International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.10, (2016), pp.417-428
http://dx.doi.org/10.14257/ijsip.2016.9.10.39

Fuzzy Partition based Similarity Measure for Spectral Clustering

Yifang Yang[1] and Yuping Wang[2]

[1]*College of Science, Xi'an Shiyou University, Xi'an 710065, China*
[2]*School of Computer Science and Technology, Xidian University, Xi'an 710071, China*
yangyifang@xsyu.edu.cn

Abstract

The efficiency of spectral clustering depends heavily on the similarity measure adopted. A widely used similarity measure is the Gaussian kernel function where Euclidean distance is used. Unfortunately, the result of spectral clustering is very sensitive to the scaling parameter and the Euclidean distance is usually not suitable to the complex distribution data. In this paper, a spectral clustering algorithm based on fuzzy partition similarity measure (FPSC) is presented to solve the problem that result of spectral clustering is very sensitive to scaling parameter. The proposed algorithm is steady extremely and hardly affected by the scaling parameter. Experiments on three benchmark datasets, two synthetic texture images are made, and the results demonstrate the effectiveness of the proposed algorithm.

Keywords: *spectral clustering, fuzzy clustering, fuzzy partition matrix, image segmentation*

1. Introduction

Image segmentation is just to segment an image into different subimages with different characters and get some interested objects. It is the important process of image analysis and image understanding [1], and plays a fundamental role in computer vision as a requisite step in such tasks as object detection, classification, and tracking [2]. As one of key methods of image segmentation, clustering algorithms have been widely used in image segmentation [3-]. In the past few decades, spectral clustering algorithms [4-29] which combined with spectral graph theory have shown great promise in data clustering and image segmentation. It realized dimension reduction by transforming the original dataset into a new one in lower-dimensional eigenspace, then performed clustering by using the eigenvectors of the normalized similarity matrix derived from the original dataset in lower-dimensional eigenspace. Spectral clustering methods attracts more and more interests because of its high performance in data clustering and simplicity in implementation, and have been successfully used to solve data clustering and graph partitioning problems.

However, there are still some open problems in traditional spectral clustering algorithms.

Firstly, the similarity measure is usually constructed by the Gaussian kernel function based on Euclidean distance. The obvious drawback of similarity measure is that the Gaussian scaling parameter is a somewhat sensitive parameter and the measure based on Euclidean distance cannot fully reflect the complex space distribution of dataset, and it is undesirable when clusters develop complicated manifold structure [12], therefore, it is crucial to choose a suitable similarity measure for spectral clustering algorithms. To overcome the influence of the scale parameter, Zelnik-Manor and Perona[13] proposed a self-tuning spectral clustering algorithm(STSC) that utilized a local scale for each data point to replace the single scale parameter, however, the local scale parameter in STSC,

ISSN: 2005-4254 IJSIP

the distance to a nearby neighbor, is still a Euclidean distance factor and cannot make any contribution to clustering better than using the scale parameter of Gaussian kernel function [20]. Fischer, B. *et. al.*, [21] proposed the path-based similarity, this similarity reflects the idea that no matter how far the physical distance between two points, they should be considered as in one cluster if they are connected by a set of successive points in dense regions. This is intuitively reasonable. However, it is not robust enough against noise and outliers [22]. Feng Zhao *et. al.*, [23] proposed fuzzy similarity measure by utilizing the partition matrix obtained by fuzzy c-means clustering algorithm. Secondly, when the scale n of the data set is relatively large, the overall time complexity and space complexity of standard spectral clustering can reach O($n^{3)}$ and O(n^2) respectively [9], which is difficult to store and decompose a large similarity matrix, especially for one image. Fowlkes *et. al.*, [4] presented the Nyström approximation technique to alleviate the computational burden of spectral clustering algorithms. Wen-Yen Chen *et. al.*, [30] generated sparse similarity matrix using the t-nearest-neighbor method to avoid the dense similarity matrix storing problem.

