Prog2: Active Contour

Zhongyan QIU UFID: 96962096

1. Energy Term

To find the accurate contour, we use two energy term.

The internal energy term is elastic energy term to control the contour length. Using the arc-length parameterization, s is the curve length, $s \in [0, L]$. Then we make the elastic energy term to be $E_{\text{int}} = \oint_c |C'(s)| ds$, which is trying to minimize the curve length.

The external energy term is the edge alignment term, which tries to force the curve to bend the same as the true object boundary. However, the true object boundary is somehow difficult to model and represent mathematically and actually finding the boundary itself is the goal we want to achieve. Fortunately, the Gradient Vector Flow (GVF) provides a possible way we can use to force our curve stop at the target object boundary. In this way, we minimize the inner product of gradient vector flow and the normal of the curve vector flow, which is

$$E_{ext} = \oint_c < V, n(s) > ds .$$

Hence, our energy functional is:

$$E_{our} = E_{\text{int}} + E_{ext} = \oint_c \alpha(s) | C'(s) | ds + \oint_c \beta(s) < V, n(s) > ds$$

$$= \oint_c \alpha(p) | \frac{c'(p)}{|c'(p)|} || c'(p) | dp + \oint_c \beta(p) < (u,v), (\frac{-y'(p), x'(p)}{|c'(p)|}) >| c'(p) | dp$$

$$= \oint_c \alpha(p) | c'(p) | dp + \oint_c \beta(p)(v \cdot x'(p) - u \cdot y'(p))dp$$

In which $\alpha(p), \beta(p)$ are the control parameters.

2. Optimization and Implementation

We can get two independent Euler-Lagrange Equations:

$$\begin{cases} \alpha \cdot \dfrac{-y_p(y_{pp}x_p - x_{pp}y_p)}{|c_p|^3} - \beta \cdot y_p(u_x + v_y) = 0 \\ \alpha \cdot \dfrac{x_p(y_{pp}x_p - x_{pp}y_p)}{|c_p|^3} + \beta \cdot x_p(u_x + v_y) = 0 \end{cases}$$

Which can be simplify as:

$$\frac{\partial E}{\partial c} = \alpha \cdot k \cdot \vec{N} + \beta \cdot div(V) \cdot \vec{N}$$

in which, k is the curvature of the curve.

Using gradient-descent minimization and numerical methods, we get:

$$\gamma(c_{t+1} - c_t) = \alpha \cdot k \cdot \vec{N} + \beta \cdot div(V) \cdot \vec{N}$$

$$\Rightarrow \begin{cases} \gamma(x_i^{t+1} - x_i^t) = \alpha \cdot \dfrac{-y_p^t(y_{pp}^t x_p - x_{pp}y_p)}{(x_p^2 + y_p^2)^{\frac{3}{2}}} - \beta \cdot y_p^t(u_x + v_y) \\ \gamma(y_i^{t+1} - y_i^t) = \alpha \cdot \dfrac{x_p^t(y_{pp}^t x_p - x_{pp}y_p)}{(x_p^2 + y_p^2)^{\frac{3}{2}}} + \beta \cdot x_p^t(u_x + v_y) \end{cases}$$

$$\Rightarrow \begin{cases} x_i^{t+1} = \dfrac{1}{\gamma}(\gamma \cdot x_i^t + \alpha \cdot \dfrac{-y_p^t(y_{pp}^t x_p - x_{pp}y_p)}{(x_p^2 + y_p^2)^{\frac{3}{2}}} - \beta \cdot y_p^t(u_x + v_y)) \\ y_i^{t+1} = \dfrac{1}{\gamma}(\gamma \cdot y_i^t + \alpha \cdot \dfrac{x_p^t(y_{pp}^t x_p - x_{pp}y_p)}{(x_p^2 + y_p^2)^{\frac{3}{2}}} + \beta \cdot x_p^t(u_x + v_y)) \end{cases}$$

in which, γ is the step size, and x_p^t and x_{pp}^t can be approximated using finite difference, which means: $x_p^t = x_i^t - x_{i-1}^t$ and $x_{pp}^t = x_{i-1}^t - 2x_i^t + x_{i+1}^t$, and the same as the y-derivatives.

3. Gradient Vector Field

To achieve the smoothing gradient vector field, we use the methods exactly as the [1] do. Setting the vector field as: $V(x,y)=[u(x,y),v(x,y)]$, then we minimize the energy functional:

$$\varepsilon=\iint \mu(u_x^2+u_y^2+v_x^2+v_y^2)+|\nabla f|^2|V-\nabla f|^2\,dxdy$$

Which forces the vector field exactly as the image gradient when the gradient is big enough, while when the gradient is not big enough, we smooth the vector field by minimize the summation of square of first derivatives.

