DAME WEB: DynAmic MEan with Whitening Ensemble Binarization for Landmark Retrieval without Human Annotation

Tsun-Yi Yang[1,2] Duy-Kien Nguyen[1,3] Huub Heijnen[1] Vassileios Balntas[1]
[1]Scape Technologies [2]National Taiwan University [3]Tohoku University
shamangary@hotmail.com kien@vision.is.tohoku.ac.jp huub@scape.io vassileios@scape.io

Abstract

In this work, we propose a simple yet effective module called DynAmic MEan (DAME) which allows a neural network to dynamically learn to aggregate feature maps at the pooling stage based on the input image, in order to generate global descriptors suitable for landmark retrieval. In contrast to Generalized Mean (GeM), which uses a predefined and static norm for pooling features into descriptors, we use a dynamic p-norm, with the p value being generated online by the model for each image. In addition, we utilize the introduced dynamic pooling method, to propose a novel feature whitening technique, Whitening Ensemble Binarization (WEB), to discover complementary information through multiple statistical projections. The memory cost of the proposed global binary descriptor is $8\times$ smaller than the state-of-the-art, while exhibiting similar or improved performance. To further demonstrate the power of DAME, we use it with features extracted from a fixed, pretrained classification network, and illustrate that our dynamic p-norm is capable of learning to pool the classification features into global descriptors suitable for retrieval. Finally, by combining DAME with WEB, we achieve state-of-the-art results on challenging large-scale landmark retrieval benchmarks.

1. Introduction

Landmark retrieval is an important component in many computer vision applications, since learning the similarity between images in a global manner is of great importance in filtering out a significant amount of unrelated data, and capable of reducing the computational cost of brute force search methods, especially in city-scale [1, 37, 42] or even global-scale applications [45].

Traditionally, global descriptors were built by aggregating hand-craft features [4, 25] into a global representation [21, 27, 44]. The use of local features, also allowed additional geometric verification by incorporating two view geometry estimation and methods such as RANSAC [10] into the global description pipeline as a post-processing step. In order to improve the quality of the global descriptors, Tolias *et al*. introduced methods, SMK and ASMK, to solely focus on aggregating high quality local features [41]. However, using aggregated information from local patches might not capture all the global information and can, therefore, lead to sub-optimal results.

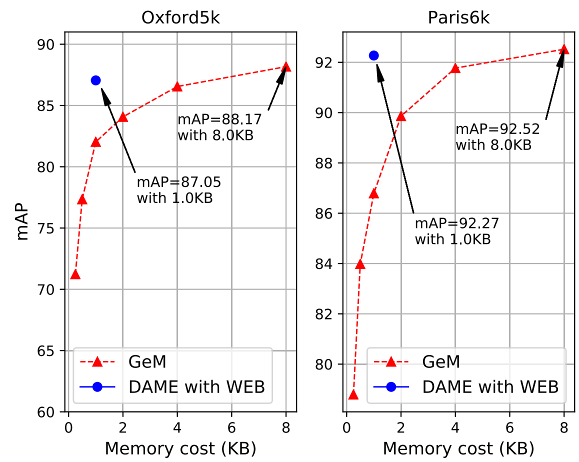


Figure 1. Comparison between performance and memory cost per global descriptor. The baseline output descriptor is a 2048 dimensional float vector from GeM [33] with whitening which costs 8 KB per descriptor. Our binary descriptor with DAME+WEB only costs 1 KB which is $8\times$ smaller while getting the comparable performance. The red nodes represent the real-valued dimensionality reduction results.

With the advent of deep learning, a significant amount of works focused on training end-to-end global descriptors, by utilizing large-scale datasets and metric learning techniques [31, 2, 28, 8, 24]. Based on the database annotations that are utilized for training the global descriptors, we can categorize the relevant methods into two main categories, the landmark based ones [39, 2, 31] and the landmark free ones [33, 32, 1, 46, 47].

For the landmark based methods, annotated landmark names (e.g. Eiffel Tower) or box annotations [2, 28, 39] are required for simultaneous landmark classification in conjunction with retrieval. For example, DELF [28] focuses on learning deep local features by a fully convolutional network, and uses the attention mask for filtering out the unnecessary parts. Teichmann *et al*. [39] adopts an additional landmark detection network (e.g. Fast-RCNN [11]) for improving regional proposal selection, and uses kernel aggregation to generate global description. However, knowing the landmark class annotation or the bounding box annotation is cumbersome and requires significant human labour.

On the other hand, landmark free methods do not require class labels. Some methods focus on unsupervised methods such as creating corresponding pairs of images using Structure-from-Motion, and discovering images that are spatially connected as pairs, and exhibiting distinct view perspectives [33, 32]. Gordo *et al*. [12, 13] utilizes the landmark pair annotation only instead of class label, and used regional sampling technique to capture region candidates for more information. NetVLAD [1] introduces a way to aggregate local features with CNN features by learning the feature cluster centers, and used image pairs from Google's Street view [42] to generate the view pairs used for training.

Nevertheless, the global descriptors still tend to exhibit very high memory consumption for large scale image retrieval problems. Datasets in the orders of billions of floating point descriptors lead to substantial memory requirements for city or global scale applications. Thus, several works focused on binarizing the descriptors [50, 3, 48, 40] in a local or global manner, in order to gain both memory and computational improvements. Yang *et al*. [48] incorporate the hashing techniques into the training of binary descriptors along with the classification. Song *et al*. [38] utilized the idea of regional proposal network into binary descriptors to improve the accuracy.

