Assignment 3:
Image Segmentation and evaluation

ED6001: Medical Image Analysis

Abhinav Azad | ED17B001

Guidelines and Index

- This assignment has the following 2 parts consisting of Task 1 and Task 2
- Under each Task there are result figures and evaluation of the task which is meant to be Task 3.

- Codes drive link:
    - https://drive.google.com/drive/folders/1TJv1E0dKoAm5yquvb9202C8uRPwAlThO?usp=sharing
    - Or https://github.com/abhiazadz/Graphcut_and_segmentation_techniques.git

Table of Contents

Objective and Tasks

- <u>Tasks 1:</u> Use histogram-based approach to assign the class label for the given images
- <u>Tasks 2:</u> Apply graph cut method to optimize the delineation of the gland from the background using minimization algorithms: alpha-expansion and alpha-beta swap
- <u>Tasks 3:</u> Compare the segmentation results with the ground truth using metrics like accuracy, Dice similarity coefficient, Jaccard index (JAC), sensitivity, specificity

Task 3- Evaluation metrices

Accuracy

Pixel accuracy is simply the percent of pixels in the image which were correctly classified.

Accuracy = $(TP+TN)/(TP+TN+FP+FN)$

Sensitivity

Measure of how well we are able to segment out the True positives or the foreground (in conventional sense). Sensitivity = $(TP)/(TP+FN)$

Specificity

Measure of how well we are able to segment out the True negatives or the background (in conventional sense). Specificity = $(TN)/(TN+FP)$

IoU or Jaccard index (JAC)

The Intersection over Union (IoU) metric, also referred to as the Jaccard index, is essentially a method to quantify the percent overlap between the target mask and our prediction output. This metric is closely related to the Dice coefficient.

$$IoU = (TP)/(TP+FP+FN)$$

F1 or Dice similarity coefficient

$$F1\ score = (2TP)/(2TP+FP+FN)$$

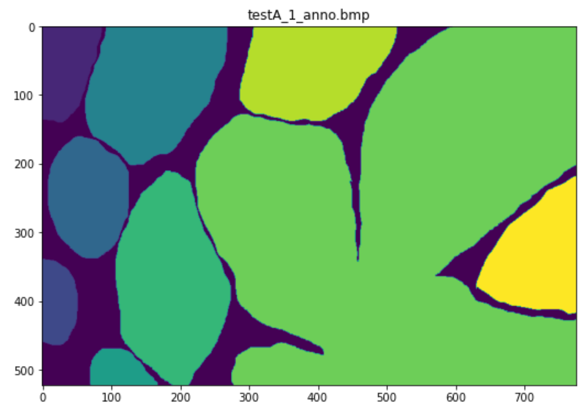
True Label corrections

Image labels provided are in multilabel format and hence it was required to convert them into binary image such that they only represent only 2 classes

1. foreground-object of interest (white-255/1/True)
2. and the background (Black-0/False)

The 2 class binarized images are utilized for final quantitative evaluation of various image segmentation attempts in this assignment.

Image 1 annotation: testA_1_anno.bmp



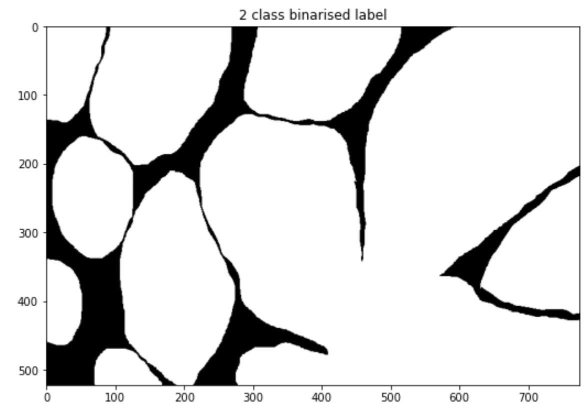
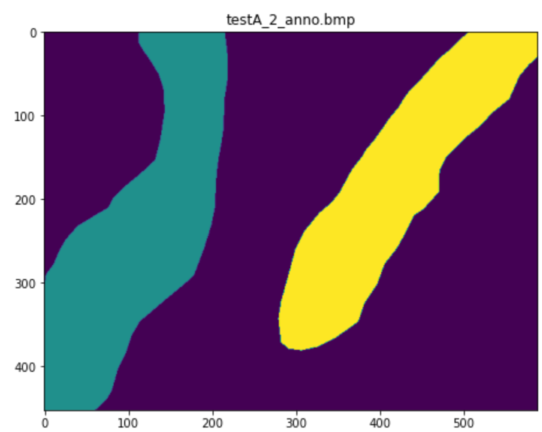


Image 2 annotation: testA_2_anno.bmp



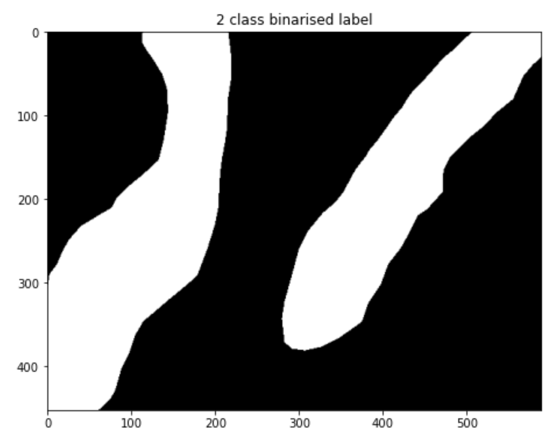
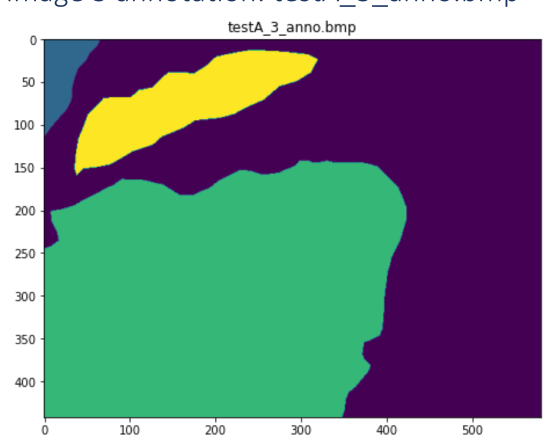


