A Fast Density Peak Clustering Algorithm Optimized by Uncertain Number Neighbors for Breast MR Image

Fan Hong¹*, Yang Jing¹, Hou Cun-cun¹, Zhang Ke-zhen² and Yao Ruo-xia³

¹Shaanxi Normal University, School of Computer Science, Xi’an 710062, China
²Xi’an Jiaotong University, College of Telecommunication, Xi’an 710049, China
³fanhong@snnu.edu.cn

Abstract. Aiming at the characteristics of abundant Magnetic Resonance (MR) image information, uneven gray scale, fuzzy boundary, and a fine structure that is difficult to distinguish and segment, this paper proposes an algorithm for segmenting MR images by an improved density clustering algorithm. First, the superpixel clustering algorithm (SLIC) is used to divide the image into a certain number of ultra-pixel regions and subsequently search the neighborhood of each superpixel according to a pre-specified threshold. Then, the KNN-DPC algorithm, in which the value of K is adaptively determined, is used to obtain the distance information of K nearest neighbors (density) and adjacent superpixels, and image segmentation is completed for each superpixel cluster. Two sets of natural image experiments show that this algorithm has high segmentation accuracy. Experiments on clinical breast MR images showed that this algorithm achieved good results for clinical MR image segmentation.

1. Introduction

Breast cancer has become a major malignancy that threatens women’s health and lives. The data show that in China, the proportions of newly diagnosed breast cancer patients who die annually were 12.2% and 9.6% [1]. At present, the international diagnosis of breast cancer is still in the exploratory stage. Medical imaging examination of breast cancer for early detection, treatment and prevention plays an important role; for example, x-ray mammography and magnetic resonance imaging (MRI) are internationally recognized and effective methods [2]. Among these techniques, magnetic resonance imaging (MRI) shows anatomical structure realistically and can make the diseased tissue and normal tissue is clearly visible with a high soft tissue contrast resolution. However, random noise is always present in the image, and a series of problems, such as gray unevenness and boundary blur, are encountered, which presents difficulties to image segmentation.

In the segmentation of medical images, clustering methods are frequently applied, and the FCM algorithm is widely used. To incorporate the neighborhood pixel information into the objective function of the FCM algorithm, Ahmed et al. proposed the FCMS algorithm [3]. To overcome the low efficiency of FCMS, Chen et al. proposed the FCMS1 and FCMS2 algorithms, which use the mean filter and median filter to calculate pixel neighborhood information [4]. The FCMS, FCMS1 and FCMS2 algorithms have many parameters, which are not easy to determine. Thus, Krinidis proposed the FLICM algorithm, which does not have any parameters [5]. Moreover, the k-means algorithm application is more popular [6-7]. For most k-means algorithms, the number of clusters needs to be set in advance; however, for users who do not have the expertise to determine the number of image clusters, this approach is highly difficult. Similar the k-means algorithm, the AP [8] algorithm is based
on a clustering method. Foster et al. applied a new affine intensity metric to select the optimal threshold level within the framework of the AP algorithm to complete the segmentation of lesion regions in positron emission tomography (PET) images [9]. But they are used only when the data shape is roughly spherical, in an image, the distribution of the pixels is not always of regular shape. In 2014, Rodriguez et al. proposed clustering fast search and identifying density peak clustering (DPC) [10]. The algorithm is suitable for large-scale data clustering analysis, can detect arbitrary clusters, and does not need to specify the number of clusters as a partition algorithm. However, it has several shortcomings: first, if the parameter truncation distance is not accurate, the decision graph is set incorrectly; second, the error will be transmitted in the subsequent transfer process, and DPC has no way to correct it; finally, since the initial cluster center is manually selected, it is difficult to make the correct decisions regarding certain data sets. In recent years, DPC researchers have proposed many new ways to overcome these drawbacks [11]. The idea of the KNN-DPC [12] algorithm is to use the K-nearest-neighbor information of the sample points to define the local density of the samples, search for and identify the density peaks of the samples, and use the peak point sample as the initial cluster center, which improves the results of the clustering algorithm.

