

SPATIAL WEIGHTED FISHER VECTOR FOR IMAGE RETRIEVAL

Chengzuo Qi^{1,2}, Cunzhao Shi^{1,2}, Jian Xu^{1,2}, Chunheng Wang^{1,2}, and Baihua Xiao^{1,2}*

¹Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China

²University of Chinese Academy of Sciences, Beijing, China

{qichengzuo2013, cunzhao.shi, xujian2015, chunheng.wang, baihua.xiao}@ia.ac.cn

ABSTRACT

Several recent works interpret convolutional features produced by deep convolutional neural networks as local descriptors. Existing high-dimensional aggregation based methods, e.g., Fisher Vector (FV) obtain inferior performance to pooling based methods in most situations, and we observe that it is mainly caused by the ignorance of spatial weights. In this paper, we propose a novel method named spatial weighted Fisher Vector (SWFV) to enhance the representation of FV by injecting the spatial weight map to FV. In addition, we further analyze the distribution of spatial weights and propose truncated spatial weighted FV (TSWFV). Experimental results on two benchmark datasets demonstrate that the two proposed methods achieve competitive results compared with other global representation based methods.

Index Terms— Fisher Vector, Spatial Weight, Convolutional Feature

1. INTRODUCTION

Image retrieval has been an active research topic for decades due to its significant applications, such as visual search [1], and person identification [2]. Traditional solutions represent an image using a bag-of-visual-words (BOVW) based on SIFT [3]. The subsequent works focus on strengthening the discriminability of global representation. Specially, Fisher Vector (FV) [4] plays an important role in image retrieval community, which generates codebooks using Gaussian Mixture Model (GMM), and obtain the image representation by taking the derivative with respect to GMM parameters. Perronnin *et.al.* [5] improve FV by power normalization and spatial pyramid. Husain and Bober [6] also modify FV by ranked-based multi-assignment so as to increase the overall robustness to noise and outliers.

Recently, some researchers pay more attention to the aggregation of local descriptors from convolutional neural network (CNN) instead of SIFT. The local convolutional descrip-

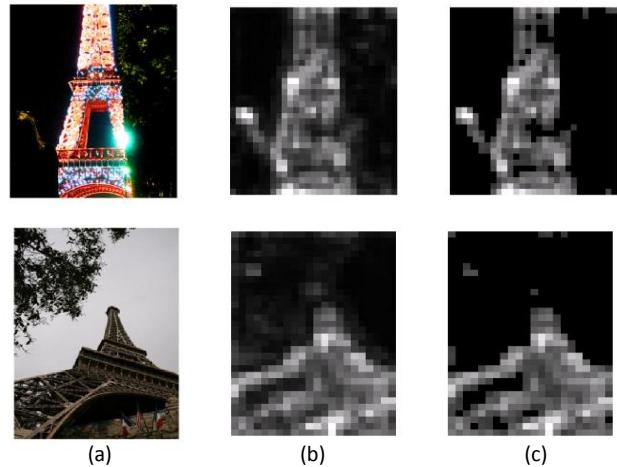


Fig. 1: Visualization of spatial weight map. (a) source images from Paris6K dataset. (b) spatial weight map. (c) truncated spatial weight map.

tors are aggregated by sum pooling combining with centering prior [7]. Kalantidis *et.al.* [8] enhance the sum pooling with cross-dimensional weighting strategy. Meanwhile, the max pooling is conducted on convolutional descriptors [9]. Tolia *et.al.* [10] perform the max pooling on the multi-scale sliding regions captured from the convolutional feature maps, and then combine the collection of regional feature vectors into a powerful global signature by sum pooling, i.e., R-MAC. The R-MAC is then combined with the networks fine-tuned on clean data [11, 12]. The sliding region strategy can further be replaced by a more powerful object detection tool, i.e., Faster R-CNN [13] to generate region-level representation [14]. The point-level and scene-level features can be fused with the region-level feature to boost the retrieval performance [15]. Traditional high-dimensional aggregation methods, e.g., vector of locally aggregated descriptors (VLAD) and FV can also be applied on the local convolutional descriptors. VLAD is conducted on the convolutional features of different layers in CNN [16, 17, 18]. The FV combined with local convolutional descriptors also achieves promising results [7, 19]. However, existing FV and VLAD based methods equally treat each convolutional descriptors which results in involving background

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant Nos. 61531019, 61601462 and 71621002. Corresponding author: Cunzhao Shi, Tel: +86-10-82544488; Fax: +86-10-62650820

noises in the global feature representation.