There are several papers reported the kernel fuzzy-clustering algorithm has better performance than the standard FCM. Authors in [24] reported good performance of kernel fuzzy-clustering algorithm on a 2-dimensional non-linearly separable synthetic dataset and compared the obtained results with those produced by the standard FCM; the classification rate for kernel FCM is much higher than standard FCM. The kernel based clustering algorithms can cluster specific nonspherical clusters such as the ring cluster, and quite well outperform FCM for the same number of clusters [25]. In this paper, a novel kernel fuzzy similarity measure is proposed and a new spectral clustering algorithm based on this measure is used in image segmentation. To alleviate the computational complexity, time and space complexity of the algorithm and avoid the dense similarity matrix storing problem, the t-nearest-neighbor method is applied to the algorithm.

The rest of this paper is organized as follows. In Section 2, we present a short overview about techniques of NJW, A new kernel fuzzy similarity measure which is used to construct the similarity matrix and the proposed FPSC method for image segmentation are described in details in Section 3. Experimental results analysis, discussion and parameter setting are described in Section 4. Finally, conclusions are given in Section 5.

2. Spectral Clustering Algorithm and the NJW Method

Spectral clustering methods widely adopt graph-based approaches for data clustering. Given a dataset $X = \{x_1, x_2 \cdots, x_n\}$ in R^d with d clusters, we represent the dataset X as a weighted graph $G(V,E)$, of which $V = \{x_i\}$ Set of n vertices represent n data points, and $E = \{W_{ij}\}$ Set of weighted edges indicate pairwise similarity between the x_i and x_j data points. The element W_{ij} of the affinity matrix is measured by a typical Gaussian function**Error! Reference source not found.**

$$W_{ij} = \begin{cases} e^{-\frac{d(x_i - x_j)^2}{\sigma^2}}, & i \neq j \\ 0, & i = j \end{cases} \tag{1}$$

Furthermore, the degree matrix D is a diagonal matrix whose element ($D_{ii} = \sum_{j=1}^{n} W_{ij}$) is the degree of the point x_i. In above framework, clustering problem can be seen as a graph partitioning problem.

As a spectral approach to graph partitioning problem, NJW method [27] is one of the most widely used spectral clustering algorithms. It uses the normalized affinity matrix as the Laplacian matrix and solves the optimization of the normalized cut criterion through considering the eigenvectors associated with the largest eigenvalues. The idea of NJW method is to find a new representation of patterns on the first k eigenvectors of the Laplacian matrix. The details of NJW method are given as follows.

(1) Form the affinity matrix $W \in R^{n \times n}$ defined by formula (1).

(2) Compute the degree matrix D and the normalized Laplacian matrix $L = D^{-1/2} W D^{-1/2}$.

(3) Let $1 = \lambda_1 \geq \lambda_2 \geq \cdots \lambda_k$ be the k largest eigenvalues of L and $p^1, p^2, \cdots, p^k$ be the corresponding eigenvectors. Form the matrix $P = [p^1, p^2, \cdots, p^k] \in R^{n \times k}$ and here p^i is the column vector.

(4) Form the matrix Y from P by renormalizing each rows of P to have unit length (i.e., $Y_{ij} = P_{ij} / (\sum_j P_{ij}^2)^{1/2}$) .

(5) Treat each row of Y as a point in R^k , and cluster them into k clusters via kmeans algorithm to obtain the final clustering of original dataset.

However, the similarity measure in NJW algorithm[4], *i.e.*, Gaussian kernel similarity measure, is sensitive to scaling parameter. To overcome this defect, the proposed new similarity measure in this paper is used to resolve the sensitivity problem of NJW method.

3. The Proposed FPSC Method

KFCM represents the kernel version of FCM by exploiting a kernel function for calculating the distance of data points from the cluster centers that is mapping the data points from the input space to a high dimensional space H (a Hilbert space usually called kernel space). In the new kernel space, the data show simpler structures or patterns. According to clustering algorithms, the data in the new space are more spherical and therefore can be clustered more easily by FCM algorithms [14-].

