Multiple Saliency and Channel Sensitivity Network for Aggregated Convolutional Feature

Xuanlu Xiang, Zhipeng Wang, Zhicheng Zhao, Fei Su
School of Communication and Information Engineering
Beijing Key Laboratory of Network System and Network Culture
Beijing University of Posts and Telecommunications
bigcabbagexxl@gmail.com,{wzpycg, zhaozc, sufei}@bupt.edu.cn

Abstract

In this paper, aiming at two key problems of instance-level image retrieval, i.e., the distinctiveness of image representation and the generalization ability of the model, we propose a novel deep architecture - *Multiple Saliency and Channel Sensitivity Network(MSCNet)*. Specifically, to obtain distinctive global descriptors, an attention-based multiple saliency learning is first presented to highlight important details of the image, and then a simple but effective channel sensitivity module based on Gram matrix is designed to boost the channel discrimination and suppress redundant information. Additionally, in contrast to most existing feature aggregation methods, employing pre-trained deep networks, MSCNet can be trained in two modes: the first one is an unsupervised manner with an instance loss, and another is a supervised manner, which combines classification and ranking loss and only relies on very limited training data. Experimental results on several public benchmark datasets, i.e., Oxford buildings, Paris buildings and Holidays, indicate that the proposed MSCNet outperforms the state-of-the-art unsupervised and supervised methods.

1 Introduction

Content-based Image Retrieval (CBIR) is a very active visual task in the field of computer vision. After the work of (Krizhevsky, Sutskever, and Hinton 2012), hand-crafted features (e.g. SIFT (Lowe 2001)) have given way to CNN-based ones. In (Gong et al. 2014) and (Babenko et al. 2014), the activations of the fully-connected layers were directly utilized to build image representation for similarity comparison. Since then, more literatures have demonstrated that the activations of convolutional layers can provide better performance for object retrieval because of conveying spatial information (Babenko and Lempitsky 2015; Tolias, Sicre, and Jégou 2016; Kalantidis, Mellina, and Osindero 2016; Jimenez, Alvarez, and Giro-i Nieto 2017; Seddati et al. 2017; Xu et al. 2018). These works mostly used networks pre-trained for image classification tasks. This is challenging because CBIR needs distinctive compact codes but the networks for classification are insensitive to intra-class differences, and do not fit the purpose of instance search. More recently, some supervised methods by fine-tuning CNN models with a triplet ranking loss (Gordo et al. 2016) and pairwise similarity (Radenović, Tolias, and Chum 2016) were proposed to enhance the power of image representation and achieved the state-of-the-art results for instance retrieval. However, their models were based on a large-scale training set, which restricted the scalability of the system. So far, the distinctiveness of image representation and the generalization ability of the model have been the key problems for this task.

Consequently, we aim at exploring three key factors of particular object retrieval and focus on learning a good image representation on a very limited training dataset. The first key point is to employ multi-scale representation(Radenović, Tolias, and Chum 2018). According to (Gordo et al. 2017)(Seddati et al. 2017), multi-scale feature extraction can enhance the robustness of the model, thus needs to be further mined. Additionally, exploiting different details of spatial semantic information could also be considered. For example, (Xu et al. 2018) defined special channels of normalized feature maps as a kind of part detectors, and then aggregated multiple ones to generate convolutional features. This proposal significantly outperformed previous state-of-the-art unsupervised CNN methods. However, the choosing of part detectors has two problems: (i) it was too simple to adapt complicated retrieval scenarios, (ii) it needed too many attention maps to achieve a good result.Therefore, how to adaptively learn saliency masks was another key factor of performance improvement. The last important factor is how to perform channel weighting for feature recalibration. For instance, (Kalantidis, Mellina, and Osindero 2016; Jimenez, Alvarez, and Giro-i Nieto 2017; Siméoni et al. 2017) used sparsity-sensitive weighting (SSW) scheme to improve the power of image representations. However, this method did not utilize the correlations between different channels and could not be end-to-end learned. Inspired by (Gatys, Ecker, and Bethge 2016; 2015), which leveraged Gram matrix to describe the structural information for image style transfer, we regard Gram matrix as a covariance matrix and present a Gram matrix based channel sensitivity weighting scheme.

In this paper, we propose a novel and simple network based on multiple saliency and Gram matrix based channel sensitivity, named as **Multiple Saliency and Channel Sensitivity Network(MSCNet)**(see Fig 1). Considering that annotation data is precious for retrieval task, we propose to train this network in both unsupervised and supervised manners based on

