Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 9794202, 12 pages
https://doi.org/10.1155/2019/9794202

Research Article

Multilayer Convolutional Feature Aggregation Algorithm for Image Retrieval

Rongsheng Dong, Ming Liu, and Fengying Li

Guangxi Key Laboratory of Trusted Software, School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, China

Correspondence should be addressed to Fengying Li; lfy@guet.edu.cn

Received 26 February 2019; Revised 9 May 2019; Accepted 2 June 2019; Published 26 June 2019

Academic Editor: A. M. Bastos Pereira

In image retrieval tasks, the single-layer convolutional feature has insufficient image semantic representation ability. A new image description algorithm ML-RCroW based on multilayer multiregion cross-weighted aggregational deep convolutional features is proposed. First, the ML-RCroW algorithm inputs an image into the VGG16 (a deep convolutional neural network developed by researchers at Visual Geometry Group and Google DeepMind) network model in which the fully connected layer is discarded. The visual feature information in the convolutional neural network (CNN) is extracted, and the target response weight map is generated by combining with the spatial weighting algorithm of the target fuzzy marker. Then, visual features in the CNN are divided into multiple regions, and the pixels of each region are weighted by regional spatial weight, regional channel weight, and regional weight. The image global vector is generated by aggregating and encoding every region in the weighted feature map. Finally, features of each layer of the VGG16 network model are extracted and then aggregated and dimensionally reduced to obtain the final feature vector of the image. The experiments are carried out on the Oxford5k and Paris6k datasets provided by Oxford VGG. The experimental results show that the average accuracy of image retrieval based on the image feature description algorithm ML-RCroW is better than that achieved by the other commonly used algorithms such as SPoC, R-MAC, and CroW.

1. Introduction

Content-based image retrieval is one of the most active research topics in the field of computer vision and has attracted increasing attention in theoretical and practical research [1]. Although content-based image retrieval (CBIR) technology has been widely used in the past few decades, creating a compact and discriminative descriptor for illumination variation, shape deformation, and background clutter is still a challenging problem [2]. With the development of deep learning technology, convolutional neural networks (CNNs) have made a series of breakthroughs in a wide range of computer vision applications, such as image classification [3, 4], target detection [5], semantic segmentation [6], and visual Q&A [7].

Early CNN-based image retrieval algorithms used the activation value of the fully connected layer as the global feature description vector. Razavian et al. [8] studied the CNN model trained by the ImageNet dataset as a black box feature extractor for image retrieval and achieved considerable results. Babenko et al. [9] proposed a neural code algorithm that uses principal component analysis (PCA) to compress depth features, showing that the visual features extracted by CNN are superior to traditional scale-invariant feature transform (SIFT) compact descriptors. Gong et al. [10] introduced multiscale orderless pooling (MOP), combining the all-connected layer features extracted in CNN with the unordered vector of the local aggregated descriptors (VLAD) encoding algorithm [11], which is better than previous algorithms.

Recently, the popularity of image retrieval research has shifted from the fully connected layer of CNN to the convolutional layer [12]. Compared with the fully connected layer feature with a global receptive field, features extracted by the convolutional layer retain more local features and spatial information of the image. The convolutional layer features are

more robust to image transformation and are therefore more suitable for image retrieval. Babenko et al. [13] aggregated convolutions using sum-pooling algorithm (sum-pooling) layer features, resulting in a more compact global image representation, and further demonstrated that the aggregated deep convolutional feature algorithm is superior to shallow features (such as SIFT) due to its highly differentiated features and different distribution characteristics. Kalantidis et al. [14] extended the summation pooling algorithm by the cross-weighting algorithm (CroW), after extracting the deep convolution feature from the last layer of CNN, each spatial location and each channel are weighted, and finally the final aggregation result is obtained by sum-pooling, which further enhanced image feature representation ability. Simultaneously, Tolias et al. [15] proposed an *R-MAC* algorithm that divides multiple image areas of different sizes directly on the convolutional layer and then performs maximum pooling. This algorithm does not require multiple MOP-CNN algorithms to input images of different regions to the network multiple times. The feature representation of multiple regions of similar images improves the performance of image retrieval. However, the *R-MAC* algorithm does not take into account the influence of background and image chaos and give these areas the same weights as the target, which affects the representation of image features. Dong et al. [16] proposed a multiregion cross-weighted aggregation deep convolution feature description algorithm-RCroW, which effectively utilized the spatial weight and the channel weight of the image to improve the retrieval accuracy.

Recent research has also shown that multiscale convolutional features play an important role in the field of computer vision. Li et al. [17] used five hierarchical convolutional feature maps for target tracking. Liu et al. [18] proposed using multiscale convolutional feature maps to deal with target detection problems. Ng et al. [19] proposed a method for extracting convolutional feature maps from different network layers and for using VLAD to encode features for image retrieval. However, how to use multiscale convolutional feature maps to improve retrieval accuracy is still an open question. In this paper, a multilayer multiregion cross-weighted matrix aggregation deep convolutional feature (ML-RCroW) algorithm is proposed for image retrieval based on the literature [16]. The contributions are summarized as follows.

First, this paper considers the influence of background and noise region when generating spatial weights and proposes a spatial weighting algorithm combining target fuzzy markers with weighting strategies. The image target area is well preserved and the noise area is discarded, further improving the performance of the CNN visual features.

Second, the influence of region weight in the regional strategy is considered, and an algorithm for weighting the region is proposed that mitigates the effects of repeated summation of image chaos and background-generated responses due to equal region weights. Moreover, it combines the spatial weighting algorithm to assign weights to feature maps in multiple regions to generate image feature vectors.

Finally, the different levels of aggregation features in CNN are connected, and multiscale features are aggregated into a vector for image retrieval by using the complementary advantages of different levels. Experiments on the Oxford5k and Paris6k datasets show that, compared with algorithms such as R-MAC and CroW, the proposed algorithm is competitive in terms of accuracy and effectiveness.

2. Related Work

Early image retrieval algorithms used the SIFT feature, which focused on generating image representations using local feature descriptors and aggregation strategies to describe the features of an image, for example, visual bag-of-words (BoW) and variants of various word-package models, such as VLAD, Fisher vector, and triangulation embedding. Furthermore, Yousuf et al. improved the performance of CBIR on the basis of visual words fusion of the SIFT and local intensity order pattern (LIOP) descriptors [20].

