ECE 6310- INTRODUCTION TO COMPUTER VISION

LAB 3: LETTERS

Submitted To:

Dr. Adam Hoover

Ashit Mohanty

C13582787

amohant@g.clemson.edu

Date of submission: 09-29-2020

Introduction

This project requires the usage of the concepts of thinning, branchpoint and endpoint detection to detect the letter e. The program was written using the language C. The user must input only threshold limits, lower and upper threshold respectively, between which the program runs to generate the total false positives and the total true positives for every threshold. The values of True positives, False Positives, False Negatives, and True Negatives are written in a text file, which is then used to compute the ROC curve using Microsoft Excel. From the ROC Curve, the most optimal value of threshold is computed that explains the value that is closest to the ideal value of True Positive Rate 1 and False Positive Rate 0. The program is submitted along with this report as a separate file title, lab3_ashit.c.

Mean Spatial Filter Image

The first step of the project involves using the mean spatial filter image from the previous lab project. The steps for which are explained underneath,

To perform this objective, first the zero mean template had to be generated of the original template image. This was generated by subtracting the average of all pixels in the template image, parenthood_e_template.ppm, from every individual pixel in the template image. This template was then used to perform convolution on the original input image, parenthood.ppm. The image that is generated is not an 8-bit image. This image is then normalized to 8-bits using the formula,

$$I_N = (I - Min)\frac{newMax - newMin}{Max - Min} + newMin$$

Where,

I = Image pixel after convolution, but not normalized

Min = Minimum pixel value in the MSF image, but not normalized

Max = Maximum pixel value in the MSF image, but not normalized

$newMin$ = New Minimum pixel value in the normalized MSF image, in this case 0

$newMax$ = New Maximum pixel value in the normalized MSF image, in this case 255

I_N = Image pixel after convolution and after being normalized

The original image and the MSF image after being normalized are shown underneath.

Preparation for parenthood is not just a matter of reading books and decorating the nursery. Here are some tests for expectant parents to take to prepare themselves for the real-life experience of being a mother or father.

4. Can you stand the mess children make? To find out, smear peanut butter onto the sofa and jam onto the curtains. Hide a fish finger behind the stereo and leave it there all summer. Stick your fingers in the flowerbeds then rub them on the clean walls. Cover the stains with crayons. How does that look?

5. Dressing small children is not as easy as it seems. First buy an octopus and a string bag. Attempt to put the octopus into the string bag so that none of the arms hang out. Time allowed for this - all morning.

7. Forget the Miata and buy a Mini Van. And don't think you can leave it out in the driveway spotless and shining. Family cars don't look like that. Buy a chocolate ice cream bar and put it in the glove compartment. Leave it there. Get a quarter. Stick it in the cassette player. Take a family-size packet of chocolate cookies. Mash them down the back seats. Run a garden rake along both sides of the car. There!. Perfect!

9. Always repeat everything you say at least five times.

11. Hollow out a melon. Make a small hole in the side. Suspend it from the ceiling and swing it from side to side. Now get a bowl of soggy Froot Loops and attempt to spoon it into the swaying melon by pretending to be an airplane. Continue until half of the Froot Loops are gone. Tip the rest into your lap, making sure that a lot of it falls on the floor. You are now ready to feed a 12-month old baby.

