GATED SQUARE-ROOT POOLING FOR IMAGE INSTANCE RETRIEVAL

Ziqian Chen[1], *Jie Lin*[*2], *Vijay Chandrasekhar*[2,3], *Ling-Yu Duan*[1]

Peking University, China[1] I^{2}R, Singapore[2] NTU, Singapore[3]

ABSTRACT

Recently Convolutional Neural Networks (CNNs) have achieved great success in different fields including image instance retrieval. However traditional global pooling approaches fail to capture all possible discriminative information of CNN activations and treat activations over channels equally regardless of the different importance between channels. In this work, we focus on the mentioned problem of global feature pooling over CNN activations for image instance retrieval. We make two contributions. First, we introduce a channel-wise SQUare-root (SQU) pooling (2-norm) approach,which makes better use of information over activation maps and is superior to Average (1-norm) and Max pooling (infinity norm), in the context of instance retrieval. Second, we further improve SQU by learning a gating function that weights the contributions of different channels, in an end-to-end manner. Extensive experiments on 6 benchmark datasets show that the proposed strategies achieve considerable improvements over state-of-the-art.

***Index Terms*—** Instance Retrieval, CNN, Square-root pooling, Learning to gate.

1. INTRODUCTION

Image instance retrieval is the discovery of images from a database representing the same object or scene as the one depicted in a query image. In this field, descriptors play a significant role in its performance. Previous works for image instance retrieval are mainly image-level descriptors aggregated from handcrafted local features, such as the well-known SIFT [1], VLAD [2] and Fisher vectors [3]. These approaches can be further improved by numerous strategies [4, 5, 6, 7] Motivated by the remarkable success of CNNs [8, 9, 10] in image classification [11], CNN-based descriptors [12, 13, 14, 15, 16, 17, 18] have been progressively replacing handcrafted descriptors [6] as state-of-the-art for image instance retrieval.

Generally speaking, deeply learned descriptors in image instance retrieval are aggregated by pooling over activation maps extracted from intermediate layers [19, 20, 12, 13, 14].Initial study [20] proposed to use representations extracted from fully connected layer of CNN.More compact descriptors [12] can be derived by performing either global max (MAC [14]) or average pooling (e.g. SPoC [12]) over activation maps output by convolutional/pooling layers. However, global max pooling focus on the max activation and global average pooling regards low and high response the same and thus these approaches may ignore potentially discriminative information in a single activation map. Further improvements are obtained by spatial and channel-wise weighting of activation maps [13], making us noticing that CNN-based features over different channels may devote different contributions to the performance. Inspired by R-CNN [21] for object detection, Tolias et al. [14] proposed ROI-based pooling on activation maps, Regional Maximum Activation of Convolutions (R-MAC), which significantly improves global pooling approaches. However, R-MAC still follows the idea of Maximum pooling, which will discard some potentially necessary information and also treat different channels equally.

In recent works [20, 15, 16, 17, 18], image classification pre-trained CNNs are repurposed for instance retrieval, by fine-tuning them with ranking loss functions such as contrastive loss [22] and triplet loss [23]. Results show that fine-tuned models outperform pre-trained ones by a large margin, when training and test datasets belong to similar domains.

For the better usage of potentially discriminative information in a CNN feature map, we introduce Square-root pooling (SQU) as an alternative operator to regular global average and max pooling, in the context of instance retrieval.SQU improves the quality of global descriptors with either global average or max pooling, which is our first contribution. Our second contribution is we propose to further boost the performance of SQU by learning a channel-wise gating function to weight the importance of different channels, termed GatedSQU. In particular, to adapt the gating function for instance retrieval, we integrate GatedSQU layer with ranking loss for learning the gating parameters in an end-to-end manner. We perform systematical evaluations for both SQU and GatedSQU on 6 benchmark datasets. Results demonstrate the effectiveness of SQU and GatedSQU over state-of-the-art.