Recently, the application of multiscale feature aggregation algorithms in computer vision tasks has been shown to bring about effective improvements. One of the most common examples of using multiscale features in traditional image algorithms is spatial pyramid matching (SPM) [21], which improves image retrieval performance by aggregating a single long vector by multi-image BoW vectors of an image size. In work related to CNNs in multiscale features, the most common method directly uses multiscale image input to a CNN and extracts multiple convolutional layer features. Gong et al. [10] extracted CNN features directly from multiscale images in image retrieval and classification tasks. Inspired by the success of image classification, CNN's feature extractor has also been applied to other recognition tasks. For example, such feature extractors and all relevant network parameters have been published in a study [22]. Moreover, this study emphasizes that since the lower layers are unlikely to contain rich semantic information, feature vectors from the last three convolutional layers are mainly evaluated. Since lower layers are unlikely to contain rich semantic information, the main evaluation relies on the last three convolutional layer feature extractors. Some pixel tagging tasks attempt to enrich pixel representations by concatenating semantic information by linking feature extractors from different layers (e.g., object segmentation [23] and boundary detection [24]). These tasks combine the characteristics of the upper and lower levels and have achieved good results.

Z. Mehmood et al. [25] proposed combining a histogram of global visual words of the entire image with a histogram of visual words on a partial rectangular region of the image to generate a feature representation of the image. The local histogram of constructing visual words on the rectangular region of the image proposed by the method can effectively utilize the spatial information of the BoVM model. N. Ali et al. [26] proposed a novel visual words integration of SIFT and speeded-up robust features (SURF). SIFT features are more robust to image scale and rotation, and SURF features are more robust to illumination variations. The image features obtained by this method are capable of producing good retrieval performance and have the advantages of fast indexing and scalability. Z. Mehmood et al. [27] proposed a new method for speech separation of unmarked static noise

audio signals using the deep trusted network (DBN) model. The method can effectively separate the music signal from the noisy audio stream, remove the static noise using the hidden layer separation model of the recurrent neural network (RNN), and then use the dictionary-based Fisher algorithm for speech classification. K.A.Qazi et al. [28] proposed two novel methods, VMIaC and FibC, to solve the semantic gap between image semantic concepts and local features of images in machine learning-based image retrieval. These two methods effectively utilize the complementarity of visual vocabulary and the complementarity of image features to improve retrieval performance.

Yoo et al. [29] proposed a method of inputting different scale images into a CNN and extracting the output features of fully connected layers using the Fisher vector algorithm to generate a single feature vector. Farabet et al. [30] cascaded CNN visual features of different scale images in the same CNN layer to generate a single feature vector.

Considering that the pretraining model of the target classification task is directly used as the feature extractor, there is a poor generalization ability for recognizing the intraclass target. Therefore, researchers have proposed some end-to-end methods to train model weights to make the model more suitable. Regarding retrieval tasks, currently, the most prominent technique in the end-to-end network model of image retrieval tasks is the image representation task generated by Gordo et al. [31] based on twin networks. Gordo et al. used a more powerful ResNet101 network model as a feature extractor in feature extraction and aggregation algorithms, and proposed a direct addition and aggregation of the three descriptors, using L2 normalization to generate feature vectors. The use of the ResNet101 model and multiresolution is very simple and offers significant improvements without increasing the descriptor length. Inspired by the algorithm proposed by Gordo et al., this landmark dataset [9] is used to fine-tune the VGG16 [14] model.

In summary, the CroW algorithm does not take into account the effects of background or noise regions when generating spatial weights. Under certain conditions, these regions also have a high response in the convolution feature map, and weighting these image parts will reduce the description ability of image features. At the same time, the R-MAC algorithm does not consider the region weight influence on the regional strategy and adds noisy data and useless background information to the feature vector without judgment. In addition, these two algorithms both extract features of the last convolutional layer as feature vectors of the image, disregard low-level features of the image, and lack some fine-grained information [32] of the image. To solve or alleviate these problems, this paper proposes an algorithm named ML-RCroW for image retrieval.

3. ML-RCroW Feature Description Algorithm

The principles and processes of the ML-RCroW algorithm are described in detail in this section. First, the network framework of convolutional visual feature extraction and description of the input image are introduced (Section 3.1). Then, the generation of image target response weights is introduced, including the specific generation process of spatial weight S and the introduction of channel weights C (Section 3.2). Finally, calculation of the regional weight w_i for multiple regions and the generation of final feature vector $\boldsymbol{MF}$ are introduced.

3.1. Convolutional Visual Feature Extraction. The CNN visual feature is extracted using a pretrained VGG model with a depth of 16 as the basic architecture. All the fully connected layers of the VGG16 network model are discarded, and the input of the original image size is kept free of the constraints of the network model on the size of the input image.

Let I denote an input image that has $M \times N$ pixels. The VGG16 model is used to extract the excitation response of the input layer in the layer l of the network, which is recorded as the feature map $X^l \in R^{W\times H\times K}$. The output of convolutional feature maps X^l can be arranged into a tensor with $W \times H \times K$ dimensions, where H and W denote the height and width of each feature map, and K denotes the number of feature maps in that layer. In addition, the extracted feature maps are passed through a rectified linear unit (ReLU) activation function to ensure nonnegative activation.

3.2. Target Response Weight. As show in Figure 1, the feature map $M = \sum_{k=1}^{K} X_k^l$ can be obtained by adding X_k^l, where k=1, 2,. . ., K and $M \in R^{W\times H}$. There is a summation activation response corresponding to $W \times H$ feature maps in the feature map $\boldsymbol{M}$. Experiments conducted in a study [33] have demonstrated that the higher the activation response of a particular position (i, j) is (corresponding to point P in Figure 1), the greater the likelihood that the corresponding region is the target. Therefore, the average value $\overline{M}$ from the activation response to all locations in $\boldsymbol{M}$ is determined. With $\overline{M}$ as the threshold, fuzzy marker determines which spatial positions correspond to the target. The position (i, j) where the activation response is higher than $\overline{M}$ is the possible position of the target, and the mask map for $\boldsymbol{M}$ is calculated:

$$\widehat{M}_{i,j} = \begin{cases} 1, & \text{if } M_{i,j} > \overline{M} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where (i, j) corresponds to a specific position in $\boldsymbol{M}$.

As shown in Figure 2, $\widehat{\mathrm{M}}$ detected the contour area of the target, but apparently $\widehat{\mathrm{M}}$ lacked some details and mixed some noise areas. For example, in the middle column of the second and third rows, compared with the original image, some details are lost in $\widehat{\mathrm{M}}$, and $\widehat{\mathrm{M}}$ in the middle column of the first row contains noise regions on the left and right. Therefore, some processing is required for $\widehat{\mathrm{M}}$, repairing important detail areas and removing suspected noise areas.

$\widehat{\mathrm{M}}$ is essentially a binary image. In this paper, mask strategy is used to blur the target position. The mask operation proceeds as follows: mark the connected area on the binary image, select the largest connected area, and delete other scattered and small areas. After obtaining the maximum connected area, the convex hull of $\widehat{\mathrm{M}}$ is determined to ensure that a larger part of the target area is in the connected area.

