IET Image Processing

Research Article

IET Journals
The Institution of Engineering and Technology

Modified mean shift algorithm

ISSN 1751-9659
Received on 4th September 2017
Revised 31st May 2018
Accepted on 6th August 2018
E-First on 18th September 2018
doi: 10.1049/iet-ipr.2018.5600
www.ietdl.org

Youness Aliyari Ghassabeh[1,2] ✉, *Frank Rudzicz*[1,2]

[1]*Toronto Rehabilitation Institute – UHN, 550 University Avenue, Toronto, Ontario, Canada*
[2]*Department of Computer Science, University of Toronto, Toronto, Ontario, Canada*
✉ *E-mail: aliyari@cs.toronto.edu*

Abstract: The mean shift (MS) algorithm is an iterative method introduced for locating modes of a probability density function. Although the MS algorithm has been widely used in many applications, the convergence of the algorithm has not yet been proven. In this study, the authors modify the MS algorithm in order to guarantee its convergence. The authors prove that the generated sequence using the proposed modified algorithm is a convergent sequence and the density estimate values along the generated sequence are monotonically increasing and convergent. In contrast to the MS algorithm, the proposed modified version does not require setting a stopping criterion a priori; instead, it guarantees the convergence after a finite number of iterations. The proposed modified version defines an upper bound for the number of iterations which is missing in the MS algorithm. The authors also present the matrix form of the proposed algorithm and show that, in contrast to the MS algorithm, the weight matrix is required to be computed once in the first iteration. The performance of the proposed modified version is compared with the MS algorithm and it was shown through the simulations that the proposed version can be used successfully to estimate cluster centres.

1 Introduction

It has been shown that the modes of a probability density function (pdf) can be used in many machine learning applications, including clustering [1], image and video segmentation [2, 3], signal denoising [4], and object tracking [5, 6]. The mean shift (MS) algorithm is an iterative, non-parametric technique that was introduced by Fukunaga and Hostetler [7] to estimate modes of a pdf. The MS algorithm was generalised by Cheng [8] and became popular in the machine learning community when its potential for feature space analysis was studied [9]. The MS algorithm starts from one of the data points, as the initial mode, and assigns a weighted average of the data set in each iteration. It can be shown that the MS vector always points toward the direction of the maximum increase in the density function. In fact, the MS algorithm is an instance of the gradient ascent algorithm with an adaptive step size [10]. The iterations continue until the norm of the difference between two consecutive mode estimates becomes less than some predefined threshold. The number of the estimated modes in this procedure is taken as the number of clusters and the estimated modes represent the cluster centres. Furthermore, all data points associated with the same mode (called the 'basin of attraction' of that mode [9]) are considered members of the same cluster.

The MS algorithm has been successfully used in many applications, but a rigorous proof for its convergence is still missing in the literature. Comaniciu and Meer [9] claimed that the sequence generated by the MS algorithm is a convergent sequence, but a crucial step in the convergence proof was not correct. Specifically, based on an incorrect use of the triangle inequality, the authors in [9] showed that the sequence generated by the MS algorithm is a Cauchy sequence (and as a result a convergent sequence), which is not true in general. Later, Carreria-Perpinán [11] showed that the MS algorithm with the Gaussian kernel is an instance of the expectation maximisation (EM) algorithm and hence the generated sequence is a convergent sequence. However, without additional conditions, the sequence of parameter estimates generated by the EM algorithm may not converge (see [12, 13]). Positively, the authors in [14, 15] showed that if an estimated pdf has a finite number of modes (or equivalently isolated modes), then the sequence of the estimated cluster centres generated by the MS algorithm is a convergent sequence. Unfortunately, the authors in [14, 15] could not provide sufficient conditions for widely used kernels (e.g. Gaussian) to have a finite number of (or isolated) modes. In a recent study, the authors investigated the connection of the MS algorithm to the kernel regression technique and suggested that exploiting the theoretical properties of the asymptotic bias of the kernel regression technique might be helpful to show the convergence of the MS algorithm [16].

Surprisingly, the authors in [17] made the same mistake as those in [9] by applying the triangle inequality to the squared Euclidean distance in order to prove the convergence of the flow line of a function f of class C^3 (Theorem 1 in [17]). Later, the authors in [17] fixed the mistake by adding an additional assumption that the end point of the flow line is an isolated mode [18].

The convergence of the MS algorithm in a one-dimensional space with an analytic kernel (e.g. Gaussian) was shown in [19]. Later, Aliyari [20] proved that a sequence generated by the MS algorithm with a certain class of kernels is a monotonic and convergent sequence in a one-dimensional space. The special one-dimensional case has limited use in practice, and the authors in [19, 20] could not generalise the convergence result to dimensions greater than one. Recently, Aliyari [21] presented a sufficient condition for the MS algorithm with the Gaussian kernel to have isolated stationary points. The sufficient condition is given as a lower bound for the bandwidth of the kernel function. Unfortunately, this condition is not practically useful. The bandwidth, as a function of the sample size, must converge to zero as the sample size goes to infinity to guarantee the asymptotic consistency of the pdf estimate [22]. Although choosing the bandwidth based on the condition in [21] guarantees isolated stationary points, it generates poor estimation of the pdf that results in an inaccurate mode estimate.

In this paper, we present a modified version of the MS algorithm to guarantee the convergence of the generated sequence. In particular, we add one step to the regular MS algorithm and assign the computed MS vector to the closest data point in the data set in each iteration. We prove that the generated sequence using the proposed modified version is a convergent sequence. We also show that, similar to the original MS algorithm, the pdf estimate

along the generated sequence is a monotonically increasing and convergent sequence. Although the additional step increases the computation cost, there is no need to update the weight matrix in each iteration. For the proposed modified version, in contrast to the original MS algorithm, the weight matrix is computed once in the first iteration. Since the convergence is guaranteed in the proposed version, there is no need to set a predefined threshold as the stopping criterion for the algorithm. It is clear that setting a threshold in general cannot be used as a reliable measure of closeness to the convergence point (if there is a convergence point). It is possible that the difference between two consecutive mode estimates becomes less than the predefined threshold and the algorithm terminates the iterations, but both points are far from the possible convergence point. Furthermore, the number of iterations is highly dependent on the stopping criterion in the original MS algorithm, i.e. a small threshold increases the number of iterations, and in general may not be an upper bound for the number of iterations. But for the proposed modified MS algorithm, we show that the number of iterations is always bounded above by the number of samples. In the next section, we briefly review the MS algorithm. The proposed modified MS algorithm is presented in Section 3. Simulation results to show the effectiveness of the proposed algorithm for clustering are given in Section 4. Section 5 is devoted to the concluding remarks.

