Fast Global Kernel Density Mode Seeking: Applications to Localization and Tracking

Chunhua Shen, Michael J. Brooks, *Member, IEEE*, and Anton van den Hengel, *Member, IEEE*

***Abstract*—Tracking objects in video using the mean shift (MS) technique has been the subject of considerable attention. In this work, we aim to remedy one of its shortcomings. MS, like other gradient ascent optimization methods, is designed to find local modes. In many situations, however, we seek the global mode of a density function. The standard MS tracker assumes that the initialization point falls within the basin of attraction of the desired mode. When tracking objects in video this assumption may not hold, particularly when the target's displacement between successive frames is large. In this case, the local and global modes do not correspond and the tracker is likely to fail. A novel multibandwidth MS procedure is proposed which converges to the global mode of the density function, regardless of the initialization point. We term the procedure *annealed MS*, as it shares similarities with the annealed importance sampling procedure. The bandwidth of the procedure plays the same role as the *temperature* in conventional annealing. We observe that an over-smoothed density function with a sufficiently large bandwidth is unimodal. Using a continuation principle, the influence of the global peak in the density function is introduced gradually. In this way, the global maximum is more reliably located. Since it is imperative that the computational complexity is minimal for real-time applications, such as visual tracking, we also propose an accelerated version of the algorithm. This significantly decreases the number of iterations required to achieve convergence. We show on various data sets that the proposed algorithm offers considerable promise in reliably and rapidly finding the true object location when initialized from a distant point.**

***Index Terms*—Annealing, fast mean shift (MS), global density mode, visual localization, visual tracking.**

I. Introduction and Motivation

KERNEL-BASED density estimation techniques for computer vision have attracted a great deal of attention. One example is the mean shift (MS) technique which has been applied to image segmentation, visual tracking, etc. [1]–[7]. MS is a versatile nonparametric density analysis tool introduced in [8]–[10]. In essence, it is an iterative mode detection algorithm in the density distribution space. The MS algorithm moves to a kernel-weighted average of the observations within a smoothing window. This computation is repeated until convergence is attained at a *local* density mode. This way the density modes can be elegantly located without explicitly estimating the density.

Manuscript received May 5, 2006; revised November 16, 2006. National ICT Australia, Ltd., is funded through the Australian Government's Backing Australia's Ability initiative, in part through the Australian Research Council. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Gaudenz Danuser.

C. Shen was with the School of Computer Science, University of Adelaide, SA 5005, Australia. He is now with the National ICT Australia, Ltd., Canberra ACT 2601, Australia (e-mail: chunhua.shen@nicta.com.au).

M. J. Brooks and A. van den Hengel are with the School of Computer Science, University of Adelaide, SA 5005, Australia (e-mail: mjb@cs.adelaide.edu.au; hengel@cs.adelaide.edu.au).

Digital Object Identifier 10.1109/TIP.2007.894233

Cheng [9] notes that MS is fundamentally a gradient ascent algorithm with an adaptive step size. Fashing and Tomasi [11] show the connection between MS and the Newton–Raphson algorithm. They also discover that MS is actually a quadratic bound optimization both for stationary and evolving sample sets. MS is also a fixed-point iteration procedure.

Since Comaniciu *et al.* [2] first introduced MS-based object tracking, it has proven to be a promising alternative to popular particle filtering based trackers [12], [13]. A number of improvements to the method have been reported in the literature. In [3], the selection of kernel scale via linear search is discussed. Elgammal *et al.* [4] reformulate the tracking framework as a general form of joint feature-spatial distributions [7]. Compared with the approach of Comaniciu *et al.*, the advantage is that spatial structure information of the tracked region is incorporated.

In [5], multiple spatially distributed kernels are adopted to accurately capture changes in the target's orientation and scale. Another approach is developed in [14] for the same purpose. Furthermore, Fan *et al.* [15] present a theoretical analysis of the similarity measure and arrive at a criterion, leading to kernel design strategies with prevention of singularity in kernel visual tracking. All the above mentioned trackers adopt MS or similar optimization strategies. Despite successful applications, MS trackers require that the displacement of the tracked target in consecutive frames is small as the search is initialized by the detected location of the target in the previous frame. Larger interframe displacements will lead the tracker to become trapped in spurious locations in the multimodal density distribution space[1] because MS is a *local* optimization method.

Fundamentally, MS has two important inherent drawbacks: 1) the fact that it is designed to find local rather than global modes and 2) its speed. Simulated annealing is often employed to attain global, rather than local, optimization—initially sampling with a reduced sensitivity to the underlying modes (on the flattened cost function surface) and then progressively increasing the sensitivity to drive samples on peaked cost regions [17]. Recently, the idea of annealing has been merged into importance sampling yielding annealed importance sampling [18], and it has also been introduced into 3-D articulated tracking [19].

Motivated by the success of both simulated annealing and annealed importance sampling, we propose a novel multibandwidth MS procedure, termed *annealed MS*. It shares similarities with the annealed importance sampling procedure in the sense that it also gradually smooths the cost function surface and gently introduces the influence of the global peak. We observe that an over-smoothed density function with a sufficiently large

[1]In contrast, particle filtering-based trackers (e.g., [16]) perform better in this situation. However, weak dynamical modeling also presents challenges to particle filters.

bandwidth[2] h_M is uni-modal. Then, with a continuation principle, we slowly decrease the bandwidth $h = h_M > h_{M-1} > \cdots > h_0$ and, at each bandwidth, we maximize the density (cost) function with MS, starting from the convergence position of the previous run. This multibandwidth MS iteration process is similar to the multilayered annealing procedure of annealed importance sampling. The main differences are as follows. 1) In annealed MS, it is the degree of smoothness of the cost function that is annealed, while in annealed importance sampling, it is the degree of flatness of the cost function. 2) Most importantly, in annealed MS, the number and positions of the modes are evolved slowly while in the annealed importance sampling, the temperature does not change the number of modes or their positions. In theory, as long as the change of bandwidth is sufficiently slow, the global maximum can be found successfully.[3] We provide technical details later.

The second drawback of MS is that it often converges slowly. In some cases, the proposed annealed MS might require more iterations than the standard version of the algorithm. This is particularly the case when applied to the localization problem which involves finding the target in an image with no prior knowledge of its location. Clearly, it is imperative that the computation complexity is minimal in real-time applications such as visual tracking. To the best of our knowledge, few attempts have been made to speed up the convergence of MS. In [1], locality sensitive hashing (LSH) is used to reduce the computational complexity of finding the nearest neighbors of a sample point involved in MS. The kd-tree can also be used to reduce the large number of nearest-neighbor queries [21]. Although a dramatic decrease in the execution time is achieved for high-dimensional clustering, these techniques are not that attractive for relatively low-dimensional problems such as visual tracking. Zhang *et al.* [22] partition the feature space into several clusters and the samples of each cluster are treated as a whole in describing the density distribution. MS iteration is then approximated by evaluating kernels only on the cluster centers. The acceleration of all these approaches is not obtained by reducing the number of iteration steps. Yang *et al.* [23] use quasi-Newton methods to perform gradient hill climbing, in which the convergence rate is super-linear. Unfortunately, Newton methods can overshoot and break the convergence [11]. A line search might be employed to determine the search direction in each iteration, which will introduce extra computation.

In this paper, we prefer an accelerated version of the MS algorithm. Compared with the conventional MS algorithm, it can significantly decrease the number of iterations required for convergence. The accelerated MS is inspired by the successful accelerated variants of bound optimization algorithms such as expectation maximization (EM). An over-relaxed strategy is adopted to accelerate convergence. Much effort has been expended to improve the efficiency of bound optimization algorithms (e.g., EM, [24]–[26]). A theoretical analysis of the convergence properties for a class of bound optimization algorithms has been given in [26] and is used as the basis for a novel adaptive over-relaxed scheme. Our proposal is inspired by this approach. Based on the findings in [11], which bridge the gap between MS and general bound optimization algorithms, we promote an adaptive over-relaxed MS algorithm which is simple to implement yet significantly more efficient than the standard counterpart.

Applications of the proposed fast, globally mode-seeking MS are given in the form of an annealed MS-based object localizer and a visual tracker. Substantially more promising results have been achieved over the conventional MS-based algorithms. The work described here extends our previous efforts published in short form in [27]. In summary, our key contributions comprise the following.

1) The development of a novel annealed MS algorithm which can more reliably find the global mode of a density distribution. This is introduced in Section III.
2) The reinterpretation of the MS algorithm, leading to an accelerated version of MS, attaining considerable speedup. We discuss these issues in Section IV.
3) The application of annealed MS to the problem of visual tracking using kernel-weighted color histogram features. Given a target model, the tracker is able to initialize automatically. It also has the capability to recover from tracking failures caused by occlusions or drastic illumination changes, in that the tracker itself can also be a localizer. In contrast, conventional MS trackers lack these desirable properties. These developments, including experimental results, are presented in Section V-B and C.

The remaining contents start with a brief review of the standard MS algorithm for completeness in Section II. We conclude the paper in Section VI with a discussion of some important issues.

II. Mean Shift Analysis

We first review the basic concepts of the MS algorithm using the notation similar to [10]. One of the most popular nonparametric density estimators is *kernel density estimation*. Given n data points $\mathbf{x}_i, i = 1, \ldots, n$, drawn from a population with density function $f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^d$, the general multivariate kernel density estimate at $\mathbf{x}$ is defined by

$$\hat{f}_K(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{x}_i) \tag{1}$$

where $K_{\mathbf{H}}(\mathbf{x}) = |\mathbf{H}|^{-(1/2)} K(\mathbf{H}^{-(1/2)}\mathbf{x})$. Here, $K(\cdot)$ is a kernel function (or window) with a symmetric positive definite bandwidth matrix $\mathbf{H} \in \mathbb{R}^{d \times d}$. A kernel function is bounded with support satisfying the regularity constraints as described in [9], [10]. For simplicity, one usually assumes an isotropic bandwidth which is proportional to the identity matrix, i.e., $\mathbf{H} = h^2\mathbf{I}$. Employing the profile definition, the kernel density estimator becomes

$$\hat{f}_K(\mathbf{x}) = \frac{c_k}{nh^d} \sum_{i=1}^{n} k\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right) \tag{2}$$

where $k(\cdot)$ is the profile of the kernel $K(\cdot)$ and c_k [also c_g in (3)] is a normalization constant. The optimization procedure of

[2]By a sufficiently large bandwidth, we mean a bandwidth which is much larger than the optimal bandwidth with the minimum asymptotic mean integrated square error (AMISE). AMISE is a measure of distance between two densities for understanding the performance of a kernel density estimator [20].

