35T

Snakes or Active Contours : Kass, Witkin, Terzopoulos '87

Idea: ① Represent initial contour by a set of N.
uniformly spaced points. (either uniformly sampled on contour arclength or use B-spline control points)

B spline control points: smooth the "signal" before subsampling. here "signal" is $V(s) = \begin{bmatrix} x(s) \\ y(s) \end{bmatrix}$

"initial snake" $\begin{bmatrix} V(s_1) \\ V(s_2) \\ \vdots \\ V(s_N) \end{bmatrix} \triangleq V(s)$

② Define an energy functional

$$E(V(s)) = \int_{s=0}^{L} \mathcal{L}(V(s))\, ds.$$

Treating $v(s)$ as a continuous function of arclength s (or some other "parameter")

"E" is a function of a function of s.
i.e. E is a functional.

e.g. $\mathcal{L}(V(s)) = \mathcal{L}(V(s), V_s(s), V_{ss}(s)) = \alpha V_s^2 + \beta V_{ss}^2 \underbrace{- |\nabla I(V)|^2}_{E_{image}}$

$V_s(s) \triangleq \frac{\partial V}{\partial s}(s)$

$V_{ss}(s) \triangleq \frac{\partial V}{\partial s^2}(s)$

$$E = \int_0^L \mathcal{L}(v, v_s, v_{ss}) = \int_0^L \Big[\underbrace{\alpha \|v_s\| + \beta \|v_{ss}\|^2}_{\substack{E_{internal} \\ \text{(or smoothness term)}}} \underbrace{- \|\nabla I(v)\|^2}_{E_{image}} \Big] ds.$$

$$E = \underset{(\text{or } E_{smoothness})}{E_{internal}} + E_{image}.$$

$E_{internal}$ = total arclength + bending energy

$\alpha \int \|v_s\| ds$ $\qquad$ $\alpha \int \|v_s(s)\|^2 ds$. $\qquad$ $\beta \int \|v_{ss}(s)\|^2 ds$

③ Minimize E by gradient descent or directly.

(find a contour $\underset{\sim}{v}(s)$ that has small length, small bending & maximum image gradient magnitude

$$\nabla_v E = ?$$

2 steps ① find $\nabla_v E$: use Calculus of Variations

② solve $\boxed{\nabla_v E = 0}$ directly or use gradient descent

Euler Lagrange eq

$$v^{j+1}(s) = v^j(s) - \gamma \cdot \nabla_v E(v^j(s))$$

can write as a Diff Eq with artificial time "t"

$$\frac{dv}{dt} = -\nabla_v E(v)$$

stop when $\frac{dv}{dt} \approx 0$. (ie. $\nabla_v E = 0$)

Everything until now assumed $v(s)$ is a continuous function of s.

Now, to solve the DE, we discretize $v(s)$ as $v(s_1), v(s_2) \ldots v(s_n)$ & time as $j = 1, 2, 3, \ldots$

$dt \approx \gamma$.

$$\rightarrow \quad v^{j+1}(s_k) = v^j(s_k) - \gamma (\nabla_v E) \begin{pmatrix} v^j(s_1) \\ v^j(s^2) \\ v^j(s_N) \end{pmatrix}$$

→ Do for all $k = 1, 2, \ldots N$.

→ iterate $j \leftarrow j+1$ until E does not change.

→ Output is a local minimizer of $\underline{E}$.

<u>Computing $\nabla_v E$</u> : Calculus <u>of variations</u>

→ See handout on calculus of variations.

