GATE: A Novel Robust Object Tracking Method Using the Particle Filtering and Level Set Method

Cheng Luo[1] Xiongcai Cai[2] Jian Zhang[1]

[1] University of New South Wales, Australia
{xcai}@cse.unsw.edu.au
[2] University of ABC, Australia
luo.cheng, xiongcai.cai, jian.zhang@nicta.com.au

Technical Report
UNSW-CSE-TR-0818
August 2008

School of Computer Science and Engineering
The University of New South Wales
Sydney 2052, Australia

Abstract

This technical report presents a novel algorithm for robust object tracking based on the particle filtering method employed in recursive Bayesian estimation and image segmentation and optimisation techniques employed in active contour models and level set methods. The proposed *G*eometric *A*ctive contour-based *T*racking *E*stimation, namely *GATE*, enables particle filters to track object of interest in complex environments using merely a simple feature. *GATE* creates a spatial prior in the state space using shape information of the tracked object. The created spatial prior is then used to filter particles in the state space in order to reshape and refine the observation distribution of the particle filtering. This improves the performance of the likelihood model in the particle filtering, so the significantly overall improvement of the particle filtering. The promising performance of our method on real sequences are demonstrated.

1 Introduction

The problem of tracking moving objects in complex environments has been an active topic of substantial research in many settings, including sports [4, 7], surveillance [1], etc. Although there are many techniques for object tracking, it is well known that each algorithm has its inherent drawbacks. Over the last decade, the particle filter [1, 21, 3], also known as sequential Monte Carlo filter, is a preferable and powerful technique for object tracking [21]. In this paper, we propose a novel tracking scheme which utilises the particel filter and the level set-based active contours to tackle object tracking in complex environments.

As a solution to recursive Bayesian filtering problems, particle filtering methods still heavily depend on some discriminable features even though these methods demonstrate the advantages of simplicity, flexibility and systematic treatment of nonlinear motion and non-Gaussian noise. Meanwhile, as a workable algorithm, particle filtering methods have to adopt resampling techniques to tackle the degeneracy problem [6]. Perez et al. [1] developed a multi-color observation model based on Hue-Saturation-Value(HSV) color histogram. The global property of color histogram enables particle filters, to some extent, to be more robust and insensitive to the changes of illumination and object appearance. They simply resample the corrected weights of particles based on the order of their weights and the total number of particles. Nevertheless, this simple resampling method cannot prevent outlier observations impoverishing the particle representation of the posterior when the adopted features are not discriminable. Vermaask et al. [20] extended the resampling method for particle filters by introducing K-means algorithm to cluster particles. The K-mean algorithm is merely utilised to decide the splitting and merging of tracked objects. Okuma et al. [7] extended the proposal distribution in particle filters by combining a AdaBoost detector for hockey players in a simple background with the multi-color observation model via incorporating current measurements, and maintained the K-mean algorithm for the resampling proposed by Vermaask et al. Brasnett et al. [10] extended the observation model for particle filters by fusing multiple features. They assumed that at least one feature among these multiple features should not be absent and indiscriminable. Usually, their assumption is not guaranteed in complex environments where the tracked object is difficult to distinguish with the background. Rathi et al. [3] used Chan-Vese algorithm to generate measurement in the observation model for each particle by segmenting regions surrounding each particle. Since the performance of segmentation strongly depends on to what extent the assumption of homogenous image data is satisfied, the desired performance on complex environments, such as camouflage or environments described in Section 3.1, is no guarantee. Although the above approaches work well in specific tasks, object tracking in complex environments with heavy clutters in the background, low resolution image sequences and non-stationary camera is still a challenge problem.

Recently, level set-based active contours were developed for image segmentation and object extraction. The basic philosophy of active contour is to attract a curve to the boundary of the object of interest starting with an initial curve C, under some constraints from the image u_0. Kass et al. [22] first presented deformable contour models or Snakes for detection and localisation of object boundaries. Caselles et al. [9] formulated Snakes in Geodesic space and implicitly represented it using the level set function for automatic detection of objects with topology changes, cusps and corners. Chan and Vese [8] extended the edge based energy functional proposed in [9] to the region-driven energy functional, which is based on a Mumford-Shah functional over the length of the contour and the sum of the fitting error over each component of the gray scale image. Rousson and Paragios [5] extended Chan-Vese algorithm by imposing shape prior information on the energy functional to be minimised. Their algorithm can perform better in presence of occlusion and distortion because of the positive influence of shape prior information during the optimisation of the total energy functional. Meanwhile, some previous work on object tracking using level set methods can be found in [3, 4, 20]. The above methods all work on image space. For general data space including sparse and non-sparse data space, Cai and Sowmya [2] first developed the level learning set to tackle pattern classification problems by extending level set-based active contour model to machine learning.