[3]For continuous variables, the assertion of success is probabilistic.

seeking the local modes is solved by setting the gradient equal to zero. Thus, we have

$$\hat{\nabla} f_K(\mathbf{x}) \equiv \nabla \hat{f}_K(\mathbf{x}) = \frac{2c_k}{h^2 c_g} \hat{f}_G(\mathbf{x}) \cdot \mathbf{m}_G(\mathbf{x}) = 0 \quad (3)$$

where

$$\hat{f}_G(\mathbf{x}) = \frac{c_g}{nh^d} \sum_{i=1}^{n} g\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right) \quad (4)$$

$$\mathbf{m}_G(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\| \frac{\mathbf{x}-\mathbf{x}_i}{h} \right\|^2 \right)}{\sum_{i=1}^{n} g\left(\left\| \frac{\mathbf{x}-\mathbf{x}_i}{h} \right\|^2 \right)} - \mathbf{x} \quad (5)$$

and $g(x) = -k'(x)$. Here, $k(\cdot)$ is called the shadow of the profile $g(\cdot)$ [11], and $\mathbf{m}_G(\mathbf{x})$ is the MS vector. Clearly, $\hat{\nabla} f_K(\mathbf{x}) = 0$ implies $\mathbf{m}_G(\mathbf{x}) = 0$, and the incremental iteration scheme is obtained immediately[4]

$$\mathbf{x} \leftarrow \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\| \frac{\mathbf{x}-\mathbf{x}_i}{h} \right\|^2 \right)}{\sum_{i=1}^{n} g\left(\left\| \frac{\mathbf{x}-\mathbf{x}_i}{h} \right\|^2 \right)}. \quad (6)$$

III. Annealed MS

Let $h_m (m = M, M-1, \ldots, 0)$ be a monotonically decreasing sequence of bandwidths such that h_0 is the optimal bandwidth for the considered data set and usually $h_M \gg h_0$.[5] A series of kernel density functions $\hat{f}_{h_M,K}(\cdot), \hat{f}_{h_{M-1},K}(\cdot), \ldots, \hat{f}_{h_0,K}(\cdot)$ are applied to the sample data, where the subscripts of $\hat{f}_{h,K}(\cdot)$ denote the bandwidth and kernel type, respectively.

Fig. 1 illustrates a 1-D example,[6] where $M = 6$. With a large bandwidth, the function $\hat{f}_{h_M,K}(\cdot)$ is uni-modal, merely representing the overall trend of the density function. Thus, the starting point of the first annealing run does not affect the mode detection. This is guaranteed by the following theorem.

Theorem 3.1 [Hall et al. [30]]: If the kernel $K(\cdot)$ in (1) is compactly supported and strictly uni-modal, and is concave in a neighborhood of its mode, and if the data $\mathbf{x}$ are drawn from a continuous distribution, then with probability 1, there exists a bandwidth h_M such that $\hat{f}_{h_M,K}$ is strictly uni-modal for all $h > h_M$ (our notation is used here).

Refer to [30] for the proof. When the kernel is Gaussian, it is well known that the number of modes of $\hat{f}$ is monotonically nonincreasing in the bandwidth.

The basic annealed MS algorithm is shown in Fig. 2.

A. Remarks

The annealing schedule is a tradeoff between efficiency and efficacy: Slow annealing is more likely to find a global maximum but could also be prohibitively expensive.

[4]Refer to [9] and [10] for the convergence proof.

[5]There is a tremendous amount of literature on how to select the optimal bandwidth in order to produce a minimum AMISE estimate (see, e.g., [28] and [29]). In this work, we assume h_0 can be obtained by existing techniques.

[6]The 1-D galaxy velocity data set is also used in [11].

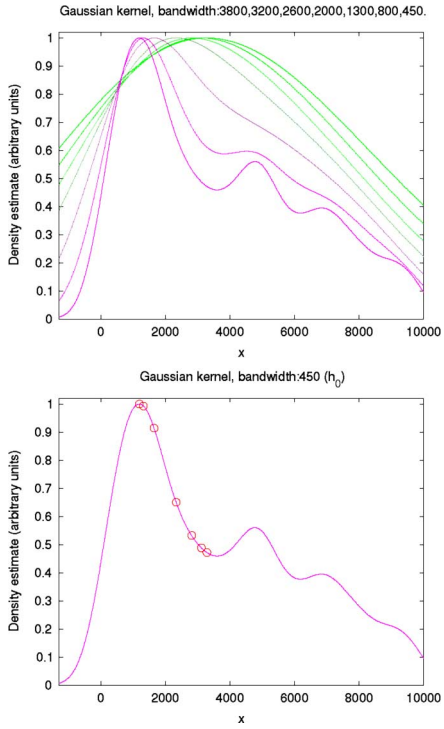

Fig. 1. Multibandwidth density estimate on 1-D galaxy velocity data. (Top) Curves from top to bottom indicate the *annealing* process with successively decreasing bandwidths. In this case, the optimal bandwidth is $h_0 = 450$. The evolution of the modes is clearly shown: With a multibandwidth MS mode detection, it is possible to find the global maximum without being distracted by local modes. (Bottom) Convergence positions at each bandwidth are marked with circles in the last curve. Note that the unit of the vertical axis is arbitrary.

1) Determine the set of values for $h_m, (m = M \cdots 0)$ (*a.k.a.* the annealing schedule).
2) Randomly select an initial starting location for the first annealing run and get the convergence location of $\hat{f}_{h_M,K}(\cdot)$, which is $\hat{\mathbf{x}}^{(M)}$, using *mean shift*.
3) For each $m = M-1, M-2, \cdots, 0$, run *mean shift* to get the convergence position $\hat{\mathbf{x}}^{(m)}$ with the initial position $\hat{\mathbf{x}}^{(m+1)}$, *i.e.*, the convergence position from the previous bandwidth. $\hat{\mathbf{x}}^{(0)}$ is then the final global mode.

Fig. 2. Annealed MS algorithm.

Annealed MS works well because the number of modes of a kernel estimator with a Gaussian kernel is monotonically nonincreasing. Annealed MS utilizes this property. A precondition of the success of the Annealed MS hierarchical search is the monotonicity of number of modes with respect to bandwidths. Note that if a non-Gaussian kernel is adopted, e.g., Epanechnikov kernel, the monotonicity property does not apply because compactly supported kernels may not have this property. However, as pointed out in [30], the lack of monotonicity typically occurs only for relatively small bandwidths. The notion of a critical bandwidth[7] for the popular kernels such as the Epanechnikov kernel is still well defined. Moreover, "just as in the Gaussian case, the critical bandwidth is of the same size as the bandwidth (h_0) that minimizes mean square error of the density estimator" [30]. This conclusion serves as one of the theoretical bases of

[7]The smallest bandwidth above which the number of modes is monotone.

Annealed MS: We are not interested in the bandwidth under h_0. Rather, we take advantage of the property of *over-smoothness* at bandwidths *above* h_0. Therefore, for the applications that we are interested in, e.g., visual localization and tracking, the problem of nonmonotonicity does not arise.

Unless otherwise specified, in our examples the (truncated) Gaussian kernel is used as it leads to fast computation. There are two reasons. 1) As we will see in Section IV, a Gaussian kernel is preferred for our fast MS. 2) Fast Gauss transform [31] can also be adopted to reduce the computational burden. When the Gaussian kernel is adopted, the mechanism is related to the well-developed scale-space theory [32], [33]. The over-smoothed kernel density is essentially a Gaussian smoothed version of the true density, obtained via convolution with an extra Gaussian kernel. The key idea of linear Gaussian scale space is blurring the original function $f(\mathbf{x})$ with a Gaussian kernel $\mathcal{N}$ of bandwidth h

$$f_{h,\mathcal{N}}(\mathbf{x}) \equiv (\mathcal{N} * f)(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^d h^d}} \int \exp -\frac{\|\mathbf{x}^\star\|^2}{2h^2} f(\mathbf{x}-\mathbf{x}^\star) d\mathbf{x}^\star. \tag{7}$$

$f_{h,\mathcal{N}}(\mathbf{x})$ becomes smooth and represents a coarser property of $f(\mathbf{x})$ when the bandwidth increases.

We note that, in the statistics literature, Chaudhuri and Marron [32] have proposed an algorithm *SiZer* to explore the significant modes in an estimated curve across multiple scales. The proposed annealed MS also relates the well known graduated nonconvexity (GNC) algorithm [34]. GNC provides a better solution by finding a set of minima along a sequence of smoothed energy functions, starting from a convex energy and progressing towards the original energy function. In computer vision, a similar strategy, termed variable-bandwidth density-based fusion (VBDF), has also been adopted to find the most significant mode of a density function in the context of information fusion for multiple motion estimation [35]. However, there are no theoretical details given in [35]. Moreover, VBDF was initially proposed for the topic of information fusion. To our knowledge, our work is the first to apply it in a very different—tracking/localization—context. We independently develop annealed MS mainly inspired by simulated annealing and annealed importance sampling. We have shown a strong connection between annealed MS and these annealing techniques. Theoretical justification is also given to show why annealed MS works. Furthermore, we use it in a novel way to solve some problems in robust visual localization and tracking.

B. Numerical Examples

One-Dimensional Example: Fig. 1 shows a simple 1-D example on the galaxy data [11]. Because of the density estimator's uni-modal property at a large bandwidth (h_M), the start position at h_M has no affect on the final convergence. Fig. 1 shows that the global maximum is successfully located with a rough seven-step annealing schedule. For this particular case, it turns out that only two steps are needed to locate the global mode.