3.1. Kernel Fuzzy C-Means Clustering (KFCM)

Given a dataset $X = \{x_1, x_2 \cdots, x_n\}$ in the p-dimensional space R^p, in KFCM, a nonlinear map is defined as $\Phi : R^p \to H,\ x \to \Phi(x).$ where $x \in X$, Φ is a nonlinear mapping function from this input space to a high dimensional feature space *H*. The key notion in kernel based learning is that mapping function Φ need not be explicitly specified, and the dot product in the high dimensional feature space can be calculated through the kernel function $K(x_i, x_j) = \Phi(x_i)\Phi(x_j)$. Based on the above, KFCM algorithm partitions *X* into *c* fuzzy subsets by minimizing the following objective function:

$$J_m(U,V) = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m \left\| \phi(x_k) - \phi(v_i) \right\|^2 \quad (2)$$

where: *c* is the number of clusters , *n* is the number of data points, $v_i (1 \leq i \leq c)$ is the centroid of i-th cluster, u_{ik} $(1 \leq i \leq c, 1 \leq j \leq c)$ represents the fuzzy membership of k-th data point belonging to the i-th cluster, satisfying $\sum_{i=1}^{c} u_{ik} = 1$, where $U = (u_{ik} | i = 1, 2 \cdots c, k = 1, 2 \cdots n)$ is partition matrix, $V = \{v_1, v_2, \cdots, v_c\} \subset R^p$ is cluster centers, m is a constant, known as the fuzzifier (or fuzziness index), which controls the fuzziness of the resulting partition. In particular, we set m = 2 in this paper. $\|\Phi(x_k) - \Phi(x_v)\|^2$ is the square of distance between $\Phi(x_k)$ and $\Phi(x_v)$.

The distance in the feature space is calculated through the kernel in the input space as follows:

$$\begin{aligned} \|\Phi(x_k)\text{-}\Phi(v_i)\|^2 &= (\Phi(x_k)\text{-}\Phi(v_i))(\Phi(x_k)\text{-}\Phi(v_i)) \\ &= \Phi(x_k)\Phi(x_k) - 2\Phi(x_k)\Phi(v_i) + \Phi(v_i)\Phi(v_i) \\ &= K(x_k, x_k) - 2K(x_k, v_i) + K(v_i, v_i) \end{aligned} \tag{3}$$

In KFCM, the Gaussian function is taken as the Kernel function, namely $K(x, y) = \exp(-\|x - y\|^2 / \rho)$. where ρ defined as Kernel width, is a positive number, then $K(x, x) = 1$, and according to Eq.(2), Eq.(1) can be rewritten as

$$J_m(U, V) = 2\sum_{i=1}^{c}\sum_{k=1}^{n}(u_{ik})^m (1 - K(x_k, v_i)) \tag{4}$$

where $1 - K(x_k, v_i)$ can be considered as a robust distance measurement derived in the kernel space [18].

Finally, solving Eq.(4) for the minimum value of J will get the partition matrix U and clustering center V as follows:

$$\mu_{ik} = \frac{\{1/[1 - K(x_k, v_i)]\}^{1/(m-1)}}{\sum_{j=1}^{c}\{1/[1 - K(x_k, v_j)]\}^{1/(m-1)}} \tag{5}$$

$$v_i = \frac{\sum_{k=1}^{n}\mu_{ik}^m K(x_k, v_i) x_k}{\sum_{k=1}^{n}\mu_{ik}^m K(x_k, v_i)} \tag{6}$$

3.2. Kernel Fuzzy Similarity Measure

The membership value μ_{ij} in partition matrix U obtained by KFCM denotes the probability that a data point belonging to a specific cluster. Therefore, we can reasonably assume that two data points belonging to the same cluster have higher similarity, while two data points belonging to different cluster have lower similarity. Let $U = \{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_i, \cdots, \mathbf{u}_n\}$, here, $\mathbf{u}_i$ is the i-th column vector of matrix U, it consists of the membership value that data point *xi* belonging to *c* clusters. We can also assume similarity between two data points through their membership distribution, namely, the greater the inner product of $\mathbf{u}_i$ and $\mathbf{u}_j$, the higher the similarity of x_i and x_j. Conversely, the smaller the inner product of $\mathbf{u}_i$ and $\mathbf{u}_j$, the lower the similarity of x_i and x_j. Based on this idea, a new kernel fuzzy similarity measure is proposed.

Algorithm 1: a new kernel fuzzy similarity measure

Input: Dataset *X* to be clustered.

Output: Obtain the similarity matrix *S* of the dataset.

Step 1. Cluster the data set *X* into *c* clusters by KFCM, and obtain the partition matrix U .