Image 3 annotation: testA_3_anno.bmp



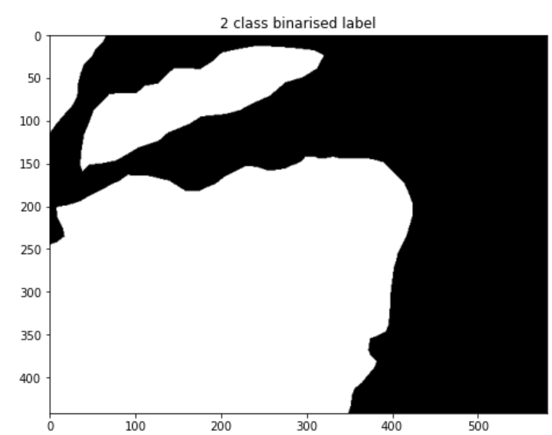


Image 4 annotation: testA_4_anno.bmp

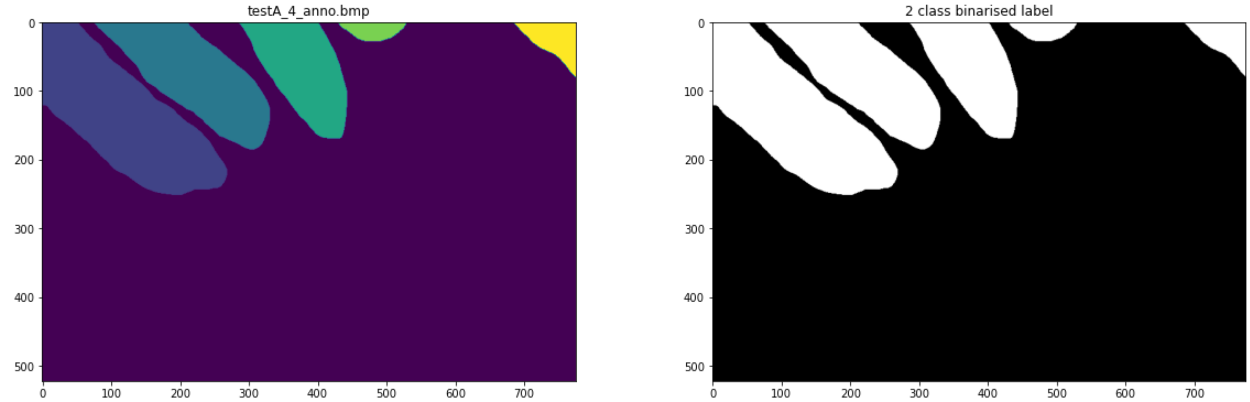
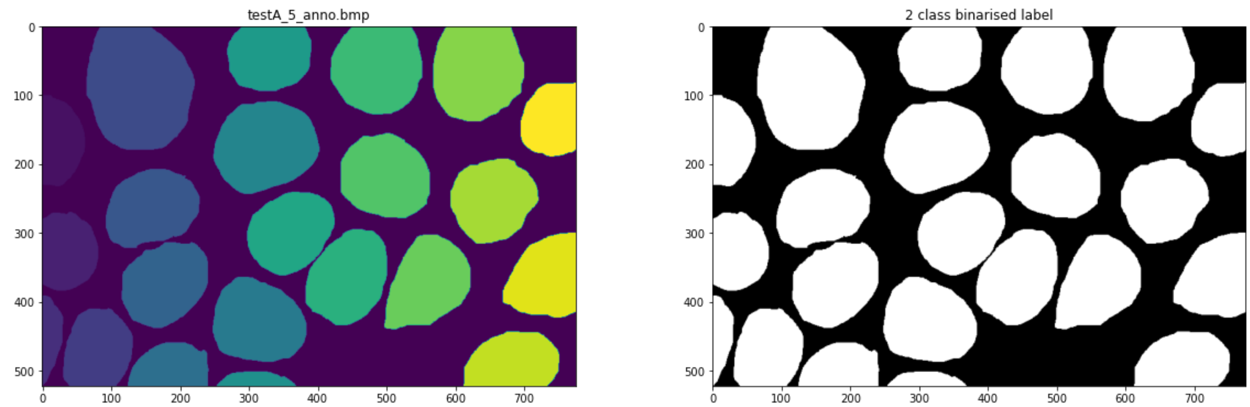


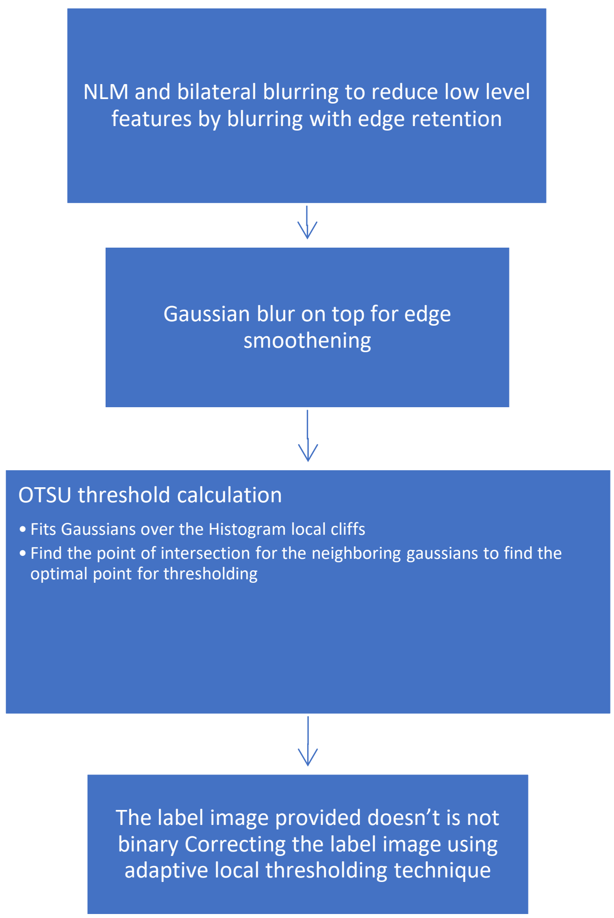
Image 5 annotation: testA_5_anno.bmp



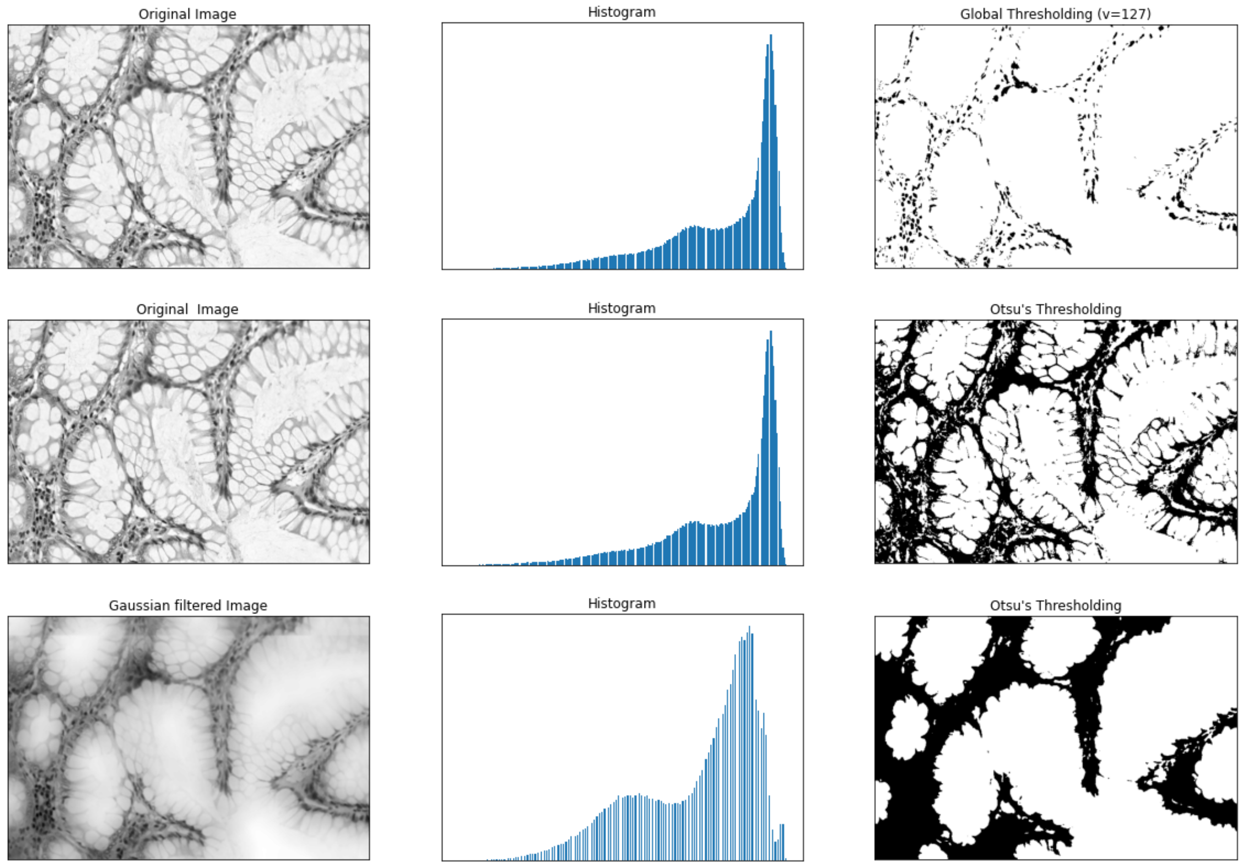
Task 1: Use histogram-based approach to assign the class label for the given images

