Mean shift-based clustering
Kuo-Lung Wu, Miin-Shen Yang

Department of Information Management, Kun Shan University of Technology, Yung-Kang, Tainan 71023, Taiwan, ROC
Department of Applied Mathematics, Chung Yuan Christian University, Chung-Li 32023, Taiwan, ROC

Received 3 June 2006; received in revised form 3 February 2007; accepted 19 February 2007

Abstract
In this paper, a mean shift-based clustering algorithm is proposed. The mean shift is a kernel-type weighted mean procedure. Herein, we first discuss three classes of Gaussian, Cauchy and generalized Epanechnikov kernels with their shadows. The robust properties of the mean shift based on these three kernels are then investigated. According to the mountain function concepts, we propose a graphical method of correlation comparisons as an estimation of defined stabilization parameters. The proposed method can solve these bandwidth selection problems from a different point of view. Some numerical examples and comparisons demonstrate the superiority of the proposed method including those of computational complexity, cluster validity and improvements of mean shift in large continuous, discrete data sets. We finally apply the mean shift-based clustering algorithm to image segmentation.

Keywords: Kernel functions; Mean shift; Robust clustering; Generalized Epanechnikov kernel; Bandwidth selection; Parameter estimation; Mountain method; Noise

1. Introduction
Kernel-based methods are widely used in many applications [1,2]. There are two ways of implementing kernel-based methods along with supervised and unsupervised learning. One way is to transform the data space into a high-dimensional feature space $F$ where the inner products in $F$ can be represented by a Mercer kernel function defined on the data space (see Refs. [1–5]). An alternative way is to find a kernel density estimate on the data space and then search the modes of the estimated density [6]. The mean shift [7,8] and the mountain method [9] are two simple techniques that can be used to find the modes of a kernel density estimate.

Fukunaga and Hostetler [7] proposed the mean shift procedure based on asymptotic unbiasedness, consistency and uniform consistency of a nonparametric density function gradient estimate using a generalized kernel approach. This technique has been applied in image analysis [10,11], texture segmentation [12,13], objective tracking [14–16] and data fusion [17]. Cheng [8] clarified the relationship between mean shift and optimization by introducing the concept of shadows. He showed that a mean shift is an instance of gradient ascent with an adaptive step size. Cheng [8] also proved some of the convergence properties of a blurring mean shift procedure and showed some peculiar behaviors of mean shift in cluster analysis with the most used Gaussian kernel. Moreover, Fashing and Tomasi [18] showed that mean shift is a bound optimization and is equivalent to Newton’s method in the case of piecewise constant kernels.

Yager and Filev [9] proposed mountain methods to find the approximate modes of the data set via the mountain function, which is equivalent to the kernel density estimate defined on the grid nodes. Chiu [19] modified the mountain method by defining the mountain function, not on the grid nodes, but on the data set. Recently, Yang and Wu [20] proposed a modified mountain method to identify the number of modes of the mountain function. The applications of the mountain method can be found in Refs. [21,22]. In this paper, we will propose a mean shift-based clustering method (MSCM) using the concept of mountain functions to solve the bandwidth selection problem in the mean shift procedure.

The bandwidth selection for a kernel function directly affects the performance of the density estimation. It also heavily...
influences the performance of the mean shift procedure. The modes found by the mean shift do not adequately present the dense area of the data set if a poor bandwidth estimate is used. Comaniciu and Meer [11] summarized four different techniques according to statistical analysis-based and task-oriented points of view. One practical bandwidth selection technique is related to the stability of decomposition for a density shape estimate. The bandwidth is taken as the center of the largest operating range over which the same number of clusters are obtained for the given data [23]. Similar concepts are used by Beni and Liu [24] to estimate the resolution parameter of a least biased fuzzy clustering method, and also to solve the cluster validity problem. In this paper, we will offer an alternative method of solving this problem. We first normalize the kernel function and then estimate the defined stabilization parameter. This estimation method will be discussed in Section 3 based on the mountain function concept.

The properties of the mean shift procedures are reviewed in Section 2.1. We discuss some special kernels with their shadows and then define generalized Epanechnikov kernels in Section 2.2. Note that most of mean shift-based algorithms are less likely to include the property of robustness [25] that is often employed in clustering [26–30]. In Section 3.1, we will discuss the relation between the mean shift and the robust statistics based on the discussed kernel functions. In Section 3.2, we proposed an alternative technique to solve the bandwidth selection problem which solved the problem from a different point of view. The purpose of the bandwidth selection is to find a suitable bandwidth (covariance) for a kernel function of a data point so that a suitable kernel function will induce a good density estimate. Our technique assigns a fixed sample variance for the kernel function so that a suitable stabilization parameter can induce a satisfactory density estimate. We then propose a technique to estimate the stabilization parameter using an adaptive mountain function in Section 3.3. According to our analysis, we propose the mean shift-based clustering algorithm based on the defined generalized Epanechnikov kernel in Section 3.4. Some numerical examples, comparisons and applications are stated in Section 4. These include the computational complexity, cluster validity, improvements in large continuous and discrete data sets and image segmentation. Finally, conclusions are given in Section 5.

2. Mean shift

Let \( X = \{x_1, \ldots, x_n\} \) be a data set in an \( s \)-dimensional Euclidean space \( \mathbb{R}^s \). Camasta and Verri [3] and Girolami [4] had recently considered kernel-based clustering for \( X \) in the feature space where the data space is transformed to a high-dimensional feature space \( F \) and the inner products in \( F \) are represented by a kernel function. On the other hand, the kernel density estimation with the modes of the density estimate over \( X \) is another kernel-based clustering method based on the data space [6]. The modes of a density estimate are equivalent to the location of the densest area of the data set where these locations could be satisfactory cluster center estimates. In the kernel density estimation, the mean shift is a simple gradient technique used to find the modes of the kernel density estimate. We first review the mean shift procedures in the next subsection.