2 MS algorithm

Let $\boldsymbol{x}_i \in \mathbb{R}^D, i = 1, \ldots, n$ be a sequence of n independent and identically distributed random variables generated from an unknown density function f. A D-variate kernel $K: \mathbb{R} \to \mathbb{R}$ is a non-negative, real-valued, and even function that integrates to one [23]. The kernel density estimate $\hat{f}$ at an arbitrary point $\boldsymbol{x}$, using a kernel $K(\boldsymbol{x})$ is given by [23]

$$\hat{f}_h(\boldsymbol{x}) = \frac{1}{nh^D} \sum_{i=1}^{n} K\left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{h}\right), \tag{1}$$

where h is called the bandwidth. A special class of kernels, called radially symmetric kernels, has been widely used for pdf estimation. Radially symmetric kernels are defined by $K(\boldsymbol{x}) = c_{k,D} k(\| \boldsymbol{x} \|^2)$, where $c_{k,D}$ is a normalisation factor that causes $K(\boldsymbol{x})$ to integrate to one and $k: [0, \infty) \to [0, \infty)$ is called the *profile* of the kernel. The profile of a kernel is assumed to be a non-increasing, non-negative, and piecewise continuous function that satisfies $\int_0^\infty k(x)\,\mathrm{d}x < \infty$. Using the profile k and the bandwidth h, the kernel density $\hat{f}(\boldsymbol{x})$ in (1) changes to the following form [23]:

$$\hat{f}_{h,k}(\boldsymbol{x}) = \frac{c_{k,D}}{nh^D} \sum_{i=1}^{n} k\left(\left\| \frac{\boldsymbol{x} - \boldsymbol{x}_i}{h} \right\|^2\right). \tag{2}$$

Assuming that the profile k is differentiable with derivative k', by taking the gradient of (2), we obtain

$$\begin{aligned} \nabla \hat{f}_{h,k}(\boldsymbol{x}) = {} & \frac{2c_{k,D}}{nh^{D+2}} \left[\sum_{i=1}^{n} g\left(\left\| \frac{\boldsymbol{x} - \boldsymbol{x}_i}{h} \right\|^2\right)\right] \\ & \times \left[\frac{\sum_{i=1}^{n} \boldsymbol{x}_i g\left(\| (\boldsymbol{x} - \boldsymbol{x}_i)/h \|^2\right)}{\sum_{i=1}^{n} g\left(\| (\boldsymbol{x} - \boldsymbol{x}_i)/h \|^2\right)} - \boldsymbol{x}\right], \end{aligned} \tag{3}$$

where $g(x) = -k'(x)$. The second term in the above representation is called the MS vector, $\boldsymbol{m}_{h,g}(\boldsymbol{x})$, and (3) can be rewritten in the following compact form:

$$\nabla \hat{f}_{h,k}(\boldsymbol{x}) = \frac{2c_{k,D}}{h^2 c_{g,D}} \hat{f}_{h,g}(\boldsymbol{x}) \boldsymbol{m}_{h,g}(\boldsymbol{x}), \tag{4}$$

where $\hat{f}_{h,g}(\boldsymbol{x})$ is the kernel density estimate at $\boldsymbol{x}$ using the kernel function $G(\boldsymbol{x}) = c_{g,D} g(\| \boldsymbol{x} \|^2)$. The modes of the estimated pdf are points $\boldsymbol{x}$ such that the gradient function is zero at those points, i.e. $\nabla \hat{f}(\boldsymbol{x}) = 0$. From (3), it can be observed that the modes of the estimated pdf are the fixed points of the following function

$$\boldsymbol{m}_{h,g}(\boldsymbol{x}) + \boldsymbol{x} = \frac{\sum_{i=1}^{n} \boldsymbol{x}_i g\left(\| (\boldsymbol{x} - \boldsymbol{x}_i)/h \|^2\right)}{\sum_{i=1}^{n} g\left(\| (\boldsymbol{x} - \boldsymbol{x}_i)/h \|^2\right)}. \tag{5}$$

To solve (5), the MS algorithm initialises the cluster centre estimate sequence to be one of the observed data points and the cluster centre estimate $\boldsymbol{y}_j$ in the jth iteration is updated by

$$\boldsymbol{y}_{j+1} = \boldsymbol{y}_j + \boldsymbol{m}(\boldsymbol{y}_j) = \frac{\sum_{i=1}^{n} \boldsymbol{x}_i g\left(\| (\boldsymbol{y}_j - \boldsymbol{x}_i)/h \|^2\right)}{\sum_{i=1}^{n} g\left(\| (\boldsymbol{y}_j - \boldsymbol{x}_i)/h \|^2\right)}. \tag{6}$$

The MS algorithm repeatedly updates the cluster centre estimate $\boldsymbol{y}_j$ using (6) until the norm of the difference between two consecutive estimates becomes less than some predefined threshold, i.e. $\| \boldsymbol{y}_{j+1} - \boldsymbol{y}_j \| < \epsilon$ for some $j \in \mathbb{N}$. The set of all points converging to the same cluster centre define the basin of attraction for that cluster. Points in a same basin of attraction belong to the same cluster. Choosing the appropriate bandwidth h plays a crucial role in the MS algorithm. On the one hand, a small bandwidth may slowdown moving of the MS sequence towards a mode, but on the other hand a large bandwidth may lead merging two modes. The problem of selecting the bandwidth h for the MS algorithm is discussed in detail in [9, 24]. According to [24], 'The best of the currently available data-driven methods for bandwidth selection seems to be the plug-in rule, which was proven to be superior to least squares cross validation and biased cross-validation'.

