Noname manuscript No.
(will be inserted by the editor)

BorderShift: Toward optimal MeanShift vector for cluster boundary detection in high dimensional data

Received: date / Accepted: date

Abstract We present a cluster boundary detection sch -eme that exploits MeanShift and Parzen window in high dimensional space. To reduce the noises interference in Parzen window density estimation process, the kNN window is introduced to replace the sliding window with fixed size firstly. Then, we take the density of sample as the weight of its drift vector to further improve the stability of MeanShift vector which can be utilized to separate boundary points from core points, noise points, isolated points according the vector models in multi-density data sets. Under such circumstance, our proposed BorderShift algorithm doesn't need multi-iteration to get the optimal detection result. Instead, the developed $Shift$ value of each data point helps to obtain it in a liner way. Experimental results on both synthetic and real data sets demonstrate that the F-measure evaluation of BorderShift is higher than that of other algorithms.

Keywords cluster boundary · MeanShift · Parzen window · High dimensional space

1 Introduction

Mining the potential patterns from unknown data sources not only can help people use the data, but also can find valuable information. Clustering technique makes the process of discovering class structures become possible. It groups data objects based on information found in the data that describes the objects and their relations. The research goal is to assign similar objects into a group and disperse dissimilar objects in other groups. Nowadays, a lot of clustering algorithms have been proposed and widely used in many related fields, such as image segmentation [1][2], information retrieval [3][4], natural language processing [5][6], bio-pharmaceuticals, financial, statistics, etc. Cluster boundary technique aims to find the data with a clear class labels but existing differences from most data objects of the clusters. Paying attention on the people carrying tumor virus but not suffering from cancers in normal people is an interesting work. These people now are healthy, but may suffer the diseases after a period of time. If we can detect these people effectively based on the sampled data which come from CT and blood data records, etc., this will greatly benefit for the medical field. Besides, the cluster boundary objects of face images always have some characteristics differ from normal face images, such as profile face, wearing sunglasses, whiskers which may paly bad influence in personal information collection process. By this research, the computers can identify the abnormal images quickly. For getting different patterns from different types of datasets, effective mining methods need to be studied.

In low dimensional space, geometric features and statistical knowledge are always used to describe the data distribution. Researchers also proposed many methods to discover the cluster structure, eliminate noises, detect isolated and boundary points. For example, BRIM [7]BAND [8]BRINK [9]EDGE [10] BERGE[11] have been developed to finish the cluster boundary detection task. For high dimensional space, manifold learning cuts the high dimensional manifold space to low dimensional space. A new low dimensional space will be generated after applying the space mapping technique. Then, it makes the idea of processing high dimensional space problems by low dimensional theory possible. Many

methods can refer to LDA, LLE [12], LPP [13],etc.. But general data mining methods always combine the dimension reduction and general patterns analysis methods to solve the high dimensional data analysis tasks. In essence, real high dimensional data analysis theory and technique should be researched. So, this paper will try to develop a high dimensional cluster boundary detection algorithm.

MartinEster [14] proposes DBSCAN algorithm, and introduces the cluster boundary based on the concept of density. However, how to get the whole boundary objects of clusters is not studied. Xia [15] et al. distinguish the core points, boundary points, noise points using reverse k nearest neighbors in BORDER algorithm. Compared to the boundary points, the number of reverse k nearest neighbor of noise points and isolated points are fewer, so the noises and isolated points are also detected as cluster boundary points by mistake in low and high dimensional space. BRIM algorithm detects the cluster boundary based on whetherthe neighborhoods distribution of a boundary point is non-uniform, it solves the problems that exist in the BORDER. In this algorithm, a diameter line divides the circular area to two parts. Calculating the number difference of points between the two parts is used to reflect the uniformity of the area in BRIM algorithm. However, if the area has a lot of noises, the area centroid may be located near the circle center. Then, the diameter may not be the best density division line. For high dimensional space, using a diameter line to divide a high dimensional sphere may be impossible. Only a high dimensional segmentation plane[16][17] can do the work, so BRIM only can be used in two-dimensional space and detecting the high dimensional cluster boundary via BRIM is impossible. To improve the detection accuracy, BAND extracts the cluster boundary based on the concept of coefficient of variation. Compared to boundary points, the noises located near the cluster edge always have the same values of variation coefficient. It leads to the detection result with noises. BRINK takes weighted Euclidean distance as the similarity measure between the data objects, and detects the cluster boundary based on the local distribution characteristics. In high dimensional space, losing measure meaning may be inevitable with the rapid increase of the dimension. The sparseness of data distribution reduces the differences between data objects. Then the effectiveness of traditional measure methods will degrade. So the BRINK cannot be effectively used in high dimensional data. BERGE uses the idea of evidence accumulation to label boundary objects with the help of multiple statistical learning. However, it aims to detect the cluster boundary for mixed attribute data sets and the error rate will increase quickly if some noises are labeled as boundary objects by mistake. So, the algorithm is sensitive to noises.

Parzen [18] et al. propose Parzen window density estimation technique to describe the data distribution. It's a nonparametric method which could capture the characteristics of the data itself without any prior knowledge or assumptions. Prior knowledge is expensive and pre-estimating also may be not accurate. So the Parzen window density estimation has been widely used in statistics and computer science. MeanShift was proposed by Fukunaga[19] in 1975. By constantly updating the mean vector of the current neighborhood, the MeanShift vector gradually reaches a local steady state, that is, the maximum value of local density, or the convergence condition is satisfied. Yizong Cheng [20] extents MeanShift by weighting each drift vector using kernel function, thus the sample points in the neighborhood of a sampling point have different importance. In 1999, Comaniciu et al. [21] introduce the MeanShift approach into the feature space analysis in image smoothing and segmentation. Meanwhile, the literature transforms the nonrigid tracking into an optimization problem for MeanShift iteration. Then it makes the image tracking in real time.

For the fact that the existing traditional cluster boundary detection algorithms cannot detect the cluster boundary of high dimensional data effectively, we propose a new cluster boundary detection algorithm for high dimensional data based on Parzen window and MeanShift in this paper. The main contributions of this paper are summarized as:
(1) introduce k nearest neighbors to replace sliding window with fixed size in kernel density estimation;
(2) propose an improved MeanShift vector;
(3) propose a cluster boundary detection algorithm for high dimension data called BorderShift.
The remainder of this paper is organized as follows. Section 2 introduces the Parzen window, MeanShift and our development work. Proposed BorderShift framework is presented in Section 3. Results obtained in various cluster boundary experiments are reported in Section 4. The discussion is reported in Section 5. Finally, the conclusions are drawn in Section 6.

2 Parzen window and Meanshift

In section 2.1, we will introduce the Parzen window density estimation method and our improved work. In section 2.2, we will introduce the MeanShift approach. Then, we will use the proposed density estimation method as the weight of drift vector to generate the new MeanShift vector.

2.1 Parzen window

We define x_i as a data object in d dimensional data space, n is the total of data space. Taking x as the center and h as the side length to make a hypercube, then the volume of it is $V = h^d$. Constructing a function $\varphi(u)$ and making it meets the condition of $\varphi(u) \geq 0$ and $\int \varphi(u)du = 1$, then the number of samples falling into the hypercube is:

$$n_v = \sum_{i=1}^{n} \varphi(\frac{x - x_i}{h}) \tag{1}$$

The probability density estimation of x_i is described as follows:

$$f(x_i) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V} \varphi(\frac{x - x_i}{h}) \tag{2}$$

where $\varphi(u)$ could be a window function or kernel function [22]. Generally, we can choose the square window, normal window, etc.. Gaussian function is a classical robust radial basis kernel which has a high anti-interference ability. Effectively smoothing noises and rotation invariance make the different dimensions have the same influence on density estimation. So it is more popular used than other window functions.