This paper uses DPC algorithm optimized by uncertain number neighbors (IKNN-DPC) to effectively segment the MR image. First, the image is represented as superpixels; Next, according to the IKNN-DPC and allocation strategy in KNN-DPC to segment image. The remainder of this paper is mainly composed of four parts: the K-nearest neighbor optimized DPC algorithm is shown in section 2. Section 3 presents the shortcomings of the KNN-DPC algorithm in image processing and the proposed algorithm. Section 4 offers massive experimental results and analysis are drawn, the clinical MR image analysis in Section 5 and summarized in Section 6.

2. KNN-DPC Algorithm
In 2014, Rodriguez et al. proposed a density peak fast search clustering algorithm (DPC)[10]. The algorithm can quickly and automatically identify the density peak of a cluster of any shape as a clustering center, efficiently cluster sample points and effectively identify outlier samples. The local density and the distance of the DPC algorithm for the sample point are defined by (1) and (2):

$$\rho_i = \sum_{j \neq i} \chi(d_{ij} - d_c)$$  \hspace{1cm} (1)

$$\delta_i = \min_{j, \rho_j > \rho_i} (d_{ij})$$  \hspace{1cm} (2)

Where $d_{ij}$ represents the distance between samples $i$ and $j$; $d_c$ represents the truncation distance; when $x < 0$, $\chi(x) = 1$, otherwise, $\chi(x) = 0$; $\delta_i$ represents the distance from sample $i$ to the nearest sample $j$ that has a greater density than that of sample $i$; and when the local density $\rho_i$ of sample $i$ is the maximum, that is, $\delta_i = \max d_{ij}$. According to the formula for the local density $\rho_i$, the cutoff distance $d_c$ will greatly affect the local density of sample $i$ (the smaller the size of the data set, the greater the impact). For data sets of different sizes, the authors use different local density calculation formulas. However, currently, there is no accurate definition for the size of the data set, so the final clustering results may vary widely for the same dataset using different calculation methods. Although the author has verified the excellent performance of a series of DPC algorithms, there are some problems that cannot be ignored. For example, the distribution strategy of DPC algorithm may cause sample misalignment.

In 2016, Xie et al. demonstrated that the DPC algorithm does have the abovementioned problems, and proposed a fast density peak search algorithm based on K-nearest neighbor (KNN-DPC) to realize the efficient allocation of samples[12]. The algorithm uses the K-nearest-neighbor information of the sample to redefine the local density of the sample $\rho_i$: 
\[
\rho_i = \sum_{j \in \text{KNN}(i)} \exp(-d_{ij}) \tag{3}
\]

Where \(\text{KNN}(i)\) presents the set of \(K\) neighboring samples of sample \(i\). According to formula (3), compared with the DPC algorithm, the scope of the density calculation in the KNN-DPC algorithm is reduced from the global density to the local density of the \(K\) nearest neighbors of sample, and the smaller the distance from sample \(i\) to its \(K\)-neighborhood is, the larger the local density is; this finding will better reflect the local information of the data set.

To avoid the effect of outliers on the final result for the data set, KNN-DPC redefines the definition of an outlier according to the \(K\)-neighborhood information of sample \(i\):

\[
k_{\text{dist}}(i) = \max_{j \in \text{KNN}(i)} \{d_{ij}\} \tag{4}
\]

\[
\text{threshold} = \frac{1}{N} \sum_{i=1}^{N} k_{\text{dist}}(i) \tag{5}
\]

\[
\text{Outlier} = \{o \mid k_{\text{dist}}(o) > \text{threshold}\} \tag{6}
\]

Where \(k_{\text{dist}}(i)\) presents the KNN distance of sample \(i\), threshold defines the threshold, if \(k_{\text{dist}}(o) > \text{threshold}\), then \(o\) is an outlier, and \(N\) is the number of samples in the data set.