Procedure:

1. NLM and bilateral blurring to reduce low level features by blurring with edge retention
2. Gaussian blur on top for edge smoothening
3. OTSU threshold calculation
    a. Fits Gaussians over the Histogram local cliffs
    b. Find the point of intersection for the neighboring gaussians to find the optimal point for thresholding
4. The label image provided doesn't is not binary Correcting the label image using adaptive local thresholding technique

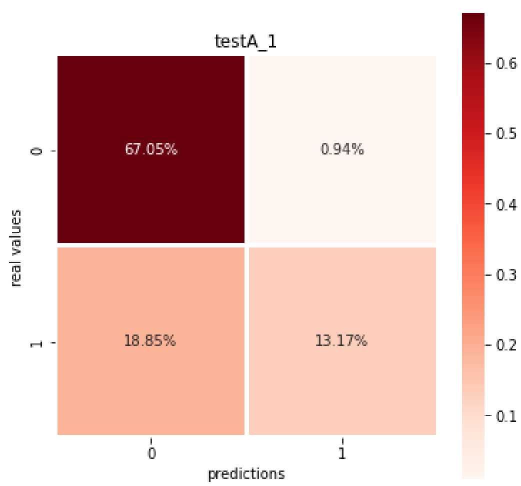


Task 1 - Results for Image 1 : testA_1.bmp



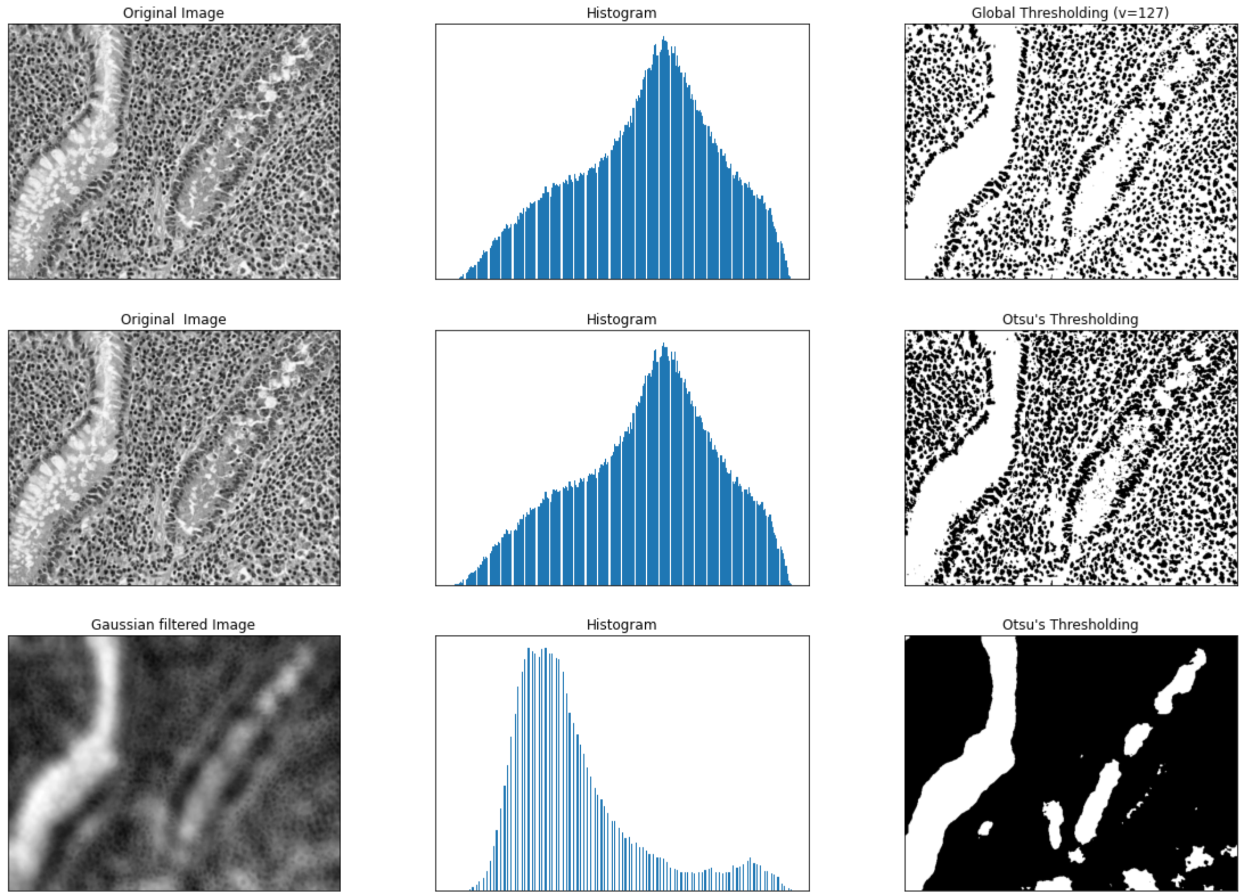

```
Otsu's algorithm implementation thresholding result:  206.12890625
```

Task 3 – Task 1 Evaluation for Histogram based Segmentation of Image 1 : testA_1.bmp



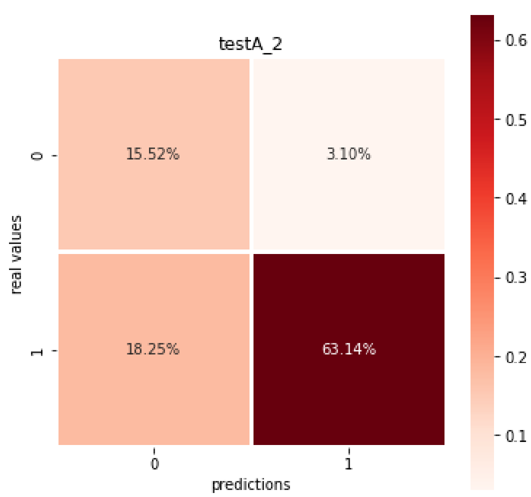

```
IoU score:  0.7721387822958062
F1 score: 0.8714202183369637
Precision: 0.7805759639928975
Sensitivity: 0.9861945847286001
Specificity: 0.41127917660775365
Accuracy: 0.802140650105055
```