Figure 1: Original Image – parenthood.ppm

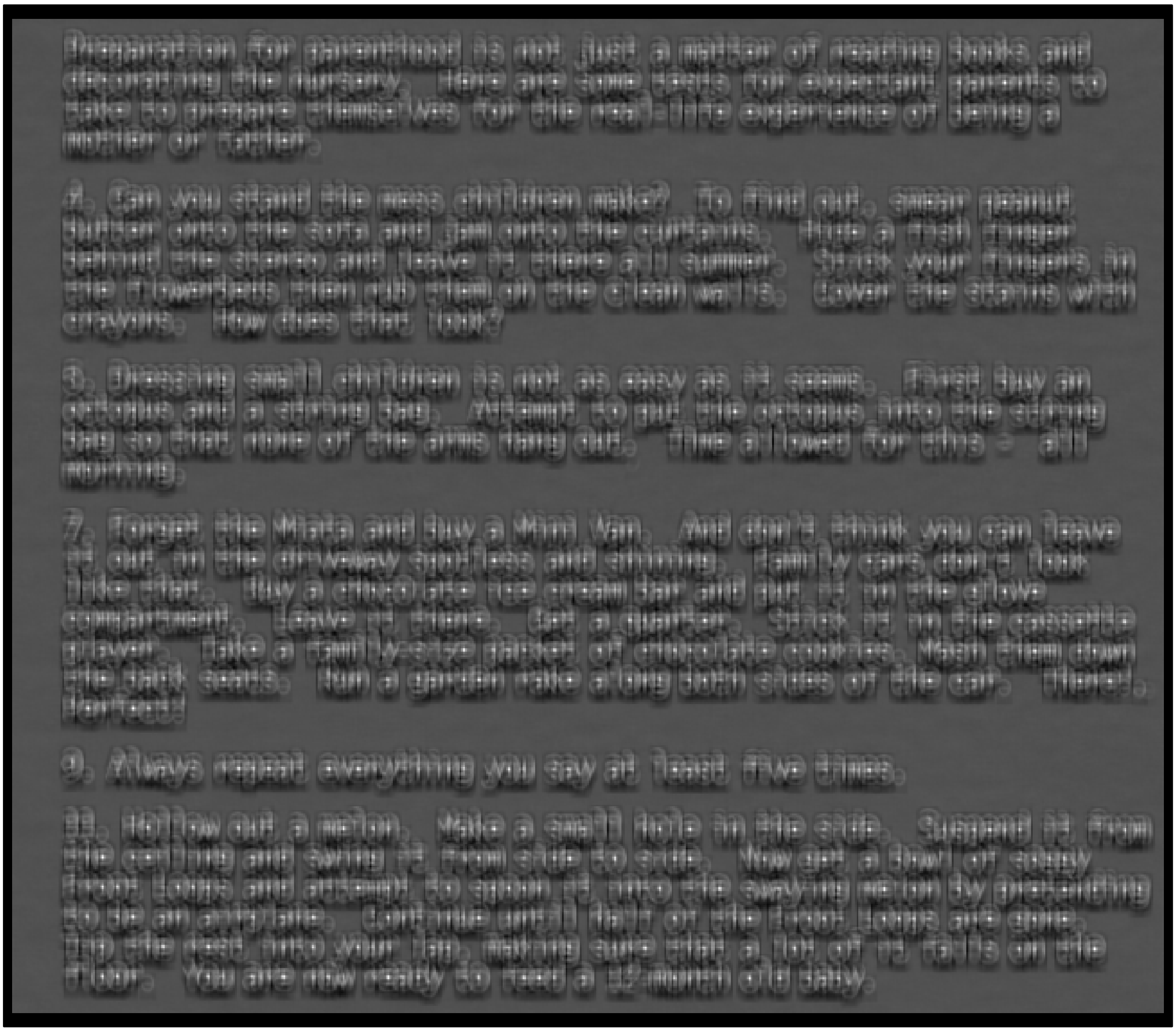

Figure 2: Mean Spatial Filer (MSF) Image

Detecting letter ‘e’ – phase 1 and Binary Image generation

In the next step, possible locations of e are detected by checking if any pixel in a 9X15 window centered at the ground truth locations in the MSF image, that was generated in the above step, is greater than the threshold value. If yes, this means that the MSF image centered at the ground truth location is detecting a possibility of letter e at that location. If no, then the letter is not detected.

If the letter is detected from the above described step, a 9X15 image is extracted centered at the ground truth location from the original image, as shown in Figure 3. In the examples shown, the letter e is extracted from the image.

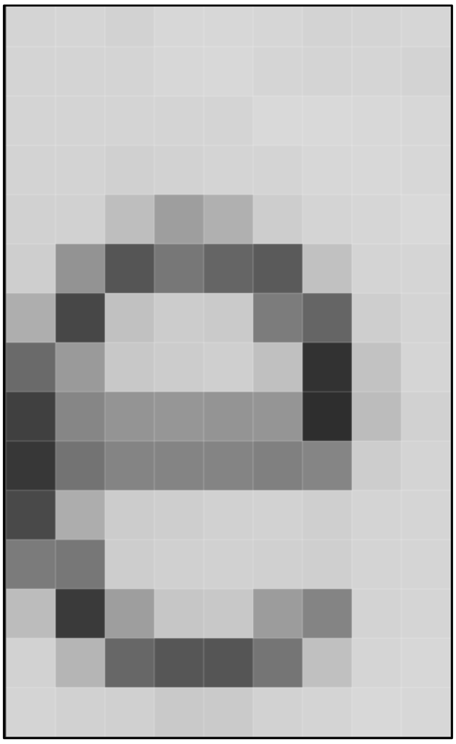

Figure 3: 9X15 image extracted from parenthood.ppm at ground truth location

This 9X15 image is then threshold at a value of 128, to generate a binary image. The image is shown in Figure 4.

