Revisiting Oxford and Paris: Large-Scale Image Retrieval Benchmarking

Filip Radenović[1] Ahmet Iscen[1] Giorgos Tolias[1] Yannis Avrithis[2] Ondřej Chum[1]
[1]VRG, FEE, CTU in Prague [2]Inria Rennes

Abstract

In this paper we address issues with image retrieval benchmarking on standard and popular Oxford 5k and Paris 6k datasets. In particular, annotation errors, the size of the dataset, and the level of challenge are addressed: new annotation for both datasets is created with an extra attention to the reliability of the ground truth. Three new protocols of varying difficulty are introduced. The protocols allow fair comparison between different methods, including those using a dataset pre-processing stage. For each dataset, 15 new challenging queries are introduced. Finally, a new set of 1M hard, semi-automatically cleaned distractors is selected.

An extensive[1] comparison of the state-of-the-art methods is performed on the new benchmark. Different types of methods are evaluated, ranging from local-feature-based to modern CNN based methods. The best results are achieved by taking the best of the two worlds. Most importantly, image retrieval appears far from being solved.

1. Introduction

Image retrieval methods have gone through significant development in the last decade, starting with descriptors based on local-features, first organized in bag-of-words [41], and further expanded by spatial verification [33], hamming embedding [16], and query expansion [7]. Compact representations reducing the memory footprint and speeding up queries started with aggregating local descriptors [18]. Nowadays, the most efficient retrieval methods are based on fine-tuned convolutional neural networks (CNNs) [10, 37, 30].

In order to measure the progress and compare different methods, standardized image retrieval benchmarks are used. Besides the fact that a benchmark should simulate a real-world application, there are a number of properties that determine the quality of a benchmark: the *reliability of the annotation*, the *size*, and the *challenge level*.

The authors were supported by the MSMT LL1303 ERC-CZ grant.
[1]We thank Facebook for the donation of GPU servers, which made the evaluation tractable.

Errors in the annotation may systematically corrupt the comparison of different methods. Too small datasets are prone to over-fitting and do not allow the evaluation of the efficiency of the methods. The reliability of the annotation and size of the dataset are competing factors, as it is difficult to secure accurate human annotation of large datasets. The size is commonly increased by adding a distractor set, which contains irrelevant images that are selected in an automated manner (different tags, GPS information, *etc*.) Finally, benchmarks where all the methods achieve almost perfect results [23] cannot be used for further improvement or qualitative comparison.

Many datasets have been introduced to measure the performance of image retrieval. Oxford [33] and Paris [34] datasets belong to the most popular ones. Numerous methods of image retrieval [7, 31, 5, 27, 47, 3, 48, 20, 37, 10] and visual localization [9, 1] have used these datasets for evaluation. One reason for their popularity is that, in contrast to datasets that contain small groups of 4-5 similar images like Holidays [16] and UKBench [29], Oxford and Paris contain queries with up to hundreds of positive images.

Despite the popularity, there are known issues with the two datasets, which are related to all three important properties of evaluation benchmarks. First, there are errors in the annotation, including both false positives and false negatives. Further inaccuracy is introduced by queries of different sides of a landmark, sharing the annotation despite being visually distinguishable. Second, the annotated datasets are relatively small (5,062 and 6,392 images respectively). Third, current methods report near-perfect results on both the datasets. It has become difficult to draw conclusions from quantitative evaluations, especially given the annotation errors [14].

The lack of difficulty is not caused by the fact that non-trivial instances are not present in the dataset, but due to the annotation. The annotation was introduced about ten years ago. At that time, the annotators had different perception of what the limits of image retrieval are. Many instances that are nowadays considered as a change of viewpoint expected to be retrieved, are *de facto* excluded from the evaluation by being labelled as *Junk*.

The size issue of the datasets is partially addressed by the Oxford 100k *distractor set*. However, this contains false negative images, as well as images that are not challenging. State-of-the-art methods maintain near-perfect results even in the presence of these distractors. As a result, additional computational effort is spent with little benefit in drawing conclusions.

Contributions. As a first contribution, we generate new annotation for Oxford and Paris datasets, update the evaluation protocol, define new, more difficult queries, and create new set of challenging distractors. As an outcome we produce *Revisited Oxford*, *Revisited Paris*, and an accompanying distractor set of one million images. We refer to them as $\mathcal{R}$Oxford, $\mathcal{R}$Paris, and $\mathcal{R}$1M respectively.

As a second contribution, we provide extensive evaluation of image retrieval methods, ranging from local-feature based to CNN-descriptor based approaches, including various methods of re-ranking.

2. Revisiting the datasets

In this section we describe in detail why and how we revisit the annotation of Oxford and Paris datasets, present a new evaluation protocol and an accompanying challenging set of one million distractor images. The revisited benchmark is publicly available[2].

2.1. The original datasets

The original Oxford and Paris datasets consist of 5,063 and 6,392 high-resolution (1024×768) images, respectively. Each dataset contains 55 queries comprising 5 queries per landmark, coming from a total of 11 landmarks. Given a landmark query image, the goal is to retrieve all database images depicting the same landmark. The original annotation (labeling) is performed manually and consists of 11 ground truth lists since 5 images of the same landmark form a *query group*. Three labels are used, namely, *positive*, *junk*, and *negative* [3].

Positive images clearly depict more than 25% of the landmark, junk less than 25%, while the landmark is not shown in negative ones. The performance is measured via mean average precision (mAP) [33] over all 55 queries, while junk images are ignored, *i.e.* the evaluation is performed as if they were not present in the database.

2.2. Revisiting the annotation

The annotation is performed by five annotators, and it is performed in the following steps.

