Deformable Object Tracking: A Kernel Density Estimation Approach Via Level Set Function Evolution

Nilanjan Ray and Baidya Nath Saha

Department of Computing Science
University of Alberta, Edmonton, Canada
{nray1,baidya}@cs.ualberta.ca

Abstract. Automated tracking of deformable objects that change shape and size drastically is challenging. For useful results, one needs an efficient deformable object model. In this regard, we propose a novel deformable object model via joint probability density of level set function and image intensity/feature values. Given the delineated object boundary on the first image frame of a video sequence, we learn the aforementioned joint probability density via kernel (Parzen window) method. From the next frame onward, we match this learned probability density with the probability density on the current frame by minimizing Kullback-Leibler divergence. This minimization procedure is cast in a variational framework and a minimizer is obtained by solving a partial differential equation (PDE). A stable and efficient numerical scheme is proposed for solving this resulting PDE. We demonstrate the efficacy of the proposed tracking method on myocardial border tracking from mouse heart cine magnetic resonance imagery (MRI).

Keywords: Kernel density estimation, Parzen window, KL divergence, level set, cine MRI.

1 Introduction

For deformable object tracking, we need an object model that has at the least the following capabilities: *Recognition capability* – the model should recognize the object from one video frame to the next and *discrimination capability* – it should accurately delineate the object boundary from the surrounding. The tracking affair is further complicated when the object being tracked *deforms heavily* from one video frame to the next. Many automated object tracking applications fall into such a problem category. An important application is myocardial border tracking from cine MRI [1].

If the objects to be tracked are not deformable, or, if accurate object boundary delineation is not required, then plenty of fixed template tracking methods can be exploited. Correlation based trackers are famous among them. More powerful methods include mean shift tracker that models the object by its intensity histogram [2].

On the other hand, if the deformation of the object is moderate then we can also employ many powerful deformable object models proposed to date (see [3] for a review of deformable models). Based on the number of training objects required, moderately deformable models can be categorized broadly into two types: (1) models that require more prior knowledge and fewer training objects, (2) models that require many training examples and less prior knowledge about the objects. Examples of the first kind include parametric contour models, such as, affine invariant g-snake [4]. For g-snake even a single example object might be adequate for modeling the object via affine transformation invariant parameters. Note that g-snake can tolerate large deformations, as long as the deformations belong to the family of shapes produced by affine transformations of a base object shape. Other examples in this category are contour models that can be described by geometric primitive shapes (such as circles, ellipses, etc.) parameterized by only a few scalar numbers. An example of the second category of moderately deformable models is principal component analysis (PCA)-based object models, such as active shape and appearance models [5]. PCA-based models use many training objects to learn the object characteristics. It is noteworthy that if the assumption of moderate object deformation holds, moderately deformable models as described above, have adequate recognition and discrimination abilities that can be utilized in object tracking.

Yet a third type of deformable object model is available that only imposes local smoothness/regularity on the object shape and is thus capable of undergoing very large and almost arbitrary deformations. Examples include classical Kass-Witkin-Terzopoulos snake model [6], spline snakes, as well as level set-based object models imposing only local regularization [7]. These large deformation models have seen much of their success in image segmentation because of their excellent ability in object boundary delineation. Typically these models lack the knowledge of object specific characteristics (features) and thus have poor recognition capability. In other words, they are suitable for unsupervised segmentation or clustering, and not as much effective in supervised tasks such as object tracking/recognition.

In this paper we propose a deformable object model that is, to a good extent, capable of undergoing large deformations as well as learning object and background features. As a result we can achieve aforementioned three desirable properties required in tracking, *viz.*, recognition, discrimination and large deformation. Our proposed deformable object model is a joint probability density function (pdf). Given an object boundary contour on the initial video frame, we compute the signed distance function (level set function). Next, the joint pdf of this level set function and image intensities (features) is constructed as the deformable object model. Note that this joint pdf is a function of object boundary. Thus in the subsequent video frames, we search for the right object boundary that results in a joint pdf similar to the joint pdf we learn on the first video frame.

In connection with the proposed deformable object model we mention two closely related previous work as follows. Leventon proposed an object model, which is a joint pdf of level set function (created from object boundary) and

image intensity at *every pixel location* in an image for the purpose of segmentation [8]. Note that his model is completely incapable of undergoing severe deformation as, every pixel (inside as well as outside the object) carries its own joint pdf, as opposed to the proposed model that is attached to an entire object. Also note that Leventon's model is not translation invariant - an indispensable property for target tracking. His method requires many training examples to learn this pixel location-based joint pdf and is unsuitable for tracking applications where the object only in the first video frame can be used for learning.