3. Image Segmentation Based on IKNN-DPC

3.1. The Problems of KNN-DPC in Image Segmentation

The KNN-DPC has been shown to be a very effective clustering algorithm on the UCI data set and the Olivetti face data set, which can accurately find the cluster centers and allocate the samples to the appropriate clusters. However, when the KNN-DPC algorithm is applied to the image, the information contained in the image is larger than that in the data set, resulting in a long time for segmentation, as shown in Table 1. In order to ensure the effectiveness of segmentation, we only intercept the part (Figure 1.(b)) shown in Figure 1.(a) to experiment. The experimental platform is Windows 10, the hardware is an Intel(R) Core(TM) i5-6600 3.30GHz CPU with 8G RAM, and all the algorithms are implemented in the MATLAB 2016a environment.

The segmentation results were evaluated from Dice coefficient [13]. Visual evaluation primarily considers the correctness of the image, whether the region is consistent, and whether the integrity of the edge is maintained. Dice coefficient to Berkeley computer vision group of the corresponding standard is a split gold standard. The Dice coefficient is calculated using formula (7):

\[
\text{DC} = \frac{2|M \cap N|}{|M| + |N|} \tag{7}
\]

Where \(M\) is a nonzero gold criterion and \(N\) is a nonzero segmentation result. The larger the DC value is, the better the segmentation result is.

From the above results and Table 1: For the image with complex structure (e.g., an airplane fuselage section as in Figure 1), the segmentation result of DPC algorithm is better than KNN-DPC algorithm's, the edge portion remains better, and the Dice coefficient is also higher relative to the KNN-DPC algorithm. In contrast to the image with simple structure (e.g., the eagle tail as in Figure 1), the segmentation result of KNN-DPC algorithm is better than DPC algorithm's. However, it can be seen from the running time in Table 1 that the running time of the two algorithms is relatively high for the intercepted small images and is difficult to apply in the actual clinical MR images.

The reason is that KNN-DPC algorithm has the following drawbacks when used on breast MR images:
Table 1. Dice Coefficient And Running Time.

| Evaluation Index | Dice Coefficient | Running Time(s) |
|------------------|------------------|-----------------|
|                  | DPC              | KNN-PC          | DPC            | KNN-DPC        |
| Airplane(60*60)  | 95.50%           | 79.14%          | 295.13         | 311.00         |
| Bird(80*90)      | 89.96%           | 95.89%          | 2253.9         | 2349.14        |

1) Since the KNN-DPC algorithm traverses the neighborhood information of each pixel in the image, and there are many pixels in each image, a great deal of time is necessary to calculate either the distance between pixels or the number of neighboring pixels. 2) When the KNN-DPC algorithm is applied directly for image processing, the input usually only contains the color information of the pixel value with no the spatial location information. Thus, the algorithm cannot well reflect the relationships between the pixels. 3) In the KNN-DPC algorithm, the K-nearest-neighbor information of the sample is introduced, and because the distribution of pixel points in an image is not uniform, the numbers of neighboring pixel points within a given range are different at different locations. If the K-nearest-neighbor information is introduced according to the original algorithm, it will not only increase the computational complexity of the algorithm but also affect the final clustering results.

3.2. IKNN-DPC Algorithm

3.2.1. SLIC Algorithm. A superpixel is an irregular pixel block that is composed of pixels with similar characteristics, such as texture, color, and brightness. Classifying the pixels relies on the similarity between the pixels such that fewer superpixels, instead of many pixels, are used to represent the image features. However, most of the superpixel algorithm processing results cannot meet the expectations of scholars[14-15]. There are two main shortcomings of these algorithms: First, a large amount of computing resources are needed, the segmentation results are of low quality, and the size and shape of each superpixel are different. Second, the algorithm contains multiple parameters that are difficult to adjust. In 2010, Achanta et al. proposed SLIC algorithm, in which a color image is transformed into CIE-LAB color space, and based on the pixel values in the CIE-LAB space and the coordinates of each pixel in the image, a five-dimensional eigenvector is constructed[16]. A distance measurement
standard is specified for this 5-dimensional eigenvector and the local clustering of image pixels, thus leading to the proposal of the SLIC algorithm.

