CSE585/EE555: Digital Image Processing II
Computer Project # 4
Texture Segmentation
Mudit Garg, Mayank Murali, Niranjan Thirusangu
Date: 04/10/2020

A. Objectives

The purpose of this project is to understand and implement Gabor filter and smoothing filter for image filtering using spatial convolution. Following must be implemented for different parameter selection -

1. Implement the Gabor filter with *g* to be a circularly symmetric Gaussian.
2. Implement the smoothing filter.

B. Methods

The algorithms/ methods used in this project for both parts are described below in this section:

Input images include: *'texture1.gif', 'texture2.gif', 'd9d77.gif', 'd4d29.gif'* out of which the first two mentioned are binary-valued images and the latter two have disordered Brodatz textures.

Algorithm Used:

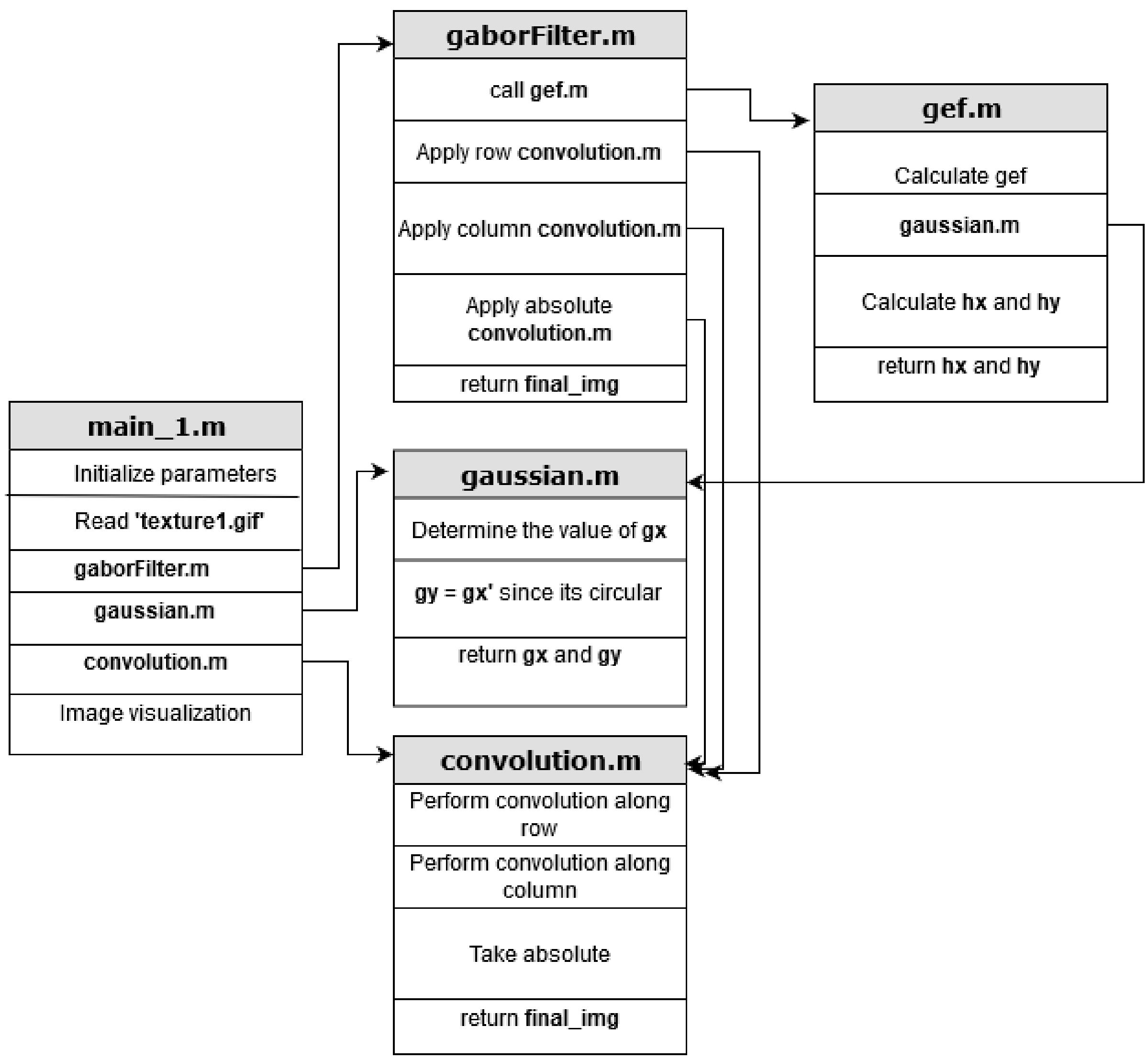
1. Gabor filters is most commonly used for texture segmentation because of their good spatial frequency localization.
2. As per L17-10, Gabor Elementary Function or GEF can be represented as:
   $h(x,y) = g(x,y)\exp\left[j2\pi F(x\,cos\theta + y\,sin\theta))\right] = g(x,y)\exp\left[j2\pi(Ux+Vy)\right]$
   where, $g(x,y)$ is a radially symmetric gaussian and is given by :
   $$g(x,y) = \frac{1}{2\pi\sigma^2}\exp\{-[\frac{x^2+y^2}{2\sigma^2}]\}$$
3. Since $\phi = o°$, $g(x,y)$ is separable as $g(x)$ and $g(y)$, which is given by:
   $g(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{x^2}{2\sigma^2}\}$ & $g(y) = \frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{y^2}{2\sigma^2}\}$

