The Level Set Method for Image Segmentation

Hossam Hassan and Aly A. Farag
Computer Vision and Image Processing Lab
University of Louisville,
Louisville, KY 40292,
USA.

www.cvip.uofl.edu
Outlines

- Level Set Function Definition and Representation,
- Traditional Techniques for tracking interfaces,
- Numerical Implementation,
- Stability Restrictions,
- Narrow Band Level Sets,
- Shape Detection Example Using Level Sets,
- Segmentation Using Level Sets,
- Parameter Estimation for the Segmentation Model,
- Experiments for Synthetic and MRA Data,
- Algorithm Evaluation.
Level Set Function Representation

Invented to account for changing topology of curves

\[ Z = \Phi(x, y, t1) \]

Zero Level Set or \( Z=0 \) plane

\[ Z = \Phi(x, y, t2) \]

Resulting Curve (Evolving Front)

www.cvip.uofl.edu
• Topological changes in the evolving front are handled naturally.

• This representation allows merging and breaking of the as $t$ advances.
Embedding curve evolution in the level set method

- Curve Contracts with time $\Phi(x, y, t) = 0$

- Level Set Function changes with time $\frac{\partial \Phi}{\partial t} dt + \frac{\partial \Phi}{\partial x} dx + \frac{\partial \Phi}{\partial y} dy = 0$

  $\frac{\partial \Phi}{\partial t} + (\frac{\partial \Phi}{\partial x}, \frac{\partial \Phi}{\partial y}).(\frac{dx}{dt}, \frac{dy}{dt}) = 0$

- Fundamental Level Set Equation $\frac{\partial \Phi}{\partial t} + |\nabla \Phi| \frac{\nabla \Phi}{|\nabla \Phi|} = 0$

- To smooth out the high curvature regions $F = \pm 1 - \varepsilon k$

- How to Compute Curvature $k = \frac{\Phi_{xx} \Phi_y^2 - 2\Phi_x \Phi_y \Phi_{xy} + \Phi_{yy} \Phi_x^2}{(\Phi_x^2 + \Phi_y^2)^{3/2}}$

www.cvip.uofl.edu
Traditional Techniques for Tracking Interfaces (By Discrete Parameterization)

\[ k_i^n = 4 \frac{(y_{i+1}^n - 2y_i^n + y_{i-1}^n)(x_{i+1}^n - x_{i-1}^n) - (x_{i+1}^n - 2x_i^n + x_{i-1}^n)(y_{i+1}^n - y_{i-1}^n)}{((x_{i+1}^n - x_{i-1}^n)^2 + (y_{i+1}^n - y_{i-1}^n)^2)^{3/2}} \]

\[ F(k_i^n) = \pm 1 - \varepsilon k_i^n \]

\[ (x_i^{n+1}, y_i^{n+1}) = (x_i^n, y_i^n) + \Delta t F(k_i^n) \frac{(y_{i+1}^n - y_{i-1}^n, x_{i+1}^n - x_{i-1}^n)}{\sqrt{(y_{i+1}^n - y_{i-1}^n)^2 + (x_{i-1}^n - x_{i+1}^n)^2}} \]

- In the equations, \( i \) denotes point index,
- \( x \) and \( y \) are the coordinates, \( k \) is the curvature, and \( F \) is the speed,
- \( n \) denotes the time sample index, \( \Delta t \) is the time step and \( \varepsilon \) is a smoothing factor.
Notes on This Traditional Technique

• It may approach zero over zero yielding a very sensitive calculation.

• The computed curvature can change drastically from one particle to the next because of unavoidable errors in the position.

• For $\varepsilon = 0.25$ the exact solution is always smooth and differentiable.

• If the time step exceeds a certain limit then oscillations occur.

• Also oscillations can occur due to small errors, local variations in the velocity, approximations of the position.

• If you use smaller time step, you also have solution unbounded.

