Evolving Mean Shift with Adaptive Bandwidth: A Fast and Noise Robust Approach

Qi Zhao, Zhi Yang, Hai Tao and Wentai Liu

School of Engineering,
University of California at Santa Cruz, Santa Cruz, CA 95064
{zhaoqi,yangzhi,tao,wentai}@soe.ucsc.edu

Abstract. This paper presents a novel nonparametric clustering algorithm called evolving mean shift (EMS) algorithm. The algorithm iteratively shrinks a dataset and generates well formed clusters in just a couple of iterations. An energy function is defined to characterize the compactness of a dataset and we prove that the energy converges to zero at an exponential rate. The EMS is insensitive to noise as it automatically handles noisy data at an early stage. The single but critical user parameter, i.e., the kernel bandwidth, of the mean shift clustering family is adaptively updated to accommodate the evolving data density and alleviate the contradiction between global and local features. The algorithm has been applied and tested with image segmentation and neural spike sorting, where the improved accuracy can be obtained at a much faster performance, as demonstrated both qualitatively and quantitatively.

1 Introduction

Mean shift (MS) and blurring mean shift (BMS) are nonparametric density based clustering algorithms that have received recent attention [1, 2]. Inspired by the Parzen window approach to nonparametric density estimation, both algorithms do not require prior knowledge of cluster numbers, and do not assume a prior model for the shape of the clusters. The bandwidth parameter, however, is the single and critical parameter that may significantly affect the clustering results.

Several works [3, 4] have recognized the sensitivity of the mean shift and blurring mean shift algorithms to the kernel bandwidth. When the local characteristics of the feature space differ across data, it is difficult to find an optimal global bandwidth [2]. Adopting locally estimated bandwidth has theoretical merits of improving the qualify of density estimate. These methods, however, are heavily relied on the training algorithms, and could result in poor performance if local bandwidths are inappropriately assigned. [4] calculates the bandwidth through a sample point estimate, and the algorithm works well with moderate training procedures. More sophisticated bandwidth estimation method incorporating the input data is reported in [5], with an increased computational complexity and manual efforts from domain experts.

The speed of the mean shift algorithm is heavily dependent on the density gradient of the data. In case the feature space includes large portions of flat plateaus where density gradient is small, the convergence rate of the mean shift procedure is low [1]. The problem is inherent, as the movements of data points are proportional to the density gradient. The blurring mean shift algorithm [6] was proposed to accelerate the convergence rate by moving all data points at each iteration. A notorious drawback of blurring mean shift is that a direction of larger variance converges more slowly rather than the reverse; as a result, blurring mean shift frequently collapses a cluster into a "line" by taking a number of iterations. After that, the blurring mean shift algorithm converges data slowly to the final state and may break the "line" into many segments.

In this paper, we present a new clustering algorithm that incorporates the mean shift principle, but is inherently different from the existing mean shift based algorithms. The main novelties of our algorithm are described as follows.

First, we use an energy function to describe the data points in terms of compactness. This offers a quantitative way to measure the clustering status.

Second, unlike the mean shift algorithm [2] where the data points are static, or the blurring mean shift method [6, 1] where all the data points are updated in each iteration, the evolving mean shift algorithm moves one selected point with the largest energy reduction at each iteration. As a result, the evolving mean shift procedure converges at an exponential rate as discussed in section 3.5.

Third, the evolving mean shift algorithm automatically handles noisy data early to prevent them misleading the clustering process of other data.

Lastly, the bandwidth estimation from the sample point estimators [3, 4] is applied for initialization. Unlike blurring mean shift, the bandwidth estimation in the evolving mean shift algorithm is data-driven and adaptively updated.

2 Energy Function

An energy function is defined to evaluate the compactness of the underlying dataset. Formally, given a dataset $X = \{x_i\}_{i=1...N}$ of N points, the energy of X is defined as the sum of energy from individual point $x_i|_{i=1...N}$ as

$$E(X) = \sum\nolimits_{i=1}^{N} E_{x_i}, \tag{1}$$

where

$$E_{x_i} = \sum\nolimits_{j=1, j\neq i}^{N} (E_{x_j.x_i} + E_{x_i.x_j}). \tag{2}$$