[2] cmp.felk.cvut.cz/revisitop

[3] We rename the originally used labels {good, ok, junk, and absent} for the purpose of consistency with our terminology. Good and ok were always used as positives.

Query groups. Query groups share the same ground-truth list and simplify the labeling problem, but also cause some inaccuracies in the original annotation. *Balliol* and *Christ Church* landmarks are depicted from a different (not fully symmetric) side in the 2^{nd} and 4^{th} query, respectively. *Arc de Triomphe* has three day and two night queries, while day-night matching is considered a challenging problem [49, 35]. We alleviate this by splitting these cases into separate groups. As a result, we form 13 and 12 query groups on Oxford and Paris, respectively.

Additional queries. We introduce new and more challenging queries (see Figure 1) compared to the original ones. There are 15 new queries per dataset, originating from five out of the original 11 landmarks, with three queries per landmark. Along with the 55 original queries, they comprise the new set of 70 queries per dataset. The query groups, defined by visual similarity, are 26 and 25 for $\mathcal{R}$Oxford and $\mathcal{R}$Paris, respectively. As in the original datasets, the query object bounding boxes are simulating not only a user attempting to remove background clutter, but also cases of large occlusion.

Labeling step 1: Selection of potential positives. Each annotator manually inspects the whole dataset and marks images depicting any side or version of a landmark. The goal is to collect all images that are originally incorrectly labeled as negative. Even uncertain cases are included in this step and the process is repeated for each landmark. Apart from inspecting the whole dataset, an interactive retrieval tool is used to actively search for further possible positive images. All images marked in this phase are merged together with images originally annotated as positive or junk, creating a list of *potential positives* for each landmark.

Labeling step 2: Label assignment. In this step, each annotator manually inspects the list of potential positives for each query group and assigns labels. The possible labels are *Easy*, *Hard*, *Unclear*, and *Negative*. All images not in the list of potential positives are automatically marked negative. The instructions given to the annotators for each of the labels are as follows.

- *Easy:* The image clearly depicts the query landmark from the same side, with no large viewpoint change, no significant occlusion, no extreme illumination change, and no severe background clutter. In the case of fully symmetric sides, any side is valid.

$\mathcal{R}$Oxford				
Labels	Easy	Hard	Uncl.	Neg.
Positive	438	50	93	1
Junk	50	222	72	9
Negative	1	72	133	63768

$\mathcal{R}$Paris				
Labels	Easy	Hard	Uncl.	Neg.
Positive	1222	643	136	6
Junk	91	813	835	61
Negative	16	147	273	71621

Table 1. Number of images switching their labeling from the original annotation (positive, junk, negative) to the new one (easy, hard, unclear, negative).

Figure 1. The newly added queries for $\mathcal{R}$Oxford(top) and $\mathcal{R}$Paris(bottom) datasets. Merged with the original queries, they comprise a new set of 70 queries in total.

Figure 2. Examples of *extreme* labeling mistakes in the original labeling. We show the **query (blue)** image and the associated database images that were originally marked as **negative (red)** or **positive (green)**. Best viewed in color.

- *Hard:* The image depicts the query landmark, but with viewing conditions that are difficult to match with the query. The depicted (side of the) landmark is recognizable without any contextual visual information.
- *Unclear:* (a) The image possibly depicts the landmark in question, but the content is not enough to make a certain guess about the overlap with the query region, or context is needed to clarify. (b) The image depicts a different side of a partially symmetric building, where the symmetry is significant and discriminative enough.
- *Negative:* The image is not satisfying any of the previous conditions. For instance, it depicts a different side of the landmark compared to that of the query, with no discriminative symmetries. If the image has any physical overlap with the query, it is never negative, but rather unclear, easy, or hard according to the above.

Labeling step 3: Refinement. For each query group, each image in the list of potential positives has been assigned a five-tuple of labels, one per annotator. We perform majority voting in two steps to define the final label. The first step is voting for {easy,hard}, {unclear}, or {negative}, grouping easy and hard together. In case majority goes to {easy,hard}, the second step is to decide which of the two. Draws of the first step are assigned to unclear, and of the second step to hard. Illustrative examples are (EEHUU) $\rightarrow$ E, (EHUUN) $\rightarrow$ U, and (HHUNN) $\rightarrow$ U. Finally, for each query group, we inspect images by descending label entropy to make sure there are no errors.

Revisited datasets: $\mathcal{R}$Oxford and $\mathcal{R}$Paris. Images from which the queries are cropped are excluded from the evaluation dataset. This way, unfair comparisons are avoided in the case of methods performing off-line preprocessing of the database [2, 14]; any preprocessing should not include any part of query images. The revisited datasets, namely, $\mathcal{R}$Oxford and $\mathcal{R}$Paris, comprise 4,993 and 6,322 images respectively, after removing the 70 queries.

In Table 1, we show statistics of label transitions from the old to the new annotations. Note that errors in the original annotation that affect the evaluation, *e.g.* negative moving to easy or hard, are not uncommon. The transitions from junk to easy or hard are reflecting the greater challenges of the new annotation. Representative examples of *extreme* labeling errors of the original annotation are shown in Figure 2. In Figure 3, representative examples of easy, hard, and unclear images are presented for several queries. This will help understanding the level of challenge of each evaluation protocol listed below.

2.3. Evaluation protocol

Only the cropped regions are to be used as queries; never the full image, since the ground-truth labeling strictly considers only the visual content inside the query region.