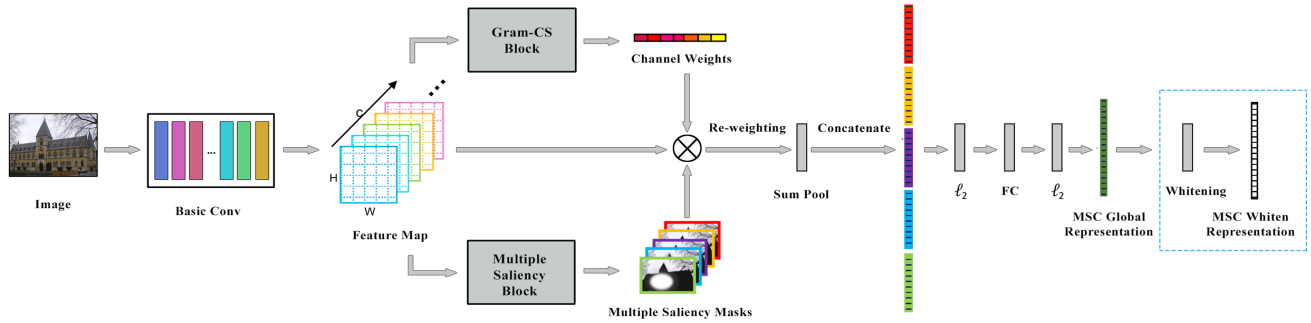

Figure 1: Our **Multiple Saliency and Channel Sensitivity Network** framework. The modules in the blue box are only adopted for testing.

a very limited training dataset containing about one-seventh of (Gordo et al. 2016) with only image level labels. For the unsupervised mode, motivated by the instance loss for image-text matching (Zheng et al. 2017), we leverage it to train our multiple saliency block (MSB) without any supervised information. For the supervised one, we use standard classification loss and triplet loss with batch hard mining (Hermans, Beyer, and Leibe 2017) to fine-tune the whole network for two-stage learning with only image-level labels. Furthermore, we design a new loss function called **Normalized Gram-Difference Loss** also based on Gram matrix to enforce the saliency masks to focus on different parts. The main contributions of this paper are:

- A novel instance-level image retrieval architecture (MSCNet) is proposed to obtain powerful image representation.
- The proposed network can be trained or fine-tuned in both an unsupervised and supervised way based on a very limited training set, which indicates a good generalization ability.
- A new loss function named as Normalized Gram-Difference Loss, is presented for saliency learning and an effective channel sensitivity weighting scheme is designed to enhance the distinctiveness of image representation.
- Extensive experiments on five public datasets demonstrate that the proposed algorithm outperforms state-of-the-art methods for both unsupervised and supervised modes.

This paper is structured as follows: In *Section 2* we introduce related works, while in *Section 3* we discribe the MSCNet framework in details. *Section 4* presents our training methods and we report the experimental results for instance retrieval in *Section 5*. Finally, *Section 6* concludes the paper.

2 Related Work

Image retrieval has embraced the power of deep learning in recent years. Early approaches directly used fully-connected layers as global image representation (Babenko et al. 2014). Recently, several literatures focused on combining convolutional features to explore regions of interest like (Tolias, Sicre, and Jégou 2016; Mohedano et al. 2016; Babenko and Lempitsky 2015). More recent works demonstrated that focusing the saliency areas of the image could eliminate the randomness of previous works.For instance, (Kalantidis, Mellina, and Osindero 2016) applied cross-dimensional weighting to boost the effect of highly active spatial responses and (Jimenez, Alvarez, and Giro-i Nieto 2017) leveraged Class Activation Maps (Zhou et al. 2016) to obtain spatial semantic-aware weights for convolutional features. Furthermore, (Xu et al. 2018) used multiple special channels of normalized feature maps as part detectors to aggregate convolutional features. However, this method was hard to adapt complicated retrieval scenarios and used too many attention maps to cover all details in the image. Impressively, (Kalantidis, Mellina, and Osindero 2016; Jimenez, Alvarez, and Giro-i Nieto 2017; Siméoni et al. 2017) leveraged SSW to restrain the disturbance of channel burstiness. However, this algorithm ignored the correlations between different channels and was not differentiable for end-to-end learning. To address the problem, we present a novel channel weighting method to intensify the distinctive feature for retrieval.

Another approach of improving the power of image representation is to fine-tune the network by metric learning. (Gordo et al. 2016) employed a triplet ranking loss, while (Radenović, Tolias, and Chum 2016) utilized a contrastive loss, and achieved the state-of-the-art performance. However, their approaches were still based on (Tolias, Sicre, and Jégou 2016) with a large-scale training set. In this paper, we propose to train in both an unsupervised and weakly supervised manner, and fine-tune the network by metric learning only rely on limited annotation data.

3 Multiple Saliency and Channel Sensitivity Network

This section introduces our MSCNet for instance-level image retrieval. Our network architecture can be decomposed into five main parts (see Fig. 1): Firstly, dense features are extracted from several convolutional (conv) layers to produce conv feature maps. Secondly, these feature maps are fed-forwarded through two parallel network modules, focus-

ing on spatial saliency and channel sensitivity respectively, to obtain several spatial weighting masks and one channel weighting vector. Thirdly, after weighted-sum aggregation, several dense descriptors are generated and concatenated as a high dimensional vector. Then normalization and linear projection are applied to get a low dimensional representation. Finally, PCA-Whitening as an effective post-processing step is considered to improve the performance of the representation.

3.1 Convolutional Features Extraction

We adopt a fully convolutional network (FCN) to extract features from an image. In particular, these FCN layers (Basic Conv) are based on a standard architecture for generic object recognition, while their classification layers are discarded.