The other related previous work sits at the opposite extreme of Leventon's work in terms of deformation capability. Freedman and Zhang proposed to track the intensity (feature) histogram of an object delineated by a flexible contour [9]. Like the proposed method their method also searches for the best contour location by minimizing the dissimilarity between a model pdf and the current feature pdf. Note that Freedman and Zhang's object model is composed of image intensity/feature only within an object and it has no information of the background at all. In this paper, by experiments, we demonstrate that their model severely lacks object delineation ability.

In the light of the aforementioned discussion on related work, our proposed object model derives desirable properties for tracking from both Leventon's and Freedman and Zhang's models. The proposed object model learns the information about object boundary, object intensities (features), and its background intensity. The proposed model is also highly deformable in nature.

2 Background

This section describes two basic ingredients of our method, *viz.*, level set function and similarity/dissimilarity measures between pdfs.

2.1 Level Set Function

A closed contour on a 2D plane has essentially two representations - parametric and geometric. In the parametric representation the coordinates $(X(s), Y(s))$ of the points on the curve are expressed as functions of a scalar parameter s . This parameter could, for example, denote the length of the curve measured from a particular point on the curve. Geometric representation however avoids this explicit parametric description altogether. It represents the closed curve as an intersection of a plane and a surface.

The set of points on a closed curve can be conceived as the zero level set of the surface, *i.e.*, the curve is the iso-contour with zero height. We refer to this surface as level set function. Given a contour, the construction of a level set function $\phi(x, y)$ that embeds this contour as a zero level set can be obtained by the signed distance transform:

$$\phi(x, y) = \begin{cases} -\sqrt{(x - X(x, y))^2 + (y - Y(x, y))^2}, & \text{if } (x, y) \text{ is inside the object} \\ \sqrt{(x - X(x, y))^2 + (y - Y(x, y))^2}, & \text{otherwise,} \end{cases}$$

where $(X(x, y), Y(x, y))$ is the point on the curve nearest to the (x, y) point (pixel location) in the image domain. We adopt the convention that the signed distance function shown here is negative inside the object and positive outside the object. Thus the curve is the zero level set of this surface $\phi(x, y)$. Refer to [7] for further details on level sets and their applications in image analysis.

2.2 Similarity/Dissimilarity Measures for PDF

Another essential ingredient in the proposed tracking method is a similarity/dissimilarity measure between pdfs. In this paper we use the dissimilarity measure Kullback-Leibler divergence (KL-divergence) [10]:

$$KL(P, Q) = \int Q(z) \log\left(\frac{Q(z)}{P(z)}\right) dz,$$

where $P(z)$ and $Q(z)$ are two pdfs that are being compared. KL-divergence measures the dissimilarity between P and Q . KL-divergence is non-negative and is zero only when P and Q are equal. Larger dissimilarity between P and Q yields larger value of KL-divergence. There is another widely used similarity measure for pdfs called Bhattacharya coefficient [2]:

$$BC(P, Q) = \int \sqrt{Q(z)P(z)} dz,$$

Bhattacharya coefficient is a number between 0 and 1. It achieves a value of 1 when P and Q are equal. The more similar P and Q are, the closer is the value of the Bhattacharya coefficient to unity. Bhattacharya coefficient has been successfully utilized in mean-shift tracking method [2]. By utilizing Jensens inequality [10] we, however, show here that decrease in KL-divergence ends up incrementing Bhattacharya coefficient and we choose to use KL-divergence in the proposed tracking method.

Proposition 1. *Decreases in KL-divergence increases Bhattacharya coefficient.*

Proof: By Jensons inequality we can write the following:

$$\log(BC) = \log\left(\int \sqrt{Q(z)P(z)} dz\right) \geq \int Q(z) \log\left(\sqrt{\frac{P(z)}{Q(z)}}\right) dz = -\frac{1}{2}KL.$$

Thus we have: $BC \geq \exp(-\frac{1}{2}KL)$, which shows that decrease of KL-divergence KL (a non-negative number) increases the Bhattacharya coefficient BC .