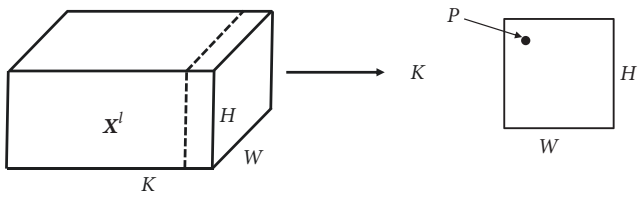

Figure 1: Schematic diagram of feature map conversion.

The modified $\widehat{M}$ (i.e., the last column of the image in Figure 3) marks the target area more accurately than the original $\widehat{M}$ and discards the noise area. Compared with the CroW algorithm, the convolution feature map is directly summed to obtain the spatial weighted graph. This algorithm highlights the target region, similar to preprocessing the target positioning, and the target and noise areas are effectively segmented.

After $\widehat{M}$ blurring marks the target area, the area outside the target area of the image is simply marked as 0, obviously discarding the scene information that is not conducive to image retrieval. To further increase the distinguishing ability of image feature vectors, this paper combines the mask map $\widehat{M}$ after the largest connected region with the feature graph $\boldsymbol{M}$ to generate the final spatial weight. First, let the feature aggregation graph $\boldsymbol{M}$ subtract the average value $\overline{M}$, and obtain a new feature map with negative, zero, or positive numbers, denoted as $\widetilde{M}$. The positive part of $\widetilde{M}$ is identified as the response of the target area, and the negative part of $\widetilde{M}$ is identified as the response of the noise or background area. Then, use the *sigmoid* function value (as shown in Figure 3) from 0 to 1 and the increasing characteristics, giving a small weight to the negative part of $\widetilde{M}$. The positive weighting part of $\widetilde{M}$ provides a large weight weighting strategy, the weighting part of the negative part and the target fuzzy estimation area is directly given the weight 1/2. Finally, spatial weight map $\boldsymbol{S}$ is calculated by the weighting strategy, wherein the value of each spatial position is expressed as (i, j):

$$S_{i,j} = \begin{cases} \dfrac{1}{2}, & if\ \widehat{M}_{i,j} = 1\ and\ \widetilde{M} < 0 \\ sigmoid\left(\widetilde{M}_{i,j}\right), & otherwise \end{cases} \tag{2}$$

where $S \in R^{W \times H}$,$sigmoid(x) = 1/(1 + e^{-x})$.

After obtaining the two-dimensional spatial weight map X^l of the feature map S, it can be applied to each channel independently, and a new feature map with improved weights for each layer is obtained; that is, the feature is reaggregated by $X_k^{l'} = X_k^l \times S$.

Using the obtained spatial weight S and the channel weights mentioned in [14], the channel weight is denoted as C, where $\in [1, K]$, weighted feature vectors $X_k^{l'}$ are generated:

$$X_{kij}^{l\ \prime} = S_{i,j} \times C_k \times X_{kij}^l \tag{3}$$

The feature vector of each spatial position is generated by sum-pooling to generate a feature vector $F_l = \{f_1, \ldots, f_k\}$:

$$f_k = \sum_{i=1}^{W} \sum_{j=1}^{H} X_{kij}^{l\ \prime} \tag{4}$$

3.3. Feature Description under Multiple Regions. Multiregion structure generation is similar to the R-MAC algorithm. Given the feature map $\boldsymbol{X}^l$, the area of width $2\min(W, H)/(L+1)$ is uniformly sampled on each scale, while the area of each scale is allowed an approximate 40% overlap between successive areas. Moreover, $L = 1,2,3$ is shown in Figure 4, where three sizes of window sliding feature maps $\boldsymbol{X}^l$ are used, respectively.

In this paper, based on the multiregion generated by R-MAC, the spatial weight S of the target fuzzy marker is reused, and the maximum normalization operation for S is limited to the range of [0, 1], generating corresponding multiregion weights, alleviating the effects of repeated superimposed image chaos and background generation caused by equal region weights. A new feature description vector is generated by replacing the convolution maximum activation (MAC) method in the R-MAC with a cross-weighting method. In this process, it is necessary to calculate the regional spatial weight S_{R_i}, the regional channel weight C and the regional weight C_k, where C_k is calculated using the channel weights proposed in [14].

Calculating the spatial weight S_{R_i} under multiple regions: in Section 3.2, the spatial weight map $\boldsymbol{S}$ of the entire feature map $\boldsymbol{X}^l$ is obtained, and the region S_{R_i} can be directly divided according to the region division.

Calculating the area weights under multiple regions w_i: in this paper, the known $\boldsymbol{S}$ is taken as a saliency map and is denoted as $\boldsymbol{A}$. Then, maximize the normalization of the saliency maps so that each element p has a range such that, $A_{p\in}[0, 1]$. In the *R-MAC* algorithm, the region R generated by the rigid network depends on the spatial dimension of the feature map $\boldsymbol{X}^l$. In the R-MAC algorithm, the region R generated by the rigid network depends on the spatial dimension of the feature map $\boldsymbol{X}^l$. Since the spatial dimension of the feature map remains in the saliency map A, the same region R can be defined on the saliency map. Then, for each region R, calculate the weight w_i

$$w_i = \max_{p \in R_i} A_P \tag{5}$$

and generate the new feature vector of the image

$$\begin{aligned} F_j &= \sum_{i=1}^{N} w_i f_{R_i} \\ &= \left[\sum_{i=1}^{N} w_1 f_{R_{i,1}} \cdots \sum_{i=l}^{N} w_k f_{R_{i,k}} \cdots \sum_{i=1}^{N} w_K f_{R_{i,K}} \right]^T \end{aligned} \tag{6}$$

where $f_{R_{i,k}} = S_{R_i} \times C_k \times X_{R_{i,k}}^l$, j is the number of layers of the convolutional feature map, F_j is the sum-pooling of the convolutional features, N is the number of divided regions, and the final dimension of F_j is equal to the number of feature channels K. This article extracts features from multiple convolutional layers instead of using the final F_j features for image retrieval. Using a simple method, this paper connects multiple F_j features with different layers of different resolutions into one vector. Figure 5 shows an illustration of