2.1 Gaussian blurring mean shift

Gaussian blurring mean shift (GBMS) is a variation of the MS algorithm that uses the Gaussian kernel and updates all data in each iteration. Let $\boldsymbol{x}_i^0 \in \mathbb{R}^D, i = 1, \ldots, n$ denote the initial data set. The data set at the kth iteration, $\boldsymbol{x}_i^k, i = 1, 2 \ldots, n$, is updated by [25]

$$\boldsymbol{x}_i^k = \frac{\sum_{j=1}^{n} \boldsymbol{x}_j^{k-1} g\left(\| (\boldsymbol{x}_i^{k-1} - \boldsymbol{x}_j^{k-1})/h \|^2\right)}{\sum_{j=1}^{n} g\left(\| (\boldsymbol{x}_i^{k-1} - \boldsymbol{x}_j^{k-1})/h \|^2\right)}, \quad i = 1, 2, \ldots, n. \tag{7}$$

Let $\boldsymbol{X}^0 = \{\boldsymbol{x}_1^0, \ldots, \boldsymbol{x}_n^0\}$ denote the initial $D \times n$ data matrix. Then by applying the GBMS algorithm, each point of the initial data set moves to a new point and we obtain a sequence of data matrices $\boldsymbol{X}^1, \boldsymbol{X}^2, \ldots$, where each data matrix is a blurred version of the previous version. Cheng [8] showed that for any initial data set and the bandwidth, the GBMS algorithm generates a sequence of data matrices that converges to a data matrix $\boldsymbol{X}$ with all points coincident. Carreira-Perpiñán [25] showed that if the GBMS algorithm stops before clusters start to move toward each other, then it can be used as a clustering tool. The typical behaviour of the GBMS algorithm has two phases. First, points merge into compact clusters which take a few iterations. In the second phase, which may take several hundred iterations, those clusters move towards each other and they finally merge into a single point [25]. It is desirable to stop the GBMS algorithm right after the first phase in which points merge into clusters. Carreira-Perpiñán [25] proposed the following stopping criterion to terminate the iterative process before data points start merging

$$\frac{1}{n} \sum_{i=1}^{n} e_i^k < \epsilon, \quad e_i^k = \| \boldsymbol{x}_i^k - \boldsymbol{x}_i^{k-1} \|, \tag{8}$$

where ϵ is a small tolerance that needs to be set *a priori*.

3 Modified MS algorithm

Comaniciu and Meer claimed that the sequence $\{\boldsymbol{y}_j\}_{j=1,2,\ldots}$ is a Cauchy sequence [9], which is not true in general. Specifically, the error in the convergence proof of the MS algorithm in [9] is due to

Input : Bandwidth h, profile function $g(x)$, and data set $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}, n \geq 2$.

Output: Estimated cluster centers $\boldsymbol{y}_1^*, \ldots, \boldsymbol{y}_k^*$, where k is the number of clusters .

begin
  for *(i=1 to n)* **do**
    Initialization: $j \leftarrow 1$; $\boldsymbol{y}_j \leftarrow \boldsymbol{x}_i$;
    /* The initial cluster center estimate $\boldsymbol{y}_1$ is chosen to be one of the observed data points. */
    repeat
      /* Evaluate the mean shift vector using (5) */
      $\boldsymbol{m}(\boldsymbol{y}_j) \leftarrow \frac{\sum_{i=1}^{n} \boldsymbol{x}_i g\left(\| \frac{\boldsymbol{y}_j - \boldsymbol{x}_i}{h} \|^2\right)}{\sum_{i=1}^{n} g\left(\| \frac{\boldsymbol{y}_j - \boldsymbol{x}_i}{h} \|^2\right)} - \boldsymbol{y}_j$;
      /* Do the cluster center update using (6) */
      $\tilde{\boldsymbol{y}}_{j+1} = \boldsymbol{m}(\boldsymbol{y}_j) + \boldsymbol{y}_j$;
      /* Find the closest data point to $\tilde{\boldsymbol{y}}_{j+1}$ */
      $\boldsymbol{y}_{j+1} = \arg\min_{\boldsymbol{x} \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}} \| \tilde{\boldsymbol{y}}_{j+1} - \boldsymbol{x} \|$;
    until *Until convergence occurs, i.e.,* $\boldsymbol{y}_{j+1} = \boldsymbol{y}_j$ *for some* $j \in \mathbb{N}$;
    $\boldsymbol{y}_i^* \leftarrow \boldsymbol{y}_j$;
  end
  Merge the estimated cluster centers that are closer than h ;
  Optional: Eliminate clusters that attract small number of data points;
end

Fig. 1 *Algorithm 1: Modified MS algorithm for clustering*

the incorrect use of the triangle inequality [21]. Through further manipulation of the proof in [9], one can show

$$\lim_{j \to \infty} \| \boldsymbol{y}_{j+1} - \boldsymbol{y}_j \| = 0. \tag{9}$$

According to the definition of the MS vectors in (6), the generated sequence $\{\boldsymbol{y}_j\}$ is always inside the convex hull of the data set. Therefore, $\{\boldsymbol{y}_j\}$ is a bounded sequence satisfying (9). Note that the last two properties are not enough to imply the convergence of $\{\boldsymbol{y}_j\}_{j=1,2,\ldots}$. As a counterexample, consider a sequence $\{\boldsymbol{z}_j\}_{j=1,2,\ldots} \in \mathbb{R}^2$ as follows:

$$\boldsymbol{z}_j = \left(\sin\left(2\pi \sum_{i=1}^{j} \frac{1}{i}\right), \cos\left(2\pi \sum_{i=1}^{j} \frac{1}{i}\right)\right), \quad i = 1, 2, \ldots. \tag{10}$$

The sequence $\{\boldsymbol{z}_j\}$ is a bounded sequence that satisfies the following inequality:

$$\| \boldsymbol{z}_{j+1} - \boldsymbol{z}_j \| \leq \frac{1}{j+1}. \tag{11}$$

The right side of (11) is the geodesic distance along the unit circle between two consecutive members of the sequence, and the left side is the length of the chord connecting two members. It can be observed that the right side (11) goes to zero as $j \to \infty$, i.e. it satisfies the limit property in (9), but $\{\boldsymbol{z}_j\}$ is not a convergent sequence and rotates on the unit circle.

As mentioned, the authors in [14, 15] showed that if the set of the stationary points of an estimated pdf are finite (or equivalently isolated), then the sequence generated by the MS algorithm converges. Unfortunately, a general and useful condition, that leads to a set of finite (or isolated) stationary points of the estimated pdf for commonly used kernels (such as the Gaussian kernel), still seems to be missing (although the author in [11] makes the plausible claim, without proof, that the set of stationary points is always finite for the Gaussian kernel).

