

Texture Image Segmentation Using Spectral Clustering

Hui Du^{1,2}, Yuping Wang^{1(✉)}, Xiaopan Dong¹,
and Yiu-ming Cheung³

¹ School of Computer Science and Technology, Xidian University,
Xi'an 710071, China

duhuiywy@hotmail.com, ywang@xidian.edu.cn

² College of Computer Science and Engineering,
Northwest Normal University, Gansu, China

³ Department of Computer Science, Hong Kong Baptist University,
Kowloon Tong, Hong Kong
ymc@comp.hkbu.edu.hk

Abstract. Clustering is a popular and effective method for texture image segmentation. However, most cluster methods often suffer the following problems: need a huge space and a lot of computation when the input data is large. To save the space and computation, we construct a novel algorithm for image segmentation. It consists of two phases: Sampling and clustering. First, we put some detectors into the data space uniformly using orthogonal design method. These detectors can move and merge according to the law of universal gravitation. When the detectors are in a stable status (i.e., do not move), these detectors are used as the representative samples to the next step. Second, to further improve the efficiency and avoid dependence on parameters, the Self-tuning Spectral Clustering (SSC) is used to the representative samples to do the clustering. As a result, the proposed algorithm can quickly and precisely realize the clustering for texture image segmentation.

Keywords: Texture image segmentation · Feature extraction · Gray level histogram · Sampling · Spectral clustering

1 Introduction

Image segmentation is an important component in image processing. It divides the image into several mutually disjoint sub regions, where the values of pixels in each sub-region have a higher similarity, while the values among different sub-regions have a lower one. Due to their simplicity and efficiency, clustering approaches were one of the best techniques used for the texture image segmentation. Texture indicates contextual information of image pixels [1], and it has been used in image segmentation. Because segmentation of texture images depends on pixel feature that are difficult to define and capture, it is more challenging than segmentation only using intensity of pixel. There have been many different approaches for segmentation based on different pixel feature extraction methods (e.g. [2, 3]).

Fuzzy c-means is one of the most popular methods for image segmentation because it doesn't need to construct the similarity matrix, is easy to implement and can achieve reasonable results at some cases. But this algorithm has the following shortcomings: it does not consider any spatial information and is easy to converge to a local optimal solution. Max Mignotte [4] proposed an original and simple segmentation strategy based on the K-means algorithm that considered spatial constraints and overcame the problem of the local convergence. D. Yang et al. [5] presented a novel clustering index in Gaussian kernel in search process, which is helpful to find the partitions for the highly overlapping and contaminating samples. And in order to improve the speed, they employed the histogram statistics to implement the pixels partition. Spectral clustering is a kind of effective clustering algorithms, and it has been proven to be powerful to image segmentation [6], applicable to different type of data sets, effective with high dimensional data, convergent to global optimal solutions. Among spectral clustering algorithms, self-tuning spectral clustering (SSC) [7] is one of the most effective ones. Since image segmentation needs to process a large amount of data, directly using SSC algorithm in image processing will get a huge similarity matrix that is far beyond the processing ability for an ordinary computer. To overcome this difficulty, we propose a novel algorithm that uses the law of universal gravitation to sample data firstly and then use SSC algorithm to cluster the resulted data (the representative samples gotten in the first phase) exactly.

The remainder of this paper is organized as follows: Sect. 2 presents the related knowledge. The proposed algorithm is presented in Sect. 3. Section 4 demonstrates the experimental results.

2 Related Knowledge

2.1 The Law of Universal Gravitation

The law of universal gravitation states that any two objects exert a gravitational force of attraction on each other. The magnitude of the force is proportional to the product of the gravitational masses of the objects, and inversely proportional to the square of the distance between them, i.e.,

$$F(t) = \frac{Gm_x m_y}{d(x(t), y(t))^2} \quad (1)$$

where t means time, and G is the acceleration of gravity.

2.2 Orthogonal Design

Orthogonal design can sample a set of representative data from a large set of data using an orthogonal matrix $L_M(Q^N)$ [8], where M refers to the number of data samples, N is the dimension of the data, Q is the number of possible values of data in each

dimension called level with Q being odd generally, and M , Q , N and J satisfy $M = Q^J$ and $N \leq \frac{Q^J - 1}{Q - 1}$ with J being the smallest positive integer.

3 The Proposed Algorithm

To do the image segmentation, we have to first select and extract the texture features of every pixel. Then, select the representative data (samples) from the texture features of pixels, and we call this step as “sampling”. Finally, cluster these representative samples.

3.1 Feature Extraction

Feature extraction for each pixel is one of the major issues for image segmentation. If the extracted features can be representative, then the image segmentation results will be better. We selected two kinds of representative features in our feature extraction stage. One is the gray level co-occurrence matrix, and the other is the gray histogram [9, 10, 11].