Given an input image I, the FCN output is a 3D tensor $\boldsymbol{\chi}$ with $C \times H \times W$ dimensions, where C is the number of feature maps in the final layer. Here we assume that the outputs are extracted from the Rectified Linear Unit (ReLU) (Nair and Hinton 2010) activation layer such that $\boldsymbol{\chi}$ is non-negative.

During test time, we adopt a multi-scale procedure similar to (Gordo et al. 2017). After extracting features, we sum-aggregate and l_2-normalize them to obtain the final one.

3.2 Multiple Saliency Block (MSB)

One of the main factors which affects the performance of instance retrieval is that the semantic objects of interest are often submerged by irrelevant objects and background. In this subsection, we present an efficient MSB to produce a certain number of saliency masks as spatial weights and apply to feature maps.

As training data is limited, this module has to be simple but effective to extract the salient regions. We are similarly training the component learners in Bagging which achieve good performance by reducing the model variance. Based on this insight, MSB is simply designed as n convolutional filters, i.e., a convolution layer with n out-channels, followed by a sigmoid activation to restrict the saliency score to $[0 \sim 1]$. To consider the neighborhood of pixels, we set the kernel size to be 3×3 ($padding = 1$). As a result, our MSB is flexible with a slight calculation cost and works well in practice.

3.3 Gram-CS Block

Another main factor affecting the performance of retrieval is that treating the deep features from different channels fairly weakens the distinctiveness between them. Hence, a channel weighting scheme to perform feature recalibration is necessary. In this subsection, we present our channel weighting method based on the Gram matrix.

The Gram Matrix. Gram matrix is originally proposed to represent the image style (Gatys, Ecker, and Bethge 2016; 2015). We formulate the vectorised feature maps in the output layer as $F = [f_1, f_2, ..., f_C]$, where C is the channel number. The feature correlations of $\boldsymbol{\chi}$ are given by computing the Gram matrix $G \in \mathbb{R}^{C \times C}$, where G_{ij} is the inner product between the vectorised feature maps f_i and f_j:

$$G_{ij} = \sum_k f_{ik} \cdot f_{jk} \tag{1}$$

The inner product of f_i and f_j, which are non-negative, describes the joint response between them. Therefore, we consider the Gram matrix as a covariance matrix, describing the correlations between different channels. Due to its reflection of the structural information of an image, we can use the Gram matrix to tune the importance of channels.

Gram Matrix Based Channel Sensitivity. Channel sensitivity (CS) means to analyze the interdependencies between the channels of deep features to selectively boost informative features and suppress redundant ones (Hu, Shen, and Sun 2017). We propose to combine the advantages of Gram matrix and sparsity-sensitive channel weights (SSW) (Kalantidis, Mellina, and Osindero 2016) to derive a channel weighting.

The Gram matrix is symmetric, and we adopt a mean aggregation for each column to get a **Mean Gram Vector** (V):

$$V = [v_1, v_2, ..., v_i, ..., v_C], \tag{2}$$

where v_i is the mean of the i-th column in the Gram matrix and describes the average strength and frequency of the correlated response between feature map f_i with others. SSW (Kalantidis, Mellina, and Osindero 2016) has indicated that infrequently occurring features could provide important signal. Therefore, based on Mean Gram Vector (V), we devise a channel weighting similar to the concept of *inverse document frequency*:

$$\mathcal{W}_i = \log\left(\frac{K\epsilon + \sum_h {v_h}^\gamma}{\epsilon + {v_i}^\gamma}\right), \tag{3}$$

where ϵ is a small constant for numerical stability and γ is the power-scaling parameter. Here we choose $\gamma = 2$ in our experiments. We combine (1) (2) (3) to form our Gram-CS block. As each step is differentiable, we can plug it into the network in both training and testing phases. Moreover, this block is non-parametric and will also take only slight computation costs.

To provide further insight into the effectiveness of our Gram-CS block, in Fig. 2, we visualize the pair-wise correlation of the vectors of our Gram-CS channel weights for both images in the Oxford5K (Philbin et al. 2007) and Paris6K (Philbin et al. 2008) query-datasets. The results show that our channel weighting is highly correlated for same class images and less correlated for different class ones. It demonstrates that the Gram-CS block can output discriminative information.

3.4 Weighted-sum Aggregation and Projection

In this subsection, we describe our method to aggregate the convolutional features into a global compact representation.

Weighted-sum Aggregation. We first use the n saliency masks and one channel weighting vector $\mathcal{W}$ to construct n dense representations by weighted-sum pooling of the $C \times H \times W$ feature maps $\boldsymbol{\chi}$:

$$\psi_i^{(k)} = \mathcal{W}_i \sum_{x=1}^{H} \sum_{y=1}^{W} \boldsymbol{\chi}_{ixy} M_{xy}^{(k)}, \tag{4}$$

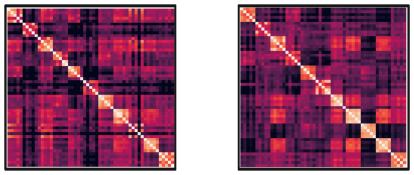

Figure 2: The correlation of Gram-CS channel weights for the 55 images in the query-dataset of the Oxford5K(left) and Paris6K(right). Images are sorted by landmark class.

where $\psi^{(k)} \in \mathbb{R}^C$ is k-th dense descriptor generated by k-th saliency mask $M^{(k)}$, and the i-th element of $\psi^{(k)}$ is computed as (4).

