Level-Set Curve Particles*

Tingting Jiang and Carlo Tomasi

Department of Computer Science, Duke University, Durham NC 27708, USA
{ruxu, tomasi}@cs.duke.edu

Abstract. In many applications it is necessary to track a moving and deforming boundary on the plane from infrequent, sparse measurements. For instance, each of a set of mobile observers may be able to tell the position of a point on the boundary. Often boundary components split, merge, appear, and disappear over time. Data are typically sparse and noisy and the underlying dynamics is uncertain. To address these issues, we use a particle filter to represent a distribution in the large space of all plane curves and propose a full-fledged combination of level sets and particle filters. Our main contribution is in controlling the potentially high expense of multiplying the cost of a level set representation of boundaries by the number of particles needed. Experiments on tracking the boundary of a colon in tomographic imagery from sparse edge measurements show the promise of the approach.

1 Introduction

Many applications require tracking boundaries that move and deform over time, whether these are the contours of an oil spill in the ocean, a plume of smoke, a hurricane, a wildfire, the dividing cells under a microscope, a running crowd of people, or the contours of an anatomical structure tracked across slices of a volumetric medical image. With imagery, measurements of the boundary of interest are typically dense and abundant. Even so, the presence of clutter often requires estimating the boundaries stochastically, by letting a probability distribution entertain multiple hypotheses about the boundary's position and shape. In other applications, observations are much more sparse and perhaps less frequent: a small number of mobile observers may know their own location through GPS, and perhaps measure their distance from the boundary of interest through imaging, range finding, or other sensors.

These problems have a common abstraction: a temporally discrete, possibly spatially sparse distribution of noisy point measurements is used to infer the shape and position of a boundary that moves and deforms on the plane under the influence of only partially understood causes.

Conceptually, the solution ingredients are known: a dynamic system models the underlying evolution phenomenon; stochastic estimation addresses sparsity

* This research was sponsored through a subcontract from Intelligent Automation, Inc. under U.S. Army STTR Phase II Grant W911NF-04-C-0114, and through NSF grant IIS-0534897.

A. Leonardis, H. Bischof, and A. Pinz (Eds.): ECCV 2006, Part III, LNCS 3953, pp. 633–644, 2006.

and uncertainty of the measurements; and level sets elegantly represent boundaries whose components merge, split, appear, and disappear. All these ingredients have been used with success. Their full combination, however, has not, because of its potentially high computational complexity: the high cost of level sets is charged to each of a large number of particles needed to represent a probability distribution in the space of plane curves. The number of particles required depends on how large the space of all possible plane curves is. With some assumptions and prior knowledge about the plane curves such as smoothness and initialization, the probability distribution of the plane curves can be represented by a particle filter of applicable size. However, the complexity of combining level sets and particle filter is still high.

In this paper we show a way to keep this complexity in check. This leads to a tracking method of great generality and flexibility, based on the core concept of *level-set curve particles.* This method capitalizes on the observation that many of the curves that populate the boundary distribution being tracked are very similar to each other. In our method, a number of *base curves* account for macroscopic differences between boundaries. Each base curve is then deformed by P *base perturbations* which provide an implicit representation of the 2^P deformations obtained by applying any subset of the P base perturbations to the base curve. Through this device, an exponential number of curves can be propagated and their likelihoods can be computed at linear cost in the framework of a particle filter. Resampling has still an exponential cost. However, the per-curve cost of resampling is trivial, while the per-curve cost of propagation is proportional to the size of the data structure needed to represent an entire level set, and the unit cost of likelihood estimation is proportional to the number of measurements. Saving on propagation has a huge effect on running time, and saving on likelihood computation has a significant effect when measurements are plentiful.

In the next Section, we review related work. Section 3 defines level-set curve particles and shows the tracking algorithm. Section 4 presents the experimental results and Section 5 concludes with a summary and plans for future work.

2 Related Work

Active contours [1, 2] are based on ideas developed initially for Brownian motion [3, 4], and later incorporated into *particle filters* (see [5] for a recent overview). These approaches capture the uncertain position of a boundary at time t by a probability distribution represented by a random sample of boundaries (*particles*). Each boundary is represented explicitly, e.g. with splines [6] and *propagated* forward in time through an assumed, uncertain motion model. Measurements update the particles through *resampling*, which weighs each particle by its posterior probability given the measurements. This is computed from the *likelihood* (conditional probability density of a measurement given a boundary) through Bayes' theorem. New particles are drawn from the posterior, and are ready for a new step of propagation. This cycle is analogous to the estimation loop of a Kalman filter [7], but maintains a multi-modal distribution rather than a Gaussian one.

Level sets [8, 9, 10] describe a boundary as the zero crossing of a function $\phi(\boldsymbol{x}, t)$ of space and time. While many functions ϕ share the same zero-crossing $\mathcal{B}$, computational considerations (see [11]) suggest using the signed distance function of $\mathcal{B}$, which is then maintained in a narrow *band* [12] around the boundary $\mathcal{B}$. The motion model is then a PDE for ϕ. The main strength of level sets is that they account effortlessly for changes of boundary topology, as exemplified in applications to image segmentation [13], object detection [14], tracking [15], shape modeling [16] and medical image segmentation [17], among others.