4. As per L18-4, $h(x,y)$ is separable and is given as $h(x,y) = h_1(x) \cdot\ h_2(y)$, where
$$h_1(x) = g(x) \cdot \exp[j2\pi Fx\ cos\theta] = g(x) \cdot \exp\,[j2\pi Ux]$$
$h_2(y)$ is computed in the similar manner.
5. Gabor filter is then applied by convoluting original image $I(x,y)$ with the GEF vectors $h(x,y)$. As given in L18-4, the equation is given as:
$$m(x,y) = |I(x,y) ** h(x,y)|$$
6. To compute the 2D-convolution, we convolute rows first followed by its output convoluting column wise. As per L18-6, the equations are given by:
$$i_1(x,y) = i(x,y) * h_1(x)$$
$$i_2(x,y) = i_1(x,y) * h_2(y)$$
$$m(x,y) = |i_2(x,y)|$$
where,
$$i_1(x,y) = \sum_{x'=x-2\sigma}^{x+2\sigma} i(x-x',y)\ \hat{h}_1(x')$$
Similarly, $i_2(x,y)$ is computed. Finally, taking the absolute values of $i_2\ (x,y)$ yields the image after applying Gabor filter.
7. Sometimes, we might need to smoothen the image obtained with gabor filter. In order to do so, convolute $m(x,y)$ with another gaussian $g'(x,y)$.
$$m'(x,y) = m(x,y) * g'(x,y)$$
8. Finally, the images obtained by filtering (or also smoothened) are scaled up and manually segmented to obtain the different textures. Threshold value is obtained using the histogram of scaled image. Image was normalized between [0,1] using the below formula:
$$I'(x,y) = \frac{I(x,y) - min}{max - min}$$
9. The segmented image is superimposed with the original image to see the different textures.

