



A Method for Analyzing the Composition of Petrographic Thin Section Image

Lanfang Dong^(✉)  and Zhongya Zhang

School of Computer Science and Technology,
University of Science and Technology of China, Hefei, China
lfdong@ustc.edu.cn, zhongya@mail.ustc.edu.cn

Abstract. The recognition and content analysis of the components in petrographic thin section image is a valuable study in geology. In this paper, we propose a two-stage method to segmentation and recognition of petrographic thin section image. In the first stage, we propose an image segmentation algorithm that can adaptively generate superpixel numbers based on SLIC algorithm. The algorithm is able to continually correct the number of superpixels in the iteration and then cluster the pixels of the image into superpixels by both color and spatial features. In the second stage, we designed a convolutional neural network and trained it with mineral grain images, which is then used to classify the superpixels obtained by first-stage. Finally, we count the categories and content of the components in the image based on the segmentation and classification results. We collected some images and invited geologists to label them for experimentation. The experimental results demonstrate the following: (1) Our proposed image segmentation algorithm is capable of dynamically generating the superpixels by the number of mineral grains in the image. (2) The CNN model we designed can accurately identify the categories of superpixel regions and has a small size. (3) The two-stage method is very effective in identifying the category of major components in an image and accurately estimating the content.

Keywords: Petrographic thin section image · Superpixels · Image segmentation · Image recognition · Component analysis

1 Introduction

The recognition and analysis of petrographic thin section plays an important role in the development of oil and gas resources. The traditional method of recognizing petrographic thin section is to cut and grind them into tens of micrometers of thin section, and then the researchers analyzes the image under the microscope [1]. Traditional methods require the knowledge of a researcher to complete, and the analysis of rock composition is cumbersome, time-consuming, and inaccurate. With the rapid development of digital image processing, image recognition technology has been widely used in petrographic thin section analysis. Segmentation and recognition of mineral grains in thin section images by image segmentation and recognition algorithms can accurately measure the categories and contents of major components in rock samples, thus better assisting petroleum development.

Because rock samples are usually formed by the bonding of mineral grains, there is a good idea for the analysis of the composition of petrographic thin section images. Using the image segmentation algorithm to segment the grains in the image and then identify each grain, the category and content of the main components in the thin section can be counted. In principle, almost all existing image segmentation algorithms can be used in the automatic segmentation of grain, such as region growing [2], ERS [3] and TurboPixel [4]. However, these algorithms are designed for ordinary scene images and may not be suitable for processing petrographic thin section image which contains large numbers of mineral grains. The superpixel segmentation algorithm is an image region over-segmentation algorithm. The image is divided into multiple irregular image blocks according to certain feature similarities. The current effective algorithms are LSC [5], SEEDS [6], SLIC [7] and so on. We believe that the sub-regions subdivided by the superpixels are similar to the contours of the mineral grains, so the superpixels obtained by the segmentation can be identified as a single grain. For example, Jiang et al. [8, 9] used a superpixel algorithm to segment and merge sandstone thin section images to obtain the contours of sandstone grains. However, the currently existing superpixel algorithm is very inefficient, or it is necessary to set the number of superpixels desired in advance. Therefore, we propose an algorithm that is fast and can dynamically adjust the number of superpixels based on the number of grains in the image.

After the image is segmented, the segmented sub-regions need to be identified. Since the crystal grains constituting the rock are generally colorless and transparent and have similar refractive indices, adjacent grains exhibit similar colors under a plane polarizing microscope, and it is sometimes difficult for the human eye to discern the difference. Unlike plane polarized light images, grains produce different interference colors under orthogonally polarized light [10]. For example, quartz and feldspar in sandstones require experts to combine orthogonal polarized and plane polarized images to distinguish them [11]. In recent years, convolutional neural networks have achieved remarkable success in a lots of computer vision tasks, such as image recognition [12, 13], object detection [14] and semantic segmentation [15]. Therefore, we have designed a CNN model that can combine the orthogonal polarized and plane polarized images to identify the sub-regions obtained by the previous segmentation. After identifying the category of each sub-region, the area of each sub-region of each category is counted, and the categories and contents of the major components in the image can be approximated.

