FAST AND ROBUST ACTIVE CONTOURS FOR IMAGE SEGMENTATION

Wei Yu[1], Franz Franchetti[1], Yao-Jen Chang[2], Tsuhan Chen[2]

[1]Carnegie Mellon University, [2] School of Electrical Computer and Engineering, Cornell University

ABSTRACT

Active models are widely used in applications like image segmentation and tracking. Region-based active models are known for robustness to weak edges and high computational complexity. We found previous region-based models can easily get stuck in local minimums if initialization is far from the true object boundary. This is caused by an inherent ambiguity in evolution direction of the level set function when minimizing the energy. To solve this problem, we propose an intensity re-weighting (IR) model to bias the evolution process in certain direction. IR model can effectively avoid local minimums and enable much faster convergence of the evolution process. The proposed method is applied to both real and synthetic images with promising results.

Index Terms— active contours, level set, image segmentation

1. INTRODUCTION

Active contour models are widely used in image segmentation problems, especially for medical images with a lot of noise and intensity inhomogeneity. State-of-the-art active contours [1, 2, 3, 4] are implemented using the level set method [5] in Eulerian framework. The level set method has several advantages such as regular computation on a grid and being easy to handle topological changes.

Existing active contour models can be roughly categorized into edge-based models and region-based models. Generally speaking, region-based models are more robust when weak edges exist and less sensitive to initialization, but more computationally expensive than edge-based models.

In this paper, we choose LBF (local binary fitting) model [4] as our baseline, which represents state-of-the-art of region-based active contour models. Earlier region-based models either lack the ability to handle image inhomogeneity like the PC (piecewise constant) model [2], or too expensive in computation like the PS (piecewise smooth) model [3]. The LBF model minimizes an energy term which encourages smooth intensity variation locally within each region and tolerates abrupt intensity change along region boundary. We focus on the 2-region segmentation problem (foreground/background) in this paper.

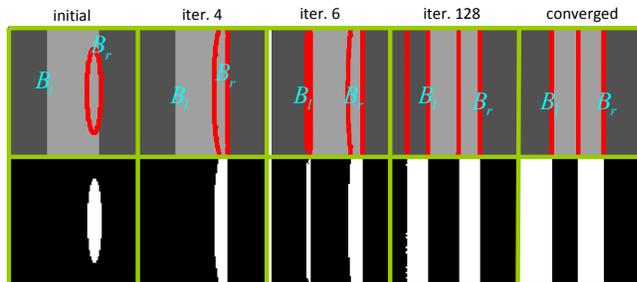**Motivation.** LBF model can easily get stuck in local

Fig. 1. An example for which LBF converges to local minimums. First row shows contour evolution, second row illustrates areas where $\phi > 0$ (white) and $\phi < 0$ (black). B_l and B_r marked true object boundary discussed in the text. (view in color)

minimums for some initializations. Fig. 1 shows such an example. In the following, R_1, R_2 refer to the brighter and darker region in an image respectively. S_1, S_2 refer to two segmented areas. ϕ is the level set function. S_1 is the segment of $\phi > 0$, and S_2 is the segment of $\phi < 0$. The goal of LBF model is to evolve ϕ such that ϕ has different signs in R_1 and R_2 to minimize energy. In the first example, the contour is initialized to be close to boundary B_r. After a few iterations, contour is formed around boundary B_l to reflect the intensity contrast, and contour around boundary B_r is attracted towards the true boundary. However, evolution of ϕ around B_l and B_r are independent in the first few iterations. This results $\phi < 0$ around B_l and $\phi > 0$ around B_r in R_1, and leads to region 1 separated into two at convergence. The reason is that the initial contour is far from B_l, leading to two ambiguous directions to evolve ϕ around B_l: either $\phi > 0$ for R_1 and $\phi < 0$ for R_2, or the other way round. Small perturbation will cause ϕ to evolve in one of the two directions, because having the same sign of ϕ along true boundary is an unstable state of high energy. But which direction ϕ evolves in is hard to predict when initialization is far away, depending on factors like boundary shape and numerical precision.