In this paper, we propose a novel encoding method named spatial weighted FV (SWFV) in the framework of FV combined with convolutional descriptors to overcome the above-mentioned drawback. The SWFV utilize a soft selection to assign adaptive weights for local descriptors. We employ spatial weight map [8] to reflect the importance of local regions, and therefore highlight the foreground area which is beneficial for the global representation, as it is visualized in Fig. 1. Moreover, we propose an adaptive thresholding method to truncate the overall spatial weights. The truncated spatial weighted FV (TSWFV) conduct a hard selection on local descriptors by directly eliminating the local descriptors captured from background area. It finally generates a more powerful global representation with less number of local convolutional descriptors. Experimental results on two benchmark datasets show that the proposed SWFV and TSWFV methods both outperform the pooling based and FV based methods.

The rest of this paper is organized as follows. Section 2 details the proposed methods, i.e., SWFV and TSWFV while Section 3 presents experimental results on benchmark datasets. The paper concludes with Section 4.

2. METHODS

In this section, we first introduce the original Fisher Vector, and then give a formal description of the proposed SWFV and TSWFV. In what follows, we introduce each component in detail respectively.

2.1. Fisher Vector

The convolutional activations from CNN form a $(W * H * C)$ tensor, where W , H and C separately denotes the width, height and channel number of the feature tensor. The tensor is commonly regarded as a map consisting of $(W \times H) C$ -dimensional local descriptors. Let denote the map as F .

The original FV [4] models the distribution of convolutional feature descriptors by learning GMM and represents an image by considering the gradient with respect to GMM parameters.

The parameters ω_k , μ_k , Σ_k of the k^{th} GMM component denotes the weight, mean vector, and covariance matrix respectively. We assume that the covariance matrix Σ_k is diagonal, so it can be denoted as σ_k^2 . GMM assigns descriptor at position (i, j) , i.e., F_{ij} to Gaussian component k with the soft assignment weight $\varphi_{ij}(k)$ given by the posteriori probability. The C -dimensional derivatives with respect to the mean μ_k and the diagonal covariance matrix σ_k^2 of Gaussian component k are respectively defined as:

$$f_{\mu_k} = \frac{1}{T\sqrt{\omega_k}} \sum_{i=1}^W \sum_{j=1}^H \varphi_{ij}(k) \left(\frac{F_{ij} - \mu_k}{\sigma_k} \right) \quad (1)$$

$$f_{\sigma_k} = \frac{1}{T\sqrt{\omega_k}} \sum_{i=1}^W \sum_{j=1}^H \varphi_{ij}(k) \left[\frac{(F_{ij} - \mu_k)^2}{\sigma_k^2} - 1 \right] \quad (2)$$

where $T = W \times H$ is the number of local descriptors in an image.

The FV global representation f_g of an image is obtained by concatenating the gradients f_{μ_k} and f_{σ_k} ($k = 1 \dots N$) for all Gaussian components and therefore the dimensionality D is equal to $2 \times C \times N$. The improved FV [5] utilizes power normalization upon f_g as follows

$$f(z) = \text{sign}(z)|z|^\alpha \quad (3)$$

The final vector is normalized by L_2 norm for further steps, and we choose the improved FV as our baseline encoding method. We set α as 0.5 for all the experiments.

2.2. Proposed Approaches

2.2.1. Spatial Weighted Fisher Vector

As we can see, the convolutional feature map is, in some way, like the traditional dense descriptor which consists of large area of background. The original FV aggregates the dense descriptors with an equal weight, which means that foreground shares the same importance with background in the final global representation. We modify the original FV by incorporating spatial weights which are generated as follows

$$S_{ij} = \sum_{c=1}^C F_{ij}^c \quad (4)$$

where F_{ij}^c denotes the activation of c^{th} channel of the feature map F at position (i, j) . Then, the spatial weight map is normalized by L_1 norm, and it is visualized in Fig. 1 (b), from which we can see that the high-weight positions mainly lie in the foreground areas. It verifies that S_{ij} can effectively capture the importance of the local descriptor at position (i, j) .

