FV, CNN) also considering combination of FV and CNN features. The results of extensive experiments, conducted on $17,155$ images related to $14,560$ ancient inscriptions, revealed that very high effectiveness can be achieved by combining FV and CNN. In fact, in more than 90% of the cases we obtained an image of the same query inscription as first result. This allows recognizing the inscription in 90% of the cases when using the first result for the classification. Moreover, we achieved a mean average precision greater than 0.70, which means that the overall ordering of the results is good. Nevertheless, the combination of FV and CNN is costly due to the extraction and storage of both FV and CNN features. We showed that PCA dimensionality reduction can be effectively used to reduce the memory occupation of FV-CNN combination without loss in accuracy. In addition, the cost of features extraction can be reduced using smaller FV since also in this case we observed that the retrieval performance of CNN features was improved by the combination with FV. However, when the feature extraction is performed directly on devices with limited resources (such as smartphones or wearable devices) a single feature could be preferable. The single use of FV and CNN led to correctly recognize the query inscription in 73% and 70% of the cases respectively, reaching a mAP of 0.55 and 0.57. Our experiments also showed that FV and CNN features perform better than BoW and VLAD approaches, tested in our previous work [Amato et al. 2014] for recognizing ancient inscriptions.

This research was conducted in the context of the Europeana network of Ancient Greek and Latin Epigraphy (EAGLE) project. All our results were obtained using open source libraries (VIR, Caffe, OpenCV), thus other researchers can freely use the tested techniques to perform visual search on dif-

[7] http://www.visitotuscany.it/index.php/en

ferent epigraphic datasets and scenarios. In fact, the techniques described in this paper are general and can be clearly used to retrieve information on other type of objects related to cultural heritage (assuming that these objects can be described by their visual appearance). For example, in this paper we show that the combination of FV and CNN features are effective also for searching the Pisa Dataset, which contains photos of monuments and landmarks located in Pisa. Since in all our tests the combination of FV and CNN has led to improve the retrieval performances, as future work we intent to further investigate this kind of combinations.

REFERENCES

G. Amato, F. Falchi, and C. Gennaro. 2013. On Reducing the Number of Visual Words in the Bag-of-Features Representation. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISIGRAPP 2013)*. 657–662. DOI:http://dx.doi.org/10.5220/0004290506570662

G. Amato, F. Falchi, and C. Gennaro. 2015. Fast Image Classification for Monument Recognition. *J. Comput. Cult. Herit.* 8, 4, Article 18 (Aug. 2015), 25 pages. DOI:http://dx.doi.org/10.1145/2724727

G. Amato, F. Falchi, F. Rabitti, and L. Vadicamo. 2014. Inscriptions visual recognition. A comparison of state-of-the-art object recognition approaches. In *Proceedings of the First EAGLE International Conference*, Vol. 26. Sapienza Universitá Editrice, 117 –131. http://archiv.ub.uni-heidelberg.de/propylaeumdok/volltexte/2015/2337

R. Arandjelovic and A. Zisserman. 2013. All About VLAD. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. 1578–1585. DOI:http://dx.doi.org/10.1109/CVPR.2013.207

A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. 2014. Neural codes for image retrieval. In *Computer Vision–ECCV 2014*. Springer, 584–599. DOI:http://dx.doi.org/10.1007/978-3-319-10590-1_38

J. M. Barrios, B. Bustos, and T. Skopal. 2014. Analyzing and dynamically indexing the query set. *Information Systems* 45 (2014), 37–47. DOI:http://dx.doi.org/10.1016/j.is.2013.05.010

H. Bay, T. Tuytelaars, and L. Van Gool. 2006. SURF: Speeded Up Robust Features. In *Computer Vision - ECCV 2006*, Ales Leonardis, Horst Bischof, and Axel Pinz (Eds.). Lecture Notes in Computer Science, Vol. 3951. Springer Berlin Heidelberg, 404–417. DOI:http://dx.doi.org/10.1007/11744023_32