Concatenation. n selected C-dimensional regional representations are obtained by n saliency masks. We concatenate them into a single $(n \times C)$-dimensional vector to retain more information:

$$\Psi = [\psi^{(1)}, \psi^{(2)}, ..., \psi^{(n)}] \tag{5}$$

Projection. After performing l_2-normalization for the high dimensional representation, we adopt a linear projection process, i.e., a fully connected layer is first linked to make it into a D-dimensional descriptor. Then a second round l_2-normalization is conducted. There are two motivations for this projection step: (i) We need to synthesize the redundant information and reduce the dimensions. (ii) This process makes the feature space into $[-1 \sim 1]$ and we found it helps to generalize better and converge faster.

3.5 Post-processing

Since the work of (Jégou and Chum 2012), whitening for the final data representation has been proven to be very essential for image retrieval. Gordo et al. (Gordo et al. 2016) learns the whitening parameters of the CNN in an end-to-end manner. However, we found this method has little effect and even weakens the power of representation in our experiments. Here we still use the original PCA-Whitening (Jégou and Chum 2012), which is faster to learn and works well in practice.

4 Training

The proposed framework can be trained in an unsupervised manner or fine-tuned in a supervised way, in which all parameters are learned with only image-level labels. Therefore, we split the training process into two phases: the first phase is multiple saliency learning, and the second one is metric learning. In both stages, the loss function can be represented:

$$\mathcal{L}_{ms} = \mathcal{L}_S + \alpha \mathcal{L}_{NGD}, \tag{6}$$

where the $\mathcal{L}_{NGD}$ is a new loss function designed for multiple saliency learning, and $\mathcal{L}_S$ is decided by the training method, α is the balance parameter. In this section, we first introduce the new loss function $\mathcal{L}_{NGD}$ and then describe our training methods in details.

4.1 Normalized Gram-Difference Loss for Multiple Saliency Learning

Our goal for multiple saliency learning is to ensure that this module is able to focus on different regions of interest. To fulfil this objective, a new loss function is designed to meet two criteria: (i) It is semantic-aware, i.e., focuses on attentive deep features. (ii) It can enforce the output saliency masks to be different. To meet the first criteria, we opt to employ a softmax-based landmark classifier with a standard cross-entropy loss as the saliency block has been embed into it. For the second one, we propose a simple loss function named as **Normalized Gram-Difference Loss (NGD loss)**, which is also based on Gram matrix (see section 3.3).

As we use 1-layer CNN with sigmoid activation to obtain n saliency masks, they can be regarded as n attention maps or a 3D tensor with $n \times H \times W$ dimensions. Here we formulate the vectorised attention maps as $T = [m_1, m_2, ..., m_n]$. We first l_2-normalize each attention vector as follows:

$$\bar{T} = [\bar{m}_1, \bar{m}_2, ..., \bar{m}_n] = [\frac{m_1}{||m_1||_2}, \frac{m_2}{||m_2||_2}, ..., \frac{m_n}{||m_n||_2}] \tag{7}$$

Then we compute the Gram matrix $\bar{G} \in \mathbb{R}^{n \times n}$ of $\bar{T}$, where $\bar{G}_{ij}$ is the inner product between normalized mask vectors $\bar{m}_i$ and $\bar{m}_j$, describing the similarity between them:

$$\bar{G}_{ij} = \sum_k \bar{m}_{ik} \cdot \bar{m}_{jk} \tag{8}$$

We employ a mean operation to evaluate the average difference between different masks and define the Normalized Gram-Difference Loss as:

$$\mathcal{L}_{NGD} = \max\{\frac{1}{(n-1)^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \bar{G}_{ij} - \beta, 0\} \quad (i \neq j), \tag{9}$$

where β is the difference threshold.

4.2 Unsupervised Training with Instance Loss

Zheng et al. (Zheng et al. 2017) proposed the instance loss for instance-level image-text matching. Based on the assumption that each image/text group is distinct, they viewed each image/text group as a class. Inspired by this insight, we propose to ignore the actual label of each image and treat it as a single class. As no annotation information is used, it is actually in an unsupervised manner, marked as **UN**.

However, optimizing all parameters in the network will tend to over-fit and weaken the representation of the image. Therefore, we fix the parameters in the Basic Conv and only train the MSB and projection layer. To this end, we replace the $\mathcal{L}_S$ in (6) with the cross-entropy loss to train the" single class" classifier for total N images, i.e., the instance loss, which is given by:

$$\mathcal{L}_{ins} = -\zeta^* \cdot \log\left(\frac{\exp(\zeta)}{\mathbf{1}^T \exp(\zeta)}\right), \tag{10}$$

where ζ is a N-dimensional vector which is the predicted probability for each instance, ζ^* is the one-hot "ground-truth" and $\mathbf{1}$ is one vector.