2. METHOD

2.1. Aggregated CNN Descriptors for Instance Retrieval

Consider an image X as input to CNN, we describe the image with activation maps extracted from intermediate layer, denoted as $X = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_C\}$, where $\boldsymbol{x}_c$ represents a activation

[*]The first two authors contributed equally.

map of width W and height H, C is the number of channels. Global feature pooling is performed to convert each activation map with size $N = W \times H$ into a single value,

$$f_\alpha(\boldsymbol{x}_c) = (\frac{1}{N}\sum_{i=1}^{N}(x_{c,i})^\alpha)^{\frac{1}{\alpha}}. \quad (1)$$

$f_\alpha(\boldsymbol{x}_c)$ for all channels are concatenated to form a C-dimensional CNN descriptor. Eq. 1 encompasses recent works on global pooling [12, 13, 14] for image retrieval. For instance, $\alpha = 1$ represents 1-norm average pooling for SPoC [12], while $\alpha = \infty$ denotes max pooling for MAC [14].

The aggregated global CNN descriptors can be further improved by post-processing techniques such as PCA whitening [12, 13, 14]. Specifically, the descriptors are firstly L2 normalized, followed by PCA projection and whitening with a pre-trained PCA matrix. Finally, the whitened vectors are L2 normalized and compared with inner product.

2.2. SQU: SQUare-root pooling

Theoretical analysis [24] has shown that max (or average) pooling may perform better than average (or max), depending on dataset as well as features. This motivates us to find an intermediate α-norm between ∞ and 1 that is superior to either of them. A natural choice could be Square-root pooling (SQU) with $\alpha = 2$, i.e. $f_\alpha^{SQU}(\boldsymbol{x}_c) = (\frac{1}{N}\sum_{i=1}^{N}{x_{c,i}}^2)^{\frac{1}{2}}$. In fact, SQU suppressed other pooling operators in image classification with Bag-of-Words built on SIFT [24]. Here, we explore the feasibility of SQU in the context of image instance retrieval with CNN features. Retrieval experiments in Section 3.3 highlight that SQU outperforms MAC and SPoC by a large margin.

2.3. GatedSQU: learning to gate SQU

Recently, Kalantidis et al. [13] proposed CroW to improve SPoC by spatial and channel-wise weighted average pooling. However, we observe that CroW is limited to SPoC, there is a significant drop in performance if apply CroW to other pooling operations like MAC. Instead of the heuristic approach like CroW, we propose to learn a data-driven gating function that weights the contributions of channels. Moreover, the gating mechanism can benefit from domain-specific knowledge for the purpose of image instance retrieval.

To this end, we design an end-to-end pipeline to learn a desirable gating function with deep neural networks. First, we crop the CNN to the last pooling layer (e.g. pool5), then append a new GatedSQU layer which performs pooling over activation maps output by pool5, finally followed by a triplet loss tailored for the retrieval task. In particular, the GatedSQU layer applies a gating function on SQU.

GatedSQU layer. There are many ways to learn a gating function [25]. In this work, we design a simple channel-wise gating function on SQU, which is defined as

$$f_\alpha^{GatedSQU}(\boldsymbol{x}_c) = \sigma(s \cdot w_c) f_\alpha^{SQU}(\boldsymbol{x}_c), \quad (2)$$

where $\boldsymbol{w} = \{w_1, ...w_C\}$ denote channel-wise weights, $\sigma(.)$ is the sigmoid function. s is a scale constant to control the speed of driving $\sigma(.)$ towards 0 or 1 (i.e. a logic gate over channels). Eq. 2 is differentiable, thus, the gating parameters $\boldsymbol{w}$ can be optimized via stochastic gradient descent. More precisely, we compute the gradient of Eq. 2 w.r.t gating parameter w_c and activation map element $x_{c,i}$ as follows

$$\frac{\partial_{f_\alpha^{GatedSQU}}}{\partial_{w_c}} = \sigma(s \cdot w_c)(1 - \sigma(s \cdot w_c)) f_\alpha^{SQU}(\boldsymbol{x}_c), \quad (3)$$

$$\frac{\partial_{f_\alpha^{GatedSQU}}}{\partial_{x_{c,i}}} = \frac{x_{c,i}\sigma(s \cdot w_c)}{N f_\alpha^{SQU}(\boldsymbol{x}_c)} \quad (4)$$

Finally, we apply l_2 normalization after GatedSQU, resulting in a C-dimensional normalized descriptor.