3 Proposed Method

3.1 Proposed Deformable Object Model

As already mentioned in the Introduction section, the proposed deformable object model is the joint pdf of level set function and image intensity learned on the first frame of a video sequence where tracking begins. Let $Q(l, i)$ denote this joint pdf, where l and i respectively denote level set function value and image

intensity value. By means of Parzen window estimate using Gaussian kernel one can express Q as:

$$Q(l, i) = \frac{1}{C} \int \int \exp\left(-\frac{(\psi(x, y) - l)^2}{2\sigma_l^2}\right) \exp\left(-\frac{(J(x, y) - i)^2}{2\sigma_i^2}\right) dx dy,$$

where $\psi(x, y)$ and $J(x, y)$ are respectively the level set function and the image intensity on the first image frame. C is a normalization factor that forces $Q(l, i)$ to integrate to unity. σ_i and σ_l are standard deviations of the Gaussian kernels. It is assumed that the object boundary is provided on the first image frame, so the construction of $\psi(x, y)$ is performed by a signed distance transform from the object boundary. On the second frame (or subsequent frames) we construct a similar joint pdf $P(l, i)$ as follows:

$$P(l, i) = \frac{1}{C} \int \int \exp\left(-\frac{(\phi(x, y) - l)^2}{2\sigma_l^2}\right) \exp\left(-\frac{(I(x, y) - i)^2}{2\sigma_i^2}\right) dx dy,$$

where $\phi(x, y)$ and $I(x, y)$ are respectively the level set function and the image intensity on the second frame. Note that $\phi(x, y)$ is not known on the second frame and we want to compute $\phi(x, y)$ so that P and Q are as close as possible. Once we find out the desired $\phi(x, y)$, the zero level set of $\phi(x, y)$ would provide us with the object boundary on the second (or the subsequent) image frame.

3.2 Matching Object Models

To compute the desired object boundary on the second or any subsequent frame, we minimize the KL-divergence between P and Q to obtain the desired ϕ^* :

$$\phi^* = \arg \min_{\phi} \int \int Q(l, i) \log\left(\frac{Q(l, i)}{P(l, i; \phi)}\right) dl di = \arg \max_{\phi} \int \int Q(l, i) \log(P(l, i)) dl di.$$

The last equality follows because Q is not a function of $\phi(x, y)$ and only P is a function of $\phi(x, y)$, i.e., $P \equiv P(l, i; \phi)$. Thus the energy functional to maximize with respect to ϕ becomes:

$$E(\phi) = \int \int Q(l, i) \log(P(l, i; \phi)) dl di.$$

Applying calculus of variations [10] one can show that the gradient ascent partial differential equation (PDE) for the maximizing level set function becomes:

$$\frac{\partial \phi}{\partial t}(x, y) = - \int \int \frac{Q(l, i)}{P(l, i; \phi)} \frac{(\phi(x, y) - l)}{\sigma_l^2} \exp\left(-\frac{(\phi(x, y) - l)^2}{2\sigma_l^2}\right) \exp\left(-\frac{(I(x, y) - i)^2}{2\sigma_i^2}\right) dl di,$$

We can compactly express this PDE by using the convolution notation “ $\ast$ ”:

$$\frac{\partial \phi}{\partial t}(x, y) = -\left(\frac{Q}{P} \ast g_1\right)(\phi(x, y), I(x, y)),$$

where g_1 is a function defined as: $g_1(l, i) = \frac{l}{\sigma_l^2} \exp\left(-\frac{l^2}{2\sigma_l^2}\right) \exp\left(-\frac{i^2}{2\sigma_i^2}\right)$ and $\frac{Q}{P}$ is simply ratio of Q and P : $\frac{Q}{P}(l, i) = \frac{Q(l, i)}{P(l, i; \phi)}$.

3.3 Efficient and Stable Implementation

In order to solve the aforementioned PDE one can use an explicit discretization scheme [11] as follows:

$$\phi^{t+1}(x, y) = \phi^t(x, y) - (\delta t) \left(\frac{Q}{P} * g_1 \right) (\phi^t(x, y), I(x, y)),$$

where the t denotes the iteration number and δt denotes time step size. The explicit scheme is however not suitable for practical purposes as instability develops unless the time step size is very small, in which case the computation takes a long time to converge. Also the solution quality is often not acceptable. Fortunately we can work around this problem using a semi-implicit scheme [11] as follows:

$$\phi^{t+1}(x, y) = \frac{\phi^t(x, y) + (\delta t) \left(\frac{lQ}{P} * g \right) (\phi^t(x, y), I(x, y))}{1 + (\delta t) \left(\frac{Q}{P} * g \right) (\phi^t(x, y), I(x, y))},$$

where g is the Gaussian convolution kernel: $g(l, i) = \exp(-\frac{l^2}{2\sigma_l^2}) \exp(-\frac{i^2}{2\sigma_i^2})$ and $\frac{lQ}{P}$ is a function defined as: $\frac{lQ}{P}(l, i) = \frac{lQ(l, i)}{P(l, i; \phi)}$. Note that although one requires two convolutions in each iteration of the semi-implicit numerical scheme as opposed to one convolution in the explicit scheme, the convergence is many times faster in the former as there is no restriction on the time step size.

