

Large-Scale Image Retrieval with Attentive Deep Local Features

Hyeonwoo Noh[†]

André Araujo*

Jack Sim*

Tobias Weyand*

Bohyung Han[†][†]POSTECH, Korea

{shgusdngogo, bhan}@postech.ac.kr

*Google Inc.

{andrearaujo, jacksim, weyand}@google.com

Abstract

We propose an **attentive local feature descriptor** suitable for **large-scale image retrieval**, referred to as **DELF (DEep Local Feature)**. The new feature is based on convolutional neural networks, which are trained only with **image-level annotations** on a **landmark image dataset**. To **identify semantically useful local features for image retrieval**, we also propose an **attention mechanism** for keypoint selection, which shares most network layers with the descriptor. This framework can be used for image retrieval as a drop-in replacement for other keypoint detectors and descriptors, enabling more **accurate feature matching and geometric verification**. Our system produces reliable confidence scores to reject false positives—in particular, it is robust against queries that have no correct match in the database. To evaluate the proposed descriptor, we introduce a new **large-scale dataset**, referred to as **Google-Landmarks dataset**, which involves challenges in both database and query such as **background clutter, partial occlusion, multiple landmarks, objects in variable scales, etc.** We show that **DELF outperforms the state-of-the-art global and local descriptors in the large-scale setting by significant margins**. Code and dataset can be found at the project webpage: <https://github.com/tensorflow/models/tree/master/research/delf>.

1. Introduction

Large-scale image retrieval is a fundamental task in computer vision, since it is directly related to various practical applications, *e.g.*, object detection, visual place recognition, and product recognition. The last decades have witnessed tremendous advances in image retrieval systems—from hand-crafted features and indexing algorithms [22, 33, 27, 16] to, more recently, methods based on convolutional neural networks (CNNs) for global descriptor learning [2, 29, 11].

Despite the recent advances in CNN-based global descriptors for image retrieval in small or medium-size datasets [27, 28], their performance may be hindered by a wide variety

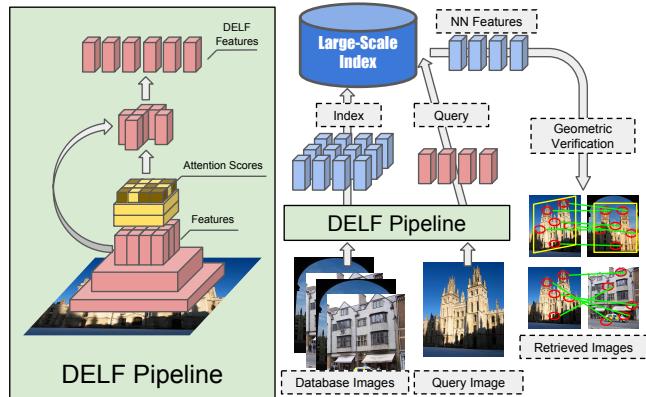


Figure 1: Overall architecture of our image retrieval system, using DEep Local Features (DELF) and attention-based keypoint selection. On the left, we illustrate the pipeline for extraction and selection of DELF. The portion highlighted in yellow represents an attention mechanism that is trained to assign high scores to relevant features and select the features with the highest scores. Feature extraction and selection can be performed with a single forward pass using our model. On the right, we illustrate our large-scale feature-based retrieval pipeline. DELF for database images are indexed offline. The index supports querying by retrieving nearest neighbor (NN) features, which can be used to rank database images based on geometrically verified matches.

of challenging conditions observed in large-scale datasets, such as **clutter, occlusion, and variations in viewpoint and illumination**. Global descriptors lack the ability to find **patch-level matches** between images. As a result, it is difficult to retrieve images based on **partial matching** in the presence of occlusion and background clutter. In a recent trend, CNN-based local features are proposed for **patch-level matching** [12, 42, 40]. However, these techniques are not optimized specifically for image retrieval since they lack the ability to detect semantically meaningful features, and show limited accuracy in practice. *semantic: relating to meaning in language or logic.*

Most existing image retrieval algorithms have been evaluated in small to medium-size datasets with few query images, *i.e.*, only 55 in [27, 28] and 500 in [16], and the images in the datasets have limited diversity in terms of landmark locations and types. Therefore, we believe that the image

retrieval community can benefit from a large-scale dataset, comprising more comprehensive and challenging examples, to improve algorithm performance and evaluation methodology by deriving more statistically meaningful results.

The main goal of this work is to **develop a large-scale image retrieval system based on a novel CNN-based feature descriptor**. To this end, we first introduce a new large-scale dataset, Google-Landmarks, which contains more than **1M landmark images** from almost **13K unique landmarks**. This dataset covers a wide area in the world, and is consequently more diverse and comprehensive than existing ones. The query set is composed of an extra **100K images** with diverse characteristics; in particular, we include images that have no match in the database, which makes our dataset more challenging. This allows to assess the robustness of retrieval systems when queries do not necessarily depict landmarks.

We then propose a **CNN-based local feature** with attention, which is trained with weak supervision using **image-level class labels only**, without the need of object- and patch-level annotations. This new feature descriptor is referred to as **DELF (DEep Local Feature)**, and Fig. 1 illustrates the overall procedure of feature extraction and image retrieval. In our approach, **the attention model is tightly coupled with the proposed descriptor; it reuses the same CNN architecture and generates feature scores using very little extra computation** (in the spirit of recent advances in object detection [30]). This enables the extraction of both local descriptors and keypoints via one forward pass over the network. We show that our image retrieval system based on DELF achieves the state-of-the-art performance with significant margins compared to methods based on existing global and local descriptors.

