Lecture 9:

Contour-based Segmentation

Segmentation

- ***Segmentation*** divides an image into meaningful sub-regions.
- Note segmentation is not the same as edge detection.
- Many applications
  - Medical Imaging: tumor detection, tissue volume measurement, locate features
  - Photography: Locate people and other objects, foreground / background separation
  - Remote Sensing: Locate buildings, vehicles, and regions in satellite images
- Segmentation is subjective.



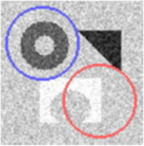
Segmentation Methods

- Thresholding
- Clustering (e.g. K-means)
- Region growing *(Shapiro-Stockman, 2001)*
- Graph partitions *(Swendsen-Wang, 1987)*, *(Shi-Malik, 2000)*, *(Grady-Schwartz, 2006)*
- Machine Learning *(Martin-Fowlkes-Malik, 2002)*

- Today we will focus on the Chan-Vese (CV) Active Countours Model, which is an example of a deformable model.

Deformable Models

- Segmenting an image pixel by pixel can lead to gaps and holes. It is also very sensitive to noise.
- It might be better to evolve a contour Γ that snaps around the object. This is called a deformable model.
- But how do we track the contour Γ?



Level Sets

- One way to track the curve Γ using a level set function *(Osher-Sethian, 1988)*.

- Let $\varphi\colon \Omega \rightarrow R$ be a smooth function whose zero crossing indicates the location of Γ.

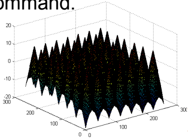
$$\vec{x}\ is\ inside\ \Gamma \Rightarrow \varphi(\vec{x}) > 0$$
$$\vec{x}\ is\ outside\ \Gamma \Rightarrow \varphi(\vec{x}) < 0$$
$$\Gamma = \{\vec{x}\colon \varphi(\vec{x}) = 0\}$$

- Level sets handle changes in topology (# segments).

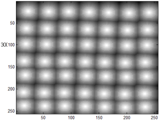
Level Sets

- In Matlab, we can draw the zero level set using the `contour` command.

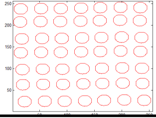
`surf(Phi);`



`imagesc(Phi);`



`contour(Phi,[0,0],'r');`



Mean Curvature Motion

- As a simple example of using level sets, let's shrink a curve using mean curvature motion.

$$\boxed{\frac{\partial \varphi}{\partial t} = \kappa\,|\nabla \varphi|}$$

- Substituting in the calculus formulas, we get

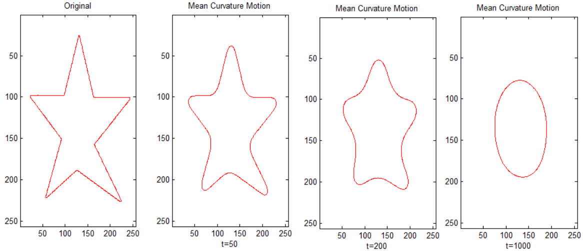
$$\frac{\partial \varphi}{\partial t} = \frac{\varphi_{xx}\varphi_y{}^2 - 2\varphi_x\varphi_y\varphi_{xy} + \varphi_{yy}\varphi_x{}^2}{\left(\varphi_x{}^2 + \varphi_y{}^2\right)^{3/2}}\left(\varphi_x{}^2 + \varphi_y{}^2\right)^{1/2}$$

$$= \frac{\varphi_{xx}\varphi_y{}^2 - 2\varphi_x\varphi_y\varphi_{xy} + \varphi_{yy}\varphi_x{}^2}{\varphi_x{}^2 + \varphi_y{}^2}$$

- Note this is very similar to our formula for TV.

Mean Curvature Motion

- The curve will shrink inwards to a circle, with the points of highest curvature moving fastest.
- This helps explain why TV rounds off corners.



Segmentation by Contours

- Let's assume the input grayscale image *f* is roughly piecewise-constant with 2 pieces to segment (phases).
- What would make a good segmenting contour Γ?

- We are looking for 3 things:
  1. Contour has minimal length: L(Γ)
  2. Color inside Γ should be roughly the same
  3. Color outside Γ should be roughly the same

Chan-Vese Active Contours Model

- This idea is called the Chan-Vese (CV) Active Contours Model *(Chan-Vese, 2001)*.