In this paper, we have mainly completed the following work: (1) We propose a two-stage method to identify the major components of petrographic thin section. In the first stage, the image is segmented into over-segmented superpixel images. In the second stage, the CNN model is trained to identify each superpixel region to obtain the category of the mineral grain to which each superpixel region belongs. (2) We have enhanced the SLIC algorithm and proposed an algorithm that can generate superpixel adaptively based on the number of mineral grains in the image. For different images, our algorithm only needs to set a fixed K value, and the algorithm can dynamically adjust the number of superpixels to be segmented in multiple iterations. (3) We designed a CNN model that effectively classifies mineral grains that are common in petrographic thin section images. The CNN model can receive orthogonal polarized and plane polarized images as inputs, and can adapt to any size. (4) We created a

dataset that included thousands of petrographic thin section images. The details of the dataset are described in Sect. 2.3.

2 Proposed Method

The compositional analysis of the petrographic thin section image is divided into two stage. In the first stage, we proposed the SLIC method of adaptive superpixel number (AS-SLIC). The AS-SLIC algorithm can adaptively generate multiple superpixels based on the number and characteristics of mineral grains in the image. In the second stage, the CNN is trained to identify the superpixels obtained by the previous stage. By identifying the category of mineral grains in each superpixel region, the approximate content of the major components in each thin section image can be calculated.

2.1 The SLIC of Adaptive Superpixel Number (AS-SLIC)

We first introduce the original SLIC algorithm. The original SLIC superpixels correspond to clusters in the CIELAB color space and spatial space. First, the algorithm needs to set two values m and K , m is a constant value, and K represents the number of superpixels that are desired to be divided. Then, the image is divided into K grids, and the center point of each grid is used as the initial cluster $C_k = [l_k, a_k, b_k, x_k, y_k]^T$. A pixel P_i color is represented in the CIELAB color space $[l, a, b]^T$, as shown in Eq. (1), the d_c is color distance between P_i and C_k . The $[x, y]^T$ represents the position of the pixel. As shown in Eq. (2), the d_s represents the spatial space distance between P_i and C_k . The D_{dist} computes the distance between P_i and C_k is measured in both color and spatial space, which is defined as Eq. (3). In the Eq. (3), $S = \sqrt{N/K}$, where N is the number of pixels of the input image. Since the initial superpixel approximates an $S \times S$ region, the search for similar pixels is done in a region $2S \times 2S$ around the cluster center. In one iteration, each pixel is assigned to the nearest cluster center, then the new cluster center is recalculated and the next iteration continues.

$$d_c = \sqrt{(l_i - l_k)^2 + (a_i - a_k)^2 + (b_i - b_k)^2} \quad (1)$$

$$d_s = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \quad (2)$$

$$D_{dist} = \sqrt{\left(\frac{d_c}{m}\right)^2 + \left(\frac{d_s}{S}\right)^2} \quad (3)$$

In the petrographic thin section image especially the sandstone thin section image, the mineral grains of different origins are different in size, so the number of grains in the image varies from tens to hundreds. The original SLIC algorithm needs to set the number of superpixels divided in advance. When the number of mineral grains in the image is large, setting a smaller K will cause a superpixel to contain multiple grains. Conversely, if the number of grains in the image is small, setting a larger K will cause

over-segmentation. Both of the above cases are not conducive to subsequent classification. As shown in Fig. 2(a), we use the original SLIC algorithm and set K to 20. The results on the left Fig. 2(a) are acceptable, but because the grain area in the right of Fig. 2(a) is small, many superpixels contain multiple grain regions.