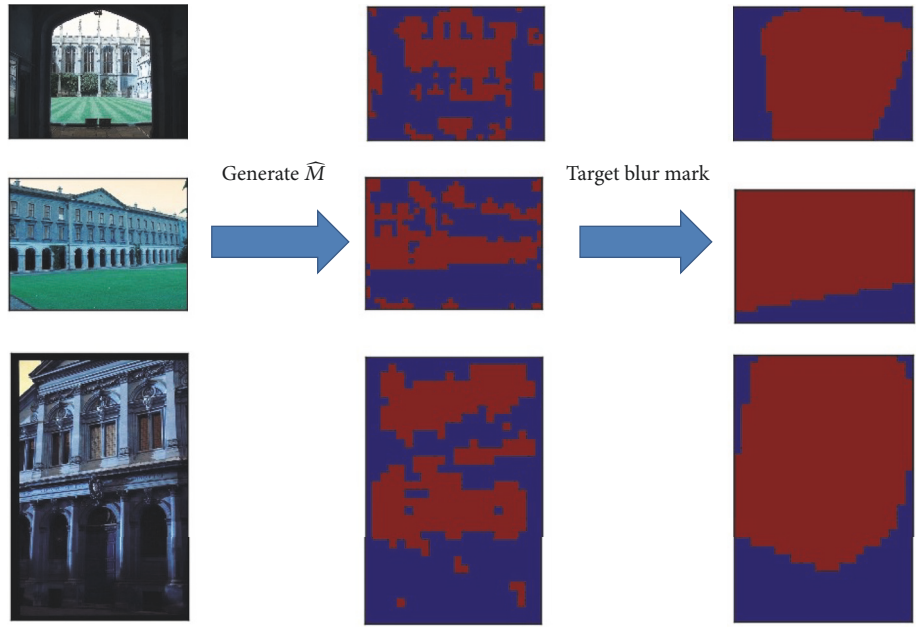

Figure 2: Feature map processing.

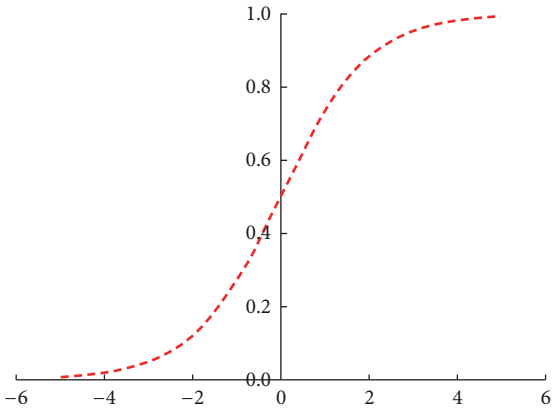

Figure 3: Sigmoid function diagram.

the ML-RCroW feature extraction process. Finally, the ML-RCroW feature can be expressed as

$$MF = \left[F_1 \ldots F_j \ldots F_L\right] \tag{7}$$

4. Datasets and Assessment Methods

This experiment evaluated the performance of the ML-RCroW algorithm on the Oxford 5k [33] and Paris6k [34] benchmark datasets provided by the Oxford VGG group.

The Oxford5k dataset consists of 5,062 images, collected from the Flickr dataset, and corresponds to 11 famous landmarks at the University of Oxford. Among them, 55 query images are retained on 11 landmarks, and annotation information is provided for evaluating search results for 55 retained query images. The benchmark files divide all the search results into 4 categories: namely, good, ok, junk and bad, of which good indicates that all the target objects appear in the result, ok indicates that the target object appears in more than 25% of the images, junk indicates that the target object appears in less than 25% of the images, and bad indicates that the target object does not appear. During the evaluation period, only the good and ok categories in the search results are used as the correct result (correlation image), bad is used as the error result, and junk is ignored (the images marked as junk do not affect the evaluation result).

The Paris6k dataset consists of 6,392 images, which correspond to famous landmarks in Paris. Similar to the Oxford5k dataset, 55 images were retained as query images, but since the datasets were collected from the Internet, the shooting environment corresponding to the same landmarks varies greatly. Each landmark's query image has a corresponding benchmark file, and the benchmark file's taxonomy is consistent with the Oxford5k dataset.

The image retrieval experiments on Oxford5k and Paris6k currently use the mean average precision (mAP) of the search images as the evaluation index. The higher the mAP, the better the performance. The two parameters involved in calculating the mAP, namely, the recall rate and the precision rate, are defined as follows:

$$r\left(ij\right) = \frac{x}{y} \times 100\% \tag{8}$$

$$p\left(ij\right) = \frac{x}{j} \times 100\% \tag{9}$$

The first query image indicates the total number of images returned by the search and the number of all related images. Referring to the percentage of correct search results with respect to the total number of correct results in the image library, the percentage of correct results retrieved is given as a percentage of the total results retrieved. The evaluation index of image retrieval performance generally includes the

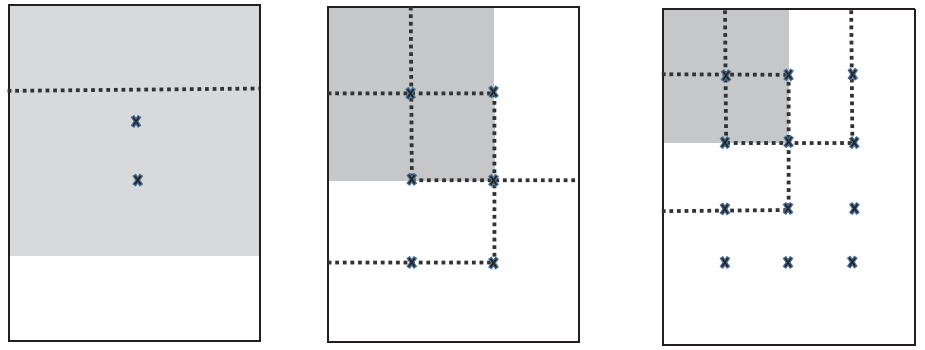

Figure 4: Sampling different regions on the feature map at different scales.