Task 1 - Results for Image 2 : testA_2.bmp



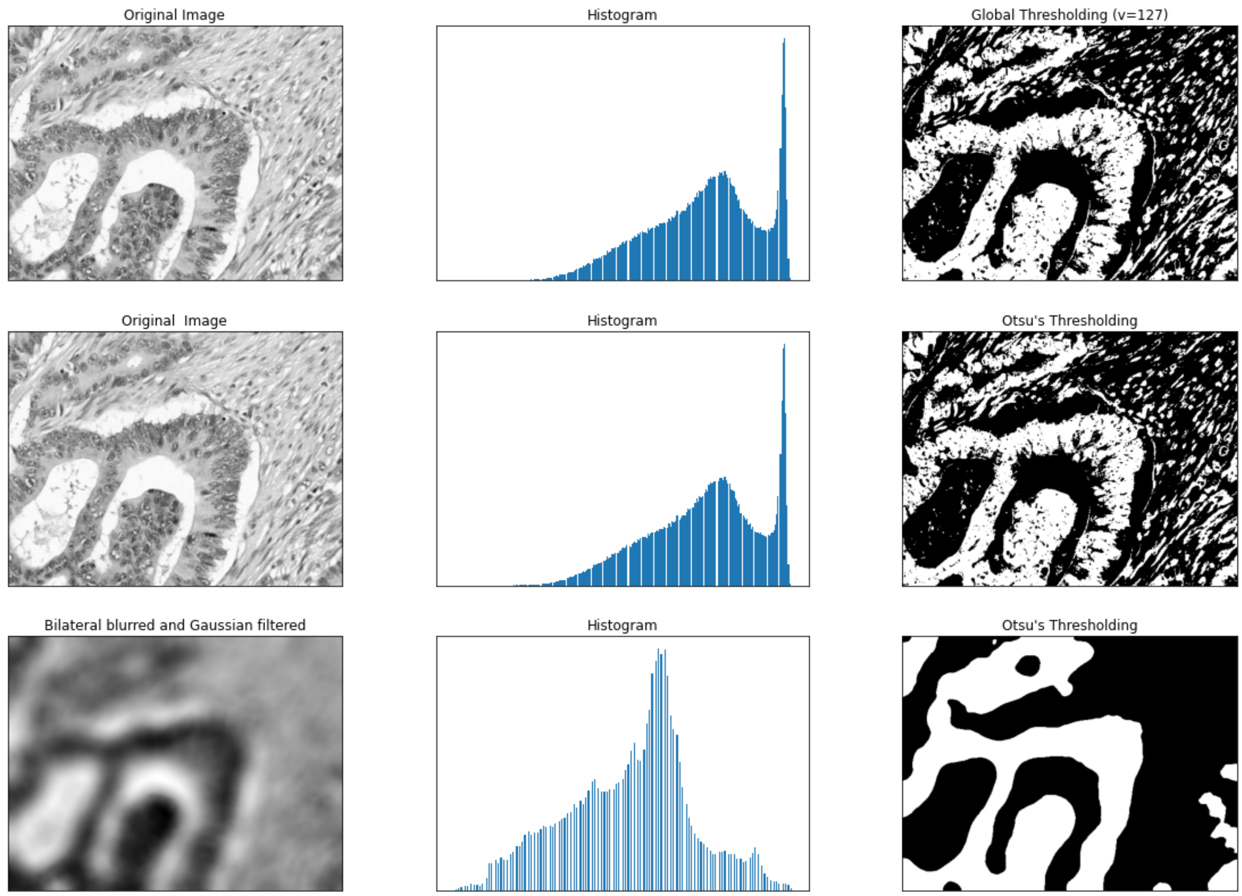

```
Otsu's algorithm implementation thresholding result:  142.908203125
```

Task 3 – Part 1 Evaluation for Histogram based Segmentation of Image 2 : testA_2.bmp



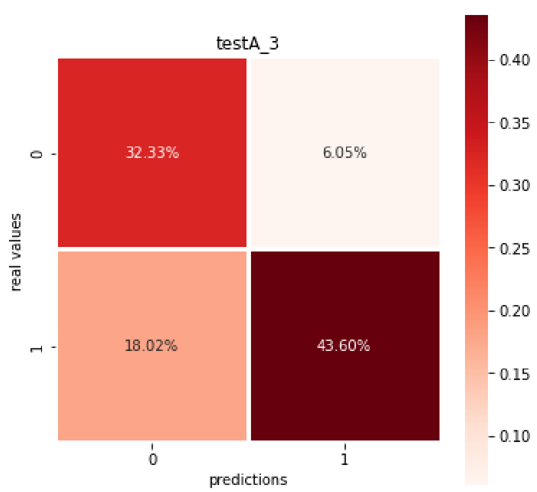

```
IoU score:  0.4373546450167026
F1 score: 0.6085549541068487
Precision: 0.4592500499522678
Sensitivity: 0.9017043720849135
Specificity: 0.7795098105777717
Accuracy: 0.8005224554657312
```

Task 1 - Results for Image 3 : testA_3.bmp



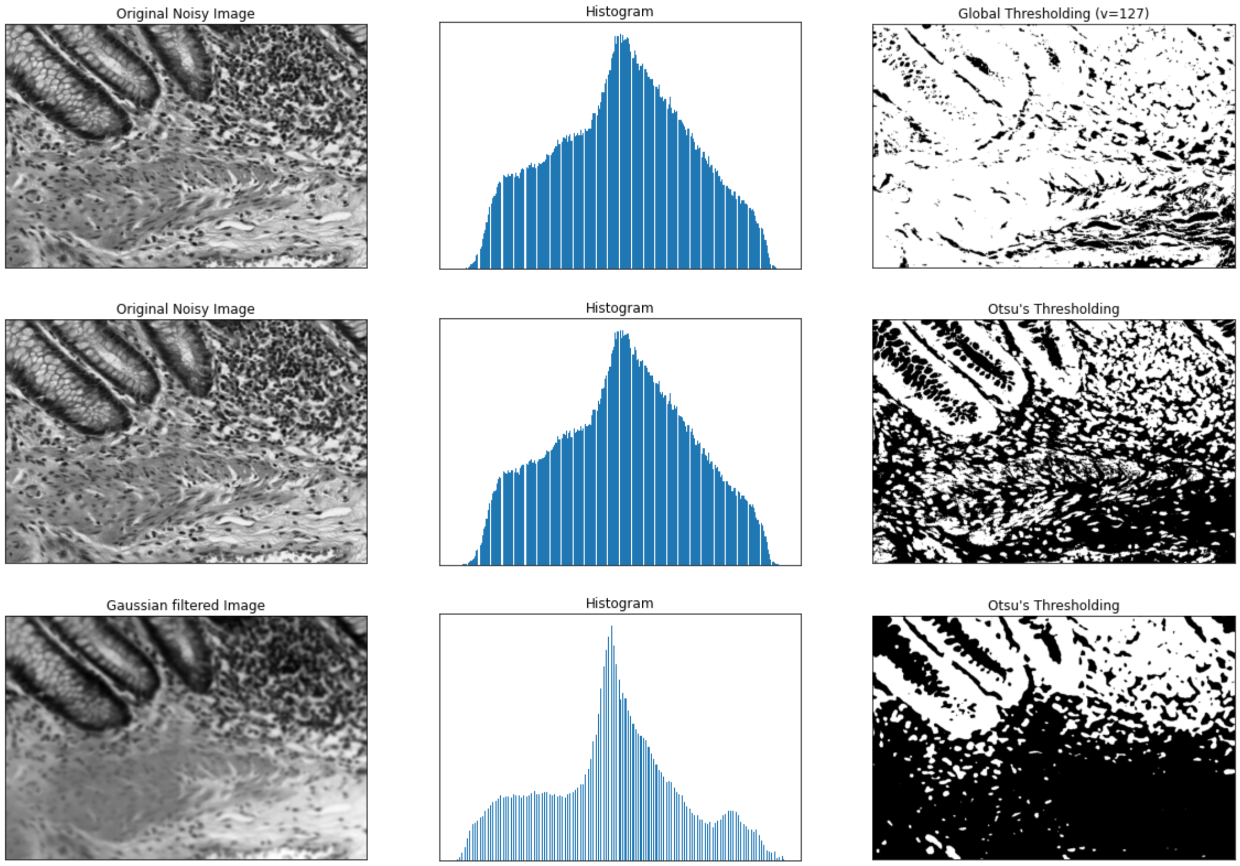

```
Otsu's algorithm implementation thresholding result:  192.03125
```

Task 3 – Task 1 Evaluation for Histogram based Segmentation of Image 4 : testA_4.bmp



