Level Set Models for Computer Graphics

David E. Breen
College of Computing & Informatics
Drexel University
Philadelphia, PA USA

Overview

- Level Set Model Introduction
- APPLICATIONS
  - 3D Volumetric Metamorphosis
  - Contour-based Surface Reconstruction
  - Volume Dataset Segmentation
  - Surface Editing
    - CSG Modeling
    - Handle-based
    - Sketch-based
    - Multiresolution

What is a Level Set Model?

- A numerical technique for tracking interfaces
- Level-set models are defined as an iso-surface, i.e. a level set, of a dynamic implicit function \( \phi \)

\[
S = \{ \vec{x} \mid \phi(\vec{x}, t) = k \}
\]

- \( k \in \mathbb{R} \) is the iso-value
- \( \vec{x} \in \mathbb{R}^n \) is a point in space on the iso-surface
- \( \phi : \mathbb{R}^n \rightarrow \mathbb{R} \) is a scalar function

What is a Level Set Model?

- A deformable implicit model, \( \phi(\vec{x}, t) = 0 \)
- Sampled representation of dimension \( n+1 \)
  - Images (2D) represent curves
  - Volumes (3D) represent surfaces
- Change level set by modifying samples
- Change sample values by solving a Partial Differential Equation (PDE)

This is a Level Set Model!

of a curve
Changing the image moves the curve
Red curve is defined by $\phi(x,y) = 127$

What is a Level Set Model?
- $\phi(x)$ is not defined by a specific equation
- $\phi(x)$ is represented by a regular 3D sampling
  - Signed distance volume dataset
- Level Set model is deformed by evolving the Level Set equation on the sampling
  - Osher & Sethian 1988
  $\partial \phi / \partial t = -\nabla \phi F(x, \partial \phi, \partial^2 \phi, ...)$
- Connects changes in sample values to changes of the level set curve/surface

The Speed Function
$$\frac{\partial \phi}{\partial t} = -\nabla \phi F(x, \partial \phi, \partial^2 \phi, ...)$$
- LS model deformation is controlled by $F(\ )$
- $F(x,...)$ defines the speed of the LS surface in direction of the surface normal $N$ at each point $x$ on the surface
- $F(\ )$ is defined by a user for each application in order to achieve a specific goal
- We have defined speed functions for several computer graphics applications

Advantages of LS Models
- Always produce closed, non-self-intersecting (simple) surfaces

Deforming Mesh May Intersect

No Self-Intersection with Level Set Deformations
Important for GCAD, Digital Manufacturing and Analysis

Figurine automatically manufactured from a level set model

Advantages of LS Models

- Easily change topological genus
  - Holes may close or open
  - Separate pieces come together/split apart
- Ideal for complex deformable models of unknown and/or changing genus

Mug-to-Chain Morph

Level Set Segmentation

Advantages of LS Models

- Concise interface for control → $F()$
- Free of mesh connectivity and quality issues
- No need to reparameterize or remesh during deformation

Disadvantages of LS Models

- No inherent parameterization (?)
- Computationally expensive (?)
  - Narrow Band methods are $O(\text{surface area})$
- High memory requirements (?)
- Cannot represent fine or sharp features (?)
- Cannot control genus (?)
  - Han et al. 2001, Bischoff & Kobbelt 2003, Ségonne et al. 2005
3D Volumetric Metamorphosis

Level Set Morphing

• Each point on surface moves in the direction of local normal. Step-size proportional to signed distance to target $\gamma_B$

\[
\frac{\partial \phi(x)}{\partial t} = \nabla \phi(x) \cdot \gamma_B(x)
\]

• Regions outside contract
• Regions inside expand
• Guaranteed convergence
• Not moving points!

Different Alignments Produce Different Morphs
Dart-to-Jet Morph

Morphing Challenge
- How to morph between these three models?
  - Polygonal Mesh
  - CSG Model
  - MRI Scan

Morphing Between Different Types of Models
- Combining
  - A variety of scan conversion algorithms
  - A flexible metamorphosis (morphing) technique based on level set models
- Produces
  - A technique for morphing between different types of geometric models

1 Minute of Fame
- Tar Monster Morphing Sequence
  Scooby-Doo 2, 2004

Contour-based Surface Reconstruction
**Problem Statement**

Create a smooth surface from parallel contours

**Level Set Approach**

- Sweep out the surface that connects the contours
  - Use 2D level set morphing
  - Map time into Z
  - Constrain the speed of the sweeping motion
  - Use a combination of Lagrangian and Eulerian techniques

**Visualizing the Speed Function**

Surface Reconstruction Via Contour Metamorphosis: An Eulerian Approach With Lagrangian Particle Tracking

- On Mission
- Social Issues
- teen Suicide

**Mouse Embryo Result**

**Volume Dataset Segmentation**

3D Models from Biomedical Scans

- How to extract specific objects from MRI, CT and other scans?
3D Models from Biomedical Scans