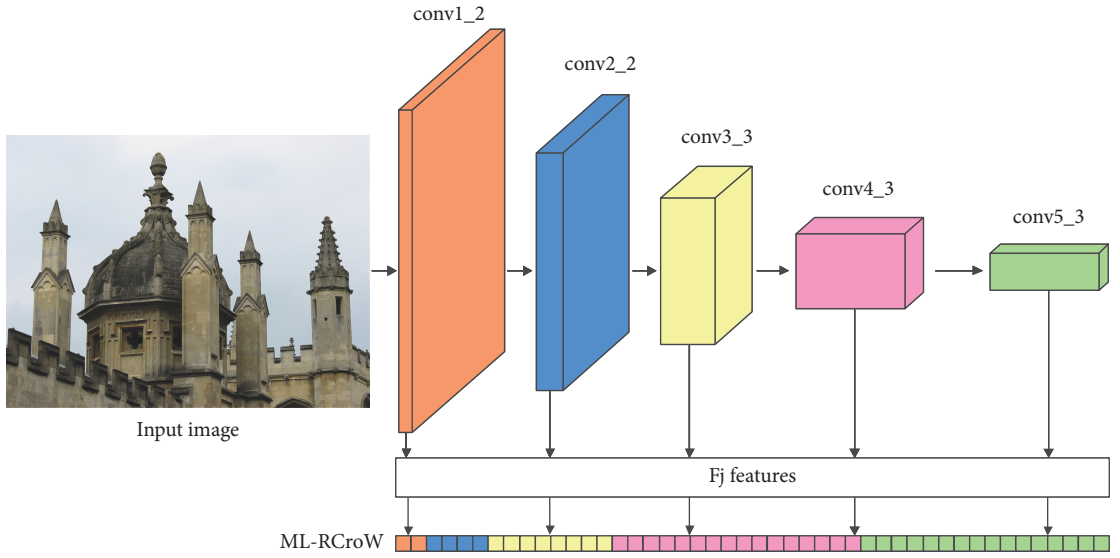

Figure 5: ML-RCroW algorithm extraction and encoding process.

precision and recall rates. The average precision (AP) of the image is calculated as

$$AP(i) = \sum_{j=1}^{n} p(ij)\,\Delta r(ij) \tag{10}$$

where $\Delta r(ij)$ is the change value of the recall rate when the number of returned images increases from *j-1* to *j*, and n is the total number of images in the image library. The average of the AP of the image dataset is calculated as

$$mAP = \frac{\sum_{i=1}^{m} AP(i)}{m} \tag{11}$$

where n is the sum of the images in the image dataset.

5. Results and Analysis

The experiment was divided into two parts. The first part of the experiment was based on the VGG16 network model for multilevel cascade experiments. The second part of the experiment was based on the fine-tuned VGG16 network model for multilevel cascade experiments.

Figure 6 shows the proposed block diagram, which illustrates the flow of the entire image retrieval algorithm by taking two layers of ML-RCroW features, namely, Pool4 and Pool5 layer feature aggregation as an example. For the training dataset, the pretrained VGG16 network is used to extract the characteristics of Pool4 and Pool5, and 512-dimensional feature vectors are obtained. Then, the Pool4 layer feature and Pool5 layer feature are aggregated to obtain a 1024-dimensional feature vector, and finally, the dimensionality reduction operation is performed. For the query map, features of the Pool4 and Pool5 layers are also extracted, and the final feature vector of the query map is obtained by aggregation reduction. Finally, the mAP value is used as an evaluation index to measure the similarity between the query image and image in the database.

The first part of the experiment used VGG16 as the network model for feature extraction. Experiments were performed on the Oxford5k and Paris6k datasets to verify the validity of the ML-RCroW algorithm. First, the experiment maintained the size of the original image of the Oxford5k and Paris6k datasets without any processing. Then, the experiment used the pretrained VGG16 network model to

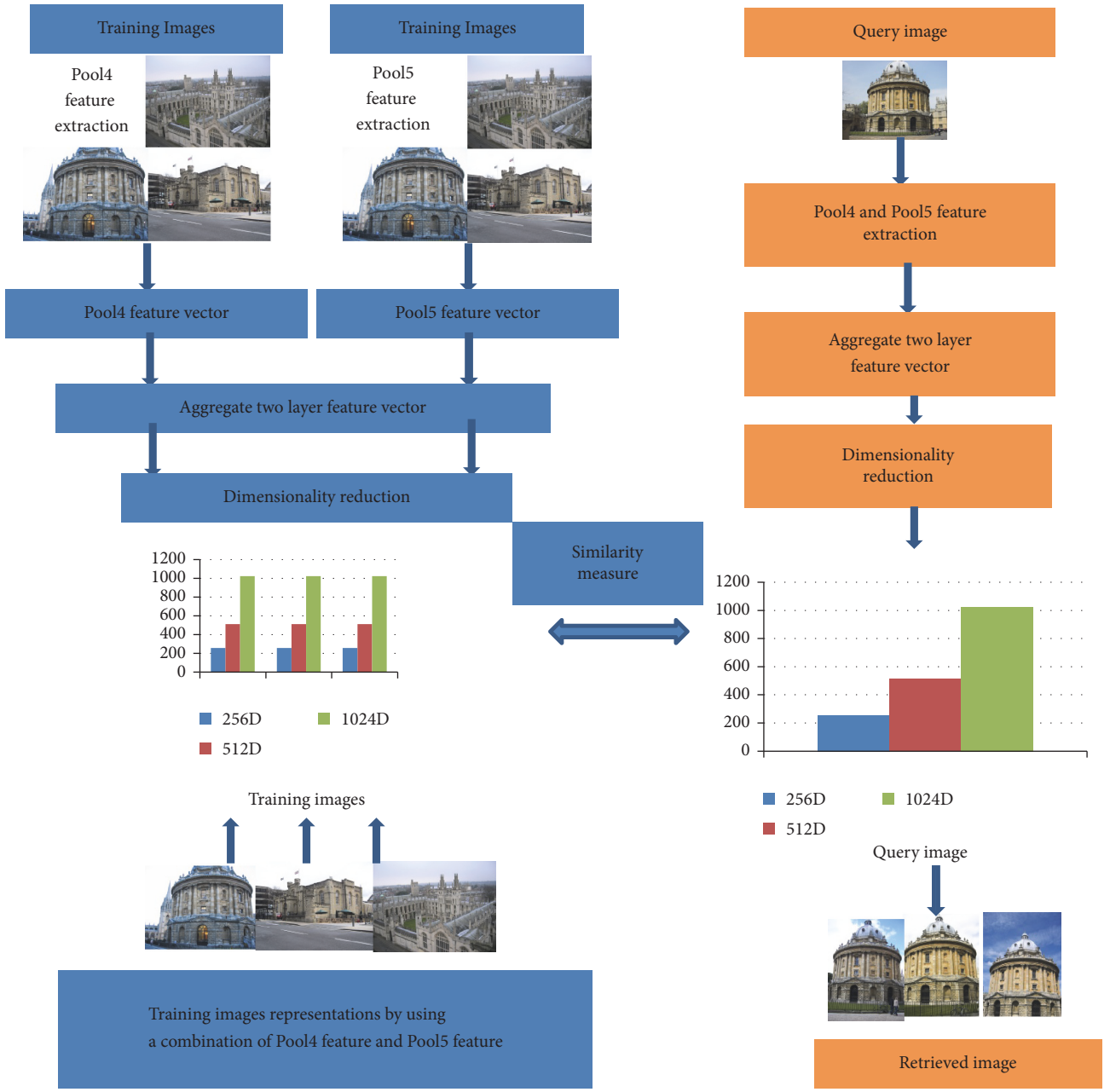

Figure 6: Block diagram of proposed research based on a fusion of multilayer feature.

extract the five-layer convolutional layer features in VGG16 (Conv1_2, Conv2_2, Conv3_3, Conv4_3, and Conv5_3, i.e., Pool1~5) after the ReLU function. Finally, the extracted CNN visual features were cascaded, and the generated feature vectors were sequentially L2 normalized, PCA whitened, and L2 normalized.