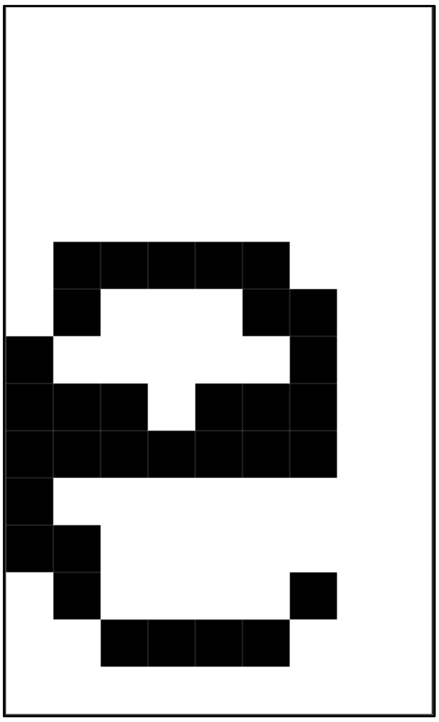

Figure 4: Binary Image of extracted image from parenthood.ppm

Thinning of image

The binary image that was generated had to be thinned down to single pixel width image to correctly detect the number of branch points and end points. To thin the image, the entire 9X15 binary image was traversed, and a particular set of conditions were checked for each individual pixel to detect if the pixel had to be removed(made white) so as to thin down the image. To check if any pixel were to be removed, the following three conditions had to be true,

- The number of edge(black pixel- value 0) to non-edge(white pixel- value 255) transitions, in a 3X3 matrix centered around the pixel in the 9X15 image, in a clockwise direction and excluding the center pixel, had to be equal to 1. This value is ‘a’.
- The number of edge neighbors (black pixel- value 0), in a 3X3 matrix centered around the pixel in the 9X15 image, excluding the center pixel, had to lie between 2 and 6(inclusive both). This value is ‘b’.
- The pixel to the north or the pixel to the east or the pixels to the south and west, of the pixel under scrutiny, should be a non-edge i.e. white pixel- value 255. This value is ‘c’.

If the above set of conditions are true, the pixel is marked for removal. This is performed for all pixels of the image and a thinned image is generated. This thinned image is again subjected to above mentioned steps of thinning and checked if more pixels can be removed. If yes, then thinning is performed, and step is repeated until no pixels can be removed any further. This final image is the thinned image (Figure 5), when no more thinning can take place. To perform this thinning operation, I have written a function thinning() that accepts the 9x15 image and the position of pixel that is to be checked for removal. And the return value of the function is the value of ‘a’, ‘b’, ‘c’ and ‘res’. ‘res’ is a variable that denotes if the pixel is to be removed or not.

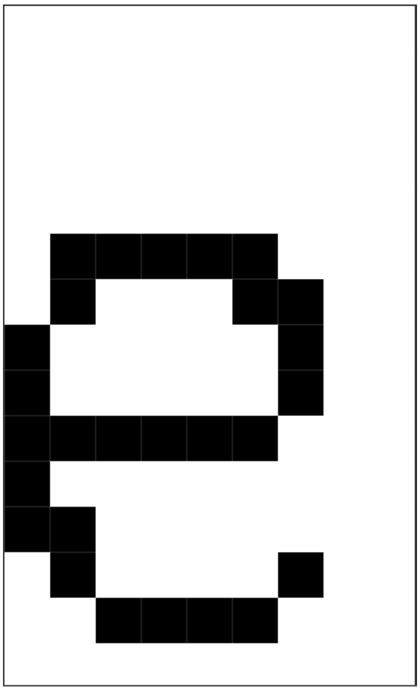

Figure 5: Thinned Binary Image

Branch points- End points detection

This thinned 9X15 image is then traversed to detect the number of branch points and end points. To perform the detection, all the pixels of the thinned 9X15 image are traversed and the number of branch points and end points are detected by the following steps.

- If the thinned image has a pixel that has only 1 edge to non-edge transition this means that, that pixel is an End point.
- If the thinned image has a pixel that has more than 2 edge to non-edge transition this means that, that pixel is a Branch point.

If a 9X15 thinned image has exactly 1 End point and 1 Branch point that means that letter 'e' is in fact detected. If not, then the letter is not an e.