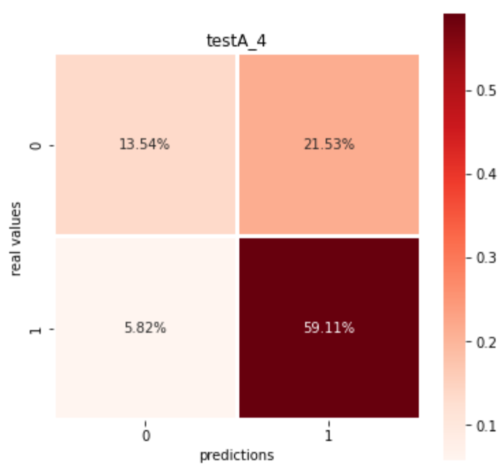

```
IoU score:  0.573213755583632
F1 score: 0.7287169382408314
Precision: 0.6420893075340665
Sensitivity: 0.8423647819161348
Specificity: 0.7075310132141887
Accuracy: 0.7592814697704846
```

Task 1 - Results for Image 4 : testA_4.bmp



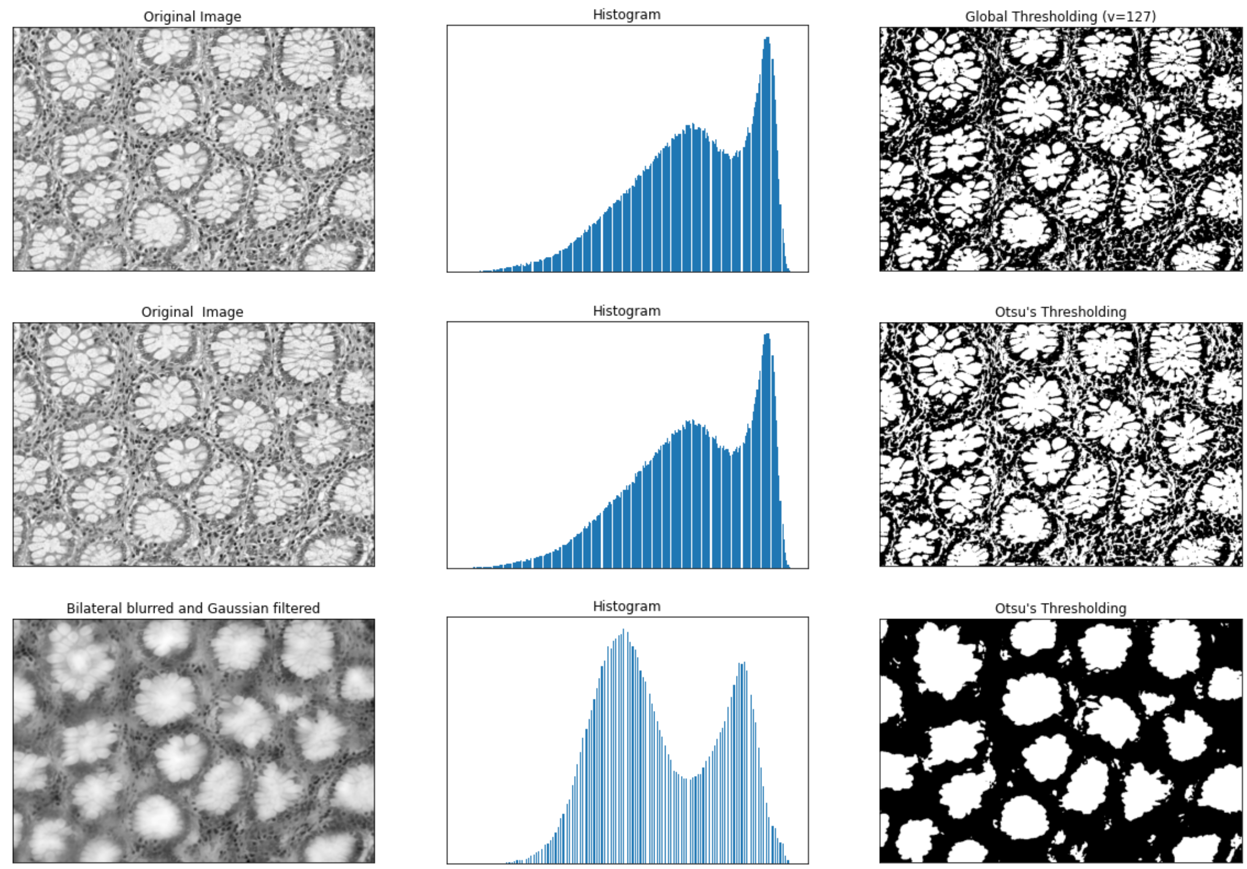

```
Otsu's algorithm implementation thresholding result:  137.83984375
```

Task 3 – Part 1 Evaluation for Histogram based Segmentation of Image 4 : testA_4.bmp



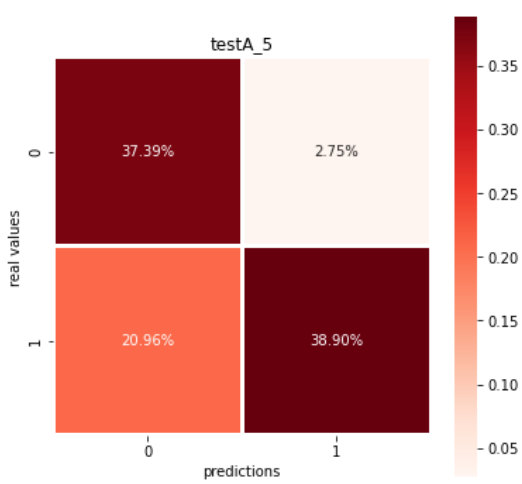

```
IoU score:  0.3312616005949178
F1 score: 0.49766567359395436
Precision: 0.6995965167649838
Sensitivity: 0.3861947925624145
Specificity: 0.9104333856157395
Accuracy: 0.7265850945494995
```

Task 1 - Results for Image 5 : testA_5.bmp




```
Otsu's algorithm implementation thresholding result:  193.171875
```

Task 3 – Task 1 Evaluation for Histogram based Segmentation of Image 5 : testA_5.bmp




```
IoU score:  0.612027672141759
F1 score: 0.7593265087423864
Precision: 0.640822965777535
Sensitivity: 0.9316020888242974
Specificity: 0.6498542298131004
Accuracy: 0.762948955629712
```

Task 2: Graph cut using Energy minimization algorithms: alpha-expansion and alpha-beta swap

Method:

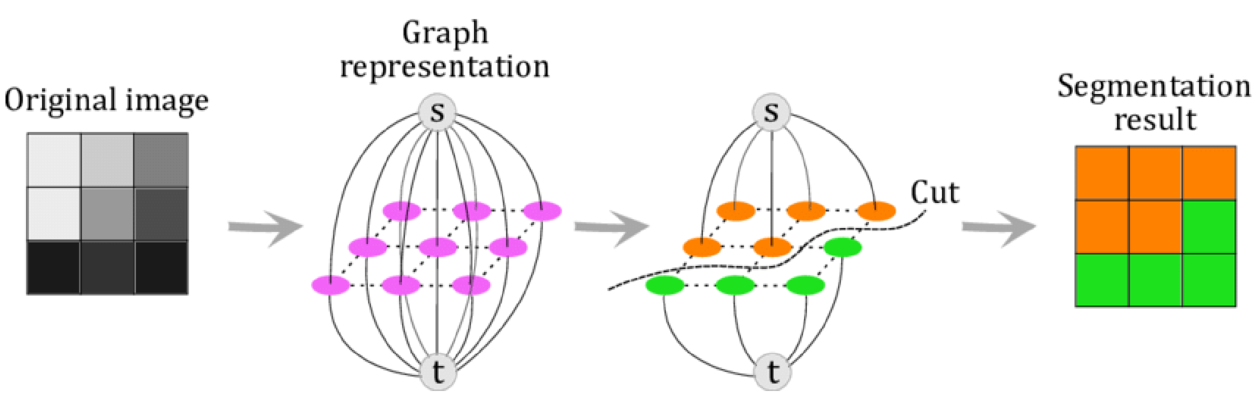
Graph Cuts finds the optimal solution to a binary problem. However, when each pixel can be assigned many labels, finding the solution can be computationally expensive. For the following type of energy, a series of graph cuts can be used to find a convenient local minimum:

$$E(f) = \sum_{p \in \mathcal{P}} D_p(i_p, f_p) + \sum_{p,q \in \mathcal{N}} V_{p,q}(f_p, f_q)$$