To avoid placing the clustering center at the edge of the image and to reduce the likelihood of selecting a noise pixel as a clustering center, the SLIC algorithm initially samples the K spatial clustering centers and moves them to a seed position with the smallest gradient in the 3*3 neighborhood. The gradient of the image is calculated according to equation (8).

\[
G(x, y) = \|I(x+1, y) - I(x-1, y)\|^2 + \|I(x, y+1) - I(x, y-1)\|^2
\]  

(8)

Where \(I(x,y)\) is the value of the lab vector at \((x,y)\) and \(\|.\|\) is the norm. The calculation of the gradient takes into account the image color and the intensity information. Each pixel in the image is associated with the nearest search area that can cover the clustering center of the pixel. After all of the pixels have been associated with the nearest cluster center, the cluster center is recalculated for each cluster with the values of all pixels that belong to the center of the cluster. This step is repeated until the algorithm converges.

3.2.2. The Steps of IKNN-DPC Algorithm. Because of the shortcomings of the above algorithms for image processing, this paper proposes an IKNN-DPC algorithm.

First, use the SLIC algorithm to divide the image into several superpixel blocks and set a threshold in advance. For each superpixel block, we traverse the superpixel blocks that are adjacent to each other, and calculate the Euclidean distance between them. If the calculated distance is less than the threshold(The value is the experimental value, which is suitable between 2-5.), the superpixel block is considered to be a neighbor of the superpixel block and the distance between the two is stored. Since this paper uses a given threshold to determine whether two adjacent superpixel blocks are neighbors, the numbers of neighbors of the hyperpixel blocks that are obtained by this algorithm are not equal. After finding all the neighbors, the superpixels are clustered to segment the image.

The steps of the MR image segmentation IKNN-DPC algorithm are as follows:

Input: the image, the number of superpixels K, and the threshold \(\xi\)

Output: image segmentation results

Step 1. Enter the number of superpixels, according to the SLIC algorithm, obtain each segment of the superpixel information and segments adjacent to the superpixel block;

Step 2. According to the result of step 1, each superpixel block is traversed, along with the superpixel block adjacent to it. Calculate the distance (Euclidean distance) between the two blocks and compare it with the threshold \(\xi\); if distance < \(\xi\), it is a superpixel block of a neighbor and is saved;

Step 3. Calculate the local density value and distance \(\delta\) of the data according to Eq. (2) and (3);

Step 4. According to step 3, the cluster center CI is selected from the decision graphs according to \(\rho\) and \(\delta\);

Step 5. Find outlier samples according to Eqs. (4) - (6);

Step 6. Assign the allocation policy 1 to the non-outlier samples that are outside the cluster center;

Step 7. Assign the allocation policy 2 to the non-disconnected sample and the outlier sample that was not assigned in step 6;

Step 8. The small number of samples that were not allocated by either step 6 or 7 are considered noise points and are classified into their own clusters of the nearest cluster;

Step 9. Assign the clustering label that corresponds to the final clustering result that was obtained in step 8 to the superpixel block.

The algorithm is followed by the allocation strategy in the KNN-DPC algorithm. Allocation strategy 1 is used for non-outliers and allocation policy 2 for outliers and the non-outgoing points that are not allocated by policy 1. These allocation strategies are described below, specific description see the literature [12].