- The 2-phase Chan-Vese model is

$$\min_{\Gamma} E_{CV}[\Gamma|f] = L(\Gamma) + \lambda_{in} \int_{inside\ \Gamma} (f - c_{in})^2 d\vec{x} + \lambda_{out} \int_{outside\ \Gamma} (f - c_{out})^2 d\vec{x}$$

where $L(\Gamma)$ is the length of the curve Γ and c is the average gray value of f in each region

$$c_{in} = \frac{\int_{inside\ \Gamma} f\ d\vec{x}}{\int_{inside\ \Gamma}\ d\vec{x}} \qquad c_{out} = \frac{\int_{outside\ \Gamma} f\ d\vec{x}}{\int_{outside\ \Gamma}\ d\vec{x}}$$

The Heaviside Function

- Determine if we are inside or outside Γ using the Heaviside function.

$$H(t) = \begin{cases} 1 \; if \; t \geq 0 \\ 0 \; if \; t < 0 \end{cases}$$

$$\vec{x} \; is \; inside \; \Gamma \Rightarrow H(\varphi(\vec{x})) = 1$$

$$\vec{x} \; is \; outside \; \Gamma \Rightarrow H(\varphi(\vec{x})) = 0$$

- The derivative is the Dirac delta function.

$$\frac{dH}{dt} = \delta(t)$$

- The length of the edge set becomes

$$\boxed{L(\Gamma) = \int_{\Omega} |\nabla H(\varphi)| d\vec{x} = \int_{\Omega} \delta(\varphi) |\nabla \varphi| d\vec{x}}$$

Level Sets

- The CV model becomes

$$\min_{\varphi} E[\varphi|f] = \int_\Omega \delta(\varphi)|\nabla\varphi| d\vec{x}$$

$$+\lambda_{in} \int_\Omega (f - c_{in})^2 H(\varphi) d\vec{x} + \lambda_{out} \int_\Omega (f - c_{out})^2 (1 - H(\varphi)) d\vec{x}$$

$$c_{in} = \frac{\int_\Omega H(\varphi) f \, d\vec{x}}{\int_\Omega H(\varphi) \, d\vec{x}} \qquad c_{out} = \frac{\int_\Omega (1 - H(\varphi)) f \, d\vec{x}}{\int_\Omega (1 - H(\varphi)) \, d\vec{x}}$$

- The Euler-Lagrange equation is

$$\nabla E = \delta(\varphi)\left[-\nabla \cdot \left(\frac{\nabla\varphi}{|\nabla\varphi|}\right) + \lambda_{in}(f - c_{in})^2 - \lambda_{out}(f - c_{out})^2\right]$$

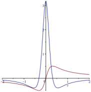
Steepest Descent

- Minimizing by steepest descent gives

$$\frac{\partial \varphi}{\partial t} = \delta(\varphi)\left[\nabla \cdot \left(\frac{\nabla \varphi}{|\nabla \varphi|}\right) - \lambda_{in}(f - c_{in})^2 + \lambda_{out}(f - c_{out})^2\right]$$

- Note the first term is also used in TV.
- For numerical implementation, we generally replace δ with a smooth approximation.

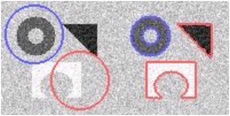
$$\delta_\varepsilon(t) = \frac{\varepsilon}{\pi(\varepsilon^2 + t^2)}$$

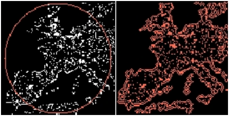


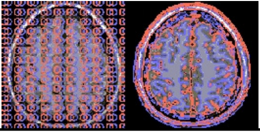
- Faster solvers exist, including multigrid methods *(Badshah-Chen, 2008)* , graph cuts *(Bae-Tai, 2009)*, and convex relaxation *(Brown-Chan-Bresson, 2010)*.

CV Segmentation

- The result depends on the initialization. Often recommended to use a checkerboard pattern for initial level set function.
- Note the segmentation does not require sharp edges at the boundary.



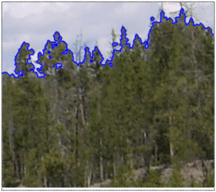




Color Images

- The CV model extends easily to color images *(Chan-Sandberg-Vese, 2000)*.

$$\min_{\Gamma} E_{CV}[\Gamma|f] = L(\Gamma) + \lambda_{in} \int_{inside\ \Gamma} \|f - c_{in}\|^2 d\vec{x} + \lambda_{out} \int_{outside\ \Gamma} \|f - c_{out}\|^2 d\vec{x}$$



Shape Priors

- If we know the shape of the object we are trying to find, we can add a term that forces the curve Γ to match our shape prior P.
- Slow because we also have to find a registration R *(Cremers-Osher-Soatto, 2004)*.

$$\min_{\Gamma,R} E_{CV}[\Gamma|f] = L(\Gamma)) + \lambda_{in} \int_{inside\ \Gamma} (f - c_{in})^2 d\vec{x} + \lambda_{out} \int_{outside\ \Gamma} (f - c_{out})^2 d\vec{x} + \mu d^2(R\Gamma, P)$$


Shape Prior


CV without prior


CV with prior

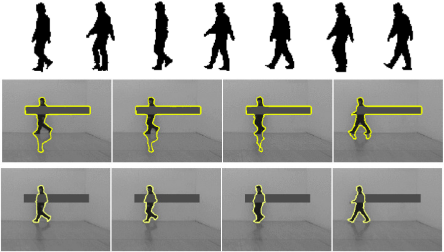
Shape Priors

- It is possible to use a library of shape priors and take the best match.