2-D Example: For this example, the data are drawn from a Gaussian mixture $0.1\cdot\mathcal{N}([-1,0]^\top, 0.13\mathbf{I}) + 0.2\cdot\mathcal{N}([1,2]^\top, \mathbf{I}) + 0.7\cdot\mathcal{N}([1,-2]^\top, \mathbf{I})$, where $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, and $\mathbf{I}$ is an identity matrix.

A four-step annealed MS with bandwidths $\{2, 1.02, 0.66, 0.45\}$ is used to locate the global mode. Fig. 3 depicts the annealing process. Again, due to the uni-modal property, no matter where annealed MS starts, the global mode is always obtained eventually. A video sequence (*GMM2D.avi*)[8] is also generated to show the mode evolution process more elaborately.

For these two examples, we do not assume any prior information about the structure of the data. The only information needed is the approximate range of the data, which is usually available.

IV. Fast Mean Shift

Generally, searching across multiple bandwidths of annealed MS could increase computational requirements. It is imperative that the computational complexity is minimal in real-time applications, such as visual tracking. We introduce a novel procedure in this section.

A. Adaptive Over-Relaxed MS

The following two theorems form the basis of the adaptive over-relaxed MS algorithm.

Theorem 4.1 [Cheng [9]]: Mean shift with kernel $G(\cdot)$ finds the modes of the density estimate with kernel $K(\cdot)$, i.e., $\hat{f}_K(\cdot)$, where $K(\cdot)$ is the shadow of the kernel $G(\cdot)$.

With the analysis in Section II, Theorem 4.1 is evident.

Theorem 4.2 [Fashing and Tomasi [11]]: Mean shift with kernel $K(\cdot)$ is a quadratic bound optimization over a density estimate with a continuous shadow of $K(\cdot)$.

These two theorems show that MS is actually a bound maximization. One step of the MS procedure of (6) finds the exact maximum of the lower bound of the objective function $\hat{f}_K(\mathbf{x}^{(\kappa)})$, where $\kappa = 1, 2, \ldots$, denotes the iteration index. From (3), we have $\mathbf{m}_G(\mathbf{x}) \propto \hat{\nabla} f_K(\mathbf{x}) / \hat{f}_G(\mathbf{x})$,, which means that MS is a gradient ascent algorithm with adaptive step size. Hence, its convergence rate is better than conventional fixed-step gradient algorithms and no step-size parameters need to be tuned. As we will see, however, from the viewpoint of bound optimization, the learning rate can be over-relaxed to make its convergence faster. The following theorem shows the asymptotic convergence property of MS.

Theorem 4.3: Generally, the asymptotic convergence rate of MS is linear. Therefore, it is slow to converge. Its asymptotic convergence rate depends on the value

$$\frac{2}{h^2} \cdot \frac{\sum_{i=1}^n (\mathbf{x}^* - \mathbf{x}_i)^2 r\left(\left\|\frac{\mathbf{x}^*-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{\mathbf{x}^*-\mathbf{x}_i}{h}\right\|^2\right)}$$

where $r(x) = -g'(x)$, i.e., $g(\cdot)$ is the shadow of the profile $r(\cdot)$. $\mathbf{x}^*$ is the local maximum/fixed point. The smaller the value, the faster MS converges.

[8]The videos mentioned in this paper can be accessed at http://www.cs.adelaide.edu.au/~vision/demo/index.html.

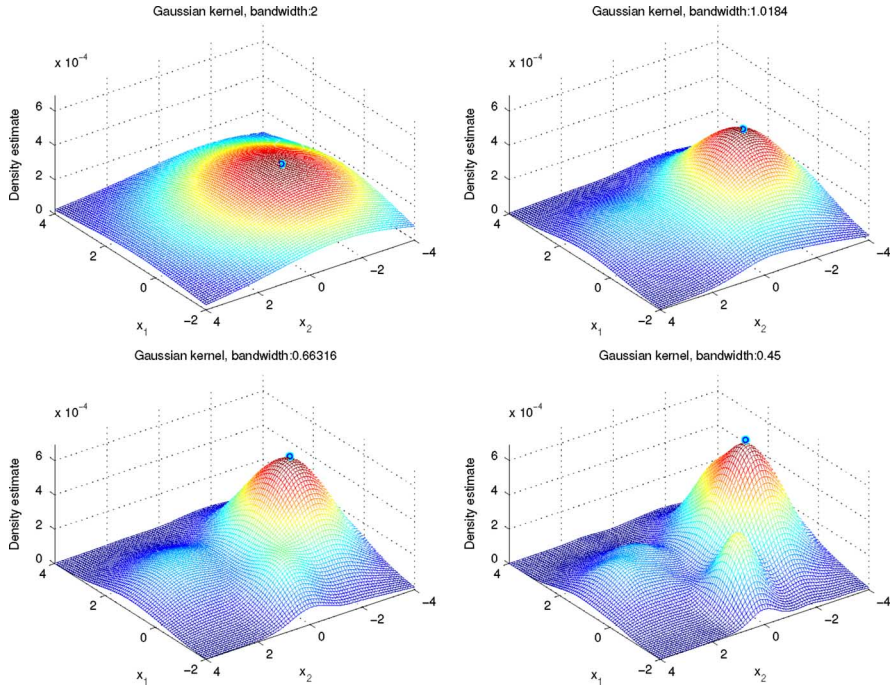

Fig. 3. Multibandwidth density estimate on 2-D artificial Gaussian mixture data. A four-step annealing schedule is employed to find the global mode. The modes found by MS across bandwidths are marked with circles. See the sequence *GMM2D.avi* which demonstrates a slower evolution across bandwidths.

When a piecewise constant profile is employed, the convergence rate is of order 2.

See Appendix for the proof.

In [36], it is proven that the fixed-point iteration algorithm for Gaussian mixture model can also be interpreted as a special case of an EM algorithm. The theoretical analysis there should also apply to mixture of other general kernel models. With these results, we know that the convergence rate is linear. Therefore, in most cases, it is very slow unless the mixture components are well separated.[9] [26]

From another point of view, bound optimization methods must usually adopt conservative bounds in order to guarantee increasing the cost function value at each iteration, leading to slow convergence [24], [26]. A lot of work has been carried out to speed up bound optimization methods, especially for the EM algorithm due to its popularity [25]. Recently, in [26], it was shown that by over-relaxing the step size, acceleration can be achieved. Denote the bound function as $\rho(\mathbf{x}, \mathbf{x}^{(\kappa)})$, then the over-relaxed bound optimization iteration is given by

$$\mathbf{x}^{(\kappa+1)} = \mathbf{x}^{(\kappa)} + \beta \left[\arg\max_{\mathbf{x}} \rho(\mathbf{x}, \mathbf{x}^{(\kappa)}) - \mathbf{x}^{(\kappa)} \right]. \quad (8)$$

Clearly, when the learning rate $\beta = 1$, over-relaxed optimization reduces to the standard bound optimization algorithm. It is easily seen that when $\beta > 1$ acceleration is realized. Nevertheless, by simply setting β to a fixed value, no convergence is secured and it seems quite difficult, if not impossible, to obtain the optimal value for β. Xu proves that in the case of the Gaussian mixture model parameter estimation with EM, convergence can be guaranteed using this method when we are *close* to a local maximum and $0 < \beta < 2$ [24]. This conclusion is generalized to the case of general bound optimization methods in [26]. Based on this important proposition, a simple adaptive over-relaxed bound optimization is readily available: The learning rate β can be adjusted by evaluating the cost function. If one observes that, for some $\beta > 1$, the cost function value worsens (becomes smaller), then β has been set too large and needs to be reduced. Simply setting $\beta = 1$ immediately, convergence can still be achieved. By regarding MS as a special case of bound optimization, these theoretical conclusions also apply to MS.

The accelerated MS algorithm obtained in this way is shown in Fig. 4. One can easily check that the following relation holds (up to a translation and a scale factor):

$$\begin{aligned} \hat{f}_K\left(\mathbf{x}^{(\kappa+1)}\right) &= \rho\left(\mathbf{x}^{(\kappa+1)}, \mathbf{x}^{(\kappa+1)}\right) \geq \rho\left(\mathbf{x}^{(\kappa+1)}, \mathbf{x}^{(\kappa)}\right) \\ &\geq \rho\left(\mathbf{x}^{(\kappa)}, \mathbf{x}^{(\kappa)}\right) = \hat{f}_K\left(\mathbf{x}^{(\kappa)}\right). \end{aligned}$$

Note that, in the above analysis, we do not take the MS with a weight function into consideration, but the accelerated algorithm also applies for the weighted case, because the two theorems concerned are derived from the weighted MS [9], [11]. The only overhead is the evaluation of the cost function. However, we will see that for the (truncated) Gaussian kernel, its special structure means that computing the MS iteration with (6) also results in evaluation of the cost function $\hat{f}_K(\mathbf{x})$. Because the shadow of the Gaussian kernel is itself, we have $\hat{f}_K(\mathbf{x}) = \hat{f}_G(\mathbf{x})$. The following theorem tells us that the Gaussian kernel and its truncated version are the only kernels that have this property.

Theorem 4.4 [Cheng [9]]: The only kernels that are their own shadows are the Gaussian kernel and its truncated version.

[9]In this case, the convergence rate is super-linear.

1) *Initialisation:*
Set iteration index $\kappa = 1$, learning rate $\beta = 1$, and step parameter $\alpha > 1$.
2) *Iterate until convergence condition is met:*
   a) Calculate $\widetilde{\mathbf{x}}^{(\kappa+1)}$ with Equation (6).
   Calculate mean shift $\mathbf{m}_G(\mathbf{x}^{(\kappa+1)}) = \widetilde{\mathbf{x}}^{(\kappa+1)} - \mathbf{x}^{(\kappa)}$.
   b) $\mathbf{x}^{(\kappa+1)} = \mathbf{x}^{(\kappa)} + \beta \cdot \mathbf{m}_G(\mathbf{x}^{(\kappa+1)})$.
   c) *if* $\hat{f}_K(\mathbf{x}^{(\kappa+1)}) > \hat{f}_K(\mathbf{x}^{(\kappa)})$,
   Accept $\mathbf{x}^{(\kappa+1)}$ and $\beta = \alpha \cdot \beta$;
   else
   Reject $\mathbf{x}^{(\kappa+1)}$, $\mathbf{x}^{(\kappa+1)} = \widetilde{\mathbf{x}}^{(\kappa+1)}$, and $\beta = 1$.
   d) Set $\kappa = \kappa + 1$. Start a new iteration.