The standard practice of reporting mean average precision (mAP) [33] for performance evaluation is followed. Additionally, mean precision at rank K (mP@K) is reported. The former reflects the overall quality of the ranked list. The latter reflects the quality of the results of a search engine as they would be visually inspected by a user. More importantly, it is correlated to performance of subsequent processing steps [7, 21]. During the evaluation, positive images should be retrieved, while there is also an ignore list per query. Three evaluation setups of different difficulty are defined by treating labels (easy, hard, unclear) as positive or negative, or ignoring them:

- **Easy (E):** *Easy* images are treated as positive, while *Hard* and *Unclear* are ignored (same as *Junk* in [33]).
- **Medium (M):** *Easy* and *Hard* images are treated as positive, while *Unclear* are ignored.
- **Hard (H):** *Hard* images are treated as positive, while *Easy* and *Unclear* are ignored.

If there are no positive images for a query in a particular setting, then that query is excluded from the evaluation.

Figure 3. Sample **query (blue)** images and images that are respectively marked as **easy (dark green)**, **hard (light green)**, and **unclear (yellow)**. Best viewed in color.

Figure 4. Sample false negative images in Oxford100k.

The original annotation and evaluation protocol is closest to our **Easy** setup. Even though this setup is now trivial for the best performing methods, it can still be used for evaluation of *e.g.* near duplicate detection or retrieval with ultra short codes. The other setups, **Medium** and **Hard**, are challenging and even the best performing methods achieve relatively low scores. See Section 4 for details.

2.4. Distractor set $\mathcal{R}$1M

Large scale experiments on Oxford and Paris dataset are commonly performed with the accompanying distractor set of 100k images, namely Oxford100k [33]. Recent results [14, 13] show that the performance only slightly degrades by adding Oxford100k in the database compared to a small-scale setting. Moreover, it is not manually cleaned and, as a consequence, Oxford and Paris landmarks are depicted in some of the distractor images (see Figure 4), hence adding further noise to the evaluation procedure.

Larger distactor sets are used in the literature [33, 34, 16, 44] but none of them are standardized to provide a testbed for direct large scale comparison nor are they manually cleaned [16]. Some of the distractor sets are also biased, since they contain images of different resolution than the Oxford and Paris datasets.

We construct a new distractor set with exactly 1,001,001 high-resolution (1024×768) images, which we refer to as $\mathcal{R}$1M dataset. It is cleaned by a semi-automatic process. We automatically pick hard images for a number of state-of-the-art methods, resulting in a challenging large scale setup.

YFCC100M and semi-automatic cleaning. We randomly choose 5M images with GPS information from YFCC100M dataset [43]. Then, we exclude UK, France, and Las Vegas; the latter due to the *Eiffel Tower* and *Arc de Triomphe* replicas. We end up with roughly 4.1M images that are available for downloading in high resolution. We rank images with the same search tool as used in labeling step 1. Then, we manually inspect the top 2k images per landmark, and remove those depicting the query landmarks (faulty GPS, toy models, and paintings/photographs of landmarks). In total, we find 110 such images.

Un-biased mining of distracting images. We propose a way to keep the most challenging 1M out of the 4.1M images. We perform all 70 queries into the 4.1M database with a number of methods. For each query and for each distractor image we count the fraction of easy or hard images that are ranked after it. We sum these fractions over all queries of $\mathcal{R}$Oxford and $\mathcal{R}$Paris and over different methods, resulting in a measurement of how *distracting* each distractor image is. We choose the set of 1M most distracting images and refer to it as the $\mathcal{R}$1M *distractor set*.

Three complementary retrieval methods are chosen to compute this measurement. These are fine-tuned ResNet with GeM pooling [37], pre-trained (on ImageNet) AlexNet with MAC pooling [38], and ASMK [46]. More details on these methods are given in Section 3. Finally, we perform a sanity check to show that this selection process is not significantly biased to distract only those 3 methods. This includes two additional methods, VLAD [18] and fine-tuned ResNet with R-MAC pooling by Gordo *et al.* [10]. As shown in Table 2, the performance on the hardest 1M distractors is hardly affected whether one of those additional methods participates or not in the selection process. This suggests that the mining process is not biased towards particular methods.

Table 2 also shows that the distractor set we choose (version 1M (1,2,3) in the Table) is much harder than a random 1M subset and nearly as hard as all 4M distractor images. Example images from the set $\mathcal{R}$1M are shown in Figure 5.

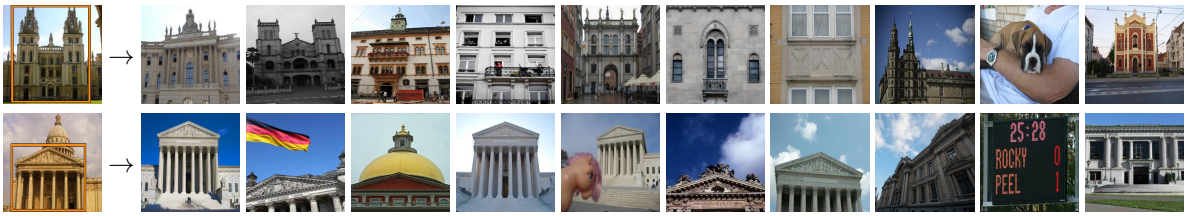

Figure 5. The most distracting images per query for two queries.

3. Extensive evaluation

We evaluate a number of state-of-the-art approaches on the new benchmark and offer a rich testbed for future comparisons. We list them in this section and they belong to two main categories, namely, classical retrieval approaches using local features and CNN-based methods producing global image descriptors.