Level Set Segmentation
- Extract closed, smooth models from volumes
- Rough estimate
- Rough estimate + Fine tuning

Models from Scans
- Frog embryo MRI scan

Dataset provided by Caltech Biological Imaging Center
Models from Scans

- Frog embryo MRI scan
- Models of blastocoel and blastoporal lip

Models From Frog Embryo MRI Scan

- Models of blastocoel and blastoporal lip

Mouse Embryo Segmentation

- Ventricles
- Eye
- Liver

Diffusion Tensor Imaging of Human Brain

- Developed metrics that capture different diffusion properties

Visualizing Brain Ventricles

- Structures from Diffusion Tensor MRI, Isotropic Diffusion

Visualizing Brain White Matter

- Structures from Diffusion Tensor MRI, Anisotropic Diffusion
Ventricles and White Matter

Level Set Surface Editing
CSG Modeling

Initial Level Set Editing
- CSG operations
- Automatic blending
- Local/Global smoothing
- Embossing

Level-Set Editing Framework

Speed Function Building Blocks

Level-Set Blending
Distance Based cut-off function
Regionally constrains speed function
Geometric property filter function
Provides user control of local geometric properties
Function of geometric measure
Maps geometric properties to surface speeds
Laser scan reconstruction with spiky errors

Offset surface inwards

Offset surface outwards to create smoothed surface

Creating The Dragon

Important for GCAD, Digital Manufacturing and Analysis

Figurine automatically manufactured from a level set model

Level Set Surface Editing
Handle-based Operators
Single-Handle-Based Operators

- The free-form editing
  - Pulling on a point, symmetric ROI
  - Pulling on a point, arbitrary ROI
  - Pulling on a curve on the surface, symmetric ROI
  - Pulling on a curve on the surface, arbitrary ROI
  - Surface Detailing
  - Interactive Carving

Interactive smoothing

- Single-Handle-Based Operators

Pulling a point, symmetric ROI

- Click on a point on the surface \( x_s \) and drag it
- Radius of influence \( r \) around \( x_s \)
- Surface moves within sphere around \( x_s \)
- Simple yet useful for designing and creating handle-like structures

\[ F(x) = \frac{\cos^2(t/2) + d(x, r)/r}{\max(d(x, r))} \]

\( d(x, r) \): geodesic distance to \( x_s \) from \( x \)
\( r \): thickness of the ROI

Pulling on a point, arbitrary ROI

- A closed boundary curve on the surface designates an ROI
- All points in the ROI move outward as a function of the geodesic distance to \( x_s \)
- Stop when \( x_s \) reaches cursor or the user releases the mouse button.

\[ d(x, r) \]

Pulling on a curve on the surface, symmetric ROI

- Draw an open curve on the surface
- Click on a point on the curve and drag the point
- ROI is symmetric around the curve
- Speed decreases with the distance from the curve

\[ F(x) = \begin{cases} 
0 & d(x, r) > r \\
1 - \frac{d(x, r)}{r} & d(x, r) \leq r 
\end{cases} \]

\( d(x, r) \): geodesic distance between \( x \) and the curve

Pulling on a curve on the surface, arbitrary ROI

- Closed curve \( C_1 \) on the surface define an ROI
- Curve \( C_2 \) on the surface used as a handle.
- Clicking and dragging the handle deforms the surface

\[ F(x) = \left( \frac{\max(d(x, r)) - d(x, r)}{\max(d(x, r))} \right) \]

Changing alpha
Surface Detailing/Carving

- Superellipsoid-shaped tool
- Add or subtract features
- Adjustable tool size
- Can be used like an “eraser”

\[ P(x) = \begin{cases} 0 & f_{sc}(V) > 0 \\ 1 & f_{sc}(V) \leq 0 \end{cases} \]
\( V = x - x_c \)

Interactive Smoothing

- Click & drag tool over the area that needs smoothing
- Curvature-based speed function
- Fixed radius of influence
- Adjustable tool size and intensity

\[ P(x) = \gamma g(d(x)) + e(x) \]
\[ g(d) = \begin{cases} 1 & d \leq r - r' \\ \frac{1}{2} + \frac{1}{2} \cos ((d - r + r')/r) & r - r' < d \leq r \\ 0 & d > r \end{cases} \]

Level Set Surface Editing

Sketch-based Operators

- Local ROI
  - Single cross-section curve (1)
  - Multiple cross-section curves (2)
  - Multiple curve handles (3)
- No ROI
  - Multiple cross-section curves

Single Cross-section Curve

- \( d_{up}(x) \): closest point on \( C_s \)
- \( x_{up} \): corresponding point on \( C_d \)
- Distance is \( |x_{up} - x| \)
- Max functions taken over all points in the ROI