2. Related Work

There are standard datasets commonly used for the evaluation of image retrieval techniques. Oxford5k [27] has 5,062 building images captured in Oxford with 55 query images. Paris6k [28] is composed of 6,412 images of landmarks in Paris, and also has 55 query images. These two datasets are often augmented with 100K distractor images from Flickr100k dataset [27], which constructs Oxford105k and Paris106k datasets, respectively. On the other hand, Holidays dataset [16] provides 1,491 images including 500 query images, which are from personal holiday photos. All these three datasets are fairly small, especially having a very small number of query images, which makes it difficult to generalize the performance tested in these datasets. Although Pitts250k [35] is larger, it is specialized to visual places with repetitive patterns and may not be appropriate for the general image retrieval task.

Instance retrieval has been a popular research problem for more than a decade. See [43] for a recent survey. Early systems rely on **hand-crafted local features** [22, 5, 8], coupled with **approximate nearest neighbor search methods using KD**

trees or vocabulary trees [6, 25]. Still today, such **feature-based techniques combined with geometric re-ranking provide strong performance when retrieval systems need to operate with high precision**.

More recently, many works have focused on aggregation methods of local features, which include popular techniques such as **VLAD** [18] and **Fisher Vector (FV)** [19]. The main advantage of such global descriptors is the ability to provide **high-performance image retrieval with a compact index**.

In the past few years, several global descriptors based on CNNs have been proposed to use pretrained [4, 34] or learned networks [2, 29, 11]. These global descriptors are most commonly **trained with a triplet loss**, in order to preserve the ranking between relevant and irrelevant images. Some retrieval algorithms using these CNN-based global descriptors make use of **deep local features as a drop-in replacement for hand-crafted features in conventional aggregation techniques such as VLAD or FV** [24, 36]. Other works have **re-evaluated and proposed different feature aggregation methods using such deep local features** [3, 21].

CNNs have also been used to **detect, represent and compare local image features**. Verdie *et al.* [37] learned a regressor for repeatable keypoint detection. Yi *et al.* [41] proposed a generic CNN-based technique to estimate the canonical orientation of a local feature and successfully deployed it to several different descriptors. MatchNet [12] and DeepCompare [42] have been proposed to jointly learn patch representations and associated metrics. Recently, LIFT [40] proposed an end-to-end framework to detect keypoints, estimate orientation, and compute descriptors. Different from our work, these techniques are not designed for image retrieval applications since they do not learn to select semantically meaningful features.

Many visual recognition problems employ **visual attention based on deep neural networks**, which include object detection [45], semantic segmentation [14], image captioning [38], visual question answering [39], etc. However, visual attention has not been explored actively to learn visual features for image retrieval applications.

3. Google-Landmarks Dataset

Our dataset is constructed based on the algorithm described in [44]. Compared to the existing datasets for image retrieval [27, 28, 16], the new dataset is much **larger, contains diverse landmarks, and involves substantial challenges**. It contains 1,060,709 images from 12,894 landmarks, and 111,036 additional query images. The images in the dataset are captured at various locations in the world, and each image is associated with a **GPS coordinate**. Example images and their geographic distribution are presented in Fig. 2 and Fig. 3, respectively. While most images in the existing datasets are **landmark-centric**, which makes **global feature descriptors work well**, our dataset contains more **realistic im-**



(a) Sample database images



(b) Sample query images

Figure 2: Example database and query images from Google-Landmarks. They have a lot of variations and challenges including background clutter, small objects, and multiple landmarks.



Figure 3: Image geolocation distribution of our Google-Landmarks dataset. The landmarks are located in 4,872 cities in 187 countries.

ages with wild variations including foreground/background clutter, occlusion, partially out-of-view objects, etc. In particular, since our query images are collected from personal photo repositories, some of them may not contain any landmarks and should not retrieve any image from the database. We call these query images *distractors*, which play a critical role to evaluate robustness of algorithms to irrelevant and noisy queries.

We use visual features and GPS coordinates for *ground-truth construction*. All images in the database are clustered using the two kinds of information, and we assign a landmark identifier to each cluster. If physical distance between

the location of a query image and the center of the cluster associated with the retrieved image is less than a threshold, we assume that the two images belong to the same landmark. Note that ground-truth annotation is extremely challenging, especially considering the facts that it is hard to predefine what landmarks are, landmarks are not clearly noticeable sometimes, and there might be multiple instances in a single image. Obviously, this approach for ground-truth construction is *noisy due to GPS errors*. Also, photos can be captured from a large distance for some landmarks (e.g., Eiffel Tower, Golden Gate Bridge), and consequently the photo location might be relatively far from the actual landmark location. However, we found very few incorrect annotations with the threshold of 25km when checking a subset of data manually. Even though there are few minor errors, it is not problematic, especially in relative evaluation, because algorithms are unlikely to be confused between landmarks anyway if their visual appearances are sufficiently discriminative.

4. Image Retrieval with DELF

Our large-scale retrieval system can be decomposed into four main blocks: (i) *dense localized feature extraction*, (ii) *keypoint selection*, (iii) *dimensionality reduction* and (iv) *indexing and retrieval*. This section describes DELF feature extraction and learning algorithm followed by our indexing and retrieval procedure in detail.