Step 2. Let $U = \{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_i, \cdots, \mathbf{u}_n\}$, here, $\mathbf{u}_i$ is the i-th column vector of matrix *U*, it consists of the membership value that data point x_i belonging to c clusters.

Step 3. For each x_i and x_j ,

If x_i and x_j belonging to the same cluster,

$s_{ij} = 1$;

Else $s_{ij} = e^{(Ln2)\times(\mathbf{u}_i \cdot \mathbf{u}_j)} - 1$.

Step 4. Finally, the similarity matrix S is obtained.

To apply our method to texture image segmentation, NSCT (Non-subsampled Contourlet Transform) texture feature extracted from the image are used as a suitable image representation. The subband energy information of NSCT decomposition can describe the image feature splendidly. We also perform three-level discrete wavelet decomposition on texture images to extract a ten-dimension energy feature using a 16×16window,

which can be written as

$$E = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} |coef\,(i, j)| \tag{7}$$

where M × N is the subband size and |coef(i, j)| is the coefficient in the i-th row and j-th column of the subband. To avoid storing the dense similarity matrix, we employ the t-nearest-neighbor method [30] when we construct the kernel fuzzy similarity matrix.

3.3. FPSC Method for Texture Image Segmentation

The usual outline of FPSC for texture image Segmentation is as follows:

Algorithms 2: FPSC

Input: A M × M image I to be segmented; the number k of the image segmentation;

t-nearest-neighbor parameter t in FPSC.

Output: Segmented image.

Step 1. Feature extraction using Non-subsampled Contourlet Transform(NSCT), compute the texture features of each pixel in the image by Eq.(7), and obtain the texture feature dataset X with M^2 instances and 10 attributes.

Step 2. Construct the similarity matrix S using proposed kernel fuzzy similarity measure.

Step 3. Compute the dominant eigenvectors matrix $V \in R^{n\times n} (n = M^2)$ by adopting Nystr¨om approximation technique.

Step 4. Let $1 = \lambda_1 \geq \lambda_2 \geq \cdots \lambda_k$ be the k largest eigenvalues of S and $p^1, p^2, \cdots, p^k$ be the corresponding eigenvectors. Form the matrix $P=[p^1, p^2,\cdots, p^k]\in R^{n\times k}$ and here p^i is the column vector.

Step 5. Form the matrix Y from P by normalizing each of s rows to have unit length $(\text{i.e.}, Y_{ij}=P_{ij}/(\sum_j P_{ij}^2)^{1/2})$

Step 6. Treat each row of Y as a point in R^k , and and cluster them into k clusters by K-means algorithm to obtain the final segmentation results of image I.

4. Experimental Results and Analysis

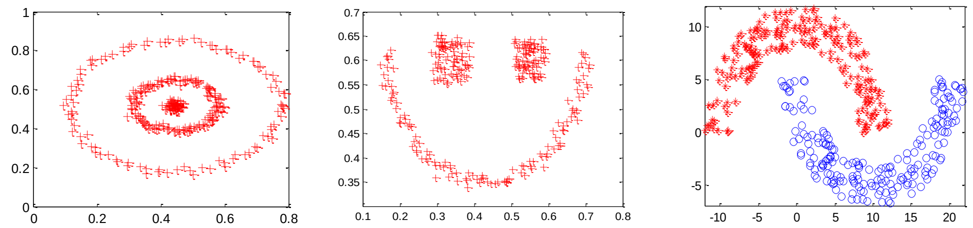
In order to investigate the quality of FPSC algorithm visually, we first tested the NJWN, STSC and FPSC algorithms on three benchmark synthetic datasets and analyzed the parameters sensitivity of the proposed FPSC algorithm. Then utilized NSCT texture feature extraction and apply the NJWN, STSC and proposed FPSC on two synthetic texture images. To avoid the dense similarity matrix storing problem, t-nearest-neighbor method was used to generate sparse similarity matrix in STSC and FPSC algorithms. For NJWN method, the number of random samples of pixels in the image is set to 100 in the following experiments. Our experimental environment are implemented in MATLAB 7.10 (R2010a) and performed on a computer with Intel (R) Xeon (R) CPU, 2.53 GHz and Windows XP Professional. In our experiments, kernel parameter ρ is set using fast bandwidth selection rule by paper [19] proposed.