Recently, several papers [14, 18, 19, 20, 21, 22] have combined active contours and level sets. These papers create suitable artificial "forces" that draw an initial boundary towards a boundary of interest. These approaches essentially seek a new boundary in each frame, view a boundary as a deterministic object, and do not incorporate prior knowledge of boundary motion. This holds also for work based on "shape averages" [23, 22] where the deformation of a boundary is decomposed into an average motion plus a set of local deformations. More specifically, in [22] the propagation of the local deformations is still deterministic for each particle, with no way to add "noise" to model the uncertainty of the underlying dynamics.

In contrast, we combine active contours and level sets into a full-fledged particle filter for boundaries. Our *level-set curve particles* marry the power of stochastic estimation methods from active contours with the flexibility of non-parametric level sets.

3 Approach

3.1 Problem Statement

A dynamic boundary $\mathcal{B}$ is a variable number of moving and deforming closed curves on the plane. These curves may merge, split, appear, and disappear over time. The boundaries $\mathcal{B}^{(1)}$ and $\mathcal{B}^{(2)}$ at initial times 1 and 2 are assumed to be known. M mobile and controllable observers make noisy measurements of the position of a point on the boundary $\mathcal{B}$. The boundary tracking problem is to use the observations $\{Q_m^{(t)}\}_{m=1}^M$ made by the observers at times $t = 3, 4, \ldots$ to estimate the posterior probability distribution of the boundary $\mathcal{B}^{(t)}$ at these points in time. The Maximum A Posteriori (MAP) boundary estimate may optionally be computed when requested.

3.2 Level-Set Curve Particles

We use a set of $P + 1$ particles $\chi_t = \{\phi_p^{(t)}, w_p^{(t)}\}_{p=0}^P$ to represent the probability distribution of the boundary $\mathcal{B}^{(t)}$. Each particle is a signed distance function $\phi_p^{(t)} : \mathbb{R}^2 \to \mathbb{R}$ whose zero level set denotes an estimate of the boundary $\mathcal{B}^{(t)}$. A weight $w_p^{(t)}$ determined by the measurements is associated to each particle. Since all the particles are estimates of the same boundary, it can be expected that the signed distance function values are similar for most points on the plane.

For efficiency, we represent particles as a multiple perturbation of a "mother" particle $\phi_0^{(t)}$. The perturbations $\{\phi_p^{(t)}\}_{p=1}^P$ are "child" particles. For each $p > 0$, define a function $\Delta\phi_p^{(t)}$ such that $\phi_p^{(t)}$ and $\phi_0^{(t)} + \Delta\phi_p^{(t)}$ share the same zero level crossing and $\Delta\phi_p^{(t)}$ is nonzero only inside a small window $W_p^{(t)}$.

This construction yields $P + 1$ "explicit" particles. However, 2^P of particles can be obtained by combining the mother particle with subsets of the deformations $\Delta\phi_p^{(t)}$ for its children. For example, a particle $\phi_{1,2}^{(t)} = \phi_0^{(t)} + \Delta\phi_1^{(t)} + \Delta\phi_2^{(t)}$ can be obtained that concurrently deforms the mother particle in the union of the two windows $W_1^{(t)}$ and $W_2^{(t)}$. More generally, given the set of child indices $X = \{1, \ldots, P\}$, for every set Z in the power set of X we can define a new perturbation $\phi_Z^{(t)} = \phi_0^{(t)} + \sum_{j \in Z} \Delta\phi_j^{(t)}$ of the mother particle $\phi_0^{(t)}$.

The probability distribution of the boundary $\mathcal{B}^{(t)}$ is encoded by the set of parameters $\chi_t = \{\phi_0^{(t)}, \{W_p^{(t)}, \Delta\phi_p^{(t)}\}_{p=1}^P, \{w_p^{(t)}\}_{p=0}^P\}$ which now represents a much bigger sample. Of course, the challenge is to propagate, update, and resample the 2^P "implicit" particles by working only with the $P + 1$ "explicit" ones. The next Sections show how to do this.

3.3 Tracking Algorithm

The outline of the proposed tracking method is Algorithm 1. Let $Q_m^{(t)}$ denote the estimate of a point on boundary $\mathcal{B}^{(t)}$ returned by observer number m for $m = 1, \ldots, M$. Let χ_t be the level-set curve particle set at time step t. Since $\mathcal{B}^{(1)}$ and $\mathcal{B}^{(2)}$ are known, the corresponding signed-distance functions $\phi^{(1)}$ and $\phi^{(2)}$ can be computed. Tracking then starts from $t = 3$. The main parts of this algorithm are discussed next.

Algorithm 1. Tracking algorithm

```
INPUT: $\chi_{t-1}$ $(t > 2)$
OUTPUT: $\chi_t$.
(1) Propagate $\chi_{t-1}$ to $\bar{\chi}_t$;
(2) Observers take measurements $\{Q_m^{(t)}\}_{m=1}^M$;
(3) Update $\bar{\chi}_t$;
(4) Resample and generate $\chi_t$.
```

Initialization. Algorithm 1 must be initialized with an initial particle set χ_2 that is, with a set of functions $\{\phi_p^{(2)}\}_{p=0}^P$ each of which is a random perturbation of a given initial boundary estimate $\mathcal{B}^{(2)}$. All functions are given the same weight. To define perturbations, we need a notion of "closeness" between boundaries. Given $\epsilon > 0$ and a boundary $\mathcal{B}$, the neighborhood of $\mathcal{B}$ within distance ϵ is defined as $N_\epsilon(\mathcal{B}) = \{\mathcal{B}' : D(\mathcal{B}, \mathcal{B}') < \epsilon\}$ where $D(\mathcal{B}, \mathcal{B}')$ is the symmetric set difference between the areas enclosed by $\mathcal{B}$ and $\mathcal{B}'$ normalized by the area of $\mathcal{B}$.