Recently, an effective method for aggregating features into global descriptors called Generalized Mean (GeM) was proposed by Radenović *et al*. [33]. By using the ℓ_p-norm over the spatial domain of the feature maps as a pooling process, the authors illustrate that the aggregation of important parts of the landmark feature maps could be greatly enhanced without requiring any additional labels or annotations. In addition, they illustrate that increasing the value p of the ℓ_p-norm leads to a smaller region of the image being pooled, while lower p values tend to aggregate across a larger area. GeM adopts a manually assigned global and constant value of p for the ℓ_p-norm in the pooling stage across all images.

However, using a constant norm power for pooling features across every image in the dataset might be too restrictive given the complex nuisances of the real-world data. For example, depending on the conditions, the target landmark could either occupy a significant area of the image or be barely visible between a highly noisy background. In addition, GeM is based on high dimensional floating point descriptors which can be problematic for large-scale system.

To address these issues, in this paper we introduce a novel module called DynAmic MEan (DAME) with a learnable layer for determining the power value p of the ℓ_p-norm used in the pooling operation in contrast to the fixed power operation used in GeM.

Our contributions are: (a) *We propose a learnable $\ell_{p(x)}$-norm module (DAME) for pooling feature maps into global descriptors, that is dynamically computed for each input image x individually* (b) *We use our learnable DAME module, to propose a novel whitening method called Whitening Ensemble Binarization (WEB) for discovering multiple projection possibilities along with descriptor binarization* (c) *By combining DAME with WEB, we construct a binary global descriptor which is $8\times$ smaller than the state-of-the-art and is able to outperform real-valued descriptors* (d) *We observe that features trained for a classification task, can be directly used in a retrieval problem by optimizing channel-wise dynamic $\ell_{p(x)}$-norm pooling only, without the need to re-learn direct feature mapping by fine-tuning*.

2. Method

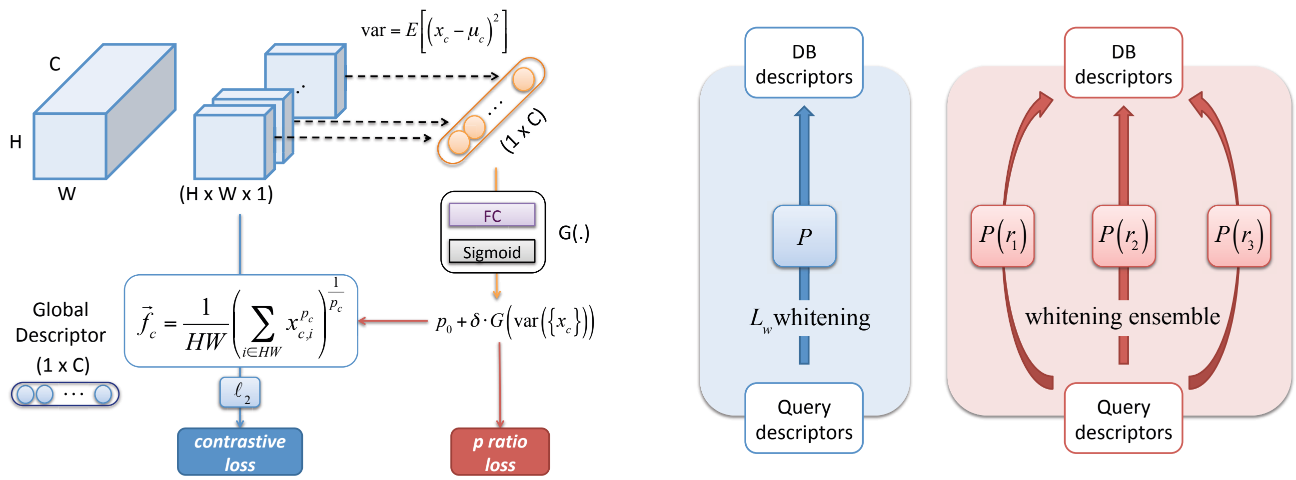
In this section, we firstly introduce the motivation and our goal while our dynamic mean based pooling (DAME) and subsequently are addressed afterward. We discuss a method of utilising the learned ℓ_p-norm to create a novel whitening method for global binary descriptors (WEB).

2.1. Motivation and problem definition

For large-scale retrieval system, people usually compare binarization/hashing methods with compression methods such as Product Quantization (PQ) [20, 23] in terms of compactness. Maintaining PQ-based methods usually requires to construct multiple tables for symmetric or asymmetric distance computation. Sometimes even more complex lookup pipeline [18] or GPU support [22] are required for speedup the searching. However, in practical landmark retrieval system, we usually have GPS-prior and the candidate images can be constrained within a certain range (e.g. $100 \sim 500$ meters). Such setting is especially useful for long-term localization [36, 35]. The bottleneck of storage and query time for a city-scale system can be easily resolved by a compact binary descriptor with fast bitwise operations (e.g. XOR, bitcount). Therefore, we focus on designing a global binary descriptor instead of building complex PQ-based pipeline at different geo-locations. Our method is suitable for efficient storage, query, and simplifies the whole system flow.

2.2. Generalized mean

For an input image $I \in \mathbb{R}^{\mathbb{H}\times\mathbb{W}}$ a global descriptor is generated by pooling features extracted from a fully convo-



(a) DAME overview (b) Whitening ensemble

Figure 2. Overview of (a) the proposed DAME module and the training process, and (b) the whitening comparison between the L_w whitening with GeM using a single projection P and our whitening ensemble with DAME and multiple projections $P(r_k)$.

lutional network, into a vector $f(I) \in \mathbb{R}^D$, which is subsequently compared against a dataset using Euclidean distance and nearest neighbour matching.

