Fast approximate surface evolution in arbitrary dimension

James Malcolm Yogesh Rathi Anthony Yezzi Allen Tannenbaum

Georgia Tech, Atlanta, GA
Brigham and Women's Hospital, Boston, MA

Motivation

Challenges of level set methods

- Numerics of implementation
- Floating point arithmetic expensive on embedded devices

Existing approaches

- Reduce the domain
- Use binary ϕ over the entire domain
- Use integer ϕ with minimal interface

Our contribution

- Simple, fast, and approximate surface evolution suitable for arbitrary energy

Level set segmentation

Overview

- Variational energy minimization:

$$E(\phi) = E_{data}(\phi) + E_{smooth}(\phi)$$
$$= \int_x \log(p_{in})(1 - H\phi) + \log(p_{out})H\phi dx + \int_x \delta\phi dx$$
$$\delta E(\phi) = \delta\phi \left(\log \frac{p_{out}}{p_{in}} + \text{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\right)$$

- Numerics required in maintaining distance function, differencing scheme, upwinding, interpolation, etc.

Discrete Approximation

1.4	0.8	0.7
0.7	-0.2	-0.3
-0.3	-1.3	-1.4

→

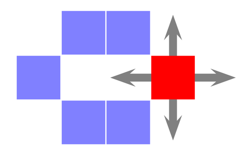
1	1	1
1	0	0
0	-1	-1

Assumption: Subpixel error matters little in most applications

Proposed method

Overview

- Maintain set of points along zero interface
- Use $\phi = -1$ for points inside, $\phi = 1$ for points outside, and $\phi = 0$ for points along zero interface
- Use sign of δE to move interface points in or out
- Simple rules for contraction and dilation of points by checking neighbors in any dimension
  - For 2D, points move in one of four directions
  - For 3D, points move in one of six directions



Algorithm

1. Compute force only along interface
2. Move interface points according to the force
3. Cleanup interface
4. Update regional statistics

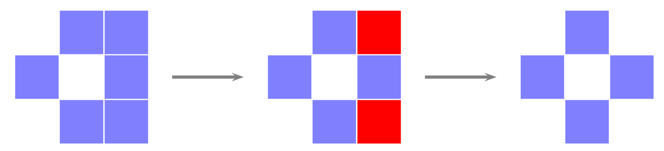
Pseudocode

for each iteration **do**
  {*Contraction*}
  Compute force: `computeforce()`
  Restrict to contraction (only allow positive forces)
  Move and cleanup
  Bookkeeping: `movein()`, `moveout()`
  {*Dilation*}
  Compute force: `computeforce()`
  Restrict to contraction (only allow negative forces)
  Move and cleanup
  Bookkeeping: `movein()`, `moveout()`
end for

Movement of points

Cleanup interface

Maintain minimal interface by dropping points that only touch one side of interface



Energy implementation

Arbitrary energies defined by three functions:

- `computeforce()` – compute scalar energy δE along surface
- `movein()`, `moveout()` – update regional statistics based on specified points moving across interface

Results

- White matter brain segmentation
- Speeds ranging from 0.8-50 ms per iteration, about 3s total
- Unit propagation greatly reduces number of iterations to convergence