We perturb $\mathcal{B}^{(2)}$ by adding a function δ to $\phi^{(2)}$. Then the zero crossing of $\phi' = \phi^{(2)} + \delta$ is the perturbed boundary $\mathcal{B}'$. To make $\mathcal{B}'$ continuous and within $N_\epsilon(\mathcal{B}^{(2)})$ for a given small $\epsilon > 0$, the function δ should be continuous and its

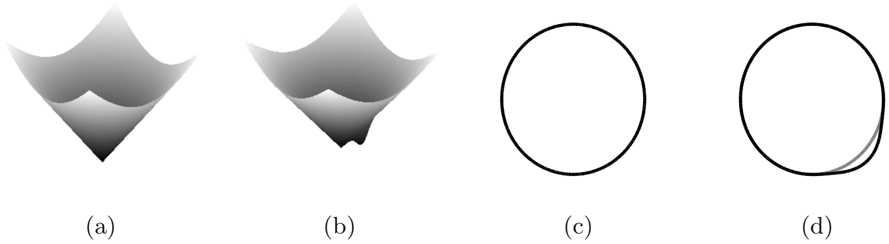
(a) (b) (c) (d)

Fig. 1. (a) The signed distance function ϕ of a boundary $\mathcal{B}$. (b) The perturbed function ϕ'. (c) The initial boundary $\mathcal{B}$. (d) The perturbed boundary $\mathcal{B}'$ (dark).

values should be small. We choose $\delta(\boldsymbol{x}) = \alpha G_2(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\Sigma})$, the product of a scalar α and a 2D Gaussian function $G_2(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\Sigma})$ with mean $\boldsymbol{y}$ and diagonal covariance matrix Σ. The vector $\boldsymbol{y}$ determines the center of the perturbation, $\boldsymbol{\Sigma}$ controls its extent, and α determines the amount of perturbation. All of these parameters are generated randomly. $\boldsymbol{y}$ is usually chosen close to $\mathcal{B}^{(2)}$, and α and $\boldsymbol{\Sigma}$ are chosen to satisfy $\mathcal{B}' \in N_\epsilon(\mathcal{B}^{(2)})$. A sample perturbation is shown in Fig. 1.

With this method we generate P boundaries $\mathcal{B}'_p(p = 1, \ldots, P)$ that perturb $\mathcal{B}^{(2)}$. From these, we compute P signed-distance functions $\phi'_p = \gamma(\phi^{(2)} + \delta_p)$ where $\gamma(\phi)$ denotes the signed distance function of the zero level set of ϕ. To construct the initial particle set χ_2, we define the mother particle $\phi_0^{(2)} = \phi^{(2)}$. The difference between each ϕ'_p and $\phi_0^{(2)}$ is $\Delta\phi_p^{(2)} = \phi'_p - \phi_0^{(2)}$. Each window $W_p^{(2)}$ is defined as the smallest rectangle which includes all the points $\boldsymbol{x}$ such that $|\Delta\phi_p^{(2)}(\boldsymbol{x})|$ is not trivial and $|\phi'_p(\boldsymbol{x})|$ is below a small positive threshold dependent on both the width of the narrow band and the dynamics of the boundary[1]. The perturbations $\Delta\phi_p^{(2)}(\boldsymbol{x})$ are then truncated to zero for $\boldsymbol{x}$ outside $W_p^{(2)}$. All particles are given the same initial weight $w_p^{(2)} = 1/(P+1)$. Given the initial particle set χ_2, we maintain the particle set χ_t over time through Algorithm 1.

Propagation. Suppose that the velocity of each point on the plane at time t is given as $\boldsymbol{v}(\boldsymbol{x})$. We wish to move all the points on the surface $\phi^{(t-1)}$ to $\bar{\phi}^{(t)}$ as $\phi^{(t-1)}(\boldsymbol{x}) = \bar{\phi}^{(t)}(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x}))$. Assume that mother particle $\phi_0^{(t-1)}$ is propagated to $\bar{\phi}_0^{(t)}$, $\phi_0^{(t-1)}(\boldsymbol{x}) = \bar{\phi}_0^{(t)}(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x}))$. Similarly, the difference between mother and child particles is propagated to $\Delta\bar{\phi}_p^{(t)}$. According to the above equation, $\bar{\phi}_Z^{(t)}(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x})) = \phi_Z^{(t-1)}(\boldsymbol{x}) = \phi_0^{(t-1)}(\boldsymbol{x}) + \Sigma_{j\in Z}\Delta\phi_j^{(t-1)}(\boldsymbol{x}) = \bar{\phi}_0^{(t)}(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x})) + \Sigma_{j\in Z}\Delta\bar{\phi}_j^{(t)}(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x}))$. Therefore, we can define $\bar{\phi}_Z^{(t)} = \bar{\phi}_0^{(t)} + \Sigma_{j\in Z}\Delta\bar{\phi}_j^{(t)}$ if $\boldsymbol{x} \to \boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x})$ is a one-to-one mapping, which implies that all "implicit" particles can be propagated by only explicitly propagating the mother particle and the differences between mother and child particles.