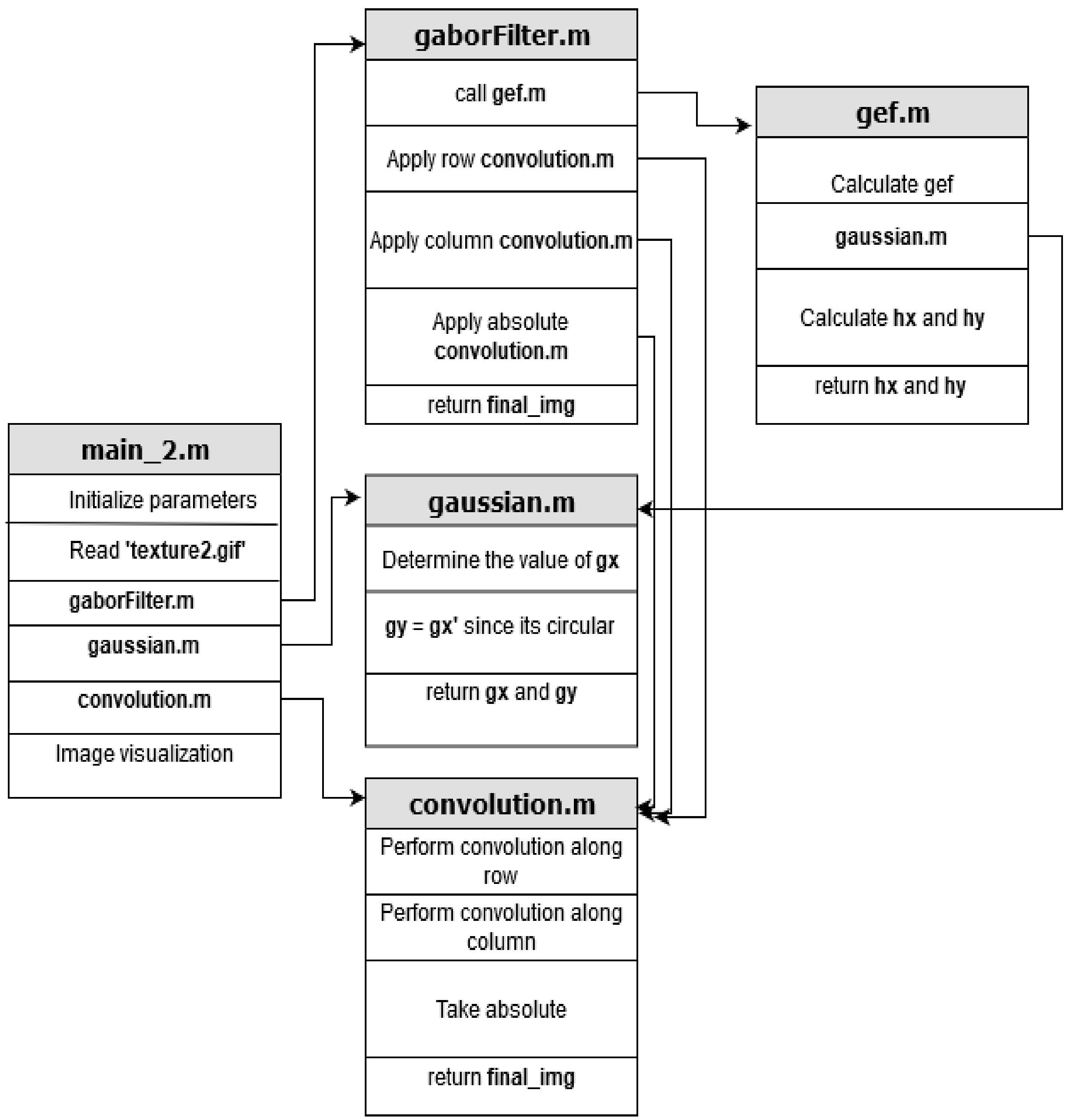
MATLAB code Flow & Execution:

In order to obtain the results which are shown in the next section, we run ***'main_1.m', 'main_2.m', 'main_3.m'*** *and* ***'main_4.m'*** each file corresponding to the 4 questions described in the methods section.