4.1. Dense Localized Feature Extraction

We extract dense features from an image by *applying a fully convolutional network* (FCN), which is constructed by using the feature extraction layers of a CNN trained with a classification loss. We employ an *FCN taken from the ResNet50 [13] model, using the output of the conv4_x convolutional block*. To handle scale changes, we explicitly *construct an image pyramid and apply the FCN for each level independently*. The obtained feature maps are regarded as a dense grid of local descriptors. *Features are localized based on their receptive fields, which can be computed by considering the configuration of convolutional and pooling layers of the FCN*. We use the pixel coordinates of the *center of the receptive field as the feature location*. The receptive field size for the image at the original scale is 291×291 . Using the image pyramid, we obtain features that describe image regions of different sizes.

We use the original ResNet50 model trained on ImageNet [31] as a baseline, and fine-tune for enhancing the discriminativeness of our local descriptors. Since we consider a landmark recognition application, we *employ annotated datasets of landmark images [4] and train the network with a standard cross-entropy loss for image classification* as illustrated in Fig. 4(a). The input images are initially *center-cropped to produce square images and rescaled to 250×250 . Random 224×224 crops are then used for training*. As a

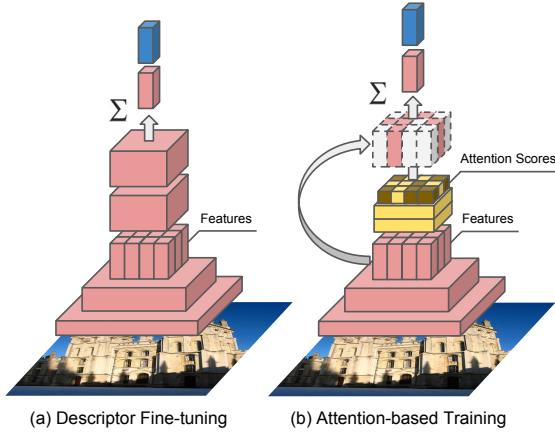


Figure 4: The network architectures used for training.

result of training, local descriptors implicitly learn representations that are more relevant for the landmark retrieval problem. In this manner, neither object- nor patch-level labels are necessary to obtain improved local descriptors.

4.2. Attention-based Keypoint Selection

Instead of using densely extracted features directly for image retrieval, we design a technique to effectively select a subset of the features. Since a substantial part of the densely extracted features are irrelevant to our recognition task and likely to add clutter, distracting the retrieval process, keypoint selection is important for both accuracy and computational efficiency of retrieval systems.

4.2.1 Learning with Weak Supervision

We propose to train a landmark classifier with attention to explicitly measure relevance scores for local feature descriptors. To train the function, features are pooled by a weighted sum, where the weights are predicted by the attention network. The training procedure is similar to the one described in Sec. 4.1 including the loss function and datasets, and is illustrated in Fig. 4(b), where the attention network is highlighted in yellow. This generates an embedding for the whole input image, which is then used to train a softmax-based landmark classifier.

More precisely, we formulate the training as follows. Denote by $\mathbf{f}_n \in \mathbb{R}^d, n = 1, \dots, N$ the d -dimensional features to be learned jointly with the attention model. Our goal is to learn a score function $\alpha(\mathbf{f}_n; \theta)$ for each feature, where θ denotes the parameters of function $\alpha(\cdot)$. The output logit \mathbf{y} of the network is generated by a weighted sum of the feature vectors, which is given by

$$\mathbf{y} = \mathbf{W} \left(\sum_n \alpha(\mathbf{f}_n; \theta) \cdot \mathbf{f}_n \right), \quad (1)$$

where $\mathbf{W} \in \mathbb{R}^{M \times d}$ represents the weights of the final fully-connected layer of the CNN trained to predict M classes.

For training, we use cross entropy loss, which is given by

$$\mathcal{L} = -\mathbf{y}^* \cdot \log \left(\frac{\exp(\mathbf{y})}{\mathbf{1}^T \exp(\mathbf{y})} \right), \quad (2)$$

where \mathbf{y}^* is ground-truth in one-hot representation and $\mathbf{1}$ is one vector. The parameters in the score function $\alpha(\cdot)$ are trained by backpropagation, where the gradient is given by

$$\frac{\partial \mathcal{L}}{\partial \theta} = \frac{\partial \mathcal{L}}{\partial \mathbf{y}} \sum_n \frac{\partial \mathbf{y}}{\partial \alpha_n} \frac{\partial \alpha_n}{\partial \theta} = \frac{\partial \mathcal{L}}{\partial \mathbf{y}} \sum_n \mathbf{W} \mathbf{f}_n \frac{\partial \alpha_n}{\partial \theta}, \quad (3)$$

where the backpropagation of the output score $\alpha_n \equiv \alpha(\mathbf{f}_n; \theta)$ with respect to θ is same as the standard multi-layer perceptron.

We restrict $\alpha(\cdot)$ to be non-negative, to prevent it from learning negative weighting. The score function is designed using a 2-layer CNN with a softplus [9] activation at the top. For simplicity, we employ the convolutional filters of size 1×1 , which work well in practice. Once the attention model is trained, it can be used to assess the relevance of features extracted by our model.

4.2.2 Training Attention

In the proposed framework, both the descriptors and the attention model are implicitly learned with image-level labels. Unfortunately, this poses some challenges to the learning process. While the feature representation and the score function can be trained jointly by backpropagation, we found that this setup generates weak models in practice. Therefore, we employ a two-step training strategy. First, we learn descriptors with fine-tuning as described in Sec. 4.1, and then the score function is learned given the fixed descriptors.