Allocation strategy 1:

1) Select an unpatched sample point from the cluster center CI as the center of the new cluster, and
the tag \(ci\) is already accessed;

2) Next, the sample point \(ci\) K-nearest neighbor set of samples into \(ci\), where the cluster initializes the queue \(V_q\) and places \(KNN(ci)\) samples in order into \(V_q\);

3) Select the header sample \(q\) from the queue \(V_q\) (i.e., the header sample is removed from the queue), for the sample element in the K-nearest-neighbor set \(KNN(q)\) of the sample point \(q\), if the following three conditions are satisfied: a) The sample element \(r\) is unassigned; b) The sample element is a non-outlier; c) The distance between the sample point \(q\) and the sample element \(r\) is less than (or equal to) the average of the distance between the sample point and all its K-nearest elements, i.e., \(d(q,r) \leq mean(\{|d_{qi}| \mid j \in KNN(r)\})\); the sample element \(r\) is classified into the cluster of the sample point \(q\); and the sample element \(r\) is added to the tail of the queue \(V_q\);

4) If the queue \(V_q\) is not empty, then return to step (3);

5) If there is an unattached sample point in the cluster center CI, go to step (1); otherwise, allocation strategy 1 ends.

Allocation strategy 2:

1) For each unassigned sample \(i\), the number of samples \(N_r(i)\) that belong to cluster \(c(1,...,|CI|)\) in its K-nearest-neighbor set \(KNN(i)\) is determined, and a \(1 \times |CI|\) vector \(N(i)\) is obtained. An unassigned sample (set to \(nr\)) is used to form an \(nr \times |CI|\) recognition matrix \(S\);

2) One or more samples \(p\), which are most likely to be correctly assigned, are selected from the recognition matrix \(S\) into the corresponding cluster.

3) Update recognition matrix \(S\): for the sample \(q\) that is not assigned in \(KNN(p)\), set \(N_q(q) = N_q(q) + 1\) and the sample \(p\) to 0 in the corresponding vector \(N(p)\) of the recognition matrix \(S\) (where \(k\) is the cluster number that was assigned to sample in step (2));

4) If no sample has been not allocated, end strategy 2; otherwise, go to step (2).

4. Experimental Results and Analysis

In this section, the algorithm is applied to the segmentation of multiple sets of images. To evaluate the performance of the algorithm, this section will be analyzed from two aspects of time complexity and segmentation accuracy. The experimental platform is the same as in Section 3.1.

4.1. Accuracy of Segmentation

4.1.1. The Partial Image Segmentation. Here, we first give the segmentation results of the proposed algorithm in Figure 1, as shown in Figure 2:

![Figure 2](image)

**Figure 2.** The segmentation result of Figure 1(b) based on the proposed algorithm; (a) airplane; (b) eagle.
With the analysis in Section 3.1, the segmentation accuracy of the two images is 94.22% and 93.08%, respectively, and the running time is 0.96 s and 1.07 s, respectively. It can be seen from the data, the algorithm in this paper reduces the time consumption, the segmentation accuracy is slightly lower than the higher. This is because the algorithm using the SLIC algorithm before the small image segmentation process, but the segmentation accuracy is not excessive loss, and it is still objective in the large image.

4.1.2. The Original Image Segmentation. To analyze the accuracy of the proposed algorithm in segmenting the image, the algorithm is applied to the segmentation of the natural image, as shown in Figure 3 and 4 (a). The results of the adaptive K-means algorithm and the AP algorithm are given for comparison.

![Figure 3. The segmentation result of natural image 1: (a) Original image; (b) Adaptive K-means; (c) AP; (d) Proposed algorithm.](image)

![Figure 4. The segmentation result of natural image 1: (a) Original image; (b) Adaptive K-means; (c) AP; (d) Proposed algorithm.](image)

As seen from the figure: Visual that Figure 3 and Figure 4 should be divided into two categories to separate the main body from the background. Figure 3 (b) in the eagle's tail split when the wrong, the two figures of the (b), (c) will be divided into a number of subjects in the cluster, and the background part of a certain error. In contrast to this algorithm, in the natural image segmentation can be good at the same time, the edge of the image segmentation retention is also superior.

| Table 2. Dice Coefficient And Running Time. |
|--------------------------------------------|
| Evaluation Index                           | Dice Coefficient |
|                                           | Adaptive K-means | AP | Proposed algorithm |
| Figure 3                                  | 96.41%           | 63.03% | 99.20%            |
| Figure 4                                  | 67.70%           | 57.04% | 98.89%            |