4.3 Supervised Fine-tuning with Classification and Ranking Loss

For the supervised training manner, we split this task into two stages: (i) training the randomly-initialized modules with a classification loss (marked as **SU-STA1**) and (ii) fine-tuning the whole network with a ranking loss (marked as **SU-STA2**).

In the first stage, we fix the parameters in Basic Conv (like section 4.2) and train the rest parts by the cross-entropy loss with the actual labels.

In the second stage, we adopt a three-stream Siamese network in which the weights of three branches are shared. Contrast to the normal one, we use the triplet loss with batch hard mining (TriHard loss) (Hermans, Beyer, and Leibe 2017) instead. The core idea is to online choose the hard samples in the training batch.

For each image anchor(a) in this batch, we select the hardest positive sample (p) and negative one (n) to form the triplet images. That is, we define A as the same class set of a, while B is the set of rest ones. We replace the $\mathcal{L}_S$ in (6) with the TriHard loss $\mathcal{L}_{th}$ which is given by:

$$\mathcal{L}_{th} = \frac{1}{P \times K} \sum_{a \in batch} \{\max_{p \in A} d_{a,p} - \min_{n \in B} d_{a,n} + m\}_+, \quad (11)$$

where m is a scalar to control the margin and $\{z\}_+$ means $\max(0, z)$.

5 Experiments

In this section, we first describe test datasets and our training details. We then evaluate different components of our proposed method, and finally report the experimental results.

5.1 Test Datasets and Implementation

Test Datasets. We evaluate the performance of our method on five standard datasets. Most experiments are conducted on **Oxford5K** dataset (Philbin et al. 2007) and **Paris6K** dataset (Philbin et al. 2008), containing respectively 5063 and 6412 images. To test in a larger-scale scenario, we also consider **Oxford105K** and **Paris106K** datasets that are extended from Oxford5K and Paris6K with 100k distractor images (Philbin et al. 2007). Finally, we present results on the INRIA **Holidays** dataset (Jegou, Douze, and Schmid 2008), composing of 1491 images and 500 queries.

Evaluation protocol. In experiments, we measure the performance by mean average precision (mAP). We adopt the following evaluation protocol: for Oxford and Paris datasets, the annotated region of interest of the queries are used, while the whole image and the "upright" version(the original one) are adopted for Holidays .

Training Dataset. We employ a subset of clean landmarks dataset (LC) (Gordo et al. 2016), which officially contains about 49000 images from 586 landmarks. However, due to invalid URLs, only **7154** images from **325** landmarks are downloaded for training, and we mark it as sub-LC.

Implementation. All experiments are based on Pytorch toolbox(Paszke et al. 2017). For training process, we firstly resize the images to make the short edge range from 400 to 512 and then extract random crops with size of 386×386.The

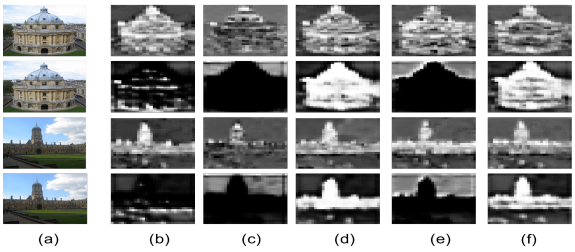

Figure 3: Visualization of the impact of NGD loss for multiple saliency learning ($n = 5$, $D = 512$) in SU-STA1 mode. (a) Two sample images in Oxford5K. (b)-(f) Different saliency mask outputs. For each image, the top is trained only by classification loss, the bottom combines the NGD loss.

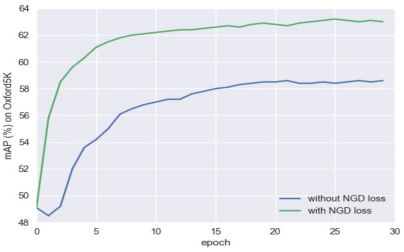

Figure 4: Evolution of the mAP on Oxford5K when training in SU-STA1 mode with/without NGD loss.

Basic Conv as shown in Fig.1 is pre-trained on the ImageNet ILSVRC challenge (Russakovsky et al. 2015). We use stochastic gradient descent (SGD) algorithm with momentum of 0.9 for training, while the learning rate is set as 0.01 for UN, 0.0005 for SU-STA1 and 0.0005 for SU-STA2 with weight decay of 0.0005. The balance parameter α is set as 0.005 for UN, 0.1 for SU-STA1 and 0.01 for SU-STA2. For NGD loss, β is set as 0.5. We online select samples with $P = 8$ and $K = 10$ for TriHard loss ($m = 0.1$).To save the memory of GPU, we only input $P \times 3$ images in each training pass. Running on a single GTX 1080 Ti GPU, a query image with 1024 pixel in the shorter edge will take 2.60ms for the entire network.

Multi-scale Feature Extraction and Whitening. During the multi-scale procedure for test, we use 4 scales with $[386, 550, 800, 1050]$ pixels shorter edge to extract the global descriptors. We find whitening on the sub-LC is ineffective, thus for a fair comparison with related works, we learn PCA-Whitening parameters on Oxford5K when testing on Paris6K and vice versa. We use the Oxford100K dataset for whitening on the Holidays.