4.1. Experiments on Three Benchmark Synthetic Datasets

The three benchmark synthetic datasets are Threecircles, LineBlobs and Twomoon. Figure 1(a)-(c), presents the three datasets respectively. How about the parameter settings of the three algorithms for three benchmark synthetic datasets are? The specific experimental settings for them are shown in Table 1. Parameter settings in NJWN: the scale parameter σ varied in the interval [0.05 1] with step length 0.01. Parameter settings in STSC and FPSC: the parameter t (nearest neighbor number) varied in the interval [5 50] with step length 5. To reduce the instability in initialization, the centroids obtained by K-means are taken as the initial centroids of FPSC. For all the methods, we performed 10 independent runs under their own parameters. Accuracy is widely used to evaluate clustering performance. To compute it for a clustering result, we need to build a permutation mapping function that maps each cluster index to a true class label. Following [26], this measure to evaluate the cluster quality is defined as:

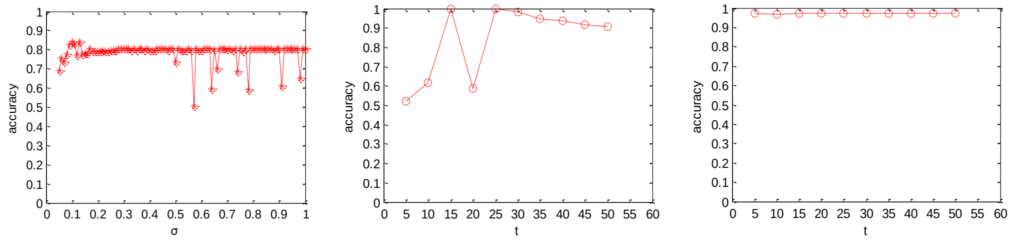
$$Accuracy=\frac{\sum_{i=1}^{n}\delta(y_i, map(c_i))}{n} \tag{8}$$

where n is the number of samples and y_i and c_i denote the true label and the algorithms clustering label respectively, $\delta(y,c)$ equals one if $y=c$, or else, it equals zero. map($\cdot$) maps each cluster label to a category label. The smaller the Clustering Error, the better the performance. The maximum average Accuracy rates were listed in Table 1.



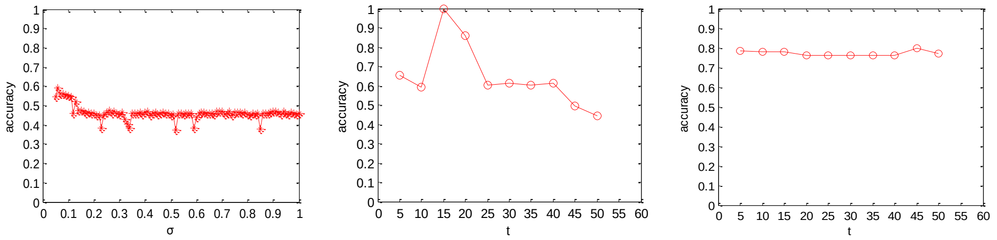
(a) Threecircles dataset (b) LineBlobs dataset (c) Twomoon dataset

Figure 1. Three Original Datasets



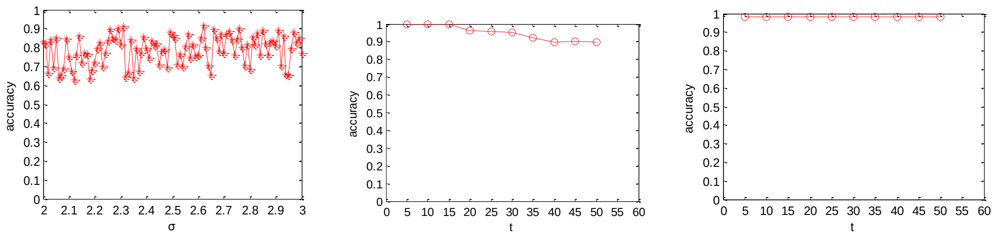
(a) NJWN: accuracy (b) STSC: accuracy (c) FPSC: accuracy

Figure 2. A Clustering Quality Comparison between NJWN, STSC and FPSC using the Three Circles Dataset