Then, we employ spatial weights on the original FV.

$$\hat{f}_{\mu_k} = \frac{1}{T\sqrt{\omega_k}} \sum_{i=1}^W \sum_{j=1}^H S_{ij} \varphi_{ij}(k) \left(\frac{F_{ij} - \mu_k}{\sigma_k} \right) \quad (5)$$

$$\hat{f}_{\sigma_k} = \frac{1}{T\sqrt{\omega_k}} \sum_{i=1}^W \sum_{j=1}^H S_{ij} \varphi_{ij}(k) \left[\frac{(F_{ij} - \mu_k)^2}{\sigma_k^2} - 1 \right] \quad (6)$$

where S_{ij} denotes the weight of the C -dimensional descriptor at position (i, j) . The difference between Equation (5), (6) and Equation (1), (2) is the bold part, i.e., S_{ij} .

By injecting the spatial weight map into FV encoding, the proposed SWFV can be interpreted as a soft selection of the local convolutional descriptors. We conduct the soft selection by assigning higher weight for the local convolutional descriptors of foreground object, leading to a more powerful global signature.

2.2.2. Truncated Spatial Weighted Fisher Vector

The proposed SWFV suppresses the background part by assigning a smaller weight for the local descriptors. We further enhance this method by ignoring the background part. By conducting truncation on the spatial weight map, the spatial weights less than pre-defined threshold are all assigned with a weight of 0. The size of the spatial weight map varies with the scale of the input image, and the maximum amplitude of each spatial weight map fluctuates widely. Hence, the pre-defined holistic threshold value is inappropriate, and we propose a new method to calculate an adaptive threshold for each spatial weight map. We first reshape the spatial weighting map S , and thus getting a M , i.e., $(W \times H)$ -dimensional vector S_v . Then the vector is sorted in descending order as follows

$$S_v = [x_1, x_2 \dots x_n, x_{n+1}, \dots x_M] \quad (7)$$

The top N elements are chosen as follows.

$$N = \arg \min_n \left(\sum_{k=1}^n x_k > T \right) \quad (8)$$

where T is a measure of the minimum portion of the elements that should be kept. Then, the truncated vector is obtained.

$$\tilde{S}_v = [x_1, x_2 \dots x_N, 0, \dots 0] \quad (9)$$

Finally, we reshape the obtained vector \tilde{S}_v back to the same size of S , and thus we can get our truncated spatial weight map \tilde{S} , and, the S_{ij} term in both Equation (5) and (6) are replaced by \tilde{S}_{ij} . We also visualize \tilde{S}_{ij} in Fig. 1 (c), from which we observe that our truncated strategy obtains a more clean spatial weight map compared with Fig. 1 (b). \tilde{S} further eliminate the top-left and top-right background area of S for the bottom image and top image respectively.

Similar to SWFV, TSWFV can be interpreted as a hard selection of the local convolutional descriptors, which keeps the local convolutional descriptors more strictly and obtains a more compact and meaningful global representation.

3. EXPERIMENTS

3.1. Implementation Details

We extract convolutional features using pre-trained model VGG16 [20], and all the input images are zero-centered by RGB mean pixel subtraction before they are fed into the network. Meanwhile, similar to [8], we keep the original size of the images and select feature maps of the last pooling layer, i.e., pool5 as our primitive convolutional features. Then, we apply the proposed encoding methods on the local convolutional descriptors. Finally, PCA is performed on the high-dimensional encoded features without whitening. All the experiments are conducted on the cropped versions of the queries.

3.2. Datasets and protocols

The experiments are conducted on two benchmark datasets.

- The Oxford5K dataset [21] consists of 5,063 photographs of Oxford landmarks. 55 images corresponding to 11 buildings/lanmarks are treated as queries.
- The Paris6K dataset [22] contains 6,412 images of Paris landmarks. Similar to Oxford5K, 55 queries are fixed.

Following the corresponding standard evaluation protocols, this paper uses the mean average precision (mAP) to evaluate the accuracy of image search, which is the mean value of the average precision (AP).