2.1. Mean shift procedures

Mean shift procedures are techniques for finding the modes of a kernel density estimate. Let \( H : X \rightarrow R \) be a kernel with \( H(x) = h(\|x - x_j\|^2) \). The kernel density estimate is given by

\[
\hat{f}_H(x) = \sum_{j=1}^{n} h(\|x - x_j\|^2)w(x_j),
\]

where \( w(x_j) \) is a weight function. Based on a uniform weight, Fukunaga and Hostetler [7] first gave the statistical properties including the asymptotic unbiasedness, consistency and uniform consistency of the gradient of the density estimate given by

\[
\nabla \hat{f}_H(x) = 2 \sum_{j=1}^{n} (x - x_j)h'(\|x - x_j\|^2)w(x_j).
\]

Suppose that there exists a kernel \( K : X \rightarrow R \) with \( K(x) = k(\|x - x_j\|^2) \) such that \( h'(r) = ck(r) \) where \( c \) is a constant. The kernel \( H \) is termed a shadow of kernel \( K \) (see Ref. [8]). Then

\[
\nabla \hat{f}_H(x) = \sum_{j=1}^{n} k(\|x - x_j\|^2)(x_j - x)w(x_j)
\]

\[
= \left[ \sum_{j=1}^{n} k(\|x - x_j\|^2)w(x_j) \right] x_j - x
\]

\[
= \hat{f}_K(x)[m_K(x) - x].
\]

The term \( m_K(x) - x = \nabla \hat{f}_H(x)/\hat{f}_K(x) \) is called the generalized mean shift which is proportional to the density gradient estimate. Formulation (2) was first remarked upon in Ref. [11] with the uniform weight case. Taking the gradient estimator \( \nabla \hat{f}_H(x) \) to be zero, we derive a mode estimate as

\[
x = m_K(x) = \frac{\sum_{j=1}^{n} k(\|x - x_j\|^2)w(x_j)x_j}{\sum_{j=1}^{n} k(\|x - x_j\|^2)w(x_j)},
\]

where \( H \) is a shadow of kernel \( K \). Eq. (3) is also called the weighted sample mean with kernel \( K \). The mean shift has three kinds of implementation procedures. The first is to set all data points as initial values and then update each data point \( x_j \) with \( m_K(x_j), j = 1, \ldots, n \) (i.e. \( x_j \leftarrow m_K(x_j) \)). This procedure is called a blurring mean shift. For the blurring process, each data point \( x_j \) is updated with each iteration. Hence the density estimator \( \hat{f}_H(x) \) is also changed. The purpose of the second one is still to set all data points as initial values, but we only update the data point \( x \) with \( m_K(x_j) \) (i.e. \( x \leftarrow m_K(x_j) \)). This procedure is called a nonblurring mean shift. In this process, most data points and the density estimate are not updated. The third one is to choose \( c \) initial values where \( c \) may be greater or smaller.
than \(n\) and then update the data point \(x\) with \(m_K(x_j)\) (i.e., \(x \leftarrow m_K(x_j)\)). This procedure is called a general mean shift. Cheng [8] proved some convergence properties of the blurring mean shift process using Eq. (3). Moreover, Comaniciu and Meer [11] also gave some properties for the discrete data. They also discussed its relation to the Nadaraya–Watson estimator from the kernel regression and robust \(M\)-estimator points of view.

If there is a shadow \(H\) of kernel \(K\), then the mean shift procedure could ascertain modes of a known density estimate \(f_H(x)\). This technique can be used to directly find the modes of the density shape estimate. If a shadow of kernel \(K\) does not exist or it has not been found yet, the mean shift can be used to estimate the alternative modes (cluster centers) of the given data set with an unknown density function.

### 2.2. Some special kernels and their shadows

In this subsection, we investigate those special kernels with their shadows. The most commonly used kernels that are their own shadows are the Gaussian kernels \(G^p(x)\) defined as

\[
G^p(x) = [g((x - x_j)^2)]^p = [\exp(-[(x - x_j)^2]/\beta)]^p
\]

with their shadows \(SG^p\) defined as

\[
SG^p(x) = G^p(x), \quad p > 0.
\]

This means that the mean shift procedures with \(x \leftarrow m_{G^p(x_j)}\) are used to find the modes of the density estimate \(f_{SG^p}(x)\). Cheng [8] showed some behaviors of mean shift in cluster analysis with a Gaussian kernel. The maximum entropy clustering algorithm [31,32] is a Gaussian kernel-based mean shift with a special weight function. Chen and Zhang [33] used the Gaussian kernel-induced distance measure to implement the spatially constrained fuzzy c-means (FCM) [34] as a robust image segmentation method. Yang and Wu [35] directly used \(f_{SG^p}(x)\) as a total similarity objective function. They then derived a similarity-based clustering method (SCM) that could self-organize the cluster number and volume according to the structure of the data.

Another important class of kernels is the Cauchy kernels defined as

\[
C^p(x) = [c((x - x_j)^2)]^p = [(1 + [(x - x_j)^2]/\beta)^{-1}]^p
\]

that are based on the Cauchy density function \(f(x) = (1/\pi)\)

\[
(1 + x^2)^{-1}, \quad -\infty < x < \infty
\]

with their shadows \(SC^p\) defined as

\[
SC^p(x) = C^{p-1}(x), \quad p > 1.
\]

The mean shift process with \(x \leftarrow m_{C^p(x_j)}\) is used to find the modes of the density estimate \(f_{SC^p}(x)\). There is less application with the Cauchy kernels. To improve the weakness of FCM in a noisy environment, Krishnapuram and Keller [36] first considered relaxing the constraint of the fuzzy \(c\)-partitions’ summation to 1 and then proposing the so-called possibilistic \(c\)-means (PCM) clustering algorithm. Eventually, these possibilistic membership functions become the Cauchy kernels. This is the only application of the Cauchy kernels to clustering what we can find.

The simplest kernel is the flat kernel defined as

\[
F(x) = \begin{cases} 
1 & \text{if } \|x - x_j\|^2 \leq 1, \\
0 & \text{if } \|x - x_j\|^2 > 1,
\end{cases}
\]

with the Epanechnikov kernel \(E(x)\) as its shadows

\[
E(x) = \begin{cases} 
1 - \|x - x_j\|^2 & \text{if } \|x - x_j\|^2 \leq 1, \\
0 & \text{if } \|x - x_j\|^2 > 1.
\end{cases}
\]

Moreover, the Epanechnikov kernel has the biweight kernel \(B(x)\) as its shadow

\[
B(x) = \begin{cases} 
(1 - \|x - x_j\|^2)^2 & \text{if } \|x - x_j\|^2 \leq 1, \\
0 & \text{if } \|x - x_j\|^2 > 1.
\end{cases}
\]

For a more general presentation, we extend the Epanechnikov kernel \(E(x)\) to be the generalized Epanechnikov kernels \(K_E^p(x)\) with the parameter \(p\) defined as

\[
K_E^p(x) = [k_E(\|x - x_j\|^2)]^p
\]

\[
= \begin{cases} 
(1 - \|x - x_j\|^2/\beta)^p & \text{if } \|x - x_j\|^2 \leq \beta, \\
0 & \text{if } \|x - x_j\|^2 > \beta.
\end{cases}
\]

Thus, the generalized Epanechnikov kernels \(K_E^p(x)\) has the corresponding shadows \(SK_E^p(x)\) defined as

\[
SK_E^p(x) = K_E^{p+1}(x), \quad p > 0.
\]

Note that we have \(K_E^0(x) = F(x), K_E^1(x) = E(x)\) and \(K_E^2(x) = B(x)\) when \(\beta = 1\). The mean shift process with \(x \leftarrow m_{K_E^p(x_j)}\)

![Fig. 1. The kernel functions with different \(p\) values: (a) Gaussian kernels, (b) Cauchy kernels and (c) generalized Epanechnikov kernels.](image-url)
is used to find the modes of the density estimate $\hat{f}_{SK_E}(x)$. In total, we present three kernel classes with their shadows. These are Gaussian kernels $G^p(x)$ with their shadows $SG^p(x) = G^p(x)$, Cauchy kernels $C^p(x)$ with their shadows $SC^p(x) = C^{p-1}(x)$ and the generalized Epanechnikov kernels $K^p_E(x)$ with their shadows $SK^p_E(x) = K^{p+1}_E(x)$. The behaviors of these kernels with different $p$ are shown in Fig. 1. The mean shift procedures using these three classes of kernels can easily find their corresponding density estimates. The parameters $\beta$ and $\rho$, called the normalization and stabilization parameters, respectively, greatly influence the performance of the mean shift procedures for the kernel density estimate. We will discuss these in the next section.