The windows $W_p^{(t-1)}$ are propagated by modifying them to $\overline{W}_p^{(t)}$ so as to properly enclose nonzero values of $\Delta\bar{\phi}_p^{(t)}$. The propagated particle set is now

[1] $W_p^{(2)}$ should be large enough to include all possible boundary points at the next time step. Derivation details for this threshold are omitted for lack of space.

represented by $\bar{\chi}_t = \{\bar{\phi}_0^{(t)}, \{\Delta\bar{\phi}_p^{(t)}, \overline{W}_p^{(t)}\}_{p=1}^P, \{w_p^{(t-1)}\}_{p=0}^P\}$. During propagation, the particle weights $w_p^{(t-1)}$ do not change. In our experiments, $\boldsymbol{v}(\boldsymbol{x})$ at each time step t is generated from the difference between the maximum-likelihood particles of the previous two time steps $t-1$ and $t-2$. However, in some applications, $\boldsymbol{v}(\boldsymbol{x})$ might be measured separately. For numerical purposes, we enforce propagated mother particle and child particles to be signed distance functions.

Update. After propagation, each observer returns a noisy measurement $Q_m^{(t)}$ of a point on the boundary. One simple likelihood function of a particle ϕ is $\Lambda_t(\phi) = \prod_{m=1}^M G(\phi(Q_m^{(t)}))$ where G is a Gaussian function whose standard deviation ζ depends on the noise statistics of the measurements, assumed to be mutually independent. And $\phi(Q_m^{(t)})$ denotes the function value of ϕ at point $Q_m^{(t)}$. If $|\phi(Q_m^{(t)})|$ is large, $Q_m^{(t)}$ is far away from the zero level set of ϕ and therefore the likelihood that the zero crossing of ϕ passes through point $Q_m^{(t)}$ is small. On the other hand, if $\phi(Q_m^{(t)}) = 0$, $Q_m^{(t)}$ lies exactly on the zero crossing of ϕ. Given $\Lambda_t(\phi)$, we can update the weight of each particle as $\bar{w}_p^{(t)} = w_p^{(t-1)} \cdot \Lambda_t(\bar{\phi}_p^{(t)})$. Since each child particle p only differs from the mother particle inside window $\overline{W}_p^{(t)}$, the ratio $H_p^{(t)} = \frac{\Lambda_t(\bar{\phi}_p^{(t)})}{\Lambda_t(\bar{\phi}_0^{(t)})}$ depends only on the difference between function values of $\bar{\phi}_0^{(t)}$ and $\bar{\phi}_p^{(t)}$ at the measurement points $Q_m^{(t)}$ inside window $\overline{W}_p^{(t)}$. So we can first calculate the likelihood $\Lambda_t(\bar{\phi}_0^{(t)})$ for the mother particle and then check for each child particle p whether there are any $Q_m^{(t)}$ inside $\overline{W}_p^{(t)}$. If not, the likelihoods of the mother and the child are same. Otherwise, calculate $H_p^{(t)}$ and get $\Lambda_t(\bar{\phi}_p^{(t)})$ by the above ratio equation. After update, the particle set $\bar{\chi}_t$ becomes $\{\bar{\phi}_0^{(t)}, \{\Delta\bar{\phi}_p^{(t)}, \overline{W}_p^{(t)}\}_{p=1}^P, \{\bar{w}_p^{(t)}\}_{p=0}^P\}$.

Resampling. In a standard particle filter, resampling is drawing (with replacement) from the set of particles with probabilities proportional to their weights. In our method, $P+1$ "explicit" particles represent 2^P "implicit" particles. From $\bar{\chi}_t$, only the weights of the "explicit" particles $\{\bar{w}_p^{(t)}\}_{p=0}^P$ are known. So we need to evaluate the weights of the "implicit" particles before resampling. First define $K_p^{(t)} = \frac{w_p^{(t)}}{w_0^{(t)}}$ and similarly $\bar{K}_p^{(t)} = \frac{\bar{w}_p^{(t)}}{\bar{w}_0^{(t)}}$ for each $p > 0$. We consider two cases:

Disjoint Windows. Assume $W_i^{(t)} \cap W_j^{(t)} = \emptyset$ for $i \neq j,\ i, j = 1, \ldots, P$ and for all t. For this case, we can draw the following conclusion:

Lemma 1. *Given $Z \subseteq X = \{1, \ldots, P\}$, let $w(\phi_Z^{(t-1)}) = w_0^{(t-1)} \cdot \prod_{j \in Z} K_j^{(t-1)}$ be the weight of the "implicit" particle $\phi_Z^{(t)}$. After propagation, if the corresponding windows $\{\overline{W}_j^{(t)}\}_{j \in Z}$ are disjoint, the weight of $\bar{\phi}_Z^{(t)}$ is $w(\bar{\phi}_Z^{(t)}) = \bar{w}_0^{(t)} \cdot \prod_{j \in Z} \bar{K}_j^{(t)}$.*

The proof is omitted here due to space limits. However, the result stands to reason because measurements in each window contribute independently to $w(\phi_Z^{(t)})$.