Summary of proposed method. To remove ambiguity in the evolution direction, we propose intensity re-weighting in the evolution process to bias $\phi > 0$ for the brighter region and and $\phi < 0$ for the darker region, or the other way round. The

basic assumption is that one region is always brighter than the other along the boundary. In this way, the level set function will evolve towards the biased direction and different parts of the same region have consistent signs of ϕ at convergence.

Synopsis. In the following, we briefly review the LBF model in section 2. Section 3 details our approach of intensity re-weighting. Section 4 shows experimental results and section 5 concludes.

2. BRIEF REVIEW OF THE LBF MODEL

In LBF model, the energy term $\mathcal{E}^{LBF}$ is defined for each point $\mathbf{x}$ in the image as following

$$
\begin{aligned}
\mathcal{E}_{\mathbf{x}}^{LBF}&(\phi, f_1(\mathbf{x}), f_2(\mathbf{x})) = \\
&\lambda_1 \int K_\sigma(\mathbf{y} - \mathbf{x})|I(\mathbf{y}) - f_1(\mathbf{x})|^2 H_1(\phi(\mathbf{y}))d\mathbf{y} \\
+ \quad &\lambda_2 \int K_\sigma(\mathbf{y} - \mathbf{x})|I(\mathbf{y}) - f_2(\mathbf{x})|^2 H_2(\phi(\mathbf{y}))d\mathbf{y} \quad (1)
\end{aligned}
$$

$\mathcal{E}_{\mathbf{x}}^{LBF}$ measures the summation of intensity variation within a local neighborhood of point $\mathbf{x}$ for two regions. $K_\sigma(\mathbf{y} - \mathbf{x})$ is a Gaussian kernel that diminishes with distance from $\mathbf{x}$. S_1 and S_2 refers to segment of $\phi > 0$ and $\phi < 0$ respectively. $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ are spatially fitting functions, capturing weighted average intensities of S_1 and S_2 from a local view of $\mathbf{x}$. $I(\mathbf{y})$ is the intensity at $\mathbf{y}$. $H_1(\phi(\mathbf{y}))$ and $H_2(\phi(\mathbf{y}))$ capture the belief of point $\mathbf{y}$ in S_1 and S_2. $H_1(\phi)$ is a Heaviside function.

$$
H_1(\phi) = \frac{1}{2}(1 + \frac{2}{\pi}\arctan(\frac{\phi}{\epsilon})) \quad (2)
$$
$$
H_2(\phi) = 1 - H_1(\phi) \quad (3)
$$

An intuitive explanation of minimizing $\mathcal{E}_{\mathbf{x}}^{LBF}$ is that $I(\mathbf{y})$ should be close to $f_1(\mathbf{x})$ if $H_1(\phi(\mathbf{y}))$ is high, and close to $f_2(\mathbf{x})$ otherwise. The complete LBF energy is defined as

$$
\mathcal{E}^{LBF}(\phi, f_1, f_2) = \int_{\mathbf{x}} \mathcal{E}_{\mathbf{x}}^{LBF}(\phi, f_1(\mathbf{x}), f_2(\mathbf{x}))d\mathbf{x} \quad (4)
$$

Keeping ϕ fixed, minimizing $\mathcal{E}^{LBF}$ with respect to $f_1(\mathbf{x}), f_2(\mathbf{x})$ gives

$$
f_i(\mathbf{x}) = \frac{K_\sigma(\mathbf{x}) * [H_i(\phi(\mathbf{x}))I(\mathbf{x})]}{K_\sigma(\mathbf{x}) * [H_i(\phi(\mathbf{x}))]}, i = 1, 2 \quad (5)
$$

Keeping f_1 and f_2 fixed, minimizing $\mathcal{E}^{LBF}$ with respect to ϕ, we derive the gradient descent flow:

$$
\frac{\partial \phi}{\partial t} = -\frac{\partial \mathcal{E}^{LBF}}{\partial \phi} = -\delta(\phi)(\lambda_1 e_1 - \lambda_2 e_2) \quad (6)
$$

$$
e_i(\mathbf{x}) = \int K_\sigma(\mathbf{x} - \mathbf{y})|I(\mathbf{x}) - f_i(\mathbf{y})|^2 d\mathbf{y} \quad (7)
$$