Learning with triplet Loss. Following existing works [15, 17, 18], we choose the triplet loss which is widely used for the retrieval task. A triplet (X^q, X^+, X^-) contains a query image X^q, a positive image X^+ and a negative image X^-. Query X^q is more similar to positive image X^+ than to negative image X^-. Image similarity is measured by l_2 normalized GatedSQU descriptors. Thus, the triplet needs to meet the condition that $k(X^q, X^+) > k(X^q, X^-)$, where $k(\cdot, \cdot)$ denotes the similarity of a pair. Accordingly, we define the triplet loss as $L_{q,+,-} = max\{0, m + k(X^q, X^-) - k(X^q, X^+)\}$, where m is a positive margin constant.

3. EXPERIMENTS

3.1. Datasets and Metrics

We evaluate our method on six benchmark datasets. INRIA **Holidays** [26] dataset is composed of 1491 scene-centric images, 500 of them are queries. Following [20], we use the rotated version of Holidays, where all images are with upright orientation. **Oxford5k** [27] and **Paris6k** [28] are buildings datasets respectively consisting of 5062 and 6412 images. For both datasets, there are 55 queries composed of 11 landmarks, each represented by 5 queries. To evaluate the performance at large scale, we additionally combine 100k Flickr images [27] with Oxford5k and Paris6k respectively, referred to as **Oxford105k** and **Paris106k** from here on. The University of Kentucky Benchmark (**UKBench**) [29] consists of 10200 VGA size images, organized into 2550 groups of common objects, each object represented by 4 images. All 10200 images are serving as queries.

Following the standard protocols, for Holidays, Oxford5k and Paris6k, retrieval performances are measured by mean Average Precision (mAP). For UKBench, we report the average number of true positives within the top 4 returned images (4×Recall@4).

Table 1. Comparison of SQU with MAC [14], SPoC [12] and R-MAC [14]. The former (latter) number in each cell represents performance generated by off-the-shelf AlexNet (VGG-16). † represents we generate the results of MAC, SPoC and R-MAC, based on the authors' released codes. All experiments are performed without PCA Whitening.

Method	Holidays	Oxford5k	Paris6k	UKBench
R-MAC†	79.8/84.7	**54.0**/57.9	**66.1/76.4**	**3.55**/3.73
MAC†	73.7/79.1	45.2/53.0	51.6/67.0	3.48/3.65
SPoC†	77.5/82.6	43.3/52.8	52.5/63.2	3.38/3.68
SQU	**81.0/86.0**	51.4/**60.0**	59.6/72.4	**3.55/3.76**

3.2. Implementation Notes

In this work, we consider 2 CNN architectures : AlexNet [8] and VGG16 [9]. We test both off-the-shelf networks pre-trained on ImageNet ILSVRC classification data set and fine-tuned one tailored for image retrieval [16].

To learn the gating parameters w in the GatedSQU layer, we leverage the training dataset released by [16], referred to as 3D-Landmarks from here on. 3D-Landmarks contains 28559 images, organized into 713 clusters of famous landmarks worldwide. Following [16], we select 551 clusters (22156 images, 5974 of them are queries) for training and 162 clusters (6403 images, 1691 of them are queries) for validation. Each training tuple contains 1 query, 1 positive and 5 hard negative images. We sample hard negatives by 3 criteria: (1) query and negative belong to different clusters, (2) negative has most similar descriptor to query and (3) 5 negatives for each query are from different clusters. We initialize the gating parameters $\boldsymbol{w} = 0$ (i.e. equal contribution 0.5 for all channels) and set scale factor $s = 10$ for the GatedSQU layer, margin m 0.1, learning rate 0.001 dividing by 2 every 5 epochs, moment 0.9, weight decay 0.001, and batch size of 5 tuples. To accelerate feature extraction, we resize all training and validation images to 312×312 with aspect ratio killed. We train the network for 30 epochs. The trained model with the highest mAP on validation set is chosen for testing.