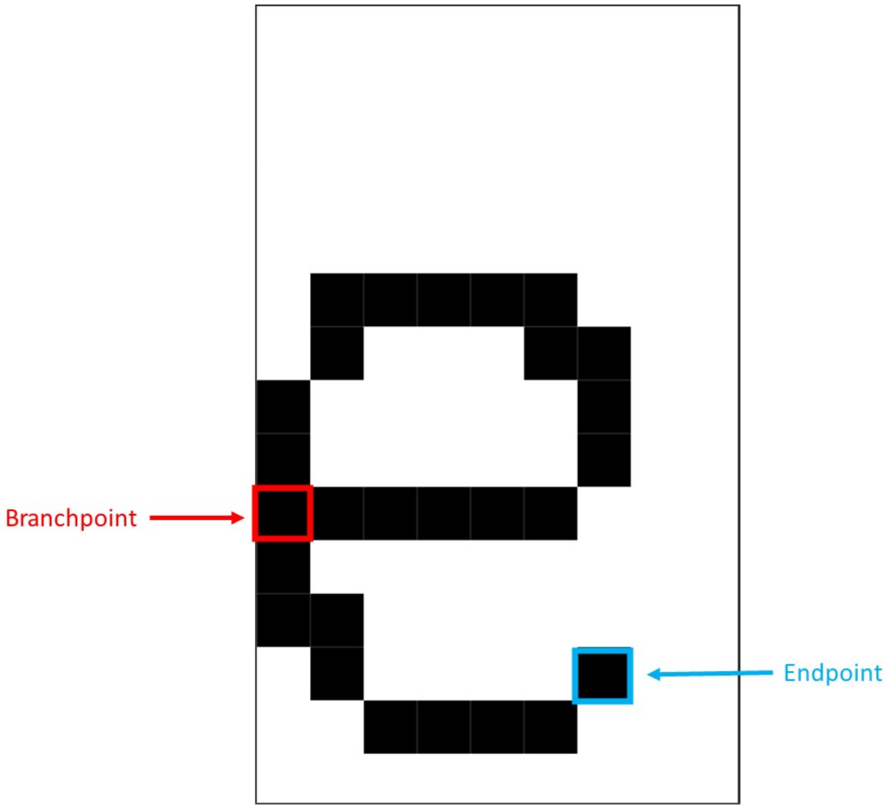

Figure 6: Endpoint and Branchpoint detection from binary image

The branchpoint and endpoint detection is shown in Figure 6.

Output of program

The above-mentioned steps are run for the entire range of thresholds as entered by the user. In the command window only the threshold value, corresponding True

Positives and False positives are printed. The outputs for a threshold range of 190 to 210 are shown in the table below.

Threshold	True Positives (TP)	False Positives (FP)
190	144	74
191	144	68
192	144	60
193	144	55
194	144	51
195	144	46
196	144	39
197	144	34
198	144	32
199	144	29
200	144	25
201	143	22
202	143	20
203	142	13
204	142	11
205	141	8
206	140	7
207	139	7
208	139	7
209	137	5
210	136	2

Also, for every particular threshold value, the number of True Positives (TP), true negatives (TN), false positives (FP), false negatives (FN), True Positive Rate(TPR) and False Positive Rate(FPR) are generated and written in a text file, output_lab3.txt, for generation of the ROC curve.

ROC Curve generation

The ROC or receiver operating characteristic curve is plot between the True Positive Rate and the False Positive Rate. These rates are calculated by the formulae,

$$TPR = \frac{TP}{TP + FN} \quad FPR = \frac{FP}{FP + TN}$$

The ROC curve was plot for the True Positive Rates (TPRs) and False Positive Rates (FPRs) of the range of thresholds, in this case 190 to 210. The ROC curve is shown in the image as attached below in Figure 7,