Regarding the PCA whitening parameters, the Oxford5k dataset used the PCA and whitening parameters learned on the Paris6k dataset, whereas the Paris6k dataset used the PCA and whitening parameters learned on the Oxford5k dataset. Additionally, to ensure fairness, the CroW algorithm used the PCA and whitening parameters learned on the Oxford 105k dataset in [30]. In this experiment, the CroW algorithm implemented the PCA and whitening parameters learned using the Oxford5k dataset. At the end of the experiment, the query expansion (QE) operation was used to extend the query to the image retrieval experiment.

The experiment extended the CroW algorithm to the ML-CroW algorithm to examine the retrieval performance comparison between different dimensions. This step extracted the CroW feature vector of each layer, performed the cascade operation, and used the same PCA whitening parameter for compression representation. Simultaneously, after a single layer of characteristic experiments, it was found that the Pool1 layer and the Pool2 layer had two obvious defects in the experiment. First, the two layers of features occupied

hundreds of GB of storage space, and the single-layer features performed image retrieval tasks. The experimental precision of these two features was also much lower than the latter three-layer convolutional characteristics. Therefore, in subsequent cascade feature experiments, the latter three layers of the convolutional layer in the VGG16 network model were mainly considered for experiments.

In the second part of the experiment, first, the convolutional architecture for fast feature embedding (Caffe) framework was used to fine-tune the VGG16 on the fine-tuning dataset so that the VGG16 network model had better classification results when considering the buildings, and the improvement occurred in the convolutional layer in the network. The feature was more conducive to detecting the detailed features of buildings. Second, the fine-tuned model was applied to the image retrieval task, and the same strategy as the first part of the experiment was used. Finally, the advantages of the fine-tuning model in the image retrieval tasks were summarized.

5.1. Retrieval Results of a Single Layer of ML-RCroW Features. The performance of all single-layer feature generation ML-RCroW descriptors in the CNN model was tested to compare the retrieval ability of multilayer features. Table 1 shows the performance comparison of different convolution layers on two standard datasets. The results show that when the ML-RCroW descriptors were generated using different convolutional layers, the image retrieval mAP value also increased linearly with the increase in convolution depth. Also, the lower layer features were several orders of magnitude weaker than the higher layer features, and at the same time, there were two obvious disadvantages in using lower layer features (such as Pool1 and Pool2 layers) as image retrieval features. First, the lower layer features are too general and lack high-level semantic information about the target of the image. Second, the lower layer features need to occupy more storage space, which is computationally expensive and therefore not suitable for image retrieval tasks. Therefore, in the following experiments, from the perspective of storage and computational power, the ML-RCroW algorithm discarded features of the Pool1 and Pool2 layers and mainly selected features extracted by the other three layers as considerations for the cascaded features.

Table 2 shows the mAP values of the CroW algorithm and the ML-RCroW algorithm on the two datasets after the fine-tuning model. By comparing the mAP values of the same three-layer features of Table 1, it is obvious that the expression of each layer is more powerful in the fine-tuning model. The specific performance is that the mAP value of the Oxford5k dataset had a 2% to 4% increase. The performance was weaker on the Paris6k dataset, but it also improved. Combining the comparisons on the two datasets proves that when the dataset is sufficient, making the corresponding fine-tuning model for the benchmark dataset is a simple and effective measure.

5.2. Retrieval Results of Two Layers of ML-RCroW Feature Fusion. This section verifies the effectiveness of the two-layer feature fusion of the VGG16 network model. The performance representations of the ML-CroW algorithm and the ML-RCroW algorithm in the retrieval task were analyzed in three dimensions on the Oxford5k and Paris6k datasets. Tables 3 and 4 show the experimental results of the two algorithms in the VGG16 network model and the fine-tuning network model. The results show that the optimal results were obtained when the dimensions of the Pool4 and Pool5 layers were combined in the three fusion modes.

First, the performance of the single-layer features in Table 1 with the cascading features was compared. To be fair, the ML-RCroW feature vector was compressed to 512 dimensions using PCA compression and compared to the ML-RCroW feature vectors of the same 512-dimensional single-layer Pool5 and Pool4. The mAP values were 1.2% and 10.4% higher on the Oxford5k dataset, and the mAP values were 0.9% and 4.8% higher on the Paris6k dataset. It can be seen that the ML-RCroW algorithm was comparable in the case of compression. Better performance than single-layer features was especially evident in the Oxford5k dataset, indicating that the two-layer feature fusion was better than the single-layer feature representation.

Second, the image retrieval performances of the ML-CroW and ML-RCroW algorithms with different cascaded features in the two benchmark datasets were also compared. The ML-RCroW algorithm was better than the ML-CroW algorithm in different dimensions, and the ML-CroW algorithm had a relatively large reduction in the expression ability after PCA compression representation. For example, when the Pool4 and Pool5 layer features were merged, the 256 dimensions on the Oxford5k dataset were 15% lower than the 1024 dimensions but only 6.5% lower on the ML-RCroW algorithm, showing that the ML-RCroW algorithm effectively responds to the target area and that the image context information is not easily lost. Even when the PCA compression representation was expressed, the loss accuracy was still in an acceptable range, which also indicates that most of the information removed when using PCA compression contains features that are invalid for image retrieval.

Finally, a two-layer feature fusion experiment was performed in the fine-tuning model. In the optimal Pool4 and Pool5 layer feature fusion, both ML-RCroW and ML-CroW algorithms improved, but the ML-RCroW algorithm still maintained a clear advantage over the contrast algorithm.

5.3. Retrieval Results of Three Layers of ML-RCroW Feature Fusion. Three-layer feature fusion experiments were performed on the VGG16 network model and the fine-tuning model. Tables 5 and 6 show the mAP results of the cascaded feature vectors in the Oxford5k and Paris6k datasets in the four dimensions of 1280, 1024, 512, and 256. Both tables show that both the ML-RCroW algorithm and the ML-CroW algorithm exhibited better performance than any single-layer feature, especially in Table 6. However, comparing performance with the highest dimension 1280 in Table 6 with the highest dimension 1024 of the two layers in Table 3, it was found that the three-layer feature fusion does not provide better performance under PCA compression. By concatenating two-layer and three-layer features in the Oxford5k dataset, the fused mAP values showed similar performance. The mAP values of the two algorithms in the

Table 1: mAP values of different layer features on two datasets.