3. Mean shift as a robust clustering

A suitable clustering method should have the ability to tolerate noise and detect outliers in the data set. Many criteria such as the breakdown point, local-shift sensitivity, gross error sensitivity and influence functions [25] can be used to measure the level of robustness. Comaniciu and Meer [11] discussed some relationships of the mean shift and the nonparametric $M$-estimator. Here, we provide a more detailed analysis of the robustness of the mean shift procedures.

3.1. Analysis of robustness

The mean shift mode estimate $m_k(x)$ can be related to the location $M$-estimator

$$\hat{\theta} = \arg \min \theta \sum_{j=1}^{n} \rho(x_j - \theta),$$

where $\rho$ is an arbitrary loss measure function. $\hat{\theta}$ can be generated by solving the equation

$$\frac{\partial \rho(x_j - \theta)}{\partial \theta} \sum_{j=1}^{n} \rho(x_j - \theta) = \sum_{j=1}^{n} \rho(x_j - \theta) = 0,$$

where $\rho(x_j - \theta) = (\partial / \partial \theta) \rho(x_j - \theta)$. If the kernel $H$ is a reasonable similarity measure which takes values on the interval $[0, 1]$, then $1 - H$ could be a reasonable loss function. Thus, a location $M$-estimator can be found by

$$\hat{\theta} = \arg \min \theta \sum_{j=1}^{n} \rho(x_j - \theta) = \arg \min \theta \sum_{j=1}^{n} \left[ 1 - h(||\theta - x_j||^2) \right]$$

$$= \arg \max \theta \sum_{j=1}^{n} h(||\theta - x_j||^2) = \arg \max \hat{f}_H(\theta).$$

This means that $m_k(x)$ is a location $M$-estimator if $K$ is a kernel with its shadow $H$.

The influence curve (IC) can help us to assess the relative influence of an individual observation toward the value of an estimate. In the location problem, we have the influence of an $M$-estimate with

$$IC(y; F, \theta) = \frac{\phi(y - \theta)}{\int \phi(y - \theta) dF(y)},$$

where $F_Y(y)$ denotes the distribution function of $Y$. The $M$-estimator has shown that $IC(y; F, \theta)$ is proportional to its $\phi$ function [25]. If the influence function of an estimator is unbounded, an outlier might cause trouble where the $\phi$ function is used to denote the degree of influence.

Suppose that $\{x_1, \ldots, x_n\}$ is a data set in the real number space $R^k$ and $SG^p(x)$ is a shadow of $G^p(x)$. Then, $m_{G^p}(x)$ is a location $M$-estimator with $\phi$ function defined to be

$$\phi_{G^p}(x_j - x) = \frac{d}{dx} [1 - SG^p(x)] = \frac{2p}{\beta} (x - x_j)G^p(x).$$

By applying the L’Hospital’s rule, we derive $\lim_{x \rightarrow \pm \infty} \phi_{G^p}(x_j - x) = 0$. Thus, we have the influence curve of $m_{G^p}(x)$ with $IC(x_j; F, x) = 0$ when $x_j$ tends to positive or negative infinity. This means that the influence curve is bounded and the influence of an extremely large or small $x_j$ on the mode estimator $m_{G^p}(x)$ is very small. We can also find the location of $x_j$ which has a maximum influence on $m_{G^p}(x)$ by solving $(\partial / \partial \theta) \phi_{G^p}(x_j - x) = 0$. Note that both $m_{C^p}(x)$ and $m_{K^p_E}(x)$ also have the previous properties where their $\phi$ functions are as follows:

$$\phi_{C^p}(x_j - x) = \frac{2(p - 1)}{\beta} (x - x_j)C^p(x),$$

$$\phi_{K^p_E}(x_j - x) = \begin{cases} \frac{2(p + 1)}{\beta} (x - x_j)K^p_E(x) & \text{if } (x - x_j)^2 \leq \beta, \\ 0 & \text{if } (x - x_j)^2 \geq \beta. \end{cases}$$

The $\phi$ function for the mean shift-based mode-seeking estimators with $\beta = 1$ is shown in Fig. 2. The influence of an individual $x_j$ on $m_{K^p_E}(x)$ is zero if $(x - x_j)^2 \geq \beta$ and an extremely large or small $x_j$ will have no effect on $m_{K^p_E}(x)$. However, the influence of an extremely large or small $x_j$ on $m_{G^p}(x)$ and $m_{C^p}(x)$ is a monotone decreasing function of the stabilization parameter $p$. An outlier has no influence when $p$ becomes large. In this paper, we will choose these generalized Epanechnikov kernels for our mean shift-based clustering algorithm because they can suitably fit the data sets even with extremely large or small data points.

3.2. Bandwidth selection and stabilization parameter

Bandwidth selection greatly influences the performance of kernel density estimation. Comaniciu and Meer [11] summarized four different techniques according to statistical analysis-based and task-oriented points of view. Here, we give new consideration to first normalizing the distance measure by dividing the normalization parameter $\beta$ and then focusing on estimating the stabilization parameter $p$. The normalization parameter $\beta$ is set to be the sample variance

$$\beta = \frac{\sum_{j=1}^{n} ||x_j - \bar{x}||^2}{n} \quad \text{where } \bar{x} = \frac{\sum_{j=1}^{n} x_j}{n}. $$
The phi function of Gaussian kernels

The phi function of Cauchy kernels

The phi function of generalized Epanechnikov kernels

Fig. 2. The $\phi$ functions with different $p$ values: (a) Gaussian kernels, (b) Cauchy kernels and (c) generalized Epanechnikov kernels.