We slightly modify the MS algorithm to guarantee the convergence of the generated sequence. The proposed modified version starts from one of the data points, computes the MS vector using (5), and updates the cluster centre estimate by assigning the computed MS vector to the closest data point. In other words, the updated cluster centre estimate is one of the data points and is computed by

$$\tilde{\boldsymbol{y}}_{j+1} = \boldsymbol{y}_j + \boldsymbol{m}(\boldsymbol{y}_j) \tag{12}$$

$$\boldsymbol{y}_{j+1} = \underset{\boldsymbol{x} \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}}{\operatorname{argmin}} \| \tilde{\boldsymbol{y}}_{j+1} - \boldsymbol{x} \|, \tag{13}$$

where $\boldsymbol{m}(\boldsymbol{y}_j)$ is the MS vector in (5), and $\{\boldsymbol{x}_1 \ldots, \boldsymbol{x}_n\}$ is the set of input data. The proposed modified MS algorithm is summarised in Algorithm 1 (see Fig. 1). Note that although we use *for* in Algorithm 1 (Fig. 1), the algorithm can run in parallel on the data set, each time initialised to one of the n data points. The MS algorithm may get stuck in low-density regions, i.e. regions with few number of data points. In this situation, if the number of input points that converge to the same mode (i.e. basin of attraction of that mode) are less than a predefined threshold M, the cluster will be eliminated. Otherwise, it will be a considered as a cluster centred at that mode. The hidden parameter M exists in the original MS algorithm too and needs to be set *a priori* depending on a specific application.

Similar to the MS algorithm, the sequence $\{\hat{f}(\boldsymbol{y})_j\}_{j=1,2,\ldots}$ generated by the modified MS algorithm is an increasing and convergent sequence. Furthermore, the cluster centre estimate sequence, $\{\boldsymbol{y}_j\}_{j=1,2,\ldots}$, is also a convergent sequence, and we have:

Theorem 1: Assume a kernel pdf estimate $\hat{f}$ with bandwidth h, and a radially symmetric kernel K having profile k which is positive, strictly decreasing, convex, and continuously differentiable on $\mathbb{R}$ (as is defined in (2)). Then the following holds:

(i) The density estimate values along the sequence of output values of the modified MS algorithm is a monotonically increasing and convergent sequence, i.e. $\{\hat{f}_{h,k}(\boldsymbol{y}_j)\}_{j=1,2,\ldots}$ is monotonically increasing and convergent.

(ii) The L_2-norm of the gradient estimate along the sequence of the output values of the modified MS algorithm is bounded above by

$$\| \nabla \hat{f}_{h,k}(\boldsymbol{y}_j) \| \leq 2c_{k,D} g(0) d_{\max} / h^{D+2}, \quad j = 1, 2, \ldots,$$

where $d_{\max} = \max_{\boldsymbol{x}_i, \boldsymbol{x}_j \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}} \| \boldsymbol{x}_i - \boldsymbol{x}_j \|$.

(iii) The cluster centre estimate sequence, $\{\boldsymbol{y}_j\}_{j=1,2,\ldots,}$ generated by the modified MS algorithm is also a convergent sequence.

The upper bound for the $L2$-norm of the gradient vector in part (ii) of Theorem 1 is much simpler and computationally less demanding than the upper bound given in Theorem 2 in [26]. For the special case of the Gaussian kernel, we have $K(\boldsymbol{x}) = (2\pi)^{-D/2} \exp(-\| \boldsymbol{x} \|^2 / 2)$. As a result, the upper bound for the $L2$-norm of the gradient vector simplifies to

$$\| \nabla \hat{f}_{h,k}(\boldsymbol{y}_j) \| \leq \frac{2}{(2\pi)^{D/2} h^{D+2}} d_{\max}. \tag{14}$$

Note that, similar to the MS algorithm after running the proposed modified version, we have to merge the estimated cluster centres that are closer than the bandwidth h. We may also need to eliminate the estimated cluster centres that attract too few points. Theorem 1 guarantees the convergence of the generated sequence $\{\boldsymbol{y}_j\}$ and, in contrast to the original MS algorithm, there is no need to set a threshold ϵ as the stopping criterion. Furthermore, there is no known explicit upper bound for the number of iterations for the original MS algorithm and it is highly dependent on the stopping criterion, but the proof of Theorem 1 shows that the proposed modified version converges in at most $n - 1$ iterations, where n is the number of data points.

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]$ denote a $D \times n$ data matrix, $\boldsymbol{W} = [w_{ij}]$ denote a $n \times n$ weight matrix whose ijth element is $w_{ij} = g(\| (\boldsymbol{x}_j - \boldsymbol{x}_i)/h \|^2)$, and $\boldsymbol{D}$ denote a diagonal matrix whose jjth element is $\sum_{i=1}^{n} g(\| (\boldsymbol{x}_j - \boldsymbol{x}_i)/h \|^2)$. For the original MS

Input : Bandwidth h, profile function $g(x)$, and data matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n], n \geq 2$

Output: Estimated cluster centers $\boldsymbol{y}_1^*, \ldots, \boldsymbol{y}_k^*$, where k is the number of clusters

begin

Compute the weight matrix $\boldsymbol{W}$, where ijth element is $w_{ij} = g(\|(\boldsymbol{x}_j - \boldsymbol{x}_i)/h\|^2)$;

Compute the diagonal weight matrix $\boldsymbol{D}$, where jjth element is $\sum_{i=1}^{n} g(\|(\boldsymbol{x}_j - \boldsymbol{x}_i)/h\|^2)$;

Initialization: $\tilde{\boldsymbol{W}} = \boldsymbol{W}$ and $\tilde{\boldsymbol{D}} = \boldsymbol{D}$;

repeat

$\tilde{\boldsymbol{Y}} = \boldsymbol{X}\tilde{\boldsymbol{W}}\tilde{\boldsymbol{D}}^{-1}$;

for *($i = 1$ to n)* **do**

$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x} \text{ is a column of } \boldsymbol{X}} \|\tilde{\boldsymbol{Y}}_i - \boldsymbol{x}\|$;

/* $\tilde{\boldsymbol{Y}}_i$ denote the ith column of matrix $\tilde{\boldsymbol{Y}}$ */

;

/* Let p denote the column number that minimizes the above, i.e., $\|\tilde{\boldsymbol{Y}}_i - \boldsymbol{x}_p\| \leq \|\tilde{\boldsymbol{Y}}_i - \boldsymbol{x}_j\|, j = 1, 2, \ldots, n$ */

;

$\tilde{\boldsymbol{Y}}_i = \boldsymbol{x}^*$;

$\tilde{\boldsymbol{W}}_i = \boldsymbol{W}_p$;

$\tilde{\boldsymbol{D}}_i = \boldsymbol{D}_p$;

end

until *convergence occurs (there is no change in columns of $\tilde{\boldsymbol{Y}}$)*;