From Lemma 1, we see that if the condition $w(\phi_Z^{(t-1)}) = w_0^{(t-1)} \cdot \prod_{j \in Z} K_j^{(t-1)}$ is satisfied before propagation at time step t, the evaluation of $w(\bar{\phi}_Z^{(t)})$ is just the product of the weight of mother particle and $\bar{K}_j^{(t)}$ of each child particle in Z. This condition is iteratively satisfied if the windows are always disjoint. In this case, resampling can be done as follows. Let particle i correspond to an "implicit" particle ϕ_{Z_i} where $Z_i \subseteq X$, $(1 \leq i \leq 2^P)$. Let $w_i = \bar{w}_0^{(t)} \cdot \prod_{j \in Z_i} \bar{K}_j^{(t)}$ be the weight for particle i and $\pi_i = \frac{w_i}{\Sigma_{i=1}^{2^P} w_i}$ be the normalized weight for particle i. We can draw with replacement from the 2^P particles with probabilities proportional to π_i. Suppose the newly generated weight for each particle is μ_i which might be different from π_i due to the limited number of samples. Since we only maintain the weights for $P+1$ "explicit" particles instead of 2^P "implicit" particles , it is desirable to represent μ_i by a new set of K_p, i.e., each μ_i could be approximated as $\frac{\Pi_{j \in Z_i} K_j}{\Sigma_{i=1}^{2^P} \Pi_{j \in Z_i} K_j}$. Thus, the goal of resampling "implicit" particles is achieved by changing the weight of "explicit" particles. This leads to minimizing[2] the function $F(K_1, K_2, \ldots, K_P) = \sum_{i=1}^{2^P} (\mu_i - \frac{\Pi_{j \in Z_i} K_j}{\Sigma_{i=1}^{2^P} \Pi_{j \in Z_i} K_j})^2$ by which we can find a new set of K_p to replace $\bar{K}_p^{(t)}$. Then the weight of each child particle is updated as $w_p^{(t)} = \bar{w}_0^{(t)} \cdot K_p$ while the weight of the mother particle does not change. After resampling, a new particle set $\chi_t = \{\phi_0^{(t)}, \{\Delta\phi_p^{(t)}, W_p^{(t)}\}_{p=1}^P, \{w_p^{(t)}\}_{p=0}^P\}$ is generated.

Remark. Two special cases need to be handled. If $K_p = 0$, the weight of particle p is very small and discarded. If K_p is very large, this perturbation will be generated with high probability, so it should be incorporated into the mother particle. In both cases, we need generate new particles to replace the ones that are eliminated. During resampling, we can also optionally find the particle with the maximum weight and return its zero level set as the MAP estimate of $\mathcal{B}^{(t)}$.

Intersecting Windows. Since windows move, even windows that are initially disjoint may eventually intersect. Now consider the case in which any two windows can intersect. Formally, $\exists j \neq k$, s. t. $W_j^{(t)} \cap W_k^{(t)} \neq \emptyset$, $j, k = 1, \ldots, P$. In this case, we need additional information to maintain the weights of the "implicit" particles. Specifically, at each time step t, given χ_t, define $\{S_{jk}^{(t)}\}$ as a set of $\binom{P}{2}$ real numbers s.t. $\forall Z \subseteq X$, $w(\phi_Z^{(t)}) = w_0^{(t)} \prod_{p \in Z} K_p^{(t)} \prod_{j,k \in Z, j \neq k} S_{jk}^{(t)}$. Then we have the following conclusion similar to Lemma 1:

Lemma 2. *Given $Z \subseteq X = \{1, \ldots, P\}$, suppose the combined particle $\phi_Z^{(t)}$ has weight $w(\phi_Z^{(t-1)}) = w_0^{(t-1)} \cdot \prod_{p \in Z} K_p^{(t-1)} \prod_{j,k \in Z, j \neq k} S_{jk}^{(t-1)}$ before propagation. After propagation, the weight of $\bar{\phi}_Z^{(t)}$ is $w(\bar{\phi}_Z^{(t)}) = \bar{w}_0^{(t)} \cdot \prod_{p \in Z} \bar{K}_p^{(t)} \prod_{j,k \in Z, j \neq k} \bar{S}_{jk}^{(t)}$ where $\bar{S}_{jk}^{(t)} = S_{jk}^{(t-1)} \cdot I_{jk}^{(t)}$ and $I_{jk}^{(t)} = exp(\sum_{Q_m^{(t)} \in \overline{W}_j^{(t)} \bigcap \overline{W}_k^{(t)}} \frac{-2\Delta\bar{\phi}_j^{(t)}(Q_m^{(t)})\Delta\bar{\phi}_k^{(t)}(Q_m^{(t)})}{\zeta^2})$.*

This lemma tells us that if intersections between windows are allowed, in addition to maintaining χ_t, it is also necessary to maintain an intersection factor set $\{S_{jk}^{(t)}\}$

[2] Minimizing $F(K_1, K_2, \ldots, K_P)$ is an approximation. If we can draw unlimited samples, μ_i will be equal to π_i and therefore $\bar{K}_p^{(t)} = K_p$. When the number of samples is limited, $\{\bar{K}_p^{(t)}\}$ is a good starting point to find $\{K_p\}$, so convergence to a correct minimum is likely. We do not yet have a proof for this conjecture.

for every pair of child particles over time. $S_{jk}^{(t)}$ is updated as $\bar{S}_{jk}^{(t)} = S_{jk}^{(t-1)} \cdot I_{jk}^{(t)}$. With $\bar{S}_{jk}^{(t)}$, we could approximately optimize $\{\bar{K}_p^{(t)}\}$ by minimizing function $F'(K_1, K_2, \ldots, K_P)$ which is defined similarly to F except that the products of intersection factors are added as constants in F'. The cost of maintaining intersection factors $\{S_{jk}^{(t)}\}$ is $O(MP^2)$.