4 Results, Comparisons and Discussion

We first illustrate by a straightforward experiment that Kullback-Leibler flow of Freedman and Zhang [9] lacks the object discrimination capability, whereas the proposed method is superior in this respect. The leftmost image of Fig. 1 shows the first frame of a synthetic image sequence where the object is a homogeneous circle on a homogeneous background. The object is shown to be delineated by a boundary for which both Freeman and Zhang's method and the proposed method learn the pdf. Note however that Freedman and Zhang's method learns the intensity histogram within the object boundary, whereas the proposed method learns a joint pdf of level set function and image intensity both inside and outside the object boundary. The middle and the rightmost images of Fig. 1 show respectively the results by Freedman and Zhang and the proposed method on the second frame of the synthetic image sequence. Note that the object in the second frame has grown larger in size and Freedman and Zhang's method could not perceive this change because it can only keep track of what is happening inside the object, not the surrounding. The proposed method is however seen to correctly delineate the object boundary on the second frame even though the object has grown bigger in size.

Next, we apply the proposed method and another popular competing active contour method, *viz.*, gradient vector flow (GVF) snake method [12] on two cine MRI sequences—BSL and PK. The PK sequence shows much rapid and vigorous cardiac motions than the BSL sequence. Fig. 2 shows a few images from the PK

Fig. 1. From left: first image frame and delineated object; second image frame and the result by Freedman and Zhang’s method; result by the proposed method

Fig. 2. Tracking myocardial boundary by the proposed tracking method—a few cine MRI frames and delineated boundaries on them

Fig. 3. Comparison of segmentation metrics for GVF snake method and the proposed method on the BSL (left column) and the PK (right column) sequence

sequence where the proposed method is shown to be able track the mouse heart boundary.

Fig. 3 shows comparisons of GVF snake method and the proposed algorithm on two cine MRI sequences. Table 1 summarizes the results by reporting only the mean values of the performance metrics. In these experiments we used two performance metrics: (a) Pratt’s figure of merit (FOM) that is widely used for measuring the accuracy of edge detection [13], and (b) segmentation score, which we define as the ratio of intersection over the union of the segmented regions for ground truth segmented image and the segmentation produced by an automated algorithm. For ground truth generations we manually labeled all the

Table 1. Comparisons of mean performance metrics for GVF snake and the proposed method

	Pratt's FOM		Segmentation Score	
	BSL	PK	BSL	PK
GVF Snake Method	0.51	0.62	0.51	0.44
Proposed Method	0.88	0.77	0.74	0.79

frames of the two cine MRI sequences. There are two parameters in the proposed method, *viz.*, the two standard deviations of the Gaussian kernels. We took $\sigma_i = 1$ and $\sigma_l = 0.5$ by cross-validation based on Pratt's FOM. These comparisons demonstrate that the performances of the proposed method on both the rapid and the slow heart rate sequences are superior.

References

1. Janiczek, R., Ray, N., Acton, S.T., Roy, R.J., French, B.A., Epstein, F.H.: Markov chain Monte Carlo method for tracking myocardial borders. Computational Imaging III. In: Bouman, C.A., Miller, E.L. (eds.) Proceedings of the SPIE, vol. 5674, pp. 211–218 (2005)
2. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-Based Object Tracking. IEEE PAMI 25(5), 564–577 (2003)
3. Montagnat, T., Delingette, H., Ayache, N.: A review of deformable surfaces: topology, geometry and deformation. Image and Vision Computing 19(14), 1023–1040 (2001)
4. Lai, C., Chin, R.: Deformable Contours: Modeling and Extraction. IEEE PAMI 17(11), 1084–1090 (1995)
5. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE PAMI 23(6), 681–685 (2001)
6. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International J. of Comp. Vis. 1(4), 1405–1573 (1988)
7. Sethian, J.A.: Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Sciences. Cambridge University Press, Cambridge (1999)
8. Leventon, M.: Statistical Models for Medical Image Analysis. Ph.D. Thesis, MIT (2000)
9. Freedman, D., Zhang, T.: Active contours for tracking distributions. IEEE Trans. Image Proc. 13(4), 518–526 (2004)
10. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
11. Ames, W.F.: Numerical methods for partial differential equations. Academic, New York (1992)
12. Xu, C., Prince, J.L.: Snakes, Shapes, and Gradient Vector Flow. IEEE Transactions on Image Processing 7(3), 359–369 (1998)
13. Pratt, W.K.: Digital Image Processing. Wiley InterScience, Chichester (2002)