3.1. Local-feature-based methods

Methods based on local invariant features [25, 26] and the Bag-of-Words (BoW) model [41, 33, 7, 34, 6, 27, 47, 4, 52, 54, 42] were dominating the field of image retrieval until the advent of CNN-based approaches [38, 3, 48, 20, 1, 10, 37, 28, 51]. A typical pipeline consists of invariant local feature detection [26], local descriptor extraction [25], quantization with a visual codebook [41], typically created with k-means, assignment of descriptors to visual words and finally descriptor aggregation in a single embedding [19, 32] or individual feature indexing with an inverted file structure [45, 33, 31]. We consider state-of-the-art methods from both categories. In particular, we use up-right hessian-affine (HesAff) features [31], RootSIFT (rSIFT) descriptors [2], and create the codebooks on the landmark dataset from [37], same as the one used for the whitening of CNN-based methods. Note that we always crop the queries according to the defined region and then perform any processing to be directly comparable to CNN-based methods.

Distractor set	$\mathcal{R}$Oxford					$\mathcal{R}$Paris				
	Method									
	(1)	(2)	(3)	(4)	(5)	(1)	(2)	(3)	(4)	(5)
4M	33.3	11.1	33.2	33.7	15.6	40.7	11.4	30.0	45.4	17.9
1M (1,2,3)	33.9	11.1	34.8	33.9	17.4	44.1	11.8	31.7	48.1	19.6
1M (1,2,3,4)	33.7	11.1	34.8	33.8	17.5	43.8	11.8	31.8	47.7	19.7
1M (1,2,3,5)	33.7	11.1	34.6	33.9	17.2	43.5	11.7	31.4	47.7	19.2
1M (random)	37.6	13.7	37.4	38.9	20.4	47.3	16.2	34.2	53.1	21.9

Table 2. Performance (mAP) evaluation with the Medium protocol for different distractor sets. The methods considered are (1) Fine-tuned ResNet101 with GeM pooling [37]; (2) Off-the-shelf AlexNet with MAC pooling [38]; (3) HesAff–rSIFT–ASMK* [46]; (4) Fine-tuned ResNet101 with R-MAC pooling [10]; (5) HesAff–rSIFT–VLAD [18]. The sanity check includes evaluation for different distractor sets, *i.e.* all, hardest subset chosen by method (1,2,3), (1,2,3,4), (1,2,4,5), and a random 1M sample.

We additionally follow the same BoW-based pipeline while replacing hessian-affine and RootSIFT with the deep local attentive features (DELF) [30]. The default extraction approach is followed (*i.e.* at most 1000 features per image), but we reduce the descriptor dimensionality to 128 and not to 40 to be comparable to RootSIFT. This variant is a bridge between classical approaches and deep learning.

VLAD. The Vector of Locally Aggregated Descriptors [18] (VLAD) is created by first-order statistics of the local descriptors. The residual vectors between descriptors and the closest centroid are aggregated w.r.t. a codebook whose size is 256 in our experiments. We reduce its dimensionality down to 2048 with PCA, while square-root normalization is also used [15].

SMK*. The binarized version of the Selective Match Kernel [46] (SMK*), a simple extension of the Hamming Embedding [16] (HE) technique, uses an inverted file structure to separately indexes binarized residual vectors while it performs the matching with a selective monomial kernel function. The codebook size is 65,536 in our experiments, while burstiness normalization [17] is always used. Multiple assignment to three nearest words is used on the query side, while the hamming distance threshold is set to 52 out of 128 bits. The rest are the default parameters.

ASMK*. The binarized version of the Aggregated Selective Match Kernel [46] (ASMK*) is an extension of SMK* that jointly encodes local descriptors that are assigned to the same visual word and handles the burstiness phenomenon. Same parametrization as SMK* is used.

SP. Spatial verification (SP) is known to be crucial for particular object retrieval [33] and is performed with the RANSAC algorithm [8]. It is applied on the 100 top-ranked images, as these are formed by a first filtering step, *e.g.* the SMK* or ASMK* method. Its result is the number of inlier correspondences, which is one of the most intuitive similarity measures and allows to detect true positive images. To assume that an image is spatially verified, we require 5 inliers with ASMK* and 10 with other methods.

HQE. Query expansion (QE), firstly introduced by Chum *et al.* [7] in the visual domain, typically uses spatial verification to select true positive among the top retrieved result and issues an enhanced query including the verified images. Hamming Query Expansion [47] (HQE) is combining QE with HE. We use same soft assignment as SMK* and the default parameters.

3.2. CNN-based global descriptor methods

We list different aspects of a CNN-based method for image retrieval, which we later combine to form different baselines that exist in the literature.

CNN architectures. We include 3 highly influential CNN architectures, namely AlexNet [22], VGG-16 [40], and ResNet101 [12]. They have different number of layers, complexity, and also produce descriptors of different dimensionality (256, 512, and 2048, respectively).

Pooling. A common practice is to consider a convolutional feature map and perform a pooling mechanism to construct a global image descriptor. We consider max-pooling (MAC) [38, 48], sum-pooling (SPoC) [3], weighted sum-pooling (CroW) [20], regional max-pooling (R-MAC) [48], generalized mean-pooling (GeM) [37], and NetVLAD pooling [1]. The pooling is always applied on the last convolutional feature map.

Multi-scale. The input image is resized to a maximum 1024×1024 size. Then, three re-scaled versions with scaling factor of 1, $1/\sqrt{2}$, and $1/2$ are fed to the network. Finally, the resulting descriptors are combined into a single descriptor by average pooling [10] for all methods, except for GeM where generalized-mean pooling is used [37]. This is shown to improve the performance of the CNN-based descriptors [10, 37].

Off-the-shelf *vs*. retrieval fine-tuning. Networks that are pre-trained on ImageNet [39] (off-the-shelf) are directly applicable on image retrieval. Moreover, we consider the following cases of fine-tuning for the task. Radenovic *et al.* [36] fine-tune a network with landmarks photos using contrastive loss [11]. This is available with MAC [36] and GeM pooling [37]. Similarly, Gordo *et al.* [10] fine-tune R-MAC pooling with landmark photos and triplet loss [50]. Finally, NetVLAD [1] is fine-tuned using street-view images and GPS information.