In this equation, $E_{x_j.x_i}$ is the energy contributed by point x_j to point x_i with kernel $K(x)$ and bandwidth h_{x_i},

$$E_{x_j.x_i} = f(h_{x_i})(K(0) - K(\frac{x_i - x_j}{h_{x_i}})), \tag{3}$$

where $K(x)$ is an arbitrary isotropic kernel with a convex profile $k(x)$, i.e., it satisfies $K(x) = k(|x|^2)$ and $k(x_1) - k(x_2) \geq k'(x_2)(x_1 - x_2)$. Without loss of generality, we set $k(0) = 1$ and Eq. 3 reduces to $E_{x_j.x_i} = f(h_{x_i})(1 - K(\frac{x_i - x_j}{h_{x_i}}))$. $f(h_{x_i})$ is a shrinking factor that is designed to be a monotonically increasing function of bandwidth h_{x_i}, as will be discussed in section 3.3. It is worthy mentioning that after assigning an initial global bandwidth h_0, bandwidth h becomes independent to the user and is trained by the evolving density estimates.

Let $f(0) = 0$ and it is straightforward to verify that the energy definition satisfies (1) $E(X) \geq 0$; (2) $E(X) = 0$ when fully clustered.

3 The Evolving Mean Shift (EMS) Clustering Algorithm

We outline the evolving mean shift clustering algorithm as follows:

Algorithm 1 The EMS Clustering Algorithm

Input: A set of data points X^k, where k is the index and is initialized to 0
Output: A clustered set of data points X_{EMS}

- Select one data point $x_i^k \in X^k$ whose movement could substantially reduce the energy as defined in Eq. 1. Point selection is discussed in section 3.1.
- Move x_i^k according to the EMS vector, specifically $x_i^{k+1} = x_i^k + \overrightarrow{EMS_x^k}$.
- Compute the updated bandwidth $h_{x_i}^{k+1}$ for point x_i^{k+1} according to Algorithm 2, and adjust the EMS vectors for all points using Eq. 4.
- If $E(X^k)$ satisfies the stopping criterion, stop; otherwise, set $k \leftarrow k + 1$ and go to the 1^{st} step.

As will be proven in section 3.2, moving a data point according to the EMS vector lowers the total energy. After each movement, the bandwidth is updated as will be described in section 3.3. The iterative procedure stops when the underlying feature space satisfies the criterion given in section 3.4, and section 3.5 proves the exponential convergence rate of the EMS algorithm. In this section, we use for a moment $\sum_{j \neq i}$ for $\sum_{j=1, j \neq i}^{N}$ to keep the formulations concise.

3.1 Point Selection

Selecting a point with the largest energy reduction for moving has several important benefits. First, it avoids operations of data that lead to small energy reduction (e.g. data points in plateau regions); therefore, requires less iterations. Second, it efficiently pushes loosely distributed points toward a localized peak, which prevents them being absorbed into nearby clusters with larger densities. Third, noisy data tend to be selected therefore processed at an early stage.

To select a data point with the largest energy reduction, at the initialization stage, the EMS vector is computed for each data point. Each following iteration moves a selected point to a new location according to the EMS vector, updates its bandwidth according to Algorithm 2 (section 3.3) and adjusts the EMS vectors for all the data points. Based on the adjusted EMS vectors, a new point corresponding to the largest energy reduction is selected for the next iteration.

Because of this point selection scheme and a bandwidth updating scheme as explained in section 3.3, EMS does not have convergence bias to directions and avoids over-partitioning a cluster into many small segments as blurring mean shift does. Consequentially, EMS can work well without a post combining procedure. A comparison of blurring mean shift and EMS is presented in Figure 1.

3.2 EMS Vector and Energy Convergence

Recall that the energy associated with a selected point x_i is defined in Eq. 2 as

$$E_{x_i} = \sum_{j \neq i}(E_{x_j.x_i}+E_{x_i.x_j}) = \sum_{j \neq i}[f(h_{x_i})(1-K(\frac{x_i - x_j}{h_{x_i}}))+f(h_{x_j})(1-K(\frac{x_j - x_i}{h_{x_j}}))].$$