In formula (2), the value of probability density estimation is heavily dependent on the side length of the hypercube, so the size of the window seriously restricts the effectiveness of Parzen window density estimation method. For example, Parzen window will degrade into the grid density estimation method in two-dimensional space. For the varying density and multi-density data set, the value of the probability density estimation is more sensitive to the size of window. In the literature [23], Vidar V. Vikjord et. al compared and analyzed some nonparametric density estimation methods. Their analysis results show that kNN density estimation method is more close to the true data distribution than fixed window density estimation methods. So, our paper will use the Gaussian window as the window function and the k nearest neighbor area as the sliding window:

$$f(x_i) = \sum_{j \in k_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_j - x_i||}{2\sigma^2}) \tag{3}$$

where σ^2 is the variance of all data objects, $||x_j - x_i||$ is the Euclidean distance between the data object x_i and x_j , k_{nn} is the k nearest neighbor collection of x_i , x_j is a data object in the k nearest neighbor collection of x_i.

2.2 MeanShift

Given a d dimensional space R^d containing n sample points, the MeanShift vector of sampling point x_i is described as follows:

$$MeanShift(x_i) = \sum_{j \in S_h} (x_j - x_i) \tag{4}$$

where $x_j - x_i$ is the drift vector which takes x_i as the starting point and x_j as the end point. S_h denotes the hypersphere that takes x_i as its center and h as its radius, the S_h is defined as follow:

$$S_h(x_i) \equiv \{x_j : (x_j - x_i)^T (x_j - x_i) \leq h^2\} \tag{5}$$

The literature [20] has proved that the MeanShift vector always points to the highest density direction of local area. The traditional MeanShift vector holds that all the points in S_h are the same importance and the weight of their drift vectors are same. However, for the varying density and multi-density data area, the MeanShift vector may point to noise or isolated data. So the approach may not perform properly in some special situation. The literature [21] introduces the concept weight coefficient to improve formula (4), and it is defined as follows:

$$MeanShift(x_i) = \frac{\sum_{j \in S_h} w_{ij}(x_j - x_i)}{\sum_{j \in S_h} w_{ij}} \tag{6}$$

wher w_{ij} is the weight of drifting vector $(x_j - x_i)$.

Formula (6) gives a relatively small weight to the drifting vector which takes the sampling point as the starting point and noise point or isolated point as the end point, a relative big weight to the drifting vector which takes the sampling point as the starting point and core point as the end point in neighborhood. Thus it makes MeanShift theory be revised and expanded. Meanwhile, literature [21] introduces kernel function to improve the MeanShift formula (6). Assuming that there a kernel function $K(x)$ and the shadow kernel of it is defined as follows:

$$g(x) = \frac{\partial K(x)}{\partial (x)} \tag{7}$$

The shadow kernel is the first-order partial derivative of $K(x)$. It will be used in formula (9). Then, we take the kernel function to estimate the density of x_i :

$$MeanShift(x_i) = \frac{\sum_{j \in S_h} w_{ij} K||x_j - x_i||}{\sum_{j \in S_h} w_{ij}} \tag{8}$$

where $H(x_i)$ is the probability density of x_i. To calculate the location which has the highest density of the neighbors area, we get the partial derivative of $H(x_i)$:

$$\frac{\partial H(x_i)}{\partial(x_i)} = -(\frac{\sum_{j\in S_h} 2w_{ij}g||x_j - x_i||^2}{\sum_{j\in S_h} w_{ij}}) \times (\frac{\sum_{j\in S_h} w_{ij}x_j g||x_j - x_i||}{\sum_{j\in S_h} w_{ij}g||x_j - x_i||^2} - x_i) \tag{9}$$

Make $\frac{\partial H(x_i)}{\partial(x_i)} = 0$, then:

$$x_i = \frac{\sum_{j\in S_h} w_{ij}x_j g(||x_j - x_i||^2)}{\sum_{j\in S_h} w_{ij}g(||x_j - x_i||^2)} \tag{10}$$

The location is the maximum local density of sampling point neighborhood. The MeanShift vector is as follows:

$$MeanShift(x_i) = \frac{\sum_{j\in S_h} w_{ij}x_j g(||x_j - x_i||^2)}{\sum_{j\in S_h} w_{ij}g(||x_j - x_i||^2)} - x_i \tag{11}$$

From mathematical point of view, the MeanShift vector is similar to the mountain climbing algorithm. We can clearly observe that the evolutionary process of MeanShift aims to search for a local optimal solution from formula (7) to (11). Speaking from the perspective of density distribution, the MeanShift principle is a process of searching for the highest density area.

In formula (11), the MeanShift vector always remains the same window size in the iteration process, so it will lose effectiveness when it encounters the varying density and multi-density area after a series of iterations. However, data objects distribution is relatively sparse in the high dimensional space. This vector still uses a fixed volume of hypersphere for sliding window, but the total number of sample points in the hypersphere is constantly changing and has no rule, so the method can not effectively address the distribution of sample points in high dimensional space. Meanwhile, formula (11) do not completely consider the relations among all drifting vectors which generates from sampling points and sample points and ignores the geometric relations among all the drifting vectors. Due to problems of formula (11), we define the kernel function as the weight of drift vector, the new MeanShift vector is:

$$MeanShift(x_i) = \frac{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} K(||x_k - x_j||^2))(x_j - x_i)}{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} K(||x_k - x_j||^2))} \tag{12}$$

where k_{nn} is the k nearest neighbors area of x_i , and j_{nn} is the k nearest neighbors area of x_j, x_j is a data object in the k nearest neighbors collection of x_i, x_k is a data object in the k nearest neighbors collection of x_j. Then, we introduce the formula (3) as the kernel function. In other words, the density of sample is defined as the weight of corresponding drift vector:

$$\sum_{k\in j_{nn}} K(||x_k - x_j||^2) = \sum_{k\in j_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_k - x_j||}{2\sigma^2}) \tag{13}$$

So, the MeanShift is changed into:

$$MeanShift(x_i) = \frac{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_j - x_i||}{2\sigma^2})(x_j - x_i)}{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_k - x_j||}{2\sigma^2})} \tag{14}$$

3 BorderShift framework

From the perspective of pattern recognition, cluster analysis divides the data objects to two types: objects within clusters and noises. Isolated points are special noises with interesting values. For clustering, noises are redundant data, and objects within clusters are valuable. Discovering the objects within clusters and giving them reasonable class labels are the goals of cluster analysis. In cluster boundary research, boundary points are our detection objects. The neighbor distributions of core points are very uniform and they are easy to be separated. But noises always have bad impact on boundary extraction. Especially for some isolated points, they are located far from clusters, but their nearest neighbor distribution may be similar to boundary points. In contrast, the noises which are located near boundary points are easy to recognize since some neighbors of them are boundary points. So, our approach divides the data objects into four types: core point, boundary point, noise, isolated point. Now, we try to analyze these four types of data: (1) The k nearest neighbors of core points always show uniform distribution. So, the norm of MeanShift vectors of core points are relatively small and close to zero.