- The first term is known as the data term. It ensures that the current labeling **f** is coherent with the observed data i. It penalizes a label $\mathbf{f_p}$ to pixel **p** if it is too different with the observed data $\mathbf{i_p}$.
- The second term is known as the smooth term. It ensures that the overall labeling **f** is smooth. It penalizes two neighboring labels $\mathbf{f_p}$ and **fq** if they are too different.
- On this smoothing term, there are 3 constraints:

1. $V(\alpha, \beta) = 0 \Leftrightarrow \alpha = \beta$, or $V(\alpha, \beta) \neq 0 \Leftrightarrow \alpha \neq \beta$
2. $V(\alpha, \beta) = V(\beta, \alpha) \geq 0$
3. $V(\alpha, \beta) \leq V(\alpha, \gamma) + V(\gamma, \beta)$

- The first two terms tell that an energy between two different labels α and β should be non-zero. If it is zero, that means the two labels are the same.
- The last term defines the triangle rule. A shortcut is always cheaper or similar than taking the whole path.
- If the smooth term only satisfies the first two terms, it is said a semi-metric term. If the last term is also satisfied, it is said a metric term.
- The alpha-expansion algorithm can only be used with metric term. Otherwise, the alpha-beta swap can be used with semi-metric and metric term.

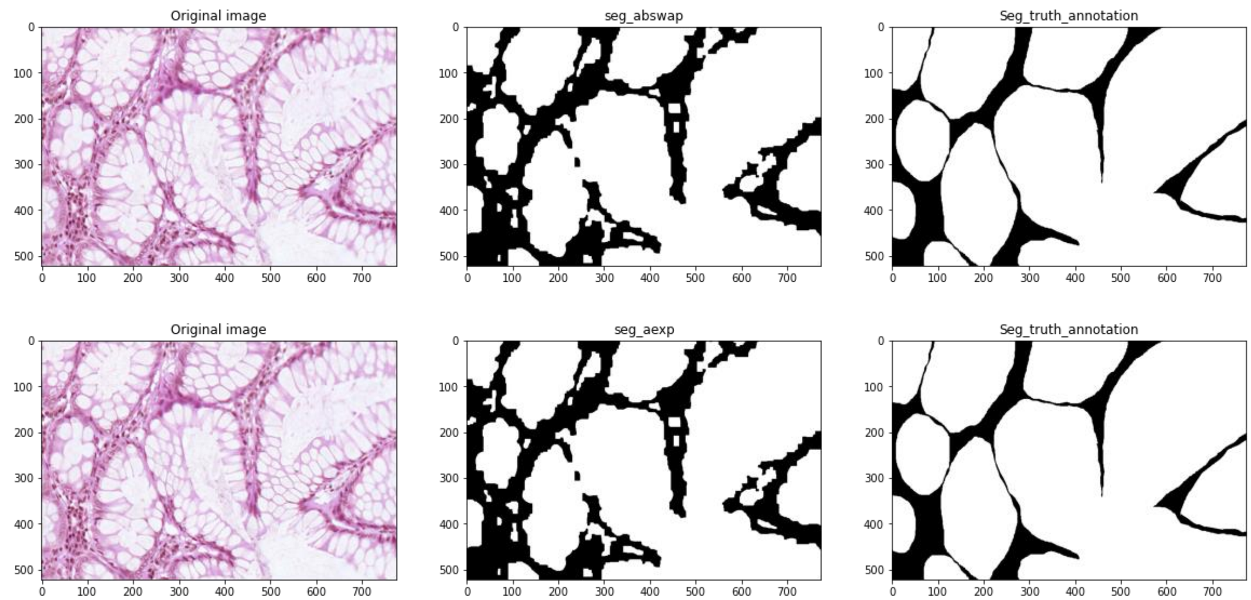


A graph is generated from this pixel distribution where the pixels are considered as nodes and two additional nodes are added that is the Source node and Sink node. All the foreground pixels are connected to the Source node and every Background pixel is connected to the Sink node. The weights of edges connecting pixels to the Source node and to the End node are defined by the probability of a pixel being in the foreground or in the background.

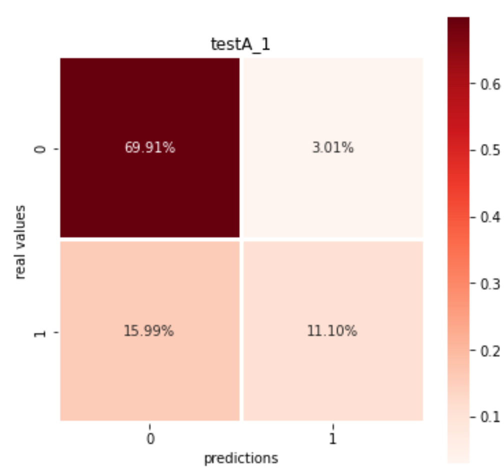
If huge dissimilarity is found in pixel color, the low weight is assigned to that edge. Then the algorithm is applied to segment the graph. The algorithm segments the graph into two, separating the source node and the sink node with the help of a cost function which is the sum of all weights of the edges that are segmented.

After the segmentation, the pixels that are connected to the Source node is labeled as foreground and those pixels which are connected to the Sink node is labeled as background. This process is done for multiple iterations as specified by the user. This gives us the extracted foreground.