Table 2 shows the division of the Dice coefficients in Figure 3 and Figure 4. According to the data, the clustering effect of the adaptive k-means algorithm is better for the background color of a single image (Figure 3). In Figure 4, the background color is not uniform, and the effect of clustering is not ideal; therefore, the background is not reconstructed correctly. For the AP algorithm, the two kinds of image clustering effects are relatively mild compared to the adaptive k-means algorithm. In this paper,
the segmentation accuracies of these two algorithms are considerably higher than that of the AP algorithm. The segmentation accuracy in Figure 3 is slightly higher than that of the adaptive K-means algorithm, but the segmentation accuracy in Figure 4 is superior to that of the adaptive K-means algorithm. From Table 2, the proposed algorithm is more accurate than the other two algorithms and has good segmentation performance.

4.2. Time Complexity Analysis
In this paper, the image is first represented as superpixels, and then the superpixels are clustered to divide the image. Therefore, the time complexity of the algorithm mainly includes two aspects: the time complexity of computing superpixels is $O(N)$, where $N$ is the number of pixels in the image; and the time complexity of clustering is $O(c^2)$, where $C$ is the number of superpixel block and $k$ is the number of adjacent superpixel blocks for each superpixel block. Since $C$ is in the range of 1000-2000, the time complexity of the whole algorithm is $O(c^2)$. Table 3 shows the running time of Figures 3 and 4. Since the size of the two images is the same, only one set of data is given here.

| Evaluation Index | Adaptive K-means | AP | Proposed algorithm |
|------------------|------------------|----|-------------------|
| Running time     | 0.86             | 2.40 | 4.32             |

Combined with the data in Tables 1 and 3, it can be seen that the DPC algorithm and KNN-DPC algorithm consume more time in processing the partial image. However, the time that the proposed algorithm processes the original size image is far less than the partial time-consuming, this shows the superiority of the proposed algorithm time. Although the time to process the original size image than the proposed algorithm. But the segmentation quality is inferior to the proposed algorithm.

5. Clinical MR Image Analysis
Although the above two sections demonstrate the excellent performance of the algorithm, the clinical MR image has particular characteristics. Due to factors that influence the imaging process, such as temperature and magnetic field, MR images always show multiple features, such as uneven gray scale and weakened boundary. Thus, the features constitute a relatively special situation in image segmentation. This section will evaluate the segmentation performance of this algorithm on clinical MR images. As shown in Figure 5, the tumor area (the circled area) is present in the image. The final aim of the algorithm is to separate the tumor lesion area for further study.

The MR imaging data were derived from the German Siemens 1.5T standard dual-emulsion magnetic resonance scanner, which was the three-dimensional T1 weighted gradient sequence after the contrast agent was injected. Scan imaging patients with prone position, bilateral breasts naturally suspended in the cavity of the empty coil. The experimental image is a three-dimensional T1 weighted gradient echo sequence diagram before and after enhancement. From top to bottom, the first row is the image of the contrast medium not injected, lines 2, 3, 4, and 5 are images acquired in time series after injection of contrast medium. The imaging related parameters are: $TR = 5.6$ ms, $TE = 2.76$ ms, layer spacing 0.3 mm, layer thickness 1.2 mm, FOV = 34 *34 cm, image size 512*512, scan time 60 s.