We proposed AS-SLIC algorithm takes into account the properties of mineral grains in petrographic thin section image. Most of the mineral grain composition is relatively single, so a single grain exhibits a uniform color in the image. Because of the optical properties of the crystals that make up the mineral grains, the differently oriented grain will show different colors under an orthogonal polarizing microscope. When a plurality of grain regions are included in one superpixel obtained by the segmentation, the superpixel will present a plurality of regions of different colors. So we can use the color histogram of all the pixels in the superpixel to determine whether the superpixel contains only a single grain. As shown in Fig. 1(a), this superpixel contains two grain regions, so the gray histogram of the superpixel has two peaks as shown in Fig. 1(b).

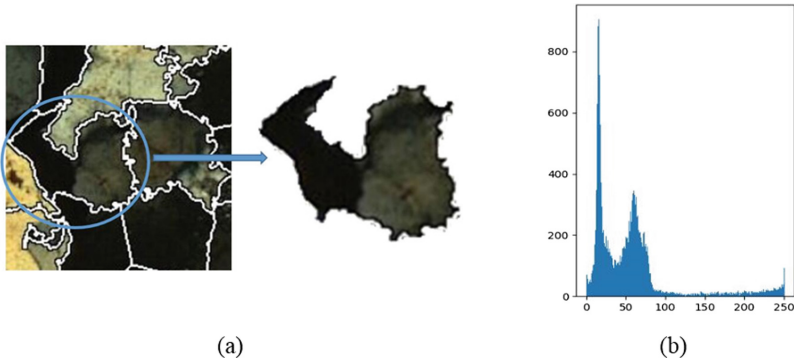


Fig. 1. (a) Is a superpixel that is not completely split. (b) Is the gray histogram of the (a).

Our method still needs to set K at the beginning, and K represents the number of cluster centers in the first iteration. Because the number of cluster centers will gradually increase during the iteration, the initial set K value is relatively small. In the original SLIC algorithm, once each pixel has been assigned to the appropriate cluster, then the mean vector of all the pixels belonging to each cluster are recalculated to update the cluster center. The difference between the algorithm we designed and the original SLIC lies in the update strategy of the cluster center. First, for each cluster center, the gray histogram H_k of all the pixels belonging to the cluster is counted. If the resulting gray histogram has only one peak, then it is likely that only one whole grain or a local region of one grain is included in the cluster. In this case, the average vector $V_k^i = [l_i, a_i, b_i, x_i, y_i]^T$ of all the pixels belonging to this cluster C_k is used as the new cluster center C_k' , as shown in the Eq. (4). If the gray histogram has two or more peaks, the pixels belonging to this cluster C_k are from multiple grain regions. Therefore, it is necessary to split the cluster C_k into multiple cluster. It is assumed that there are T peaks in the gray histogram H_k , the corresponding gray value at the peak is represented

by $Peak_i$, and V_k^j represents a vector of pixels in the cluster C_k whose gray value is equal to $Peak_i$. The previous cluster C_k is split into T new cluster according to peak. The calculation method of the new cluster C_k^i is as shown in the Eq. (5), where m represents the number of pixels in the cluster center C_k whose gray value is equal to $Peak_i$.

$$C_k' = \frac{1}{n} \sum_{i=1}^n V_k^i, n = |C_k| \quad (4)$$

$$C_k^i = \frac{1}{m} \sum_{j=1}^m V_k^j, m = |P_j|, P_j \in C_k \text{ and } P_j \text{ gray value is equal to } Peak_i \quad (5)$$

In order to avoid a superpixel splitting too many sub-regions, in each iteration we only select the two peaks containing the largest number of pixels in H_k to split. And in the experiment we set a maximum value K_{max} , when the number of cluster in the iteration is greater than K_{max} , the splitting strategy is no longer executed. We use L_2 norm shown in Eqs. (6) and (7) to compute the residual error R between the new cluster and previous cluster locations. The iteration is stopped when R is less than a certain threshold. The entire algorithm is shown in Table 1.