/* Columns of $\tilde{\boldsymbol{Y}}$ represent the cluster centers */

;

Merge the estimated cluster centers that are closer than h;

Optional: Eliminate clusters that attract small number of data points;

end

Fig. 2 *Algorithm 2: Modified MS algorithm, matrix form*

algorithm, the first iteration can be written in the following matrix form

$$\boldsymbol{Y}_1 = \boldsymbol{X}\boldsymbol{W}\boldsymbol{D}^{-1}, \tag{15}$$

where $\boldsymbol{Y}_1$ is a $D \times n$ matrix whose ith column represents the mode estimate starting $\boldsymbol{x}_i$ in the first iteration. In the original MS algorithm, columns of $\boldsymbol{Y}_1$ are used to update the weight matrix, and in general we have

$$\boldsymbol{Y}_{k+1} = \boldsymbol{X}\boldsymbol{W}_k\boldsymbol{D}_k^{-1}, \quad k = 2, 3, \ldots, \tag{16}$$

where ijth element of $\boldsymbol{W}_k$, w_{ij}^k, is given by $g(\| (\boldsymbol{y}_j^k - \boldsymbol{x}_i)/h \|^2)$, $\boldsymbol{D}_k$ is a $n \times n$ diagonal matrix whose jjth element is given by $\sum_{i=1}^{n} g(\| (\boldsymbol{y}_j^k - \boldsymbol{x}_i)/h \|^2)$, and $\boldsymbol{y}_j^k$ is the jth column of $\boldsymbol{Y}_k$. Therefore, the weight matrices $\boldsymbol{W}_k$ and $\boldsymbol{D}_k$ are required to be updated for each iteration in the original MS algorithm, that increases the computational cost. For the GBMS algorithm, the matrix form of the computations has the following simple form [25]:

$$\boldsymbol{X}_{\text{new}} = \boldsymbol{X}_{\text{old}}\boldsymbol{W}\boldsymbol{D}^{-1}, \tag{17}$$

where $\boldsymbol{W}$ and $\boldsymbol{D}$ are defined as before, and $\boldsymbol{X}_{\text{old}}$ and $\boldsymbol{X}_{\text{new}}$ represent the old and updated data matrices. Since the GBMS algorithm updates the whole data set in each iteration, it does not require updating the weight matrices in each iteration, so they can be computed once, but as mentioned before, the main problem with the GBMS algorithm is the possibility of merging all data points into a single data point. Setting the appropriate stopping threshold in (8) to terminate the iterations before merging the clusters is a challenging task and highly dependent on the specific data set.

Although searching for the closest data point to $\tilde{\boldsymbol{y}}_j, j = 1, 2, \ldots, n$ in each iteration increases the overall computational cost, the proposed modified MS algorithm does not require computing the weight matrix in each iteration. The weight matrices $\tilde{\boldsymbol{W}}$ and $\tilde{\boldsymbol{D}}$ are initialised to $\boldsymbol{W}$ and $\boldsymbol{D}$, respectively. Then, in each iteration based on the nearest neighbour data point their columns are updated by replacing by an appropriate column from $\boldsymbol{W}$ and $\boldsymbol{D}$, i.e. the column associated to the nearest data point. The only computational cost in subsequent iterations is searching for the nearest neighbours to $\tilde{\boldsymbol{y}}_j$ in the data set. The matrix form of the proposed modified MS algorithm is given in Algorithm 2 (see Fig. 2). It can be observed from Algorithm 2 (Fig. 2) that the weight matrices $\boldsymbol{W}$ and $\boldsymbol{D}$ are computed just once in the initialisation step using the data samples and their columns are used to update $\tilde{\boldsymbol{W}}$ and $\tilde{\boldsymbol{D}}$. Each iteration consists of two steps: (i) the matrix product $\tilde{\boldsymbol{Y}} = \boldsymbol{X}\tilde{\boldsymbol{W}}\tilde{\boldsymbol{D}}^{-1}$, where columns of $\tilde{\boldsymbol{Y}}$ are in fact $\tilde{\boldsymbol{y}}$ in (13). Note that the computational cost of matrix multiplication can be asymptotically accelerated to $N^{2.376}$ [27]. (ii) the search for the nearest neighbour, where columns of $\tilde{\boldsymbol{Y}}$ are compared with the data set and each column is replaced with its nearest neighbour in the data set, and the columns of $\tilde{\boldsymbol{W}}$ and $\tilde{\boldsymbol{D}}$ are updated using $\boldsymbol{W}$ and $\boldsymbol{D}$.

The medoid shift algorithm is also another modification of the MS algorithm that constrains the generated mode estimates to pass through the input data points [28]. The medoid shift algorithm runs the MS algorithm once for each data points and then updates the mode estimate sequence in each iteration through solving a constrained optimisation problem [28]. The proposed modified version in this paper runs the MS algorithm in each iteration and updates the mode estimate sequence by finding the nearest data point to the updated MS vector. The authors in [29] showed that the medoid shift algorithm may fail to properly identify the modes of an unknown density function. By providing an example, the authors in [29] showed that the medoid shift may not correctly identify all the modes of a density function and even applying the algorithm on the modes iteratively does not lead to satisfactory results. The authors in [29] presented quick shift algorithm as a new mode-seeking procedure to address the shortcomings of the medoid shift algorithm. The authors in [29] showed that the quick shift technique is simpler and faster than the MS algorithm [29]. However, unlike the MS algorithm (and the proposed modified version in this paper), the quick shift requires an additional parameter η to be set *a priori*. The quick shift algorithm connects all the points into a single tree and then the modes of a density function are recovered by breaking the branches of the tree that are longer than the threshold η. The optimal amount of parameter η can be find through running the algorithm for all the possible values of η, which can be an exhaustive task. The quick shift algorithm does not involve any calculation of the MS vector (given in (6) and (7)), and in fact it is related to a classic technique introduced in [30]. The main difference between the quick shift and the technique in [30] is that maximising the gradient approximation must be done in a neighbourhood of each point defined *a priori* by the parameter η [29]. Recently, Jiang investigated some statistical properties of the quick shift algorithm and showed the statistical consistency guarantee of quick shift on mode and cluster recovery under mild distributional assumptions [31]. Statistical properties of the MS algorithm, including consistency, is discussed in [16] through connecting the MS algorithm to kernel regression technique. The results in [16] can be adopted and extended to the proposed modified version, but this is out of scope of the paper and subject of our future work.