Figure 5 (b) - (d)show the results of the adaptive k-means algorithm, the AP algorithm and the proposed algorithm on the images in Figure 5 (a). According to the segmentation result graph, the adaptive K-means algorithm blurs the boundary of the tumor region and the region cannot be segmented accurately. The AP algorithm for dealing with breast MR image segmentation produces a mottled background and the border remains unclear. According to the two breast MR local graphs, this algorithm is more accurate for the region that contains the tumor and almost completely separates the tumor from the surrounding tissue. The edge of the tumor is better maintained, thereby achieving good localization of the lesion site, which facilitates the doctor's diagnosis. This algorithm achieves good performance in clinical MR image segmentation.
6. Conclusion and Perspectives
This paper proposes a fast density peak clustering algorithm optimized by uncertain number neighbors. First, the image is divided into a certain number of superpixels by the simple, quick and easy-to-use SLIC algorithm, and the nearest neighbor to the superpixel is found. Then, the superpixels are clustered to divide the image. Two sets of natural image experiments show that this algorithm has higher segmentation accuracy. The clinical breast MR image segmentation experiments show that this algorithm performs better for mammography in the MR imaging part of the tumor and preserves the clarity of the edges. This property is more convenient for the localization of the lesion and facilitates follow-up medical diagnosis for clinical research. However, the algorithm for MR image segmentation requires prior knowledge, and the image should be clearly divided into several categories. Thus, this element of the algorithm needs to be improved.

![Figure 5](image_url)

Figure 5. The segmentation result of Clinical breast MR image; (a) Original image; (b) Adaptive K-means; (c) Ap; (d) Proposed algorithm.
Acknowledgments
The authors would like to the reviewers. This work supported in part by the National Natural Science Foundation of China under Grant 11471004, in part by the Shaanxi Provincial Natural Science Foundation under Grant 2014JM2-6115, and in part by the Shaanxi Province Science and Technology Research and Development Program under Grant 2012K06-36.

References
[1] Achanta R, Shaji A and Smith K 2010 Slic superpixels
[2] Ahmed M N, Yamany S M and Farag A A 1999 Bias field estimation and adaptive segmentation of MRI data using a modified fuzzy C-means algorithm (USA: Computer Society Conference on Computer Vision and Pattern) pp 250-255
[3] Cui Y, Tan Y and Zhao B 2009 Malignant lesion segmentation in contrast-enhanced breast MR images based on the marker-controlled watershed vol36 (Medical physics) pp 4359-4369
[4] Fan L, Strasser-Weippl K and Li J J 2014 Breast cancer in China (The lancet oncology vol 15) e279-e289
[5] Dice L R 1945 Measures of the amount of ecologic association between species (Ecology vol 26) pp 297-302
[6] Foster B, Bagci U and Xu Z 2014 Segmentation of PET images for computer-aided functional quantification of tuberculosis in small animal models (IEEE Transactions on Biomedical Engineering vol 61) pp 711-724
[7] Frey B J and Dueck D 2007 Clustering by passing messages between data points (Science vol 315) pp 972-976
[8] Ganesh M, Naresh M and Arvind C 2017 MRI Brain Image Segmentation Using Enhanced Adaptive Fuzzy K-Means Algorithm (Intelligent Automation&Soft Computing vol 23) pp 325-330
[9] Krinidis S and Chatzis V 2010 A robust fuzzy local information C-means clustering algorithm (IEEE Transactions on Image Processing vol 19) pp 1328-1337
[10] Levinstein A, Sminchisescu C and Dickinson S 2013 Multiscale symmetric part detection and grouping (International journal of computer vision 104) pp 117-134
[11] Liang Z and Chen P 2016 Delta-density based clustering with a divide-and-conquer strategy: 3DC clustering (Pattern Recognition Letters vol 73) pp 52-59
[12] Ng H P, Ong S H and Foong K W C 2006 Medical image segmentation using k-means clustering and improved watershed algorithm (In Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation) pp 61-65
[13] Rodriguez A and Laio A 2014 Clustering by fast search and find of density peaks(Science 344) pp 1492-1496
[14] Vedaldi A and Soatto S 2008 Quick shift and kernel methods for mode seeking (Computer vision CECCV) pp 705-718
[15] Xie J, Gao H and Xie W 2016 Robust clustering by detecting density peaks and assigning points based on fuzzy weighted K-nearest neighbors (Information Sciences vol 354) pp 19-40
[16] Zhang D Q and Chen S C 2004 A novel kernelized fuzzy c-means algorithm with application in medical image segmentation (Artificial intelligence in medicine vol 32) pp 37-50