Descriptor whitening is known to be essential for such descriptors. We use the same landmark dataset [37] to learn the whitening for all methods. We use PCA whitening [15, 3] for all the off-the-shelf networks, and supervised whitening with SfM labels [24, 36] for all the fine-tuned ones. One exception is the tuning that includes the whitening in the network [10].

Query Expansion is directly applicable on top of global CNN-based descriptors. More specifically, we use α query expansion [37] (αQE) and diffusion [14] (DFS).

Method	Oxf	$\mathcal{R}$Oxford			Par	$\mathcal{R}$Paris		
		E	M	H		E	M	H
HesAff–rSIFT–SMK*	78.1	74.1	59.4	35.4	74.6	80.6	59.0	31.2
R–[O]–R-MAC	78.3	74.2	49.8	18.5	90.9	89.9	74.0	52.1
R–[37]–GeM	87.8	84.8	64.7	38.5	92.7	92.1	77.2	56.3
R–[37]–GeM+DFS	90.0	86.5	69.8	40.5	95.3	93.9	88.9	78.5

Table 3. Performance (mAP) on Oxford (Oxf) and Paris (Par) with the original annotation, and $\mathcal{R}$Oxford and $\mathcal{R}$Paris with the newly proposed annotation with three different protocol setups: Easy (E), Medium (M), Hard (H).

Method	Memory (GB)	Time (sec)		
		Extraction		Search
		GPU	CPU	
HesAff–rSIFT–ASMK*	62.0	n/a + 0.06	1.08 + 2.35	0.98
HesAff–rSIFT–ASMK*+SP				2.00
DELF–ASMK*+SP	10.3	0.41 + 0.01	n/a + 0.54	0.52
A–[37]–GeM	0.96	0.12	1.99	0.38
V–[37]–GeM	1.92	0.23	31.11	0.56
R–[37]–GeM	7.68	0.37	14.51	1.21

Table 4. Time and memory measurements. Extraction time on a single thread GPU (Tesla P100) / CPU (Intel Xeon CPU E5-2630 v2 @ 2.60GHz) per image of size 1024x768, the memory requirements and the search time (single thread CPU) reported for the database of $\mathcal{R}$Oxford+$\mathcal{R}$1M images. Feature extraction + visual word assignment is reported for ASMK*. SP: Geometry information is loaded from the disk and the loading time is included in search time. We did not consider geometry quantization [31].

4. Results

We report a performance comparison between the old and the revisited datasets. Additionally, we provide an extensive evaluation of the state-of-the-art methods on the revisited dataset, with and without the new large-scale distractor set, setting up a testbed for future comparisons.

The evaluation includes local feature-based approaches (see Section 3.1 for details and abbreviations), referred to by the combination of local feature type and representation method, *e.g.* HesAff–rSIFT–ASMK*. CNN-based global descriptors are denoted with the following abbreviations. Network architectures are AlexNet (A), VGG-16 (V), and ResNet101 (R). The fine-tuning options are triplet loss with GPS guided mining [1], triplet loss with spatially verified positive pairs [10], contrastive loss with mining from 3D models [36] and [37], and finally the off-the-shelf [O] networks. Pooling approaches are as listed in Section 3.2. For instance, ResNet101 with GeM pooling that is fine-tuned with contrastive loss and the training dataset by Radenovic *et al.* [37] is referred to as R–[37]–GeM.

Revisited *vs*. original. We compare the performance when evaluated on the original datasets, and the revisited annotation with the new protocols. The results for four representative methods are presented in Table 3. The old setup appears to be close to the new **Easy** setup, while **Medium** and **Hard** appear to be more challenging. We observe that the performance of the **Easy** setup is nearly saturated and, therefore, do not use it but only evaluate **Medium** and **Hard** setups in the subsequent experiments.