In order to enhance the discriminative ability of the global descriptor, Radenović *et al*. proposed Generalized Mean (GeM) [33] pooling, which focuses on suppressing the features that are irrelevant to landmark retrieval and encouraging the aggregation of the relevant ones. By denoting $\{x_c\}$ as the feature maps $(H \times W \times C)$ inside the network with x_c as the $(H \times W)$ matrix corresponding to channel c, GeM can be formulated as

$$\vec{f}_c = \left(\frac{1}{HW} \sum_{i \in HW} x_{c,i}^p \right)^{\frac{1}{p}}. \tag{1}$$

In [33], the value of p can either be assigned manually or learned as a single global parameter during the optimisation of the global descriptors. After training on the large-scale SfM-120k dataset, Radenović *et al*. obtained the learned value which is $p^* = 3$ (or 2.90) and achieved state-of-the-art results by a very simple form of ℓ_p-norm along the channel dimensions. However, since both the manually assigned p or learned p are fixed for each image, it follows that GeM only focuses on a single globally consistent p instead of adaptively changing based on the individual characteristics of the input image.

2.3. Dynamic mean

Considering that higher p value focuses on a very specific part of the image as it was illustrated in [33], while the object of interest size is not fixed across all the images in the database, we hypothesize that using different ℓ_p-norm for different input images is a more reasonable assumption for image retrieval. It is similar to learning the saliency region in many different problems. [49, 15, 7, 6, 43, 16, 17, 9]

Thus, we introduce a new method, DynAmic MEan (DAME), to dynamically adapt the size of the focus region of the feature maps, by allowing p to be a function of the input feature maps $\{x_c\}$. We formulate DAME as

$$\vec{f}_c(\{x_c\}) = \left(\frac{1}{HW} \sum_{i \in HW} x_{c,i}^{p(\{x_c\})} \right)^{\frac{1}{p(\{x_c\})}}, \tag{2}$$

$$p(\{x_c\}) = max\left[(p_0 + \delta \cdot \triangle p), 1\right] \tag{3}$$

with $p_0 = 1$ and a learnable network $G(.)$ which contains one fully connected layer with one output and a Sigmoid activation function to determine the residual change $\triangle p = G(var(\{x_c\}))$. Output p is one dyanmic value for the whole feature map. The overview of our DAME training pipeline is shown in Figure 2 (a). The variances of the feature maps are computed on the spatial dimensions $(H \times W)$, and the inputs of the G network are the variance values vector with dim $(1 \times C)$. The δ is controlled by the relation between the globally learned $p^* = 3$ in GeM and the normal state $p = 1$. It defines the possible range of values p which is defined as $\delta = 2 \cdot (p^* - 1)$ which makes sure the final dynamic value $p \in [3 - 2, 3 + 2]$.

2.4. Training loss

Similar to the previous work, we use contrastive loss for the positive and the negative pairs along with a siamese network architecture [33]:

$$J_{con} = \begin{cases} \frac{1}{2} \left\| f(a) - f(i) \right\|^2 & \text{if } y = 1 \\ \frac{1}{2} \left(\max\left[0, \tau - \left\| f(a) - f(j) \right\| \right] \right)^2 & \text{if } y = 0 \end{cases} \tag{4}$$

with a as the anchor image index and y as the paired ground truth label for (a, i). $y = 1$ represents the positive pair while (a, j) with $y = 0$ is the negative pair. f is the ℓ_2-normalized global descriptor.

Since larger p corresponds to a more local region [33], we hypothesise that small p indicates that most of the features across different positions of the feature maps, are relevant for landmark retrieval, and thus there is no need to suppress the noise. This indicates that such an input image is more likely to represent a landmark. On the other hand, the samples j in negative pair might exhibit higher p values, since the network cannot locate the landmark, and thus needs to focus on a more local region to find discriminative parts. However, the landmark being very clear and easily distinguishable represents and ideal scenario, which is not usually seen in the training data. Thus, by optimizing the ratio of p values, we allow some positive pair samples to have high p due to challenging factors such as noisy background.

Based on this intuition, we develop a new loss called *p ratio loss* (J_{pr}) for encouraging the above behavior.

$$J_{pr} = \frac{E\left[p\left(\{x_c\}\right)\right]_{a,i \text{ with } y=1}}{E\left[p\left(\{x_c\}\right)\right]_{j \text{ with } y=0}} \tag{5}$$

$E\left[.\right]$ represents the mean value over a batch. *Considering we only care about landmark retrieval, the positive pair samples always belong to the same hyper-class which is the building while the sample j in negative pair may or may not be building.* The ratio loss encourages sample j in negative images to have relatively larger p than the positive pair samples, and it prevents us from defining the absolute value of the p in the loss. More details about this, will be addressed in our analysis 3.4 and visualization 3.5 sections.

Our final loss is computed by combining the standard Siamese contrastive loss (Equation 4) and our p-ratio loss (Equation 5) with a hyper-parameter γ as

$$J = J_{con} + \gamma J_{pr} \tag{6}$$

for optimizing the global descriptor for image retrieval.

2.5. Whitening ensemble binarization

Considering the significant improvement that can be obtained by whitening the descriptors in the post-processing stage [33], we examine how the whitening method [5, 26] can be improved by generating projection variants by utilizing dynamic p. The comparison is shown in Figure 2 (b).

