Aggregating Local Features of Convolutional Neural Network for Material Image Retrieval

Qing Qing
Shanghai University, Shanghai, 200444, China
Email: qqing@shu.edu.cn

Abstract. Large-scale microscopic images in materials science need to be indexed and managed using practical management tools. Content-based Image Retrieval (CBIR), which indexes and searches images based on the image features, allows for long-term data management in large-scale image datasets. Considering the difference between material microscopy images and natural ones, we propose a novel CBIR method for material microscopic images. In the proposed method, convolutional neural networks (CNN) are used to extract local features from an image, and the scale-invariant feature transform (SIFT) model is used to generate a keypoint density map (KDM). Experiments on a material microscopic image dataset show that the proposed method achieves an approving retrieval performance.

1. Introduction
Microscopic images record the spatial and morphological information of materials at different observation scales, which are important data resources in materials science. With the accumulation of data, many databases have been established to store material microscopic images. In the large-scale image database scenario, content-based image retrieval (CBIR) can quickly find images associated with the given visual information by analyzing the image features [1]. Therefore, introducing the CBIR method to material microscopic image databases can improve the efficiency of data utilization and promote the development of materials science in the era of big data.

The last decades have witnessed tremendous advances in CBIR approaches [2, 3]. However, most of the existing methods are designed for natural scene images (such as images in Oxford5k [4] and Paris6k [5]). Compared with images of natural scenes, microscopic images of materials have distinct characters, which makes certain differences between the retrieval tasks of the two types of images. First, natural scene image retrieval aims to find images containing similar object instances, while material microscopic image retrieval attempts to search for images with great attention to the right kind of details. Then, the imaging conditions involving observation scale and physical or chemical experimental environments result in a complex structure of material microscopic image datasets [6]. Finally, material microscopic images are usually gray-level images with highly repetitive structures [7]. Based on the above discussion, we propose a CBIR method for material microscopic images following the discussed pipeline. The highlight of this method is the image descriptor generating module. For a material microscopic image, the proposed method uses CNN to extract local features and aggregates the local features into an image descriptor by a new scheme.
2. Methods

2.1. Local feature extraction.
Deep networks are used to extract local features in the proposed method. Networks pre-trained on the ImageNet [8] dataset are fine-tuned on a material microscopic image dataset using category labels. The fine-tuning strategy is to fix the parameters of the shallow convolutional layers and fine-tune the parameters of the higher convolutional layers and fully connected layers. When extracting features, the output values of an intermediate convolutional layer $M$ are extracted as local features. Each local feature $f_s$ represents a sub-region $s$ based on its receptive field and is a $C$-dimensional vector, $N$ is the number of channels in layer $M$. Let $(W, H)$ be the size of an input image, the size of a sub-region is $\left(\frac{W}{a}, \frac{H}{a}\right)$, $d$ is the down-sample factor of the layer $M$.

2.2. Keypoint density map calculation.
The keypoints are some prominent points in the image, which have good invariance to changes in lighting, scale, rotation, etc. The scale-invariant feature transform (SIFT) [9] algorithm can detect keypoints of an image by constructing a Gaussian pyramid. The number of keypoints in each sub-region within an image reflects the significance of the sub-region. Therefore, an original keypoint density map (KDM) can be obtained by calculating the number of keypoints in the sub-regions of the input image. Let $k_s$ be the number of keypoints in the sub-region $s$, $K$ be the set of all sub-region’s keypoint amount in the image. The KDM value $\lambda_s$ of the sub-region $s$ is formulated by

$$\lambda_s = \frac{k_s}{\text{max}(K)} \quad (1)$$

2.3. Local feature encoding using KDM.
Applying the KDM value directly to the encoding scheme is still inefficient. The problem comes from two aspects. First, there is a square area in each material microscopic image that records the imaging information. This irrelevant area can generate a large number of keypoints and will be allocated to a large weight. Second, some sub-regions containing no keypoint are not encoded into the image descriptor because they don’t have weights. Thus, the image descriptor, based on the original KDM value, could lose details and lead to biased retrieval results. To address the above issues, we propose to filter and generalize the original KDM values.