4 Simulation results

In this section, we demonstrate the effectiveness of the proposed modified MS algorithm for clustering using the $S1$ [32] and $R15$ [33] data sets. The $S1$ data set contains 5000 two-dimensional vectors artificially generated with varying complexity in terms of spatial data distributions with 15 predefined clusters [32]. For both the MS algorithm and the proposed modified MS algorithm, we used the Gaussian kernel with the bandwidth $h = 50,000$. The stopping criterion for the MS algorithm is set to 0.01, i.e. $\epsilon = 0.01$ and only estimated cluster centres which attract more than five data points are considered. Fig. 3 shows the distribution of samples from data set $S1$ and estimated cluster centres using the MS algorithm and the proposed modified version. The data points are

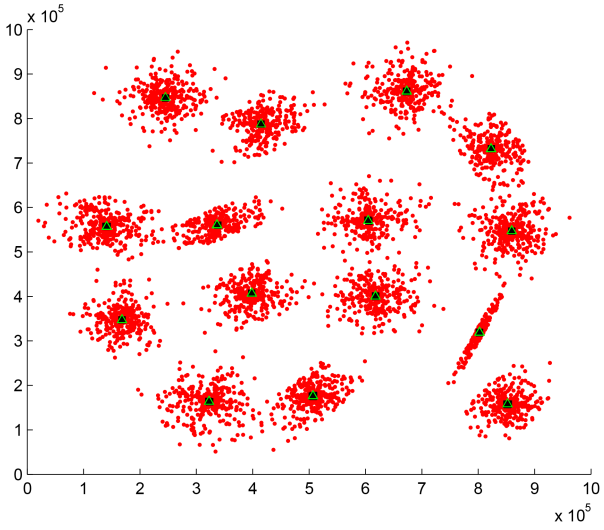

Fig. 3 *Data points from the data set S1 are shown using filled red circles. The estimated cluster centres using the MS algorithm are shown using empty green triangles, and the estimated cluster centres using the proposed modified MS algorithm are shown using the filled black squares*

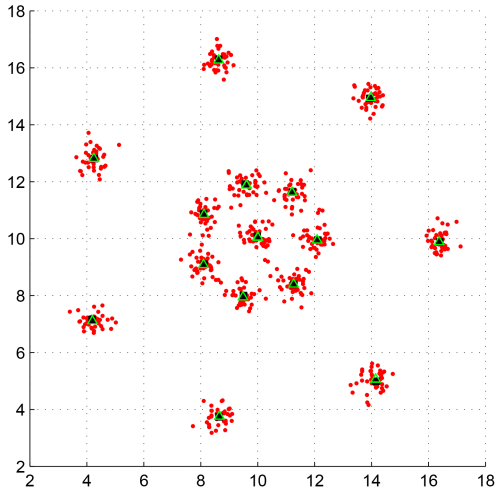

Fig. 4 *Data points from the data set R15 are shown using filled red circles. The estimated cluster centres using the MS algorithm are shown using empty green triangles, and the estimated cluster centres using the proposed modified MS algorithm are shown using the filled black squares*

shown with red circles, the cluster centre estimates using the MS algorithm are shown using green triangles, and the estimated cluster centres using the proposed modified MS algorithm are shown with the black squares. It can be observed from Fig. 3 that the proposed modified MS algorithm successfully estimates the cluster centres and the estimated cluster centres almost coincide with the estimated cluster centres using the MS algorithm.

The data set *R*15 consists of 600 samples of 15 similar two-dimensional Gaussian clusters that are positioned in rings [33]. We set the bandwidth h to be 0.4 for both the MS algorithm and the modified version. The stopping threshold for the MS algorithm is $\epsilon = 0.01$ and we only consider the cluster centre estimates that attract more than 10 samples. Fig. 4 shows the distribution of the data points and the estimated cluster centres resulting from the MS algorithm and the proposed modified version. The estimated cluster centres resulting from the MS algorithm are shown with green triangles and the estimated cluster centres using the proposed modified MS algorithm are shown with black squares. It can be observed from Fig. 4 that the proposed modified version successfully finds that the cluster centres and the estimated centres are close to the output of the MS algorithm.

5 Conclusion

The MS algorithm has been widely used in many applications, but the convergence of the generated mode estimate sequence has not been proved and one needs to set a threshold as the stopping criterion to terminate the iterative process. In this paper, we modify the MS algorithm and prove that the generated sequence using the proposed modified MS algorithm is a convergent sequence. Therefore, there is no need to set a stopping criterion for the proposed modified version and the generated sequence converges after a finite number of iterations (when the number of samples n is finite). We also show that the estimated pdf along the generated sequence is a monotonically increasing and convergent sequence. Furthermore, the proposed modified MS algorithm provides an upper bound (i.e. $n-1$) for the required number of iterations for each data point. Such an upper bound does not exist for the original MS algorithm (or the GBMS algorithm), and the number of iterations is dependent on the stopping criterion (a small stopping criterion increases the number of iterations and vice versa). We also present the matrix form of the proposed modified MS algorithm and show that, in contrast to the MS algorithm, the weight matrices only need to be computed once. Finally, through simulations, we show that the proposed modified MS algorithm can be successfully used to estimate the cluster centres. This paper focuses mainly on the theoretical properties of the proposed modified MS algorithm. Using the proposed modified MS algorithm for applications such as image segmentation and object tracking are the subject of our future work.

Proof of Theorem 1:

(i) Let $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ denote the data set. Let $\boldsymbol{y}_{j+1} \neq \boldsymbol{y}_j$, we show $\hat{f}_{h,k}(\boldsymbol{y}_{j+1}) > \hat{f}(\boldsymbol{y}_j)$. From (2), we have

$$\begin{aligned}\hat{f}_{h,k}(\boldsymbol{y}_{j+1}) - \hat{f}_{h,k}(\boldsymbol{y}_j) &= \frac{c_{k,D}}{nh^D}\left[\sum_{i=1}^{n} k\left(\left\| \frac{\boldsymbol{y}_{j+1}-\boldsymbol{x}_i}{h} \right\|^2\right) - k\left(\left\| \frac{\boldsymbol{y}_j-\boldsymbol{x}_i}{h} \right\|^2\right)\right] \\ &\geq \frac{c_{k,D}}{nh^{D+2}} \sum_{i=1}^{n} k'\left(\left\| \frac{\boldsymbol{y}_j-\boldsymbol{x}_i}{h} \right\|^2\right) \\ &\quad \times \left(\| \boldsymbol{y}_{j+1}-\boldsymbol{x}_i \|^2 - \| \boldsymbol{y}_j-\boldsymbol{x}_i \|^2 \right),\end{aligned} \tag{18}$$