Interactive Smoothing

- \( \gamma \): constant that controls the amount of smoothing
- \( d(x) \): Euclidean distance from \( x \) to cursor location
- \( \kappa \): mean curvature
- \( r \): radius of influence
Multiple Cross-Section Curves

- The speed function is a combination of speed functions calculated per curve
  \[ F(x) = \sum \alpha_c(x) \cdot F_c(x) \]
  \( \alpha_c \) is the weight function that determines the effect of curve \( c \) on point \( x \).
- \( \alpha_c(x) \) decreases linearly/exponentially? with \( d(x) \)

Blending Functions

- Results with \( \alpha_c(x) \) shown below

Multiple Curve Handles

- All cross-section curves initially drawn on the surface
- Approximate 3D intersection points
- Curves stay connected
- Curve changes are propagated to intersecting curves via intersection points
- Use multiple curve speed functions to evolve surface

Global Deformations with Multiple Curves

- Create a 3D outline sketch of a model
- Initialize a simple level-set model inside (example: sphere)
- Deform the model until it reaches the sketch curves
  \[ F_c(x) = \frac{\max(\alpha_c(x) \cdot F_c(x))}{\max(\alpha_c(x) \cdot F_c(x))} \]
- \( F(x) = \sum \alpha_c(x) \cdot F_c(x) \)

Localized Catmull-Rom Splines

- We have developed techniques that expand and generalize the result of modifying one C-R control point
  - Provides greater flexibility, control and expressiveness
  - Allows the user to define the range and type of influence that manipulation of a single control point may produce on a C-R curve
  - Provides a versatile and powerful localized, multiresolution curve editing capability

Level Set Data Structures
General Framework

1. User input is translated into level-set speed functions
2. The level-set PDE is solved on a portion of the narrow band by the VISPACK library
3. The resulting edited model is displayed in the UI

VISPACK Narrow-Band Data Structures

- A set of doubly linked lists point into 3D array
- The order of voxels within each list is arbitrary
- Requires that level-set PDE be solved over the whole surface

Extended Narrow-Band Data Structures

- Four additional data structures
- Inside, Outside, Surface sublist
- 3D array of pointers to VISPACK elements
- Limit computation to a subset of the level-set surface
- C++ vectors of pointers point to entries within VISPACK linked lists
- Easy to access a set of voxels that are spatially contiguous
- Utilizes flood fill algorithm to identify ROI voxels
- Constant-time access to any narrow-band element

Voxel Storage

- Voxels stored in a 3D spatial hash table
- Many advantages
  - Sparse
  - Constant time random access
  - "Out-of-the-box" computing
- Disadvantage
  - No info away from the narrow band

Results: Handle-Based

Creating a bear
Results: Handle-Based
Creating a fantasy character

Results: Sketch-Based
Creating a duck and a shark

Results: Sketch-Based
Creating a shamrock and a bear

Results: Performance
FPS inversely proportional to # voxels in ROI

<table>
<thead>
<tr>
<th>Model</th>
<th>Resolution</th>
<th>fps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shark</td>
<td>85 x 99 x 54</td>
<td>66-90</td>
</tr>
<tr>
<td>Duck</td>
<td>79 x 67 x 37</td>
<td>66-90</td>
</tr>
<tr>
<td>Shamrock</td>
<td>45 x 50 x 35</td>
<td>83</td>
</tr>
</tbody>
</table>

Table 2: Editing details and running times for the final results. Speed is in frames-per-second (fps).
Results
High-Res Models

Dimensions: 654 × 600 × 1794
703,965,600 voxels in dense volume vs.
11,852,818 voxels in sparse volume

Results
High-Resolution Models

Open level set!

Dimensions: 3081 × 2057 × 69
437,295,573 voxels in dense volume vs.
44,512,586 voxels in sparse volume

Open level set!

Level Set Surface Editing
Multiresolution Operators

Hierarchical LS Model

• Smoothing & Differencing
• Details stored in particles
• Binomial filtering for smoothing
• LS evolution restores details
• Speed function based on info from particles

Hierarchical LS Model

LS Multiresolution Modeling

• Fast Marching Methods
• Blended using CBS

Multires Results
Geometric Texture Transfer

- Details particles can be transferred from one model to another to add texture

Summary

- Level Set Model Introduction
  - APPLICATIONS
    - 3D Volumetric Metamorphosis
    - Contour-based Surface Reconstruction
    - Volume Dataset Segmentation
    - Surface Editing
      - CSG Modeling
      - Handle-based
      - Sketch-based
      - Multiresolution

Papers


Acknowledgements

- Collaborators: Al Bar, Cindy Green, Ken Museth, R. Todd Whitaker, Leond Zlokhor
- Students: Manolya Eyiyurekli, Sean Mauch, Ola Nilsson
- This research was supported by the NSF STC for Graphics and Visualization, NSF grants ACI-99-82273 ACI-00-89915 and CCF-0702441, and the National Library of Medicine “Insight” Project #HL-0-3303.