The gradient of E_{x_i} with respect to x_i can be obtained by exploring the linearity of kernel $K(x)$ as $\nabla E_{x_i} = -2[\frac{\sum_{j \neq i}(\frac{x_j f(h_{x_i}) g(|\frac{x_i - x_j}{h_{x_i}}|^2)}{h_{x_i}^2} + \frac{x_j f(h_{x_j}) g(|\frac{x_j - x_i}{h_{x_j}}|^2)}{h_{x_j}^2})}{\sum_{j \neq i}(\frac{f(h_{x_i}) g(|\frac{x_i - x_j}{h_{x_i}}|^2)}{h_{x_i}^2} + \frac{f(h_{x_j}) g(|\frac{x_j - x_i}{h_{x_j}}|^2)}{h_{x_j}^2})} - x_i] \times \sum_{j \neq i}[\frac{f(h_{x_i})}{h_{x_i}^2} g(|\frac{x_i - x_j}{h_{x_i}}|^2) + \frac{f(h_{x_j})}{h_{x_j}^2} g(|\frac{x_j - x_i}{h_{x_j}}|^2)]$. The first bracket contains the sample evolving mean shift vector

$$\overrightarrow{EMS_{x_i}} = \frac{\sum_{j \neq i}(\frac{x_j f(h_{x_i}) g(|\frac{x_i - x_j}{h_{x_i}}|^2)}{h_{x_i}^2} + \frac{x_j f(h_{x_j}) g(|\frac{x_j - x_i}{h_{x_j}}|^2)}{h_{x_j}^2})}{\sum_{j \neq i}(\frac{f(h_{x_i}) g(|\frac{x_i - x_j}{h_{x_i}}|^2)}{h_{x_i}^2} + \frac{f(h_{x_j}) g(|\frac{x_j - x_i}{h_{x_j}}|^2)}{h_{x_j}^2})} - x_i. \tag{4}$$

As will be proven in Theorem 1, moving the point along the EMS vector with length no larger than twice of the EMS vector magnitude, the energy reduces.

Theorem 1 *Energy is reduced by moving the selected point according to the EMS vector.*

Proof. After the selected point x_i moves to $x_i^{'}$, the energy associated with $x_i^{'}$ is

$$E_{x_i^{'}} = \sum_{j \neq i}(E_{x_i^{'}.x_j} + E_{x_j.x_i^{'}}). \tag{5}$$

In this proof, we assume that the bandwidths of all the data points remain static. The cases with adaptive bandwidth are validated in section 3.3.

Without loss of generality, let $x_i = 0$. Applying the energy definition (Eq. 2) for $x_i^{'}$ and x_i, and considering the convexity of the kernel profile $k(x)$, the energy change of the dataset X is

$$\triangle E(X) = E_{x_i^{'}} - E_{x_i} \leq \sum_{j \neq i}(\frac{f(h_{x_i})}{h_{x_i}^2} g(|\frac{x_j}{h_{x_i}}|^2) + \frac{f(h_{x_j})}{h_{x_j}^2} g(|\frac{x_j}{h_{x_j}}|^2))(|x_i^{'}|^2 - 2x_i^{'} x_j). \tag{6}$$

Applying the definition of EMS vector for x_i (Eq. 4) results

$$\triangle E(X) = (\sum_{j \neq i}(\frac{f(h_{x_i})}{h_{x_i}^2} g(|\frac{x_j}{h_{x_i}}|^2) + \frac{f(h_{x_j})}{h_{x_j}^2} g(|\frac{x_j}{h_{x_j}}|^2)))(|x_i^{'}|^2 - 2x_i^{'}\overrightarrow{EMS_{x_i}})$$

Since $\sum_{j \neq i}(\frac{f(h_{x_i})}{h_{x_i}^2} g(|\frac{x_j}{h_{x_i}}|^2) + \frac{f(h_{x_j})}{h_{x_j}^2} g(|\frac{x_j}{h_{x_j}}|^2))$ is strictly positive, to guarantee the energy reduction, it is required that

$$|x_i^{'}|^2 - 2x_i^{'}\overrightarrow{EMS_{x_i}} = |x_i^{'} - \overrightarrow{EMS_{x_i}}|^2 - |\overrightarrow{EMS_{x_i}}|^2 \leq 0 \quad (7)$$

Particularly, $|x_i^{'}|^2 - 2x_i^{'}\overrightarrow{EMS_{x_i}}$ achieves the minimal value of $-|\overrightarrow{EMS_{x_i}}|^2$ when $x_i^{'} = \overrightarrow{EMS_{x_i}}$. This completes the proof.