(2) The most k nearest neighbors of core points always are located at one side close to the clusters, the opposite side has relative few points and most of them are noise points. So the norm of MeanShift vectors of boundary points are relatively big compared to that of core points.

(3) The noise points which close to boundary points have the similar distribution compared to that of boundary points. But their norms of MeanShift vectors are

bigger than that of boundary points.
(4) The isolated points which far from clusters are sparsely distributed and irregular. It is difficult to distinguish the norms of MeanShift among boundary points and isolated points.
The definition of average distances among x_i and its k nearest neighbor objects is as follows:

$$Ak(x_i) = \frac{1}{k}\sum_{j\in k_n n} dist(x_j - x_i) \tag{15}$$

then there exists the truth:

$$Ak(Isolated) \gg Ak(noise) > Ak(Boundary) > Ak(Core) \tag{16}$$

Based on this, we try to eliminate the influence of isolated points. We take the function of $y = e^x$ further to discretize inequality (16). There are some discretization functions in common use:

$$y = e^x, y = x^2, y = lg(x) \tag{17}$$

Then we contrast the partial derivatives of the three functions:

$$\frac{\partial e^x}{\partial x} = e^x, \frac{\partial x^2}{\partial x} = 2x, \frac{\partial lg(x)}{\partial x} = \frac{1}{x}, \tag{18}$$

Their partial derivatives show their ability for discretization, under normal conditions:

$$e^x \gg 2x > \frac{1}{x}, (x > 1) \tag{19}$$

In other words, the discretization ability of $y = e^x$ is much larger than that of $y = x^2$ and $y = lg(x)$. This is the reason why we choose the function for discretization. Then we give a new extension form of MeanShift vector:

$$MeanShift(x_i) = exp(\frac{1}{k}\sum_{j\in k_{nn}} dist(x_j - x_i)) \times \frac{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_k - x_j||}{2\sigma^2})(x_j - x_i)}{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_k - x_j||}{2\sigma^2})} \tag{20}$$

and its norm is described as follows:

$$MeanShift(x_i) = exp(\frac{1}{k}\sum_{j\in k_n n} dist(x_j - x_i)) \times norm(\frac{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_k - x_j||}{2\sigma^2})(x_j - x_i)}{\sum_{j\in k_{nn}}(\sum_{k\in j_{nn}} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{||x_k - x_j||}{2\sigma^2})}) \tag{21}$$

Algorithm : BorderShift
Input: X // data set
k // number of nearest neighbors
λ_1 // serial number of boundary begin
λ_2 // serial number of boundary end
Output: S // boundary collection.

Step1: find the k nearest neighbors collections of each data object;
Step2: calculate the $Shift$ value of each data object according to formula (21) and store them in matrix α;
Step3: generate the matrix β by sorting α in ascending λ_2 times;
for i=1:1:n
if $\beta(\lambda_1) \le \alpha(i) \le \beta(\lambda_2)$
$S = S \cup x_i$
end if
end for

where $norm$ shows the modular operation for vectors. According to the value of $Shift$, we can easily distinguish the four kinds of data objects. Below, we will give the BorderShift algorithm.

The BorderShift algorithm first calculates the k neighbor collections of each data object, then calculates the value of $Shift$ for each data object and outputs boundary points according to their $Shift$ value. In this paper, the steps of BorderShift algorithm are relatively straightforward: Step 1 calculates the k nearest neighbors of each data object and the time complex is $O(nlog(n))$ by using k-tree; Step 2 calculates the $Shift$ value of each data object and the time complex is $O(n)$; Step 3 sorts α λ_2 times to generate β with the size of $1 \times \lambda_2$, which covers the top λ_2 values of α, and then outputs boundary points using $\beta(\lambda_1)$ and $\beta(\lambda_2)$. The time complex of this step is $O(n)$. In summary, the time complexity of the three steps is $O(nlog(n) + n)$. Besides this, the time complexity of BORDER, BAND, BRINK are all $O(kn^2)$. BRIM also uses the k-d tree to calculate the neighborhood objects and the time complexity is $O(nlog(n))$. So the time complexity for BORDER, BAND, BRINK are close to each other, but lower than that of BRIM and BorderShift. The main time consumption of each algorithm is the kNN collection calculation and the general calculation cost is $O(kn^2)$. Although these algorithms' time complexity is high, their time consumption will be reduced to $O(nlog(n))$ when the k-tree way is applied. So, their time complexity of reported algorithms has no differences in essence.

4 Experimental

4.1 Pretreatment and Evaluation

In this paper, we conduct a series of experiments to compare and verify our promised performance of BorderShift. Experiments of synthetic data sets and medical data sets are performed to compare different algorithms' detection abilities. Experiments of image data sets will verify the detection ability of BorderShift on several image datasets. It aims to explore the cluster boundary research in some new fields. Detailed information are:
(1) the comparison of boundary detection ability of BORDER, BRIM, BorderShifton on two 2-dimension numerical data sets who contain noises;
(2) the comparison of boundary detection ability of BORDER, BAND,BRINK,BorderShift on the medical data sets Biomed and Cancer , Colon and Prostate data sets;
(3) the detection of handwritten digits boundary and cluster center area on MNIST data set;
(4) the detection of face boundary and cluster center area on ORL data set;
(5) sorting each image by the value of Shift in ascending order on the Tiger data set.
Data preprocessing methods used in the following are reported as follows:
(a) The value of each data objects are dived 10^3.
(b) The value of each data objects are dived 10^4.
(c) Converting each image to $n \times m$ grayscale matrix G and average each dimension of gray level matrix, get the size of $1 \times m$ gray centroid matrix, then use the matrix to represent the image data and the specific way is described as follows:

$$G = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}$$

$$G \Rightarrow \left[\sum_{i=1}^{n} a_{i1} \ \sum_{i=1}^{n} a_{i2} \ \cdots \ \sum_{i=1}^{n} a_{im}\right]$$

(d) Converting each image to $n \times m$ grayscale matrix and average each dimension of gray level matrix, get the size of $1 \times (nm)$ gray centroid matrix, then use the matrix to represent the image data and the specific way is as follows:

$$G \Rightarrow \left[a_{11} \ a_{12} \ \cdots \ a_{1m} \ a_{21} \ a_{22} \ \cdots \ a_{2m} \ \cdots \ a_{nm}\right]$$

Table 1 has reported the information of used data sets and we have sorted the data sets by dimensions in ascending order. F-measure is used to evaluate the performance of each algorithm, related definition are described as follows:

$$Precision = \frac{number\ of\ correct\ boundary\ detected}{number\ of\ boundary\ detected}$$

$$Recall = \frac{number\ of\ correct\ boundary\ detected}{number\ of actual\ boundary}$$

$$F - measure = \frac{2}{1/Precision + 1/Recall}$$

The lower the accuracy rate, the weaker the detection capability of the algorithm will be. The recall rate reflects the completeness of the detection results. Accuracy rate and recall rate exists mutual restriction relations and the greater the F-measure is, the stronger the robustness of the algorithm will be.

Table 1: Information of different data sets.