When
$$E = \int_0^L \mathcal{L}(v, v_s, v_{ss})\, ds.$$

$$\nabla_v E = \frac{\partial \mathcal{L}}{\partial v} - \frac{\partial}{\partial s}\left(\frac{\partial \mathcal{L}}{\partial v_s}\right) + \frac{\partial^2}{\partial s^2}\left(\frac{\partial \mathcal{L}}{\partial v_{ss}}\right)$$

$$v_s \triangleq \frac{\partial v}{\partial s}$$

$$v_{ss} \triangleq \frac{\partial^2 v}{\partial s^2}$$

One common choice of energy is

$$E(v) = \int_0^L \mathcal{L}(v, v_s, v_{ss})\, ds.$$

$$\mathcal{L}(v, v_s, v_{ss}) = \alpha \cdot \|v_s^{(s)}\|^2 + \beta \|v_{ss}^{(s)}\|^2 - \|\nabla I(v(s))\|^2$$

$$\frac{\partial \mathcal{L}}{\partial v} = \frac{\partial}{\partial v}\left[-\|\nabla I(v)\|^2\right] = (-)2(\nabla^2 I)(\nabla I)$$

$$\frac{\partial}{\partial s}\left(\frac{\partial \mathcal{L}}{\partial v_s}\right) = \frac{\partial}{\partial s}(2\alpha v_s) = 2\alpha v_{ss}$$

$$\frac{\partial^2}{\partial s^2}\left(\frac{\partial \mathcal{L}}{\partial v_{ss}}\right) = \frac{\partial^2}{\partial s^2}(2\beta v_{ss}) = 2\beta v_{ssss}$$

$$\nabla_v E = -2\alpha v_{ss} + 2\beta v_{ssss} + \frac{\partial}{\partial v}\left[-\|\nabla I(v)\|^2\right]$$

Problems

① $\frac{\partial}{\partial v}(-\|\nabla I\|^2)$ is non zero only in the vicinity of an edge: cannot pull contour if initialized very far away.

Fix 1: Smooth image before taking gradient

Fix 2: Gradient Vector Flow Snake (Xu & Prince)

② Initially start with uniformly sampling arclength. As iterate, points are no longer uniformly sampled. Also, total contour length may increase or decrease.

Fix 1 : Snake growing or Topologically Adaptive Snakes or reparameterize at every iter

Fix 2 : Use Level Set Representation of contour.

— — — —

other energy functionals : "region-based"

$$E(v, \mu_i, \mu_o) = \int_{v_{inside}} (I(x,y) - \mu_i)^2 \, dx\,dy + \int_{v_{outside}} (I - \mu_o)^2 \, dx\,dy$$

$$+ \alpha \int_s \| v_s \| \, ds.$$

$\boxed{\phi^0}$

Can show that Alternating Minimiz algo is

$I > .5$ → inside move contour outside

① $\nabla_v E = \overset{(-)(\mu_i - \mu_o)}{\lambda} \left[I(v) - \frac{(\mu_i + \mu_o)}{2} \right] \vec{N} + \alpha \, \kappa \, \vec{N}$

.5

Solve $\frac{\partial v}{\partial t} = -\nabla_v E$ till convergence

outward ~~direction~~ ~~inward~~

$\vec{N}$ = normal to contour

κ = curvature

&

② $$\mu_i = \frac{\int_{v_{inside}} I(x,y) \, dx\,dy}{\int_{v_{inside}} dx\,dy}$$

$\frac{\partial \phi}{\partial t} = -\nabla_\phi E$

$\phi^{t+1} = \phi^t - \nabla_\phi E$

$\mu_i > \mu_o$

$I > .5$ then move contour out

$$\mu_o = \frac{\int_{v_{outside}} I(x,y) \, dx\,dy}{\int_{v_{outside}} dx\,dy}$$

③ Go back to step ①

Another form of Alt min is :

perform only one (or a few) iterations of Gradient Descent at each step

<idea: just to reduce value at each step>.

Another one

~~$E(V, \mu_i, \mu_o) =$~~

$$E(V) = -(\mu_i - \mu_o)^2$$

$$\mu_i = \frac{\int_{V_{inside}} I(x,y)\, dx\, dy}{\int_{V_{inside}} dx\, dy}.$$

$$\mu_o = \frac{\int_{V_{outside}} I(x,y)\, dx\, dy}{\int_{V_{outside}} dx\, dy}$$

Another : Geodesic Active Contour

TO DO
1. Calculus of variations + implement spatial derivatives
2. GVF
3. Snake growing
4. Level set · Method
5. Re-explain region based energy
6. Region growing segmentation.