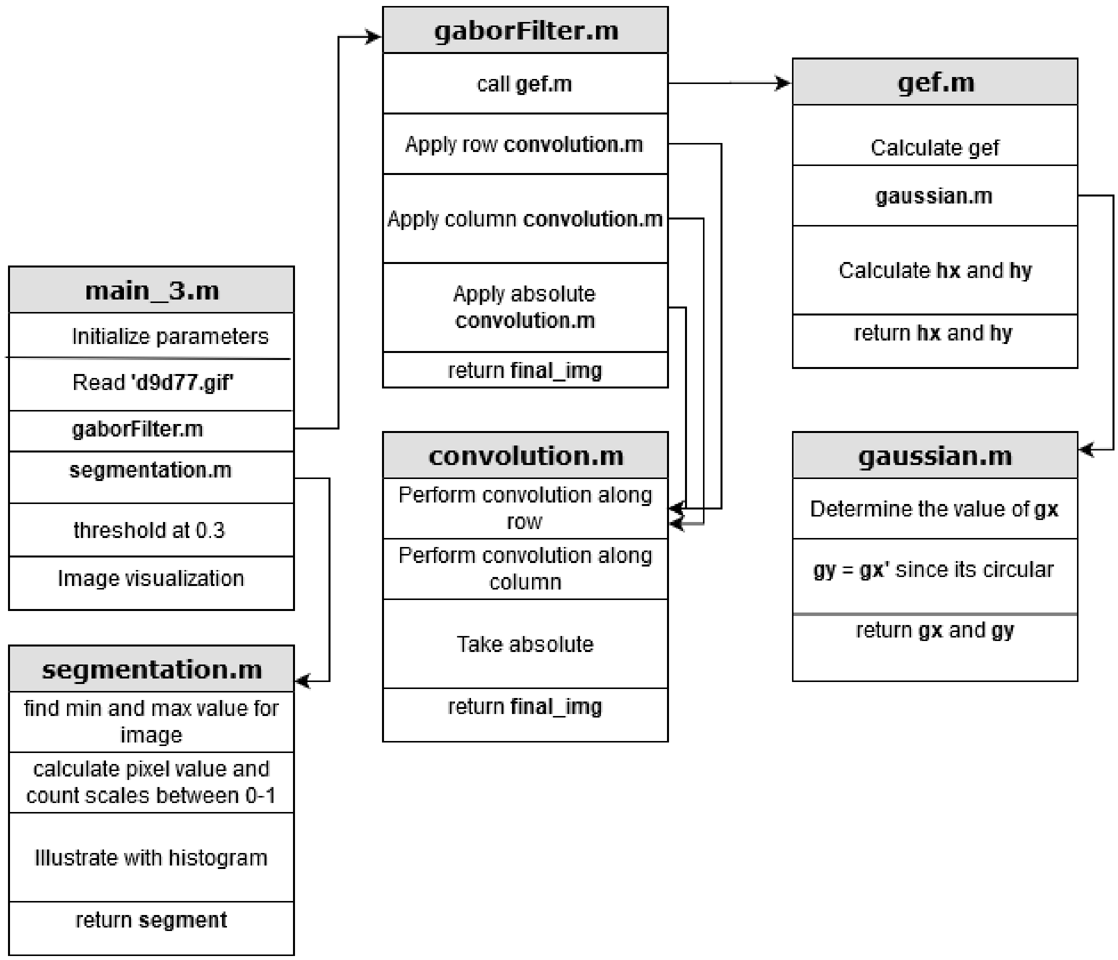
In ***'main_1.m'*** file, we set the initial input parameters ($F = 0.059$, $\theta = 135^{\circ}$ and $\sigma = 8$) for Gabor filter (**'gaborFilter.m'**). We use $\sigma = 24$ for applying smoothing filter (**'gaussian.m'**) over the Gabor filter output. The row and column results are then convoluted (**'convolution.m'**) over the Gabor output. Original image ***'texture1.gif'*** was used as input.