We have the properties of

$$\lim_{p \to 0} m_{Gp}(x) = \frac{1}{n} \sum_{j=1}^{n} G(x) x_j = \bar{x},$$

$$\lim_{p \to 0} m_{Cp}(x) = \lim_{p \to 0} m_{Kp}(x) = \bar{x},$$

This means that when $p$ tends to zero, the kernel density estimate has only one mode with the sample mean. Fig. 3 shows the histogram of a three-cluster normal mixture data with its corresponding density estimates $\hat{f}_{SGp}(x)$, $\hat{f}_{SCp}(x)$ and $\hat{f}_{SKp}(x)$. In fact, a small $p$ will lead to only one mode $\bar{x}$ in the estimated density, as shown in Fig. 3 for the case of $p = 1$. However, a too large $p$ will cause the density estimate to have too many modes as shown in Fig. 3 for the case of $p = 50$. This can be explained as follows. We denote $\hat{G}(x) = \max_j G(x)$ and $G(x)' = G(x)/\hat{G}(x)$. We then have

$$\lim_{p \to \infty} m_{Gp}(x) = \lim_{p \to \infty} \frac{1}{n} \sum_{j=1}^{n} G(x) x_j = \frac{1}{n} \sum_{j=1}^{n} G(x) x_j$$

$$= \lim_{p \to \infty} \frac{1}{n} \sum_{j=1}^{n} (G(x)')^p x_j = \frac{1}{n} \sum_{j=1}^{n} (G(x)') x_j.$$

This means that, as $p$ tends to infinity, the data point which is the closest to the initial value will become the peak. Hence, we have

$$\lim_{p \to \infty} m_{Gp}(x) = \lim_{p \to \infty} m_{Cp}(x) = \lim_{p \to \infty} m_{Kp}(x).$$

In this case, each data point will become a mode in the blurring and nonblurring mean shift procedures.

The stabilization parameter $p$ can control the performance of the density estimate. This situation is somehow similar to the bandwidth selection, but with different merits. The purpose of the bandwidth selection is to find a suitable bandwidth
describe the proposed method as follows. In our experiments, we use the mean shift algorithm provided by Yager and Filev [9] which is used to find the initial seeds for suitable density estimates. As shown in Fig. 3, different \( p \) corresponds to different shapes of the density estimates. A suitable \( p \) will correspond to a satisfactory kernel density estimate so that the modes found by the mean shift can present the dense area of the data set.

### 3.3 A stabilization parameter estimation method

A practical bandwidth selection technique is related to the decomposition stability of the density shape estimates. The bandwidth is taken as the center of the largest operating range over which the same number of clusters are obtained for the given data [23]. This means that the shapes of the estimated density are unchanged over this operating range. Although this technique can yield a suitable bandwidth estimate, it needs to find all cluster centers (modes) for each bandwidth over the chosen operating range. It thus requires a large computation. The increased shift for the stabilization parameter \( p \) is \( M=1 \) and \( M=2 \). We now explain how to use Eq. (8) to find a suitable stabilization parameter \( p \). Suppose that the density shape estimates are unchanged with \( p = 10 \) and \( p = 11 \), the correlation value of \( \{ \hat{f}_{SK_E}^p(x_1), \ldots, \hat{f}_{SK_E}^p(x_n) \} \) between \( p = 10 \) and \( p = 11 \) will be very close to 1. In this situation, \( p = 10 \) will be a suitable parameter estimate. In this way, we can skip the step for finding the modes of the density estimate. In our experiments, a good operating range for \( p \) always falls between 1 and 50. This is because we normalize the kernel function by dividing the normalization parameter \( \beta \).

#### Example 1

We use the previously described graphical method of correlation comparisons to find the suitable \( p \) for the data set shown in Fig. 3. We have the correlation values of \( \{ \hat{f}_{SK_E}^{10}(x_1), \ldots, \hat{f}_{SK_E}^{10}(x_n) \} \) and \( \{ \hat{f}_{SK_E}^{11}(x_1), \ldots, \hat{f}_{SK_E}^{11}(x_n) \} \) as shown in Fig. 4. Figs. 4(a)–(d) give the results of the cases with the increased shifts \( M = 1, 2, 3 \) and 4, respectively. In Fig. 4(a), the \( y \)-coordinate of the first solid circle point denotes the correlation value between \( p = 1 \) and 2. The \( y \)-coordinate of the second solid circle point denotes the correlation value between \( p = 2 \) and 3. The 1st, 2nd, 3rd solid circle points, etc., denote the correlation values of the respective pairs \( (p = 1, p = 2), (p = 2, p = 3) \) and...
(p = 3, p = 4), etc. In Fig. 4(b), the y-coordinate of the first solid circle point denotes the correlation value between p = 1 and p = 1 + 2 = 3. The y-coordinate of the second solid circle point denotes the correlation value between p = 3 and p = 5. The 1st, 2nd, 3rd solid circle points, etc., denote the correlation values of the respective pairs (p = 1, p = 3), (p = 3, p = 5) and (p = 5, p = 7), etc. The others can be similarly induced. Since the density shape will be unchanged when the correlation value is close to 1, we may choose p with the solid circle point close to the dotted line of value 1.

In Fig. 4(a), with the increased shift M = 1, the 10th solid circle point is very close to the dotted line (i.e. the correlation value between p = 10 and p = 11 is very close to 1). Hence, p = 10 is a suitable estimate. We will explain why we do not choose the point that lies on the dotted line later. In Fig. 4(b), with the increased shift M = 2, the sixth solid circle point is close to the dotted line (i.e. the correlation value between p = 11 and 13 is close to 1). Hence, p = 11 is a suitable estimate. In Fig. 4(c), with the increased shift M = 3, the correlation value between p = 13 and 16 is very close to 1. Hence, p = 13 is a suitable estimate. Similarly, p = 13 is a suitable estimate as shown in Fig. 4(d). Fig. 5 illustrates the density estimates using \( \hat{f}\) with p = 10–13. We find that these density estimates actually match the histogram of the data. The following is another example with two-dimensional data.

Example 2. We use a two-dimensional data set in this example. Fig. 6(a) shows a 16-cluster data set where the data points in each group are uniformly generated from rectangles. Figs. 6(b)–(d) are the correlation values with the increased shift M = 1–3, respectively. In Fig. 6(b), the 13th point is close to the dotted line and hence p = 13 is a good estimate. In Fig. 6(c), the 7th point is close to the dotted line and hence we choose p = 13. Fig. 6(d) also indicates that p = 13 is a suitable estimate. In this example, our graphical method of correlation comparisons with different increased shift M presents all the same results. The density estimates using \( \hat{f}\) with p = 1, 5, 10 and 15 are shown in Figs. 7(a), (b), (c) and (d), respectively.

In Section 3.2, Eqs. (4) and (5) show that a small p value will cause the kernel density estimate to have only one mode with the sample mean. Figs. 3 and 7(a) also verify this point. The selected p = 13 is between p = 10 and 15 whose density shapes are shown in Figs. 7(c) and (d) that match the original data structure well.