In this report, we tackle object tracking in complex environments, including heavy clutters in the background, low resolution image sequences and non-stationary camera, where the existence of multiple features or well object detection is not guaranteed. Our contribution is the development of the *GATE* method that significantly improves the posterior distribution of particle filters by reshaping and refining the likelihood distribution, and improves the overall performance of particle filters. Specifically, *GATE* utilises the level set-based active contour method to build a decision boundary in the state space using shape information and pose invariance of the tracked object, such that the region inside the decision boundary has high probability to be the true object. After reweighting the particles due to the constructed decision boundary, particles which locate outside the decision boundary will be rejected during the resampling.

The paper is organized as follows: In Section 2, the particle filter and active contour methods will be discussed. In Section 3 the Geometirc active contour estimator will be described in detail and experimental results will be provided in Section 4. Concluding remarks are discussed in Section 5.

2 Preliminaries

2.1 Sequential Monte Carlo Filter

Let$\{\mathbf{x}_{0:t}^i, \omega_t^i\}_{i=1}^{N_s}$ denote a *Random Measure* of the posterior probability density function $p(x_t|y_{1:t})$, $y_{1:t} = (y_1, y_2, ..., y_t)$ the observations up to time t, and $\{\mathbf{x}_{0:t}^i, i = 0, 1, \ldots, N_s\}$ N_s particles in state space with associated weights $\{\omega_t^i, i = 1, 2, \ldots, N_s\}$. Then the posterior density can be approximated discretely:

$$p(\mathbf{x}_{0:t}|y_{1:t}) = \sum_{i=1}^{N_s} \omega_t^i \delta(\mathbf{x}_{0:t} - \mathbf{x}_{0:t}^i), \tag{2.1}$$

with the Dirac function $\delta(\mathbf{x})$. The weight of each particle is chosen in the principle of Importance Sampling [16, 17].

In the prediction step, the common choice of the proposal distribution is $f(x_{t+1}|x_t, y_{t+1}) = p(x_{t+1}|x_t)$. Particle filters using this distribution are known as bootstrap filters. Following [1], the second-order auto-regressive process is adopted to describe the dynamics of motion of the moving object. In the correction step, each particle at time t is weighted in proportion to the likelihood function $\mathbf{L}(y_t|x_t^{(i)})$, which is an observation y_t given the state $x_t^{(i)}$,

$$\omega_t^{(i)} \propto \omega_{t-1}^i \frac{\mathbf{L}(y_t|x_t^{(i)})p(x_t^{(i)}|x_{t-1}^{(i)})}{f(x_t^{(i)}|x_{t-1}^{(i)}, y_t)} = \omega_{t-1}^i \frac{\mathbf{L}(y_t|x_t^{(i)})}{\sum_{i=1}^N \mathbf{L}(y_t|x_t^{(i)})}, \tag{2.2}$$

where $p(x_t^{(i)}|x_{t-1}^{(i)})$ is the state transition function for all the particles.

2.2 Active Contour Methods

Many level set methods with the shape prior information [18, 5] have been demonstrated to extract objects of interest robustly in presence of heavy occlusion or distortion. Let Ω represent the whole image domain, $I(\mathbf{x})$ the intensity at position $\mathbf{x} = (x, y)$ in an image domain, $\Phi : \Omega \to \mathbf{R}$ the level set function. The boundary of objects or the interface $\partial\Omega$ of the level set Φ is represented implicitly as the zero level set $\Phi(x, t) = 0$. The segmentation problem is defined as an optimisation problem involving high-level shape information, with respect to some constraints. The energy to be minimised for partitioning an object region from its background is expressed as

$$\mathbf{E} = \mathbf{E}_{region} + \mathbf{E}_{shape} + \nu \cdot \mathit{Length}(\partial\Omega), \tag{2.3}$$

where $\mathbf{E}_{region}$ represents the energy term calculated from the image data which was defined in [8] as a region-driven term,

$$\mathbf{E}_{region} = \int_\Omega (I(x, y) - c_1)^2 \mathbf{H}(\mathbf{\Phi}) + (I(\mathbf{x}) - c_2)^2 (1 - \mathbf{H}(\mathbf{\Phi}))d\mathbf{x}, \tag{2.4}$$

where Φ_t means the level set function at time t. c_1 and c_2 are the values fitted iteratively by the pixel values of regions inside and outside the interface $\partial\Omega$, respectively. And $\mathbf{E}_{shape}$ was derived from the shape prior [22] as,

$$\mathbf{E}_{shape} = \int_\Omega \delta(\Phi)(s\Phi - \Phi_0(\mathbf{A}))^2 d\mathbf{x}, \tag{2.5}$$

where Φ_0 is the learned shape prior, and $\delta(x)$ is the Dirac function. And the scale parameter s is multiplied with the level set function Φ to consider the scale up/down effects of the transformation [18]. $\mathit{Length}(\partial\Omega)$ is the length of the interface $\partial\Omega$ with the weight ν that emphasizes the significance of curve length in the energy function.