Then the Euler Lagrange Equations will be:

$$\begin{cases}\dfrac{\partial\varepsilon}{\partial u}=F_u-\dfrac{d}{dx}F_{ux}-\dfrac{d}{dy}F_{uy}=0\\[2mm] \dfrac{\partial\varepsilon}{\partial v}=F_v-\dfrac{d}{dx}F_{vx}-\dfrac{d}{dy}F_{vy}=0\end{cases}$$

$$\Rightarrow\begin{cases}(\nabla f)^2(u-f_x)-\mu(u_{xx}+u_{yy})=0\\ (\nabla f)^2(v-f_x)-\mu(v_{xx}+v_{yy})=0\end{cases}$$

Taking gradient decent algorithm and numerical methods, we get:

$$\begin{cases}u^{t+1}(x,y)=u^t(x,y)+\mu\Delta u-(f_x^2+f_y^2)(u-f_x)\\ v^{t+1}(x,y)=v^t(x,y)+\mu\Delta v-(f_x^2+f_y^2)(v-f_y)\end{cases}$$

4. Results and Comparison

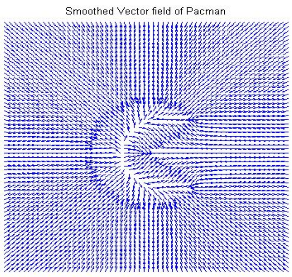
The parameters we use are:

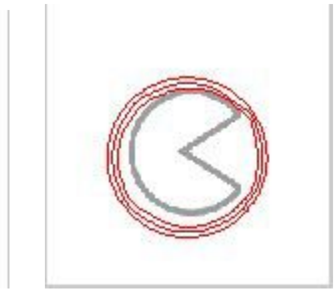
$\alpha=0.2$, $\beta=1$, $\gamma=0.5$, $\mu=0.2$.

A. Our results:

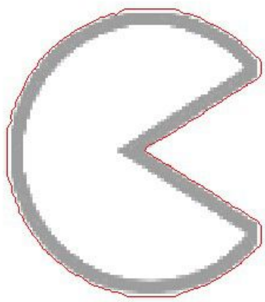
1. For pacman image:

Smoothed vector field & Initialized contour & Two intermediate stages of the contour:



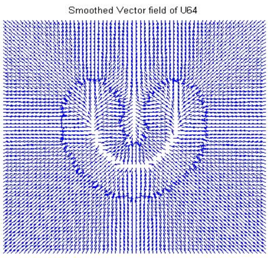


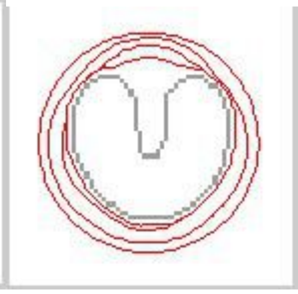
The final achieved result:

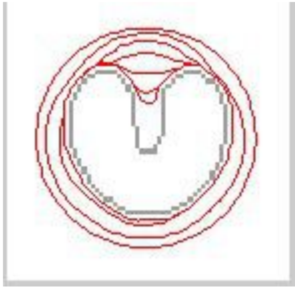


2. For U64 image:

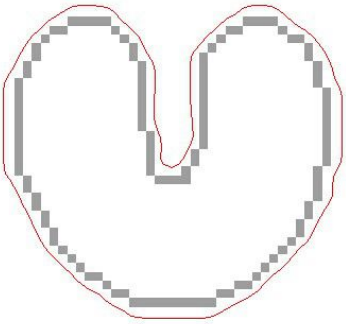
Smoothed vector field &Initialized contour & Two intermediate stages of the contour:







The final achieved result:

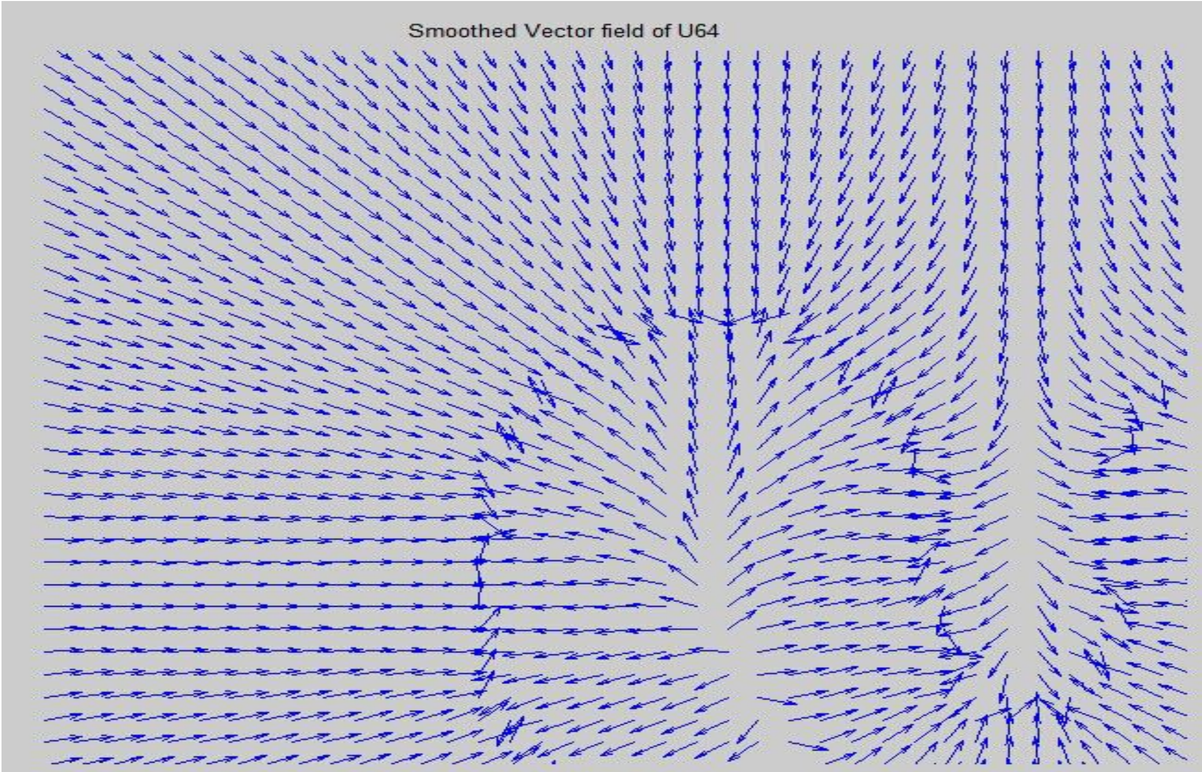


B. GVF Results:

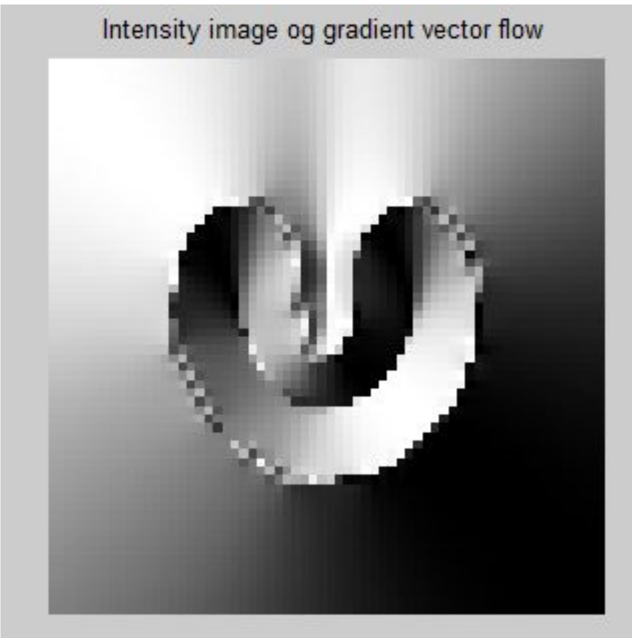
For pacman:

For U64:

C. Findings and Analysis



The Contour found by our methods is not sticky exactly on the contour but sticky outer on the object boundary comparing with results of GVF. That is because the external energy we use is alignment energy, which forces the curve vector flow to be at the same direction. But in the gradient vector field we produce (As the upper partial vector Flow image show), all the vectors out of the object contour point inside while all the vectors inside the object contour point outside. And at the boundary, there are vectors point inside and point outside mixed. Considering our edge alignment equation: $\oint_c < v, n > ds$, which calculates the inner product of vector flow and normal of the contour. Hence when the contour is at the very outside of the object boundary, to minimize the alignment energy is to force the contour flow to go inside (Pay attention that the normal vector point outside), while if the initial contour is at the very inside of the object boundary, the alignment energy will force the contour flow to go outside. But when the active contour is near the object boundary, it will stop because there are gradient vectors pointing both outside and inside, which means the alignment energy does not work. Hence, the curve cannot be enforced to shrink to be sticky on the object boundary but it will stop at somewhere out of the boundary keeping bending the same with object boundary.



However, the external energy used in the GVF Snake is the gradient vector flow which has great values at both sides out of boundary but has small values at the position of the boundary (Please see the Intensity Image of the gradient vector flow), in this way the active contour will stop at the position of the boundary.

5. Reference:

[1] Chenyang Xu, Jerry L. Prince, “Snakes, Shapes, and Gradient Vector Flow,” IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 7, NO. 3, MARCH 1998