Fig. 4. Over-relaxed adaptive MS algorithm.

The proof is straightforward. We have $g(x) = -k'(x) = k(x)$ and $k(\cdot)$ also must satisfy the conditions being a kernel. Then taking the integral of both sides leads to the conclusion that $k(\cdot)$ is the Gaussian kernel. If discontinuities are allowed in $k(\cdot)$, it can also be the truncated Gaussian kernel.

A question naturally arises, what if a kernel other than Gaussian, e.g., Epanechnikov kernel, is adopted? The observation that we can reliably judge the behavior of $\hat{f}_K(\mathbf{x})$ through the estimate $\hat{f}_G(\mathbf{x})$ is only satisfied when these two kernel functions generate density estimates of the same degree of smoothness. For different kernels, as long as the bandwidths are adjusted accordingly, all of the kernels are asymptotically equivalent under the AMISE error criterion. Therefore, the kernel type is not of importance in MS analysis but the bandwidth plays a critical role. For a non-Gaussian kernel, the shadow is different from itself ($\hat{f}_K(\mathbf{x}) \neq \hat{f}_G(\mathbf{x})$). The smoothness of two kernel density estimates with the same bandwidth but different kernels might also be quite different. As a consequence, usually we cannot reuse the density $\hat{f}_G(\mathbf{x})$ calculated in (6) and an extra evaluation of the cost function $\hat{f}_K(\mathbf{x})$ needs to be made.

In fact, if the bandwidths of two different kernels h_A, h_B satisfy

$$\frac{h_A}{h_B} = \frac{\delta_{0,A}}{\delta_{0,B}}$$

where δ_0 is a kernel's *canonical bandwidth*, then the density estimates based on these two kernels have the same degree of smoothness [20]. Utilizing this knowledge, if the canonical bandwidths associated with a kernel and its shadow kernel are, in practice, comparable, we still can reuse $\hat{f}_G(\mathbf{x})$. Although no details on this topic are presented in this paper, we have validated this conclusion with numerical experiments. However, one should be aware that the measurement of *comparable* is application dependent.

B. Numerical Experiments

Mode Seeking: We compare the performance of the proposed accelerated MS algorithm with the standard MS algorithm on both synthetic data and real application data sets. Note that rejected iterations are also counted for the accelerated MS algorithm.

The test data sets we use in the experiment are described as follows.

TABLE I
COMPARISON OF NUMBER OF ITERATIONS FOR CONVERGENCE. THE INITIAL LOCATION FOR EACH RUN IS SHOWN IN THE SECOND COLUMN

data set	*initial*	*number of iterations*	
		fast mean shift	mean shift
data set #1	-0.8	13	51
	1.5	16	77
	3.6	11	33
data set #2	9800	12	49
	-1005	8	15
	3200	10	31
data set #3	$(-5, 20)$	12	34
	$(-10, 16)$	11	29
	$(20, 10)$	13	35
data set #4	$(1, -1.4)$	29	119
	$(1.5, 0.4)$	17	65
	$(0.3, 0.3)$	12	36

1) Data set #1 (1-D synthetic data). A total of 1000 data points are drawn with equal probability from four normals: $\mathcal{N}(3,1), \mathcal{N}(1,1), \mathcal{N}(0,1)$, and $\mathcal{N}(-2,1)$.
2) Data set #2 1-D galaxy velocity data (also used in [11]).
3) Data set #3 (2-D synthetic data). A total of 1050 bivariate data points are drawn with equal probability from three normals: $\mathcal{N}(\begin{bmatrix}-7\\10\end{bmatrix}, \begin{bmatrix}5.5 & -4.5\\-4.5 & 5.5\end{bmatrix})$, $\mathcal{N}(\begin{bmatrix}0\\12\end{bmatrix}, \begin{bmatrix}8.5 & 6.5\\6.5 & 8.5\end{bmatrix})$, and $\mathcal{N}(\begin{bmatrix}12\\15\end{bmatrix}, \begin{bmatrix}14.4 & -4.5\\-4.5 & 3.6\end{bmatrix})$.
4) Data set #4 (2-D vowel data). This data set contains 640 time series of 12D LPC cepstrum coefficients taken from nine male speakers [37]. We use the first two dimension vectors for tests. In all the tests, we use[10] $\alpha = 1.25$ and the convergence tolerance

$$\varepsilon = \frac{\hat{f}_K(\mathbf{x}^{\kappa+1}) - \hat{f}_K(\mathbf{x}^{\kappa})}{\hat{f}_K(\mathbf{x}^{\kappa})} = 0.001.$$

Unless otherwise specified, the Gaussian kernel is adopted. The resulting mode locations found by the two algorithms are so close that the difference is negligible. We run the comparison with three arbitrarily selected start points on each data set. The experimental results are reported in Table I. The proposed algorithm is significantly more efficient than the standard MS. The evaluation results are promising: Speedup by a factor of about 2–5 can be achieved in these evaluations. We have also developed an accelerated MS tracker, which outperforms the conventional MS tracker [38].

The accelerated MS's performance with fewer convergence iterations has proven commensurate with its standard counterpart. In theory when the start point is extremely close to the local maximum, the rejection in the proposed accelerated MS procedure might happen frequently, resulting in a resource waste. In practice, these cases are rare. Moreover one can devise smarter step-size adjustment strategies to cope with this extreme case.

In Fig. 5, we have depicted the paths of the two methods and the contours of the density function as well as the learning curves for different runs and data sets. The standard MS algorithm takes many steps to reach the fixed point while the accelerated MS algorithm has a much better overall performance.

Data Clustering: In the second experiment, we compare the performance of three algorithms (stand MS, accelerated MS,

[10]It is possible to adapt the value of α for better performance.

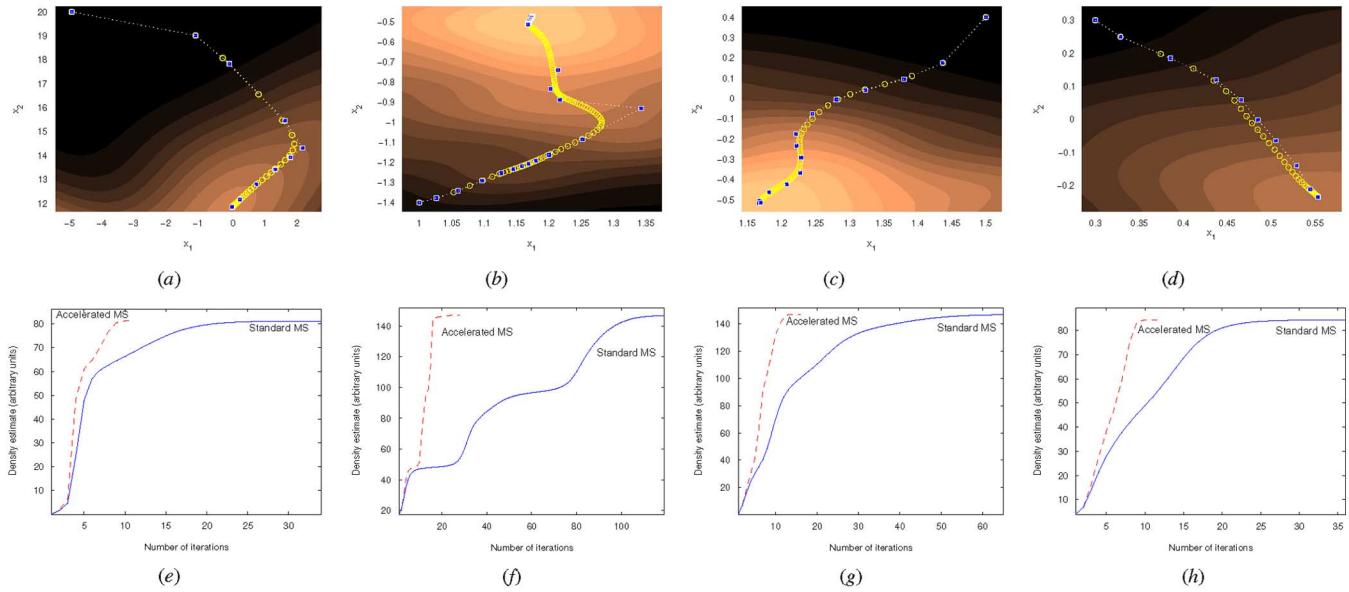

Fig. 5. Fast MS iteration versus the standard MS procedure. The four cases correspond to Run #1 for data set #3; Run #1, #2, and #3 for data set #4, respectively. (a)–(d) Zoomed mode seeking trajectories (squares: fast MS; circles: standard MS). (e)–(h) Learning rate curves of (a)–(d).

TABLE II
COMPARISON OF CPU TIME (IN SECONDS) FOR DATA CLUSTERING. THE STANDARD DEVIATION IS SHOWN IN THE BRACKET

data set	*CPU time (std.)*		
	mean shift	fast mean shift	quasi-Newton
data set #5	3.259 (0.008)	1.251 (0.003)	1.258 (0.004)
data set #6	71.12 (0.82)	24.62 (0.49)	24.40 (0.49)
data set #7	293.13 (6.91)	159.09 (4.28)	159.71 (3.51)
data set #8	113.77 (4.47)	54.13 (2.05)	52.56 (2.66)

and quasi-Newton [23], [39]) for clustering. The L-BFGS algorithm [40] is adopted for implementing the quasi-Newton algorithm. The clustering is achieved by running the MS mode seeking starting from each data point. Also, the Gaussian kernel is used and the step parameter α is set to 1.25.