3 Geometric Active Contour-based Tracking Estimation

3.1 Problems in Particle Filters

For the sake of convenience, we define particles which locate inside the boundary of the ground truth of the tracked object as *target particles*. In contrary, all the other particles are defined as *outlier particles*. Note that particle filters

Figure 3.1: Particles, which are scattering chaotically are drawn by little blue crossings. The estimated position for the helicopter is centered by the green ellipse. The ground truth of the boundary of tracked object is highlighted by the red line. This figure is cropped out from the video sequence data when the helicopter is in front of brown leaves. It is very difficult to recognize the helicopter even for human beings.

suffer from the degeneracy problem, which causes all but one particle have negligible weight after a few iterations. To reduce the variance of the importance weight and impose emphasis on larger weights which appear to be more helpful in representing the posterior, resampling methods [6] are followed after the correction step. To track the object robustly in complex environments such as in the presence of noise, clutter or occlusion, the highly discriminable feature plays a significant role in the observation model $p(y_t|\mathbf{x}_t)$. Usually, a fusion of multiple low-level features, such as texture, color and edge [4, 10], is selected as a potential feature. To determine properly the weights for individual features and select an appropriate strategy for fusion is always a non-trivial task. Moreover, the fundamental to expect a feature can perform well for a tracking task is that the adopted feature can be measured from image data. However, such measurability is no guarantee, especially for low resolution or real-world image data. Figure 3.1 shows the environment where the color information is not sufficient to distinguish the remotely controlled helicopter from the cluttered leaves of trees in the background. Moreover, the edge and texture information is not measurable since the small size of helicopter, the movement of leaves and the non-stationary camera. The particle filter will lose immediately the location of the helicopter when the helicopter is moving in front of the leaves.

Meanwhile, the traditional resampling algorithm such as sampling importance resampling [6] or systematic resampling algorithm [15] imposes negative influence on particle filters when the feature is not discriminable. The rationale is that at the resampling step all the particles are simply sorted with their meaningless weights $w_i, i \in N_s$, and then the re-sampled particles $\hat{w}_i, i \in N_s$are decided stochastically by $w_i, i \in N_s$ and the number of total particles. Therefore, the resampling step only emphasizes on these particles with larger weights without any further consideration. This simplification causes many useful geometric information, such as the allocation and scattering of particles, the shape of object, to be discarded. Figure 3.2 demonstrates the indiscriminable feature resampling step imposes the negative impact on the tracking procedure.

To solve the above problems, we define the state $\mathbf{x}$ of particle filtering as the image coordinate (x, y). Certainly, it is flexible to incorporate other parameters, such as the scale s and the rotation θ, into the state. An observation model using the geometric information and the level set-based active contour is then proposed in our work to classify *target particles* and *outlier particles*, and remove

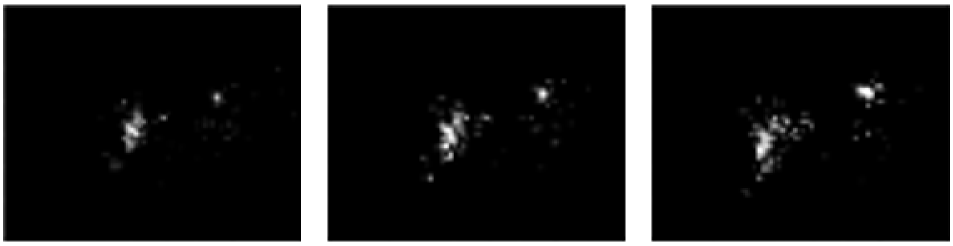


Figure 3.2: Systematic resampling with a indiscriminable feature: the negative influence is imposed on the tracking procedure. The group of gray-scale spots in each figure at left-hand side represents *target particles*, and the right-hand side group is *outlier particles*. *outlier particles* will be emphasized incorrectly when enter the scope of resampling. This incorrectness is demonstrated in the last two pictures in Figure 3.2.

outlier particles from the current particle population.