Data sets	Number	Dimensions	Preprocessing way
DS1	7832	2	-
DS2	5034	2	-
Biomed	209	4	-
Cancer	699	10	-
Mnist	10000	28	(c)
Colon	62	2000	(a)
ORL	400	10304	(d)
Prostate	102	10509	(b)
Tiger	32	246440	(d)

4.2 Data description

Fig. 1(a) is a synthetic data set with varying density, including 7832 points and we call it DS1. Fig. 2(a) is a synthetic data set with four clusters, including 5034 points and we call it DS2. In order to facilitate observation, we mark the boundary detection result of BorderShift on DS1 in Fig. 1(b), DS2 in Fig. 2(b). Then we show the different best cluster boundary detection results of different algorithms on the two data sets. Meanwhile, we provide the used parameters of different algorithms behind the figure caption. Before the experiments, we take the statistical experiments on DBSCAB to get 640 cluster boundary points on DS1, 538 cluster boundary points on DS2. The results have been reported in Table 2.

With the completion of human genetic planning groups [24] and the rapid development of gene chip technology [25], gene expression data [26] has been widely used in the field of tumor. Gene expression data tends to have the characteristics of high dimensions and small number of samples. Many scholars pay their attentions on clustering these data sets. But there are fewer researches for

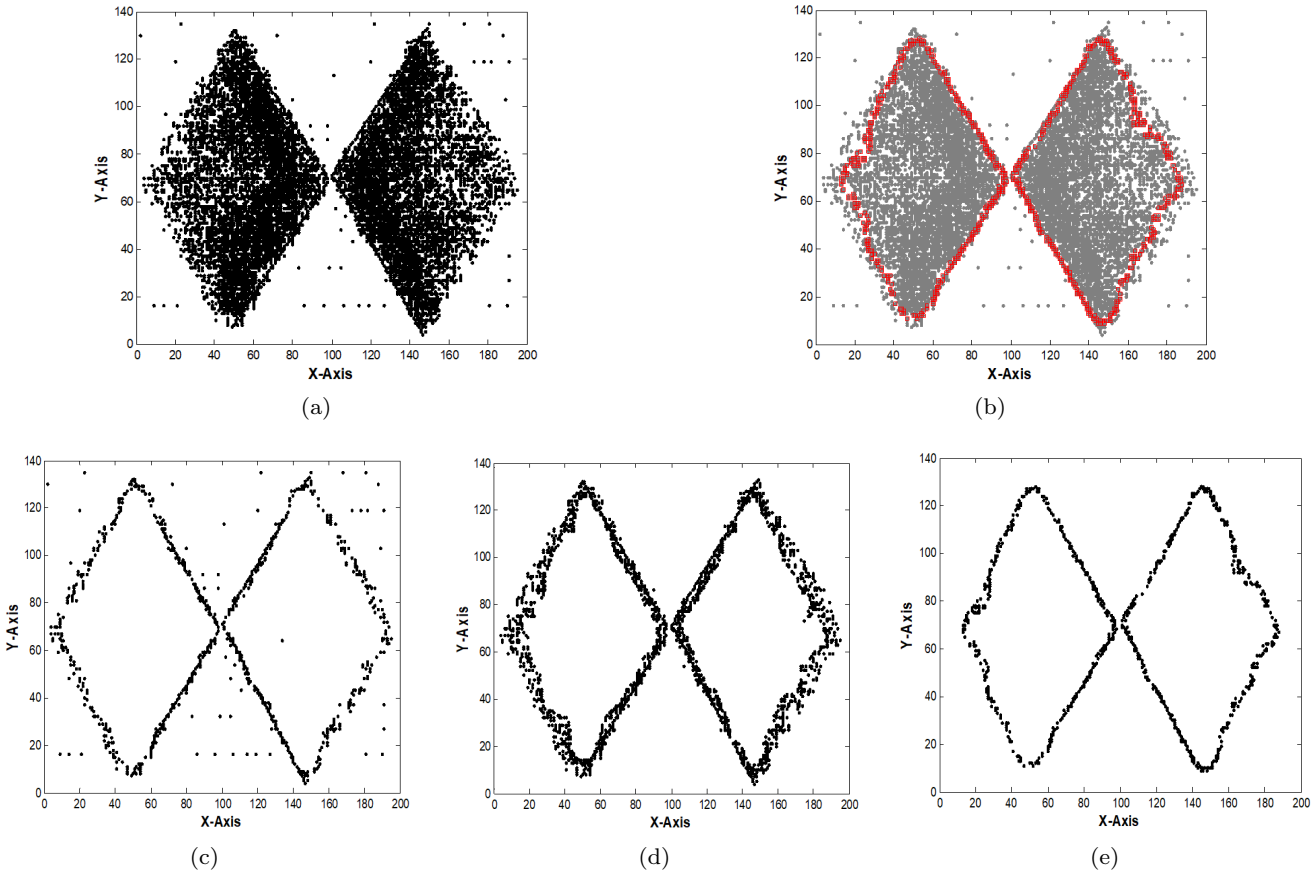


Fig. 1: The best cluster boundary detection results of different algorithms on DS1. (a)DS1. (b) The marked cluster boundary points. The used parameter of each algorithm are (c)BORDER ($k = 120, n = 1200$), (d) BAND ($k = 50, w = 0.65, BPT = 0.66$), (e)BorderShift ($k = 100, \lambda_1 = 530, \lambda_2 = 1197$).

extracting the individuals of gene mutation by the similarity among gene segments. These individuals usually exist the risk of suffering from tumor, many of them will die of cancer after some years. We define these people as cluster boundary objects of normal people. If we take treatment for these people, we can control their illness state and even cure them. So it will be a research field with profound significance. Biomed data set [39] has 134 normal objects and 75 virus infected objects. But there are 30 virus carriers in the normal objects. Cancer data set [40] has 241 malignant tumor objects and 75 benign tumor objects. But there are 30 benign tumor objects which may become malignant tumor patients, so these people can be defined as the cluster boundary of normal people. Colon [41] is a colon cancer gene expression data set with 62 samples, including 22 normal gene samples and 40 colon cancer samples. In addition, each sample has 2000 genes. Prostate [42] is also a gene data set which has 102 samples, including 50 norm samples and 52 prostate cancer samples. In this data set, each sample has 10509 genes. Before experiments, we take the statistical experiments on DBSCAB to get 7 cluster boundary objects on Colon, 18 cluster boundary objects on Prostate. Then we process these data sets as described in Table 1.

