

1 Introduction

The source code of the assignment is included in the package. The function file is *relaxed = relaximage(image, niters)*. The top level wrapper of the function (a.k.a. program entry) is *hw5.m*.

I have taken reference from <http://www.mathworks.com/matlabcentral/fileexchange/34807-nonlinear-relaxation-labeling-for-image-processing>. Since the reference has pretty good data struct, I used its data structure.

The following sections explain the questions of the assignment.

2 What are the nodes?

The nodes of relaxation labelling is simply the pixels of each image. The neighbours of each node is the neighbouring pixels of the particular pixel. For example, if the particular pixel is $N0$, and the neighbouring pixels are $n1, n2, n3, n4, n5, n6, n7, n8$. Below are the illustration of the neighbourhood.

$n1 \ n2 \ n3$

$n4 \ N0 \ n5$

$n6 \ n7 \ n8$

As a result, each node has 8 neighbours, and the neighbourhood has 8 nodes. When the node is at the edge of the frame, I fill the unfilled slots with replicated confidences.

3 What are the labels?

Since this is a binary labelling problem, we will only have two labels, foreground (objects) and background. Background are white, and objects are black as shown in the given pictures.

4 What are the initial confidences be set?

The initial confidences is the pixel density of each node. The pixel density ranges from 0 to 255 in the original image as 255 being all black and 0 being all white. To make it between 0 and 1, I divided all pixels by 255.

After division, the probability of the node being a part of the object, P_{object} is $density/255$, and the probability of the node being a part of the background, $P_{background}$ is $1 - P_{object}$ for all pixels (nodes).

5 What should the neighbourhood relationships be?

For all 8 neighbours ($P_{neighbour}$) of the chosen node (P_{node}):

compute $P(object|object)$, this notation stands for $P_{node}(object)$ given $P_{neighbour}(object)$

do the same thing for $P(object|background)$, $P(background|background)$, and $P(background|object)$.

This step can be interrupted as what is current node's probability of being object or background if the given neighbour has x probability of being background or object.

and then divide the results by ($P_{node}(object)$)

This is our results for neighbourhood relationships.

6 How should the compatibility fields be structured?

For each labels (we only have 2)

The compatibility is the sum of $P_{neighbour} * neighbourhood\ relationships$ for all neighbours.

7 results

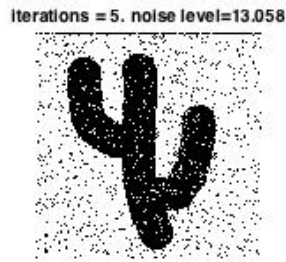


Figure 1: cactus picture with 5 iterations, we see there are a lot of noises

iterations = 10. noise level=12.432

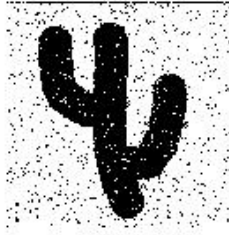


Figure 2: cactus picture with 10 iterations, noise reduces

iterations = 20. noise level=11.506

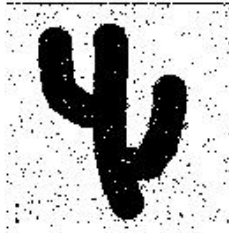


Figure 3: cactus picture with 20 iterations, noise reduces

iterations = 50. noise level=10.655

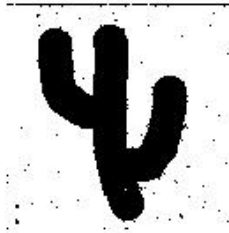


Figure 4: cactus picture with 50 iterations, noise reduces

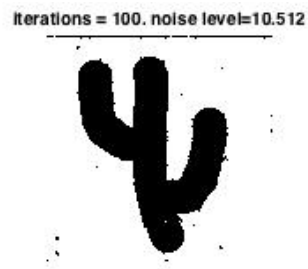


Figure 5: cactus picture with 100 iterations, noise reduces slightly

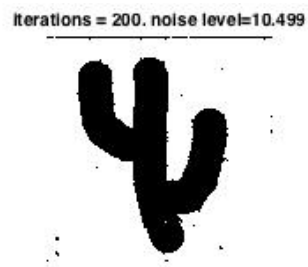


Figure 6: cactus picture with 100 iterations, we can barely see any improvement. Since the algorithm does not guarantee converge, it is possible that relaxation labelling reaches a limit

