COSE490 Digital Image Processing (Fall 2020)
Instructor: Prof. Won-Ki Jeong
Due date: Dec 13, 2020, 11:59 pm.

Assignment 3: Level-set Image Segmentation (200 pts)

In this assignment, you will implement a level-set image segmentation method. The algorithm we are implementing is Geodesic Active Contour formulation of level set method by Caselles et al. (Equation 15).

Caselles, V., Kimmel, R. & Sapiro, G. Geodesic Active Contours. International Journal of Computer Vision 22, 61–79 (1997).

In this model, the implicit zero level-set contour is evolving over time, and its update is defined as a time-dependent PDE as follows:

$$\begin{aligned}\frac{\partial u}{\partial t} &= g(I)|\nabla u|\operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) + cg(I)|\nabla u| \\ &= g(I)(c+\kappa)|\nabla u|,\end{aligned}$$

Here, *u* is the distance field, *g* is the edge indicator term, c is a constant weight, and *k* is the curvature term. A commonly used edge indicator term is as follows:

$$g = \frac{1}{1+|\nabla \hat{I}|^p}$$

where $\hat{I}$ is a smoothed version of the input image I, and p = 2. You can use any image smoothing algorithm to generate $\hat{I}$ (Gaussian smoothing is commonly used). This edge indicator works as follows: when the level set curve is moving closer to edges, then the magnitude of gradient becomes larger and therefore g becomes smaller, which makes the movement of the curve smaller.

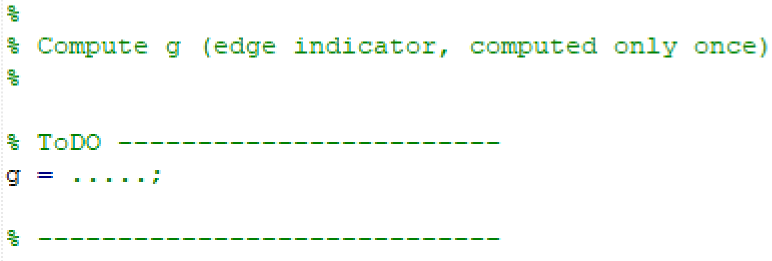
In order to update the distance field u, you need to compute the above terms and update as follows:

$$u' = u + dt(g(I)(c+k)|\nabla u|)$$

Note that c is a scalar constant while k is per-pixel curvature map (same size as the input image). g(I) is also per-pixel edge indicator map.

1. Implementation

The provided `assign_3_skeleton.m` is the main driver code that calls the level set update function (`levelset_update.m`). In the main driver code, you are <u>only allowed</u> to modify the following part to compute the edge indicator term g:

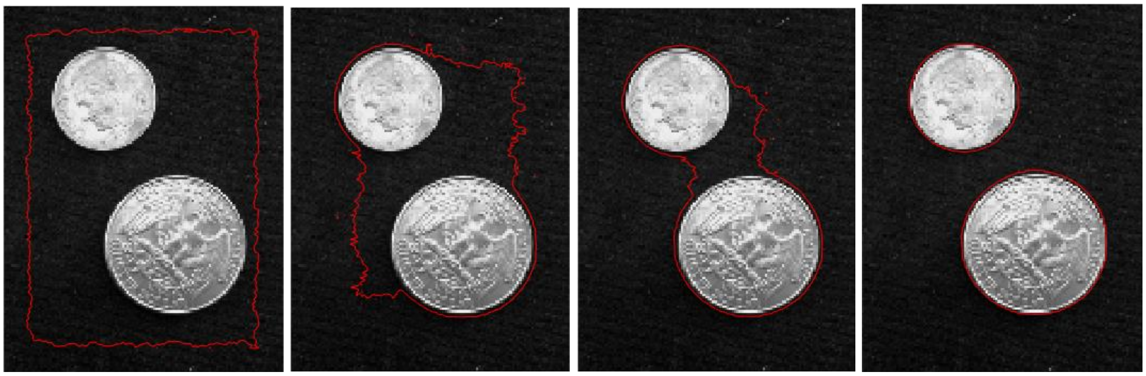


```
%
% Compute g (edge indicator, computed only once)
%

% ToDO ------------------------
g = .....;

% ----------------------------
```

This should be computed only once and reused because this is not changing as the level set evolves.

You can freely edit the function `levelset_update.m`, which accepts the distance field (phi) and update its value.

If your code is working correctly, then you will get the results as follows (from left to right, iteration 10, 100, 200, and 400):



2. Submission

You need to write a short report explaining the results (test with your own images, try with different parameters). Submit the report (pdf) and source code via blackboard. As usual, the report is worth 10 pts.

Good luck and have fun!