Our graphical method of correlation comparisons can find a good density estimate. This estimation method can accomplish the tasks where the bandwidth selection method can do it. However, our method skips the step of finding all modes of the density estimate so that it is much simpler and less computational. We estimate a suitable stabilization parameter \( \sigma \) where its operating range is always located between 1 and 50. Note that, if the increased shift \( M \) is large such as \( M = 4 \) or 5, then it may miss a good estimate for \( \sigma \). However, a too small increased shift \( M \), such as \( M < 1 \), may take too much computational time. We suggest that taking \( M = 1 \) for the graphical method of correlation comparisons may perform well in most simulations.

We now explain why we did not choose the point that lies on the dotted line according to the following two reasons. Note that the stabilization parameter \( \sigma \) is similar to the number of the bars drawn on the data histogram. The density shape with a large \( \sigma \) corresponds to the histogram with a large number of bars and hence has too many modes. Eqs. (6) and (7) also show this property. The stabilization parameter \( \sigma \) is the power of the kernel function which takes values between 0 and 1. Each data point will have the value of Eq. (8) being close to 1 with a large \( \sigma \) case and hence the correlation value becomes large. Figs. 4 and 6 also show this tendency on the curvilinear tail. According to these two reasons, we suggest to choose the estimate of \( \sigma \) with the point being very close to the dotted line.

3.4. A mean shift-based clustering algorithm

Since the generalized Epanechnikov kernel \( K_E \) has the robustness property as analyzed in Section 3.1, we use the \( K_E \) kernel in the mean shift procedure. By combining the graphical method of correlation comparisons for the estimation of the stabilization parameter \( \sigma \), we propose a mean shift-based clustering method (MSCM) with the \( K_E \) kernel, called MSCM(\( K_E \)). The proposed MSCM(\( K_E \)) procedure is therefore constructed with four steps: (1) select the kernel \( K_E \); (2) estimate the stabilization \( \sigma \); (3) use the mean shift procedures; (4) identify the clusters. The MSCM(\( K_E \)) procedure is with its diagram as shown in Fig. 8 and summarized as follows.

The MSCM(\( K_E \)) procedure:

- **Step 1.** Choose the \( K_E \) kernel.
- **Step 2.** Use the graphical method of correlation comparisons for estimating \( \sigma \).
- **Step 3.** Use the nonblurring mean shift procedure.
- **Step 4.** Identify the clusters.

We first implement the MSCM(\( K_E \)) procedure for the data set shown in Fig. 3. According to Fig. 4(a), p = 10 is a suitable estimate for the data set in Fig. 3(a). The results of the MSCM(\( K_E \)) procedure are shown in Fig. 9. The curve
represents the density estimate using $\hat{f}_{SK_E}^p$ with $p = 10$. The locations of the data points are illustrated by the histograms. Figs. 9(a)–(f) show these data histograms after implementing the MSCM($K_E^p$) procedure with the iterative times $T = 1, 5, 10, 20, 25$ and $77$. The MSCM($K_E^p$) procedure is convergent when $T = 77$. The algorithm is terminated when
Select a kernel
1. Choose the kernel $K^p_E$ (used in this paper)
2. Choose the kernel $G^p$ or $C^p$

Estimate the stabilization parameter $p$
1. Use the proposed graphical method of correlation comparisons (used in this paper)
2. Directly assign a value

Use the mean shift procedure
1. Nonblurring mean shift (used in this paper)
2. General mean shift
3. Blurring mean shift

Identify clusters
1. Merge data
2. Other methods (discuss in Section 4)

Fig. 8. The diagram of the proposed MSCM($K^p_E$) procedure.

Fig. 9. The density estimates using the MSCM($K^p_E$) procedure with $p = 10$ and the histograms of the data points where (a) $T = 1$, (b) $T = 5$, (c) $T = 10$, (d) $T = 20$, (e) $T = 25$ and (f) $T = 77$.

the locations of all data points are unchanged when the iterative time achieves the maximum $T = 100$. Fig. 9(f) shows that all data points are centralized to three locations, which are the modes of the density estimate $\hat{f}_{SK_E}$. The MSCM($K^p_E$) procedure finds that these three clusters do indeed match the data structure.

We also implement the MSCM($K^p_E$) procedure on the data set in Fig. 6(a). According to the graphical method of correlation comparisons as shown in Fig. 6(b), $p = 13$ is a suitable estimate. The MSCM($K^p_E$) results after the iterative times $T = 1$, 2 and 5 are shown in Figs. 10(a), (b) and (c), respectively. The data histograms show that all data points are centralized to 16 locations which match the data structure when the iterative time $T = 5$. We mention that no initialization problems occurred with our method because the stabilization parameter $p$ is estimated by the graphical method of correlation comparisons. We can also solve the cluster validity problem by merging the data points which are similar to the many progressive clustering methods [30,37,38].

We have shown that the mean shift can be a robust clustering method by definition of the $M$-estimator and also discussed the $\phi$ function that denotes the degree of influence of an individual observation for the kernels $G^p$, $C^p$ and $K^p_E$. In the comparisons of computational time, our graphical method of correlation comparisons for finding the stabilization parameter $p$ is much faster than with the bandwidth selection method. Finding all cluster centers for the bandwidth selection method requires the computational complexity $O(n^2st_1)$ for the first selected bandwidth where $s$ and $t_1$ denote the data dimension and the algorithm iteration number, respectively. Suppose that
we choose the maximum 50 bandwidths of the operating range for the graphical method of correlation comparisons, then this would require the computational complexity $O(n^2 \sum_{i=1}^{50} t_i)$ for finding a suitable bandwidth. In the MSCM($K_p^E$) procedure, the stabilization parameter $p$ is always in Refs. [1,50] because we normalize the kernel function. Thus, it requires $O(50n^2s)$ to find a suitable $p$ when the increased shift $M = 1$. Then, the MSCM($K_p^E$) procedure should be faster if we set the increased shift $M$ to be greater than 1.

We finally further discuss the MSCM($K_p^E$) procedure. The first step is to choose a kernel function where we recommend the selection of $K_p^E$. You may also choose the kernels $G^P$ and $C^P$. The second step is to estimate the stabilization parameter $p$ using the proposed graphical method of correlation comparisons. Of course, the user can also directly assign a value for $p$. In our experiments, a suitable $p$ always fell between 10 and 30. The third step is to implement the mean shift procedure. Three kinds of mean shift procedures mentioned in Section 2 can be used. We suggest the nonblurring mean shift. The fourth step is to classify all data points. If the data set only contains the round shape clusters, the nonblurring mean shift can label the data points by merging them. However, if the data set contains other cluster shapes such as line or circle structures, then the data points may not be centralized to some small locations. In this case, merging the data points will not work well. The next step is to discuss this problem and also provide some numerical examples and comparisons. We then apply these to image processing.

4. Examples and applications

Some numerical examples, comparisons and applications are stated in this section. We also consider the computational complexity, cluster validity and improvements of the mean shift in large continuous, discrete data sets with its application to the image segmentation.