3.2 One-class Classification Problem and Active Contour Methods

GATE performs the level-set based active contour with shape prior and pose invariance to build the decision boundary in the state space, and only permits *target particles* entering the scope of resampling. The decision boundary construction problem is defined as an optimisation problem involving data partitioning in the state space, using the learned shape prior information, with respect to some constraints. This is actually a one-class classification problem [12] which is a new branch in pattern recognition that attempts to model one class of objects and distinguish it from all other possible outlier objects. Motivated by the work of [2], *GATE* directly constructs the decision boundary in the state space of particle filtering. The designed one-class boundary maximizes the inner class similarity, and minimises the dissimilarity between the boundary and the transformed prior shape while maintains the smoothness of the decision boundary.

However, except *GATE* focuses on refining the likelihood distribution, it is still fundamentally different from [2] in the classification point of view since *GATE* incorporates learned shape prior into the level set functional to construct a decision boundary in the state space

$$\begin{aligned}\mathbf{E} &= \int_\Omega \delta(\Phi)(\Phi-\Phi_0(\mathbf{A}))^2 d\mathbf{x} + \nu \int_\Omega |\bigtriangledown \mathbf{H}(\Phi)| d\mathbf{x} + \\ &\int_\Omega (f(\mathbf{x})-c_1)^2 \mathbf{H}(\boldsymbol{\Phi}) + (f(\mathbf{x})-c_2)^2(1-\mathbf{H}(\boldsymbol{\Phi}))d\mathbf{x}, \qquad (3.1)\end{aligned}$$

where$f(\mathbf{x}) = \{v_x | x \in \Omega'\}$ is a function representing the value v_x at point $\mathbf{x}$ in the state space Ω'. c_1 and c_2 represent average state values inside and outside the evolving interface $\partial\Omega$ which is defined in Section 2.2, respectively. The global property of averaging guarantees that the energy function has less local minimums compared with edge-driven energy term [21]. That means the interface $\partial\Omega$ can be attracted easily to the desired boundary even with the deepest descent method for optimisation. Two-dimension transformation matrix $\mathbf{A}$ maximizes the similarity between the transformed shape prior $\Phi(\mathbf{A})$ and the evolving decision boundary Φ. $\nu \int_\Omega | \bigtriangledown \mathbf{H}(\Phi)| d\mathbf{x}$ in (6) controls the degree of freedom of length and thus regularizing for the interface $\partial\Omega$. The Heaviside function $\mathbf{H}(\mathbf{\Phi})$ is a soft version of unit step function with $\lim_{\Phi\to 0} \mathbf{H}(\Phi) = 0.5$.

It is not straightforward to transform the non-sparse image data to general sparse one [2]. For non-sparse image data, the instances to be classified cover the whole working domain where the active contour performs segmentation or two-class classification. Since the positions $\mathbf{x}$ of particles are transmitted according to the dynamics of motion, multiple particles could be mapped into an identical position in the state space by dividing the state space into bins. According to Monte Carlo approximation [6],

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{i=1}^{N_s} w_t^i \delta(\mathbf{x}_t - \mathbf{x}_t^i) = \sum_{m=1}^{N_n} \sum_{n=1}^{N_m} w_t^n x_t^n, \sum_{m=1}^{N_n} N_m = N_s, \tag{3.2}$$

where N_s is the number of particles, N_n and N_m are the number of bins and the number of particles overlapped on the bin m in the state space, respectively. Hence, the value v_x can be derived:

$$v_x = \sum_{n=1}^{N_m} w_t^n. \tag{3.3}$$

The *target particles* and *outlier particles* are labeled by Ω_1 and Ω_2, respectively. After the construction of a decision boundary at each time t, particles can be classified by the construction of an indicator function that ties the locations of particles to their classes:

$$\mathbf{I}(\mathbf{x}) = \begin{cases} 1 & x \in \Omega_1 if \Phi(\mathbf{x}) \geq 0 \\ 0 & x \in \Omega_2 if \Phi(\mathbf{x}) < 0. \end{cases} \tag{3.4}$$

Examples of state space for particles and the corresponding decision boundary are shown in Figure 3, where the left figure illustrates particles in the state space and the corresponding decision boundary constructed via (6). The right figure shows the shape prior function Φ_0.

From the above description, *GATE* improves the likelihood distribution for particle filters in order to refine the posterior distribution $p(\mathbf{x}_t|y_{1:t})$.

3.3 Impoverishment Problem

Because of the capability of the level set to change the topology of the interface $\partial\Omega$ [14, 9], it is a natural to rectify the loss of the multiple peaks in probability density $p(\mathbf{x}_t|y_{1:t})$, which is known as impoverishment problem. Although a Voronoi tessellation could be used to find the clusters for particles [20] and then resample particles within each Voronoi cell, *GATE* can deal with clustering and classification simultaneously because it represents the interface $\partial\Omega$ implicitly. The weight μ for region-driven energy term controls the ability to maintain multiple peaks in probability density $p(\mathbf{x}_t|y_{1:t})$.