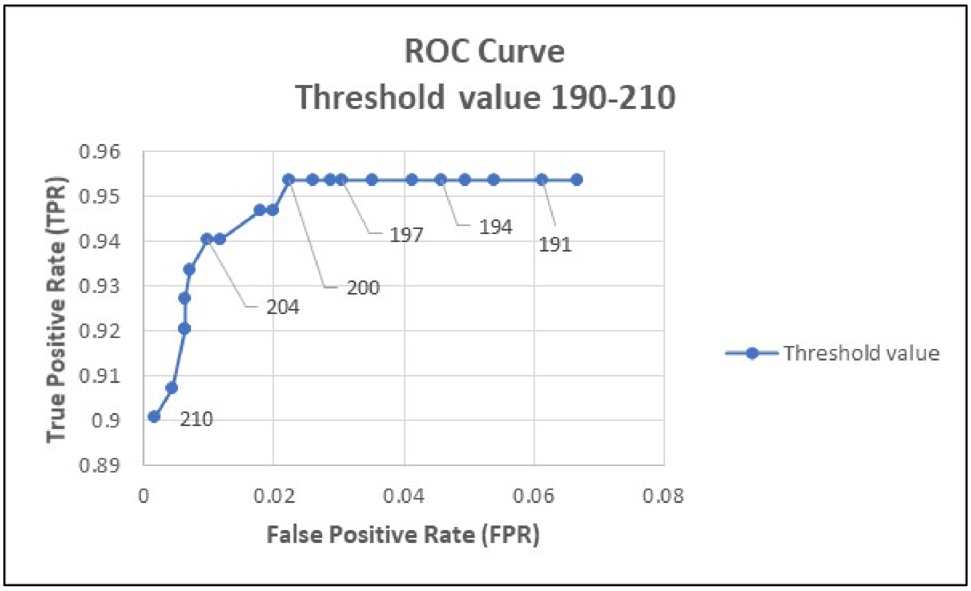

Figure 7: ROC Curve for input threshold value 190-210

The ROC curve was also plot for the entire range of possible threshold values, i.e. 0-255. This way the actual shape of the ROC curve can be verified.

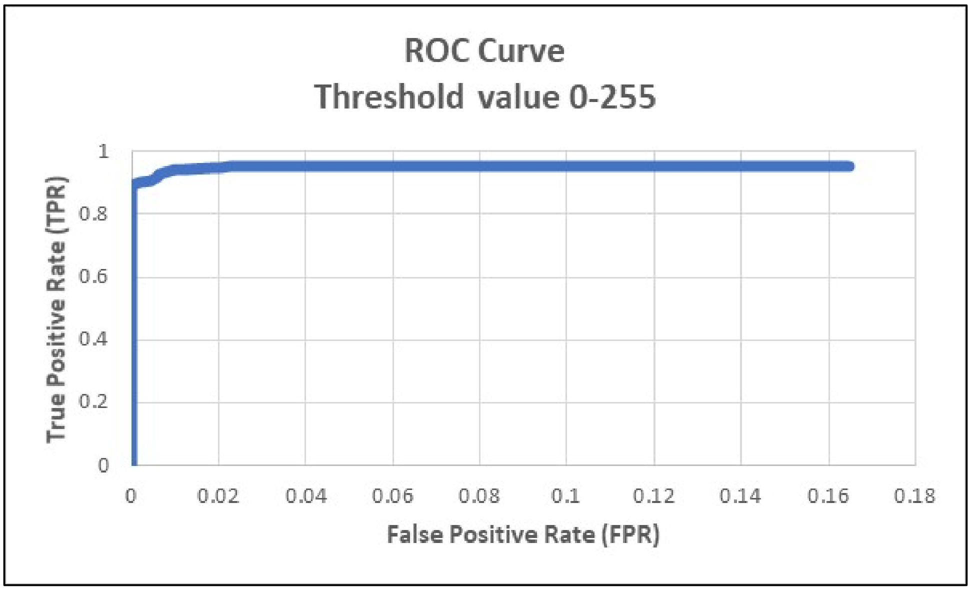

Figure 8: ROC Curve for input threshold value 0-255

As can be seen from the ROC curve for the entire range of threshold values, the ROC curve is analogous to the shape it is supposed to be and the operation of detecting the letter 'e' in the parenthood.ppm image file was successful.

The optimal value of threshold would be a tradeoff between the value of the TPR and the FPR. The optimal value of threshold would be the knee of the curve. The knee of the curve is the value of the curve that is closest to the desired value. The best-case scenario in this problem statement would be to have detected all the letters e and have detected no other letters. This means that TPR should be 1 and FPR should be 0. So, the knee of the curve would be the point that is closest to the ideal point of TPR=1 and FPR=0. This value is found by computing the Euclidean distance between the plot points to the ideal point. Therefore, the optimal threshold value turned out to be a threshold of 200 because it was the nearest to the ideal point of TPR=1 and FPR=0.

Threshold	TP	FP	FN	TN	TPR	FPR
200	144	25	7	1086	0.953642	0.022502

Therefore, the optimal threshold value is **200**, with a True Positive Rate of 0.954 and a False Positive Rate of 0.022.