Shape Priors

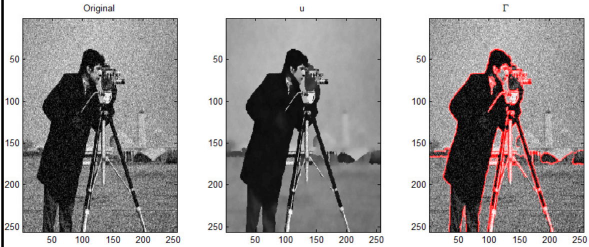
CV without prior

CV with priors



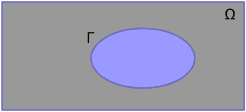
The Mumford-Shah Model

- An extension of the CV model is to performs both denoising and segmentation simultaneously *(Mumford-Shah, 1989).*



Energy with Known Edges

- Suppose the “important” edges of the images are given by some set Γ in the image domain Ω.

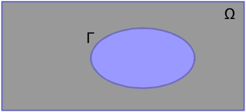


- Then an energy that roughly corresponds to anisotropic diffusion is

$$\min_u E[u|f,\Gamma] = \int_{\Omega\setminus\Gamma} |\nabla u|^2 d\vec{x} + \lambda \int_{\Omega} (u-f)^2 d\vec{x}$$

Energy with Known Image

- But how do we know the edge set Γ?



- If we were given a cartoon (piecewise-constant) image u, then we could find Γ by minimizing

$$\min_{\Gamma} E[\Gamma|f,u] = \int_{\Omega\setminus\Gamma} |\nabla u|^2 d\vec{x} + \alpha\, L(\Gamma)$$

where L(Γ) denotes the length of Γ.

- Minimizing the length prevents fractal-like edges or small islands around noise points.

Mumford-Shah Energy

- *(Mumford-Shah, 1989)* proposed a joint minimization over image u and edge set Γ:

$$\min_{u,\Gamma} E_{MS}[u,\Gamma|f\,] = \int_{\Omega\setminus\Gamma} |\nabla u|^2 d\vec{x} + \lambda \int_{\Omega} (u-f)^2 d\vec{x} + \alpha L(\Gamma)$$

Smoothness Fidelity Edge Length

- This is called a *free boundary problem* *(DeGiorgi, 1991)*.
- This is generally performed by an alternating minimization.
  - Fix Γ. Find u.

$$\min_{u} E[u|f,\Gamma] = \int_{\Omega\setminus\Gamma} |\nabla u|^2 d\vec{x} + \lambda \int_{\Omega} (u-f)^2 d\vec{x}$$

  - Fix u. Find Γ.

$$\min_{\Gamma} E[\Gamma|f,u] = \int_{\Omega\setminus\Gamma} |\nabla u|^2 d\vec{x} + \alpha\, L(\Gamma)$$

Existence of Γ

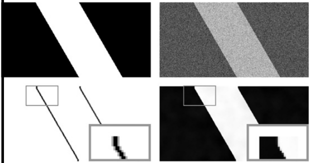
<u>Mumford-Shah Conjecture</u> *(Mumford-Shah, 1989)*

- The edge set Γ consists of the finite union of $C^{1,1}$ (smooth) arcs. Furthermore, every endpoint of a curve is one of the following.
  1. Curve intersects image boundary at $90°$ angle.
  2. Curve ends abruptly ("crack tip").
  3. 3 curves intersect at $120°$ angle ("triple junction").
- *(Bonnet, 1995)* proved the conjecture if we assume Γ consists of finite number of connected components.

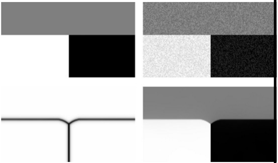
Mumford-Shah Conjecture

- The Mumford-Shah Conjecture can be seen numerically, but it is usually very subtle *(Vitti, 2012)*.

90° Angle at Boundary



Triple Junction



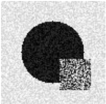
MS Inpainting

- Similar to TV, we can adapt Mumford-Shah to inpaint a damaged region D.

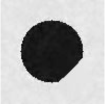
$$\min_{u,\Gamma} E_{MS}[u,\Gamma|f\,] = \int_{\Omega\backslash\Gamma} |\nabla u|^2 d\vec{x} + \lambda \int_{\Omega\backslash D} (u-f)^2 d\vec{x} + \alpha L(\Gamma)$$

- But the minimal length edge can give undesirable results in inpainting.
- *(Esedoglu-Shen, 2002)* propose adding a curvature term.

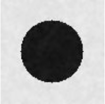
$$\min_{u,\Gamma} E_{MS}[u,\Gamma|f\,] = \int_{\Omega\backslash\Gamma} |\nabla u|^2 d\vec{x} + \lambda \int_{\Omega\backslash D} (u-f)^2 d\vec{x} + \alpha L(\Gamma) + \beta \int_{\Gamma} \kappa^2 ds$$



Original



MS Inpainting



MS + Curvature