3.2 Sampling

To save the space and computation, we performed sampling on the data set firstly. We use orthogonal design to put some detectors in the data space, and these detectors move based on the law of universal gravitation, small mass detectors are attracted by larger ones and may move very near to larger ones. When the distance of two detectors is smaller than a given threshold value, they will be merged into one. When all detectors do not move and cannot be merged, they are called in a stable state and are seen them as the representative samples. The detail is as follows.

1. Put detectors and classify all data. Use the orthogonal design proposed in [8] to put detectors in the data space, e.g., yellow dots in Fig. 1(a). Then each detector x defines a set $C(x)$ of data in the data space. Data z is in $C(x)$ if and only if the

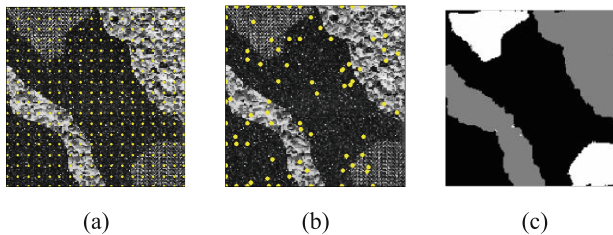


Fig. 1. (a) Put the detectors on the image (yellow points represent detectors) (b) Detectors after moving and emerging (c) Result of segmentation by proposed method (each of different gray colors represents one cluster) (Colour figure online).

degree of membership between data z and detector x is smaller than that between data z and any other detector. We call set $C(x)$ the class set of detector x .

2. Move and merge. The mass of each detector is defined as the number of data points contained in the circle of radius E with itself as the center. Thus the mass of any detector cannot be smaller than 1. Detectors move based on the law of universal gravitation as follows.

- (I) Compute the radius E using Eq.(2),

$$E = pS/(2Q - 1) \quad (2)$$

where S is the largest distance between two data points in data sets, Q is defined in Sect. 2.2, p is a parameter to control the size of radius E . Based on experiments, it is better to take the value of p between 0.8 and 1.2.

- (II) Compute gravity between two detectors using Eq. (1) with $G = 1$.
- (III) For each detector x , find out a detector y in its neighborhood which has the largest gravity to x in the neighborhood, where the smaller mass detector will move to the larger mass one. Without loss of generality, suppose that the mass of y is smaller, then y will move to x by formula (3) iteratively.

$$y(t+1) = (1 - \lambda)y(t) + \lambda x(t), \quad 0 < \lambda < 1 \quad (3)$$

where t is time, λ represents the step size. We take $\lambda = 1/3$ in our experiments.

- (IV) If the distance between two detectors x and y is shorter than αE , merge them and keep the larger mass detector, say x . Then, put data in $C(y)$ into $C(x)$ and delete $C(y)$ and y . The parameter α is used to control the speed of the merge. In experiments, we set $\alpha = 0.25$.
 - (V) Repeat step (II) to (IV) until all detectors no longer move and merge, these detectors are representative samples. For example, the representative samples of detectors in Fig. 1(a) are shown in Fig. 1(b). If E is too small, the detectors may not move and merge. If E is too large, it will cause a lot of computation. Through the experiments we found that E is better to take the value between 0.8 and 1.2.
3. These detectors are representative exemplars when they are in a stable state.

3.3 Clustering

After we get these representative exemplars from the previous phase, we will use the Self-tuning Spectral Clustering algorithm (SSC) to cluster these representative exemplars. For the details of algorithm SSC please refer to [7]. When these representative exemplars are grouped into several clusters, the classes $C(x)$ of the exemplars of each cluster will form a cluster in the image segmentation.

4 Experimental Results

4.1 Dataset

In the experiments for image segmentation, we use one natural image from Berkeley segmentation dataset [12] and constructed two synthetic images by selecting several images from Brodatz database [13].

4.2 Results and Comparisons

Figure 2 shows the experimental results of the four compared algorithms (SOM [14], FCM [15], K-means and our proposed method) on two synthetic images (the first two images in Fig. 2) and one natural image (the third image in Fig. 2). Table 1 shows the comparison results of the accuracy on these three images. The accuracy is computed by Eq. (4), where the number of pixels correctly classified was computed by comparing the results obtained by the corresponding method with the manual segmentation results.

$$Accuracy = \frac{\text{the number of pixels correctly classified}}{\text{the total number of pixels}} \quad (4)$$