To reduce the influence from the irrelevant area, we first locate the values of the area in KDM. Based on the position of these areas in the images, the KDM values of an irrelevant area are in the last few rows of the KDM matrix. The positions of these values rely on the size of the sub-regions, which change with the feature extraction model. Then, the KDM values of the irrelevant area are set to the minimum of the KDM values. To modify the weights to non-zero values, we use the neighbors of each sub-region to generalize its KDM values. Let $\text{neigh} (s)$ be the neighbors of sub-regions $s$, $N$ be the number of these neighbors. The generalization process is defined as:

$$\omega_s = \lambda_s + \frac{1}{N} \sum_{n \in \text{neigh}(s)} \lambda_n \quad (2)$$

Then, the output values $\omega_s$ work as encoding weights and is called KDM based weights. The whole progress of KDM generation is shown in Figure 1.

![Figure 1 The generation of KDM based weights.](image-url)
2.4. Local feature encoding.
To encode local features, a codebook is trained on a set of local features using a clustering algorithm. These local features are extracted from dataset images. The clustering centroids serve as the visual words of the codebook, and the number of the visual words decides the length of the final image descriptor. Then, based on the distances between the local features and the visual words, each local feature is quantified to the nearest visual word by putting the local feature's weights to the corresponding visual word. Let $v_i$ be the $i$-th visual words, $k$ is the size of the codebook, $NN(f)$ be the function to calculate the nearest neighbor of local features $f$. For an input image $I$, the KDM descriptor can be represented as:

$$KDM(I) = \left[ \sum_{NN(f)=v_1} \omega_1, \cdots, \sum_{NN(f)=v_k} \omega_k \right]$$

(3)

After image descriptor calculating, each image is represented as a descriptor vector. We use Cosine similarity to measure the distances of descriptor vectors between the query image and the reference images, and the reference images are ordered by their distances to the query image.

3. Experiment

3.1. Dataset.
This paper constructed a new material microscopic image dataset to realize and evaluate the proposed CBIR methods. The images in this dataset are all scanning electron microscopy (SEM) images, which cover the microstructure of eight types of materials: ceramics, cashmere, fibers, films, particles, powders, porous sponges, and tips. The images in this dataset are from the NFFA-EUROPE [10] project, which provides a dataset containing 21169 scanning electron microscope images at the nano-scale. The category labels of these images have been checked and confirmed by nanoscientists. The newly constructed dataset is divided into two subsets. One subset is used to train the CNN models, and another one is used to evaluate the performance of the retrieval algorithm, respectively. The training set includes 9100 images with class tags. The retrieval set contains 30 query images and 2270 reference images, and each query image is marked with a set of similar reference images to calculate the retrieval accuracy of the image retrieval algorithm. The images in these two subsets are all 1024×728 pixels in JPG format.

3.2. Experimental details.
We use VGG16 [11] and ResNet50 [12] models to implement the proposed method. The purpose of choosing two network models for experiments is to verify that the algorithm has robust performance on multiple models. In order to make these models fit material microscopic images, the cross-entropy loss function is used to fine-tune the models on the training subset. Considering the limitation of computing resources and the input size of the CNN model, the size of the input image is directly adjusted to 224×224 pixels when model training and feature extracting. As a result, the classification accuracy of VGG16 and ResNet50 is 91.36% and 94.10% on the testing set, respectively. During feature extraction, the output values of the conv5_2 and res5a layers are extracted from VGG16 and ResNet50, respectively, as the local features. For VGG16 and ResNet50, the input image is divided into 14×14 and 7×7 sub-regions, and each sub-region corresponds to a 512-dimensional local feature vector. In the retrieval performance evaluation stage, the experiment uses the Mean average precision (mAP) [4] value to evaluate the retrieval accuracy of the tested model.