Task 2 - Results for Image 1 : testA_1.bmp

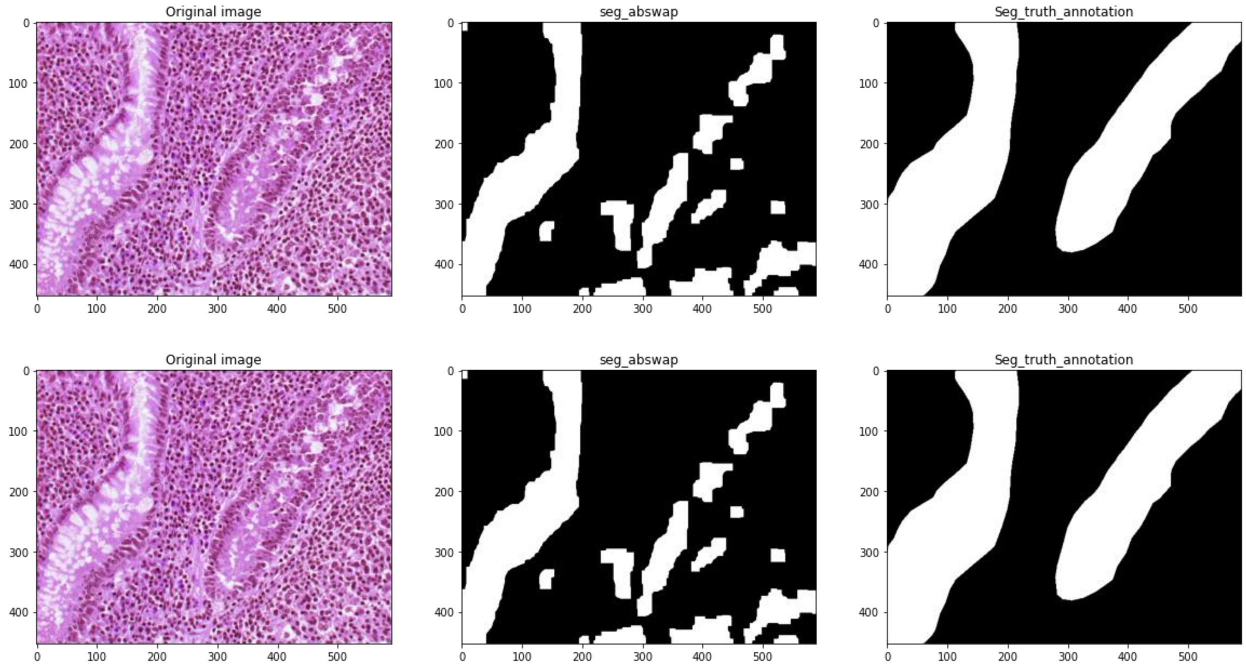


Task 3 – Task 2 2 Evaluation for Energy min-based Graph-cut Seg of Image 1: testA_1.bmp

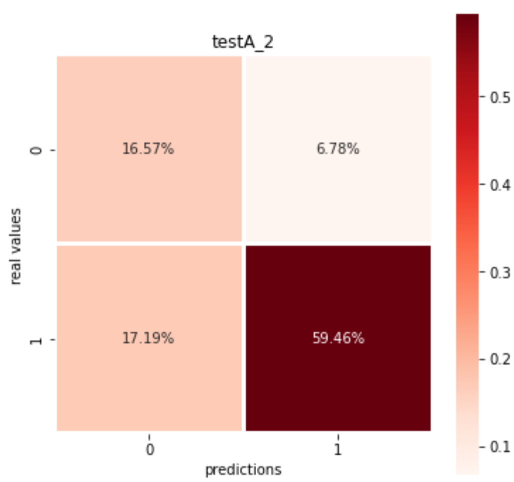



```
IoU score:  0.7863661135332325
F1 score: 0.880408677231218
Precision: 0.8138865626627759
Sensitivity: 0.9587729171186808
Specificity: 0.4097869932648256
Accuracy: 0.8100778642936596
```

Task 2 - Results for Image 2 : testA_2.bmp

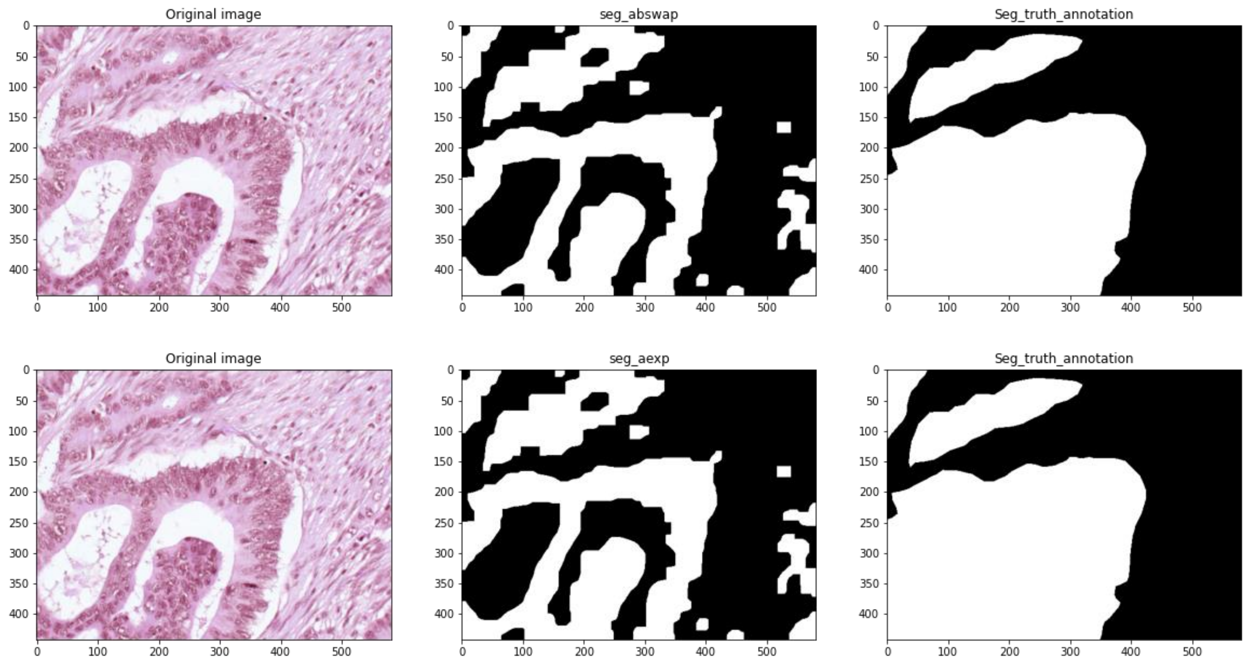


Task 3 – Task 2 Evaluation for Energy min-based Graph-cut Seg of Image 1: testA_1.bmp

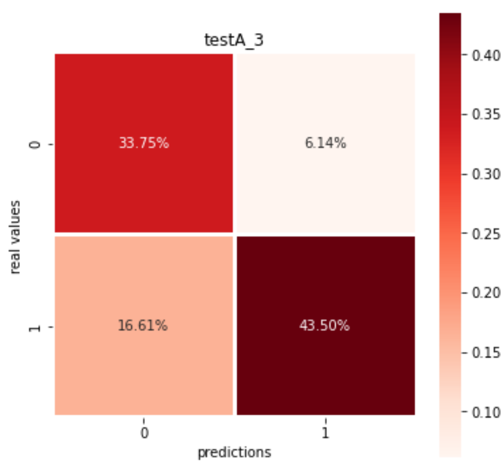



```
IoU score:  0.21606749007498896
F1 score: 0.3553544385298305
Precision: 0.21749217414470617
Sensitivity: 0.9705751226036559
Specificity: 0.7141750800794713
Accuracy: 0.7335739476869915
```