Dataset	Pool1	Pool2	Pool3	Pool4	Pool5
Oxford5k	9.4	20.9	46.3	64.2	**73.4**
Oxford5k+QE	11.1	24.5	52.5	73.6	**78.2**
Paris6k	15.0	32.8	62.4	77.1	**81.0**
Paris6k+QE	16.9	41.7	68.8	81.1	**87.1**

Table 2: mAP values of different layer features in the fine-tune model on two datasets.

	Oxford5k			Paris6k		
	Pool3	Pool4	Pool5	Pool3	Pool4	Pool5
CroW	46.0	67.3	**77.2**	60.6	75.3	**82.0**
ML-RCroW	48.9	69.8	**79.3**	64.1	78.3	**81.7**
CroW+QE	50.3	71.0	**80.0**	67.1	79.8	**85.4**
ML-RCroW+QE	54.8	75.6	**83.4**	69.8	82.2	**85.5**

Table 3: mAP values of different combinations of two layers on two datasets.

	Pool4+Pool5			Pool3+Pool4			Pool3+Pool5		
Dimension	1024	512	256	768	512	256	768	512	256
Oxford5k+ML-CroW	**74.5**	65.9	59.5	66.4	61.2	45.1	71.4	68.1	43.9
Oxford5k+ML-RCroW	**76.8**	74.6	70.3	69.1	67.2	61.2	73.4	73.9	71.3
Paris6k+ML-CroW	**81.0**	72.9	70.0	73.4	70.9	60.9	79.4	77.3	58.2
Paris6k+ML-RCroW	**82.9**	81.9	79.0	77.5	78.1	75.6	81.1	81.5	78.7

Table 4: mAP values of different combinations of two layers in the fine-tuning model on the two datasets.

	Pool4+Pool5			Pool3+Pool4			Pool3+Pool5		
Dimension	1024	512	256	768	512	256	768	512	256
Oxford5k+ML-CroW	**77.3**	67.5	61.4	67.0	61.3	46.7	76.4	70.9	46.0
Oxford5k+ML-RCroW	**79.5**	78.0	74.1	69.9	68.4	63.3	78.7	78.1	74.8
Paris6k+ML-CroW	**82.1**	75.5	73.9	75.2	73.7	64.7	80.9	78.2	60.6
Paris6k+ML-RCroW	**83.5**	82.2	80.0	78.2	78.6	77.1	82.3	82.1	80.1

Table 5: mAP values of different combinations of three layers on two datasets.

Dimension	1280	1024	512	256
Oxford5k+ML-CroW	**74.5**	72.5	61.3	44.9
Oxford5k+ML-RCroW	**76.8**	76.7	73.7	70.8
Paris6k+ML-CroW	**80.7**	79.0	70.9	60.7
Paris6k+ML-RCroW	83.0	**83.7**	81.9	79.1

Table 6: mAP values of different combinations of three layers in the fine-tune model on two datasets.

Dimension	1280	1024	512	256
Oxford5k+ML-CroW	**76.9**	73.3	61.9	47.3
Oxford5k+ML-RCroW	**79.7**	79.4	77.4	72.5
Paris6k +ML-CroW	**81.9**	80.4	73.5	64.4
Paris6k +ML-RCroW	83.6	**83.9**	82.4	80.2

Table 7: Comparison of commonly used image retrieval methods on two common datasets.

CNN-based Method	Dimension	Oxford5k	Paris6k
CNN-ss [8]	32-120k	55.6	69.7
Neural codes [9]	4096	54.5	38.6
OxfordNet [12]	256	53.3	67
SPoC [13]	256	58.9	-
CroW [14]	512	68.2	79.6
R-MAC [15]	512	66.9	83.0
ML-RCroW (one layer)	512	79.3	81.7
ML-CroW (aggregate two layers)	1024	77.3	82.1
ML-RCRoW (aggregate two layers)	1024	79.5	83.5
ML-CroW (aggregate three layers)	1280	76.9	81.9
ML-RCRoW (aggregate three layers)	1280	**79.7**	**83.6**

Paris6k dataset only slightly improved when compared to previous results.

Obviously, the fusion of the three-layer features in this VGG16 network model is not superior to the two-layer feature, showing that the real impact on the retrieval in the experiment was the Pool5 layer feature. The Pool3 and Pool4 layer features provided relatively few features. The effect of the search results, and in the case of three-layer feature fusion, the comparison of the mAP values of the 1280 and 1024 dimensional features revealed that the cascaded feature vector had comparable performance capabilities to the two-layer feature, and the ML-RCroW algorithm when compressed by PCA was only 0.1% lower on the Oxford5k dataset and 0.7% on the Paris6k dataset. Finally, it can be concluded that the Pool4 and Pool5 layer fusion features show optimal performance in the PCA compression mode and use less storage space than the 3-layer feature in the retrieval system.

5.4. Comparison with Current Commonly Used Retrieval Algorithms. Tables 1–6 analyze the effects of ML-RCroW single-layer features, two-layer feature fusion, and three-layer feature fusion on image retrieval performance in the case of fine-tuning and no fine-tuning. Table 1 shows the image retrieval performance when using different layers of ML-RCroW features. When the depth is increased, the retrieval performance is also improved. The mAP values of 78.2 and 87.1 are achieved on the Oxford5k and Paris6k datasets, respectively. Therefore, this paper considers whether the fusion of different layer features can improve retrieval performance. It can be seen that as the number of layers of feature fusion increases, the performance of image retrieval is better. In the fusion of the two layers, the fusion of the Pool4 layer and the Pool5 layer can reach the highest mAP value, and the mAP values of 76.8 and 82.9 are achieved on the Oxford5k and Paris6k datasets, respectively. In the case of three-layer ML-RCroW feature fusion, the performance of image retrieval is only slightly improved compared with the result of two-layer feature fusion. The mAP values of 76.8 and 83.7 were achieved on the Oxford 5k and Paris6k datasets, respectively. These results show that the method of this paper can effectively capture the basic image features.

Table 7 compares the ML-RCroW algorithm with other deep learning image retrieval methods, including CNN-ss [8], neural code [9], OxfordNet [12], SPoC [13], CroW [14], and R-MAC [15]. Table 7 shows that the ML-RCroW algorithm proposed in this paper maintained good retrieval performance both in the Oxford 5k and Paris6k datasets. These results show that the ML-RCroW algorithm can effectively capture the basic features of the image, especially on the Oxford5k dataset. Similarly, the ML-RCroW algorithm does not consider lower-level features (such as Pool1 and Pool2 layer features) and significantly improves image feature vector space storage.