4.1. Numerical examples with comparisons

Example 3. In Fig. 11(a), a large cluster number data set with 32 clusters is presented where we also add some uniformly noisy points. The graphical plot of the correlation comparisons for the data set is shown in Fig. 11(b) where $p=20$ is a suitable estimate. The MSCM($K_p^E$) results when $T = 1, 3, 5$ and 11 are shown in Figs. 11(c), (d), (e) and (f), respectively. We can see that all data points are centralized to 32 locations that suitably match the structure of data. The results are not affected by the noisy points. The cluster number can easily be found by merging the data points which are centralized to the same locations. Thus, the MSCM($K_p^E$) procedure can be a simple and useful unsupervised clustering method. The superiority of our proposed method to other clustering methods is discussed below.

We also use the well-known FCM clustering algorithm [39] to cluster the same data set shown in Fig. 11(a). Suppose that we do not know the cluster number of the data set. We adopt the validity indexes, such as the partition coefficient (PC) [40], Fukuyama and Sugeno (FS) [41], and Xie and Beni (XB) [42], to solve the validity problem when using FCM. Therefore, we need to process the FCM algorithm for each cluster number $c = 2, 3, \ldots$. However, the first problem is to assign the locations of the initial values in FCM. We simulate two cases of assignments with the random initial values and the designed initial values. Figs. 12(a)–(c) present the PC, FS and XB validity indexes when the random initial values are assigned. FCM cannot detect those 32 separated clusters very effectively, but it can when the designed initial values are assigned as shown in Fig. 13. Note that the PC index has a large value tendency when the cluster number is small. Thus, a local optimal value of the PC index curve may present a good cluster number estimate [43]. In the case of the designed initials assignment, all validity measures should have an optimal solution with $c = 32$. However, this result is obtained only when the FCM algorithm can divide the data into 32 well-separated clusters. If the initializations are not properly chosen (for example, no centers are initialized in 1 of these 32 rectangles), we cannot ensure good partitions being found by the FCM algorithm. In fact, most clustering algorithms always have this initialization problem. This is a more difficult problem in a high-dimensional case. Here we find that our proposed MSCM($K_p^E$) procedure has the necessary robust property for the initialization.

We had mentioned that the computational complexity of finding the stabilization parameter estimate $p$ in MSCM($K_p^E$) is $O(50n^2s)$. After estimating $p$, the computational complexity of implementing the MSCM($K_p^E$) procedure is $O(n^2st_K)$ where $t_K$ denotes the number of iterations. In FCM, the computational complexity is $O(ncst_c)$, where $t_c$ is the number of iterations and $c$ is the cluster number. For solving the validity problem, we
need to process FCM from \( c = 2 \) to \( c = C_{\text{max}} \) where \( C_{\text{max}} \) is the maximum number of clusters. Thus, the computational complexity of FCM is \( O(n(C_{\text{max}} - 1)s \sum_{c=1}^{C_{\text{max}}} t_c) \). Although the complexity of the MSCM\((K_p^E)\) procedure is high when the data set is large, it can directly solve the validity problem in a simple way. In order to reduce the complexity of the
MSCM\((K_p E)\) procedure for the large data set, we will propose
a technique to deal with it later.

4.2. Cluster validity and identified cluster for the MSCM\((K_p E)\)
procedure

Note that, after we have found the MSCM\((K_p E)\) clustering re-
sults, we had simply identified clusters and the cluster number
by merging the data points which are centralized to the same
location. However, if the data set contains different shape clus-
ters such as a line or a circle structure, the data points may not
be centralized to the same small region. On the other hand, the
estimated density shape may have flat curves on the modes. In
this case, the method of identifying clusters by merging data
points may not be an effective means of finding the clusters.
If we use the Agglomerative Hierarchical Clustering (AHC)
algorithm with some linkage methods such as single linkage,
complete linkage and Ward’s method, etc., we can identify dif-
ferent shape clusters from the MSCM\((K_p E)\) clustering results.
We demonstrate this identified method with AHC as follows.

Example 4. Fig. 14(a) shows a three-cluster data set with
different cluster shapes. The graphical plot of the correlation
comparisons is shown in Fig. 14(b) where \(p = 28\) is a suit-
able stabilization parameter estimate. The locations of all data
points after processing the MSCM\((K_p E)\) are shown in Fig. 14(c).
In this situation, the identified method of merging data points
may cause difficulties. Since the mean shift procedure can be
seen as the method that shifts the similar data points to the
same location, the AHC with single linkage can help us to find
clusters so that similar data points can be easily linked into
the same cluster. We process the AHC with single linkage for
the MSCM\((K_p E)\) clustering results shown in Fig. 14(c). We ob-
tain the hierarchical clustering tree as shown in Fig. 15(a). The
hierarchical tree shows that the data set contains three well-
separated clusters. These clusters are shown in Fig. 15(b) with
different symbols. In fact, most clustering algorithms are likely
to fail when they are applied to the data set shown in Fig. 14(a).

By combing MSCM\((K_p E)\) with AHC, these methods can
detect different shape clusters for most data sets. On the
other hand, it may also decrease the convergence time for
the MSCM\((K_p E)\) procedure. Note that, the estimated density
shape may have flat curves on the modes for the data set so
that the MSCM\((K_p E)\) procedure is too time consuming in this
case, especially for a large data set. Since the AHC method
can connect the close data points into the same cluster, we
can then stop the MSCM\((K_p E)\) procedure when the data points
shift to the location near the mode. Therefore, we can set a
larger stopping threshold for the MSCM\((K_p E)\) procedure or set
a smaller maximum iteration.

4.3. Implementation for large continuous data sets

We know that one important property of the MSCM\((K_p E)\)
procedure is to take all data points as the initial values so
that these data points can be centralized to the locations
of the modes. The cluster validity problems can be solved

Fig. 14. (a) The data set. (b) The graph of correlation comparisons where \(p = 28\) is a suitable estimate. (c) The data locations after using the MSCM\((K_p E)\) procedure.

Fig. 15. (a) The hierarchical tree of the data set in Fig. 14(c). (b) Identified three clusters with different symbols.
simultaneously in this way. However, this process wastes too much time for a large data set. One way to reduce the number of data points for the MSCM\((K_p^p)\) procedure is to first merge similar data points from the original data set. Although this technique can reduce the number of data points, the MSCM\((K_p^p)\) clustering results will differ as to the original nonblurring mean shift procedure. Since the mode represents the location of where most data are located, these data points will shift to the mode at a few mean shift iterations. However, these data points have to continue to process the mean shift procedure until all data points are centralized to these modes. This is because the mean shift procedure is treated as a batch type.