Table 1. The main steps of AS-SLIC algorithm

Algorithm 1.AS-SLIC

Input: I , the petrographic thin section image.
 K , the initial number of superpixels (The default value is 20).
Output: C , the initial segmentation results of I , $C = \{C_k\}_{k=1}^*$.

repeat
 for each cluster center C_k **do**
 for each pixel P_i in a $2S \times 2S$ region around C_k **do**
 Compute the distance D_{dist} between C_k and pixel P_i .
 Assign P_i to the best matching cluster C_k .
 end for
 end for
 for each cluster center C_k **do**
 Draw a gray histogram H_k of all the pixels in C_k .
 if The number of peaks in H_k is greater than 1 **then**
 Compute new cluster C_k^i by Eq.(5). Delete the cluster C_k .
 The new clusters C_k^i based on C_k is added to the clusters set C .
 else
 Compute new cluster centers C_k' .
 Set $C_k = C_k'$.
 end if
 end for
 Computer residual error R .
until $R \leq threshold$

$$R = \frac{1}{|C|} \sum_{k=1}^C R_k \quad (6)$$

$$R_k = \begin{cases} \|C_k^i - C_k\|_2^2, & \text{if } C_k \text{ is not split.} \\ \frac{1}{T} \sum_{i=1}^T \|C_k^i - C_k\|_2^2, & \text{if } C_k \text{ is split.} \end{cases} \quad (7)$$

2.2 Classification of Mineral Grains

Convolutional neural networks have been used for classification tasks for many years. Therefore, we designed a CNN model to identify the superpixel region obtained by the first stage segmentation. The orthogonal polarized image of the thin section focuses on the color features of the rock, and the plane polarized image focuses on the texture features of the rock. In order to make full use of the information in plane polarized and orthogonal polarized images, and to adapt to the superpixels of various sizes as input, we designed the CNN model to combine depthwise separable convolution [16] and spatial pyramid pooling [17].

Our CNN model architecture is shown in Table 2. The input to the model is two images, which are the orthogonal polarized image I_o and the plane polarized image I_p of the rock thin section. We expect the model to extract specific low-level semantic features from the two images, respectively. In the first input layer, the input I_o and I_p are group convolved with a $3 \times 3 \times 3 \times 32$ convolutional filter. The layer in front of the model uses a depth separable convolution, which is computationally intensive and leaves no information to communicate between the two images entered. After extracting the low-level semantic features from the two images, in order to combine the features in the two images to obtain the high-level semantic features of the rock sample as a whole, the 1×1 pointwise convolution is used for information exchange. All layers are followed by a ReLU nonlinearity and batchnorm. Down sampling is handled with strided convolution in the depthwise convolutions.

Since the size of the superpixel obtained in the first stage is uncertain, and in order to prevent loss of features due to image scaling, we hope that the model can accept superpixel regions of any size or scale as input. Therefore, our model uses the strategy of spatial pyramid pooling to produce a fixed-size representation. There are two fully connected layers after the spatial pyramid pooling layer and feeds into a softmax layer for classification. To prevent overfitting, the dropout strategy is used and the value is set to 0.3. Finally, cross entropy loss is employed as the loss function to train the network. Counting convolutions and fully connected layers, our model has 28 layers.

2.3 Image Dataset

To verify the effect of our method on petrographic thin section images, we collected the images and invited geologists to mark them. First, we created the Petrographic Thin Section Image Dataset (PTSID). The PTSID includes images of 801 rock samples in 7 categories, each sample with a plane polarized image and a orthogonal polarized image. So there are 1602 images in the PTSID, and the resolution of each image is $1392 \times$

Table 2. We designed the CNN Architecture. The “dw” represents a depthwise convolution, and “group” represents a group convolution. When the input resolution of the first layer of the model is 256×256 , the convolution kernel and input size of each layer are as follows.