3.3. Comparison with other retrieval methods.
In order to verify the superiority of the proposed method on microscopic images, this section compares KDM based image descriptor with several existing methods. The following three types of models are discussed: 1) Neural codes [13] using the output of the fully connected layer of CNN models as feature descriptors; 2) MAC [14] using the maximum pooling strategy to aggregate local convolutional features to an image descriptor; 3) models based on BoW and VLAD. The CNN features are extracted from VGG16 and ResNet50, respectively. The comparison results are shown in table 1. The Size in the table represents the dimensions of the corresponding image descriptors. On VGG16 and ResNet50, the
retrieval accuracy of the KDM descriptor reached 79.93% and 83.57% (mAP value). Figure 2 shows the retrieval results.

Table 1  The retrieval performance of different methods based on VGG16 and ResNet50

| Model    | Size | VGG16 mAP(%) | ResNet50 Size | mAP(%) |
|----------|------|--------------|---------------|--------|
| Neural Codes | (1, 512) | 74.42        | (1, 512) | 77.06  |
| VLAD     | (10,512) | 75.33        | (10,512) | 78.22  |
| BoW      | (1, 1000) | 77.22        | (1, 500) | 82.53  |
| MAC      | (1, 512) | 78.3         | (1, 512) | 79.78  |
| KDM      | (1, 1000) | 79.93        | (1, 500) | 83.57  |

Figure 2  Retrieval results based on the KDM descriptor

4. Conclusion
In this work, a new content-based image retrieval method is proposed for the microscopic images of materials. The proposed method uses CNN models to extracted local features. These features are the output of the middle convolutional layer and contain semantic information and local details. In addition, SIFT model is used to detect keypoints, which are robust to the changes in scale, lighting, and rotation. Based on the receptive fields of local features, an image is divided into fix-size sub-regions. Calculating the number of keypoints belonging to each sub-region, a keypoint density map (KDM) is generated to measure the importance of local features. The obtained KDM values are used to encode local features into the image descriptor.

Acknowledgments
I pleased to acknowledge my friend Li Pan for her invaluable assistance throughout the preparation of the original manuscript.

References
[1] Qayyum A, Anwar S M, Awais M and Majid M 2017 Medical image retrieval using deep convolutional neural network Neurocomputing 266 8-20.
[2] Alzu’bi A, Amira A and Ramzan N 2015 Semantic content-based image retrieval: A comprehensive study J. Vis. Commun. Image R. 32 20-54.
[3] Zhou W, Li H and Tian Q 2017 Recent Advance in Content-based Image Retrieval: A Literature Survey Journal of Nanjing University of Information Science Technology arXiv:1706.06064.
[4] Philbin J, Chum O, Isard M, Sivic J and Zisserman A 2007 Object retrieval with large vocabularies and fast spatial matching Computer Society Conference on Computer Vision & Pattern Recognition Minneapolis, pp. 1-8.

[5] Philbin J, Chum O, Isard M, Sivic J and Zisserman A 2008 Lost in quantization: improving particular object retrieval in large scale image databases IEEE Conference on Computer Vision and Pattern Recognition pp. 1-8.

[6] Adachi Y, Taguchi M and Hirokawa S 2016 Microstructure Recognition by Deep Learning Tetsu-hagane 102 722-729.

[7] Ge M, Su F, Zhao Z and Su D 2020 Deep Learning Analysis on Microscopic Imaging in Materials Science Materials Today Nano 11 (100087).

[8] Deng J, Dong W, Socher R, Li L, Li K and Li F F 2009 ImageNet: a large-scale hierarchical image database IEEE Conference on Computer Vision and Pattern Recognition pp. 248-255.

[9] Lowe D G 2004 Distinctive Image Features from Scale-Invariant Keypoints 2004 Int. J. Comput. Vis. 60 91-110.

[10] Aversa R, Modarres M H, Cozzini S, Ciancio R and Chiusole A 2018 The first annotated set of scanning electron microscopy images for nanoscience Sci. Data, 5.

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

[12] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition IEEE Conference on Computer Vision and Pattern Recognition pp. 770-778.

[13] Babenko A, Slesarev A, Chigorin A and Lempitsky V 2014 Neural Codes for Image Retrieval European Conference on Computer Vision Springer, Switzerland, pp. 584-599.

[14] Tolias G, Sicre R and Jégou H 2016 Particular object retrieval with integral max-pooling of CNN activations, arxiv:1511.05879.