Although all data points are taken to be the initial values, there are no constrains on these initial values when we process the mean shift clustering algorithm using Eq. (3). This means that we can treat the mean shift as a sequential clustering algorithm. The input of the second data point is performed only when the first data point shifts to the mode. The input of the third data point is performed only when the second data point shifts to the mode. The rest points can be similarly induced. In this sequential-type mean shift procedure, only the data points that are far away from modes require more iteration. However, most data points (nearby the modes) require less iteration. The computational complexity of this sequential-type mean shift procedure is \(O(n \sum t_i)\) where \(t_i\) denotes the iterative count of data point \(x_i\). However, the iteration of each data point will be all equal to the largest iteration number in the batch-type mean shift procedure where the computational complexity is \(O(n^2st_K)\) with \(t_K = \max(t_1, \ldots, t_n)\).

Note that this technique is only feasible for the nonblurring mean shift procedure, such as the proposed MSCM\((K_p^p)\) procedure. Since the blurring mean shift algorithm update the data points at each iterative, the location of each data point will be changed so that the sequential type is not feasible for the blurring mean shift procedure.

### 4.4. Implementation for large discrete data sets

Suppose that the data set \(\{x_1, \ldots, x_n\}\) only takes values on the set \(\{y_1, \ldots, y_m\}\) with corresponding counts \(n_1, \ldots, n_m\). That is, there are \(n_1\) observations of \(\{x_1, \ldots, x_n\}\) with values \(y_1\), \(n_2\) observations of \(\{x_1, \ldots, x_n\}\) with values \(y_2\), etc. For example, a 128 \(\times\) 128 gray level image will have 16 384 data points (or pixels). However, it only takes values on the gray level set \(\{0, 1, \ldots, 255\}\). The mean shift procedure has large computational complexity in this \(n=16\ 384\) case. The following theorem is a technique that can greatly reduce computational complexity.

**Theorem 1.** Suppose that \(x \in \{x_1, \ldots, x_n\}\), \(y \in \{y_1, \ldots, y_m\}\) and \(x = y\). The mean shift of \(y\) is defined by

\[
y = m_K(y) = \frac{\sum_{i=1}^m k(\|y - y_i\|^2)w(y_i)n_i}{\sum_{i=1}^m k(\|y - y_i\|^2)w(y_i)n_i}.
\]

Then we have \(m_K(x) = m_K(y)\). That is, the mean shift of \(x\) using Eq. (3) can be replaced by the mean shift of \(y\) using Eq. (9) with the equivalent results.

**Proof.** Since \(\{x_1, \ldots, x_n\}\) only take values on the set \(\{y_1, \ldots, y_m\}\) with corresponding counts \(n_1, \ldots, n_m\), we have \(\sum_{i=1}^m k(\|y - y_i\|^2)w(y_i)n_i = \sum_{j=1}^n k(\|x - x_j\|^2)w(x_j)\) and \(\sum_{i=1}^m k(\|y - y_i\|^2)w(y_i)n_iy_i = \sum_{j=1}^n k(\|x - x_j\|^2)w(x_j)x_j\). Thus, \(m_K(x) = m_K(y)\). \(\square\)

We can take \(\{y_1, \ldots, y_m\}\) to be initial values and then process the mean shift procedure using Eq. (9) for the MSCM\((K_p^p)\) procedure. The final locations of \(\{y_1, \ldots, y_m\}\) using Eq. (9) will be equivalent to the final locations of \(\{x_1, \ldots, x_n\}\) using Eq. (3).

In this discrete data case, the computational complexity for the MSCM\((K_p^p)\) procedure becomes \(O(m^2st)\) where \(t\) denotes the iterative counts and \(m < n\). The previously discussed sequential type of continuous data case can also be used to greatly reduce computational complexity in this discrete data case. The following is a simple application in the image segmentation.

### 4.5. Application in image segmentation

**Example 5.** Fig. 16(a) is the well-known Lenna image with the size 128 \(\times\) 128. This image contains 16 384 pixel values with a maximum value of 237 and minimum value of 33. This means that it only takes values on \(\{33, 34, \ldots, 237\}\) and the count frequency of these pixel values is shown in Fig. 17(a). The computational complexity of the original nonblurring mean shift procedure is \(O(n^2st) = O(16\ 384^2st)\). After applying the described technique using Eq. (9), the computational complexity of the nonblurring mean shift procedure in this discrete case becomes \(O(m^2st) = O(205^2st)\) which actually greatly reduces computational complexity. The graph of correlation comparisons for the MSCM\((K_p^p)\) procedure is shown in Fig. 17(b) where \(p=15\) is a suitable estimate. The estimated density shape with \(p=15\) is shown in Fig. 17(c) which quite closely matches
the pixel counts with four modes. The MSCM($K^p_E$) clustering results when $t = 5, 15$ and $98$ are shown in Figs. 16(b), (c) and (d), respectively. Following convergence, all pixel values are centralized to four locations presented by the arrows in the x-coordinate as shown in Fig. 17(d). Note that, for most c-means based clustering algorithms, $c = 8$ is mostly used as the cluster number of the Lenna data set. If we combine a validity index to search for a good cluster number estimate for this data, the computational complexity should become very large. In the MSCM($K^p_E$) procedure, a suitable cluster number estimate is found automatically and the computational complexity for this data set can be reduced by using the above-mentioned technique.

For color image data, each pixel contains a three-dimensional data point that each dimension takes values on the set $\{0, 1, \ldots, 255\}$ and hence we will have $255 \times 255 \times 255$ possible pixel values. However, for a 128 * 128 color image, the worst situation is to have 128 * 128 different pixel values. In general, an image data will contain many overlapping pixel values and hence the technique can also significantly reduce computational complexity even for color images.

4.6. Comparisons to other kernel-based clustering algorithms

Comaniciu [10] proposed the variable-bandwidth mean shift with data-driven bandwidth selection. To demonstrate their bandwidth selection method, Comaniciu [10] used the data set drawn with equal probability from normal distributions $N(8, 2)$, $N(25, 4)$, $N(50, 8)$ and $N(100, 16)$ with total $n = 400$. Fig. 18(a) shows the histogram of this data set. Comaniciu [10] used 12 analysis bandwidths in the range of
Fig. 19. (a) Data set. (b) The graph of correlation comparisons. (c) The hierarchical tree of the data after implementing the MSCM($K_p^E$) algorithm. (d) Four identified clusters presented in different symbols.

Fig. 20. (a) The graph of correlation comparisons and (b) the hierarchical trees after implementing the MSCM($K_p^E$) algorithm for the Iris data.

Fig. 21. (a) The graph of correlation comparisons and (b) the hierarchical trees after implementing the MSCM($K_p^E$) algorithm for the Wine data.
1.5–17.46 where four classes of bandwidth were detected (see Ref. [10, pp. 284-285]). Based on our proposed MSCM($K_p^E$), the graph of correlation comparisons for the data set of Fig. 18(a) gives $p = 8$ and there are four clusters presented as shown in Figs. 18(b) and (c), respectively.