Type/Stride	Filter shape	Input size
Conv group/S = 2	$3 \times 3 \times 3 \times 32 \times 2$ group	$256 \times 256 \times 6$
Conv dw/S = 1	$3 \times 3 \times 32$ dw	$128 \times 128 \times 32$
Conv/S = 1	$1 \times 1 \times 32 \times 64$	$128 \times 128 \times 32$
Conv dw/S = 2	$3 \times 3 \times 64$ dw	$128 \times 128 \times 64$
Conv/S = 1	$1 \times 1 \times 64 \times 128$	$64 \times 64 \times 64$
2×	Conv dw/S = 1	$3 \times 3 \times 128$ dw
	Conv/S = 1	$1 \times 1 \times 128 \times 128$
Conv dw/S = 1	$3 \times 3 \times 128$ dw	$64 \times 64 \times 128$
Conv/S = 1	$1 \times 1 \times 128 \times 256$	$64 \times 64 \times 128$
Conv dw/S = 2	$3 \times 3 \times 256$ dw	$64 \times 64 \times 256$
2×	Conv/S = 1	$1 \times 1 \times 256 \times 256$
	Conv dw/S = 1	$3 \times 3 \times 256$ dw
Conv/S = 1	$1 \times 1 \times 256 \times 512$	$32 \times 32 \times 256$
Conv dw/S = 2	$3 \times 3 \times 512$ dw	$32 \times 32 \times 256$
2×	Conv/S = 1	$1 \times 1 \times 512 \times 512$
	Conv dw/S = 1	$3 \times 3 \times 512$ dw
Conv/S = 1	$1 \times 1 \times 512 \times 512$	$16 \times 16 \times 512$
Conv dw/S = 2	$3 \times 3 \times 512$ dw	$16 \times 16 \times 512$
Conv/S = 1	$1 \times 1 \times 512 \times 512$	$16 \times 16 \times 512$
Spatial pyramid pooling	2×2 pooling 1×1 pooling	$8 \times 8 \times 512$
FC1	2560×512	5×512
FC2	512×21	1×512
Softmax	–	1×21

1040 pixels. Each image is detailed with the category of sample, the category and content of the main components (the content is expressed as a decimal, and the sum of the components is 1). There are 500 images in the PTSID that are pixel-level annotations for typical mineral grains, with a total of 2126 mineral grains in 21 categories. To train the CNN model we designed, we also produced the Mineral Grain Classification Dataset (MGCD). The mineral grain region marked in the PTSID is first extracted to generate a sub-image whose size is equal to the minimum circumscribed rectangle of the grain contour. In addition to the grain region in the sub-image, the pixel values of the other regions are filled with 0. The sub-image is then rotated, scaled, and cropped to augment the dataset. In order to maintain data balance between categories, we rotate the categories with fewer images to rotate more angles to get more new images. After the above process, a total of 8150 mineral grain images in 21 categories were included in the MGCD.

3 Experiments

3.1 Superpixel Segmentation Experiment

We selected sandstone thin section images from PTSID to verify the performance of the AS-SLIC algorithm. The reason for selecting the sandstone image is that it contains a large number of grains of different sizes, which can reflect the characteristics of the algorithm adaptively generating the number of superpixels. Figure 2 shows a comparison of our AS-SLIC with the original SLIC. The initial K values of both algorithms are set to 20. As shown in the right column of Fig. 2, when the number of grains in the image is large, the superpixels segmented by the original SLIC contain multiple grains, which is not as good as our AS-SLIC. Obviously, our algorithm can generate different superpixel number segmentation results for different images.