The whitening projection P is defined as $P = \left[C_S^{-\frac{1}{2}} eig\left(C_S^{-\frac{1}{2}} C_D C_S^{-\frac{1}{2}}\right)\right]^T$ and the whitened descriptor as $f_w\left(i\right) = P\left(f\left(i\right) - \mu\right)$ with $\mu = \frac{1}{N}\sum_i f\left(i\right)$. Note that C_S is the intraclass covariance matrix of the positive pair while C_D is the interclass covariance matrix. The eig term is responsible for the feature space rotation, and

	wht.	extra	KB	Oxford5k	Paris6k
Real-valued descriptor (2048 dim)					
GeM	-	-	8	81.18	87.82
	L_w			88.17	92.6
DAME	-			80.79	87.72
	L_w			**88.24**	93.00
	$L_{w,r}$			87.88	**93.42**
Dimension reduction					
GeM	L_w	PCA	2	82.03	86.79
	L_w		1	77.34	83.98
Binary descriptor					
SSDH	-	SS	0.06	63.79	83.87
gDRH	-	-	0.13	74.8	77.3
		-	0.5	78.3	81.5
DRH all	-	QE	0.5	85.1	84.9
DAME	L_{b2}	-	0.5	86.21	91.79
	L_{b4}	-	1	**87.05**	**92.27**

Table 1. Real-valued and binary descriptors comparison. The bit numbers are transformed into kilobytes for clear comparison. For example, $(2048 \cdot 4)$ bits $= 1$ KB. **Red:** The new state-of-the-art results for the binary ones. Note that our method doesn't use any extra search such as spatial search (SS) or query expansion (QE).

dimension reduction can be achieved by choosing the D largest eigenvalues for the projection, with the descriptors being subsequently ℓ_2-normalized. Radenović *et al.* refer to this projection as L_w or supervised whitening. The score (inverse of distance) between images with indices a and i is computed with function $s\left(a, i\right) = f_w\left(a\right)^T f_w\left(i\right) = \left[Pf\left(a\right)\right]^T\left[Pf\left(i\right)\right]$, with μ being ignored for brevity.

Since our method is optimized based on the *p-ratio loss*, we can exploit this for building a novel whitening method based on this. For each positive pair of images (a, i) we compute their cumulative p value, $p_{ai} = p\left(a\right) + p\left(i\right)$. Subsequently, we sort all the available pairs according to their p_{ai} values, and we remove pairs with large p_{ai} which are considered to be more noisy than the ones with smaller p_{ai}. Given a percentage ratio r , a new intra-class $C_{S,r}$ covariance matrix can be computed, which is built using only items at the top r % of the ranked pairs, according to their p_{ai}. Using $C_{S,r}$, we can generate a new projection matrix, $P\left(r\right) = \left[C_{S,r}^{-\frac{1}{2}} eig\left(C_{S,r}^{-\frac{1}{2}} C_D C_{S,r}^{-\frac{1}{2}}\right)\right]^T$.

By using different ratio factors r_k, several whitening projection matrices can be generated, and several descriptors from the different $P\left(r_k\right)$ projections can be computed. However, concatenating several high-dimensional floating point descriptors, would result in impractical memory and computational requirements. Inspired by ASV [51] and HE [19], we binarize the descriptors before concatenation, using a median based binarization function $B\left(.\right)$. That is, every element which is smaller than median value becomes 1, and -1 otherwise. Given n ratios r_k with $k \in [1, n]$

Method	white.	K	ROxf5k(M)		ROxf5k(M)+1M		RPar6k(M)		RPar6k(M)+1M		ROxf5k(H)		ROxf5k(H)+1M		RPar6k(H)		RPar6k(H)+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
Backbone: VGG16 (real-valued dim: 512)																		
NetVLAD	-	2	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
MAC	V		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
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
GeM	L_w		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
Backbone: Resnet101 (real-valued dim: 2048)																		
MAC	-	8	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-MAC	-		60.09	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
GeM	L_w		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**
DAME	-	8	55.95	76.24	33.75	58.00	69.89	96.43	40.61	92.29	28.27	39.29	12.01	17.86	44.39	81.86	14.42	53.43
	L_w		**65.32**	85.00	44.74	70.14	77.13	98.43	50.52	94.57	**40.35**	56.29	22.82	35.57	56.04	88.00	21.95	69.00
	L_w^{90}		64.94	84.71	44.37	70.00	77.61	98.57	51.44	94.57	40.24	56.00	22.42	35.71	56.76	88.86	22.83	71.00
	L_w^{80}		64.80	84.29	44.26	70.00	77.61	98.57	51.42	94.57	40.19	56.00	22.34	35.57	56.70	88.86	22.83	71.00
	L_w^{50}		64.74	84.43	43.37	68.81	77.38	98.57	51.16	94.71	39.98	**56.57**	21.13	33.71	56.08	88.57	22.56	70.43
Backbone: Resnet101 (binary descriptor)																		
DAME	L_{b2}	**0.5**	61.57	85.00	42.54	68.38	74.64	98.57	50.74	94.71	36.55	53.86	21.83	35.00	52.74	89.29	21.94	69.71
	L_{b4}	**1**	62.93	**85.86**	44.33	69.43	75.62	**98.71**	51.53	95.00	38.17	55.29	**22.58**	**36.14**	54.30	**89.57**	22.84	71.71

Table 2. Overall comparison of the landmark free methods. **Red:** The new state-of-the-art results for the landmark free ones. Landmark free methods do not use human annotation at all while landmark based methods (DELF) [28] need annotated landmark class dataset [2] for the training. Our methods are the most compact one ($0.5 \sim 1$ KB) and outperform most of the landmark free real-valued methods under the mP@10 metric.

we define the final concatenated binary descriptor $f_b = \text{cat}\left[B\left(f_{w,r_1}\right), B\left(f_{w,r_2}\right), \ldots, B\left(f_{w,r_n}\right)\right]$. The similarity between two binary descriptors is computed using the Hamming inner product [41]. We call the whole procedure as Whitening Ensemble Binarization (WEB).