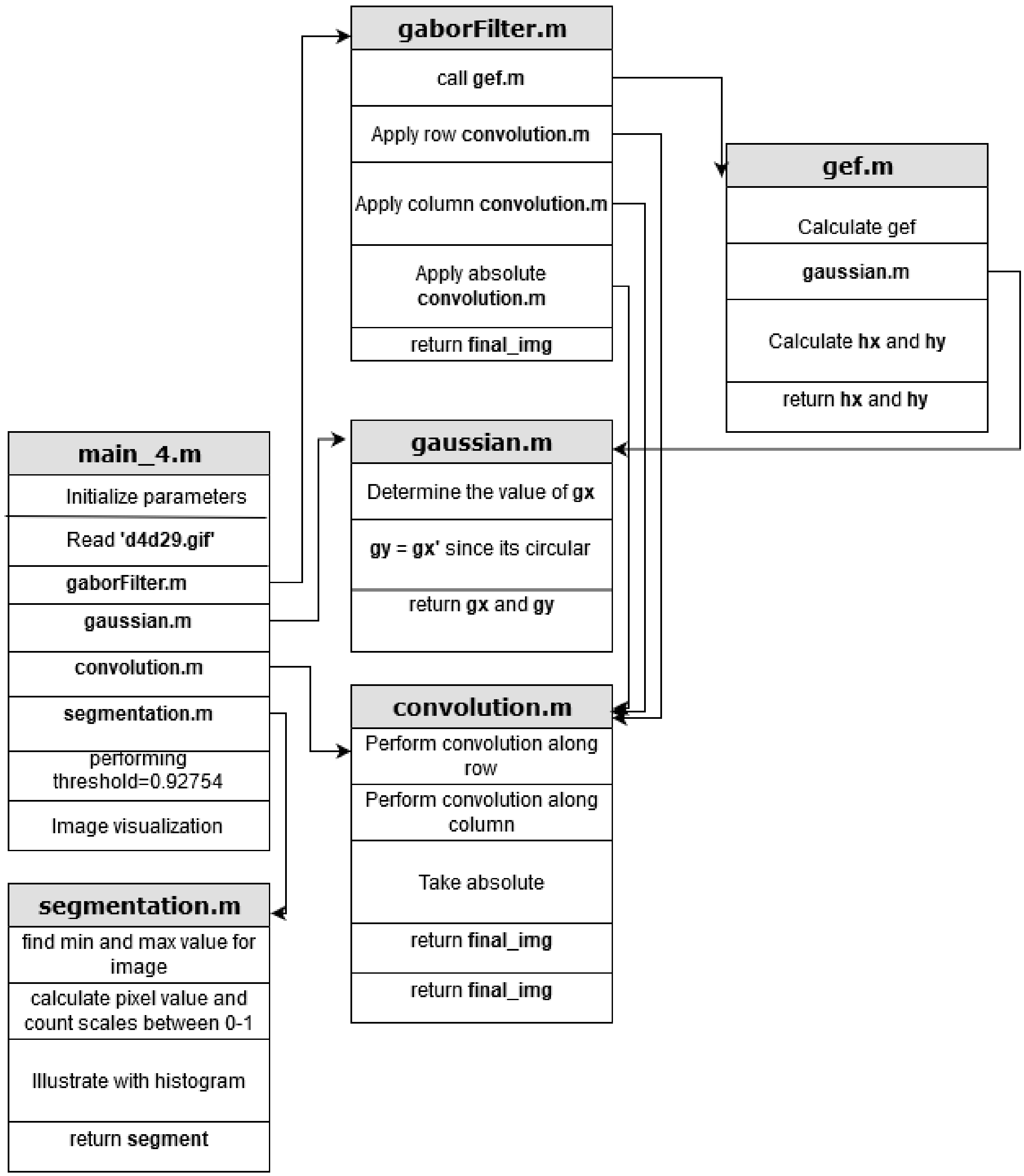
The ***'main_2.m'*** Matlab file inputs the ***'texture2.gif'*** image. Gabor filter (***'gaborFilter.m'***) was applied with the input parameters $F = 0.042$, $\theta = 0^{\circ}$, $\sigma = 24$ and $width = 4\sigma + 1$. Followed by application of smoothing filter (**'gaussian.m'**) with $\sigma = 24$ and convoluting (**'convolution.m'**) the output over row and column.



The ***'main_3.m'*** Matlab file inputs the Brodatz ***'d9d77.gif'*** image. Gabor filter (***'gaborFilter.m'***) is applied using the following parameters $F = 0.063$, $\theta = 60^{\circ}$, $\sigma = 36$ and $width = 4\sigma + 1$. No smoothing was done here.



In ***'main_4.m'*** file inputs Brodatz image ***'d4d29.gif'***. Gabor filter (***'gaborFilter.m'***) is applied using the following parameters $F = 0.6038$, $\theta = -50.5^{\circ}$, $\sigma = 8$ and $width = 4\sigma + 1$. This is followed by application of smoothing filter (**'gaussian.m'**) with $\sigma = 40$.