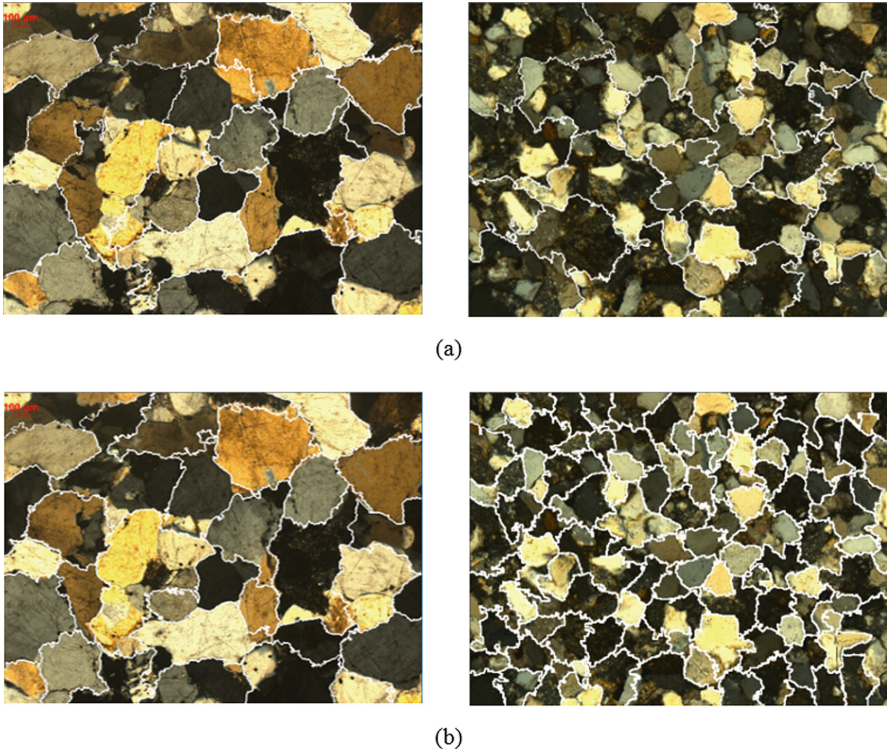


Fig. 2. The comparison between our AS-SLIC and the original SLIC. In the experiment, the initial K values of both algorithms were 20. The image in (a) is the result of segmentation using the SLIC, and (b) is the segmentation result of our AS-SLIC algorithm.

In terms of performance, we have found that majority of the image clusters do not change after 4 iterations. After 11 iterations, the residual error R of most images will be less than the threshold. Therefore, the time complexity of our algorithm in practical

applications is close to that of the SLIC algorithm. We experimented with an image with a resolution of 1392×1040 . The average computational complexity of our AS-SLIC and other existing superpixel algorithms is shown in Table 3. Because our algorithm needs to count the gray histogram in the iteration, and the number of clusters may increase, it takes more time than the original SLIC algorithm. However, compared to other classic superpixel algorithms such as GC [18], TP [4] and QS [19], our algorithm is still very fast.

Table 3. Time complexity compared to existing superpixel algorithms

Algorithm	GC [18]	TP [4]	QS [19]	SLIC [7]	Our AS-SLIC
Time (s)	77.6	203.6	70.3	5.4	7.1

3.2 Image Classification Result

Training Methods. Our model was built on the TensorFlow framework and was trained using the GTX 1080 GPU. We used the “Xavier” algorithm [20] to initialize the weights of all layers. The initial learning rate was 0.001 and reduced to $\sqrt{0.1}$ every 10 epochs. The training used stochastic gradient descent with 0.9 momentum. The batch size was set to 32 and the training “early stopping” (When the loss of the training set in five consecutive epochs is no longer reduced, the training is stopped) strategy is used. We use the “Multi-Size Training” strategy [17] mentioned in SPPNet for training. We resize the training set to four scales $s = \{224, 192, 160, 128\}$ and randomly selected one scale image for training in each epoch.