The test data sets are as follows.

1) Data set #5 (2-D synthetic data). A total of 1000 data points are drawn from a Gaussian mixture model which is described in Section III-B (Fig. 3).
2) Data set #6 (Corel image features). The original data set contains 68 040 9D vectors (color moments) [37]. We subsample 4000 data points from it and only take the first two dimensions.
3) Data set #7 and #8 are two images with size 120×80 and 60×90, respectively. Therefore, there are totally 9600 and 5400 data points. Channels R and G are used for clustering.

In this experiment, all codes are written in C++ and run on a PC with Pentium-IV 3.4-GHz CPU, Linux 2.6 OS. We repeat all the tests 20 times and the average CPU time as well as the standard deviation is reported in Table II.

In terms of efficiency, the proposed fast MS algorithm is consistently better (around 2–3 times faster) than standard MS and similar to the quasi-Newton algorithm in these tests. However, we observe that the fast MS is better than quasi-Newton in the accuracy of clustering.

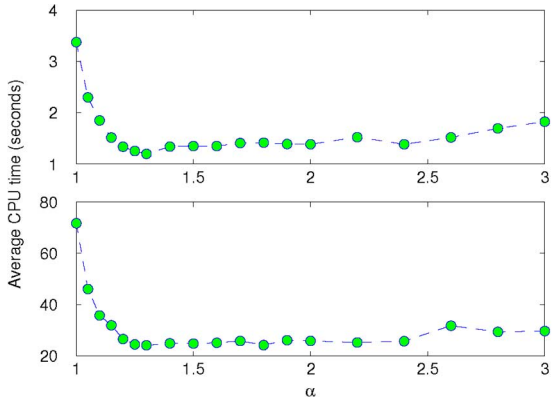

Fig. 6. CPU time at various values of α, test on data set (top) #5 and (bottom) #6. In both cases, $\alpha = 1.3$ achieves fastest computation.

It is well known that the quasi-Newton gradient algorithm can *over-shoot*[11] when the search step is too large. This is potentially acceptable in generic optimization because the purpose is often to find any local optimum—not necessarily the nearest one. In contrast, in clustering, the purpose is to drive each data point to its nearest mode. It is for this reason that MS is preferred for clustering such as image segmentation.

Although theoretically fast MS could also over-shoot when α is large, in both mode-seeking and data-clustering experiments, fast MS with $\alpha = 1.25$ obtains results identical to its standard counterpart. We deliberately set α very large to test the accuracy. In Fig. 7, we show the clustering results on data set #5. All three algorithms find the same modes but quasi-Newton has many initial points converging to the wrong modes. Hence, while quasi-Newton might be a good choice for some optimization tasks like object tracking [39], it is less useful for clustering.

A question on the fast MS is how to determine the precise α value. To examine this, we test different values of α on two

[11]By over-shooting, we mean the algorithm converges to a local optimum which is not the nearest one to the initial position.

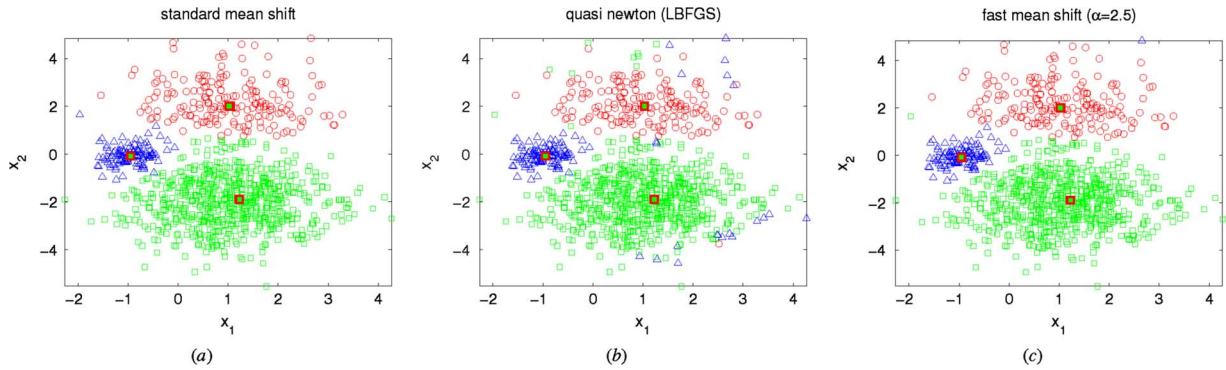

Fig. 7. Clustering results of (a) stand MS and fast MS with $\alpha = 1.25$; (b) quasi-Newton; (c) fast MS with $\alpha = 2.5$. 32 out of 1000 points are wrongly clustered for the quasi-Newton method; three for the fast MS with $\alpha = 2.5$ Cluster centers are marked with thick squares.

datasets. Fig. 6 shows how α can change the performance of the algorithm. Empirically, we see that in a certain α range, the algorithm is not sensitive to change in α. However, in Fig. 7, we see that the larger the α, the more likely it is to overshoot. For this reason, we choose $\alpha = 1.25$ in this work.[12]

V. Annealed MS-Based Visual Localization and Tracking

We have proposed a framework for fast global kernel density mode seeking. In this section, we apply the annealed MS to visual localization and tracking. In all the localization and tracking experiments we use RGB color histograms, consisting of $16 \times 16 \times 16$ bins. The tracking framework presented in [2] is adopted, but we use an annealing procedure for global mode seeking. We first briefly review the MS tracker [2] in Section V-A, followed by the experiments on the proposed visual localization (Section V-B) and tracking (Section V-C).

A. Implementation

In the experiments, the object is represented by a square region which is cropped and *normalized* into a unit circle for the convenience of derivation [2]. By denoting $\hat{\mathbf{q}}$ as the color histogram of the target model, and $\hat{\mathbf{p}}(\mathbf{x})$ as the target candidate color histogram with the center at $\mathbf{x}$, the (dis-)similarity function between $\hat{\mathbf{q}}$ and $\hat{\mathbf{p}}(\mathbf{x})$ is written $\hat{\rho}(\mathbf{x}) = \rho(\hat{\mathbf{q}}, \hat{\mathbf{p}}(\mathbf{x}))$. Here, $\rho(\cdot, \cdot)$ can, for example, be the Bhattacharyya divergence [2], [41], the Kullback-Leibler (KL) distance [4], [42], or the Matusita metric [5] as a similarity measurement.[13] Let $\{\mathbf{y}_i\}_{i=1}^n$ be the pixel positions in the region with the center at $\mathbf{x}$. In order to make the cost function smooth—otherwise gradient-based MS optimization cannot be applied—a kernel with profile $k(\cdot)$ is employed to assign smaller weights to those pixels farther from the center, considering the fact that the peripheral pixels are less reliable. We then build an m-bin color histogram for the target candidate located at $\mathbf{x}$, $\hat{\mathbf{p}}(\mathbf{x}) = \{\hat{p}_u(\mathbf{x})\}_{u=1}^m$, where

$$\hat{p}_u(\mathbf{x}) = \frac{1}{c} \sum_{i=1}^{n} k(\|\mathbf{y}_i\|^2) \delta((b(\mathbf{y}_i) - u)). \tag{9}$$

[12]In [26], α is set to 1.1 for over-relaxed EM.

[13]These three density metrics have been reported in tracking literature. Other metrics might also be used.

The constant c guarantees $\hat{\mathbf{p}}(\mathbf{x})$ a normalized density. $\delta(\cdot)$ is the Kronecker function and function $b(\cdot)$ maps a normalized pixel $\mathbf{y}_i$ to the histogram bin associated with the color of $\mathbf{y}_i$. The same strategy is used to obtain the target model $\hat{\mathbf{q}}$.

Given an initial position $\mathbf{x}^\star$, the problem of localization/tracking is to estimate a best displacement $\Delta\mathbf{x}$ such that the measurement $\hat{\mathbf{p}}(\mathbf{x}^\star + \Delta\mathbf{x})$ at the new location best matches the target $\hat{\mathbf{q}}$, i.e.,

$$\Delta\mathbf{x}^\star = \arg\min_{\Delta\mathbf{x}} \rho(\hat{\mathbf{q}}, \hat{\mathbf{p}}(\mathbf{x}^\star + \Delta\mathbf{x}))$$

where $\rho(\cdot, \cdot)$ is large when the two distributions are similar (e.g., the KL or Matusita distance); otherwise (e.g., Bhattacharyya distance)

$$\Delta\mathbf{x}^\star = \arg\max_{\Delta\mathbf{x}} \rho(\hat{\mathbf{q}}, \hat{\mathbf{p}}(\mathbf{x}^\star + \Delta\mathbf{x})). \tag{10}$$

By Taylor expanding $\rho(\cdot, \cdot)$ at the start position $\mathbf{x}^\star$ and keeping only the linear item (first-order Taylor approximation), the optimization problem (10) can be resolved by an efficient MS procedure. See [2] and [5] for more details.

B. Visual Localization

Standard MS is used for tracking motions with small displacements due to its lack of global mode seeking capability, and is not used for localization. Armed with annealed MS, it is possible to locate a target no matter from which initial position the MS localizer starts, given the target template.