where the last inequality comes from the convexity of the profile function k, i.e. $k(x_2) - k(x1) \geq k'(x_1)(x_2 - x_1)$. By the triangle inequality, we have

$$\| \boldsymbol{y}_{j+1} - \tilde{\boldsymbol{y}}_{j+1} \| \leq \| \boldsymbol{y}_{j+1} - \boldsymbol{x}_i \| + \| \tilde{\boldsymbol{y}}_{j+1} - \boldsymbol{x}_i \|, \quad i = 1, 2, \ldots, n, \tag{19}$$

where $\tilde{\boldsymbol{y}}_{j+1}$ is given in (12). Using (18) and (19), we obtain (see (20)) . From (13), we have $\| \boldsymbol{y}_{j+1} - \tilde{\boldsymbol{y}}_{j+1} \|^2 - \| \tilde{\boldsymbol{y}}_{j+1} - \boldsymbol{x}_i \|^2 \leq 0$ for $\boldsymbol{x}_i \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, and as a result we have

$$\begin{aligned}\hat{f}_{h,k}(\boldsymbol{y}_{j+1}) - \hat{f}_{h,k}(\boldsymbol{y}_j) \geq \frac{c_{k,D}}{nh^{D+2}} \sum_{i=1}^{n} k'\left(\left\| \frac{\boldsymbol{y}_j-\boldsymbol{x}_i}{h} \right\|^2\right) \\ \times \left(\| \boldsymbol{y}_{j+1}-\tilde{\boldsymbol{y}}_{j+1} \|^2 - \| \tilde{\boldsymbol{y}}_{j+1}-\boldsymbol{x}_i \|^2 \right. \\ \left. -2 \| \boldsymbol{y}_{j+1}-\tilde{\boldsymbol{y}}_{j+1} \| \| \tilde{\boldsymbol{y}}_{j+1}-\boldsymbol{x}_i \| - \| \boldsymbol{y}_j-\boldsymbol{x}_i \|^2 \right).\end{aligned} \tag{20}$$

$$\sum_{i=1}^{n} k'\left(\| \frac{\boldsymbol{y}_j - \boldsymbol{x}_i}{h} \|^2\right) \left(\| \boldsymbol{y}_{j+1} - \tilde{\boldsymbol{y}}_{j+1} \|^2 - \| \tilde{\boldsymbol{y}}_{j+1} - \boldsymbol{x}_i \|^2\right) > 0, \tag{21}$$

where the above inequality is true since the profile k is a strictly decreasing function and $k'(x) < 0$. Furthermore, we have

$$\sum_{i=1}^{n} k'\left(\| \frac{\boldsymbol{y}_j - \boldsymbol{x}_i}{h} \|^2\right) \left(-2 \| \boldsymbol{y}_{j+1} - \tilde{\boldsymbol{y}}_{j+1} \| \| \tilde{\boldsymbol{y}}_{j+1} - \boldsymbol{x}_i \| - \| \boldsymbol{y}_j - \boldsymbol{x}_i \|^2\right) > 0. \tag{22}$$

Combining (20)–(22), we obtain

$$\hat{f}_{h,k}(\boldsymbol{y}_{j+1}) - \hat{f}_{h,k}(\boldsymbol{y}_j) > 0, \tag{23}$$

which implies that $\{\hat{f}_{h,k}(\boldsymbol{y}_j)\}_{j=1,2,\ldots}$ is an increasing sequence. From (13), it is obvious that $\boldsymbol{y}_j \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}, j = 1, 2, \ldots,$ and since n is finite then $\hat{f}_{h,k}(\boldsymbol{y}_j)$, given in (2), is bounded. Thus, as long as $\boldsymbol{y}_{j+1} \neq \boldsymbol{y}_j$, the sequence $\{\hat{f}_{h,k}(\boldsymbol{y}_j)\}_{j=1,2,\ldots}$ is a bounded and strictly increasing sequence, which two previous conditions imply the convergence of $\{\hat{f}_{h,k}(\boldsymbol{y}_j)\}$.

(ii) From (3), we have

$$\| \nabla \hat{f}_{h,k}(\boldsymbol{y}_j) \| = \frac{2c_{k,D}}{nh^{D+2}} \left[\sum_{i=1}^{n} g\left(\| \frac{\boldsymbol{y}_j - \boldsymbol{x}_i}{h} \|^2\right)\right] \| \boldsymbol{m}_{h,g}(\boldsymbol{y}_j) \|, \quad j = 1, 2, \ldots, n, \tag{24}$$

Since the profile $k(x)$ is a convex function, then $g(x)$ is a monotonically non-increasing function and we have

$$\sum_{i=1}^{n} g\left(\| \frac{\boldsymbol{y}_j - \boldsymbol{x}_i}{h} \|^2\right) \leq ng(0). \tag{25}$$

From definition of the modified MS algorithm in (12) and (13), $\boldsymbol{y}_j \in \mathcal{X}$, i.e. $\boldsymbol{y}_j = \boldsymbol{x}_i$ for some $1 \leq i \leq n$. Therefore, for some $1 \leq k, l \leq n$, we have

$$\| \boldsymbol{m}_{h,g}(\boldsymbol{y}_j) \| = \| \tilde{\boldsymbol{y}}_{j+1} - \boldsymbol{y}_j \| = \| \boldsymbol{x}_k - \boldsymbol{x}_l \| \leq d_{\max}, \tag{26}$$

where $d_{\max} = \max_{\boldsymbol{x}_i, \boldsymbol{x}_j \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}} \| \boldsymbol{x}_i - \boldsymbol{x}_j \|$. Combining (24)–(26), we obtain

$$\| \nabla \hat{f}_{h,k}(\boldsymbol{y}_j) \| \leq \frac{2c_{k,D}}{h^{D+2}} g(0) d_{\max}. \tag{27}$$

(iii) The proposed modified MS algorithm starts from one of the data points, and in each iteration, the cluster centre estimate is assigned to be one of the data points. The algorithm stops when two consecutive estimates become equal, i.e. $\boldsymbol{y}_{j+1} = \boldsymbol{y}_j$ for some $j \geq 1$. From part (i), in each iteration each data point can be assigned to the cluster centre estimate at most one time, otherwise $\hat{f}_{h,k}(\boldsymbol{y}_{j+k}) = \hat{f}_{h,k}(\boldsymbol{y}_j)$ for some $k \geq 1$ which contradicts part (i). Since the number of data samples n is finite, after a finite number of iterations, the convergence for the sequence $\{\boldsymbol{y}_j\}$ occurs. □