Results and Analyses. Our CNN model and other popular models were tested on the MGCD dataset. Table 4 compares our model to the VGG16 [21], ResNet [13], GoogLeNet [22] and MobileNet [16]. Overall, the ResNet34 model has the highest accuracy. This may be because the color and texture features in the thin section image are easily lost after multi-layer convolution, and the “shortcut connection” structure in the ResNet model makes the low-level information easier to forward propagation. Although the VGG model has the largest parameter, it is easier to overfit and the

Table 4. The experiment results of our model comparison to popular models

Network	MGCD test accuracy (%)		Number of parameters
	Top-1	Top-3	
VGG16	94.2	98.6	134 M
ResNet34	95.4	99.1	21.3 M
ResNet50	96.1	99.4	23.6 M
GoogLeNet	95.3	98.9	5.82 M
MobileNet	95.0	99.1	3.21 M
Ours	95.4	99.2	2.74 M

gradient disappears, so the performance is the worst. Our model is nearly as accurate as ResNet and is the smallest one of all models. Compared with the MobileNet and GoogLeNet models of the light weight, our model has the highest accuracy and the smallest model size. This shows that our model architecture is suitable for processing petrographic thin section images.

3.3 Analysis of Components

We summarize the results of the classification of each superpixel in the second stage, and the sum of the areas of the superpixel regions of each category is taken as the content of the mineral grains. As shown in Fig. 3, and 3(a) is a petrographic thin section image, and Fig. 3(b) is the segmentation and recognition result of Fig. 3(a). Each superpixel in the image covers an entire mineral grain or a part of a mineral grain, so the recognition result of the superpixel can be used as the category of the mineral grain. The different categories of mineral grains identified in the image are represented by different colors, and the percentage of each category of grains in the entire sample is counted. The results of the component analysis in this experiment were reviewed by several geologists. Experts believe that our analysis results are close to the results of manual analysis, and have a good application prospect, which provides a good solution for the automated analysis of petrographic thin section images in the future geology field.

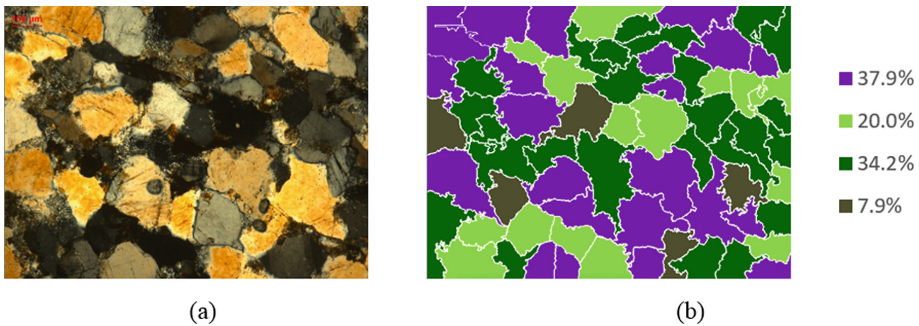


Fig. 3. Analysis of petrographic thin section image components. (a) is a petrographic thin section image. (b) is the segmentation and recognition result of (a).

3.4 Computation Cost

Our method is implemented using Python3.6 and Tensorflow1.12 on a PC with an GTX1080 GPU card (8 GB RAM) and Intel Core i7 3.6 GHz. The average processing time for an image with 1392×1040 resolution is 8.9 s. In the first-stage, since only the CPU was used for calculation, the average time was 7.1 s. In the second-stage, the superpixel region was identified using 1.7 s, and finally the statistics were consumed for 0.1 s. It can be seen that the first stage consumes a lot of time, and in our future research, implementing our superpixel segmentation algorithm as an algorithm that can be executed in parallel on the GPU can achieve faster execution efficiency.

4 Conclusion

In this paper, we propose a two-stage method for recognizing the category and content of the major components in the petrographic thin section image. We proposed an AS-SLIC algorithm, which adaptively adjusts the number of superpixels according to the gray histogram of the superpixel in each iteration. In the experiment, our AS-SLIC algorithm maintains the same speed as the original SLIC algorithm. When the number of grains in the image is large, it can adaptively increase the number of superpixels, and the generated superpixel is more suitable for the edge of the grain. We designed a CNN model to identify mineral grains that showed excellent performance in our MGCD dataset. Our model is a light weight network and has the same excellent classification results as other classic models such as ResNet. The results of our two-stage method for the analysis of petrographic thin section image were recognized and praised after evaluation by several geologists. Our method has important reference value for component analysis in other types of images, such as cancer cell analysis in medical images, remote sensing image analysis, and so on.