Method	Medium								Hard							
	$\mathcal{R}$Oxf		$\mathcal{R}$Oxf+$\mathcal{R}$1M		$\mathcal{R}$Par		$\mathcal{R}$Par+$\mathcal{R}$1M		$\mathcal{R}$Oxf		$\mathcal{R}$Oxf+$\mathcal{R}$1M		$\mathcal{R}$Par		$\mathcal{R}$Par+$\mathcal{R}$1M	
	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10
HesAff–rSIFT–VLAD	33.9	54.9	17.4	34.8	43.6	90.9	19.6	76.1	13.2	18.1	5.6	7.0	17.5	50.7	3.3	21.1
HesAff–rSIFT–SMK⋆	59.4	83.6	35.8	64.6	59.0	97.4	34.1	89.1	35.4	53.7	16.4	27.7	31.2	72.6	10.5	47.6
HesAff–rSIFT–ASMK⋆	60.4	85.6	45.0	76.0	61.2	97.9	42.0	95.3	36.4	56.7	25.7	42.1	34.5	80.6	16.5	63.4
HesAff–rSIFT–SMK⋆+SP	59.8	84.3	38.1	67.1	59.2	97.4	34.5	89.3	35.8	54.0	17.7	30.3	31.3	73.6	11.0	49.1
HesAff–rSIFT–ASMK⋆+SP	60.6	86.1	46.8	79.6	61.4	97.9	42.3	95.3	36.7	57.0	26.9	45.3	35.0	81.7	16.8	65.3
DELF–ASMK⋆+SP	67.8	87.9	53.8	81.1	76.9	99.3	57.3	98.3	43.1	62.4	31.2	50.7	55.4	93.4	26.4	75.7
A – [O] –MAC	28.3	44.7	14.1	28.3	47.3	88.6	18.7	69.4	8.8	15.5	3.5	5.1	23.1	61.6	4.1	29.0
A – [O] –GeM	33.8	51.2	16.3	32.4	52.7	90.1	23.8	78.1	10.4	16.7	3.9	6.3	26.0	68.0	5.5	31.6
A – [36]–MAC	41.3	62.1	23.9	43.0	56.4	92.9	29.6	85.4	17.8	28.2	8.4	11.9	28.7	69.3	8.5	40.9
A – [37]–GeM	43.3	62.1	24.2	42.8	58.0	91.6	29.9	84.6	17.1	26.2	9.4	11.9	29.7	67.6	8.4	39.6
V – [O] –MAC	37.8	57.8	21.8	39.7	59.2	93.3	33.6	87.1	14.6	27.0	7.4	11.9	35.9	78.4	13.2	54.7
V – [O] –SPoC	38.0	54.6	17.1	33.3	59.8	93.0	30.3	83.0	11.4	20.9	0.9	2.9	32.4	69.7	7.6	30.6
V – [O] –CroW	41.4	58.8	22.5	40.5	62.9	94.4	34.1	87.1	13.9	25.7	3.0	6.6	36.9	77.9	10.3	45.1
V – [O] –GeM	40.5	60.3	25.4	45.6	63.2	94.6	37.5	88.6	15.7	28.6	7.6	12.1	38.8	79.0	14.2	55.9
V – [O] –R-MAC	42.5	62.8	21.7	40.3	66.2	95.4	39.9	88.9	12.0	26.1	1.7	5.8	40.9	77.1	14.8	54.0
V – [1] –NetVLAD	37.1	56.5	20.7	37.1	59.8	94.0	31.8	85.7	13.8	23.3	6.0	8.4	35.0	73.7	11.5	46.6
V – [36]–MAC	58.4	81.1	39.7	68.6	66.8	97.7	42.4	92.6	30.5	48.0	17.9	27.9	42.0	82.9	17.7	63.7
V – [37]–GeM	61.9	82.7	42.6	68.1	69.3	97.9	45.4	94.1	33.7	51.0	19.0	29.4	44.3	83.7	19.1	64.9
R – [O] –MAC	41.7	65.0	24.2	43.7	66.2	96.4	40.8	93.0	18.0	32.9	5.7	14.4	44.1	86.3	18.2	67.7
R – [O] –SPoC	39.8	61.0	21.5	40.4	69.2	96.7	41.6	92.0	12.4	23.8	2.8	5.6	44.7	78.0	15.3	54.4
R – [O] –CroW	42.4	61.9	21.2	39.4	70.4	97.1	42.7	92.9	13.3	27.7	3.3	9.3	47.2	83.6	16.3	61.6
R – [O] –GeM	45.0	66.2	25.6	45.1	70.7	97.0	46.2	94.0	17.7	32.6	4.7	13.4	48.7	88.0	20.3	70.4
R – [O] –R-MAC	49.8	68.9	29.2	48.9	74.0	97.7	49.3	93.7	18.5	32.2	4.5	13.0	52.1	87.1	21.3	67.4
R – [37]–GeM	64.7	84.7	45.2	71.7	77.2	98.1	52.3	95.3	38.5	53.0	19.9	34.9	56.3	89.1	24.7	73.3
R – [10]–R-MAC	60.9	78.1	39.3	62.1	78.9	96.9	54.8	93.9	32.4	50.0	12.5	24.9	59.4	86.1	28.0	70.0
Query expansion (QE) and diffusion (DFS)																
HesAff–rSIFT–HQE	66.3	85.6	42.7	67.4	68.9	97.3	44.2	90.1	41.3	60.0	23.2	37.6	44.7	79.9	20.3	51.4
HesAff–rSIFT–HQE+SP	71.3	88.1	52.0	76.7	70.2	98.6	46.8	93.0	49.7	69.6	29.8	50.1	45.1	83.9	21.8	61.9
DELF–HQE+SP	73.4	88.2	60.6	79.7	84.0	98.3	65.2	96.1	50.3	67.2	37.9	56.1	69.3	93.7	35.8	69.1
R – [O] –R-MAC+αQE	51.9	70.3	30.8	49.7	77.3	97.9	55.3	94.7	21.8	35.2	5.2	15.9	57.0	87.6	28.0	76.1
V – [37]–GeM+αQE	66.6	85.7	47.0	72.0	74.0	98.4	52.9	95.9	38.9	57.3	21.1	34.6	51.0	88.4	25.6	75.0
R – [37]–GeM+αQE	67.2	86.0	49.0	74.7	80.7	98.9	58.0	95.9	40.8	54.9	24.2	40.3	61.8	90.6	31.0	80.4
R – [10]–R-MAC+αQE	64.8	78.5	45.7	66.5	82.7	97.3	61.0	94.3	36.8	53.3	19.5	36.6	65.7	90.1	35.0	76.9
V – [37]–GeM+DFS	69.6	84.7	60.4	79.4	85.6	97.1	80.7	97.1	41.1	51.1	33.1	49.6	73.9	93.7	65.3	93.1
R – [37]–GeM+DFS	69.8	84.0	61.5	77.1	88.9	96.9	84.9	95.9	40.5	54.4	33.1	48.2	78.5	94.6	71.6	93.7
R – [10]–R-MAC+DFS	69.0	82.3	56.6	68.6	89.5	96.7	83.2	93.3	44.7	60.5	28.4	43.6	80.0	94.1	70.4	89.1
HesAff–rSIFT–ASMK⋆+SP → R– [37]–GeM+DFS	79.1	92.6	74.3	87.9	91.0	98.3	85.9	97.1	52.7	66.1	48.7	65.9	81.0	97.9	73.2	96.6
HesAff–rSIFT–ASMK⋆+SP → R– [10]–R-MAC+DFS	80.2	93.7	74.9	87.9	92.5	98.7	87.5	97.1	54.8	70.6	47.5	62.4	84.0	98.3	76.0	96.3
DELF–ASMK⋆+SP → R– [10]–R-MAC+DFS	75.0	87.9	68.7	83.6	90.5	98.0	86.6	98.1	48.3	64.0	39.4	55.7	81.2	95.6	74.2	94.6