C. M. Bishop. 2006. *Pattern Recognition and Machine Learning*. Springer.

V. Chandrasekhar, J. Lin, O. Morère, H. Goh, and A. Veillard. 2015. A Practical Guide to CNNs and Fisher Vectors for Image Instance Retrieval. *CoRR* abs/1508.02496 (2015). http://arxiv.org/abs/1508.02496

D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, Huizhong Chen, R. Vedantham, R. Grzeszczuk, and B. Girod. 2011. Residual Enhanced Visual Vectors for on-device image matching. In *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on*. 850–854. DOI:http://dx.doi.org/10.1016/j.sigpro.2012.06.005

G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. 2004. Visual categorization with bags of keypoints. *Workshop on statistical learning in computer vision, ECCV* 1, 1-22 (2004), 1–2.

J. Delhumeau, P-H. Gosselin, H. Jégou, and P. Pérez. 2013. Revisiting the VLAD Image Representation. In *Proceedings of the 21st ACM International Conference on Multimedia (MM 2013)*. ACM, New York, NY, USA, 653–656. DOI:http://dx.doi.org/10.1145/2502081.2502171

J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 248–255. DOI:http://dx.doi.org/10.1109/CVPR.2009.5206848

J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. 2013. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. *CoRR* abs/1310.1531 (2013). http://arxiv.org/abs/1310.1531

M. A. Fischler and R. C. Bolles. 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* 24, 6 (June 1981), 381–395. DOI:http://dx.doi.org/10.1145/358669.358692

R. Girshick, J. Donahue, T. Darrell, and J. Malik. 2014. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. 580–587. DOI:http://dx.doi.org/10.1109/CVPR.2014.81

I. Goodfellow, Y. Bengio, and A. Courville. 2016. Deep Learning. (2016). http://www.deeplearningbook.org Book in preparation for MIT Press.

R. M. Gray and D. L. Neuhoff. 1998. Quantization. *Information Theory, IEEE Transactions on* 44, 6 (Oct 1998), 2325–2383. DOI:http://dx.doi.org/10.1109/18.720541

T. Jaakkola and D. Haussler. 1998. Exploiting Generative Models in Discriminative Classifiers. In *In Advances in Neural Information Processing Systems 11*. MIT Press, 487–493. http://dl.acm.org/citation.cfm?id=340534.340715

H. Jégou, M. Douze, and C. Schmid. 2010. Improving Bag-of-Features for Large Scale Image Search. *International Journal of Computer Vision* 87, 3 (2010), 316–336. DOI:http://dx.doi.org/10.1007/s11263-009-0285-2

H. Jégou, M. Douze, and C. Schmid. 2011. Product Quantization for Nearest Neighbor Search. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33, 1 (Jan 2011), 117–128. DOI:http://dx.doi.org/10.1109/TPAMI.2010.57

H. Jégou, M. Douze, C. Schmid, and P. Pérez. 2010. Aggregating local descriptors into a compact image representation. In *IEEE Conference on Computer Vision & Pattern Recognition*. DOI:http://dx.doi.org/10.1109/CVPR.2010.5540039

H. Jégou, F. Perronnin, M. Douze, J. Sànchez, P. Pérez, and C. Schmid. 2012. Aggregating Local Image Descriptors into Compact Codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 9 (2012), 1704–1716. DOI:http://dx.doi.org/10.1109/TPAMI.2011.235

Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*. ACM, 675–678. DOI:http://dx.doi.org/10.1145/2647868.2654889

J. Kamahara and N. Nagamatsu, T.and Tanaka. 2012. Conjunctive Ranking Function Using Geographic Distance and Image Distance for Geotagged Image Retrieval. In *Proceedings of the ACM Multimedia 2012 Workshop on Geotagging and Its Applications in Multimedia (GeoMM '12)*. ACM, New York, NY, USA, 9–14. DOI:http://dx.doi.org/10.1145/2390790.2390795

A. Krizhevsky, I. Sutskever, and G. E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger (Eds.). Curran Associates, Inc., 1097–1105.