In our experiments, the annealed MS localizer starts at arbitrarily selected positions. All successfully locate the target. A total of six runs for each example are marked in Fig. 8. Four objects are located successfully in different environments. For the first example, the bandwidths are $\{60, 40, 20, 10\}$. We plot the cost function values for this example in Fig. 9 to illustrate how annealed MS works in this case. The influence of the most significant peak is introduced gradually, which guides search towards the global mode. One can see that even at $h_1 = 20$, there are plenty of local modes which can easily make the search stop prematurely. At $h_0 = 10$, there are three major modes corresponding to the three faces in the figure. Note that MS does not converge to the exact modes in Fig. 9 due to the Taylor approximation [2]. However, it converges to a position close to the true

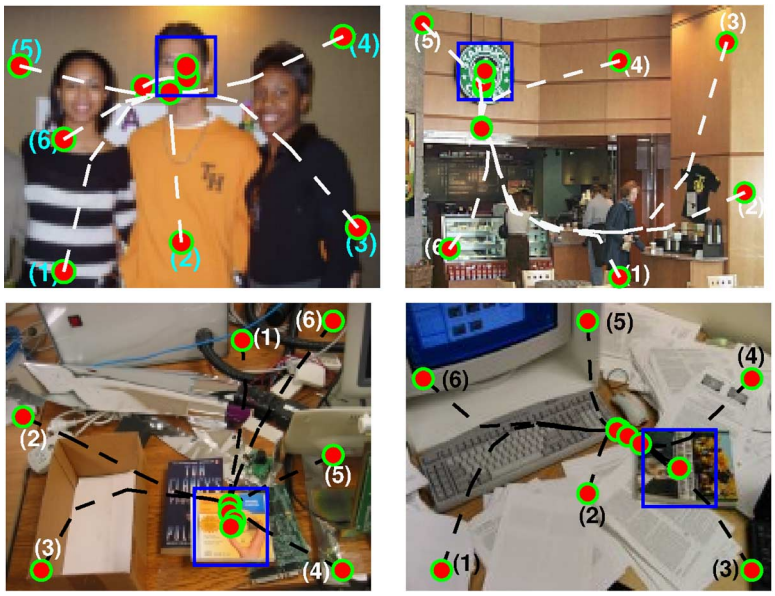

Fig. 8. We locate (left top) a specified human face beside two spurious faces, (right top) the STARBUCKS logo, (left bottom) a CD, and (right bottom) a book cover within cluttered backgrounds. The annealed MS is started at arbitrarily selected positions. Dashed lines indicate the MS searching trajectories for each run. Dots indicate the start and convergence positions of MS foreach bandwidth. See the videos $\mathrm{localizer}\{1,2,3,4\}.\mathrm{avi}$ for an intuitive demonstration of the annealing convergence processes.

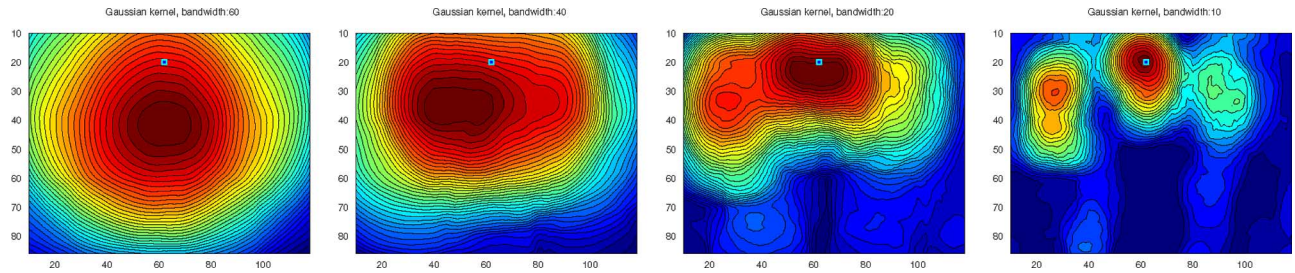

Fig. 9. Cost functions (corresponding to the first example in Fig. 8) at different bandwidths: 60, 40, 20, and 10 are plotted as contours of 2-D translations. The true mode is marked with a square.

mode. For localization and tracking, this accuracy loss is negligible. The size of the image in this example is 128×96 and the target is 10×10. To locate the target, an exhaustive search needs 11 543 evaluations of the distance between the candidate and the target. Comparatively an average 35.5 iterations is needed for annealed MS.

The other three examples begin at $h_4 = 80$ and a five-step annealing guarantees a global mode in these cases. For the CD and book cover localization, we take the template models from other images undergoing large geometric and slight illumination changes. The success shows that the color histogram is a relatively robust feature. It is straightforward to include other features, e.g., a histogram of intensity gradient orientation, to make the localizer more robust. Without the annealing procedure, most runs reach local maxima—only when the initial positions are located in the small area close to the global mode can standard MS find the target. When no prior knowledge is available about the global maximum we are seeking, it is always beneficial to employ a relatively broad bandwidth MS procedure, which can provide a coarse location of the global mode. In the experiments here, although we do not carefully design the annealing schedule, global modes are always found.

C. Visual Tracking

Tracking algorithms typically have several drawbacks. 1) They work well only when the displacements between consecutive frames are relatively small. 2) Usually, they cannot self-start. 3) They are not robust to occlusions and are unable to recover from momentary tracking failures. Standard MS trackers are no exception. Annealed MS alleviates these weaknesses by incorporating an efficient bottom-up localization functionality. To introduce a detector/localizer into visual tracking is generally helpful. The detection/localization process is robust to momentary tracking failures because it does not rely on any temporal information. For example, Okuma *et al.* combine an AdaBoost detector into mixture particle filters to track a varying number of nonrigid objects [43]. AdaBoost helps to generate proposal distributions for particle filters. Better performances have been observed. The single scale MS approach assumes that the initialization point for the tracker (which is typically the mode from the previous frame) falls within the basin of attraction of the desired mode. The limits of the basin of attraction are, however, determined by the bandwidth of the kernel used, which is in turn determined by

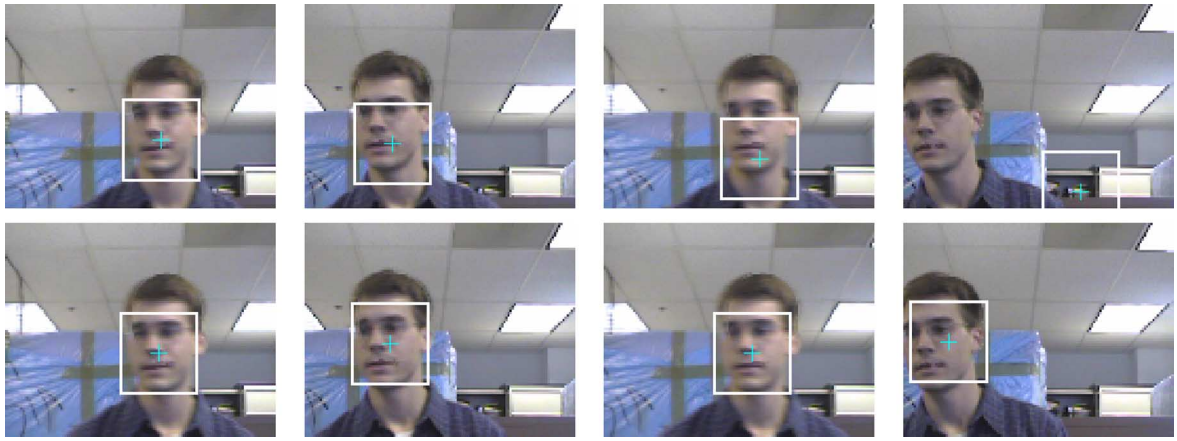

Fig. 10. Face-tracking sequence with (top) standard MS and (bottom) annealed MS. Frames #5, #14, #22, and #25 are shown. The object is *accurately* detected and tracked by annealed MS despite large displacements. In contrast, MS is more likely to become trapped into local modes and gives inaccurate results (# 5, #14, and #22) or even fails completely (#25). See the video *facetracker.avi* for details.

the scale of the object being tracked. There is little reason, however, to assume that there is any relationship between the scale of the object being tracked and the amount that it might move between frames.

Face Tracking Example: The tracked target moves fast, leading to large displacements between consecutive frames. An annealing schedule of $\{60, 30, 18\}$ is used by annealed MS. The annealed MS tracker is automatically started by a localization process, while the MS tracker is manually started. As in MS tracking, annealed MS also starts at the position of the previous frame. Unless otherwise noted, in all the tracking experiments, the convergence tolerance is the ℓ_2-norm distance between two iterations, with $\varepsilon = 0.2$ pixels. Fig. 10 summarizes the tracking results. The annealed MS tracker is more robust and accurate than the standard MS tracker. When the displacement is large, the standard MS tracker is easily trapped in spurious modes.

Implementation Issues: MS might get stuck at false modes caused by the discrete nature of the color values of pixels. Wang and Suter observe this phenomenon in grey image histogram clustering [44]. Their analysis also applies to color image histograms. We avoid this problem by imposing a ceiling on the MS step $\lceil \mathbf{m}_G(\mathbf{x}) \rceil$ [see (5)]. This modification increases the size of the shift steps, hence leading to quicker convergence. The drawback is that it might lose accuracy. We use the original step by (5) at the last bandwidth h_0. Because we are only interested in the last convergence position, accuracy is retained. Both in localization and tracking, it has been observed that this simple treatment results in satisfactory convergence without accuracy loss. We compare the number of convergence iterations per frame for the face tracking video in Fig. 11.[14] One can see that in this example their convergence speeds are similar (note that the fast MS algorithm is not implemented for both trackers). In many frames, annealed MS is even faster. The is because MS at the first few bandwidths $(h_{M...1})$ can move close to the mode quickly with the above implementation. A recent paper [45] suggests that, besides avoiding local optima, simulated annealing also speeds up convergence to the optimum. Our observation coincides with this finding. However, larger bandwidths of annealed MS mean that slightly more computation might be needed to build the histograms. We also have implemented the proposed accelerated MS algorithm (Section IV) into tracking, and a considerable speedup has been achieved [38].

[14]Only frames #1 . . . , 22 are compared because from # 23 on, the MS tracker fails. For annealed MS, we count the sum of iterations at each bandwidth.

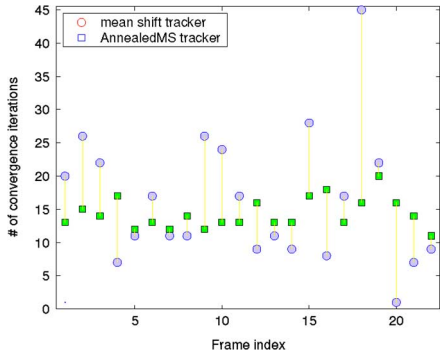

Fig. 11. Comparison of the number of iterations per frame: MS (marked with circles) versus annealed MS (marked with squares) for the face tracking sequence.