Another improvement to our models is obtained by random image rescaling during attention training process. This is intuitive, as the attention model should be able to generate effective scores for features at different scales. In this case, the input images are initially center-cropped to produce square images, and rescaled to 900×900 . Random 720×720 crops are then extracted and finally randomly scaled with a factor $\gamma \leq 1$.

4.2.3 Characteristics

One unconventional aspect of our system is that keypoint selection comes after descriptor extraction, which is different from the existing techniques (e.g., SIFT [22] and LIFT [40]), where keypoints are first detected and later described. Traditional keypoint detectors focus on detecting keypoints repeatedly under different imaging conditions, based only on their low-level characteristics. However, for a high-level recognition task such as image retrieval, it is also critical to select

keypoints that **discriminate different object instances**. The proposed pipeline achieves both goals by training a model that encodes higher level semantics in the feature map, and learning to select discriminative features for the classification task. This is in contrast to recently proposed techniques for learning keypoint detectors, *i.e.*, LIFT [40], which collect training data based on SIFT matches. **Although our model is not constrained to learn invariances to pose and viewpoint, it implicitly learns to do so—similar to CNN-based image classification techniques.**

4.3. Dimensionality Reduction

We reduce the dimensionality of selected features to obtain improved retrieval accuracy, as common practice [15]. First, the selected features are L_2 normalized, and their **dimensionality** is reduced to 40 by PCA, which presents a good trade-off between compactness and discriminativeness. Finally, the features **once again undergo L_2 normalization**.

4.4. Image Retrieval System

We extract feature descriptors from query and database images, where **a predefined number of local features with the highest attention scores per image are selected**. Our image retrieval system is based on **nearest neighbor search**, which is implemented by a combination of **KD-tree** [7] and **Product Quantization (PQ)** [17]. We encode each descriptor to a 50-bit code using PQ, where **each 40D feature descriptor is split into 10 subvectors with equal dimensions**, and we identify 2^5 centroids per subvector by k -means clustering to achieve 50-bit encoding. We perform **asymmetric distance calculation**, where the **query descriptors are not encoded to improve the accuracy of nearest neighbor retrieval**. To speed up the nearest neighbor search, we construct an inverted index for descriptors, using a codebook of size 8K. To reduce encoding errors, a **KD-tree is used to partition each Voronoi cell**, and **a Locally Optimized Product Quantizer** [20] is employed for each subtree with fewer than 30K features.

When a query is given, we perform approximate nearest neighbor search for each local descriptor extracted from a query image. Then for the top K nearest local descriptors retrieved from the index, we aggregate all the matches per database image. Finally, we perform geometric verification using **RANSAC** [10] and **employ the number of inliers as the score for retrieved images**. Many distractor queries are rejected by this geometric verification step because features from distractors may not be consistently matched with the ones from landmark images.

This pipeline requires less than 8GB memory to index 1 billion descriptors, which is sufficient to handle our large-scale landmark dataset. The latency of the nearest neighbor search is less than 2 seconds using a single CPU under our experiment setup, where we soft-assign 5 centroids to each query and search up to 10K leaf nodes within each inverted

index tree.

5. Experiments

This section mainly discusses the performance of DELF compared to existing global and local feature descriptors in our dataset. In addition, we also show how **Delf can be employed to achieve good accuracy in the existing datasets**.

5.1. Implementation Details

Multi-scale descriptor extraction We construct image pyramids by using scales that are a $\sqrt{2}$ factor apart. For the set of scales with range from 0.25 to 2.0, **7 different scales are used**. The size of receptive field is inversely proportional to the scale; for example, for the 2.0 scale, the receptive field of the network covers 146×146 pixels.

Training We employed landmarks dataset [4] for fine-tuning descriptors and training keypoint selection. In the dataset, there are the “full” version, referred to as **LF** (after removal of overlapping classes with Oxf5k/Par6k, by [11]), containing **140,372 images from 586 landmarks**, and the “clean” version (**LC**) obtained by a **SIFT-based matching procedure** [11], with **35,382 images of 586 landmarks**. We use **LF** to train our attention model, and **LC** is employed to fine-tune the network for image retrieval.

Parameters We identify the **top K ($= 60$) nearest neighbors for each feature in a query** and **extract up to 1000 local features from each image**—each feature is **40-dimensional**.

5.2. Compared Algorithms

Delf is compared with several recent global and local descriptors. Although there are various research outcomes related to image retrieval, we believe that the following methods are either relevant to our algorithm or most critical to evaluation due to their good performance.

Deep Image Retrieval (DIR) [11] This is a recent **global descriptor** that achieves the state-of-the-art performance in several existing datasets. DIR feature descriptors are 2,048 dimensional and multi-resolution descriptors are used in all cases. We also evaluate with **query expansion (QE)**, which typically improves accuracy in the standard datasets. We use the released source code that implements the version with **ResNet101** [13]. For retrieval, a **parallelized implementation of brute-force search is employed to avoid penalization by the error from approximate nearest neighbor search**.

siaMAC [29] This is a recent **global descriptor** that obtains high performance in existing datasets. We use the released source code with **parallelized implementation of brute-force search**. The CNN based on VGG16 [32] extracts 512 dimensional **global descriptor**. We also experiment with **query expansion (QE)** as in DIR.