5.2 Preliminary Experiments

Impact of NGD Loss. We visualize the multiple saliency masks training in SU-STA1 mode ($n = 5$, $D = 512$) to present the impact of NGD loss in Fig. 3. For learning without NGD loss, the activations for five masks are approximately the same, and all masks focus on the outline of the object

Table 1: mAP on Oxford5K and Paris6K varying different weighting schemes.

Method	**SU**	**Scale**	Oxford5K	Paris6K
SW + SSW(CroW)	no	-	70.8	79.7
SW + Gram-CS	no	-	71.4	81.5
SW + SSW(CroW)	yes	-	78.7	84.9
SW + Gram-CS	yes	-	80.6	87.1
MSB+Gram-CS	yes	single	81.5	86.5
MSB+Gram-CS	yes	multiple	**86.1**	**90.1**

Table 2: Comparison of different network architecture while SU means training in supervised manner and UN means training in unsupervised one.

Mask	Dim	Training	Oxf5K	Par6K	Holidays
1	512	SU	82.5	88.1	-
2	512	SU	83.8	88.2	-
5	512	SU	86.1	**90.1**	85.4
10	512	SU	83.0	87.4	-
5	1024	SU	**87.1**	88.2	87.6
10	1024	SU	86.7	88.4	-
10	2048	SU	85.7	88.8	-
5	512	UN	73.0	83.7	88.3
5	1024	UN	75.2	84.7	88.2
10	1024	UN	79.3	85.3	-
10	2048	UN	**80.5**	**86.7**	**90.2**

in the image. The outputs with NGD loss are significantly different. For instance, (d)(f) boost key object and suppress non-salient locations while (b)(e) focus some different details of images. Furthermore, compared with this situation without NGD loss, (d)(f) can pop out the salient instance much better. The results show the NGD loss can help the multiple saliency block to focus on different regions of interest. Fig. 4 exhibits the impact of NGD loss for the mAP on Oxford5K when learning in SU-STA1 mode (without post-processing). The impressive results demonstrate that NGD loss can speed up the convergence and obtain a better representation.

Architecture Discussion. In Table 1, we present the mAP on Oxford5K and Paris6K varying different weighting schemes. We set (Kalantidis, Mellina, and Osindero 2016) as our baseline and arrange our experiments with both off-the-shelf and fine-tuned network. In the first part of the table, to compare the channel weighting scheme, the spatial weighting(SW) is both the one in (Kalantidis, Mellina, and Osindero 2016) and the input images keep the original size for fair comparisons. The results reflect that Gram-CS channel weighting works better than SSW (Kalantidis, Mellina, and Osindero 2016) as our method considers the correlations between different filter responses. Moreover, we conduct the comparative experiments with the supervised manner ($n = 5$, $D = 512$) pre-trained by MSCNet. In the second part of the table, we present the results with Gram-CS block in which the effectiveness of this channel weighting scheme has been clearly indicated. As it is an embedded part of the whole network and thanks to the collected supervised information, the distinctive feature maps would be enhanced during the learning process so that fantastic performance is achieved. Also, we carry out the comparison of single-scale and multiple-scale representations, which proves that the multi-scale input makes remarkable improvements.

Design Discussion. Table 2 summarizes the mAP on Oxford5K and Paris6K varying the saliency mask number and the output descriptor dimension. For training in supervised manner, the mAP becomes higher with the number of saliency masks increasing within a certain range. The results achieve the best when the number of saliency mask is 5, and begins to reduce when it continues to increase, and the increasing of the final dimension of descriptor helps little for better representation. This is mainly caused by the limited supervised information. More complicated the model is, higher the possibility of over-fitting becomes. On the other hand, if the model is trained in unsupervised manner, the mAP will increase with the mask number and dimensionality.

Supervised vs Unsupervised. From Table 2, we can see that training in supervised mode will obtain much better mAP on landmarks evaluation datasets as the Basic Conv has been fine-tuned by metric learning. We also present mAP on Holidays when training in different manners to explore the generalization capability of the model. The supervised model for landmarks learning still works well on this scene search dataset, meanwhile, the unsupervised one with instance loss achieves fantastic result. This demonstrates that our unsupervised method generalize well for similar retrieval tasks.

5.3 Comparison to State-of-the-art Methods

Finally, Table 3 summarizes the performance of our best models and the state-of-the-art works, and all methods are based on VGG16 (Simonyan and Zisserman 2014) network for image retrieval. The average query expansion (QE) (Chum et al. 2007) is also considered as it has recently become a standard policy for CNN global image representation. In the first part of the table, we compare our approach with other methods that obtain descriptors in an unsupervised manner. The proposed unsupervised method outperforms the state-of-the-art methods algorithms on all datasets whether they contain QE or not. The results demonstrate that the instance loss is effective to extract salient information and learn a good linear projection for image search so that discriminative representation can be obtained after post-processing.