(a) NJWN: accuracy (b) STSC: accuracy (c) FPSC: accuracy

Figure 3. A Clustering Quality Comparison between NJWN, STSC and FPSC using the LineBlobs Dataset



(a) NJWN: accuracy (b) STSC: accuracy (c) FPSC: accuracy

Figure 4. A Clustering Quality Comparison between NJWN, STSC and FPSC using the Twomoon Dataset

Table 1. Comparison of Maximum Average Accuracy Rate(%)

Alogrithm	LineBlobs	Threecircles	Twomoon	σ	t
NJWN	86.43	61.20	81.60	0.05:0.01:1	-------
STSC	86.20	62.31	98.71	-------	5:5:50
FPSC	97.37	79.60	98.75	-------	5:5:50

4.2. Sensitivity Analysis of Parameters

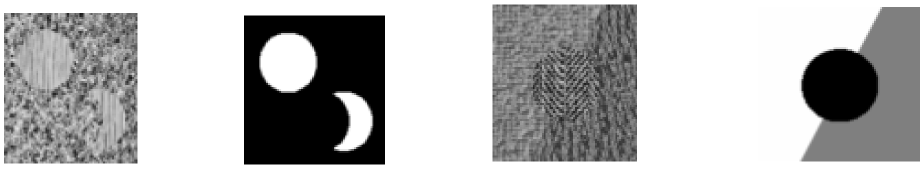
It can be seen from Table 1, that the average maximum accuracy of FPSC is better than NJWN and STSC algorithms. Figure 2-4, show the clustering accuracy performance of NJWN, STSC and FPSC on Threecircles, LineBlobs and Twomoon datasets respectively. For the Threecircles dataset, the clustering accuracy of NJWN and STSC algorithms greatly change with the scaling parameter, however, the clustering results of FPSC algorithm are steady extremely and hardly affected by the scaling parameter in t-nearest-neighbor method. For the LineBlobs and Twomoon datasets , likewise, the clustering results of SASC algorithm are more stable than those of NJWN and STSC algorithms. Moreover, the average accuracy rate of FPSC is more higher than that of NJWN and STSC on the three datasets. It is obvious that our method is robust to scaling parameter on the three synthetic datasets.

4.3. Experiments on two Textures Images

In order to apply our method to texture image segmentation, we tested NJWN, STSC and the proposed FPSC methods on two synthetic images. Figure 5, shows that the two synthetic images texture with two and three categories respectively and their ideal segmentation. For all the methods, we performed 10 independent runs. The maximum average Accuracy rates were listed in Table 2. As shown in Table 2, the maximum accuracy of FPSC slightly outperform NJWN and STSC methods. Figure 6 shows the optimal segmentation results of the synthetic texture Image1 with two categories using the three algorithms, respectively. It can be seen that there are some misclassified spots in black regions in Figure 6(a) using the NJWN method at $\delta = 0.2$; whereas, the segmented results of STSC and FPSC methods at t=20, presented in Figure 6(b), and and 6(c), are better than the first algorithms. The statistical results in Table 2, also agree with the visually inspection in Figure 6. From the segmentation results of two categories texture Image1, FPSC method obtains similar performance to STSC method and is more effective than NJWN method. Also, we can see that the segmented results obtained by FPSC method are better than those of STSC method in boundary regions. As shown in Table 2, when considering the clustering quality in accuracy, FPSC achieves comparable performance to STSC and performs slightly better than NJWN.

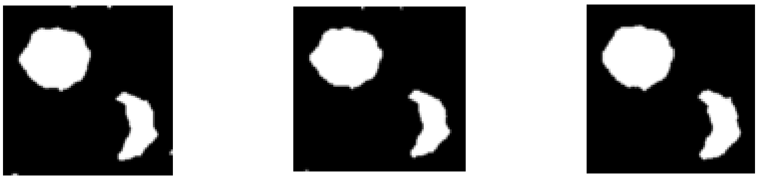
Figure 7, shows the optimal segmentation results of the synthetic texture Image2 with three categories using the three algorithms, respectively. It can be seen, there are more misclassified spots in the peripheral region of black circle in Figure 7(a), than in Figure 7(b), and in Figure 7(c). For FPSC method under t=40, we can find the optimal segmentation result is slightly better than STSC method under t=40 and much better than NJWN method at $\delta = 0.1$.The statistical results in Table 2, also agree with the visually inspection in Figure 7. The results of FPSC in Table 2, indicate that our method obtains the maximum average accuracy rate and the optimal visual result.