3. Experimental Evaluation

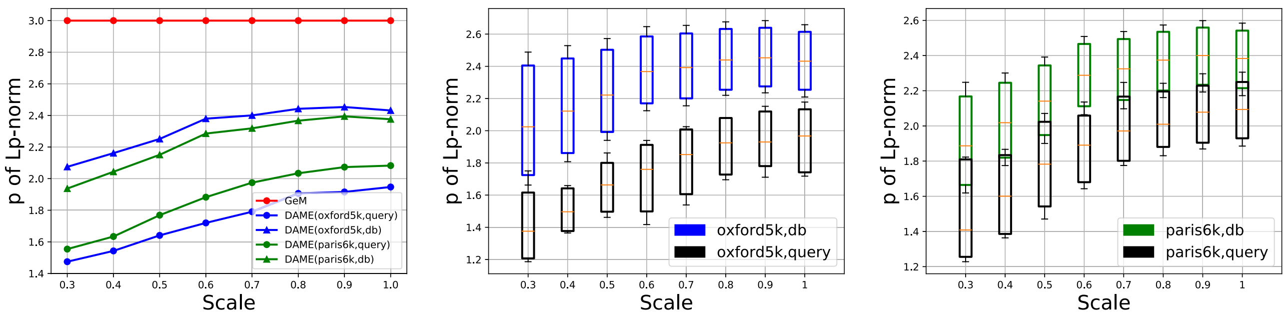
In this section, we describe the parameters and protocols used in our experiments, and we analyze the proposed DAME & WEB modules under different scenarios.

3.1. Setting and details

For training and validation, we follow the same setting as Radenović *et al*. [33]. Training is done on SfM-120K dataset which contains 7.4 million images collected from Flickr with popular places around the world. Exponential decay schedule with 30 epochs and Adam optimizer ($lr = 5 \times 10^{-7}$) are used. The batchsize is 5 and 5 negative samples are chosen by the online hard negative selection. The training of DAME is based on a two stage process. First, we train the backbone model (e.g. Resnet101 [14]) by GeM pipeline for finding the optimal static p^*. Secondly, we fix the weights on the backbone and replace the GeM module with DAME module with $\delta = 2\left(p^* - 1\right)$, and train the DAME individually with $\gamma = 1$. The margin τ is set to 0.85. We also examined the possibility of training the backbone network along with the DAME module. We observed that simultaneously learning the features and the dynamic norm operation, resulted in unstable learning in both parts. However, since our objective is to explore the effect of the dynamic pooling on retrieval, investigation of end-to-end training is out of the scope of this work. The resolution during training is 362×362, and the testing resolution is 1024×1024. Note that using an unbounded activation such as ReLU in $G\left(.\right)$ is an intuitive choice for learning the dynamic p (Section 2.3). However, we found that using the unbounded ReLU or setting δ too high led to unstable training with the loss going into nan. Therefore using a Sigmoid function with a proper δ is a more suitable choice.

The results are tested on Oxford5k [29] and Paris6k datasets [30] which consist of 55 query images. Note that post whitening on SfM-120K dataset and multi-scale results fusion by GeM are used by default unless otherwise indicated. Mean-Average-Precision (mAP) is adopted as our evaluation metric, similarly to previous works. Both datasets have recently been revisited and corrected since the original annotation was not consistently of good quality. This revisit led to the ROxford5k and RParis6k datasets, which are characterized by expansion of the number of queries to 70, new evaluation protocols [31] and removal of false positives and false negatives. In addition, three different types of positive samples are included in the dataset (Easy, Unclear, Hard), while the images with the Junk label [29] are ignored. We choose the **Medium** and **Hard** cases in our evaluation similarly to [39].

With respect to the whitening configurations, we define L_w^{90} as the whitening done using $r = 0.9$, and real de-



(a) Mean p comparison (b) Oxford5k dataset (c) Paris6k dataset

Figure 3. Dynamic p analysis with $\gamma = 0$. Training is fully based on contrastive loss for the results presented in this figure.

scriptors *i.e*. without binarization. For the Whitening Ensemble Binarization (Section 2.5), we examining two possible configurations, L_{b2} with $\{r_k\} = \{1, 0.9\}$ and L_{b4} with $\{r_k\} = \{1, 0.9, 0.8, 0.5\}$.

3.2. Competing methods

Here we briefly introduce the baselines used for the experiments below. **Generalized Mean (GeM)**[33] is the most related method, and it uses a single p with simple ℓ_p-norm over channels as pooling to suppress the false spatial regions. **MAC** [32] constructs the global descriptor by global max-pooling. **R-MAC** [12] uniformly samples the windows for the regional max-pooling. **NetVLAD** [1] introduced a trainable aggregation module by learning the feature centers. annotations for fine-tuning. **SSDH** [48] optimizes a classification and retrieval binary code jointly to unify classification and retrieval problem together. **DRH** [38] is based on an end-to-end deep neural network which is jointly optimized for object proposal, feature extraction, and hashing.

3.3. Landmark retrieval comparison

In Table 1, we can observe that the real-valued DAME slightly improves GeM when using the standard Oxf5k and Par5k datasets. However, the significant result is that DAME with WEB (L_{b2}, L_{b4}) binary descriptor outperforms others binary descriptors by a huge margin. The state-of-the-art DRH pipline (DRH all) includes three steps, Global Region Hashing (gDRH) search, and Local Region Hashing (lDRH) for the re-ranking. Finally, Query Expansions (QE) are done based on regional hashing. SSDH also performs additional spatial search for the verification. Note that we do not apply any additional search techniques such as query expansion, re-ranking, or any geometric verfication. Not only our method is $8\times$ smaller than the real-valued GeM, our performance is comparable to the strongest real-valued one and outperforms the best binary ones.