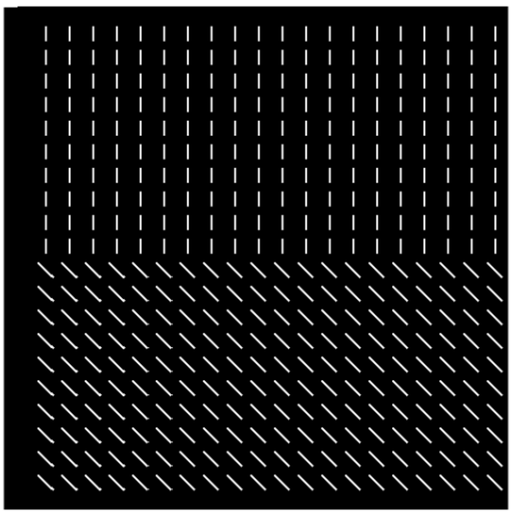


C. Results

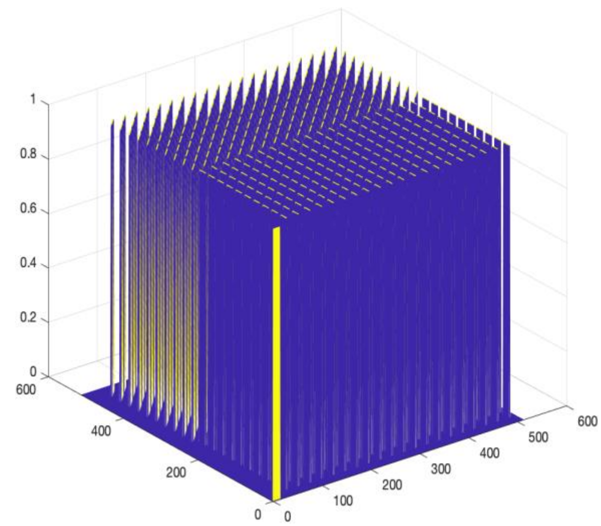
Problem 1:

In this problem, we are given an image '$texture2.gif$', shown in Fig 1(a). We need to apply the gabor filter using the following parameters:

1. F=0.059 cycles/pixel
2. $\theta = 135°$
3. $\sigma = 8$



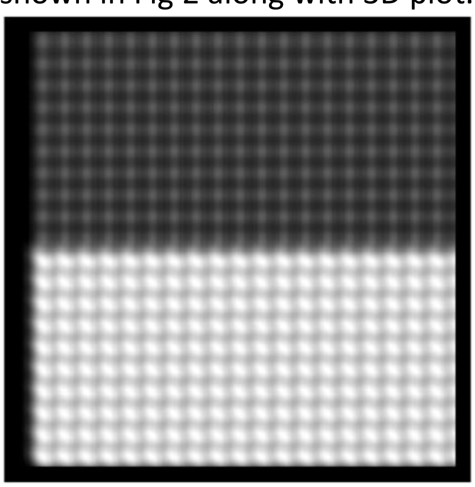
(a) Original Image (texture2.gif)



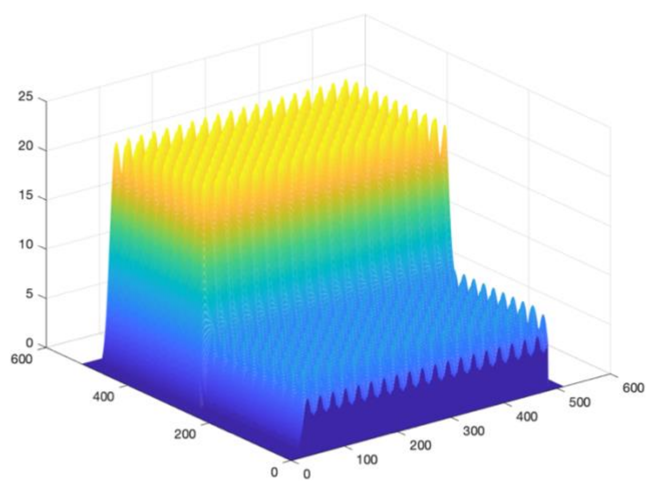
(b) 3D- plot

Figure 1: Original Images

We applied gabor filter using the parameters given in the problem. The results obtained are shown in Fig 2 along with 3D plot.