In Eq.(6), $\delta(\phi)$ is the smooth Dirac function, which is the derivative of $H_1(\phi)$. $e_i(\mathbf{x})$ measures the intensity coherence

of $I(\mathbf{x})$ with the average intensity of S_i near $\mathbf{x}$. Larger e_i means less coherent. Eq.(6) can be interpreted as increasing the belief of $\mathbf{x}$ in S_i if $e_i < e_{3-i}, i = 1, 2$.

The complete energy definition has two extra regularization terms: $\mathcal{E} = \mathcal{E}_{\mathbf{x}}^{LBF} + \mu\mathcal{P}(\phi) + \nu\mathcal{L}(\phi)$. $\mathcal{P}(\phi)$ keeps ϕ to be close to the signed distance function. $\mathcal{L}(\phi)$ minimized the contour length. We keep these two terms unchanged.

3. PROPOSED METHOD

The goal of segmentation is to make $S_i = R_i$ or $S_i = R_{3-i}$ ($i = 1, 2$) at convergence. Without lose of generality, we assume the goal is $S_i = R_i$, i.e., $\phi > 0$ in brighter region and $\phi < 0$ in darker region at convergence. We propose to incorporate bias in the level set function evolution process by adjusting the intensity weight when computing f_1 and f_2 in Eq.(5). From Eq.(5), we know $f_i(\mathbf{y})$ measures the average intensity of S_i around point $\mathbf{y}$. Given the assumption that R_1 is brighter than R_2 along the boundary, we can put higher weights on brighter pixels when computing f_1 and higher weights on darker pixels when computing f_2.

$$
f_i = \frac{K_\sigma(\mathbf{x}) * [H_i(\phi(\mathbf{x}))I(\mathbf{x})W_i(I(\mathbf{x}))]}{K_\sigma(\mathbf{x}) * [H_i(\phi(\mathbf{x})W_i(I(\mathbf{x}))]}, i = 1, 2 \quad (8)
$$
$$
W_1(I(\mathbf{x})) = I(\mathbf{x}) \quad (9)
$$
$$
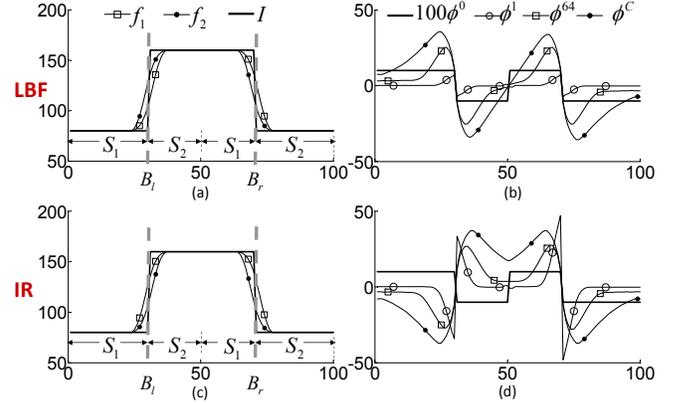W_2(I(\mathbf{x})) = 255 - I(\mathbf{x}) \quad (10)
$$

Fig. 2. Comparison of LBF and IR for the first example in Fig. 1. Here we only show value for the middle row of the image. Initialization is the same as the convergence state in Fig. 1. One the left, we show f_1, f_2 and I in the first iteration. On the right, we show evolution of ϕ. ϕ^t is ϕ at iteration t. ϕ^0 is the initialization, and ϕ^C is ϕ at convergence. LBF does not change sign of ϕ in evolution process, so getting stuck in local minimum. IR drives sign of ϕ to flip in iteration 1, and converges to global minimum.