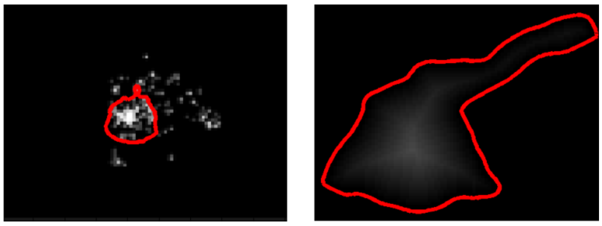


Figure 3.3: An example of the state space for particle filtering from the 404th frame of a helicopter video sequence. The left figure shows particles in the constructed state space which are represented by gray-scale spots, and the corresponding decision boundary which is highlighted by red contours after 50 iterations. The right figure illustrates the shape prior Φ_0 for the helicopter, and the boundary of the helicopter is highlighted by the red contour. Pictures are zoomed for demonstration.

3.4 Summary of *GATE*

The similarity transformation $\mathbf{A} = (s; \theta; \mathbf{T})$ with a scale factor s, a rotation angle θ and a translation vector $\mathbf{T}$ is sufficient to describe the "global" motion between the evolving level set Φ and the learned shape prior Φ_0 [11].

The proposal algorithm can be summarised as follows:

1. *Construct the working state space.* Determine the size of the state space. In our case, it consists of height R and width C of state space. The size of state space could be fixed in prior, or it could be determined by the scattering of particles for every input frame:

$$C = \max_{i \in N_n}(x_i) - \min_{i \in N_n}(x_i), \tag{3.5}$$

$$R = \max_{i \in N_n}(y_i) - \min_{i \in N_n}(y_i), \tag{3.6}$$

   where N_n is the number of bins in the state space and $\mathbf{x_i} = (x_i, y_i)$ is the position of bin i. For every bin in state space, the value of bin in the state space for particle filtering is defined as,

$$v_x = \sum_{n=1}^{N_m} w_t^n. \tag{3.7}$$

2. *Build decision boundary.* The final energy E for optimisation is defined:

$$\begin{aligned} \mathbf{E} &= \lambda_1 \mathbf{E}_{region} + \lambda_2 \mathbf{E}_{shape} + \nu Length(\partial\Omega) \\ &= \lambda_1 \int_\Omega (f(\mathbf{x}) - c_1)^2 \mathbf{H}(\mathbf{\Phi}) + (f(\mathbf{x}) - c_2)^2 (1 - \mathbf{H}(\mathbf{\Phi}))d\mathbf{x} + \nu \int_\Omega |\bigtriangledown \mathbf{H}(\Phi)| d\mathbf{x} \\ &\quad + \lambda_2 \int_\Omega \delta(\Phi)(\Phi - \Phi_0(\mathbf{A}))^2 d\mathbf{x}. \end{aligned} \tag{3.8}$$

   Thus, using the calculus of variations in (12), the level set function and its parameters evolve simultaneously as follows:

$$\begin{cases} \frac{d}{dt}\Phi = \delta(\Phi)[-2s\lambda_2(s\Phi - \Phi_0(\mathbf{A})) - \lambda_1 \log\frac{p_n}{p_0} + \nu div(\frac{\bigtriangledown\Phi}{|\bigtriangledown\Phi|}) \\ \frac{d}{dt}s = -2\int\int_\Omega[\mathbf{H}(\Phi)(s\Phi - \Phi_0((A)))(\Phi - \bigtriangledown\Phi_0(\mathbf{A}))\cdot\frac{d}{ds}(\mathbf{A_x})] \\ \forall j \in [1, N], \\ \frac{d}{dt}a_j = 2\int\int_\Omega[\mathbf{H}(\Phi)(s\Phi - \Phi_0(\mathbf{A}))(\bigtriangledown\Phi_0(\mathbf{A}))\cdot\frac{d}{da_j}(\mathbf{a})]. \end{cases} \tag{3.9}$$

3. *Refine the likelihood distribution* After some iterations, the category of a particle can be determined using (9). And the particles are filtered by multiplying the weight for each particle with the indicator function $\mathbf{I}(\mathbf{x})$ as,