Task 2 - Results for Image 3 : testA_3.bmp

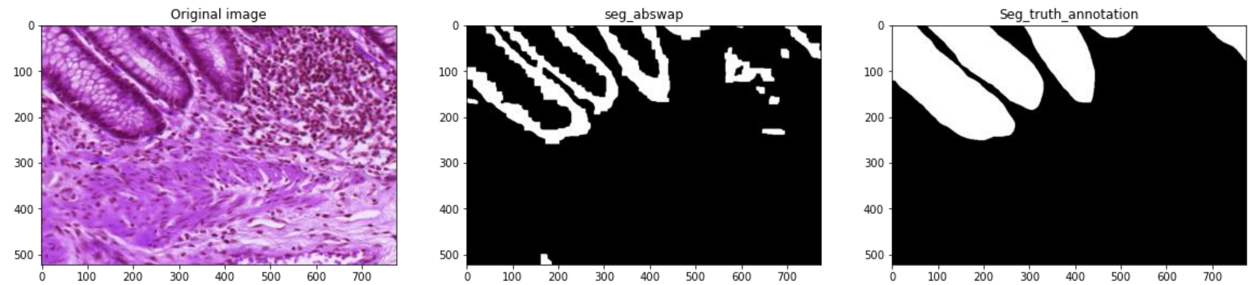


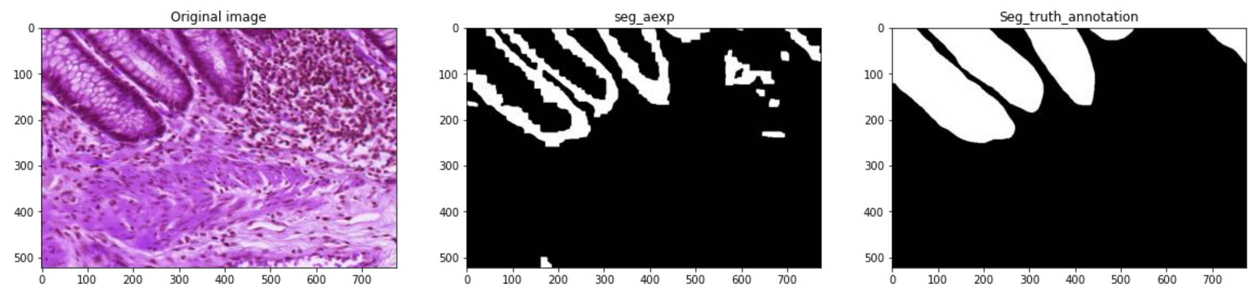
Task 3 – Task 2 2 Evaluation for Energy min-based Graph-cut Seg of Image 3: testA_3.bmp



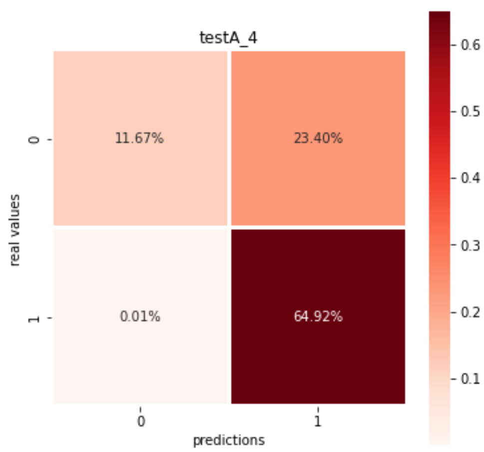

```
IoU score:  0.59731192059827
 F1 score: 0.7478964038214252
Precision: 0.6702009187508701
 Sensitivity: 0.845968371729793
Specificity: 0.723733820499864
 Accuracy: 0.7724939836917158
```

Task 2 - Results for Image 4 : testA_4.bmp



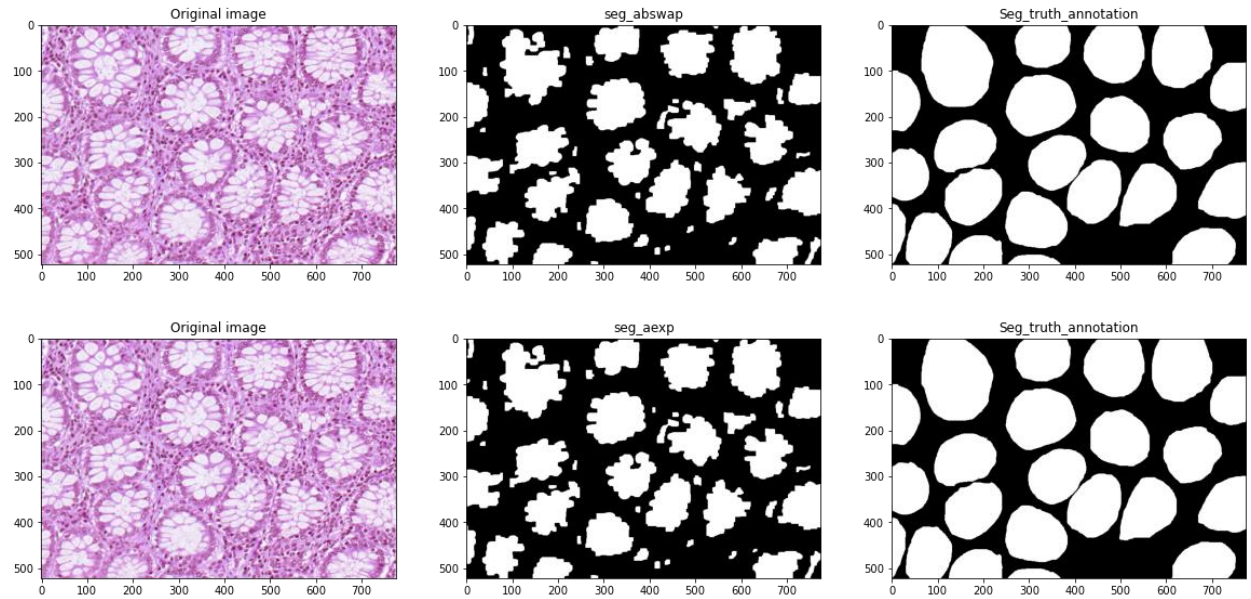


Task 3 – Task 2 2 Evaluation for Energy min-based Graph-cut Seg of Image 4: testA_4.bmp

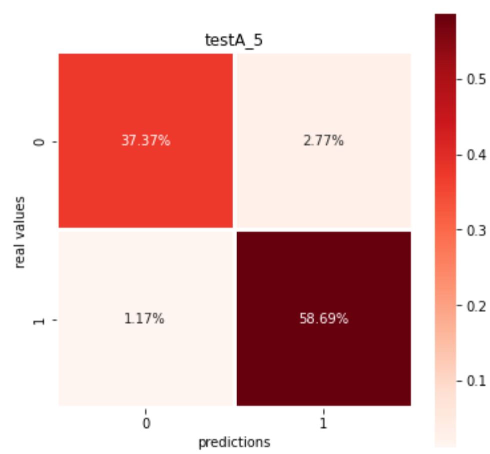



```
IoU score:  0.3326474961069342
F1 score: 0.49922803603908467
Precision: 0.9990476996656368
Sensitivity: 0.3327530061885899
Specificity: 0.9998286862903348
Accuracy: 0.7658880237300705
```

Task 2 - Results for Image 5 : testA_5.bmp



Task 3 – Task 2 Evaluation for Energy min-based Graph-cut Seg of Image 5: testA_5.bmp




```
IoU score:  0.9060399479458924
F1 score: 0.9507040489075967
Precision: 0.9673056823106534
Sensitivity: 0.9346626598024484
Specificity: 0.9788158340284603
Accuracy: 0.9610925719935731
```

Result Comparison - F1 scores

Image:	Histogram based	Graphcut(aswpa & aexp)
testA_1	0.871	**0.880**
testA_2	**0.608**	0.355
testA_3	0.728	**0.748**
testA_4	0.497	**0.499**
testA_5	0.759	**0.951**

Conclusion:

Task 1:

1. Histogram based Segmentation worked appropriately only for Image 1 and Image 5.
2. Histogram based segmentation Doesn't work when there was pixel intensity overlaps between Object(foreground) and background.
3. Edge retaining Blurring methods like NLM, and Bilateral blurring helps in the above mentioned scenario by mixing the local regions while retaining the boundaries.

Task 2:

1. Using Graph-cut method, we are able to achieve better segmentation than the histogram-based approach because it considers pixel similarity-based costs to segment the regions.
2. Since we are not providing any seeds, Graph-cut results shows that similar pixels to the background are grouped together even when it lies inside the cells. User labelling or Seeding can solve this problem.
3. Only for the 2nd CT image "testA_2.bmp", OTSU's thresholding (histogram based) segmentation give better result than the graph-cut method.

Task 3:

1. IOU and F1 score are much sensitive parameters than pixel accuracy, precision and sensitivity.
2. Accuracy for histogram-based segmentation was good and comparable to Graph-cut.
3. IOU and F1 were more sensitive for betterment in ROI segmentation and hence saw contrasting improvement in these metrices other than accuracy, sensitivity and specificity.