We call LBF with intensity re-weighting as IR model, which naturally encourages ϕ of brighter pixels to increase

and ϕ of darker pixels to decrease along boundary. Fig. 2 shows for the first example in Fig. 1, why LBF gets stuck at local minimum and IR converges to the desired global minimum. We initialize sign of ϕ to be the convergence state in Fig. 1, with absolute value of 0.1. Here we just show value for the middle row in the image (other rows are similar given no vertical intensity variation in the image). Eq.(7) can be approximated as $e_i(\mathbf{x}) = |I(\mathbf{x}) - f_i(\mathbf{x})|^2$ because $f_i(\mathbf{y}) \approx f_i(\mathbf{x})$ when $\mathbf{y}$ is close to $\mathbf{x}$. Eq.(6) can be interpreted as increasing ϕ if $|I(\mathbf{x}) - f_1(\mathbf{x})| < |I(\mathbf{x}) - f_2(\mathbf{x})|$, and decreasing ϕ otherwise. In LBF around boundary B_l, $\phi > 0$ for darker pixels and $\phi < 0$ for brighter pixels when initialized, resulting $f_1 < f_2$. So near B_l, Eq.(6) drives ϕ to increase in $S_1(\phi > 0)$ and decrease in in $S_2(\phi < 0)$, as shown in Fig. 2(b). In IR, around B_l, $f_1 > f_2$ because brighter pixels are weighted more in f_1, and darker pixels are weighted more in f_2. So near B_l, Eq.(6) drives ϕ to decrease in $S_1(\phi > 0)$ and increase in $S_2(\phi < 0)$, causing the sign of ϕ to flip around B_l. At convergence, ϕ has consistent sign in each region.
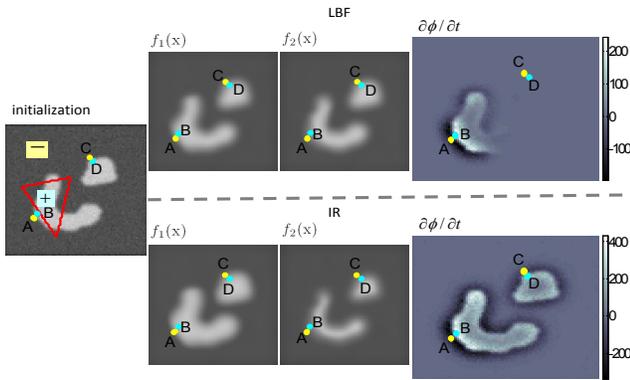

Fig. 3. Comparing $f_1(\mathbf{x})$, $f_2(\mathbf{x})$ and $\frac{\partial \phi}{\partial t}$ in LBF and IR models. '+' and '-' indicates sign of the level set function ϕ. (view in color)

Fig. 3 shows a real example. Evolution direction of ϕ is visualized in the last column: increase ϕ for bright pixels ($\frac{\partial \phi}{\partial t} > 0$) and decrease ϕ for dark pixels ($\frac{\partial \phi}{\partial t} < 0$). In both LBF and IR, if a pixel is close to initial contour and true boundary (like A, B), then $f_1 > f_2$. So Eq.(6) drives ϕ to increase for bright pixel B and decrease for dark pixel A. However, if a pixel is close to true boundary but far from initialization (like C, D), in LBF $f_1 \approx f_2$. This makes it unclear how ϕ evolves to reflect the intensity contrast on that boundary. Either increasing ϕ for C and decreasing ϕ for D or vice versa can decrease the energy. Such uncertainty of which direction to go is exactly the reason that can lead to local minimums (convergence shown in Fig. 6 first column). In IR model, $f_1 > f_2$ near C and D, driving ϕ to evolve in the desired direction.