Handwritten digit recognition [27-29] plays an important role in the field of artificial intelligence. With a series of theories and application techniques being proposed, handwritten digits recognition has been used widely in tax bills, statistical statements etc.. Due to the influence of personal preferences and habits, same digit images always be presented in different shapes, sizes, line widths, etc. Sometimes, it may even appear synechia, overlapping, ink, etc., so these digits images increase the difficult of handwritten digit recognition. Nowadays, there are fewer researches on the cluster boundary of handwritten digits, these images have irregular geometric characteristics, larger similarity between digits, and are difficult to be distinguished among digits and English characters. For example, digit characters '0' and digit characters 'o', digit characters '5', English character 's' are these characters that have some

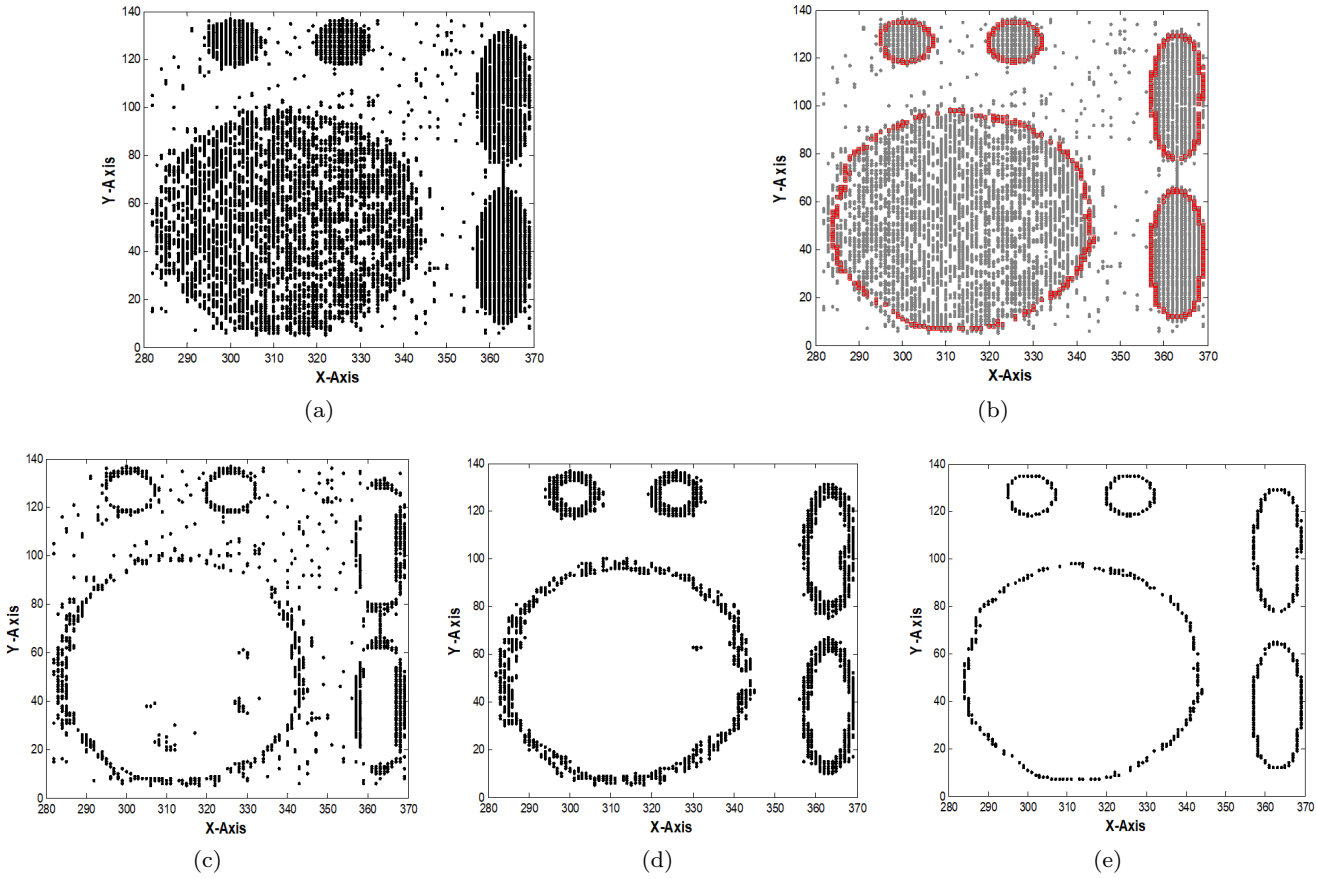


Fig. 2: The best cluster boundary detection results of different algorithms on DS2. (a)DS2. (b) The marked cluster boundary points. The used parameter of each algorithm are (c) BORDER ($k = 120, n = 1200$), (d) BAND ($k = 50, w = 0.40, BPT = 0.80$), (e)BorderShift ($k = 60, \lambda_1 = 543, \lambda_2 = 1078$).

similarities between each other. Effectively extracting the cluster boundary of handwritten digits provides an important reference for handwritten recognition. In other words, this work will contribute to extract the characteristics and improve the cluster accuracy of handwritten digits. Mnist data set [43] includes 10 types of handwritten digits, including 60000 training image samples and 10000 test image samples. There are 8 bit depth of BMP image for all the image size, stored in the form of 28×28 pixel size, and each pixel gray value is in the range of 0-255. We select the handwritten digit '8' from the test image samples, a total of 974 images, to verify the cluster boundary detection ability of BorderShift.

Face recognition [30-33] technology is a set of comprehensive research direction in the field of computer image processing, computer vision, human-computer interaction, etc. and plays a very important role in intelligent transportation [34], remote sensing image [35], criminal investigation, financial, military defense and so on. Compared to normal face, face boundary objects can be defined as face images with strong illumination, faint illumination, sunglasses, profile face, etc. The images affect the accuracy of face recognition significantly, so it will provide an important reference for the feature extraction and recognition accuracy. ORL data set [44] consists with 400 images of 40 different people that each covers a range of poses from profile to frontal views. There are 8 bit depth of BMP image for all the image size, stored in the form of 92×112 pixel size, and each pixel gray value is in the range of 0-255.

Tiger data set is a biological data set which built by the Data Mining Team of Zhengzhou University. The training set has 9 different tiger head portraits, 2 leopard head portraits, 1 civet head portraits. The test set has 16 different tiger head portraits, 2 leopard head portraits, 1 civet head portrait, 1 lion head portrait. All the images of the data set are collected from the internet, scaled with uniform size, stored in PNG format with 505*488 pixel size. We all know that tigers, leopards, civets, lions all are felid animal and there are close kinship and high similar face between tiger and leopard. So the leopards and civet can be defined as the cluster

boundary of tigers. Especially, leopards have more close relations with tigers than civets and lions. By analyzing the close relations among species, we can further research the potential relevance of different species. It will have a positive impact on endangered species protection [36], bio-archaeologists [37], animal classification [38], etc.

5 Results and discussions

We first evaluate the effects of varying the density and dimension of the dataset. Fig. 1 and Fig. 2 show the experiment results on the synthetic dataset. As we can see, BORDER and BAND are both sensitive to the noises, they detect many noise points as boundary points. While our proposed BorderShift outperforms other two algorithms, it can distinguish the boundary points and noise points effectively. Furthermore, F-measure analysis of Table 2 also shows BorderShift has better detection ability than the proposed low dimensional cluster boundary detection algorithms, which further verifies the effectiveness of our improved Parzen window density technique. Compared to traditional density estimation methods, our proposed method increases the differences between core points and noise points via Knn dynamic window. This is our first contribution in this paper.