It is interesting to point out that while our training process does not include any binarization, our binary descriptor still outperforms other state-of-the-art methods, which

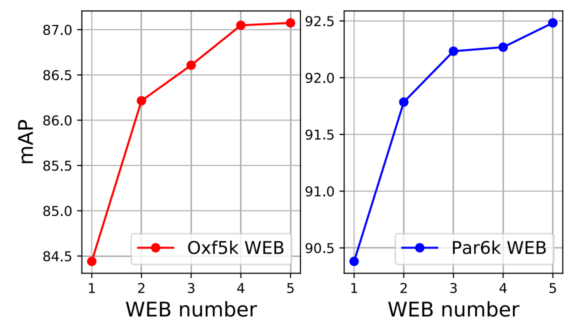


Figure 4. Analysis over number of variants inside the whitening ensemble. The performance increases dramatically when the variants number increases.

include a binarization step during training. We attribute this property to thresholding the descriptors after the whitening process, leading to preservation of the improvement from the whitening stage.

It is clear that using additional annotations will benefit the retrieval problem. However, it is surprising to see that our DAME with WEB binary descriptors outperform most of the landmark free real-valued ones under the standard $mP@10$ evaluation metric. This indicates that our method is more robust than the others when considering the top ranked retrieved results, something that is important for high-accuracy scenarios. We can also observe that the different ratio r of whitening data could be beneficial, such in the hard case in RParis6k(H). The $mP@10$ improves from 88.00 to 88.86 when $r = 0.9$. Our methods achieve best results in eight cases, despite the minimal memory requirements.

3.4. Analysis

To verify our assumption on the relation between the image samples and p (Section 2.4), we perform a quantitative analysis. Our goal is to observe the distributions of dynamic p values over the dataset and query images, and to analyse the differences between them and a global value of $p = 3$ which GeM uses. In addition, we examine the effect of image size on the results, by downscaling the images several times from scale 1 (1024×1024) to the minimum scale of 0.3 times the original.

In Figure 3(a), the mean values of p across all the dataset images are shown. The red line corresponds to GeM which uses a static p across all images. The blue lines represent the database and query images from Oxford5k, and the green lines from Paris6k. An interesting observation is that p is smaller in the query set than the database. This can be explained by the fact that in both Oxford5k and Paris6k, the query image is always bounded by a manual bounding box which correctly eliminates the background noise.

In Figure 3 (b)(c), we plot the distribution of p values for different scales. The boxes extend from the lower to upper quartile values of the samples, and the orange lines are the median values. Paris6k clearly has closer distribution between the query and the retrieved database, and it also has higher mAP than Oxford5k in our experiments. This suggests that there is a correlation between high p values and high image noise. On the other hand, the wide uncertainty ranges in both query and database images also suggest that only a relative relation can be observed. Note that for all experiments in Figure 3, we use contrastive loss only (*i.e.* $\gamma = 0$) in order to show the nature of our DAME module.

Whitening ensemble analysis. After choosing different variant projections by filtering out a subset positive pairs as indicated in Section 2.5, descriptors are combined with median binarization and concatenation. In Figure 4, we can observe that the performance improves dramatically when the number of variants inside the ensemble increases. Considering that the required memory grows linearly with the total number of ratios, we examine configurations up to 5 projections in the ensemble, to represent a reasonable compromise between memory requirements and performance.

Filtering choices. To verify that higher p values are more likely to be associated with noisier samples, we perform a simple experiment by building the whitening projection matrix using two strategies. In the first case, the subset of images with high p, defined by the ratio r, are filtered before the computation of the whitening projection $P(r)$ (Section 2.5), while in the second case, the subset of images with low p are filtered. Observing the results in Figure 5, it is clear that building the whitening projection matrix using the subset of images with low p values (filtering high p values), always outperforms a whitening method built with images with high p. Note that for this experiment, we plot the results using dimensions corresponding to the largest 128 eigenvalues, since they are able to capture the majority of the information available.

3.5. Visualization

In Figure 6, we crop three parts of Big Ben to examine the difference between the DAME and GeM. For the first 2 rows that include discriminative parts such as the roof and the clock, our method generates relative low p such as $p(x) = 1.76$ and $p(x) = 1.51$. Contrary, since GeM uses a static value ($p = 3$) it misses some areas which contain useful information. In the last row of Figure 6, the patch contains more vague information and leads our method to generate higher p (2.34), to force the network to focus on more specific local areas.

In Figure 7, sample images from Oxford5k and Paris6k

Backbone	Method	white.	Oxford5k	Paris6k
ImageNet Res50 (fixed)	ℓ_3-norm	-	47.15	67.33
	DAME	-	58.04	79.02
		L_w	**75.02**	**90.01**
		L_{b4}	71.07	87.06

Table 3. Fixed ImageNet classification backbone network without fine-tuning it, and train DAME module with channelwise dynamic p only for mAP comparison.

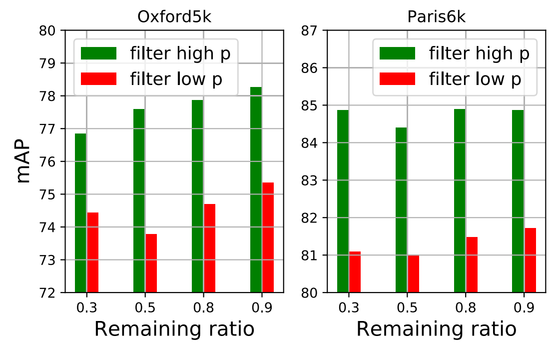


Figure 5. Building the whitening projection matrix using subsets of images with high and low p values.