New Component. If some measurement $Q_m^{(t)}$ is far away from the zero level sets of the current (implicit) particles, it is possible that a new component of the dynamic boundary has appeared. To account for cases like this, we randomly generate a small boundary component near $Q_m^{(t)}$ and add it to the mother particle, so that all child particles inherit this component automatically. Results of adding new components are shown in Section 4.

Efficiency. We now analyze the computation cost of the proposed method. Assume that all level set functions are defined on a grid of size N. The computation cost of the signed-distance function γ is $O(N)$. For initialization, we generate P child particles based on the given mother particle. For each child particle, we randomly generate a 2D Gaussian perturbation, at a cost $O(N)$. Propagation advances $P+1$ particles by the motion field, at a cost $O(PN)$. For each "explicit" particle, evaluating the weight according to M measurements will takes $O(PM)$ at worst. Resampling evaluates the weights for all 2^P "implicit" particles and then resamples 2^P numbers to generate the new μ_i, so the cost of this stage is $O(2^P)$. Note however that this computation occurs on numbers instead of functions. So the constant factor that multiplies 2^P is small. For the case of intersecting windows, the cost of maintaining intersection factors $\{S_{jk}^{(t)}\}$ is $O(MP^2)$. Optimization of function F or F' is fast practically because $\{\bar{K}_p^{(t)}\}$ is very close to $\{K_p\}$. The generation of new components takes at most $O(MN)$ time. From this analysis, the total complexity for our tracking algorithm is $O(2^P + PN + PM + MN + MP^2)$. If we had used a standard particle filter with 2^P "explicit" particles the cost for propagation and update would be $O(2^P(N+M))$ which is much higher than that for the proposed approach.

4 Experiments

Experimentation with our proposed framework runs apparently into two conflicting requirements: On one hand, we want to demonstrate the performance of our tracker when the set of available measurements is sparse, because sparseness is one of the main challenges of this problem. On the other hand, we need reference ground truth against which we can measure performance, and this requires dense information about the true location of the boundary. In order to meet both requirements, we have tracked the boundaries in a data set where we have dense information, but where we simulate sparseness by withholding that information from our algorithm. Specifically, we have obtained a sequence of 355 slices from a Computerized Axial Tomography (CAT) scan of a human colon. The boundaries in this data are very complex (see Fig. 3(d)): the colon tube is convoluted, and its frequent turns make boundaries appear and disappear as we progress from one slice to the next. In our experiments, we detected each boundary with a standard edge detector. In Fig. 3(a)-(c), these "true" boundaries are drawn in red. These are our ground truth, to which the tracker has no direct access. Instead, the tracker obtains limited information through a set of observers, the green marks in the figure, and generates the "cyan" tracking results. Only 10 child particles are used in this experiment.

Fig. 2 illustrates the variety that 1 mother and 5 child particles can make. There are 32 combinations created some of which are very different from both mother and child particles. Fig. 3(a) shows tracking details during frames 7-10. The yellow window indicates the position where a new component is about to appear. The four small figures shows what happens inside the yellow window during frames 7-10. In particular, in frame 8 a cyan curve is generated close to the red (true) component because some measurement point reports that it is possible that a new boundary component has appeared there. Fig. 3(b) shows a connected component inside the yellow window splitting during frames 52-55. The tracking algorithm captures the topological change. Conversely, Fig. 3(c) shows two com-