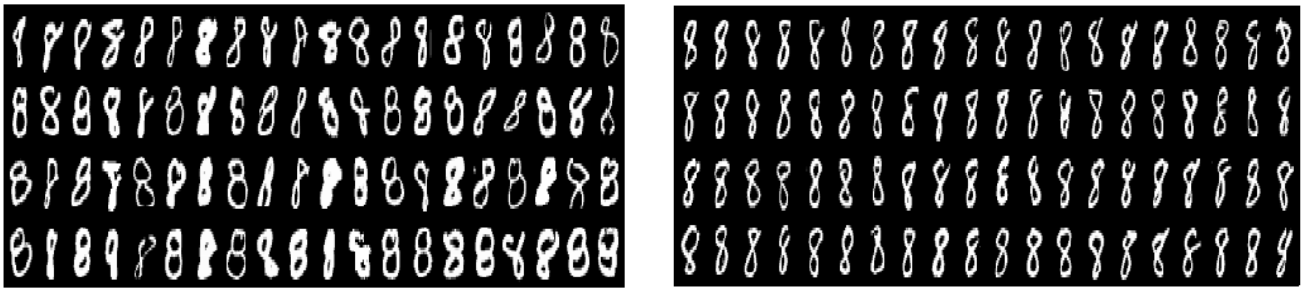
Table 2 reports the experiment results on several real high dimensional datasets. By observing the F-measure, we find that the ability of cluster boundary detection of BorderShift is more effective than that of other algorithms on Cancer and Biomed. It's worth noting that the recall of BRINK on Biomed is 1.0000, which does not mean its detection ability is good. Because the recall rate only reflects the number of real boundary points in the detecting result. On the other hand, if the algorithm detects many data objects as cluster boundary, the accuracy rate of the result must be low. Thus neither of the recall rate and the accuracy rate can reflect the quality of the result. For Colon and Prostate, each algorithm can effectively detect the boundary because of the sparsity and non-noises of these samples. MeanShift vector is used to track the moving target in computer vision. The sensitivity of noise points, however, makes the MeanShift vector always move into sparse areas. To solve this, we adopt our novel Parzen window to smooth noises and reduce the interference of noises. Interestingly, we try to use the models of MeanShift vector to separate boundary points according their lengths. Compared to the traditional algorithm, our proposed BorderShift can detect the boundary effectively on both high and low dimensional datasets, which can be conducted from the F-measure values. This is our second contribution in this paper.

Our third contribution is to apply our work in several real scenarios. For instance, we attempt to detect the cluster boundary objects in handwritten digits, face images, and animal recognition. In Fig. 3, we find BorderShift can effectively detect the boundary of 8 and the standard normalized digits as the cluster center objects. Fig. 4 is the result of sorting each face image by the value of $Shift$ in ascending on a face cluster of ORL. Through the observation, the first three images can be used as cluster center and the farther from the cluster center, the greater the angle of profile face. Fig. 5(a) is the boundary detection result on ORL, including the face images of left profile face and right profile face. Fig. 5(b) are the cluster center objects of BorderShift algorithm on ORL , and the results almost are frontal faces. These studies will improve the recognition accuracy and provide a novel method to resolve such problems. In the experiments on Tiger dataset, we sort the images by the $Shift$ in an ascending order, and select the last three images as the boundary objects of the training set (see Fig. 6(a)). Meanwhile, we sort the images with in same way on the test set (see Fig. 6(b)), and the last four images are the boundary objects. The leopards are more similar to tigers than civets and lions in the facial features. Due to much hair of the lion, it performs relative alienation with tigers. The experiment results satisfy the biology logic relationship among the above species. It also shows the effective of BorderShift on cluster boundary detection. Significantly, our cluster boundary research on animals will benefits related research fields. Traditional image and computer vision methods in artificial intelligent always use the face and head characteristics to recognize objects such as the locations of eyes, nose, and ears, or color characteristics of image. Capturing the local features is their primary idea. While in this paper, we adopt the cluster boundary technique to solve the image recognition problems, which based the similarity of image objects.

Next, we discuss the effects of the parameters. BorderShift has three parameters: k, λ_1, λ_2. From all the above experimental results, we may observe that BorderShift achieves high quality result when $k \in$[50,100] on density and multi-density datasets, and $k \in$[2,10] for general data sets. Thus we suggest k=10 to get the detecting result. To select the parameters reasonably, we compare the F-measure when tunning different k. Finally, we discuss the sensitivity of k of BorderShift. Fig. 7 shows the change of F-measure whenselecting different values of k. The meanings of λ_1, λ_2, are described as follows':
(1) if $\beta(\lambda_2) \leq \alpha(i) \leq \beta(\lambda_1)$, x_i is boundary object and

Table 2: The boundary detection results of different algorithms on different data sets.

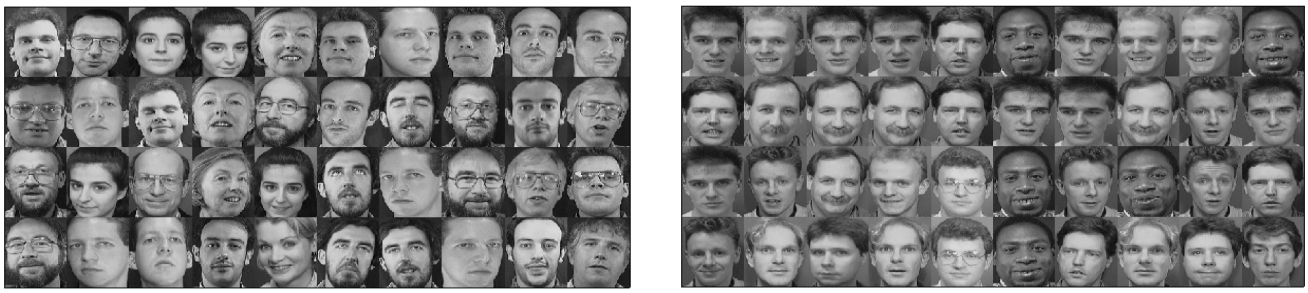
Data sets	Algorithms	Real	Detected	Correct	Precision	Recall	F-measure
DS1	BAND	640	823	556	0.6756	0.8688	0.7601
	BORDER		723	540	0.7469	0.8438	0.7924
	BRINK		667	520	0.7795	0.8125	0.7957
	BERGE		662	532	0.8036	0.8313	0.8172
	BorderShift		680	578	0.8500	0.9031	**0.8757**
DS2	BAND	538	749	454	0.6061	0.8439	0.7055
	BORDER		669	445	0.6366	0.8271	0.7195
	BRINK		499	438	0.8778	0.8141	0.8447
	BERGE		553	472	0.8535	0.8773	0.8652
	BorderShift		599	514	0.8581	0.9554	**0.9041**
Biomed	BAND	30	26	22	0.8462	0.7333	0.7857
	BORDER		26	23	0.8846	0.7667	0.8214
	BRINK		36	30	0.8333	1.0000	0.9089
	BERGE		26	24	0.9231	0.8000	0.8572
	BorderShift		30	28	0.9333	0.9333	**0.9333**
Cancer	BAND	37	37	25	0.6757	0.6757	0.6757
	BORDER		37	28	0.7568	0.7568	0.7568
	BRINK		37	29	0.7837	0.7837	0.7837
	BERGE		37	28	0.7568	0.7568	0.7568
	BorderShift		36	35	0.9722	0.9459	**0.9589**
Colon	BAND	7	6	5	0.8333	0.7143	0.7692
	BORDER		7	7	1.0000	1.0000	**1.0000**
	BRINK		6	5	0.8333	0.7143	0.7692
	BERGE		6	5	0.8333	0.7143	0.7692
	BorderShift		7	7	1.0000	1.0000	**1.0000**
Prostate	BAND	18	17	16	0.9412	0.8889	0.9143
	BORDER		19	18	0.9474	1.0000	0.9730
	BRINK		17	16	0.9412	0.8889	0.9143
	BERGE		17	16	0.9412	0.8889	0.9143
	BorderShift		18	18	1.0000	1.0000	**1.0000**



(a) (b)

Fig. 3: The expriments on the Mnist data set. (a)The boundary objects of '8' ($k = 10, \lambda_1 = 895, \lambda_2 = 974$). (b)The cluster center objects of '8'($k = 10, \lambda_1 = 1, \lambda_2 = 80$).