S. Lloyd. 1982. Least squares quantization in PCM. *Information Theory, IEEE Transactions on* 28, 2 (Mar 1982), 129–137. DOI:http://dx.doi.org/10.1109/TIT.1982.1056489

D. G. Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60, 2 (2004), 91–110. DOI:http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94

G. McLachlan and D. Peel. 2000. *Finite Mixture Models*. Wiley.

F. Perronnin and C. Dance. 2007. Fisher Kernels on Visual Vocabularies for Image Categorization. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*. 1–8. DOI:http://dx.doi.org/10.1109/CVPR.2007.383266

F. Perronnin, Yan Liu, J. Sànchez, and H. Poirier. 2010a. Large-scale image retrieval with compressed Fisher vectors. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. 3384–3391. DOI:http://dx.doi.org/10.1109/CVPR.2010.5540009

F. Perronnin, J. Sànchez, and T. Mensink. 2010b. Improving the Fisher Kernel for Large-Scale Image Classification. In *Computer Vision - ECCV 2010*. Lecture Notes in Computer Science, Vol. 6314. Springer Berlin Heidelberg, 143–156. DOI:http://dx.doi.org/10.1007/978-3-642-15561-1_11

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. 2007. Object Retrieval with Large Vocabularies and Fast Spatial Matching. In *Computer Vision and Pattern Recognition (CVPR), 2007 IEEE Conference on*. 1–8. DOI:http://dx.doi.org/10.1109/CVPR.2007.383172

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. 2008. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. 1–8. DOI:http://dx.doi.org/10.1109/CVPR.2008.4587635

G. A. Pratt. 2015. Is a Cambrian explosion coming for robotics? *Journal of Economic Perspectives* 29, 3 (August 2015), 51–60. DOI:http://dx.doi.org/10.1257/jep.29.3.51

A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. 2014. CNN features off-the-shelf: an astounding baseline for recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*. IEEE, 512–519. DOI:http://dx.doi.org/10.1109/CVPRW.2014.131

G. Salton and M. J. McGill. 1986. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA.

J. Sànchez, F. Perronnin, T. Mensink, and J. Verbeek. 2013. Image Classification with the Fisher Vector: Theory and Practice. *International Journal of Computer Vision* 105, 3 (2013), 222–245. DOI:http://dx.doi.org/10.1007/s11263-013-0636-x

K. Simonyan and A. Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1409.1556 (2014). http://arxiv.org/abs/1409.1556

J. Sivic and A. Zisserman. 2003. Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV '03)*, Vol. 2. IEEE Computer Society, 1470–1477. DOI:http://dx.doi.org/10.1109/ICCV.2003.1238663

B. Thomee, E. M. Bakker, and M. S. Lew. 2010. TOP-SURF: A Visual Words Toolkit. In *Proceedings of the International Conference on Multimedia (MM '10)*. ACM, 1473–1476. DOI:http://dx.doi.org/10.1145/1873951.1874250

G. Tolias and H. Jégou. 2013. *Local visual query expansion: Exploiting an image collection to refine local descriptors*. Research Report RR-8325. https://hal.inria.fr/hal-00840721

G. Tolias, R. Sicre, and H. Jégou. 2015. Particular object retrieval with integral max-pooling of CNN activations. *arXiv preprint arXiv:1511.05879* (2015). http://arxiv.org/abs/1511.05879

J.C. Van Gemert, C.J. Veenman, A.W.M. Smeulders, and J.-M. Geusebroek. 2010. Visual Word Ambiguity. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32, 7 (July 2010), 1271–1283. DOI:http://dx.doi.org/10.1109/TPAMI.2009.132

W.L. Zhao, H. Jégou, and G. Gravier. 2013. Oriented pooling for dense and non-dense rotation-invariant features. In *BMVC - 24th British Machine Vision Conference*.

Y. T. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T.S. Chua, and H. Neven. 2009. Tour the world: Building a web-scale landmark recognition engine. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 1085–1092. DOI:http://dx.doi.org/10.1109/CVPR.2009.5206749

B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. 2014. Learning Deep Features for Scene Recognition using Places Database. In *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger (Eds.). Curran Associates, Inc., 487–495.

Received m YYYY; revised m YYYY; accepted m YYYY