• If you had chosen a smaller $\varepsilon$, the trajectories would have come closer together and a smaller time step would be required for stability.
To Enhance The Propagation of The Interface

- Smooth the speed function.
- Redistribute the points according to arc-length or a related quantity.
- Add a filter to remove noise (oscillations) in the point positions as they develop.
Numerical Implementation

Using Taylor’ Series Expansion:

\[ \Phi(x, y, t + \Delta t) = \Phi(x, y, t) + \Delta t \frac{\partial \Phi}{\partial t} \]

\[ \Phi_x = \frac{\Phi(x + \Delta x, y, t) - \Phi(x - \Delta x, y, t)}{2 \Delta x} \]

\[ \Phi_y = \frac{\Phi(x, y + \Delta y, t) - \Phi(x, y - \Delta y, t)}{2 \Delta y} \]

\[ \Phi_{xx} = \frac{\Phi(x + 2\Delta x, y, t) - 2\Phi(x, y, t) + \Phi(x - 2\Delta x, y, t)}{(2\Delta x)^2} \]

\[ \Phi_{yy} = \frac{\Phi(x, y + 2\Delta y, t) - 2\Phi(x, y, t) + \Phi(x, y - 2\Delta y, t)}{(2\Delta y)^2} \]

\[ \Phi_{xy} = \frac{\Phi(x + \Delta x, y + \Delta y, t) - \Phi(x - \Delta x, y + \Delta y, t) - \Phi(x + \Delta x, y - \Delta y, t) + \Phi(x - \Delta x, y - \Delta y, t)}{4\Delta x \Delta y} \]

\[ |\nabla \Phi| = \sqrt{\Phi_x^2 + \Phi_y^2} \]
Stability and CFL Condition

It requires the front to across no more than one grid cell at each time step.

Basic Level Set Equation
\[ \Phi_t + H(\Phi_x, \Phi_y) = 0, \quad H = F \sqrt{\Phi_x^2 + \Phi_y^2} \]

CFL Restriction
\[ 1 \geq \Delta t \left( \left| \frac{\partial H}{\partial \Phi_x} \right| \cdot \frac{1}{\Delta x} + \left| \frac{\partial H}{\partial \Phi_y} \right| \cdot \frac{1}{\Delta y} \right) \]

F is independent on the level set function partial derivatives w.r.t x and y.

\[ \Delta t \leq \frac{\sqrt{\Phi_x^2 + \Phi_y^2}}{\left( \frac{|\Phi_x|}{\Delta x} + \frac{|\Phi_y|}{\Delta y} \right) F} \]

Best Value that Guarantees Stability
\[ \Delta t = \min \left\{ \frac{\sqrt{\Phi_x^2 + \Phi_y^2}}{\left( \frac{|\Phi_x|}{\Delta x} + \frac{|\Phi_y|}{\Delta y} \right) F} \right\} \]

Courant-Friedrichs-Levy (CFL restriction)

www.cvip.uofl.edu
Calculating Additional Quantities

- **Enclosed Area**
  \[ A = \iint_D H(\Phi) \, dx \, dy, \]

- **Length of Interface**
  \[ L = \iint_D \delta(\Phi) | \nabla \Phi | \, dx \, dy, \]

- **Mainly used to track the Interface.**

  \[ \delta_\alpha(\Phi) = (1 + \cos(\Phi \pi / \alpha)) / (2\alpha), |\Phi| \leq \alpha \]

  \[ H_\alpha(\Phi) = 0.5(1 + \frac{\Phi}{\alpha} + \frac{1}{\pi} \sin(\pi \Phi / \alpha)), |\Phi| \leq \alpha \]

  \[ H_\alpha(\Phi) = 1, |\Phi| > \alpha \]
Narrow Banding

• Points of the front are only the points of interest.

• The points (highlighted) are called the narrow band.

• The change of the level set function at these points only are considered.

• Other points (outside the narrow band) are called far away points and take large positive or large negative values.

• It speeds up the iterations.
Example: Shape Detection

- The goal now is to define a speed function coefficient (term) from the image data that acts as a halting criterion for this speed function.
- Define the speed function as follows:
  \[ F = \pm 1 - \varepsilon k \]
- Multiply the above speed function by the term:
  \[ gI(x, y) = \frac{1}{1 + |\nabla(G\sigma * I(x, y))|} \]
- Where \( G\sigma * I \) denotes the image convolved with a Gaussian smoothing filter whose characteristics width is \( \sigma \).
- \( |G\sigma * I| \) is essentially zero except where the image gradient changes rapidly in which case the value becomes larger.
- Thus the filter \( gI(x, y) \) is close to unity away from boundaries and drops to zero near sharp changes in the image gradient.
Example: Shape Detection (continued)

- The initial front can consist of many fronts; because of topological capabilities of the level set method, these fronts will merge into a single front as it grows into the particular shape.