As seen in Figure 6-7, and Table 2, FPSC have obtained more satisfying segmentation results than other two algorithms. Considering the segmentation experimental results of three algorithms on the two synthesized texture images, we can conclude that FPSC has obtained impressive and encouraging partitioning results.



(a) Image1 (b) Image1: ideal segmentation (c) Image2 (d) Image2: ideal segmentation

Figure 5. Synthetic Texture Images and their Ideal Segmentation



(a) NJWN (b) STSC (c) FPSC

Figure 6. Optimal Segmentation Results of Image1

(a) NJWN (b) STSC (c) FPSC

Figure 7. Optimal Segmentation Results of Image2

Table 2. Comparison of Maximum Average Accuracy Rate(%)

Alogrithm	Image1	Image2	σ	t
NJWN	97.54	97.71	0.05:0.01:1	-------
STSC	97.71	97.76	-------	20:20:100
FPSC	97.82	97.80	-------	20:20:100

5. Conclusion

In this paper, a novel similarity measure called kernel fuzzy similarity measure is proposed to solve the problem that the result of spectral clustering is very sensitive to initialization and scaling parameter. Then this novel measure is integrated into spectral clustering to get the new FPSC algorithm. The clustering results of FPSC algorithm are steady extremely and hardly affected by the scaling parameter in t-nearest-neighbor method. It can not only avoid the influence of the scale parameter on spectral clustering effectively, but also is reliable than the Euclidean distance for the ”neighbour” selection. Experiments conducted on three benchmark datasets, two synthetic texture images and two real images show that the similarity measure and proposed FPSC method is effective. Our further works include the adaptive determination for the clustering number in our algorithm. In addition, texture features extraction is another key technique in image segmentation, which deserves further study.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grant No. 61272119 and 61472297.

References

[1] J. Shi and J. Malik, "Normalized cuts and image segmentation" Proceedings. 1997 IEEE ,Computer Society Conference on Computer Vision and Pattern Recognition, vol. 3, no. 2, **(1997)**, pp. 731-737.

[2] A. Yilmaz, O. Javed and M. Shah, "Object tracking: a survey", ACM Computing Surveys, vol. 38, no. 4, **(2006)**, pp. 1-45.

[3] S. C. Chen and D. Q. Zhang, "Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure", IEEE Transactions on Systems Man and Cybernetics Part B: Cybernetics, vol. 34, no. 4, **(2004)**, pp. 1907-1916.

[4] C. Fowlkes, S. Belongie, F. Chung and J. Malik, "Spectral grouping using the nyström method", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 2, **(2004)**, pp. 214-225.

[5] J. Shi and J. Malik, "Normalized cuts and image segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, **(2000)**, pp. 888-905.

[6] F. Zhao, L.C. Jiao, H. Liu and X.B. Gao, "A novel fuzzy clustering algorithm with nonlocal adaptive spatial constraint for image segmentation", Signal Processing, vol. 91, no. 4, **(2011)**, pp. 988-999.

[7] H. Liu, F. Zhao and L. Jiao, "Fuzzy spectral clustering with robust spatial information for image segmentation", Applied Soft Computing, vol. 12, no. 11, **(2012)**, pp. 3636-3647.

[8] H. Q. Liu, L. C. Jiao and F. Zhao, "Non-local spatial spectral clustering for image segmentation", Neurocomputing, vol. 74, no. 1-3, **(2010)**, pp. 461-471.

[9] S. P. Gou, X. Zhuang and L. C. Jiao, "Quantum Immune Fast Spectral Clustering for SAR Image Segmentation", IEEE Geoscience and Remote Sensing Letters, vol. 9, no. 1, **(2012)**, pp. 8-12.

[10] W. Chen and G. Feng, "Spectral clustering with discriminant cuts", Knowledge-Based Systems, vol. 28, **(2012)**, pp. 27-37.

[11] N. Rebagliati and A.Verri, "Spectral clustering with more than K eigenvectors", Neurocomputing, vol. 74, no. 9, **(2011)**, pp. 1391-1401.