(a) Image after applying gabor filter

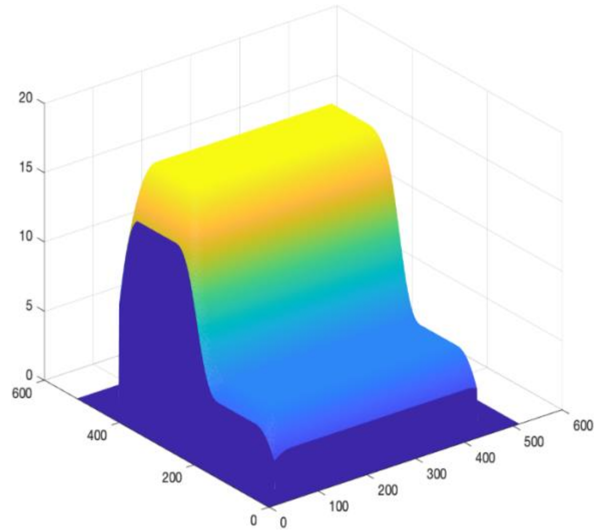


(b) 3D- plot

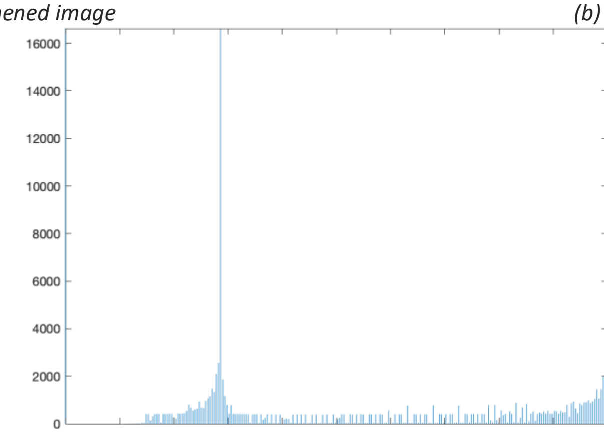
Figure 2: Image after applying gabor filter with F=0.059, $\theta = 135°$ and $\sigma = 8$

The image obtained is fine, but it can further be smoothened. It is done by convoluting it with another gaussian having $\sigma = 24$. The results after smoothening are shown in Fig. 3.

(a) Smoothened image



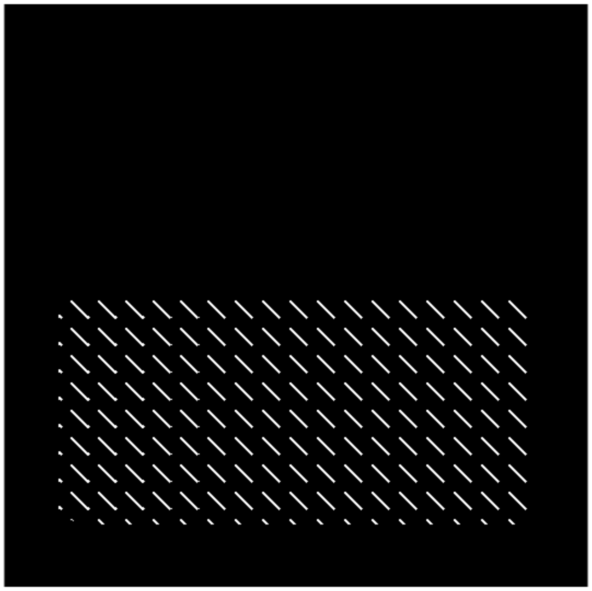
(b) 3D- plot



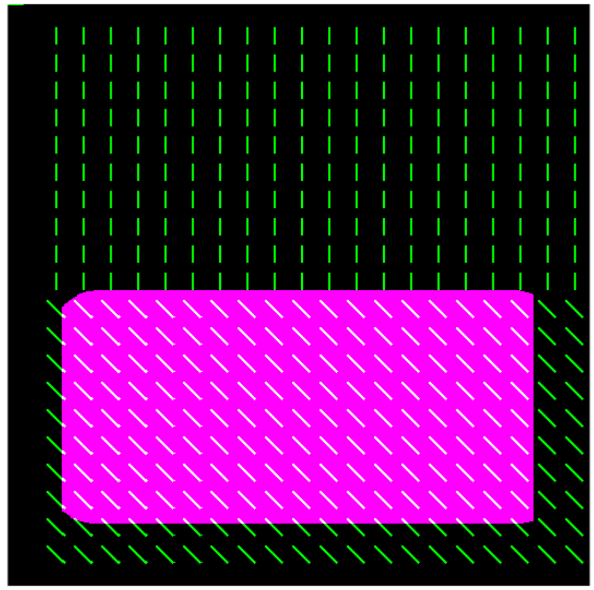
(c) Histogram of smoothened Gabor filter image

Figure 3: Image after applying smoothing filter with $\sigma = 24$

The histogram of the image shown Figure 3(c) was obtained and the **threshold value was found to be 0.6.** The segmented image was superimposed with the original image. The results are shown in Fig. 4.