- The front can follow intricate twists and turns in the desired boundary.

- The technique can be used to extract three dimensional shapes as well by initializing in a ball inside the desired region.

- Small isolated spots of noise where the image gradient changes substantially are ignored; the front propagates around these points breaks into two and the ring around the isolated spot closes back in on itself and then disappears.
We consider the following:

- The class $(\Omega_i)$ is a partition of image space $(\Omega)$.

- Gaussian distribution property of the classes which is minimum with respect to the class.

- The sum of the length of interfaces is minimum.

- $\Gamma_{ij}$ denotes the interface (front) between two classes (i and j).
Multi-Phase Function

- The level set function is defined as follows:
  \[ \Phi_i(x,t) > 0 \quad \Rightarrow \quad x \in \Omega_i \]
  \[ \Phi_i(x,t) = 0 \quad \Rightarrow \quad x \in \Gamma_i \]
  \[ \Phi_i(x,t) < 0 \quad \Rightarrow \quad \text{Otherwise} \]

- Use the Heaviside and delta functions to get the class partition and boundary.
PDE Description

- The partition condition ($F_1$)
- The data term condition ($F_2$)- (Gaussian property of the class)
- The length shortening of interfaces between regions ($F_3$)

- $F_1 + F_2 + F_3$ is minimized

\[
F_1 = \frac{\lambda}{2} \int_\Omega \sum_{i=1}^{K} (H_\alpha(\phi_i) - 1)^2 \, dx
\]

\[
F_2 = \sum_{i=1}^{K} e_i \int_\Omega H_\alpha(\phi_i) \left( \frac{u_o - u_i}{\partial_i^2} \right)^2 \, dx
\]

\[
F_3 = \sum_{i=1}^{K} \gamma_i \int_\Omega \delta_\alpha(\phi_i) |\nabla \Phi_i| \, dx
\]

\[
\Phi_i^{t+1} = \Phi_i^t - \Delta t \delta_\alpha(\Phi_i) \left[ e_i \frac{(u_o - u_i)^2}{\partial_i^2} - \gamma_i \text{div} \left( \frac{\nabla \Phi_i}{|\nabla \Phi_i|} \right) + \lambda \sum_{i=1}^{K} H_\alpha(\Phi_i) - 1 \right]
\]
Segmentation Algorithm

0- Fix the level set functions for \( i=1 \ldots K \)

1- \( t = t + 1 \)

2- For \( i=1 \ldots K \) solve the \( K \) coupled PDE's

3- Every \( n \) iteration re-initialize the level set function

4- Go to 1

Step 0 is the initialization of the level set functions which is very important. Bad initialization leads to a bad classification. To overcome this we use automatic seed initialization.
Why Re-Initialization?

- The level set function must have positive and negative parts to have a zero level set (Signed Distance Function).
- Non-Preserving of this property leads to solution divergence.
- The used model does not preserve this property.
- Use the following equation each n-iterations:

\[
\frac{\partial \Phi_i}{\partial t} = \text{sgn}(\Phi_i)(1 - |\nabla \Phi_i|)
\]

At Steady State

\[
0 = 1 - |\nabla \Phi_i| \quad \text{At Steady State}
\]

By M. Sussman, P. Smereka, and S. Osher

- When it’s -ve, the information flows one way and when it’s +ve, the information flows the other way.
- The net effect is to “straighten out” the level sets on either side of the zero level set. If done often enough, few iterations are required.
Automatic Seed Initialization

• It divides the image into non-overlapped windows.

• For each window, the average gray level is calculated.

• The average gray level is compared with each class mean to decide the nearest class. Then a signed distance function is initialized.

• This method speeds up the iterations.

• It’s less sensitive to noise.