CONGAS [8, 23] CONGAS is a 40D hand-engineered local feature, which has been widely used for instance-level image matching and retrieval [1, 44]. This feature descriptor is extracted by collecting Gabor wavelet responses at the detected keypoint scale and orientation, and is known to have very similar performance and characteristic to other gradient-based local descriptors like SIFT. A Laplacian-of-Gaussian keypoint detector is used.

LIFT LIFT [40] is a recently proposed feature matching pipeline, where keypoint detection, orientation estimation and keypoint description are jointly learned. Features are 128 dimensional. We use the source code publicly available.

5.3. Evaluation

Image retrieval systems have typically been evaluated based on mean average precision (mAP), which is computed by sorting images in descending order of relevance per query and averaging AP of individual queries. However, for datasets with distractor queries, such evaluation method is not representative since it is important to determine whether each image is relevant to the query or not. In our case, the absolute retrieval score is used to estimate the relevance of each image. For performance evaluation, we employ a modified version of precision (PRE) and recall (REC) by considering all query images at the same time, which are given by

$$\text{PRE} = \frac{\sum_q |\mathcal{R}_q^{\text{TP}}|}{\sum_q |\mathcal{R}_q|} \quad \text{and} \quad \text{REC} = \sum_q |\mathcal{R}_q^{\text{TP}}|, \quad (4)$$

where \mathcal{R}_q denotes a set of retrieved images for query q given a threshold, and $\mathcal{R}_q^{\text{TP}} (\subseteq \mathcal{R}_q)$ is a set of true positives. This is similar to the micro-AP metric introduced in [26]. Note that in our case only the top-scoring image per landmark is considered in the final scoring. We prefer unnormalized recall values, which present the number of retrieved true positives. Instead of summarizing our result in a single number, we present a full precision-recall curve to inspect operating points with different retrieval thresholds.

5.4. Quantitative Results

Fig. 5 presents the precision-recall curve of DELF (denoted by DELF+FT+ATT), compared to other methods. The results of LIFT could not be shown because feature extraction is extremely slow and large-scale experiment is infeasible¹. DELF clearly outperforms all other techniques significantly. Global feature descriptors, such as DIR, suffer in our challenging dataset. In particular, due to a large number of distractors in the query set, DIR with QE degrades accuracy significantly. CONGAS does a reasonably good job, but is still worse than DELF with substantial margin.

To analyze the benefit of fine-tuning and attention for image retrieval, we compare our full model (DELF+FT+ATT)

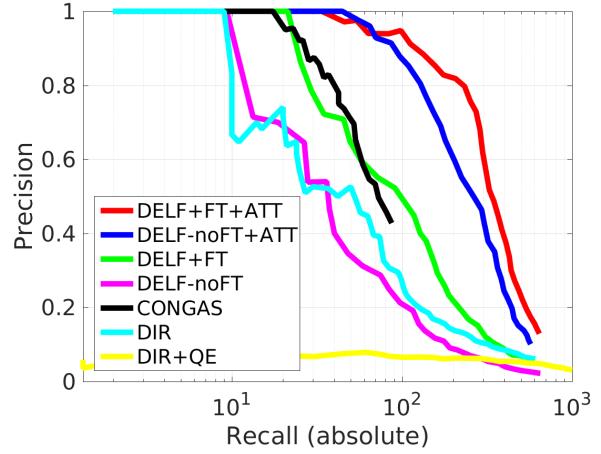


Figure 5: Precision-recall curve for the large-scale retrieval experiment on the Google-Landmarks dataset, where recall is presented in absolute terms, as in Eq. (4). DELF shows outstanding performance compared with existing global and local features. Fine-tuning and attention model in DELF are critical to performance improvement. The accuracy of DIR drops significantly with query expansion, due to many distractor queries in our dataset.

with its variations: DELF-noFT, DELF+FT and DELF-noFT+ATT. DELF-noFT means that extracted features are based on the pretrained CNN on ImageNet without fine-tuning and attention learning. DELF+FT denotes the model with fine-tuning but without attention modeling while DELF-noFT+ATT corresponds to the model without fine-tuning but using attention. As illustrated in Fig. 5, both fine-tuning and attention modeling make substantial contributions to performance improvement. In particular, note that the use of attention is more important than fine-tuning. This demonstrates that the proposed attention layers effectively learn to select the most discriminative features for the retrieval task, even if the features are simply pretrained on ImageNet.

In terms of memory requirement, DELF, CONGAS and DIR are almost equally complex. DELF and CONGAS adopt the same feature dimensionality and maximum number of features per image; they require approximately 8GB of memory. DIR descriptors need 8KB per image, summing up to approximately 8GB to index the entire dataset.

5.5. Qualitative Results

We present qualitative results to illustrate performance of DELF compared to two competitive algorithms based on global and local features—DIR and CONGAS, respectively. Also, we analyze our attention-based keypoint detection algorithm by visualization.

DELF vs. DIR Fig. 6 shows retrieval results, where DELF outperforms DIR. DELF obtains matches between specific local regions in images, which helps significantly to find the

¹LIFT feature extraction approximately takes 2 min/image using a GPU.