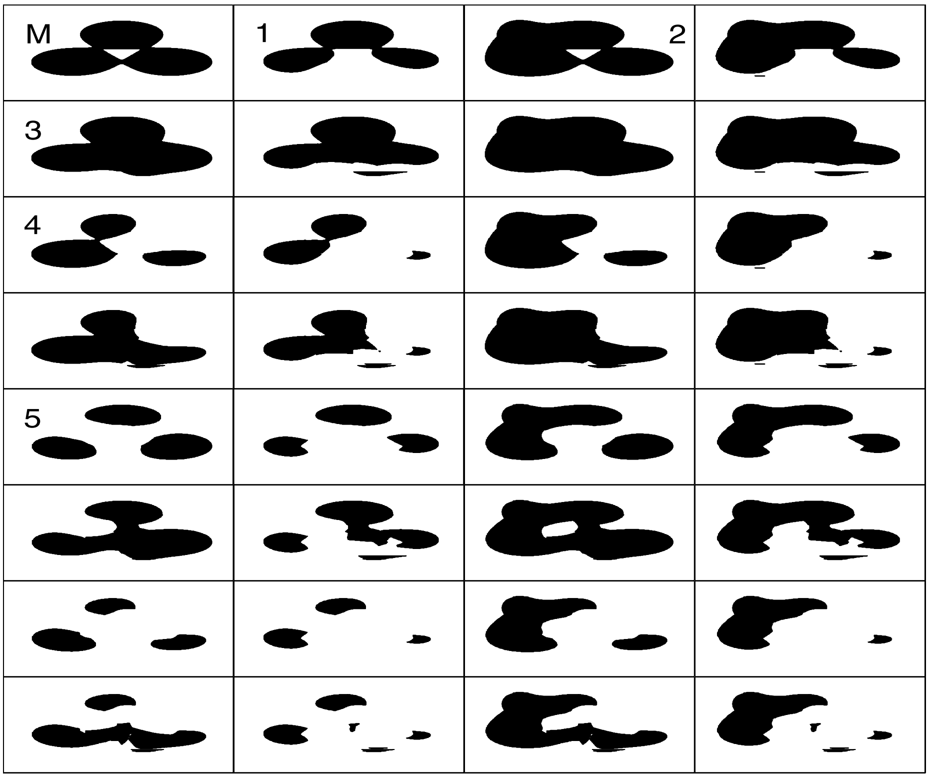

Fig. 2. A set of 32 "implicit" particles generated by one mother particle (top left) and 5 "child" particles (marked)

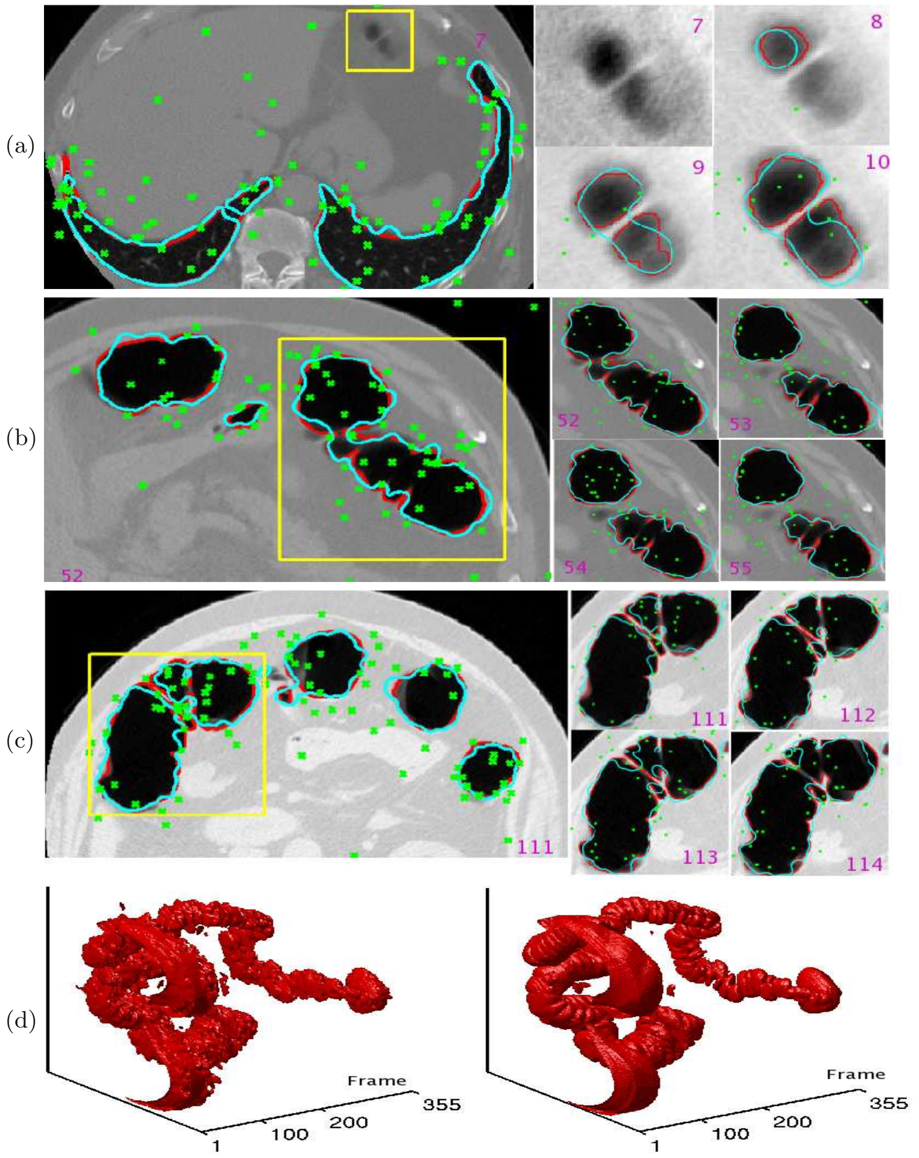

Fig. 3. (a) A new component appears in the yellow window (left figure) which is detected in frame 7 (top middle) and tracked from then on (see frames 8-10). Red is ground truth from edge detection, cyan is the maximum-likelihood particle. The cyan boundary will split in later frames (see figure (b)). The number inside each figure denotes the frame index. (b) The component in the yellow window splits in two between frames 52 and 55. (c) The two components in the yellow window merge into one between frames 111 and 114. (d) 3D reconstruction of the entire colon boundary. The left figure is based on the tracking result and the right is the ground truth.

ponents merging together during frames 111-114. The tracker, again, complies. Finally, Fig. 3(d) displays the 3D reconstruction from all the tracking results for 355 slices in the frame sequence and compares it to the ground truth obtained from edge detection. We use 100 measurement points in each slice, so the total number of measurements for the sequence is $100 \times 353 = 35,300$. The entire boundary has 363,069 points at pixel resolution, so the ratio between reconstructed points and measurements is about 10. The total tracking time for the sequence is 7121s, that is, about two hours for the sequence, or 20 seconds per frame. We are still far from real time performance. However, the code runs in Matlab and has several nested loops, so a substantial speedup is likely just by code optimization in C. Parallel implementation is of course trivially possible in obvious ways for particle filters. Now that preliminary experiments have shown the conceptual validity of our approach, we plan to turn some of our efforts to increasing efficiency through appropriate data structures and approximation algorithms as well.