Finally, post-processing can be applied to the GatedSQU descriptors. We choose PCA whitening in this work. To be consistent with most related papers [12, 14, 13], we learn PCA matrix on Paris6k when evaluating on Oxford5k and vice versa. For Holidays and UKBench, we simply use the 3D-Landmarks dataset for PCA learning.

3.3. Evaluation on SQU

SQU vs. MAC, SPoC and R-MAC. We conduct retrieval experiments following standard practice in [14, 16, 17]: (1) Image size. We use the original image for Oxford5k, Paris6k and UKBench, while for Holidays we down sample the resolution to longer side equals to 1024 with aspect ratio maintained and (2) Cropped query. For Oxford5k and Paris6k, we crop query images using the provided bounding boxes. These setups are applied to the subsequent sections as well.

Table 2. Comparison of GatedSQU* with SQU* and CroW [13]. † denotes our implementations of SPoC and CroW based on the authors' released codes. All experiments are performed without PCA whitening.

Method	Holidays	Oxford5k	Paris6k	UKBench
SPoC†	82.6	52.8	63.2	3.68
CroW†	82.9	60.4	70.9	3.65
SQU	**86.0**	60.0	72.4	**3.76**
GatedSQU	**86.0**	64.1	75.4	3.75
SQU*	81.8	73.7	76.9	3.52
GatedSQU*	83.3	**76.3**	**78.2**	3.54

Table 1 shows the comparisons of SQU with MAC, SPoC and R-MAC, using off-the-shelf AlexNet and VGG16. We observe that SQU outperforms MAC and SPoC by a large margin on all datasets for both networks. Compared to R-MAC, SQU performs worse on Oxford5k and Paris6k, but achieves better performance on Holidays and UKbench.

3.4. Evaluation on GatedSQU

We train the GatedSQU layer with 2 configurations: GatedSQU with off-the-shelf VGG16 pre-trained on ImageNet (denotes as GatedSQU), and GatedSQU with fine-tuned VGG16 for instance retrieval task [16] (denotes as GatedSQU*).

From SQU* to GatedSQU*. Table 2 studies the advantage of GatedSQU* over SQU*. GatedSQU* consistently improves its SQU* counterpart on all four test datasets. For GatedSQU, similar trends can be observed on Oxford5k and Paris6k. Overall, the improvements on Oxford5k and Paris6k is relatively larger than on Holidays and UKBench. This is reasonable as 3D-Landmarks training set is building-oriented like Oxford5k and Paris6k, while Holidays and UKBench are scene-centric and object-centric respectively.

GatedSQU vs. CroW. Table 2 compares GatedSQU to CroW [13], which performs both channel weighting and spatial weighting to improve SPoC [12]. First, consistently with [13], we find CroW is superior to SPoC. Second, SQU performs comparable or better than CroW on all datasets. GatedSQU further increases the gap against CroW.

Off-the-shelf vs. Fine-tuned. Table 2 also compares GatedSQU with GatedSQU*. We observe GatedSQU* largely outperforms GatedSQU on buildings dataset (e.g. from 64.1% from 76.3% on Oxford5k), but is slightly worse than GatedSQU on Holidays and UKBench. Again, this is probably due to the fine-tuned network is biased toward a building-centric retrieval set. Training on more diverse datasets would improve the generalization ability of the network to general image retrieval scenarios.

3.5. Comparison with state-of-the-art

In this section, we compare PCA whitened GatedSQU* with the state-of-the-art. We split the comparisons into two parts according to the network used for experiments: off-the-shelf VGG16 and fine-tuned VGG16.

Table 3. Comparison of GatedSQU with the state-of-the-art, using off-the-shelf VGG16. Post.: post-processing, Dim.: dimensionality. § denotes query images for Oxford5k/Paris6k are without query object cropping, ‡ results reported in another paper [16], ◇ query object cropping is performed on activation maps rather than images for Oxford5k/Paris6k, † our implementations based on the authors' codes, following the same setup as GatedSQU. Gray background refers to global pooling approaches, in which the best results are highlighted in **bold**.