Another advantage of IR model is that it enables faster convergence. LBF relies on the closeness of current contour
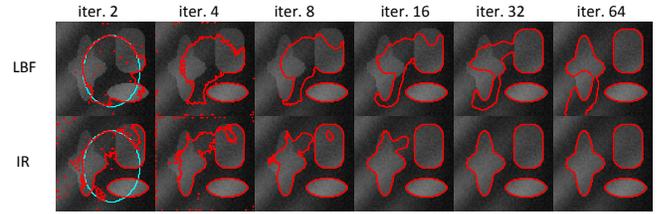

Fig. 4. Comparing evolution contour (in red) at iteration 2–64 for the same initialization (in cyan) for LBF and IR model. (view in color)

to the true boundary to guide evolution in the right direction. Whereas IR provides additional driving force from intensity contrast to guide evolution even when current contour is far from the true boundary. Fig. 4 gives such an example.

Implementation. The implementations of IR model and LBF model are very similar. The only difference comes from computing f_i in Eq.(5) and Eq.(8). In LBF, $K_\sigma * I$ and $K_\sigma * \mathbf{1}$ ($\mathbf{1}$ is an array of 1s, and of the same size as I) can be pre-computed once. In each iteration, Eq.(5) can be computed from two convolutions C_1 and C_2.

$$C_1 = K_\sigma(\mathbf{x}) * H_1(\phi(\mathbf{x})) \tag{11}$$
$$C_2 = K_\sigma(\mathbf{x}) * [H_1(\phi(\mathbf{x}))I(\mathbf{x})] \tag{12}$$
$$f_1 = C_2/C_1 \tag{13}$$
$$f_2 = (K_\sigma * I - C_2)/(K_\sigma * \mathbf{1} - C_1) \tag{14}$$

In IR model, we need to pre-compute $K_\sigma * I^2$, $K_\sigma * I$ and $K_\sigma * \mathbf{1}$ once. In each iteration, Eq.(8) can be computed from three convolutions C_1, C_2 and C_3. C_1, C_2 are the same as Eq.(11–12),

$$C_3 = K_\sigma(\mathbf{x}) * [H_1(\phi(\mathbf{x}))I^2(\mathbf{x})] \tag{15}$$
$$f_1 = C_3/C_2 \tag{16}$$
$$f_2 = \frac{255(K_\sigma * I - C_2) - K_\sigma * I^2 + C_3}{255(K_\sigma * \mathbf{1} - C_1) - K_\sigma * I + C_2} \tag{17}$$

Compared to LBF, we observe about 10%–15% increase in the computing time for each iteration in IR model.

4. EXPERIMENTAL RESULT

All experimental results use the same parameter setting $\lambda_1 = \lambda_2 = 1$, $\nu = 0.001 \times 255^2$, $\mu = 1$, $\sigma = 3.0$, $\epsilon = 1$, except for the first example in Fig. 5, we set $\nu = 0.003 \times 255^2$.

We first show some examples for which LBF get stuck in local minimums, but IR successfully converges to the desired boundary in Fig. 6. The last column shows an example where IR model does not work. This is an example when the basic assumption is violated. The two regions are of completely symmetric intensity. Unsurprisingly, at convergence $\phi > 0$
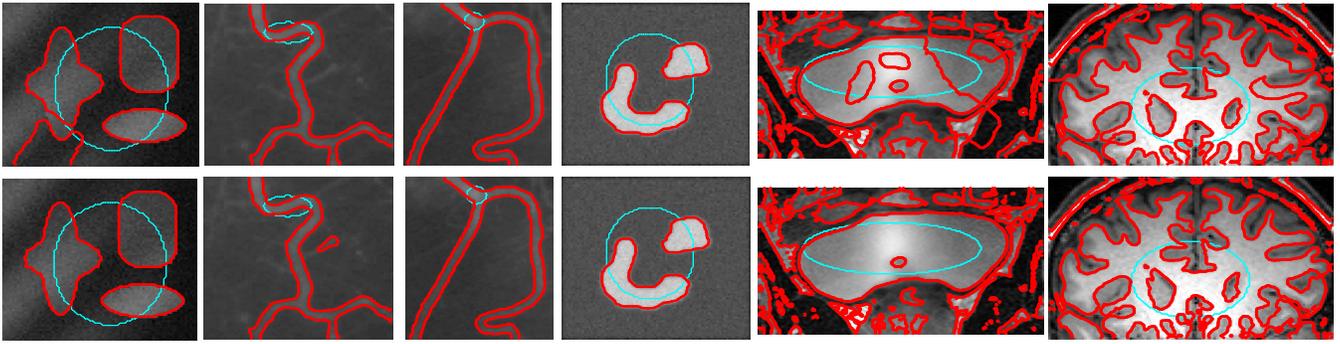
Fig. 5. Comparison of LBF (row 1) and IR (row 2) model, with initializations in cyan and final contours in red. (view in color)