5 Conclusions and Future Work

To our knowledge this is the first full-fledged formulation of particle filters for level sets. In our method, resampling has a cost proportional to the number of particles, but very small constant factors. The propagation cost is proportional to the *logarithm* of the number of particles, because P explicit particles represent 2^P particles implicitly. This is crucial, because the propagation cost per (explicit) particle is high for level sets. Preliminary experiments on tomographic imagery have shown the practicality of the approach by reconstructing a surface of $363,000$ points from only $35,300$ image measurements.

Immediate targets for future work are the improvement on the cost of maintaining the intersection factors for intersections windows and the use of multiple mother particles to represent macroscopic differences between particles. A multi-resolution hierarchy of overlapping particles is our ultimate goal in this respect. Now we are working on a finite-element perturbation strategy to make the tracking process more efficient. Incidentally, the approximation error for Finite Element Method is inherently better understood than that of mixture of Gaussian functions. Other plans for future work focus on the further reduction of constant factors and on the design of strategies for dispatching travelling observers as the boundaries move. Our experiments suggest in particular concentrating more observers close to high-curvature points on the boundary. This will make it possible, for instance, to send robots to autonomously track the moving boundary of an oil spill, wildfire, or cloud of pollutant. We intend to investigate the application of our methods to other domains as well.

References

1. Isard, M., Blake, A.: Condensation-conditional density propagation for visual tracking. IJCV **29** (1998) 5–28
2. Blake, A., Isard, M.: Active Contours. Springer, New York, NY (1999)

3. Einstein, A.: Zur Theorie der Brownschen Bewegung. Ann. d. Phys. **19** (1906) 180
4. Risken, H.: The Fokker-Planck Equation: Methods of Solution and Applications. Springer, New York, NY (1996)
5. Doucet, A., de Freitas, N., N. Gordon, editors: Sequential Monte Carlo in Practice. Springer, New York, NY (2001)
6. De Boor, C.: A Practical Introduction to Splines. Springer, New York, NY (2001)
7. Kalman, R.E.: A new approach to linear filtering and prediction problems. Trans. of the ASME J. on Basic Eng. **82** (1960) 34–45
8. Osher, S., Sethian, J.A.: Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. J. of Comp. Phys. **79** (1988) 12–49
9. Caselles, V., Morel, J.M., Sapiro, G., A. Tannenbaum, editors: Special issue on partial differential equations and geometrydriven diffusion in image processing and analysis. IEEE Trans. on Image Proc. **7** (1998) 269–473
10. Nielsen, M., Johansen, P., Olsen, O.F., J. Weickert, editors. In: Scale Space Theories in Computer Vision. Volume 1682. Springer, Berlin (1999)
11. Osher, S.J., Fedkiw, R.P.: Level Set Methods and Dynamic Implicit Surfaces. Springer, New York, NY (2002)
12. Adalsteinsson, D., Sethian, J.A.: A fast level set method for propagating interfaces. J. of Comp. Phys. **118** (1995) 269 – 277
13. Cremers, D., Soatto, S.: A pseudo-distance for shape priors in level set segmentation. 2nd IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision (2003) 169–176
14. Paragios, N., Deriche, R.: Geodesic active contours and level sets for the detection and tracking of moving objects. IEEE Trans. PAMI **22** (2000) 266–280
15. Zhang, T., Freedman, D.: Tracking objects using density matching and shape priors. ICCV **2** (2003) 1050–1062
16. Malladi, R., Sethian, J.A., Vemuri, B.C.: Shape modeling with front propagation: A level set approach. IEEE Trans. on PAMI **17** (1995) 158–175
17. Tsai, A., Yezzi, A., Wells, W. Jr., Tempany, C., Tucker, D., Fan, A., Grimson, W.E., Willsky, A.: A shape-based approach to the segmentation of medical imagery using level sets. IEEE Trans. on Med. Im. **22(2)** (2003) 137–154
18. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic active contours. IJCV **22** (1997) 61–79
19. Kichenassamy, S., Kumar, A., Olver, P., Tannenbaum, A., Yezzi, A.: Gradient flows and geometric active contour models. In: ICCV. (1995) 810–815
20. Bertalmio, M., Sapiro, G., Randall, G.: Morphing active contours. IEEE Trans. on PAMI **22** (2000) 733–737
21. Mansouri, A.: Region tracking via level set PDEs without motion computation. IEEE Trans. on PAMI **24** (2002) 947–961
22. Rathi, Y., Vaswani, N., Tannenbaum, A., Yezzi, A.: Particle filtering for geometric active contours with application to tracking moving and deforming objects. CVPR **2** (2005) 2–9
23. Yezzi, A.J., Soatto, S.: DEFORMOTION: Deforming motion, shape average and the joint registration and approximation of structures in images. IJCV **53(2)** (2003) 153–167