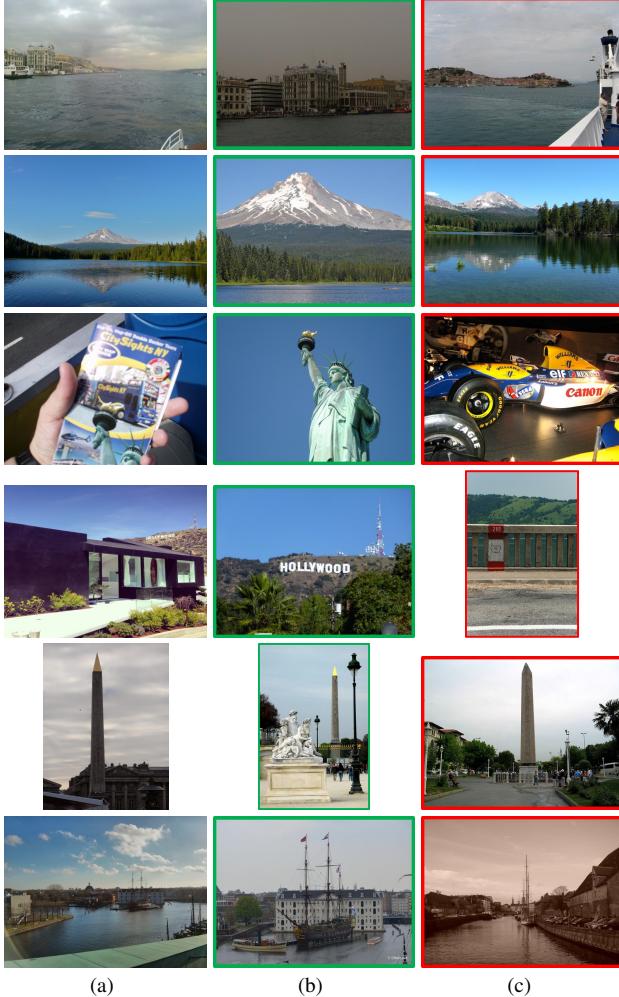


Figure 6: Examples where DELF+FT+ATT outperforms DIR: (a) query image, (b) top-1 image of DELF+FT+ATT, (c) top-1 image of DIR. The green borders denote correct results while the red ones mean incorrect retrievals. Note that DELF deals with clutter in query and database images and small landmarks effectively.

same object in different imaging conditions. Common failure cases of DIR happen when the database contains similar objects or scenes, e.g., obelisks, mountains, harbors, as illustrated in Fig. 6. In many cases, DIR cannot distinguish these specific objects or scenes; although it finds semantically similar images, they often do not correspond to the instance of interest. Another weakness of DIR and other global descriptors is that they are not good at identifying small objects of interest. Fig. 7 shows the cases that DIR outperforms DELF. While DELF is able to match localized patterns across different images, this leads to errors when the floor tiling or vegetation is similar across different landmarks.

DELF vs. CONGAS The main advantage of DELF over CONGAS is its recall; it retrieves more relevant landmarks than CONGAS, which suggests that DELF descriptors are

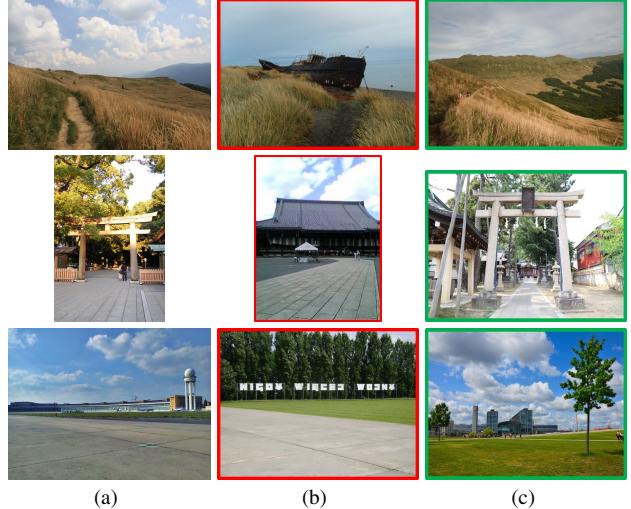


Figure 7: Examples where DIR outperforms DELF+FT+ATT: (a) query image, (b) top-1 image of DELF+FT+ATT, (c) top-1 image of DIR. The green and red borders denotes correct and incorrect results, respectively.

more discriminative. We did not observe significant examples where CONGAS outperforms DELF. Fig. 8 shows pairs of images from query and database, which are successfully matched by DELF but missed by CONGAS, where feature correspondences are presented by connecting the center of the receptive fields for matching features. Since the receptive fields can be fairly large, some features seem to be localized in undiscriminative regions, e.g., ocean or sky. However, in these cases, the features take into account more discriminative regions in the neighborhood.

Analysis of keypoint detection methods Fig. 9 visualizes three variations of keypoint detection, where the benefit of our attention model is clearly illustrated qualitatively while the L_2 norm of fine-tuned features is marginally different from the one without fine-tuning.

5.6. Results in Existing Datasets

We demonstrate the performance of DELF in existing datasets such as Oxf5k, Par6k and their extensions, Oxf105k and Par106k, for completeness. For this experiment, we simply obtain the score per image using the proposed method, and make a late fusion with the score from DIR by computing a weighted mean of two normalized scores, where the weight for DELF is set to 0.25. The results are presented in Tab. 1. We present accuracy of existing methods in their original papers and our reproductions using public source codes, which are very close. DELF improves accuracy nontrivially in the datasets when combined with DIR, although it does not show the best performance by itself. This fact indicates that DELF has capability to encode complementary information that is not available in global feature descriptors.