Method	Post.	Dim.	Holidays	Oxford5k	Paris6k	UKBench	Oxford105k	Paris106k
Triang.+ Democr. aggr. [6]	PCA	1024	72.0	56.2	-	3.51	50.2	-
Triang.+ Democr. aggr. [6]	PCA	512	70.0	52.8	-	3.49	46.1	-
Neural Codes [20]	PCA	512	74.9	43.5§	-	3.43	39.2§	-
Orderless Pooling [19]	PCAW	2048	80.8	-	-	-	-	-
R-MAC [14]	PCAW	512	86.9‡	66.9	83.0	3.74†	61.6	75.7
MAC [14]	PCAW	512	76.7‡	56.4‡	72.3‡	3.62†	47.8‡	58.0‡
SPoC [12]	PCAW	256	80.2	53.1◇	-	3.65	50.1	-
SPoC [12]	PCAW	512	82.8†	66.9†	76.5†	3.65†	-	-
CroW [13]	PCAW	512	84.9	68.2	79.6	3.63†	63.2	71.0
GatedSQU (Ours)	PCAW	512	**88.8**	**69.4**	**81.3**	**3.74**	**63.9**	**73.4**

Table 4. Comparison of GatedSQU* with the state-of-the-art, using fine-tuned VGG16. Most of the marks are with the same meaning as in Table 3, expect that ¶ denotes PCA is trained end-to-end with tri-R-MAC [17, 18].

Method	Post.	Dim.	Holidays	Oxford5k	Paris6k	UKBench	Oxford105k	Paris106k
Neural Codes [20]	PCA	512	78.9	55.7§	-	3.30	52.2§	-
NetVLAD [15]	PCAW	512	86.1	67.6◇	74.9◇	-	-	-
NetVLAD [15]	PCAW	512	85.8†	57.9†	64.6†	3.57†	-	-
tri-R-MAC [17, 18]	PCAW¶	512	87.9	79.9	85.9	3.59	-	-
tri-R-MAC + RPN [17, 18]	PCAW¶	512	88.7	83.2	87.2	3.62	78.6	79.7
sia-R-MAC [16]	PCAW	512	82.3	76.3	84.5	3.47†	68.5	77.1
sia-R-MAC [16]	Lw	512	82.5	77.0	83.8	3.59†	69.2	76.4
sia-MAC [16]	PCAW	512	77.1	76.1	79.0	3.43†	68.9	69.1
sia-MAC [16]	Lw	512	79.5	79.7	82.4	**3.55**†	73.9	74.6
GatedSQU* (Ours)	PCAW	512	**83.0**	**80.5**	**86.1**	3.50	**75.4**	**80.0**

Off-the-shelf network. Table 3 compares GatedSQU to state-of-the-art with off-the-shelf VGG16. Most of the approaches are post-processed with PCA whitening, and with the same dimensionality (i.e., 512). We observe that GatedSQU performs consistently better than other global pooling operations (in gray background) on all six test datasets. More importantly, GatedSQU is superior to R-MAC on all datasets expect Paris6k and Paris106k.

Fine-tuned network. Table 4 compares GatedSQU* to the state-of-the-art with VGG16 fine-tuned. GatedSQU* outperforms NetVLAD [15] by a large margin on Oxford5k and Paris6k, e.g. +20% when NetVLAD performs query cropping directly on images (denoted as †), while it performs a bit worse on Holidays and UKBench.

We compare GatedSQU* to sia-MAC/sia-R-MAC [16] which fine-tuned off-the-shelf VGG16 + MAC pooling with contrastive loss. Both sia-MAC and sia-R-MAC are post-processed by either PCA whitening or more advanced discriminative projection learned with labels (denoted as Lw) [16, 30]. We observe that GatedSQU* significantly outperforms all variants of sia-MAC/sia-R-MAC on all datasets.

In Table 4, we also present the best results of ROI-based pooling method tri-R-MAC [17, 18], which simultaneously trains all VGG16 layers, RPN layers, PCA whitening and R-MAC pooling with triplet loss in an end-to-end framework, with a much larger training set which contains ∼200k noisy landmark images and ∼50k clean ones selected from the noisy set. Our global GatedSQU* performs slightly better than tri-R-MAC without RPN (i.e. with default ROI sampling in R-MAC) on Oxford5k and Paris6k, though the GatedSQU layer and PCA whitening are trained separately with the much smaller 3D-Landmarks dataset.