$$w_i^\dagger = \mathbf{I}(\mathbf{x}) * w_i. \tag{3.10}$$

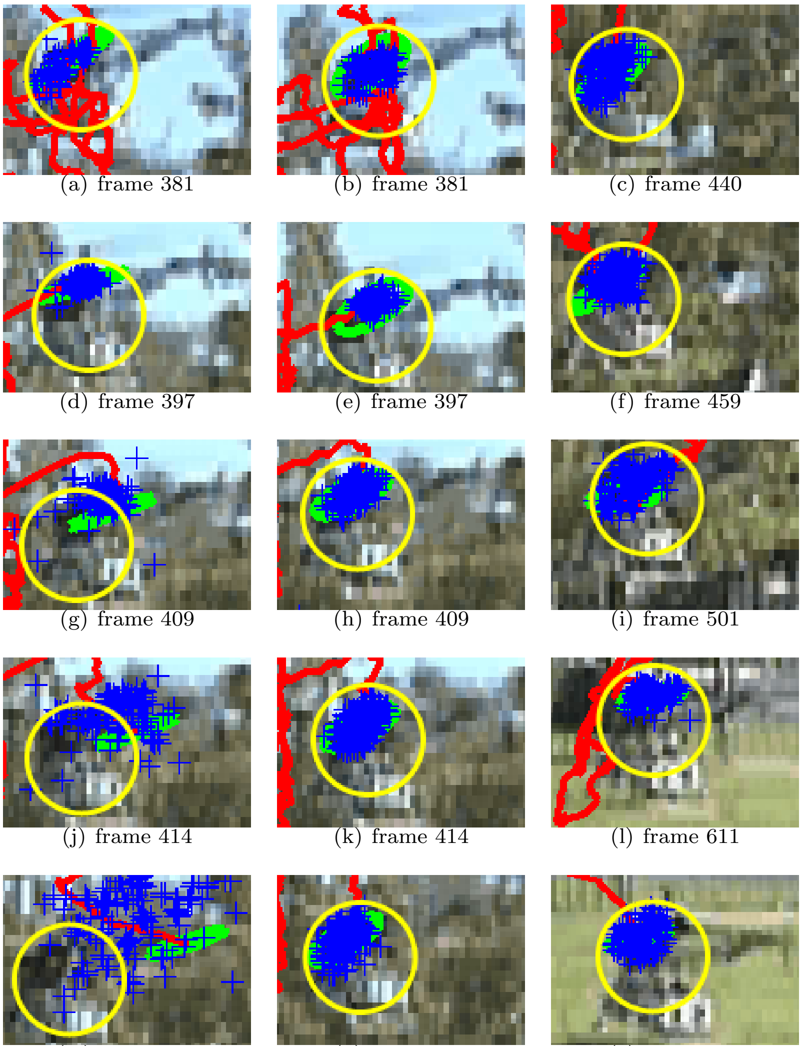
4. *Resample.* The filtered particles can be further re-sampled using any traditional resampling method. In our work, systematic resampling is preferred since it takes $\mathbf{O}(N)$ time to complete.

4 Experiments

In the following, *GATE* will be applied to object tracking in complex environments, where the feature in the observation model is not discriminable. To demonstrate the capability of our method to track the target in complex environments using a weak feature, only a simple color observation model [1] based on Hue-Saturation-Value(HSV) color histogram is adopted. Meanwhile, the similarity between the reference color model and a candidate color model is measured using the Bhattacharyya similarity coefficient. Since the learning of the coefficients of the auto-regressive model for the dynamics of motion and shape prior are out of the scope of this paper, we assume that those settings are learned using separate algorithms [1] and [17].

4.1 Description of Experimental Video Sequences

The experimental video sequence about the motion of remotely controlled helicopter manifests complex environments for object tracking in the following aspects. Firstly, as described in Section 3.1, there are no appropriate features, such as color, edge or/and texture, capable of distinguishing the helicopter from the trees in the background as shown in Figure 1. Secondly, the helicopter is kept floating at some positions in the image coordinate for many times. All of floating movements last for a non-trivial duration and make the dynamics of motion useless, because the helicopter is surrounded by the indiscriminable background. For example, when the helicopter is getting out of the cluttered leaves, the helicopter floats in front of a background clutter. And the helicopter is slowly moving away when 40 frames pasted. Finally, the low resolution video data($320 * 240$), the small size of the helicopter being tracked, the medium frame rate per second(15fps) and the dramatic motion of the camera all make the tracking task extremely difficult to deal with.



(a) frame 381 (b) frame 381 (c) frame 440

(d) frame 397 (e) frame 397 (f) frame 459

(g) frame 409 (h) frame 409 (i) frame 501

(j) frame 414 (k) frame 414 (l) frame 611

(m) frame 422 (n) frame 422 (o) frame 630

Figure 4.1: A complex environment for tracking a helicopter: *GATE* boosts a simple color feature to tackle complex environments and indiscriminable features. The left-most column shows the results of the sampling importance resampling particle filter. The trajectory of the helicopter is lost when the helicopter is in front of trees from the 409th frame. The middle and right-most columns show our method is still reliable to discover the position of the helicopter. Particles are represented by little blue crossings, the estimated position of the helicopter is centered by the green ellipse, and the trajectory of the helicopter is represented by the red line. The ground truth of the tracked helicopter is shown by the yellow circle. For the details, please have a look at our supplemental material.