6. Conclusions and Future Work

Aiming at the problem of insufficient semantic representation of images by single-layer convolutional features, an ML-RCroW algorithm for generating feature vectors using multi-layer feature aggregation is proposed for image retrieval. ML-RCroW is simultaneously tested on two benchmark datasets of buildings, and the effectiveness and robustness of the ML-RCroW algorithm are verified.

There are two research priorities for the future. First, we can consider using a hash algorithm to further reduce the dimensions of the feature vector based on the existing feature vectors. Second, for large-scale image data, the CNN feature map is not necessary for a complete image feature representation. Adding traditional local features as auxiliary and supplemental components to CNN features should be considered to further improve the performance of the retrieval system.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interests regarding the publication of this paper.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grant 61762024 and in part by the Natural Science Foundation of Guangxi Province under Grant no. 2017GXNSFDA198050 and Grant no. 2016GXNSFAA380054.

References

[1] L. Zheng, Y. Yang, and Q. Tian, "SIFT meets CNN: a decade survey of instance retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 5, pp. 1224–1244, 2018.

[2] Y. Jing and S. Baluja, "VisualRank: applying pagerank to large-scale image search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1877–1890, 2008.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 770–778, Las Vegas, Nev, USA, June 2016.

[4] Z. Ji, S. Wu, F. Wang, L. Xu, Y. Yang, and X. Hu, "Mining regional co-occurrence patterns for image classification," *Mathematical Problems in Engineering*, vol. 2018, Article ID 4945304, 14 pages, 2018.

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 779–788, July 2016.

[6] R. Dong, X. Pan, and F. Li, "DenseU-net-based semantic segmentation of small objects in urban remote sensing images," *IEEE Access*, vol. 7, pp. 65347–65356, 2019.

[7] S. Antol, A. Agrawal, J. Lu et al., "VQA: Visual question answering," in *Proceeding of the International Conference on Computer and Applications*, pp. 2425–2433, 2015.

[8] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '14)*, pp. 512–519, IEEE, Columbus, Ohio, USA, June 2014.

[9] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, "Neural codes for image retrieval," in *Proceedings of the European Conference on Computer Vision*, vol. 8689 of *Lecture Notes in Computer Science*, pp. 584–599, Springer International Publishing, 2014.

[10] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *Proceedings of the European Conference on Computer Vision*, vol. 8695, pp. 392–407, Springer, 2014.

[11] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 3304–3311, June 2010.

[12] A. S. Razavian, J. Sullivan, A. Maki et al., "A baseline for visual instance retrieval with deep convolutional networks," in *Proceeding of the International Conference on Learning Representations*, ICLR, San Diego, Calif, USA, 2015.

[13] A. Babenko and V. Lempitsky, "Aggregating deep convolutional features for image retrieval," in *Proceedings of the International Conference on Computer Vision*, pp. 1269–1277, 2015.

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the International Conference on Learning Representations*, 2015.

[15] G. Tolias, R. Sicre, and H. Jegou, "Particular object retrieval with integral maxpooling of cnn activations," in *Proceedings of the International Conference on Learning Representations*, pp. 1–12, 2016.

[16] R. Dong, D. Cheng, and F. Li, "Aggregating deep convolutional features for image retrieval using multi-regional cross weighting," *Journal of Computer Aided Design & Computer Graphics*, vol. 30, no. 4, pp. 658–665, 2018.

[17] Y. Li, Y. Zhang, Y. Xu, J. Wang, and Z. Miao, "Robust scale adaptive kernel correlation filter tracker with hierarchical convolutional features," *IEEE Signal Processing Letters*, vol. 23, no. 8, pp. 1136–1140, 2016.

[18] W. Liu, D. Anguelov, D. Erhan et al., "SSD: single shot multibox detector," in *Proceedings of the European Conference on Computer Vision*, vol. 9905, pp. 21–37, 2016.

[19] J. Y. Ng, F. Yang, and L. S. Davis, "Exploiting local features from deep networks for image retrieval," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 53–61, Boston, MA, USA, June 2015.

[20] M. Yousuf, Z. Mehmood, H. A. Habib et al., "A novel technique based on visual words fusion analysis of sparse features for effective content-based image retrieval," *Mathematical Problems in Engineering*, vol. 2018, Article ID 2134395, 13 pages, 2018.

[21] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 2169–2178, June 2006.

[22] J. Donahue, "DeCAF: a deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31st International Conference on Machine Learning*, pp. 647–655, 2014.

[23] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 3431–3440, USA, June 2015.

[24] G. Bertasius, J. Shi, and L. Torresani, "High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision," in *Proceedings of the 15th IEEE International Conference on Computer Vision, ICCV 2015*, pp. 504–512, Chile, December 2015.

[25] M. Zahid, A. S. Muhammad, A. Nouman et al., "A novel image retrieval based on a combination of local and global histograms of visual words," *Mathematical Problems in Engineering*, vol. 2016, Article ID 8217250, 12 pages, 2016.

[26] N. Ali, K. B. Bajwa, R. Sablatnig et al., "A novel image retrieval based on visual words integration of SIFT and SURF," *PLoS ONE*, vol. 11, no. 6, Article ID e0157428, pp. 1–20, 2016.

[27] K. A. Qazi, T. Nawaz, Z. Mehmood, M. Rashid, and H. A. Habib, "A hybrid technique for speech segregation and classification using a sophisticated deep neural network," *PLoS ONE*, vol. 13, no. 3, Article ID e0194151, 2018.

[28] Z. Mehmood, M. Rashid, A. Rehman et al., "Effect of complementary visual words versus complementary features on clustering for effective content-based image search," *Journal of Intelligent & Fuzzy Systems Preprint*, pp. 1–14, 2018.

[29] D. Yoo, S. Park, J.-Y. Lee, and K. Inso, "Multi-scale pyramid pooling for deep convolutional representation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW 2015*, pp. 71–80, USA, June 2015.

[30] C. Farabet, C. Couprie, L. Najman, and Y. Lecun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.

[31] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "End-to-end learning of deep visual representations for image retrieval," *International Journal of Computer Vision*, vol. 124, no. 2, pp. 237–254, 2017.

[32] X.-S. Wei, J.-H. Luo, J. Wu, and Z.-H. Zhou, "Selective convolutional descriptor aggregation for fine-grained image retrieval," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2868–2881, 2017.

[33] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1–8, IEEE, Minneapolis, Minn, USA, June 2007.

[34] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: improving particular object retrieval in large scale image databases," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–8, IEEE, Anchorage, Alaska, USA, June 2008.