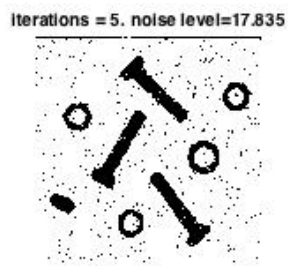


Figure 7: parts picture with 5 iterations, we see there are a lot of noises

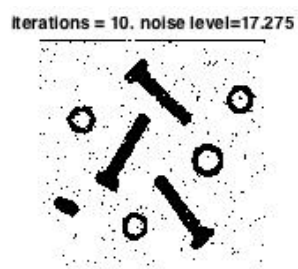


Figure 8: parts picture with 10 iterations, noise reduces

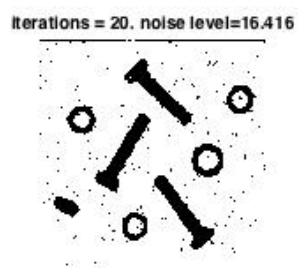


Figure 9: parts picture with 20 iterations, noise reduces

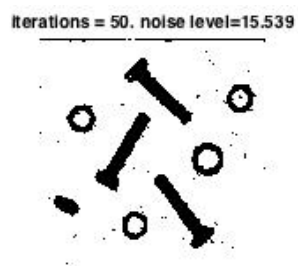


Figure 10: parts picture with 50 iterations, noise reduces

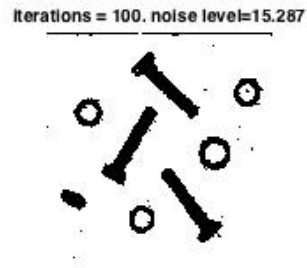


Figure 11: parts picture with 100 iterations, noise reduces slightly

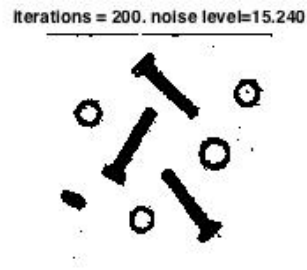


Figure 12: parts picture with 200 iterations, we can barely see any improvement. Since the algorithm does not guarantee converge, it is possible that relaxation labelling reaches a limit. However, I do see noise level reduces

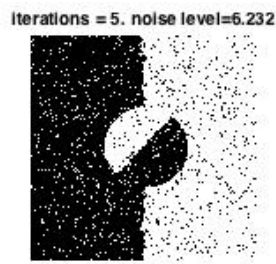


Figure 13: bump picture with 5 iterations, we see that instead of showing a bump-ish figure, it shows a black-white sphere, which is not really what we are looking for, the main reason is that the bump picture has the shading effect, and our probability model only makes a binary decision (black \rightarrow object, white \rightarrow background), meaning if P is less than 0.5 it is considered background, vice versa.

iterations = 10. noise level=6.218

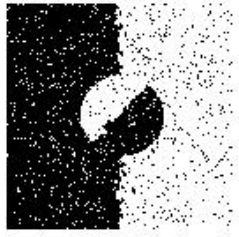


Figure 14: bump picture with 10 iterations, noise reduces, but it still only shows binary decisions

iterations = 20. noise level=6.164



Figure 15: bump picture with 20 iterations, noise reduces

iterations = 50. noise level=5.950

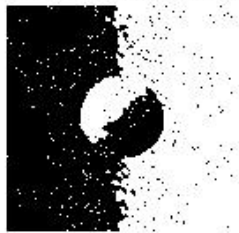


Figure 16: bump picture with 50 iterations, noise reduces

iterations = 100. noise level=5.746



Figure 17: bump picture with 100 iterations, noise reduces slightly

iterations = 200. noise level=5.694



Figure 18: bump picture with 200 iterations. Though the noises are removed, we see some unwanted shapes, and I have explained the reasons in iteration=5 picture.