The second surveillant video sequence about pedestrian is still a tough case for particle filters when the size and center of initial color template for observation are not delicately selected. The problem is caused by another highly similar pedestrian standing close to the object of interest in the background. Furthermore, it is not easy to select the size and center of initial color template without massive experiments or an elegant machine learning algorithm.

4.2 Experimental Setting and Results

Except adjusting the λ_1 to impose different emphasis on region-driven energy term in level set function, all other parameters are fixed in our experiments. The setting of parameters λ_2 and ν are 1 and 0.015, respectively. All the following figures are cropped for the purpose of demonstration, and the complete results can be found in the uploaded supplemental material.

As previously described, the trajectory of the helicopter tracked by the standard particle filter shown in the left-most column in Figure 4 is immediately unreliable when the helicopter is moving in front of the leaves. The middle and right-most columns in Figure 4 show the tracking results using our proposal method. It illustrates that the trajectory of the helicopter is really reliable because *outlier particles* are rejected and the posterior distribution is improved as described in Section 3.

Figure 5 shows our algorithm can maintain successfully the multiple peaks of posterior probability $p(\mathbf{x}_t|y_{1:t})$ because of the utility of the *GATE*. This initialises a novel approach to tackle the impoverishment problem for particle filters. As shown in the first row of Figure 5, the two peaks of posterior probability $p(\mathbf{x}_t|y_{1:t})$ is merging automatically. It demonstrates that the dynamics of motion of the target and the decision boundary in the state space can gradually interact with each other to achieve a better performance. Furthermore, the ability of *GATE* to maintenance of multiple peaks can be adjusted by changing the weight for region-driven energy term in the energy function of the level set method. The second row in Figure 5 shows the result of such adjustment. The two clusters of particles will not appear when the helicopter is crossing the ambiguous background clutter.

Figure 6 shows the results on pedestrian sequences. The first row in Figure 6 shows the performance of the standard particle filtering. Since there is a highly similar person standing close in the background to the person being tracking, the tracking fails when the person being tracked overlaps partially and then stays away from the background person. The second row in Figure 6 shows the performance of our proposal algorithm. Since our method refines the posterior distribution in particle filters, the loss of tracking has been rectified. λ_1 is set as 0.1.

5 Concluding Remarks

In this paper, we have proposed a novel tracking algorithm, *GATE*, using the level set-based active contour method and the particle filtering, for robust object tracking in complex environments in video. The *GATE* solves the ambiguous outlier problem of particle filters by preventing particles, which do not locate inside the ground truth of the boundary of the tracked object, entering the scope of resampling using pose invariance and the learned shape prior informa-

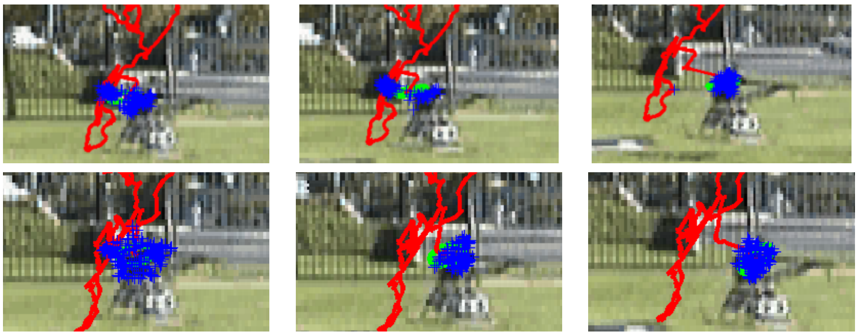


Figure 4.2: Maintaining multi-modality for particle filters via our method. Through adjusting the parameter λ_1 to impose different emphasis on region-driven energy term in (16), the ability to maintain multiple peaks of posterior probability $p(\mathbf{x}_t|y_{1:t})$ can be adjusted. The performance of maintenance of multiple peaks is illustrated in the first row where λ_1 is 0.3, and the release of such ability is illustrated in the second row where λ_1 is 0.2.

(a) frame 2 (b) frame 67 (c) frame 70 (d) frame 78 (e) frame 97

(f) frame 2 (g) frame 67 (h) frame 70 (i) frame 78 (j) frame 97

Figure 4.3: Pedestrian tracking. The first row shows the results of the standard particle filters. The particle filtering losses immediately the trajectory of the pedestrian from the 67th frame because the existence of a highly indiscriminable region in the background. The second row demonstrates the performance of our method which can track the pedestrian successfully.

tion. It significantly improves the performance of object tracking in complex environments, including heavy clutters in the background, low resolution and noisy image sequences, medium frame rate per second and non-stationary camera, even using a simple feature. Meanwhile, the proposed method initialises a novel approach to tackle the impoverishment problem of the particle filtering.