Table 5. Performance evaluation (mAP, mP@10) on $\mathcal{R}$Oxford ($\mathcal{R}$Oxf) and $\mathcal{R}$Paris ($\mathcal{R}$Par) without and with $\mathcal{R}$1M distractors. We report results with the revisited annotation, using Medium and Hard evaluation protocols. We use a color-map that is normalized according to the minimum (white) and maximum (**green** / **orange**) value per column.

State of the art evaluation. We perform an extensive evaluation of the state-of-the-art methods for image retrieval. We present time/memory measurements in Table 4 and performance results in Table 5. We additionally show the average precision (AP) per query for a set of representative methods in Figures 6 and 7, for $\mathcal{R}$Oxford and $\mathcal{R}$Paris, respectively. The representative set covers the progress of methods over time in the task of image retrieval. In the evaluation, we observe that there is no single method achieving the highest score on every protocol per dataset. Local-feature-based methods perform very well on $\mathcal{R}$Oxford, especially at large scale, achieving state-of-the-art performance, while CNN-based methods seem to dominate on $\mathcal{R}$Paris. We observe that BoW-based classical approaches are still not obsolete, but their improvement typically comes at significant additional cost. Recent CNN-based local features, *i.e.* DELF, reduce the number of features and improve the performance at the same time.

CNN fine-tuning consistently brings improvements over the off-the-shelf networks. The new protocols make it clear that improvements are needed at larger scale and the hard setup. Many images are not retrieved, while the top 10 results mostly contain false positives. Interestingly, we observe that query expansion approaches (*e.g.* diffusion) degrade the performance of queries with few relevant images (see Figures 6 and 7). This phenomenon is more pronounced in the revisited datasets, where the the query images are removed from the preprocessing. We did not include separate regional representation and indexing [38], which is previously shown to be beneficial. Preliminary experiments with ResNet and GeM pooling show that it does not deliver improvements that are significant enough to justify the additional memory and complexity cost.

The best of both worlds. The new dataset and protocols reveal space for improvement by CNN-based global descriptors in cases where local features are still better. Diffu-

Figure 6. Performance (AP) per query on $\mathcal{R}$Oxford + $\mathcal{R}$1M with Medium setup. AP is shown with a bar for 8 methods. The methods, from left to right, are HesAff–rSIFT–ASMK⋆+SP, DELF–ASMK⋆+SP, DELF–HQE+SP, V–[O]–R-MAC, R–[O]–GeM, R–[37]–GeM, R–[37]–GeM+DFS, HesAff–rSIFT–ASMK⋆+SP → R–[37]–GeM+DFS. The total number of easy and hard images is printed on each histogram. Best viewed in color.

Figure 7. Performance (AP) per query on $\mathcal{R}$Paris + $\mathcal{R}$1M with Medium setup. AP is shown with a bar for 8 methods. The methods, from left to right, are HesAff–rSIFT–ASMK⋆+SP, DELF–ASMK⋆+SP, DELF–HQE+SP, V–[O]–R-MAC, R–[O]–GeM, R–[37]–GeM, R–[37]–GeM+DFS, HesAff–rSIFT–ASMK⋆+SP → R–[37]–GeM+DFS. The total number of easy and hard images is printed on each histogram. Best viewed in color.

sion performs similarity propagation by starting from the query's nearest neighbors according to the CNN global descriptor. This inevitably includes false positives, especially in the case of few relevant images. On the other hand, local features, *e.g.* with ASMK⋆+SP, offer a verified list of relevant images. Starting the diffusion process from geometrically verified images obtained by BoW methods combines the benefits of the two worlds. This combined approach, shown at the bottom part of Table 5, improves the performance and supports the message that both worlds have their own benefits. Of course this experiment is expensive and we perform it to merely show a possible direction to improve CNN global descriptors. There are more methods that combine CNNs and local features [53], but we focus on the results related to methods included in our evaluation.

5. Conclusions

We have revisited two of the most established image retrieval datasets, that were perceived as performance saturated. To make it suitable for modern image retrieval benchmarking, we address drawbacks of the original annotation. This includes new annotation for both datasets that was created with an extra attention to the reliability of the ground truth, and an introduction of 1M hard distractor set.

An extensive evaluation provides a testbed for future comparisons and concludes that image retrieval is still an open problem, especially at large scale and under difficult viewing conditions.