4. CONCLUSION

In this work, we study the problem of pooling on deep convolutional features for image instance retrieval. We propose two strategies to improve global pooling. The first is a square-root pooling (SQU), a 2-norm operator between the classic 1-norm average and infinite norm max operations. The second is a gating function that further enhances SQU. Both SQU and GatedSQU achieve remarkable performance compared to state-of-the-art.

5. ACKNOWLEDGE

This work is supported by the National Natural Science Foundation of China (61661146005,U1611461) and in part by the PKU-NTU Joint Research Institute (JRI) sponsored by a donation from the Ng Teng Fong Charitable Foundation. And this work is partially supported by A*STAR AME Programmatic Fund A1892b0026.

6. REFERENCES

[1] D. Lowe, "Distinctive Image Features from Scale-invariant Keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[2] H. Jégou, M. Douze, C. Schmid, and P. Perez, "Aggregating Local Descriptors into a Compact Image Representation," in *CVPR*, 2010.

[3] F. Perronnin, Y. Liu, J. Sanchez, and H. Poirier, "Large-scale Image Retrieval with Compressed Fisher Vectors," in *CVPR*, 2010.

[4] R. Arandjelović and A. Zisserman, "All about vlad," in *CVPR*, 2013.

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

[6] Hervé Jégou and Andrew Zisserman, "Triangulation embedding and democratic aggregation for image search," in *CVPR*, 2014.

[7] Relja Arandjelović and Andrew Zisserman, "Three things everyone should know to improve object retrieval," in *CVPR*, 2012.

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[9] K Simonyan and A Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *ICLR*, 2015.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *CVPR*, 2016.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[12] A. Babenko and V. Lempitsky, "Aggregating local deep features for image retrieval," in *ICCV*, 2015.

[13] Yannis Kalantidis, Clayton Mellina, and Simon Osindero, "Cross-dimensional weighting for aggregated deep convolutional features," in *arXiv:1512.04065*, 2015.

[14] Giorgos Tolias, Ronan Sicre, and Hervé Jégou, "Particular object retrieval with integral max-pooling of CNN activations," in *ICLR*, 2016.

[15] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *CVPR*, 2016.

[16] Filip Radenović, Giorgos Tolias, and Ondřej Chum, "CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples," in *ECCV*, 2016.

[17] Albert Gordo, Jon Almazán, Jérome Revaud, and Diane Larlus, "Deep image retrieval: Learning global representations for image search," in *ECCV*, 2016.

[18] Albert Gordo, Jon Almazán, Jérome Revaud, and Diane Larlus, "End-to-end learning of deep visual representations for image retrieval," in *IJCV*, 2017.

[19] Yunchao Gong, Liwei Wang, Ruiqi Guo, and Svetlana Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *ECCV*, 2014.

[20] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky, "Neural Codes for Image Retrieval," in *ECCV*, 2014.

[21] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014.

[22] Sumit Chopra, Raia Hadsell, and Yann Lecun, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR*, 2005.

[23] J.Wang, Y.Wang, and et al., "Learning Fine-grained Image Similarity with Deep Ranking," in *CVPR*, 2014.

[24] Y-Lan Boureau, Jean Ponce, and Yann LeCun, "A theoretical analysis of feature pooling in vision algorithms," in *ICML*, 2010.

[25] Chen-Yu Lee, Patrick W. Gallagher, and Zhuowen Tu, "Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree," in *AISTATS*, 2016.

[26] H. Jégou, M. Douze, and C. Schmid, "Hamming Embedding and Weak Geometric Consistency for Large Scale Image Search," in *ECCV*, 2008.

[27] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object Retrieval with Large Vocabularies and Fast Spatial Matching," in *CVPR*, 2007.

[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 *CVPR*, 2008.

[29] D. Nistér and H. Stewénius, "Scalable Recognition with a Vocabulary Tree," in *CVPR*, 2006.

[30] Krystian Mikolajczyk and Jiri Matas, "Improving descriptors for fast tree matching by optimal linear projection," in *ICCV*.