Figure 8: Visualization of feature correspondences between images in query and database using DELF+FT+ATT. For each pair, query and database images are presented side-by-side. DELF successfully matches landmarks and objects in challenging environment including partial occlusion, distracting objects, and background clutter. Both ends of the red lines denote the centers of matching features. Since the receptive fields are fairly large, the centers may be located outside landmark object areas. For the same queries, CONGAS fails to retrieve any image.

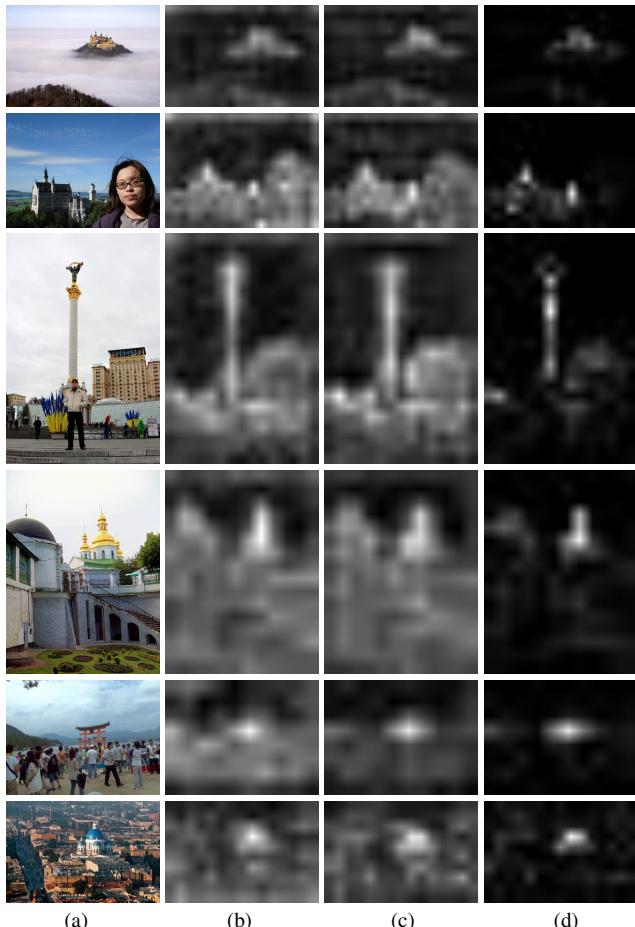


Figure 9: Comparison of keypoint selection methods. (a) Input image (b) L_2 norm scores using the pretrained model (DELF-noFT) (c) L_2 norm scores using fine-tuned descriptors (DELF+FT) (d) Attention-based scores (DELF+FT+ATT). Our attention-based model effectively disregards clutter compared to other options.

Table 1: Performance evaluation on existing datasets in mAP (%). All results of existing methods are based on our reproduction using public source codes. We tested LIFT only on Oxf5k and Par6k due to its slow speed. (*) denotes the results from the original papers.)

Dataset	Oxf5k	Oxf105k	Par6k	Par106k
DIR [11]	86.1	82.8	94.5	90.6
DIR+QE [11]	87.1	85.2	95.3	91.8
siaMAC [29]	77.1	69.5	83.9	76.3
siaMAC+QE [29]	81.7	76.6	86.2	79.8
CONGAS [8]	70.8	61.1	67.1	56.8
LIFT [40]	54.0	—	53.6	—
DIR+QE* [11]	89.0	87.8	93.8	90.5
siaMAC+QE* [29]	82.9	77.9	85.6	78.3
DELF+FT+ATT (ours)	83.8	82.6	85.0	81.7
DELF+FT+ATT+DIR+QE (ours)	90.0	88.5	95.7	92.8

6. Conclusion

We presented DELF, a new local feature descriptor that is designed specifically for large-scale image retrieval applications. DELF is learned with weak supervision, using image-level labels only, and is coupled with our new attention mechanism for semantic feature selection. In the proposed CNN-based model, one forward pass over the network is sufficient to obtain both keypoints and descriptors. To properly evaluate performance of large-scale image retrieval algorithm, we introduced Google-Landmarks dataset, which consists of more than 1M database images, 13K unique landmarks, and 100K query images. The evaluation in such a large-scale setting shows that DELF outperforms existing global and local descriptors by substantial margins. We also present results on existing datasets, and show that DELF achieves excellent performance when combined with global descriptors.

Acknowledgement This work was performed while the first and last authors were in Google Inc., CA. It was partly supported by the ICT R&D program of MSIP/IITP [2016-0-00563] and the NRF grant [NRF-2011-0031648] in Korea.