3.3 Bandwidth Updating

To calculate the local bandwidth, a pilot density estimate is first calculated as

$$p(x_i) = \frac{1}{h_0^d} \sum_{j \neq i} K(\frac{x_i - x_j}{h_0}), \quad (8)$$

where h_0 is a manually specified global bandwidth and d is the dimension of the data space. Based on Eq. 8, local bandwidths are updated as [3, 4]

$$h_{x_i} = h_0[\frac{\lambda}{p(x_i)}]^{0.5}, \quad (9)$$

where $p(x_i)$ is the estimated density at point x_i, λ is a constant which is by default assigned to be geometric mean of $\{p(x_i)\}|_{i=1...N}$.

In each iteration, the density estimate associated with the selected point is updated using a sample point density with bandwidth estimated from Eq. 9 as

$$p(x_i^{'}) = \sum_{j \neq i} \frac{1}{h_{x_j}^d} K(\frac{x_i^{'} - x_j}{h_{x_j}}). \quad (10)$$

The procedure of updating the bandwidth is summarized as follows:

Algorithm 2 Adaptive bandwidth updating using sample point estimator

---

Input: The data point x_i^k that is selected to move in the k^{th} iteration and its corresponding bandwidth $h_{x_i}^k$

Output: An updated bandwidth $h_{x_i}^{k+1}$ for the selected point

- Calculate the updated density estimate $p(x_i^{k+1})$ for the selected point according to Eq. 10.
- Calculate the updated bandwidth $h_{x_i}^{k+1}$ for the selected point using Eq. 9 with the updated pilot density estimate $p(x_i^{k+1})$. If $h_{x_i}^{k+1} < h_{x_i}^k$, update the bandwidth with $h_{x_i}^{k+1}$; otherwise, set $h_{x_i}^{k+1} \leftarrow h_{x_i}^k$.

During iterations, the bandwidth of each data point adapts to the local density. Though Algorithm 2 only updates bandwidth when it becomes smaller, experiments show that the bandwidth associated with the selected point frequently reduces after the movement. This phenomenon is intuitive, as the EMS iteration compacts a dataset, which leads to a smaller bandwidth according to

Eq. 9. To satisfy that $E_{x_j.x_i}$ is a monotonically increasing function of $h(x)$, we have $\frac{\partial E_{x_j.x_i}}{\partial h_{x_i}} \geq 0$. For both Gaussian and Epanechnikov kernels, the requirements on $f(h_x)$ are the same

$$f(h_x) \sim O(h_x^\alpha),\ \alpha \geq 2 \tag{11}$$

3.4 Stopping Criterion

In this work, we use a broad truncated kernel with an adaptive bandwidth, based on which a reliable stopping criterion using the EMS vector or the total energy can be given. A broad truncated kernel $K_B(x)$ is defined as

$$K_B(x) = \begin{cases} K(x), & x < Mh_x \\ 0, & x \geq Mh_x, \end{cases} \tag{12}$$

where M is a positive constant satisfying that Mh_x can cover a large portion or the whole feature space. At an early stage of EMS iterations where a clear configuration of clusters has not been formed, the kernel K_B is similar to a broad kernel. Through iterations, the bandwidth reduces and converges to zero. As a result, K_B becomes a truncated kernel, which only covers a small region in the feature space and prevents the attraction of different clusters.

3.5 Convergence Rate of the EMS Algorithm with a Broad Kernel

In this section, we use the most widely used broad kernels, i.e., the Gaussian kernel and Epanechnikov kernel, with global bandwidth as examples to validate the fast convergence rate of the EMS algorithm.

Theorem 2 *The EMS algorithm with a broad kernel converges at an exponential rate.*

Proof. According to the energy definition (Eq. 3), the energy from point x_j to x_i using a broad kernel with bandwidth h is

$$E_{x_i.x_j} = f(h)(1 - K(\frac{x_i - x_j}{h})). \tag{13}$$

The EMS vector for point x_i in this case is calculated according to Eq. 4 as

$$\overrightarrow{EMS_{x_i}} = \frac{\sum_{j \neq i} x_j g(|\frac{x_j - x_i}{h}|^2)}{\sum_{j \neq i} g(|\frac{x_j - x_i}{h}|^2)} - x_i. \tag{14}$$