In the second part of Table 3, we exhibit the results compared with the state-of-the-art supervised methods. Without performing query expansion, we are comparable with most datasets and only slightly lower than (Radenović, Tolias, and Chum 2018) on Oxford5K. Especially for the Oxford105K and Paris106K datasets, we improve a lot with our 1024-dimensional model. This indicates that our method is robust for large-scale datasets with serious noises. After query expansion, our 512-dimensional representation outperforms the others except (Radenović, Tolias, and Chum 2018) on Oxford5K and Oxford105K. Considering all the above comparison results, we can conclude that focusing on multiple saliency and channel sensitivity is crucial and using features fine-tuned by metric learning for the particular task is quite helpful to obtain better representation.

In Table 4, we present the mAP compared with the state-of-the-art ResNet (He et al. 2016) based methods. We out-

Table 3: Performance comparison with state-of-the-art CNN (VGG16) based retrieval methods. SU: Use of the supervised information (yes), otherwise (no). Dim: Dimensionality of the final compact image representation. Our method are marked with $\star$ and the architecture information is also presented. We do not report QE results on Holidays as it is not a standard practice.

Method	**Dim**	**SU**	Oxf5K	Par6K	Oxf105K	Par106K	Holidays
SPoC (Babenko and Lempitsky 2015)	256	no	53.1	-	50.1	-	80.2
CroW (Kalantidis, Mellina, and Osindero 2016)	512	no	70.8	79.7	65.3	72.2	85.1
BOW-CNN (Mohedano et al. 2016)	25k	no	73.8	82.0	59.3	64.8	-
R-MAC (Tolias, Sicre, and Jégou 2016)	512	no	66.9	83.0	61.6	75.7	-
CAMOfA (Jimenez, Alvarez, and Giro-i Nieto 2017)	512	no	71.2	80.5	67.2	73.3	-
PWA (Xu et al. 2018)	4096	no	79.1	86.1	73.6	80.4	-
$\star$ MSCNet (5-512)	512	no	73.0	83.7	69.3	76.8	88.3
$\star$ MSCNet (10-2048)	2048	no	**80.5**	**86.7**	**77.5**	**80.8**	**90.2**
CroW+QE (Kalantidis, Mellina, and Osindero 2016)	512	no	74.9	84.8	70.6	79.4	-
R-MAC+AML+QE (Tolias, Sicre, and Jégou 2016)	512	no	77.3	86.5	73.2	79.8	-
CAMOfA+R+QE (Jimenez, Alvarez, and Giro-i Nieto 2017)	512	no	80.1	85.5	76.9	80.0	-
PWA+QE (Xu et al. 2018)	4096	no	81.7	89.2	80.6	84.7	-
$\star$ MSCNet+QE (10-2048)	2048	no	**85.9**	**89.5**	**83.9**	**85.0**	-
NetVLAD (Arandjelovic et al. 2016)	4096	yes	71.6	79.7	-	-	83.1
siaFV (Ong, Husain, and Bober 2017)	512	yes	81.5	82.4	76.6	-	-
siaMAC (Radenović, Tolias, and Chum 2016)	512	yes	79.7	83.8	73.9	76.4	82.5
R-MAC (Gordo et al. 2016)	512	yes	83.1	87.1	78.6	79.7	86.7
GeM(Radenović, Tolias, and Chum 2018)	512	yes	**87.9**	87.7	83.3	81.3	-
$\star$ MSCNet (5-512)	512	yes	86.1	**90.1**	82.7	82.1	85.4
$\star$ MSCNet (5-1024)	1024	yes	87.1	88.2	**84.6**	**82.7**	**87.3**
siaMAC+QE (Radenović, Tolias, and Chum 2016)	512	yes	85.0	86.5	81.8	78.8	-
R-MAC+QE (Gordo et al. 2016)	512	yes	89.1	91.2	87.3	86.8	-
GeM+αQE(Radenović, Tolias, and Chum 2018)	512	yes	**91.9**	91.9	**89.6**	87.6	-
$\star$ MSCNet+QE (5-512)	512	yes	88.7	**93.1**	87.3	**87.8**	-
$\star$ MSCNet+QE (5-1024)	1024	yes	90.6	90.9	89.3	86.9	-

Table 4: Comparison with state-of-the-art CNN (ResNet) based retrieval methods.

Method	**Dim**	**SU**	Oxf5K	Par6K	Oxf105K	Par106K	Holidays
R-MAC (Gordo et al. 2017)	2048	yes	86.1	**94.5**	82.8	**90.6**	90.3
DELF+ATT (Noh et al. 2017)	-	yes	83.8	85.0	82.6	81.7	-
GeM(Radenović, Tolias, and Chum 2018)	2048	yes	87.8	92.7	84.6	86.9	-
$\star$ MSCNet (5-2048)	2048	no	76.0	86.7	72.0	81.5	89.9
$\star$ MSCNet (5-2048)	2048	yes	**88.3**	91.6	**85.9**	86.3	**91.4**

perform them with our supervised model on Oxford5K, Oxford105K, while (Gordo et al. 2017) achieves better on Paris. However, we only use approximate one seventh of their training data and generalize better on other particular task (Holidays), which is more meaningful for image retrieval.