3.3. Retrieval results

Table 1 and 2 present the results of proposed SWFV compared with original FV on the Oxford5K and Paris6K datasets respectively. From Table 1 and 2, we can see that the proposed SWFV remarkably outperforms the original FV on both Oxford5K and Paris6K datasets, which verifies that the soft selection of the spatial weighting strategy is effective, and therefore takes full advantage of the discriminability of FV. Table 1 shows that the performance gets better with the increase of the dimension after PCA on Oxford5K. While interestingly, the results for high dimensional SWFV on Paris6k achieves inferior (around 1% less) results than low dimensional SWFV. This phenomena may be explained by the different feature distribution of the two separate datasets, and PCA eliminates the influences of redundancies on Paris6K dataset. We also observe that the performance of both SWFV and original FV vary slightly with the change of GMM parameters. This may be caused by the highly overlapped receptive fields of two nearby local convolutional descriptors, and GMM with a less number of components is enough to capture the distribution of the dense convolutional descriptors.

Fig. 2 shows the comparison between TSWFV and SWFV. Several observations can be drawn from these results. First, comparing with SWFV, TSWFV achieves better results in most situations on both Oxford5K and Paris6K datasets. This phenomenon demonstrates the effectiveness of the hard selection strategy of TSWFV, and also verifies that TSWFV can further suppresses the background noise while making the global FV signature focusing on the foreground object. Secondly, consistently with Table 1 and 2, features at relatively low dimensions may achieve better performance on Paris6K while they perform the opposite way for Oxford5K. The opposite tendency of Fig. 2 for Pairs6K and Oxford5K may be induced by the different distributions of the two datasets. Images in Paris6K possess relatively smaller foregrounds, which can also be verified in Table 3 (the mean proportion of the remaining positions for Paris6K is smaller). That is to say, after truncation, the remaining number of foreground descriptors in Paris6K is obviously smaller than Oxford5K. Hence,

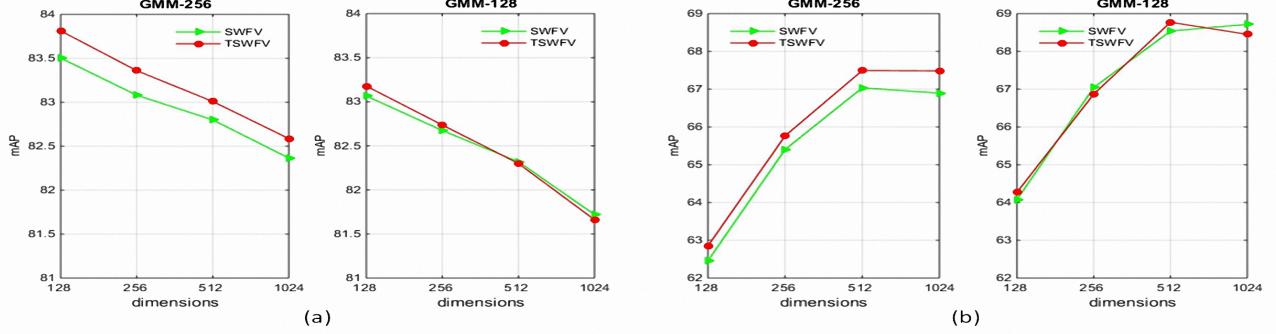


Fig. 2: Comparison of TSWFV and SWFV. (a) results on Paris6K. (b) results on Oxford5K.

Table 1: Comparison with original FV on the Oxford5K dataset

GMM	Dimension	FV	SWFV
64	128	0.455	0.641
	256	0.522	0.667
	512	0.568	0.672
	1024	0.584	0.671
128	128	0.444	0.641
	256	0.508	0.670
	512	0.560	0.685
	1024	0.580	0.687
256	128	0.443	0.625
	256	0.498	0.654
	512	0.541	0.670
	1024	0.572	0.669

Table 2: Comparison with original FV on the Paris6K dataset

GMM	Dimension	FV	SWFV
64	128	0.716	0.834
	256	0.717	0.828
	512	0.716	0.824
	1024	0.711	0.818
128	128	0.703	0.831
	256	0.707	0.827
	512	0.709	0.823
	1024	0.707	0.817
256	128	0.712	0.835
	256	0.716	0.831
	512	0.717	0.828
	1024	0.716	0.824

for Paris6K, lower dimensional global feature is enough to represent the whole image while increasing the dimension introduces redundant information which will injure the performance. However, considering that Oxford5k owns larger number of descriptors, higher dimensional global feature obtains a better global signature.