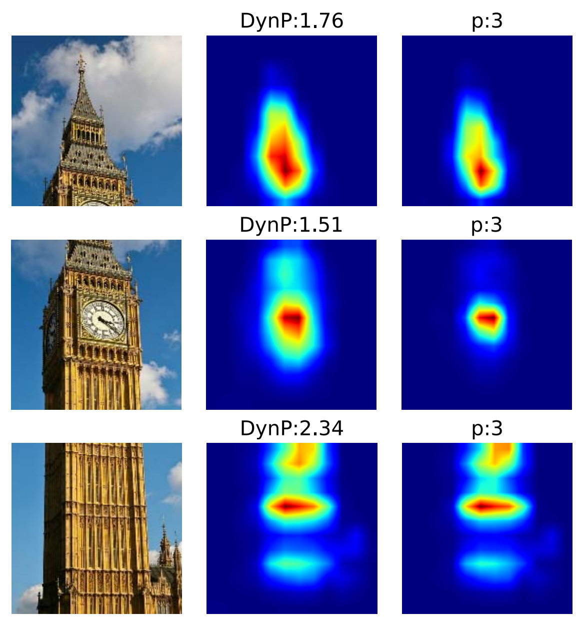


Figure 6. Visualisation of different parts of Big Ben, together with the heatmaps extracted from the summation over channels of the feature maps after DAME (column 2) or GeM (column 3).

Figure 7. Visualization of the input image and the corresponding dynamic p ($DynP$).

are shown with their corresponding dynamic p generated by DAME. For the database images, we can observe that large p indicates either a negative sample, or an image where the landmark occupies a small area inside a noisy background. The query images in both datasets were manually cropped, thus a cleaner view of the landmark can be usually observed, leading to generally lower p values. However, we also discover some cases without noisy background which exhibit high p values (cases in Figure 7 annotated by red labels). This might be due to the fact that the images are not distinguishable enough due to repetitive structures (*i.e.* windows and columns) or the angle of view is oblique. Thus, the most discriminative features can only be found by focusing on sufficiently small regions.

3.6. Feature generalization

In order to show the effectiveness of our DAME module, we perform an experiment of optimizing a channelwise version of DAME which we build by replacing the output dimension of the fully connected layer from 1 to C and generating different $\vec{p}_c$ values for each channel c.

No learnable mapping. Normally, to fine-tune a network for a new task, additional fully connected or convolutional layers are added on networks pre-trained on large scale classification datasets (e.g. ImageNet [34]),

To illustrate the power of DAME, we fix the classification backbone networks as feature extractors, and we only train the power operation with $p\left(x\right)$ from network $G\left(.\right)$. Using the fixed extracted features x, we apply our dynamic $\ell_{p(x)}$-norm as

$$\vec{f}_c = \frac{1}{HW}\left(\left|x_{c,1}\right|^p + \left|x_{c,2}\right|^p + ... + \left|x_{c,HW}\right|^p\right)^{\frac{1}{p}}$$

Note that the global descriptor is still based on the static classification based features x without any complex remapping of the features to a new task.

We show the results in Table 3 for networks pre-trained on the popular object classification dataset, ImageNet [34]. We can observe that there is a significant improvement over using a ℓ_3-norm that aggregates the original classification features. To the best of our knowledge, it is the first time to observe that a high retrieval performance can be achieved with fixed pre-trained features from a classification network, by solely using a learnable dynamic $\ell_{p(x)}$-norm in the pooling stage.

4. Conclusion

In this paper, we introduce a new way to dynamically determine $\ell_{p(x)}$-norm in the pooling stage based on the input image. Based on a DynAmic MEan based pooling (DAME), a whitening ensemble method can be built by filtering out different ratios of the available data according to their p values. Our method is able to achieve comparable or outperform the state-of-the-art real-valued global descriptors while costing $8\times$ less memory. Finally, we show that by using DAME, feature generalization can be achieved between classification and retrieval tasks by learning how to aggregate fixed feature maps.

References

[1] R. Arandjelovic, P. Gronát, A. Torii, T. Pajdla, and J. Sivic. Netvlad: CNN architecture for weakly supervised place recognition. In *CVPR*, 2016. 1, 1, 3.2

[2] A. Babenko, A. Slesarev, A. Chigorin, and V. S. Lempitsky. Neural codes for image retrieval. In *ECCV*, 2014. 1, 2

[3] V. Balntas, L. Tang, and K. Mikolajczyk. Bold-binary online learned descriptor for efficient image matching. In *CVPR*, 2015. 1

[4] H. Bay, A. Ess, T. Tuytelaars, and L. J. V. Gool. Speeded-up robust features (SURF). *CVIU*, 2008. 1

[5] H. Cai, K. Mikolajczyk, and J. Matas. Learning linear discriminant projections for dimensionality reduction of image descriptors. *PAMI*, 2011. 2.5

[6] Y.-C. Chen and W. H. Hsu. Saliency aware: Weakly supervised object localization. In *ICASSP*, 2019. 2.3

[7] Y.-C. Chen, P.-H. Huang, L.-Y. Yu, J.-B. Huang, M.-H. Yang, and Y.-Y. Lin. Deep semantic matching with foreground detection and cycle-consistency. In *ACCV*, 2018. 2.3

[8] Y.-C. Chen, Y.-J. Li, X. Du, and Y.-C. F. Wang. Learning resolution-invariant deep representations for person re-identification. In *AAAI*, 2019. 1

[9] Y.-C. Chen, Y.-Y. Lin, M.-H. Yang, and J.-B. Huang. Show, match and segment: Joint learning of semantic matching and object co-segmentation. *arXiv*, 2019. 2.3

[10] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 1981. 1

[11] R. B. Girshick. Fast R-CNN. In *ICCV*, 2015. 1

[12] A. Gordo, J. Almaz, and C. V. May. End-to-end Learning of Deep Visual Representations for Image Retrieval. *IJCV*, 2017. 1, 3.2