• The window size needs to be smaller for more details.
Exp1: Synthetic Image

\[ K = 3, \alpha = 0.3, e_i = 2.0, \]
\[ \lambda_i = 2.0, \gamma_i = 0.4, \Delta t = 0.2, \]
\[ \mu_1 = 25.0, \mu_2 = 76.0, \mu_3 = 25.0, \]
\[ \sigma_1 = \sigma_2 = \sigma_3 = 1; \]
Exp2: Automatic Seed Initialization

www.cvip.uofl.edu
Exp3: MRA Results

Automatic Seed Initialization of slice #60

Slices #1, 30, 60, 117

Segmentation Results for Slices #1, 30, 60, 117

www.cvip.uofl.edu
Visualization

- Visualization of the two data sets after applying the connectivity process.
- The connectivity process makes sure that the tree is continuous and it removes the non-vessel areas.
Parameters Estimation

• The segmentation model is very sensitive to the parameters (α, e, γ, and λ).

• Using the equation at steady state we can get a relation between these parameters.

• From histogram analysis, we can get some points having a high probability to belong to a certain class.

• Points of high probability that belongs to class 1.

• Points of high probability that belongs to class 2.

• Steady State equation is as follows:

\[
\Delta t \delta_\alpha (\Phi_i) \left[ e_i \left( \frac{u_o - u_i}{\partial x} \right)^2 - \gamma_i \text{div} \left( \frac{\nabla \Phi_i}{\left| \nabla \Phi_i \right|} \right) + \lambda \left( \sum_{i=1}^{K} H_\alpha (\Phi_i) - 1 \right) \right] = 0
\]
Parameters Estimation (continued)

Using the points of high probability of each class, we can estimate the level set functions values using the definition of multi-phase function.

Now, we have N-points \((p_i)\) that satisfy the steady state equation creating the following matrix equation:

\[
\begin{bmatrix}
\left(\frac{u_o - u_i}{\partial_i^2}\right)_{p_1} & -\text{div}\left(\frac{\nabla \Phi_i}{\nabla \Phi_i}\right)_{p_1} & \left(\sum_{i=1}^{K} H_a(\Phi_i) - 1\right)_{p_1} \\
\left(\frac{u_o - u_i}{\partial_i^2}\right)_{p_2} & -\text{div}\left(\frac{\nabla \Phi_i}{\nabla \Phi_i}\right)_{p_2} & \left(\sum_{i=1}^{K} H_a(\Phi_i) - 1\right)_{p_2} \\
& \ddots & \ddots \\
\left(\frac{u_o - u_i}{\partial_i^2}\right)_{p_N} & -\text{div}\left(\frac{\nabla \Phi_i}{\nabla \Phi_i}\right)_{p_N} & \left(\sum_{i=1}^{K} H_a(\Phi_i) - 1\right)_{p_N}
\end{bmatrix}
\begin{bmatrix}
e \\
\gamma \\
\lambda
\end{bmatrix} =
\begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\]

Use \(\text{SVD}(C)\) to get the solution for the vector \(m\).

\(\alpha\) is a tunable parameter that determines the size of the bandwidth of numerical smearing.

A typical good value is \(\alpha=1.5\Delta x\) (\(\Delta x=1\)), making the interface width equal to 3 grid cells when \(\Phi\) is normalized to a signed distance function.

\(www.cvip.uofl.edu\)
Segmentation Quality Measurement

- A phantom is designed to simulate the MRA data.
- Evaluation Factor (Segmentation Accuracy $SA$)= 0.94

$SA = \frac{\text{# of Correctly Classified Pixels}}{\text{Total # of Pixels}} \times 100\%$
Conclusion

• We presented a supervised segmentation approach using level sets.

• A parameter estimation technique is proposed based on segmented data.

• Automatic seed initialization is used to initialize level set functions.

• The synthetic and MRA data results show that the approach is accurate and robust.

• The approach is validated using a phantom simulating the MRA data.
Future Work

- The results are promising but we will try to improve the segmentation model to enhance results using new techniques.
- Using non-linear techniques for the parameters estimation to get more accurate parameter values.
- Parallel Algorithms can be used to speed up the iterations.
- This model can be extended to multi-dimensional case.
Thank You