We further analyze the distribution of spatial weights. We visualize the sorted spatial weight vector, i.e., S_v in Equation (7) of one random selected image from Oxford5K and Paris6K in Fig. 3. And we also tag the truncated position and total number of local convolutional descriptors on Fig. 3. We observe that positions with high weights only occupy a small portion while nearly half positions own a small weight. By a heuristic method, we select the truncation position at a relatively flat area, and then set the truncation threshold, i.e., T in Section 2.2.2 of Oxford5K and Paris6K as 0.9 and 0.8. We also calculate the percentage of the remaining positions after truncation and denote it as P , e.g., $P = \frac{299}{704}$ for the left figure in Fig. 3, where P is equal to $\frac{N}{M}$ in Section 2.2.2. Then we present the mean and variance value of P for all the images from the two datasets in Table 3, from which we observe that the variance is very small, which verifies that the sorted spatial weights of all the images follow a consistent distribution which is similar with the shape in Fig. 3. The relatively smaller mean values demonstrate that TSWFV finally keeps only half number of the original convolutional descriptors.

Table 3: Mean and Variance of P (proportion of the remaining positions)

Dataset	Mean	Variance
Paris6K	0.4822	0.0081
Oxford5K	0.6473	0.0055

Table 4 shows the comparison between our approaches and other methods. The proposed TSWFV achieves the best results on Paris6K dataset for all the dimensions, i.e., 512, 256 and 128. For Oxford5K dataset, our methods (both TSWFV and SWFV) perform better than all other methods with 512-dimensional features. For lower dimensions, con-

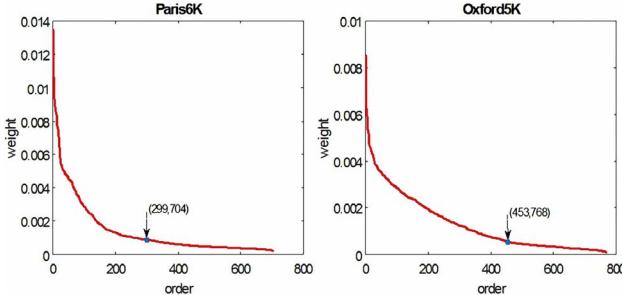


Fig. 3: Sorted spatial weights curve. Left: the spatial weights of paris_defense_000075 from Paris6K. Right: the spatial weights of ashmolean_000063 from Oxford5K. The text on the figure denotes the order value of the truncated position and the maximum order value.

sidering the fact that our methods only utilize one layer of VGG16, we still get comparable results with CCS [15] which fuses three-level features. The comparison of our methods and uCroW [8], i.e., sum pooling shows that our approaches which are based on Fisher Vector can better encode convolutional features than simple pooling strategies. Our methods also outperform CroW [8], which incorporates spatial weight and channel sparsity sensitive channel weight while we only use the spatial weight map.

Table 4: Comparison with other methods

Method	Dimension	Oxford5K	Paris6K
uCroW [8]	128	0.580	0.729
CroW [8]	128	0.592	0.746
CCS [15]	128	0.648	0.768
SWFV	128	0.641	0.835
TSWFV	128	0.643	0.838
uCroW [8]	256	0.635	0.759
CroW [8]	256	0.654	0.779
R-MAC [10]	256	0.561	0.729
CCS [15]	256	0.676	0.744
SWFV	256	0.670	0.831
TSWFV	256	0.669	0.834
uCroW [8]	512	0.666	0.767
CroW [8]	512	0.682	0.796
R-MAC [10]	512	0.668	0.830
CCS [15]	512	0.673	0.722
SWFV	512	0.685	0.828
TSWFV	512	0.688	0.830

The computational complexity of our method is almost equal to the original Fisher Vector based methods with the negligible element-wise addition cost. Therefore, the proposed SWFV and TSWFV do not increase the computation complexity nor introduce additional parameters. As the com-

pared methods do not report their retrieval time, we analyze the retrieval time of our method and the baseline method (i.e., Fisher Vector). The running time for computing our spatial weighting map only takes $0.3ms$ per image, whereas the Fisher Vector encoding and succeeding PCA operations take about $0.08s$ per image. In other words, the proposed method is almost as fast as the original Fisher Vector while obtaining a better performance. Besides, the proposed method is based on global representation, which can be combined with other binary coding methods efficiently. All the experiments are conducted with open source package (Caffe and VLFeat) on a Windows server with 3.1GHz CPU, 112G RAM and one K40 GPU in Matlab 2014b.