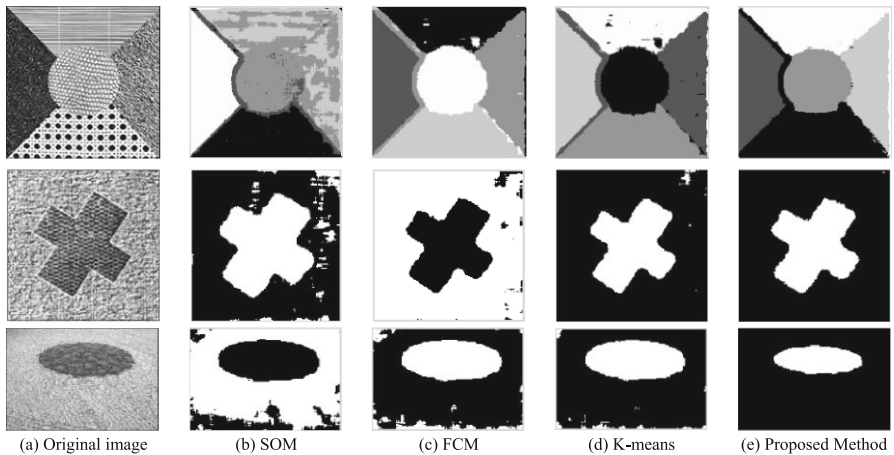


Fig. 2. Experimental results on the synthetic and natural images

Table 1. Comparison of Accuracy

Image	Size	SOM	FCM	K-means	Proposed Method
1	[153 × 55]	0.5997	0.9157	0.7300	0.9201
2	[150 × 150]	0.9362	0.9502	0.9430	0.9589
3	[100 × 150]	0.8371	0.9256	0.9377	0.9382

5 Conclusion

We propose a novel segmentation method for texture image. This method includes two phases, namely sampling and clustering. We design a new method for sampling based on the law of gravity, and then use the Self-tuning Spectral Clustering (SSC) to cluster the sampling results. The experiment results on synthetic and natural images indicate the proposed algorithm can achieve the higher segmentation accuracy compared to other three methods.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (No.61472297) and supported by the Fundamental Research Funds for the Central Universities (No.BDZ021430)

References

1. Zheng, Y., Kwong, M.T., MacCormick, I.J.C., Beare, N.A.V., Harding, S.P.: A Comprehensive Texture Segmentation Framework for Segmentation of Capillary Non-Perfusion Regions in Fundus Fluorescein Angiograms. *PloS one*, 9(4) (2014)
2. Sandberg, B., Chan, T., Vese, L.: A level-set and Gabor-based active contour algorithm for segmenting textured images. *CAM Report CAM 02-39*, UCLA Department of Mathematics (2002)
3. Zheng, Y., Chen, K.: A hierarchical algorithm for multiphase texture image segmentation. *ISRN Signal Process*, pp. 1–11 (2012)
4. Mignotte, M.: A de-texturing and spatially constrained K-means approach for image segmentation. *Pattern Recogn. Lett.* **32**, 359–367 (2011)
5. Yang, D.D., Wang, L., Hei, X.H., Gong, M.G.: An efficient automatic SAR image segmentation framework in AIS using kernel clustering index and histogram statistics. *Appl. Soft Comput.* **16**, 63–79 (2014)
6. Schultz, T., Kindlmann, G.L.: Open-Box spectral clustering: applications to medical image analysis. *IEEE Trans. Visual Comput. Graphics* **19**(12), 2100–2108 (2013)
7. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. *Adv. Neural Inf. Process. Syst.* **17**, 1601–1608 (2005)
8. Leung, Y.W., Wang, Y.P.: An orthogonal genetic algorithm with quantization for global numerical optimization [J]. *IEEE Trans. Evol. Comput.* **5**(1), 41–53 (2001)
9. Vasantha, M., Bharathi, D.V.S., Dhamodharan, R.: Medical image feature, extraction, selection and classification. *Int. J. Eng. Sci. Technol.* **2**, 2071–2076 (2010)
10. Sheshadri, H.S., Kandaswamy, A.: Experimental investigation on breast tissue classification based on statistical feature extraction of mammograms. *Comput. Med. Imaging Graph.* **31**, 46–48 (2007)
11. Hassan, M., Chaudhry, A., Khan, A., Kim, J.Y.: Carotid artery image segmentation using modified spatial fuzzy c-means and ensemble clustering. *Comput. Methods Programs Biomed.* **108**, 1261–1276 (2012)
12. <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>
13. <http://www.ux.uis.no/~tranden/brodatz.html>
14. Kohonen, T.: The Self-organizing Maps. *Proc. IEEE* **78**(9), 1464–1480 (1990)
15. Bezdek, J.C., Ehrlich, R., Full, W.: FCM: The Fuzzy C-means clustering algorithm. *Comput. Geosci.* **10**, 2–3 (1984)