The second issue is the design of the annealing schedule. As in simulated annealing, the slower the annealing schedule, the more likely the algorithm is to find an optimal solution. However, a slow annealing procedure incurs heavy computational costs. Therefore, choosing a proper cooling schedule is of considerable value. At this stage, we determine the annealing schedule empirically.

Walker Tracking Example: We track a walker in an office environment. In this sequence, the tracked person disappears for several frames. The annealed MS tracker automatically initializes when the tracked walker comes back to the scene. In contrast, the MS tracker fails to recover. An annealing schedule of $\{70, 30, 13\}$ is used by annealed MS in this example. See Fig. 12 for details.

Basketball Tracking Example: This example again shows the annealed MS tracker's ability to recover from temporal failures. The original sequence is down sampled by a factor of 2 to make the target's displacements larger. The MS tracker fails as early

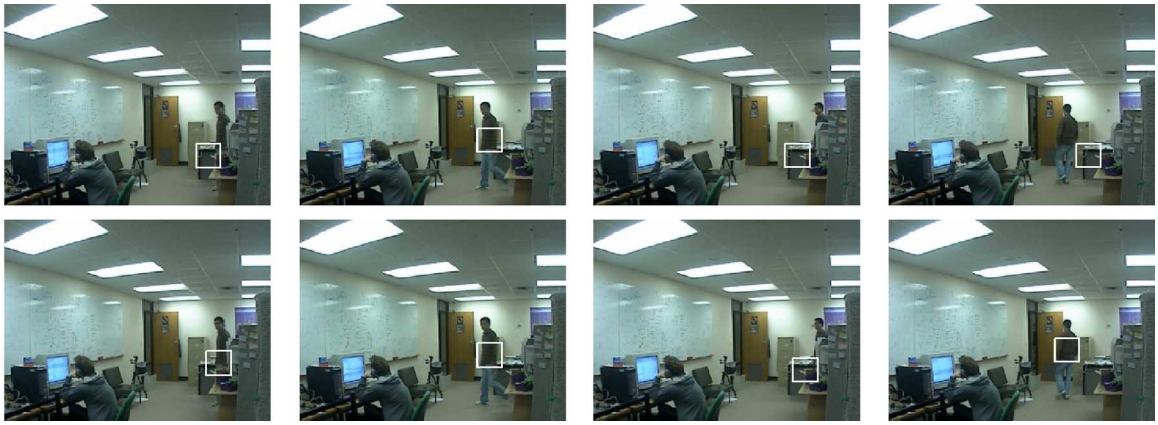

Fig. 12. Walker tracking results with (top) MS and (bottom) annealed MS. Frames #17, #19, #58, and #62 are shown. The annealed MS tracker successfully recovers from complete occlusion in #$9 \sim 16$ and #$40 \sim 58$. MS cannot recover from the second occlusion. See *office.avi* for details.

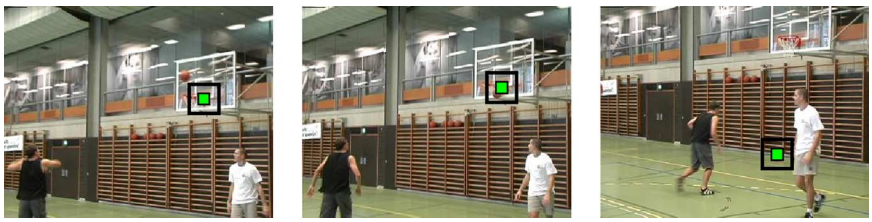

Fig. 13. Basketball tracking results with annealed MS. Frames #18, #20, and #29 are shown. See *basketball.avi* for details.

Fig. 14. Weetbix box tracking results with annealed MS. Frames #2, #9, and #17 are shown. See *weetbixbox.avi* for details.

as at #6. Therefore, we only show the tracking results of annealed MS in Fig. 13. annealed MS tracks across bandwidths $\{30, 15, 8\}$ and works successfully. At #18, annealed MS loses the target due to illumination changes. However, it recovers immediately at #19. It drifts slightly because of the basket's occlusions at #20 and recovers at the next frame. Again, we observe annealed MS tracking is efficient: An average of only 8.1 iterations per frame is needed.

Weetbix Box Tracking Example: We track a part of a weetbix box, which is recorded by a very unstable camera. Annealed MS again shows its robustness over the MS tracker. Here, the annealing schedule is $\{100, 30, 20\}$. The conventional MS loses the target very early because of the drastic camera motion, while the annealed MS achieves significantly better performance. The tracking results are shown in Fig. 14.

VI. Conclusion and Discussion

When the displacement between neighboring video frames is large compared with the scale of the adopted kernel, MS tracking is susceptible to failure. In this paper, we addressed this shortcoming with the introduction of a new global mode seeking MS, termed annealed MS. Improvements over standard MS were obtained when the density has multiple peaked modes. Promising results were obtained in both tracking and localization applications, even with the most elementary of annealing schedules.

An adaptive over-relaxed MS was also proposed to accelerate convergence. Compared with the standard MS algorithm, the number of iterations to convergence was almost always significantly decreased. The method provides an additional speedup to existing techniques such as LSH [1] and fast Gaussian transform [7], [31].

Visual localization and tracking are key problems in computer vision. Considerable efforts have been expended in this area due to their utility in applications such as visual surveillance, intelligent vision-based human computer interaction and smart vehicle driving systems. Future work will explore the effects of annealing schedule design on the localization and tracking performances, and the use of other discriminative features (rather than simple color histograms) for better localization and tracking performance. Consideration will also be given to exploring the application of the proposed method to other computer vision problems.

APPENDIX

We prove Theorem 4.3 in this Appendix. The MS procedure can be considered as a fixed-point iteration which has a general form $\mathbf{x} = S(\mathbf{x}) \cdot S(\cdot)$ is the mapping function, which is in the format of (6) for MS. We confine ourself to the univariate case here, i.e., $\mathbf{x} \in \mathbb{R}$, because of the following:

— it is straightforward to generalize the following analysis to the multivariate case;

— under the isotropic bandwidth assumption ($\mathbf{H} = h^2\mathbf{I}$), the multivariate case can be decomposed into multiple independent univariate cases.

Let the local maximum be $\mathbf{x}^*$. $\mathbf{x}^*$ is the fixed point: $\mathbf{x}^* = S(\mathbf{x}^*)$. We have $\mathbf{x}^{(\kappa+1)} - \mathbf{x}^* \approx \nabla S(\mathbf{x}^*)(\mathbf{x}^{(\kappa)} - \mathbf{x}^*)$ when $\mathbf{x}^{(\kappa+1)}$ is close to $\mathbf{x}^*$. Thus

$$\left\|\mathbf{x}^{(\kappa+1)} - \mathbf{x}^*\right\| \leq \|\nabla S(\mathbf{x}^*)\| \cdot \left\|\mathbf{x}^{(\kappa)} - \mathbf{x}^*\right\|.$$

That is to say, MS is a first-order (linear) algorithm provided $\nabla S(\mathbf{x}^*) \neq 0$.[15] Therefore, generally, MS's convergence is slow.

Now let us analyse MS's asymptotic convergence property. Since

$$S(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)} \tag{11}$$

and using the equation $\mathbf{x}^* = S(\mathbf{x}^*)$, we have

$$\nabla S(\mathbf{x}^*) = -\frac{2}{h^2} \frac{\sum_{i=1}^{n} (\mathbf{x}^* - \mathbf{x}_i)^2 g'\left(\left\|\frac{\mathbf{x}^*-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{\mathbf{x}^*-\mathbf{x}_i}{h}\right\|^2\right)}. \tag{12}$$

Because $\mathbf{x}^*$ is the maximum, we have $\nabla^2 \hat{f}_K(\mathbf{x}^*) < 0$ (for multivariate $\mathbf{x}$, the Hessian matrix is negative definite or negative semi-definite). After some manipulation, the following relation holds:

$$-\frac{2}{h^2}\sum_{i=1}^{n}(\mathbf{x}^* - \mathbf{x}_i)^2 g'\left(\left\|\frac{\mathbf{x}^* - \mathbf{x}_i}{h}\right\|^2\right) < \sum_{i=1}^{n} g\left(\left\|\frac{\mathbf{x}^* - \mathbf{x}_i}{h}\right\|^2\right). \tag{13}$$

Define $r(x) = -g'(x)$, then $g(\cdot)$ is the shadow of the profile $r(\cdot)$. We know that $r(\cdot)$ satisfies all the requirements to be a profile if $r(x)$ is not always $0, \forall x \geq 0$ [9]. We now rewrite (13)

$$\frac{2}{h^2}\sum_{i=1}^{n}(\mathbf{x}^* - \mathbf{x}_i)^2 r\left(\left\|\frac{\mathbf{x}^* - \mathbf{x}_i}{h}\right\|^2\right) < \sum_{i=1}^{n} g\left(\left\|\frac{\mathbf{x}^* - \mathbf{x}_i}{h}\right\|^2\right). \tag{14}$$

The l.h.s. of (14) is a weighted density estimate with non-negative weights, and it must be non-negative because $\forall x \geq 0, r(x) \geq 0$. We have

$$0 \leq \nabla S(\mathbf{x}^*) < 1. \tag{15}$$

[15] It can get higher order convergence when $\nabla S(\mathbf{x}^*) = 0$; [11] also shows this conclusion from a different view.

With the fixed-point iteration theorem, it is easy to see that there is an interval $\mathcal{I}_\delta = [\mathbf{x}^* - \delta, \mathbf{x}^* + \delta], \delta > 0$, such that the iteration $\mathbf{x}^{(\kappa+1)} = S(\mathbf{x}^{(\kappa)})$ monotonically converges to the local maximum $\mathbf{x}^*$ for every $\mathbf{x}^{(0)} \in \mathcal{I}_\delta$. In this case, $\mathbf{x}^*$ is an attractive fixed point. Otherwise, if $\|\nabla S(\mathbf{x}^*)\| > 1$, the iteration will not converge to $\mathbf{x}^*$. We say that $\mathbf{x}^*$ is a repelling fixed point and the iteration exhibits local divergence.