4. CONCLUSIONS

This paper proposes two novel methods to improve existing FV based convolutional feature encoding for image retrieval. We first inject spatial weight map into Fisher Vector encoding, and experimental results show that our proposed spatial weighted FV (SWFV) provides a better encoding of convolutional features than original FV. Then, we further analyze the distribution of spatial weights and propose truncated spatial weighted FV (TSWFV). Results on two benchmark datasets demonstrate that TSWFV effectively suppresses the background noise and achieves better performance with only half number of local convolutional descriptors. Meanwhile, there are several future study directions, for example, combining the convolutional features of different layers to generate a more powerful global signature, and further filtering the background outliers of the spatial weighing map.

5. REFERENCES

- [1] Relja Arandjelović and Andrew Zisserman, “Three things everyone should know to improve object retrieval,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2911–2918.
- [2] Liang Zheng, Shengjin Wang, Lu Tian, Fei He, Ziqiong Liu, and Qi Tian, “Query-adaptive late fusion for image search and person re-identification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1741–1750.
- [3] David G Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [4] Florent Perronnin and Christopher Dance, “Fisher kernels on visual vocabularies for image categorization,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.

- [5] Florent Perronnin, Jorge Sánchez, and Thomas Mensink, “Improving the fisher kernel for large-scale image classification,” in *European conference on computer vision*. Springer, 2010, pp. 143–156.
- [6] Syed Sameed Husain and Miroslaw Bober, “Improving large-scale image retrieval through robust aggregation of local descriptors,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.
- [7] Artem Babenko and Victor Lempitsky, “Aggregating local deep features for image retrieval,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1269–1277.
- [8] Yannis Kalantidis, Clayton Mellina, and Simon Osindero, “Cross-dimensional weighting for aggregated deep convolutional features,” *arXiv preprint arXiv:1512.04065*, 2015.
- [9] Ali S Razavian, Josephine Sullivan, Stefan Carlsson, and Atsuto Maki, “[paper] visual instance retrieval with deep convolutional networks,” *ITE Transactions on Media Technology and Applications*, vol. 4, no. 3, pp. 251–258, 2016.
- [10] Giorgos Tolias, Ronan Sicre, and Hervé Jégou, “Particular object retrieval with integral max-pooling of cnn activations,” *arXiv preprint arXiv:1511.05879*, 2015.
- [11] Filip Radenović, Giorgos Tolias, and Ondřej Chum, “Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples,” *arXiv preprint arXiv:1604.02426*, 2016.
- [12] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus, “Deep image retrieval: Learning global representations for image search,” *arXiv preprint arXiv:1604.01325*, 2016.
- [13] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.
- [14] Amaia Salvador, Xavier Giró-i Nieto, Ferran Marqués, and Shin’ichi Satoh, “Faster r-cnn features for instance search,” *arXiv preprint arXiv:1604.08893*, 2016.
- [15] Ke Yan, Yaowei Wang, Dawei Liang, Tiejun Huang, and Yonghong Tian, “Cnn vs. sift for image retrieval: Alternative or complementary?,” in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 407–411.
- [16] Joe Yue-Hei Ng, Fan Yang, and Larry S Davis, “Exploiting local features from deep networks for image retrieval,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 53–61.
- [17] Yunchao Gong, Liwei Wang, Ruiqi Guo, and Svetlana Lazebnik, “Multi-scale orderless pooling of deep convolutional activation features,” in *European Conference on Computer Vision*. Springer, 2014, pp. 392–407.
- [18] Relja Arandjelović, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic, “Netvlad: Cnn architecture for weakly supervised place recognition,” *arXiv preprint arXiv:1511.07247*, 2015.
- [19] Vijay Chandrasekhar, Jie Lin, Olivier Morère, Hanlin Goh, and Antoine Veillard, “A practical guide to cnns and fisher vectors for image instance retrieval,” *Signal Processing*, vol. 128, pp. 426–439, 2016.
- [20] Karen Simonyan and Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [21] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.
- [22] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman, “Lost in quantization: Improving particular object retrieval in large scale image databases,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.