Fig. 4: The result of sorting each face by the value of $Shift$ in ascending on the face cluster.



(a) (b)

Fig. 5: The expriments on the ORL data set. (a)Boundary objects ($k = 10, \lambda_1 = 361, \lambda_2 = 400$). (b)Cluster center objects ($k = 10, \lambda_1 = 1, \lambda_2 = 40$).

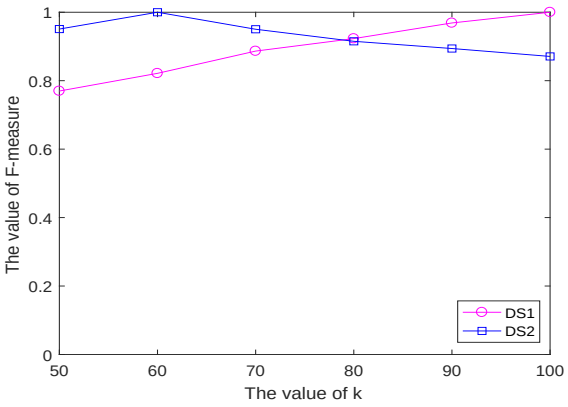
(a) (b)

Fig. 6: The result of sorting each image by the value of $Shift$ in ascending on the training set and test set of Tiger data set, respectively.

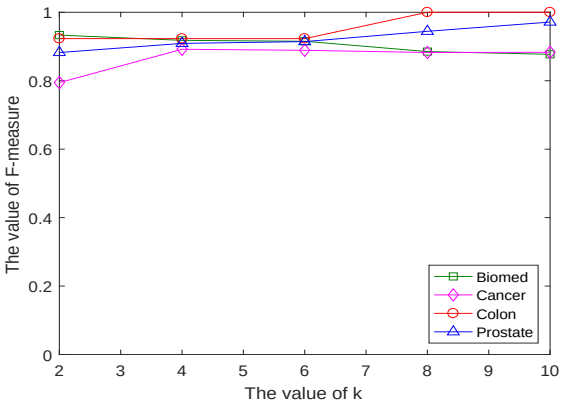
the number of boundary objects is $\lambda_2 - \lambda_1 + 1$;
(2) if $\alpha(i) \geq \beta(\lambda_2)$, x_i is a noise object with the number of $n - \lambda_2$;
(3) if $\alpha(i) \leq \alpha(\lambda_1)$, x_i is a core object with the number of $\lambda_1 - 1$.
Obviously, BorderShift can control the number of these three kinds of objects well, which is beneficial for users to select λ_1 and λ_2.

The above explanation describes the three types of data objects using the two input parameters. Next, we will analyze our proposed detection approach in different data distribution settings. According to the point of clustering, any data set can be clustered into certain classes. In fact, it is a density recognition process. Different density area are captured as different class structures. For the uniform class without noises, our model is very strong since there are only core and cluster boundary objects in the data set. The neighborhood distribution of core objects are uniform and the MeanShift vector models are close to zero. Completely different from this is that most neighbors of boundary objects are core objects because no noises are distributed in the data set. So, its MeanShift vector will point to class interior with a large model. When the data set has noises, the MeanShift vector model of boundary objects will reduce, but there are no influence on core objects. In the multi-density classes, the uniform is no related with the density. The neighbors of core objects still uniformly surrounded them and construct a Mean-

Sift vector with a small model. Meanwhile, most of the neighbors of boundary points are still core objects and only small number of them are noises. So, we can conclude that our idea can be applied in different data distribution settings and the improved MeanShift vector still will work.



(a)



(b)

Fig. 7: The sensitivity of k on different data sets.

6 Conclusion

In this paper, we propose a cluster boundary detection algorithm based on Parzen window and MeanShift. To smooth noises, we use kNN widow to replace the sliding window with fixed size in the density estimation process. New MeanShift vector based on dynamic sampling also reduce the sensitivity to noise. Then it can be effectively used to detect the cluster boundary in varying density, multi-density, high dimensional space data sets. The algorithm can obtain the detection results without many iterations. Meanwhile we have attempted to detect cluster boundary in different fields and get expected result. We have extended the cluster boundary application scope to some extent.

In future work, we will test the algorithm performance in real dataset with higher dimensions. Time consumption will be a worth attention work. Now, there are rich data sets with high dimension and large number available in the real world, therefore cluster boundary research in big data will be a valuable work. How to separate boundary objects from big data quickly will be challenging. For example, detecting the virus carriers from blood bank data, recognizing the abnormal signature, updating the electronic map etc..

7 Appendix

$$
\begin{aligned}
&\frac{\partial H(x_i)}{\partial (x_i)} \\
&= -(\frac{\sum_{j\in S_h} 2w_{ij}(x_j - x_i)g||x_j - x_i||^2}{\sum_{j\in S_h} w_{ij}}) \\
&= -(\frac{\sum_{j\in S_h} 2w_{ij}g||x_j - x_i||^2}{\sum_{j\in S_h} w_{ij}}) \\
&\quad \times (\frac{\sum_{j\in S_h} w_{ij}x_j g||x_j - x_i|| - w_{ij}x_i g||x_j - x_i||}{\sum_{j\in S_h} w_{ij}g||x_j - x_i||^2}) \\
&= -(\frac{\sum_{j\in S_h} 2w_{ij}g||x_j - x_i||^2}{\sum_{j\in S_h} w_{ij}}) \\
&\times (\frac{\sum_{j\in S_h} w_{ij}x_j g||x_j - x_i||}{\sum_{j\in S_h} w_{ij}g||x_j - x_i||^2} - \frac{\sum_{j\in S_h} w_{ij}x_i g||x_j - x_i||}{\sum_{j\in S_h} w_{ij}g||x_j - x_i||^2}) \\
&= -(\frac{\sum_{j\in S_h} 2w_{ij}g||x_j - x_i||^2}{\sum_{j\in S_h} w_{ij}}) \\
&\quad \times (\frac{\sum_{j\in S_h} w_{ij}x_j g||x_j - x_i||}{\sum_{j\in S_h} w_{ij}g||x_j - x_i||^2} - x_i)
\end{aligned}
$$

References

1. A. Faktor; M. Irani, Clustering by CompositionUnsupervised Discovery of Image Categories. IEEE Transactions on Pattern Analysis and Machine Intelligence, 36(6), (2014) 1092 1106.
2. Y.X. Chen; J. Z. Wang; R. Krovetz, CLUE: cluster-based retrieval of images by unsupervised learning. IEEE Transactions on Image Processing, 14(8) (2005) 1187 1201.
3. Y. Horng; S.M. Chen; Y.C. Chang; C.H. Lee, A new method for fuzzy information retrieval based on fuzzy hierarchical clusteringand fuzzy inference techniques, IEEE Transactions on Fuzzy Systems, 13(2) (2005) 216 228.
4. Q. Li, Y. P. Chen, Personalized text snippet extraction using statistical language models. Pattern Recognition, 43(1) (2010) 378-386.
5. S. Kim; C. D. Yoo; S. Nowozin; P. Kohli, Image Segmentation Using Higher-Order Correlation clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 36 (9) (2014) 1761 1774.