[12] M.C. Su and C.H. Chou, "A modified version of the K-means algorithm with a distance based on cluster symmetry", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, **(2001)**, pp. 674-680.

[13] L. Z. Manor and P. Perona, "Self-tuning spectral clustering. Advances in Eighteenth Neural Information Processing Systems", **(2004)**, pp. 1601-1608.

[14] D. W. Kim, K. Y. Lee, D. Lee and K. H. Lee, "Evaluation of the performance of clustering algorithms in kernel-induced feature space", Pattern Recognition, **(2005)**, vol. 38, no. 4, pp. 607-611.

[15] D. Graves and W. Pedrycz, Performance of kernel-based fuzzy clustering, Electronics Letters, vol. 43, no. 25, **(2007)**, pp. 1445-1446.

[16] D. Graves and W. Pedrycz, "Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study", Fuzzy Sets and Systems, vol. 161, no. 4, **(2010)**, pp. 522-543.

[17] Chen, Long; Chen, C. L. Philip; Lu,Mingzhu, "A Multiple-Kernel Fuzzy C-Means Algorithm for Image Segmentation", It IEEE TRANSACTIONS ON SYSTEMS,MAN, AND CYBERNETICSłPART B: CYBERNETICS, vol. 41, no. 5, **(2011)**, pp. 1263-1274.

[18] D. Q. Zhang and S. C. Chen, "A novel kernelized fuzzy C-means algorithm with application in medical image segmentation", Artificial Intelligence in Medicine, vol. 32, no. 1, **(2004)**, pp. 37-50.

[19] D. M. Tsai and C. C. Lin, "Fuzzy C-means based clustering for linearly and nonlinearly separable data", Pattern Recognition, vol. 44, **(2011)**, pp. 1750-1760.

[20] X. Zhang, J. Li and H.Yu, "Local density adaptive similarity measurement for

[21] spectral clustering", Pattern Recognition Letters, **(2011)**, vol. 32, pp. 352-358.

[22] B. Fischer and J.M. Buhmann, "Path-based clustering for grouping of smooth curves and texture segmentation",IEEE Trans. Pattern Anal. Machine Intell., **(2003)**, vol. 25, no. 4, pp. 513-518.

[23] H, Chang and D. Y.Yeung, "Robust path-based spectral clustering, Pattern Recognit", vol. 41, no. 1, **(2008)**, pp. 191-203.

[24] F. Zhao, H. Liu and L. Jiao, "Spectral clustering with fuzzy similarity measure", Digital Signal Processing, **(2011)**, vol. 21, pp. 701-709.

[25] L. Zeyu, T. Shiwei, X. Jing and J. Jun, "Modified FCM clustering based on kernel mapping", Proc. of the Internat. Society for Optical Engineering, vol. 4554, **(2001)**, pp. 241-245.

[26] D. Graves and W. Pedrycz, "Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study", Fuzzy Sets and Systems, vol. 161, **(2010)**, pp. 522-543.

[27] M.Wu and B.Sch¨olkopf, "A local learning approach for clustering", In Proceedings of NIPS, **(2007)**, pp. 1529-1536.

[28] Y. Ng, M.I. Jordan and Y. Weiss, "On spectral clustering: analysis and an algorithm", Advances in Neural Information Processing Systems, **(2002)**, pp. 849-856.

[29] U. V. Luxburg, "A tutorial on spectral clustering, Statistics and Computing", vol. 17, no. 4, **(2007)**, pp. 395-416.
[30] F.R Bach and M.I. Jordan, "Learning spectral clustering", In Proceedings of NIPS, **(2003)**.
[31] W. Y. Chen and Y. Song and H. Bai, "Parallel Spectral Clustering in Distributed Systems", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 3, **(2011)**, pp. 568-586.

Authors

Yifang Yang, received the Ph.D. Degree from Xidian University in 2016. She is a Lecturer in Xi'an Shiyou University. Her current research interests include image processing, pattern recognition and evolutionary computation.

Yuping Wang, received Ph.D. from Xi'an Jiaotong University in 1993. He is a professor and supervisor for Ph.D. Student in Xidian University. His current research interests include evolutionary computation, data mining, global optimization and multiobjective programming.