References

- [1] H. Aradhye, G. Toderici, and J. Yagnik. Video2text: Learning to Annotate Video Content. In *Proc. IEEE International Conference on Data Mining Workshops*, 2009. 6
- [2] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. In *Proc. CVPR*, 2016. 1, 2
- [3] A. Babenko and V. Lempitsky. Aggregating Local Deep Features for Image Retrieval. In *Proc. ICCV*, 2015. 2
- [4] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural Codes for Image Retrieval. In *Proc. ECCV*, 2014. 2, 3, 5
- [5] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, 2008. 2
- [6] J. S. Beis and D. G. Lowe. Shape Indexing Using Approximate Nearest-Neighbour Search in High-Dimensional Spaces. In *Proc. CVPR*, 1997. 2
- [7] J. L. Bentley. Multidimensional Binary Search Trees Used for Associative Searching. *Communications of the ACM*, 19(9), 1975. 5
- [8] U. Buddemeier and H. Neven. Systems and Methods for Descriptor Vector Computation, 2012. US Patent 8,098,938. 2, 6, 8
- [9] C. Dugas, Y. Bengio, C. Nadeau, and R. Garcia. Incorporating Second-Order Functional Knowledge for Better Option Pricing. In *Proc. NIPS*, 2001. 4
- [10] M. Fischler and R. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381–395, 1981. 5
- [11] A. Gordo, J. Almazan, J. Revaud, and D. Larlus. Deep Image Retrieval: Learning Global Representations for Image Search. In *Proc. ECCV*, 2016. 1, 2, 5, 8
- [12] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg. MatchNet: Unifying Feature and Metric Learning for Patch-Based Matching. In *Proc. CVPR*, 2015. 1, 2
- [13] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Proc. CVPR*, 2016. 3, 5
- [14] S. Hong, H. Noh, and B. Han. Decoupled Deep Neural Network for Semi-supervised Semantic Segmentation. In *Proc. NIPS*, 2015. 2
- [15] H. Jégou and O. Chum. Negative Evidences and Co-Occurrences in Image Retrieval: The Benefit of PCA and Whitening. In *Proc. ECCV*, 2012. 5
- [16] H. Jégou, M. Douze, and C. Schmid. Hamming Embedding and Weak Geometric Consistency for Large Scale Image Search. In *Proc. ECCV*, 2008. 1, 2
- [17] H. Jegou, M. Douze, and C. Schmid. Product Quantization for Nearest Neighbor Search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1), 2011. 5
- [18] H. Jégou, M. Douze, C. Schmidt, and P. Perez. Aggregating Local Descriptors into a Compact Image Representation. In *Proc. CVPR*, 2010. 2
- [19] H. Jégou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid. Aggregating Local Image Descriptors into Compact Codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9), 2012. 2
- [20] Y. Kalantidis and Y. Avrithis. Locally Optimized Product Quantization for Approximate Nearest Neighbor Search. In *Proc. CVPR*, 2014. 5
- [21] Y. Kalantidis, C. Mellina, and S. Osindero. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features. In *Proc. ECCV Workshops*, 2015. 2
- [22] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 2004. 1, 2, 4
- [23] H. Neven, G. Rose, and W. G. Macready. Image Recognition with an Adiabatic Quantum Computer I. Mapping to Quadratic Unconstrained Binary Optimization. *arXiv:0804.4457*, 2008. 6
- [24] J. Y.-H. Ng, F. Yang, and L. S. Davis. Exploiting Local Features from Deep Networks for Image Retrieval. In *Proc. CVPR Workshops*, 2015. 2
- [25] D. Nistér and H. Stewenius. Scalable Recognition with a Vocabulary Tree. In *Proc. CVPR*, 2006. 2
- [26] F. Perronnin, Y. Liu, and J.-M. Renders. A Family of Contextual Measures of Similarity between Distributions with Application to Image Retrieval. In *Proc. CVPR*, 2009. 6
- [27] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object Retrieval with Large Vocabularies and Fast Spatial Matching. In *Proc. CVPR*, 2007. 1, 2
- [28] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases. In *Proc. CVPR*, 2008. 1, 2
- [29] F. Radenović, G. Tolias, and O. Chum. CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples. In *Proc. ECCV*, 2016. 1, 2, 5, 8
- [30] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Proc. NIPS*, 2015. 2
- [31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), 2015. 3
- [32] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proc. ICLR*, 2015. 5
- [33] J. Sivic and A. Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proc. ICCV*, 2003. 1
- [34] G. Tolias, R. Sicre, and H. Jégou. Particular Object Retrieval with Integral Max-Pooling of CNN Activations. In *Proc. ICLR*, 2015. 2
- [35] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi. Visual Place Recognition with Repetitive Structures. In *Proc. CVPR*, 2013. 2

- [36] T. Uricchio, M. Bertini, L. Seidenari, and A. Bimbo. Fisher Encoded Convolutional Bag-of-Windows for Efficient Image Retrieval and Social Image Tagging. In *Proc. ICCV Workshops*, 2015. [2](#)
- [37] Y. Verdie, K. M. Yi, P. Fua, and V. Lepetit. TILDE: A Temporally Invariant Learned Detector. In *Proc. CVPR*, 2015. [2](#)
- [38] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proc. ICML*, 2015. [2](#)
- [39] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked Attention Networks for Image Question Answering. In *Proc. CVPR*, 2016. [2](#)
- [40] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua. LIFT: Learned Invariant Feature Transform. In *Proc. ECCV*, 2016. [1](#), [2](#), [4](#), [5](#), [6](#), [8](#)
- [41] K. M. Yi, Y. Verdie, P. Fua, and V. Lepetit. Learning to Assign Orientations to Feature Points. In *Proc. CVPR*, 2016. [2](#)
- [42] S. Zagoruyko and N. Komodakis. Learning to Compare Image Patches via Convolutional Neural Networks. In *Proc. CVPR*, 2015. [1](#), [2](#)
- [43] L. Zheng, Y. Yang, and Q. Tian. SIFT Meets CNN: A Decade Survey of Instance Retrieval. *arXiv:1608.01807*, 2016. [2](#)
- [44] Y.-T. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T.-S. Chua, and H. Neven. Tour the World: Building a Web-Scale Landmark Recognition Engine. In *Proc. CVPR*, 2009. [2](#), [6](#)
- [45] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. In *CVPR*, 2016. [2](#)