Bibliography

[1] Perez, P., Hue, C., Vermaak, J. and Gangnet, M.: Color-Based Probabilistic Tracking. ECCV 2002. Lecture Notes in Artificial Intelligence 2350, pp. 661-675. Springer-Verlag Berlin Heidelberg(2002).

[2] Cai, X., Sowmya, A.: Level Learning Set: A Novel Classifier Based on Active Contour Models. ECML 2007. Lecture Notes in Artificial Intelligence 4701, pp. 79-90. Springer-Verlag Berlin Heidelberg(2007).

[3] Rathi, Y., Vaswani, N., and Tannenbaum, A.: Particle Filtering for Geometric Active Contours with Application to Tracking Moving and Deforming Objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(6), pp. 1470-1475 (2007).

[4] Yilmz, A., Li, X., and Shah, M.: Contour-Based Object Tracking with Occlusion Handling in Video Acquired Using Mobile Cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(11), pp. 1531-1536(2004).

[5] Rousson, M., Paragios, N.: Shape Priors for Level Set Representations. ECCV 2002. LNCS 2351, pp. 78-92. Springer-Verlag Berlin Heidelberg(2002).

[6] Arulampalam, S., Maskell, S., Gordon, N., and Clapp, T.: A Tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking. IEEE Transactions on Signal Processing 50(2), pp. 174-188(2001).

[7] Okuma, K., Taleghani, A., Freitas, N., Little, J., and Lowe, D.: A Boosted Particle Filter: Multitarget Detection and Tracking. ECCV 2004. Lecture Notes in Computer Science 3021, pp. 28-39. Springer-Verlag Berlin Heidelberg(2004).

[8] Chan, T.FF., Vese, L.A.: Active Contours Without Edges. IEEE Transactions on Image Processing 10(2), pp. 266-277(2001).

[9] Caselles, V., Kimmel, R., Sapiro, G.: Geodesic Active Contours. International Conference on Computer Vision , pp. 694-699(1995).

[10] Brasnett, P., Mihaylova, L., Bull, D., and Canagarajah, N.: Sequential Monte Carlo Tracking by Fusing Multiple Cues in Video Sequences. Image and Vision Computing 28(1), pp. 1217-1227. Elsevier Science(2007).

[11] Soatto, S., Yezzi, A.: DEFORMATION Deforming Motion, Shape Average and the Joint Registration and Segmentation of Images. International Journal of Computer Vision 53(2), pp. 153-167. Springer(2003).

[12] Tax, D.: One-class Classification. PhD thesis, Delft University of Technology(2001).

[13] Malladi, R., Sethian, J.A., Vemuri, B.C.: Shape Modeling with Front Propagation: a Level Set Approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 17(2), pp. 158-178. IEEE Computer Society(1995).

[14] Kitagawa, G.: Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models. Journal Of Computational and Graphical Statics 5(1), pp. 1-25. 1996.

[15] Bergman, N.: Recursive Bayesian Estimation: Navigation and Tracking Application. PhD thesis, Linkoping University(1999).

[16] Doucet, A.: On Sequential Monte Carlo Methods for Bayesian Filtering. Technical Report, University of Cambridge(1998).

[17] Paragios, N., Rousson, M., and Ramesh, V.: Matching Distance Functions: A Shape-to-Area Variational Approach for Global-to-Local Registration. ECCV 2002. Lecture Notes in Computer Science 2351, pp. 775-789. Springer-Verlag Berlin Heidelberg(2002).

[18] Vermaak, J., Doucet, A., and Perez, P.: Maintaining Multi-Modality through Mixture Tracking. International Journal of Computer Vision, pp. 1110-1116. IEEE Computer Society(2003).

[19] Brox, T., Weickert, J.: Level Set Based Image Segmentation with Multiple Regions. Pattern Recognition 2004. Lecture Notes in Computer Science 3175, pp. 415-423. Springer-Verlag Berlin Heidelberg(2004).

[20] Fussenegger, M., Deriche, R., and Pinz, A.: Multiregion Level Set Tracking with Transformation Invariant Shape Priors. ACCV 2006. Lecture Notes in Computer Science 3851, pp. 674-683. Springer-Verlag Berlin Heidelberg(2003).

[21] Yilmaz, A., Javed, O., and Shah, M.: Object Tracking: A Survey. ACM Computing Surveys 38(4), pp. 1-45. ACM Press(2006).

[22] Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active Contour Models. International Journal of Computer Vision, pp. 321-331(1988).