Theorem 2.1 Convergence rate under a broad Gaussian kernel. Begin with the Gaussian kernel, the corresponding energy reduction of moving a point from location x_i to $x_i^{'}$ according to Eq. 6 is

$$\triangle E(X) = E_{x_i^{'}} - E_{x_i} \geq \sum_{j \neq i} \frac{f(h)}{h^2} exp^{-\frac{(x_j - x_i)^2}{h^2}} (|x_i^{'} - x_i|^2 - 2(x_i^{'} - x_i)\overrightarrow{EMS_{x_i}}) \tag{15}$$

Substituting $x_i^{'} = x_i + \overrightarrow{EMS_{x_i}}$ into Eq. 15 yields

$$\triangle E(X) \geq |\overrightarrow{EMS_{x_i}}|^2 \sum_{j \neq i} \frac{f(h)}{h^2} exp^{-\frac{(x_j - x_i)^2}{h^2}}. \tag{16}$$

To obtain a convergence rate of the energy function, we project the points in the original d-dimensional space $\mathbb{R}^d$ onto an 1-dimensional space $\mathbb{R}^1$, i.e., $\forall x_i \in X \subset \mathbb{R}^d$ is projected to $u_i \in U \subset \mathbb{R}^1$. Denoting D_X and D_U as the maximal distance between points in $\mathbb{R}^d$ and its projected distance in $\mathbb{R}^1$, we have $D_U \leq D_X$. Further denote $\overrightarrow{EMS_{u_i}}$ as the projection of the EMS vector $\overrightarrow{EMS_{x_i}}$ onto $U \in \mathbb{R}^1$. Without loss of generality, assume $u_i - u_j \geq 0$ for $i \geq j$, we have

$$|\overrightarrow{EMS_{x_i}}| > |\overrightarrow{EMS_{u_i}}| = |\frac{\sum_{j \neq i} u_j exp^{-\frac{(x_j - x_i)^2}{h^2}}}{\sum_{j \neq i} exp^{-\frac{(x_j - x_i)^2}{h^2}}} - u_i|. \tag{17}$$

Particularly, for $|\overrightarrow{EMS_{u_1}}|$ and $|\overrightarrow{EMS_{u_N}}|$, we have $|\overrightarrow{EMS_{u_1}}| > exp^{-\frac{D_X^2}{h^2}} \frac{\sum_{j \neq 1}(u_j - u_1)}{N-1}$, $|\overrightarrow{EMS_{u_N}}| > exp^{-\frac{D_X^2}{h^2}} \frac{\sum_{j \neq N}(u_N - u_j)}{N-1}$. Summing them up gives

$$max(|\overrightarrow{EMS_{u_1}}|, |\overrightarrow{EMS_{u_N}}|) \geq exp^{-\frac{D_X^2}{h^2}} D_U/2. \tag{18}$$

Clearly D_U can be chosen to be as large as D_X. Combining Eq. 18 with Eq. 15, the energy reduction induced by moving the point corresponding to the largest energy reduction is

$$max(\triangle E(X)) \geq (N-1) \frac{f(h)}{4h^2 D_X^2} exp^{-2\frac{D_X^2}{h^2}} \tag{19}$$

According to the definition of energy (Eq. 1 and Eq. 13), an upper bound of the required amount of iterations from $E(X)$ to arbitrary small number ε is

$$\frac{(1 - exp^{-\frac{D_X^2}{h^2}}) ln \frac{E(X)}{\varepsilon}}{\frac{D_X^2}{4h^2} exp^{-2\frac{D_X^2}{h^2}}} N \;\sim O(N) \tag{20}$$

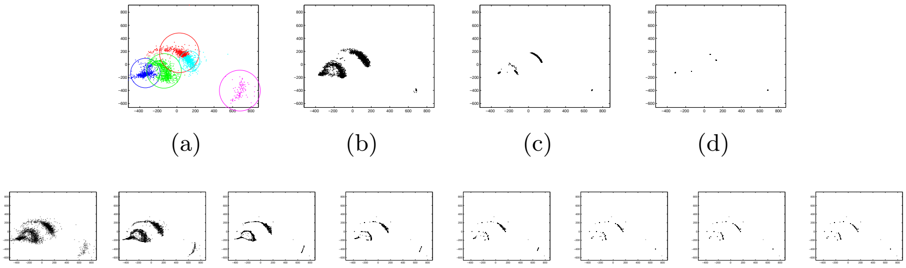
Theorem 2.2 Convergence rate under a broad Epanechnikov kernel. A broad Epanechnikov kernel can be approximated as a broad Gaussian kernel with large bandwidth $h \gg D_X$,

$$K_E(x) = \begin{cases} 1 - \frac{x^2}{h^2} \approx exp^{-\frac{x^2}{h^2}}, & if\ |x| < h \\ 0, & otherwise. \end{cases} \tag{21}$$