[13] A. Gordo, J. Almaz, J. Revaud, and D. Larlus. Deep Image Retrieval : Learning global representations for image search. In *ECCV*, 2016. 1

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3.1

[15] K.-J. Hsu, Y.-Y. Lin, and Y.-Y. Chuang. Deepco3: Deep instance co-segmentation by co-peak search and co-saliency detection. In *CVPR*, 2019. 2.3

[16] K.-J. Hsu, C.-C. Tsai, Y.-Y. Lin, X. Qian, and Y.-Y. Chuang. Unsupervised cnn-based co-saliency detection with graphical optimization. In *ECCV*, 2018. 2.3

[17] Y.-T. Hu, J.-B. Huang, and A. G. Schwing. Videomatch: Matching based video object segmentation. In *ECCV*, 2018. 2.3

[18] H. Jain, J. Zepeda, P. Pérez, and R. Gribonval. Learning a complete image indexing pipeline. In *CVPR*, 2018. 2.1

[19] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008. 2.5

[20] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *PAMI*, 2010. 2.1

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

[22] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus. *TBD*, 2019. 2.1

[23] Y. Kalantidis and Y. Avrithis. Locally optimized product quantization for approximate nearest neighbor search. In *CVPR*, 2014. 2.1

[24] Y.-J. Li, Y.-C. Chen, Y.-Y. Lin, X. Du, and Y.-C. F. Wang. Recover and identify: A generative dual model for cross-resolution person re-identification. 2019. 1

[25] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 1

[26] K. Mikolajczyk and J. Matas. Improving descriptors for fast tree matching by optimal linear projection. In *ICCV*, 2007. 2.5

[27] J. Y. Ng, F. Yang, and L. S. Davis. Exploiting local features from deep networks for image retrieval. In *CVPRw*, pages 53–61, 2015. 1

[28] H. Noh, A. Araujo, J. Sim, and T. Weyand. Large-Scale Image Retrieval with Attentive Deep Local Features. In *ICCV*, 2017. 1, 2

[29] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007. 3.1

[30] 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. 3.1

[31] F. Radenović, A. Iscen, G. Tolias, Y. Avrithis, and O. Chum. Revisiting Oxford and Paris : Large-Scale Image Retrieval Benchmarking. In *CVPR*, 2018. 1, 3.1

[32] F. Radenović, G. Tolias, and O. Chum. CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples. In *ECCV*, 2016. 1, 3.2

[33] F. Radenović, G. Tolias, and O. Chum. Fine-tuning CNN Image Retrieval with No Human Annotation. *PAMI*, 2018. 1, 2.2, 2.2, 2.3, 2.4, 2.4, 2.5, 3.1, 3.2

[34] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 3.6

[35] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. 2019. 2.1

[36] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *CVPR*, 2018. 2.1

[37] T. Sattler, A. Torii, J. Sivic, M. Pollefeys, H. Taira, M. Okutomi, and T. Pajdla. Are large-scale 3d models really necessary for accurate visual localization? In *CVPR*, 2017. 1

[38] J. Song, T. He, L. Gao, X. Xu, and H. T. Shen. Deep Region Hashing for Generic Instance Search from Images. In *AAAI*, 2018. 1, 3.2

[39] M. Teichmann and J. Sim. Detect-to-Retrieve: Efficient Regional Aggregation for Image Search. In *CVPR*, 2019. 1, 3.1

[40] Y. Tian, B. Fan, and F. Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *CVPR*, 2017. 1

[41] G. Tolias, Y. Avrithis, H. Jégou, G. Tolias, Y. Avrithis, H. Jégou, G. Tolias, and Y. Avrithis. Image search with selective match kernels : aggregation across single and multiple images. *IJCV*, 2015. 1, 2.5

[42] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla. 24/7 place recognition by view synthesis. In *CVPR*, 2015. 1, 1

[43] C.-C. Tsai, W. Li, K.-J. Hsu, X. Qian, and Y.-Y. Lin. Image co-saliency detection and co-segmentation via progressive joint optimization. *TIP*, 2018. 2.3

[44] T. Uricchio, M. Bertini, L. Seidenari, and A. D. Bimbo. Fisher encoded convolutional bag-of-windows for efficient image retrieval and social image tagging. In *ICCV*, 2015. 1

[45] T. Weyand, I. Kostrikov, and J. Philbin. Planet - photo geolocation with convolutional neural networks. In *ECCV*, 2016. 1

[46] J. Xu, C. Shi, C. Qi, C. Wang, and B. Xiao. Unsupervised Part-based Weighting Aggregation of Deep Convolutional Features for Image Retrieval. In *AAAI*, 2018. 1

[47] J. Xu, C. Wang, C. Qi, C. Shi, and B. Xiao. Unsupervised Semantic-based Aggregation of Deep Convolutional Features. *TIP*, 2019. 1

[48] H.-f. Yang, K. Lin, and C.-s. Chen. Supervised Learning of Semantics-Preserving Hash via Deep Convolutional Neural Networks. *PAMI*, 2018. 1, 3.2

[49] T.-Y. Yang, Y.-T. Chen, Y.-Y. Lin, and Y.-Y. Chuang. Fsa-net: Learning fine-grained structure aggregation for head pose estimation from a single image. In *CVPR*, 2019. 2.3

[50] T.-Y. Yang, J.-H. Hsu, Y.-Y. Lin, and Y.-Y. Chuang. Deepcd: Learning deep complementary descriptors for patch representations. In *ICCV*, 2017. 1

[51] T.-Y. Yang, Y.-Y. Lin, and Y.-Y. Chuang. Accumulated stability voting: A robust descriptor from descriptors of multiple scales. In *CVPR*, 2016. 2.5