Furthermore, if $\nabla S(\mathbf{x}^*) \neq 0$, then the convergence is linear with the value $\nabla S(\mathbf{x}^*)$. In other words, the smaller $\nabla S(\mathbf{x}^*)$ is, the faster the iteration converges. When the bandwidth $h \to +\infty, \nabla S(\mathbf{x}^*) \to 0$. In this case, the convergence is fast (superlinear).

Alternatively, if $\nabla S(\mathbf{x}^*) = 0$, then the convergence is of order 2 (quadratic). When the MS procedure employs a piecewise constant profile $g(\cdot), r(\cdot) = -g'(\cdot) = 0$ and consequently $\nabla S(\mathbf{x}^*) = 0$. In this case, the convergence rate is of order 2. This result coincides with Theorem 2 in [11].

ACKNOWLEDGMENT

The authors would like to thank Dr. S. Birchfield, Dr. K. Nummiaro for the test videos (the face video and the basketball video), J. Tebneff, and R. Hill for helping collect test data.

REFERENCES

[1] B. Georgescu, I. Shimshoni, and P. Meer, "Mean shift based clustering in high dimensions: A texture classification example," in *Proc. IEEE Int. Conf. Computer Vision*, Nice, France, 2003, vol. 2, pp. 456–463.

[2] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.

[3] R. Collins, "Mean-shift blob tracking through scale space," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, Madison, WI, Jun. 2003, vol. 2, pp. 234–240.

[4] A. Elgammal, R. Duraiswami, and L. S. Davis, "Probabilistic tracking in joint feature-spatial spaces," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, Madison, WI, 2003, vol. 1, pp. 781–788.

[5] G. D. Hager, M. Dewan, and C. V. Stewart, "Multiple kernel tracking with SSD," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, Washington, DC, 2004, vol. 1, pp. 790–797.

[6] H. Wang and D. Suter, "MDPE: A very robust estimator for model fitting and range image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 139–166, 2004.

[7] C. Yang, R. Duraiswami, and L. Davis, "Efficient spatial-feature tracking via the mean-shift and a new similarity measure," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, San Diego, CA, 2005, vol. 1, pp. 176–183.

[8] K. Fukunaga and L. D. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Trans. Inf. Theory*, vol. IT-21, no. 1, pp. 32–40, Jan. 1975.

[9] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[10] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[11] M. Fashing and C. Tomasi, "Mean shift is a bound optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 471–474, Mar. 2005.

[12] S. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-based modeling in particle filters," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1491–1506, Nov. 2003.

[13] E. Arnaud, E. Memin, and B. Cernuschi-Frias, "Conditional filters for image sequence-based tracking—Application to point tracking," *IEEE Trans. Image Process.*, vol. 14, no. 1, pp. 63–79, Jan. 2005.

[14] Z. Zivkovic and B. Krose, "An EM-like algorithm for color-histogram-based object tracking," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2004, vol. 1, pp. 798–803.

[15] Z. Fan, Y. Wu, and M. Yang, "Multiple collaborative kernel tracking," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, San Diego, CA, 2005, vol. 2, pp. 502–509.

[16] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proc. Eur. Conf. Computer Vision*, Copenhagen, Denmark, 2002, vol. 2350, pp. 661–675, Lecture Notes Comput. Sci..
[17] P. van Laarhoven and E. Aarts, *Simulated Annealing: Theory and Applications*. New York: Springer Verlag, 1987.
[18] R. M. Neal, "Annealed importance sampling," *Statist. Comput.*, vol. 11, no. 2, pp. 125–139, Apr. 2001.
[19] J. Deutscher and I. Reid, "Articulated body motion capture by stochastic search," *Int. J. Comput. Vis.*, vol. 61, no. 2, pp. 185–205, 2005.
[20] W. Härdle, M. Müller, S. Sperlich, and A. Werwatz, *Nonparametric and Semiparametric Models*, ser. Springer Ser. Statist. New York: Springer, 2004.
[21] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, Sep. 1975.
[22] K. Zhang, M. Tang, and J. Kwok, "Applying neighborhood consistency for fast clustering and kernel density estimation," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, San Diego, CA, Jun. 2005, vol. 2, pp. 1001–1007.
[23] C. Yang, R. Duraiswami, D. DeMenthon, and L. Davis, "Mean-shift analysis using quasi-newton methods," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, 2003, vol. 3, pp. 447–450.
[24] L. Xu, "On convergence properties of the EM algorithm for Gaussian mixtures," *Neural Comput.*, vol. 8, no. 1, pp. 129–151, Jan. 1996.
[25] L. E. Ortiz and L. P. Kaelbling, "Accelerating EM: An empirical study," in *Proc. Uncertainty in Artificial Intell.*, Stockholm, Sweden, 1999, pp. 512–521.
[26] R. Salakhutdinov and S. Roweis, "Adaptive overrelaxed bound optimization methods," in *Proc. Int. Conf. Machine Learning*, Washington, DC, 2003, pp. 664–671.
[27] C. Shen, M. J. Brooks, and A. van den Hengel, "Fast global kernel density mode seeking with application to localisation and tracking," in *Proc. IEEE Int. Conf. Computer Vision.*, Beijing, China, Oct. 2005, vol. 2, pp. 1516–1523.
[28] D. Comaniciu, "An algorithm for data-driven bandwidth selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 281–288, Feb. 2003.
[29] M. C. Jones, J. S. Marron, and S. J. Sheather, "A brief survey of bandwidth selection for density estimation," *J. Amer. Statist. Assoc.*, vol. 91, no. 433, pp. 401–407, Mar. 1996.
[30] P. Hall, M. C. Minnotte, and C. Zhang, "Bump hunting with non-Gaussian kernels," *Ann. Statist.*, vol. 32, no. 5, pp. 2124–2141, 2004.
[31] C. Yang, R. Duraiswami, N. Gumerov, and L. Davis, "Improved fast Gauss transform and efficient kernel density estimation," in *Proc. IEEE Int. Conf. Comp. Vis.*, Nice, France, Oct. 2003, vol. 1, pp. 464–471.
[32] P. Chaudhuri and J. S. Marron, "Scale space view of curve estimation," *Ann. Statist.*, vol. 28, no. 2, pp. 408–428, 2000.
[33] T. Lindeberg, *Scale-Space Theory in Comp. Vis.*. Norwell, MA: Kluwer, 1994.
[34] A. Blake and A. Zisserman, *Visual Reconstruction*. Cambridge, MA: MIT Press, 1987.
[35] D. Comaniciu, "Nonparametric information fusion for motion estimation," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, Madison, WI, Jun. 2003, vol. 1, pp. 59–66.
[36] M. Carreira-Perpiñán and C. Williams, "On the number of modes of a Gaussian mixture," Tech. Rep. EDI-INF-RR-0159, School Informatics, Univ. Edinburgh, Edinburgh, U.K., Feb. 2003 [Online]. Available: http://www.inf.ed.ac.uk/publications/report/0159.html
[37] S. Hettich and S. D. Bay, "The UCI KDD archive," Tech. Rep., Dept. Inf. Comput. Sci., University of California, Irvine, CA, 1999 [Online]. Available: http://kdd.ics.uci.edu
[38] C. Shen and M. J. Brooks, "Adaptive over-relaxed mean shift," presented at the 8th Int. Symp. Signal Process. Applications, Sydney, Australia, Aug. 2005.
[39] T. Liu and H. Chen, "Real-time tracking using trust-region methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 397–402, Mar. 2004.
[40] D. C. Liu and J. Nocedal, "On the limited memory method for large scale optimization," *Math. Programm. B*, vol. 45, no. 3, pp. 503–528, 1989.
[41] T. Kailath, "The divergence and Bhattacharyya distance measures in signal selection," *IEEE Trans. Commun. Technol.*, vol. 15, no. 1, pp. 52–60, Feb. 1967.
[42] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.
[43] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little, and D. G. Lowe, "A boosted particle filter: Multitarget detection and tracking," in *Proc. Eur. Conf. Computer Vision*, Prague, Czech Republic, 2004, vol. 1, pp. 28–39.
[44] H. Wang and D. Suter, "False-peaks-avoiding mean shift method for unsupervised peak-valley sliding image segmentation," in *Proc. 8th Digital Image Computing: Techniques and Applications*, Sydney, Australia, 2003, pp. 581–590.
[45] A. Kalai and S. Vempala, "Simulated annealing for convex optimization," *Math. Oper. Res.* [Online]. Available: http://ttic.uchicago.edu/~kalai. In press.

Chunhua Shen received the B.Sc. and M.Sc. degrees from Nanjing University, Nanjing, China, in 1999 and 2002, respectively, and the Ph.D. degree from the School of Computer Science, University of Adelaide, Australia, in 2005.

He is currently a Researcher with the Computer Vision Program, National ICT Australia, Ltd., Canberra, and an Adjunct Research Fellow at the Australian National University. His main research interests include statistical pattern analysis and its application in computer vision.

Michael J. Brooks (M'91) received the Ph.D. degree in computer science from the University of Essex, Essex, U.K., in 1983.

He joined Flinders University in 1980 and the University of Adelaide, Australia, in 1991, where he holds the Chair in Artificial Intelligence and is Head of the School of Computer Science. Since mid-2005, he has been a Nonexecutive Director of National ICT Australia, Ltd., Canberra. He has interests that include tracking, video surveillance, parameter estimation, and self-calibration. His patented surveillance work has seen worldwide application at airports, major facilities, and iconic structures around the world.

Anton van den Hengel (M'04) received the Ph.D. degree in computer science from the University of Adelaide, Australia, in 2000.

He was appointed Lecturer in 1996 and Senior Lecturer in 2003 in the School of Computer Science, University of Adelaide. He is the inaugural Director of the Australian Centre for Visual Technologies and Deputy Chair of the South Australian SIGGRAPH Chapter. His research interests include video surveillance over large networks cameras, parameter estimation for computer visions problems, and high-level scene modeling from point clouds.