References

[1] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *CVPR*, 2016. 1, 5, 6, 7

[2] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012. 3, 5

[3] A. Babenko and V. Lempitsky. Aggregating deep convolutional features for image retrieval. In *ICCV*, 2015. 1, 5, 6

[4] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang. Spatial-bag-of-features. In *CVPR*, 2010. 5

[5] O. Chum, A. Mikulik, M. Perdoch, and J. Matas. Total recall II: Query expansion revisited. In *CVPR*, 2011. 1

[6] O. Chum, M. Perdoch, and J. Matas. Geometric min-hashing: Finding a (thick) needle in a haystack. In *CVPR*, 2009. 5

[7] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *ICCV*, 2007. 1, 3, 5

[8] M. A. Fischler and R. C. Bolles. Random sample consensus. *Communications of ACM*, 1981. 5

[9] S. Gammeter, L. Bossard, T. Quack, and L. Van Gool. I know what you did last summer: object-level auto-annotation of holiday snaps. In *CVPR*, 2009. 1

[10] A. Gordo, J. Almazan, J. Revaud, and D. Larlus. End-to-end learning of deep visual representations for image retrieval. *IJCV*, 2017. 1, 4, 5, 6, 7

[11] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006. 6

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[13] A. Iscen, Y. Avrithis, G. Tolias, T. Furon, and O. Chum. Fast spectral ranking for similarity search. In *CVPR*, 2018. 4

[14] A. Iscen, G. Tolias, Y. Avrithis, T. Furon, and O. Chum. Efficient diffusion on region manifolds: Recovering small objects with compact cnn representations. In *CVPR*, 2017. 1, 3, 4, 6

[15] H. Jégou and O. Chum. Negative evidences and cooccurences in image retrieval: The benefit of PCA and whitening. In *ECCV*, 2012. 5, 6

[16] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008. 1, 4, 5

[17] H. Jégou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *CVPR*, 2009. 5

[18] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010. 1, 4, 5

[19] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local descriptors into compact codes. *PAMI*, 2012. 5

[20] Y. Kalantidis, C. Mellina, and S. Osindero. Cross-dimensional weighting for aggregated deep convolutional features. In *ECCVW*, 2016. 1, 5, 6

[21] Y. Kalantidis, G. Tolias, Y. Avrithis, M. Phinikettos, E. Spyrou, P. Mylonas, and S. Kollias. Viral: Visual image retrieval and localization. *Multimedia Tools and Applications*, 2011. 3

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 6

[23] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 1

[24] K. Mikolajczyk and J. Matas. Improving descriptors for fast tree matching by optimal linear projection. In *ICCV*, 2007. 6

[25] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 2005. 5

[26] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, T. Schaffalitzky, F.and Kadir, and L. Van Gool. A comparison of affine region detectors. *IJCV*, 2005. 5

[27] A. Mikulik, M. Perdoch, O. Chum, and J. Matas. Learning vocabularies over a fine quantization. *IJCV*, 2013. 1, 5

[28] E. Mohedano, K. McGuinness, N. E. O'Connor, A. Salvador, F. Marques, and X. Giro-i Nieto. Bags of local convolutional features for scalable instance search. In *ICMR*, 2016. 5

[29] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006. 1

[30] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han. Large-scale image retrieval with attentive deep local features. In *ICCV*, 2017. 1, 5

[31] M. Perdoch, O. Chum, and J. Matas. Efficient representation of local geometry for large scale object retrieval. In *CVPR*, 2009. 1, 5, 6

[32] F. Perronnin, Y. Liu, J. Sanchez, and H. Poirier. Large-scale image retrieval with compressed Fisher vectors. In *CVPR*, 2010. 5

[33] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007. 1, 2, 3, 4, 5

[34] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, 2008. 1, 4, 5

[35] F. Radenović, J. L. Schönberger, D. Ji, J.-M. Frahm, O. Chum, and J. Matas. From dusk till dawn: Modeling in the dark. In *CVPR*, 2016. 2

[36] F. Radenović, G. Tolias, and O. Chum. CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples. In *ECCV*, 2016. 6, 7

[37] F. Radenović, G. Tolias, and O. Chum. Fine-tuning CNN image retrieval with no human annotation. In *arXiv*, 2017. 1, 4, 5, 6, 7, 8

[38] A. S. Razavian, J. Sullivan, S. Carlsson, and A. Maki. Visual instance retrieval with deep convolutional networks. *ITE Trans. MTA*, 2016. 4, 5, 6, 7

[39] 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. *IJCV*, 2015. 6

[40] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *arXiv*, 2014. 6

[41] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, 2003. 1, 5

[42] R. Tao, E. Gavves, C. G. Snoek, and A. W. Smeulders. Locality in generic instance search from one example. In *CVPR*, 2014. 5

[43] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 2016. 4

[44] G. Tolias and Y. Avrithis. Speeded-up, relaxed spatial matching. In *ICCV*, 2011. 4

[45] G. Tolias, Y. Avrithis, and H. Jégou. To aggregate or not to aggregate: Selective match kernels for image search. In *ICCV*, 2013. 5

[46] G. Tolias, Y. Avrithis, and H. Jégou. Image search with selective match kernels: aggregation across single and multiple images. *IJCV*, 2015. 4, 5

[47] G. Tolias and H. Jégou. Visual query expansion with or without geometry: refining local descriptors by feature aggregation. *Pattern Recognition*, 2014. 1, 5

[48] G. Tolias, R. Sicre, and H. Jégou. Particular object retrieval with integral max-pooling of CNN activations. In *ICLR*, 2016. 1, 5, 6

[49] Y. Verdie, K. Yi, P. Fua, and V. Lepetit. TILDE: a temporally invariant learned detector. In *CVPR*, 2015. 2

[50] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014. 6

[51] J. Yue-Hei Ng, F. Yang, and L. S. Davis. Exploiting local features from deep networks for image retrieval. In *CVPR*, 2015. 5

[52] L. Zheng, S. Wang, Z. Liu, and Q. Tian. Lp-norm idf for large scale image search. In *CVPR*, 2013. 5

[53] L. Zheng, S. Wang, J. Wang, and Q. Tian. Accurate image search with multi-scale contextual evidences. *IJCV*, 2016. 8

[54] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian. Spatial coding for large scale partial-duplicate web image search. In *ACM Multimedia*, 2010. 5