6. N. Hohn; D. Veitch; P. Abry, Cluster processes: a natural language for network traffic. IEEE Transactions on Signal Processing, 51(8) (2003) 2229 2244.
7. B.Z. Qiu, Y. F, J.Y. Shen, BRIM: An efficient boundary points detecting algorithm[C]. Advances in Knowledge Discovery and Data Mining. Springer Berlin Heidelberg. (2007) 761-768.
8. L.X. Xue, B. Z. Qiu, Boundary Points Detection Algorithm Based on Coefficient of Variation[J]. Pattern Recognition and Artificial Intelligence. 22(5) (2009) 799-802.
9. B.Z. Qiu, Y. Yang, X.W. Du, BRINK: An Algorithmof Boundary Points of Clusters Detection Based On Local Qualitative Factors[J]. Journal of Zhengzhou University (Engine-ering Science).
10. B.Z. Qiu, H.L. Cao, An efficient boundary points detecting algorithm based on joint entropy, Control and Decision. 1 (2011) 71-74.
11. L.X. Li, P. Geng, B.Z. Qiu, clusteringboundary detection technology for mixed attribute data set, Kongzhi yu Juece/Control and Decision, 2015, 30(1):171-175.
12. S.T. Roweis, L.K. Saul, Nonlnear dimensionality reduction by locally linear embedding. Science. 290 (5500) (2000) 2323-2326.
13. X. He, P. Niyogi. Locality preserving projections. In: Thrun S, Saul L, Schlkopf B, eds, Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2003.
14. M. Ester, H.P .Kriegel, J. Sander, et al, A density-based algorithm for discovering clusters in large spatial databases with noise[C]. KDD. 96 (1996) 226-231.
15. C.Y. X, W. Hsu, M.L. Lee, et al, BORDER: An efficient computation of boundary points, Knowledge and Data Engineering, IEEE Transactions. 3 (2006) 289-303.
16. V.N. Vapnik, An overview of statistical learning theory. IEEE Transactions on Neural Networks, 10(5) (1995) 988-999.
17. M. A .Hearst, S. T. Dumais, E. Osman, J. Platt, Scholkopf B.Sup-port Vector Machines, IEEE Intelligent Systems. 13(4) (1998) 18-28.
18. P. Emanuel, On the estimation of a probability density function and the mode, Ann. Math. Stat. 32 (1962) 10651076.
19. K. Fukunaga, L.D. Hostetler, The Estimation of the Gradient of a Density Function[J]. IEEE Trans. Information Theory. 21 (1975) 32-40.
20. Y.Z. Cheng. MeanShift, Mode seeking, and clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 17(8): (1995) 790-799.
21. D. Comaniciu, P. Meer, MeanShift Analysis and Application[C]. Proceedings of the International Conference on Computer Vision. (1999) 1197-1204.
22. X.G. Chen, G.H. Xu, F. Liu, X. Wan, Q. Zhang,An Adaptive Alarm Method for Tool Condition Monitoring Based on Probability Density Functions Estimated with the Parzen Window. Engineering Asset Management - Systems, Professional Practices and Certification Lecture Notes in Mechanical Engineering. (2015) 1-8.
23. V. Vikjord, R. Jenssen, Information theoretic clustering using a k-nearest neighbors approach. Pattern Recognition. 47 (2014) 30703081.
24. J. Ryu, K. Park, Planning Rehabilitation Strategy of Sewer Asset Using Fast Messy Genetic Algorithm. 9th WCEAM Research Papers Lecture Notes in Mechanical Engineering. (2015) 61-71.
25. H. Zimdahl , N. Hbner, Gene Chip Technology and Its Application to Molecular Medicine. Encyclopedic Reference of Genomics and Proteomics in Molecular Medicine.(2006) 650-655.
26. P. A. Jaskowiak , R.J. Campello, Ivan G Costa, On the selection of appropriate distances for gene expression data clustering. BMC Bioinformatics, 2014.
27. Q. Zhu, H. Xin, Feature Extraction and Filter in Handwritten Numeral Recognition. Geo-Informatics in Resource Management and Sustainable Ecosystem. Communications in Computer and Information Science. 398 (2013) 58-67.
28. M. Juan, A. Weber, M. Paz Sesmero , Input Transformation and Output Combination for Improved Handwritten Digit Recognition. Artificial Neural Networks. Springer Series in Bio-/Neuroinformatics. 4 (2015) 435-443.
29. Y.M. Wang, Alexander Peyls, Y. Pan, Luc Claesen, X.L. Yan, A Fast Self-Organizing Map Algorithm for Handwritten Digit Recognition. Multimedia and Ubiquitous Engineering. Lecture Notes in Electrical Engineering. 240(2013) 177-183.
30. S.C. Huang, J. Chen, Z. Luo, Retraction Note to: Sparse tensor CCA for color face recognition. Neural Computing and Applications. December 2014, Volume 25, Issue 7-8, p 2091.
31. B. Bhaskar, K. Mahantesh, G. P. Geetha, An Investigation of fSVD and Ridgelet Transform for Illumination and Expression Invariant Face Recognition. Advances in Intelligent Informatics. Advances in Intelligent Systems and Computing. 320 (2015) 31-38.
32. K.D. Dang, Thai Hoang Le, Local Region Partitioning for Disguised Face Recognition Using Non-negative Sparse Coding. Advanced Methods for Computational Collective Intelligence. Studies in Computational Intelligence .457 (2013) 197-206.
33. H.J. Zang, S. Zhan, M.J. Zhang, J.J Zhao, Z.C. Liang, 3D Face Recognition by Collaborative Representation Based on Face Feature. Biometric Recognition. Lecture Notes in Computer Science. 8833 (2014) 182-190.
34. O. Mitrea, System Dynamics Modeling of Intelligent Transportation Systems Human and Social Requirements for the Construction of Dynamic Hypotheses. Selected Topics in Nonlinear Dynamics and Theoretical Electrical Engineering Studies in Computational Intelligence. 459 (2013) 191-206.
35. W.Y. Ge, P.W. Li, A Fuzzy Expectation-Maximization Algorithm of Electronic Remote-Sensing Image. Advances in Mechanical and Electronic Engineering. Lecture Notes in Electrical Engineering. 178 (2013) 323-329.
36. K. P. Sheldon, Navigating for Noah: Setting New Directions for Endangered Species Protection in the 21st Century. Saving Biological Diversity. (2008) 21-33.
37. T. Colard, B. Bertrand, S. Naji, Y. Delannoy, A. Bcart. Toward the adoption of cementochronology in forensic context. International Journal of Legal Medicine, 2015.
38. C. Colombo, J.H. Leopold, R. Bellazzi, A, Abu-Hanna. Comparison of Probabilistic versus Non-probabilistic Electronic Nose Classification Methods in an Animal Model. Artificial Intelligence in Medicine. Lecture Notes in Computer Science. 9105 (2015) 298-303.
39. Biomed data set: http://lib.stat.cmu.edu/datasets/
40. Cancer data set:http://archive.ics.uci.edu/ml/datasets.html
41. Colon data set:http://genomics-pubs.princeton.edu/ oncology/affydata/
42. Prostate data set:http://www.gems-system.org/
43. Mnist data set: http://yann.lecun.com/exdb/mnist/
44. PRL data set: http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html