Furthermore, we use the data set of nonlinear structures with multiple scales in Comaniciu [10] for our next comparisons as shown in Fig. 19(a). The graph of correlation comparisons, the hierarchical tree of the data after implementing the MSCM($K_p^E$) algorithm and four identified clusters presented in different symbols are shown in Figs. 19(b)–(d), respectively. These results with four identified clusters from MSCM($K_p^E$) are coincident to those of the variable bandwidth mean shift in Comaniciu [10]. Note that the variable-bandwidth mean shift in Comaniciu [10] is a good technique to find a suitable bandwidth for each data point, but it requires implementing the mean shift process only one time.

The kernel $k$-means proposed by Girolami [4] is another kernel-based clustering algorithm. The cluster numbers of the data sets Iris ($c = 3$), Wine ($c = 3$) and Crabs ($c = 4$) are correctly detected by kernel $k$-means (see Ref. [4, pp. 783–784]). We implement the proposed method for the data sets Iris, Wine and Crabs where the results of these real data sets from MSCM($K_p^E$) are shown in Figs. 20, 21, and 22, respectively. The MSCM($K_p^E$) algorithm with $p = 10$ indicates that $c = 2$ is suitable for the Iris data. However, there are some clues to see $c = 3$ is also suitable for the Iris data according to the results of Fig. 20(b). Note that this is because one of the clusters

---

Fig. 22. (a) The graph of correlation comparisons and (b) the hierarchical trees after implementing the MSCM($K_p^E$) algorithm for the Crabs data.

Fig. 23. (a) Estimated $p$ for Cmc data. (b), (c) and (d) are the single linkage, complete linkage and Ward’s method for the Cmc data after implementing the MSCM($K_p^E$) algorithm.
of Iris is separable from the other two overlapping clusters. The results of Wine data from the MSCM($\mathcal{K}_E^c$) with a linkage method, such as single linkage, may not give a clear cluster number estimate. This situation appears in some real data applications. We recommend that we use more linkage methods such as complete linkage, Ward’s method, etc., to offer more clustering information. In our final example, we use the Cmc data set from the UCI machine learning repository [44] in the comparisons of the proposed MSCM($\mathcal{K}_E^c$) with kernel $k$-means. There are 10 attributes, 1473 observations and three clusters in the Cmc data set. The MSCM($\mathcal{K}_E^c$) results for the Cmc data set are shown in Fig. 23 where the single linkage, complete linkage and Ward’s method indicate that the data set contains three clusters that are coincident to the data structure. Although kernel $k$-means using the eigenvalue decomposition of the kernel matrix proposed by Girolami [4] can estimate the correct cluster number for the Iris, Wine and Crabs data sets, the estimates of the cluster number quite depend on the selected radial basis function (RBF) kernel width. The eigenvalue decomposition results with different RBF kernel widths for the Cmc data set based on the kernel matrix are shown in Fig. 24. The correct result $c = 3$ shown in Fig. 24(a) depends on the suitable chosen RBF kernel width 15. However, other choices with 10 and 5 as shown in Figs. 24(b) and (c) cannot detect the cluster number of $c = 3$.

Finally, we mention that merging data points using AHC after the mean shift process MSCM($\mathcal{K}_E^c$) not only provide the estimated cluster number, but also can give more information about the data structure. For example, the observation 19 of Wine data as shown in Fig. 21(b) and the observation 141 of Crabs data as shown in Fig. 22(b) can be seen as the abnormal observations. Fig. 18(c) also shows that the observation 314 is far away from other data points. In unsupervised clustering, our object is not only to detect the correct cluster number for an unknown cluster number data set, but also to discover a reasonable cluster structure for the data set.

5. Conclusions

In this paper, we proposed a mean shift-based clustering procedure, called MSCM($\mathcal{K}_E^c$). The proposed MSCM($\mathcal{K}_E^c$) can be robust with three facets. In facet 1, since we combined the mountain function to estimate the stabilization parameter, the bandwidth selection problem for the density estimation can be solved in a graphical way. Also the operating range of the stabilization parameter is always fixed for various data sets. This led to the proposed method being robust for the initializations.

In facet 2, we discussed the problem of the mean shift procedure faced in the problems of cluster validity and identified clusters. According to the properties of the nonblurring mean shift procedure, we suggested combining the AHC with the single linkage to identify different shaped clusters. This technique can also save the computational complexity of the MSCM($\mathcal{K}_E^c$) procedure by setting a larger stopping threshold or setting a smaller maximum iterative count. Thus, the proposed method can be robust for different cluster shapes in the data set.

In facet 3, we analyzed the robust properties of the mean shift procedure according to the nonparametric $M$-estimator and the $\phi$ functions of three kernel classes. We then focused on the generalized Epanechnikov kernel where extremely large or small data points will have no influence on the mean shift procedures. Our demonstrations showed that the proposed method is not influenced by noise and hence is robust as to the noise and outliers. We provided some numerical examples, comparisons and applications to illustrate the superiority of the proposed MSCM($\mathcal{K}_E^c$) procedure including the computational complexity, cluster validity, improvements of the mean shift in large continuous, discrete data sets and image segmentation.

Acknowledgments

The authors are grateful to the anonymous referees for their critical and constructive comments and suggestions to improve
the presentation of the paper. This work was supported in part by the National Science Council of Taiwan, under Kuo-Lung Wu’s Grant: NSC-95-2118-M-168-001 and Miin-Shen Yang’s Grant: NSC-95-2118-M-033-001-MY2.

References


About the Author—KUO-LUNG WU received the BS degree in mathematics in 1997, the MS and PhD degrees in applied mathematics in 2000 and 2003, all from the Chung Yuan Christian University, Chungli, Taiwan. Since 2003, he has been an Assistant Professor in the Department of Information Management at Kun Shan University of Technology, Tainan, Taiwan. He is a member of the Phi Tau Phi Scholastic Honor Society of Taiwan. His research interests include fuzzy theorem, cluster analysis, pattern recognition and neural networks.

About the Author—MIIN-SHEN YANG received the BS degree in mathematics from the Chung Yuan Christian University, Chungli, Taiwan, in 1977, the MS degree in applied mathematics from the National Chiao-Tung University, Hsinchu, Taiwan, in 1980, and the PhD degree in statistics from the University of South Carolina, Columbia, USA, in 1989. In 1989, he was an Associate Professor of the Department of Applied Mathematics at the Chung Yuan Christian University. Since 1994, he has been a Professor at the same university, where, from 2001 to 2005, he was the Chairman of the Department of Applied Mathematics. During 1997–1998, he was a Visiting Professor with the Department of Industrial Engineering, University of Washington, Seattle, USA. His current research interests include applications of statistics, fuzzy clustering, neural fuzzy systems, pattern recognition and machine learning. Dr. Yang is an Associate Editor of the IEEE Transactions on Fuzzy Systems.