6 References

[1] Yuan, X.T., Hu, B.G., He, R.: 'Agglomerative mean-shift clustering', *IEEE Trans. Knowl. Data Eng.*, 2012, **24**, pp. 209–219

[2] Tao, W., Jin, H., Zhang, Y.: 'Color image segmentation based on mean shift and normalized cuts', *IEEE Trans. Syst. Man Cybern. B Cybern.*, 2007, **37**, pp. 1382–1389

[3] Wang, J., Thiesson, B., Xu, Y., *et al.*: 'Image and video segmentation by anisotropic kernel mean shift'. Proc. European Conf. Computer Vision, Prague, Czech Republic, 2004, vol. 2, pp. 238–250

[4] Aliyari Ghassabeh, Y., Rudzicz, F.: 'Incremental algorithm for finding principal curves', *IET Signal Process.*, 2015, **9**, pp. 521–552

[5] Yilmaz, A.: 'Object tracking by asymmetric kernel mean shift with automated scale and orientation selection'. Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Minnesota, USA, 2007, pp. 18–23

[6] Comaniciu, D., Ramesh, V., Meer, P.: 'Real-time tracking of non-rigid objects using mean shift'. Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Hilton Head Island, SC, USA, 2000, pp. 142–149

[7] Fukunaga, K., Hostetler, L.D.: 'Estimation of the gradient of a density function, with applications in pattern recognition', *IEEE Trans. Inform. Theory*, 1975, **2**, pp. 32–40

[8] Cheng, Y: 'Mean shift, mode seeking and clustering', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1995, **17**, pp. 790–799

[9] Comaniciu, D., Meer, P.: 'Mean shift: a robust approach toward feature space analysis', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002, **24**, pp. 603–619

[10] Fashing, M., Tomasi, C.: 'Mean shift is a bound optimization', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2005, **27**, pp. 471–474

[11] Carreira-Perpiñán, M.A.: 'Gaussian mean shift is an EM algorithm', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2007, **29**, pp. 767–776

[12] Wu, C.F.J.: 'On the convergence properties of the EM algorithm', *Ann. Stat.*, 1983, **11**, pp. 95–103

[13] Boyles, R.A.: 'On the convergence of the EM algorithm', *J. Royal Stat. Soc. B*, 1983, **45**, pp. 47–50

[14] Li, X., Hu, Z., Wu, F.: 'A note on the convergence of the mean shift', *Pattern Recognit.*, 2007, **40**, pp. 1756–1762

[15] Aliyari Ghassabeh, Y.: 'Asymptotic stability of equilibrium points of mean shift algorithm', *Mach. Learn.*, 2015, **98**, pp. 359–368

[16] Aliyari Ghassabeh, Y., Rudzica, F.: 'The mean shift algorithm and its relation to kernel regression', *Inf. Sci.*, 2016, **348**, pp. 198–208

[17] Arias-Castro, E., Mason, D., Pelletier, B.: 'On the estimation of the gradient lines of a density and the consistency of the mean-shift algorithm', *J. Mach. Learn. Res.*, 2016, **17**, pp. 1–28

[18] Arias-Castro, E., Mason, D., Pelletier, B.: 'ERRATA: on the estimation of the gradient lines of a density and the consistency of the mean-shift algorithm', *J. Mach. Learn. Res.*, 2016, **17**, pp. 1–4

[19] Aliyari Ghassabeh, Y., Linder, T., Takahara, G.: 'On some convergence properties of the subspace constrained mean shift', *Pattern Recognit.*, 2013, **46**, pp. 3140–3147

[20] Aliyari Ghassabeh, Y.: 'On the convergence of the mean shift algorithm in the one-dimensional space', *Pattern Recognit. Lett.*, 2013, **34**, pp. 1423–1427

[21] Aliyari Ghassabeh, Y.: 'A sufficient condition for the convergence of the mean shift algorithm with Gaussian kernel', *J. Multivariate Anal.*, 2015, **135**, pp. 1–10

[22] Tsybakov, A.B.: '*Introduction to nonparametric estimation*' (Springer-Verlag, New York, 2008)

[23] Silverman, B.W: '*Density estimation for statistics and data analysis*' (Chapman and Hall, London, 1986)

[24] Comaniciu, D., Ramesh, V., Meer, P: 'Mean shift: a robust approach toward feature space analysis'. Proc. Eighth Int. Conf. Computer Vision (ICCV01), Vancouver, BC, Canada, 2001, pp. 438–445

[25] Carreira-Perpiñán, M.A.: 'Fast nonparametric clustering with Gaussian blurring mean-shift'. Proc. of the 23rd Int. Conf. Machine learning (ICML06), New York, NY, USA, 2006, pp. 153–160

[26] Liu, Y., Li, S.Z., Wu, W., *et al.*: 'Dynamics of a mean-shift-like algorithms and its applications on clustering', *Inf. Process. Lett.*, 2013, **113**, pp. 8–16

[27] Coppersmith, D., Winograd, S.: 'Matrix multiplication via arithmetic progressions', *J. Symb. Comput.*, 1990, **9**, pp. 251–280

[28] Sheikh, Y.A., Khan, E.A., Kanade, T.: 'Mode-seeking by medoidshifts'. Proc. Int. Conf. Computer Vision (ICCV 07), Rio De Janeiro, Brazil, 2007, pp. 1–8

[29] Vedaldi, A., Soatto, S.: 'Quick shift and kernel methods for mode seeking'. Proc. European Conf. Computer Vision (ECCV 08), Marsielle, France, 2008, pp. 705–718

[30] Koontz, W.L.G., Narendra, P., Fukunaga, K.: 'A graph-theoretic approach to nonparametric cluster analysis', *IEEE Trans. Comput.*, 1976, **c-25**, pp. 936–944

[31] Jiang, H.: 'On the consistency of quick shift'. Advance in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, USA, 2017, pp. 46–55

[32] Fränti, P., Virmajoki, O.: 'Iterative shrinking method for clustering problems', *Pattern Recognit.*, 2006, **39**, pp. 761–765

[33] Veenman, C.J., Reinders, M.J.T., Backer, E.: 'A maximum variance cluster algorithm', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002, **2**, pp. 1273–1280