References

1. Tetley, M.G., Daczko, N.R.: Virtual Petrographic Microscope: a multi-platform education and research software tool to analyse rock thin-sections. *Aust. J. Earth Sci.* **61**(4), 631–637 (2014)
2. Izadi, H., Sadri, J., Mehran, N.A.: A new intelligent method for minerals segmentation in thin sections based on a novel incremental color clustering. *Comput. Geosci.* **81**, 38–52 (2015)
3. Liu, M.Y., Tuzel, O., Ramalingam, S., Chellappa, R.: Entropy rate superpixel segmentation. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2097–2104. IEEE (2011)
4. Levinshstein, A., Stere, A., Kutulakos, K.N., et al.: Turbopixels: fast superpixels using geometric flows. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(12), 2290–2297 (2009)
5. Li, Z., Chen, J.: Superpixel segmentation using linear spectral clustering. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1356–1363. IEEE (2015)
6. Bergh, M.V.D., Boix, X., Roig, G., et al.: SEEDS: superpixels extracted via energy-driven sampling. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) *ECCV 2012*. LNCS, vol. 7578, pp. 13–26. Springer, Heidelberg (2012). <https://doi.org/10.1007/s11263-014-0744-2>
7. Achanta, R., Shaji, A., Smith, K., et al.: SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(11), 2274–2282 (2012)
8. Jiang, F., Gu, Q., Hao, H.Z., et al.: Grain segmentation of multi-angle petrographic thin section microscopic images. In: *IEEE International Conference on Image Processing (ICIP)*, pp. 3879–3883. IEEE (2017)
9. Jiang, F., Gu, Q., Hao, H.Z., et al.: Feature extraction and grain segmentation of sandstone images based on convolutional neural networks. In: *2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 2636–2641. IEEE (2018)
10. Tarquini, S., Favalli, M.: A microscopic information system (MIS) for petrographic analysis. *Comput. Geosci.* **36**(5), 665–674 (2010)

11. Asmussen, P., Conrad, O., Gnther, A., et al.: Semi-automatic segmentation of petrographic thin section images using a seeded-region growing algorithm with an application to characterize weathered subarkose sandstone. *Comput. Geosci.* **83**, 89–99 (2015)
12. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: *Proceedings of Advances in Neural Information Processing Systems, Lake Tahoe*, pp. 1097–1105 (2012)
13. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas*, pp. 770–778 (2016)
14. Girshick, R., Donahue, J., Darrell, T., et al.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus*, pp. 580–587 (2014)
15. He, K., Gkioxari, G., Dollár, P., et al.: Mask R-CNN. *arXiv preprint [arXiv:1703.06870](https://arxiv.org/abs/1703.06870)* (2017)
16. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., et al.: MobileNets: efficient convolutional neural networks for mobile vision applications. *arXiv preprint [arXiv:1704.04861](https://arxiv.org/abs/1704.04861)* (2017)
17. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1904–1916 (2015)
18. Veksler, O., Boykov, Y., Mehrani, P.: Superpixels and supervoxels in an energy optimization framework. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *ECCV 2010*. LNCS, vol. 6315, pp. 211–224. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15555-0_16
19. Vedaldi, A., Soatto, S.: Quick shift and kernel methods for mode seeking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008*. LNCS, vol. 5305, pp. 705–718. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88693-8_52
20. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: *13th International Conference on Artificial Intelligence and Statistics*, vol. 9, pp. 249–256 (2010)
21. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)* (2014)
22. Szegedy, C., Liu, W., Jia, Y., et al.: Going deeper with convolutions. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9. IEEE (2015)