6 Conclusions

In this paper, we present an approach based on multiple saliency and Gram matrix based channel sensitivity (MSCNet) to address two key problems of instance-level image retrieval, i.e., the distinctiveness of image representation and the generalization ability of the model. To this end, we propose a multiple saliency block to focus on different salient parts of the image, and a Gram matrix based channel sensitivity block to intensify distinctive features. Our novel architecture can be trained or fine-tuned with an unsupervised instance loss, and a classification loss combined with ranking loss on very limited training data. With this approach, we report results that outperform the state-of-the-art methods.

7 Acknowledgments

This work is supported by Chinese National Natural Science Foundation(61532018, 61471049, 61372169).

References

Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; and Sivic, J. 2016. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5297–5307.

Babenko, A., and Lempitsky, V. 2015. Aggregating local deep features for image retrieval. In *Proceedings of the IEEE international conference on computer vision*, 1269–1277.

Babenko, A.; Slesarev, A.; Chigorin, A.; and Lempitsky, V. 2014. Neural codes for image retrieval. In *European conference on computer vision*, 584–599. Springer.

Chum, O.; Philbin, J.; Sivic, J.; Isard, M.; and Zisserman, A. 2007. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 1–8. IEEE.

Gatys, L.; Ecker, A. S.; and Bethge, M. 2015. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems*, 262–270.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2016. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2414–2423.

Gong, Y.; Wang, L.; Guo, R.; and Lazebnik, S. 2014. Multi-scale orderless pooling of deep convolutional activation features. In *European conference on computer vision*, 392–407. Springer.

Gordo, A.; Almazán, J.; Revaud, J.; and Larlus, D. 2016. Deep image retrieval: Learning global representations for image search. In *European Conference on Computer Vision*, 241–257. Springer.

Gordo, A.; Almazan, J.; Revaud, J.; and Larlus, D. 2017. End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision* 124(2):237–254.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hermans, A.; Beyer, L.; and Leibe, B. 2017. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*.

Hu, J.; Shen, L.; and Sun, G. 2017. Squeeze-and-excitation networks. *arXiv preprint arXiv:1709.01507*.

Jégou, H., and Chum, O. 2012. Negative evidences and cooccurences in image retrieval: The benefit of pca and whitening. *Computer Vision–ECCV 2012* 774–787.

Jegou, H.; Douze, M.; and Schmid, C. 2008. Hamming embedding and weak geometric consistency for large scale image search. *Computer Vision–ECCV 2008* 304–317.

Jimenez, A.; Alvarez, J. M.; and Giro-i Nieto, X. 2017. Class-weighted convolutional features for visual instance search. In *BMVC*.

Kalantidis, Y.; Mellina, C.; and Osindero, S. 2016. Cross-dimensional weighting for aggregated deep convolutional features. In *European Conference on Computer Vision*, 685–701. Springer.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 1097–1105.

Lowe, D. G. 2001. Local feature view clustering for 3d object recognition. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, I–I. IEEE.

Mohedano, E.; McGuinness, K.; O'Connor, N. E.; Salvador, A.; Marqués, F.; and Giró-i Nieto, X. 2016. Bags of local convolutional features for scalable instance search. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, 327–331. ACM.

Nair, V., and Hinton, G. E. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, 807–814.

Noh, H.; Araujo, A.; Sim, J.; Weyand, T.; and Han, B. 2017. Large-scale image retrieval with attentive deep local features. In *The IEEE International Conference on Computer Vision (ICCV)*.

Ong, E.-J.; Husain, S.; and Bober, M. 2017. Siamese network of deep fisher-vector descriptors for image retrieval. *arXiv preprint arXiv:1702.00338*.

Paszke, A.; Gross, S.; Chintala, S.; and Chanan, G. 2017. Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration.

Philbin, J.; Chum, O.; Isard, M.; Sivic, J.; and Zisserman, A. 2007. Object retrieval with large vocabularies and fast spatial matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 1–8. IEEE.

Philbin, J.; Chum, O.; Isard, M.; Sivic, J.; and Zisserman, A. 2008. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1–8. IEEE.

Radenović, F.; Tolias, G.; and Chum, O. 2016. Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In *European Conference on Computer Vision*, 3–20. Springer.

Radenović, F.; Tolias, G.; and Chum, O. 2018. Fine-tuning cnn image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115(3):211–252.

Seddati, O.; Dupont, S.; Mahmoudi, S.; and Parian, M. 2017. Towards good practices for image retrieval based on cnn features. In *The IEEE International Conference on Computer Vision (ICCV)*.

Siméoni, O.; Iscen, A.; Tolias, G.; Avrithis, Y.; and Chum, O. 2017. Unsupervised deep object discovery for instance recognition. *arXiv preprint arXiv:1709.04725*.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Tolias, G.; Sicre, R.; and Jégou, H. 2016. Particular object retrieval with integral max-pooling of cnn activations. In *ICLR*.

Xu, J.; Shi, C.; Qi, C.; Wang, C.; and Xiao, B. 2018. Unsupervised part-based weighting aggregation of deep convolutional features for image retrieval. In *AAAI*.

Zheng, Z.; Zheng, L.; Garrett, M.; Yang, Y.; and Shen, Y.-D. 2017. Dual-path convolutional image-text embedding. *arXiv preprint arXiv:1711.05535*.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2921–2929.