Applying Eq. 20 to the broad Epanechnikov kernel (Eq. 21) gives the upper bound of the number of iterations to converge as

$$\frac{(1 - exp^{-\frac{D_X^2}{h^2}}) ln \frac{E(X)}{\varepsilon}}{\frac{D_X^2}{4h^2} exp^{-\frac{D_X^2}{h^2}}} N|_{h \gg D_X} \approx 4 ln \frac{E(X)}{\varepsilon} N. \tag{22}$$

This completes the proof. In practice, the total number of iterations is usually a couple of times the total number of points.



(a) (b) (c) (d)

(e) (f) (g) (h) (i) (j) (k) (l)

Fig. 1. Performance comparison of EMS and blurring mean shift. (a) - (d) display the snapshots of EMS at $0, 1, ..., 3$ iterations per point. The grouping results shown in (a) are obtained through 5 isolated modes in (d). As a comparison, (e) - (t) display the snapshots of blurring mean shift at $0, 2, 4, ..., 14$ iterations per point.

4 Experiments

4.1 Experiments with Toy Dataset

In the first set of experiments we compare the EMS and BMS clustering methods using a toy dataset (4000 sample points) that delivers typical challenges of applications (irregular cluster geometry, density variation, sparse region, noise events, etc.). As shown in Figure 1, the first 4 - 6 iterations of BMS collapse the data into “lines”. Afterwards, the convergence speed dramatically reduces. Besides, collapsed “lines” are clearly broken into many segments. As a result, a post processing algorithm is critical.

4.2 Experiments with Image Segmentation

In the second set of experiments we apply the clustering algorithms to segment both grayscale and color images. Formally, each pixel in the image is represented by spatial and range features, i.e., $(x, y, I) \in \mathbb{R}^3$ for grayscale images and $(x, y, R, G, B) \in \mathbb{R}^5$ for color images where (x, y) denotes the image coordinate, I and (R, G, B) represent the pixel value in a grayscale or color image respectively. Figure 2 (a) - (f) display EMS results for *cameraman*. When compared with the MS and BMS algorithms [6] (Figure 2 (q) - (v)), the EMS method (Figure 2 (m) - (p)) deals better with textured regions with noticeably less noise. For the experiments on the *hand* image, the EMS iterative procedure stops after 5 iterations (compared with $20 \sim 100+$ for MS and $10 \sim 25$ for BMS).

4.3 Experiments with Neural Spike Sorting

As a third set of experiment, we conduct quantitative comparisons of the EMS, MS, and BMS algorithms on neural spike sorting [7], using 12 sequences (500