maps to brighter areas. Preprocessing the image (like computing the gradient image) can meet the basic assumption.

In Fig. 5, we show both real and synthetic examples, and compare CPU time using LBF and IR in Table 1. To make fair comparison of computing time, we choose examples in which both models converge (or closely) to desired region boundaries. Table 1 lists CPU time for Matlab code on a Dell XPS 720 machine with 2.66G Hz Intel Core 2 Extreme QX6700 CPU and 2GB memory. We check convergence at every 25 iterations. If the average percentage of pixels that change the sign of ϕ is less than 0.2%, then the model converges. It is clear that IR model converges much faster than LBF model.

ImgID	1	2	3	4	5	6
LBF(CPU time)	3.16	4.23	7.47	0.80	7.02	2.33
iter #	350	225	350	75	275	175
IR(CPU time)	0.78	2.14	1.80	0.64	5.18	1.47
iter #	75	100	75	50	175	100

Table 1. CPU time (in second) and iteration number at convergence for LBF and IR model in Fig. 5.

5. CONCLUSION

In this paper, we propose an intensity re-weighting (IR) model, which is an extension of LBF model. LBF model can easily get stuck in local minimums if initialization is far from the true region boundary. Given the basic assumption that one region is brighter than the other along the region boundary, IR model can effectively avoid local minimums and allow much faster convergence. Experimental results demonstrate the effectiveness of the proposed method.

When multiple regions exist, it's very common to see one region been split when initialization is not close enough to true boundary. Current IR model cannot handle multiple regions. Extending the idea of IR may enable auto-segmentation without manual initialization.
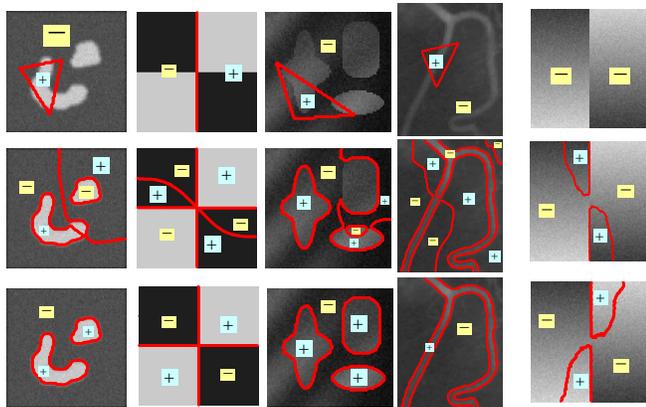

Fig. 6. Top to bottom: initialization, final contours in LBF model, final contours in IR model. '+' and '-' indicates sign of the level set function ϕ. First four columns show examples in which LBF model gets stuck in local minimums and IR model converges to the desired boundary. The last column shows an example that IR model fails.(view in color)

6. REFERENCES

[1] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," in *IJCV*, 1997.

[2] T. Chan and L. Vese, "Active contours without edges," in *IEEE Trans. on Image Processing*, 2001.

[3] L. Vese and T. Chan, "A multiphase level set framework for image segmentation using the mumford and shah model," in *IJCV*, 2002.

[4] C. Li, C. Kao, J. C. Gore, and Z. Ding, "Minimization of region-scalable fitting energy for image segmentation," in *IEEE Trans. on Image Processing*, 2008.

[5] J. A. Sethian, ," in *Level set methods and fast marching methods*. 1999, Cambridge: Cambridge University Press.