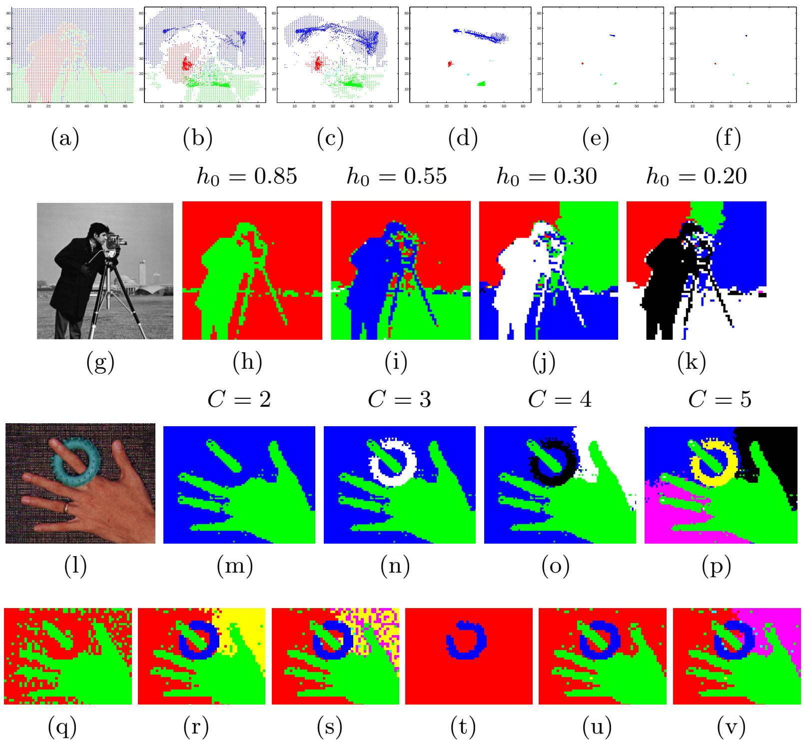


Fig. 2. (a) - (f) EMS results for *cameraman* with initial bandwidth h_0 equal to 50% of the data standard deviation, (a) - (f) correspond to 0, 0.5, 1, 1.5, 2, 3 iterations per point. (g) The original *cameraman* image (64× 64 pixels), (h) - (k) segmentation results using EMS with different h_0. (l) The original *hand* image (93×72 pixels), (m) - (p) segmentation results with fixed h_0 and different cluster numbers C. The clustering procedures are stopped after 5 iterations. (q) - (v) Segmentation results using MS ((q) - (s)) and BMS ((t) - (v)) on *hand*, copied from [6].

spikes per sequence) from a public spike data set [8]. A performance summary is listed in Table 1. The clustering accuracy is calculated as the total error subtracting the same spike detection error [7].

We also study the convergence rates of the EMS, MS and BMS algorithms. Since one necessary condition of the stopping criteria for all algorithms is that the movements of the data points, i.e., the EMS vectors for the EMS algorithm and the root mean square of the MS vectors for the MS and BMS algorithms approach zero, we use the magnitude of the data movement as a quantitative measure for the compactness of the the clusters. In these experiments, the same initial bandwidth is assigned. To test the convergence speed of each algorithm, the EMS/MS vector magnitude is curve fitted with $10^{-\alpha \cdot N}$ where N is the number of iterations and α is a parameter describing the convergence speed.

Algorithm	EMS	MS	BMS
Accuracy	97%	89 %	94%
α	0.99±0.11	0.16 ±0.13	0.3±0.23
Iterations	$2 \sim 6$	$15 \sim 50+$	$8 \sim 18$

Table 1. Quantitative results over 12 sequences from a public spike data base [8].

5 Conclusions

This paper presents an evolving mean shift algorithm. It defines an energy function to quantify the compactness of the dataset and iteratively collapses the data to isolated clusters in a couple of iterations. The single parameter of bandwidth is initialized based on sample point estimators and updated to accommodate the evolving procedure. The main theoretical contributions in this work are the validations of two theorems stating that the evolving mean shift procedure converges, and further, the energy reduces at an exponential rate which guarantees an extremely efficient convergence. Experiments with different data dimensionality demonstrate the advantage of EMS in terms of accuracy, robustness, speed, and ease of parameter selection. The low computational cost and superior performance makes it suitable to apply in many other practical tasks or subjects in additional to those mentioned in this paper.

References

1. Cheng, Y.: Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence **17**(8) (August 1995) 790–799
2. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**(5) (May 2002) 603–619
3. Comaniciu, D., Ramesh, V., Meer, P.: The variable bandwidth mean shift and data-driven scale selection. In: IEEE International Conference on Computer Vision. (2001) I: 438–445
4. Hall, P., Hui, T., Marron, J.: Improved variable window kernel estimate of probability densities. The Annals of Statistics **23**(1) (September 1995) 1–10
5. Comaniciu, D.: An algorithm for data-driven bandwidth selection. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(2) (February 2003) 281–288
6. Carreira-perpinan, M.: Fast nonparametric clustering with gaussian blurring mean-shift. In: International Conference on Machine Learning. (2006) 153–160
7. Yang, Z., Zhao, Q., Liu, W.: Spike feature extraction using informative samples. In: Advances in Neural Information Processing Systems. (2009) 1865–1872
8. Quian Quiroga, R., Nadasdy, Z., Ben-Shaul, Y.: Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. Neural Computation **16**(8) (August 2004) 1661–1687