(a) Superimposed segmented image with original image



(b) Fused segmented and original image

Figure 4: Superimposed original and segmented images

The bounding box for the segmented portion was found to be (49,253), (49,457), (463,253) and (463,457). The bounding box is shown on the original image in Fig 5.

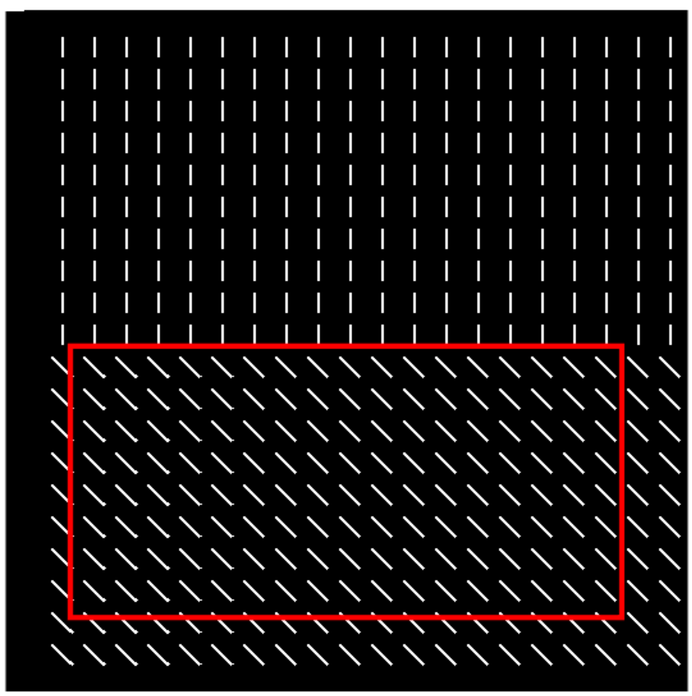


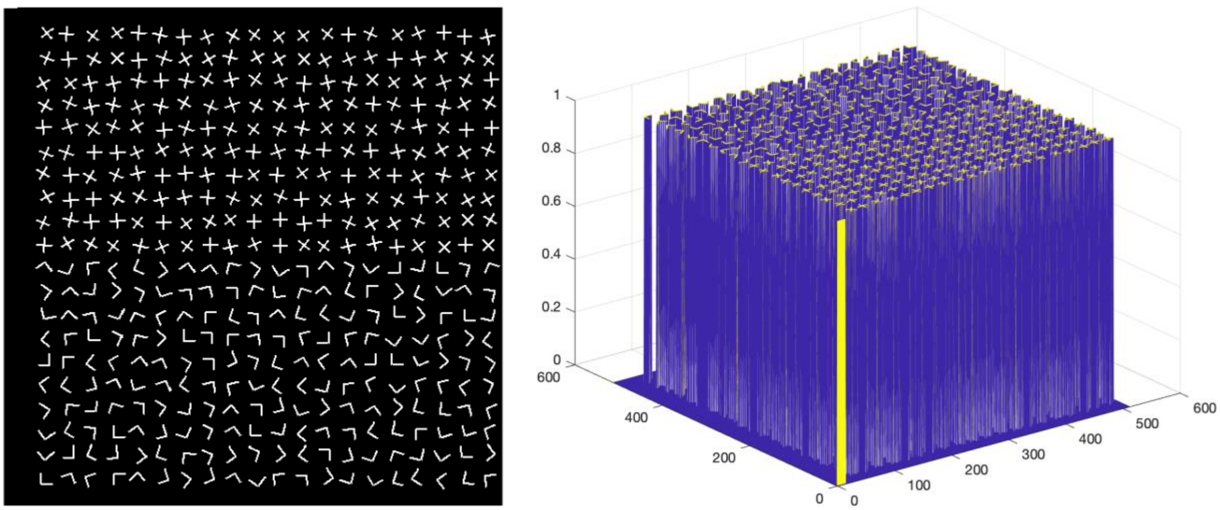
Figure 5: Segmented portion bounding box

Problem 2:

In this problem, we are given an image '$texture1.gif$', shown in Fig 6(a). We need to apply the gabor filter using the following parameters:

1. F=0.042 cycles/pixel
2. $\theta = 0°$
3. $\sigma = 24$

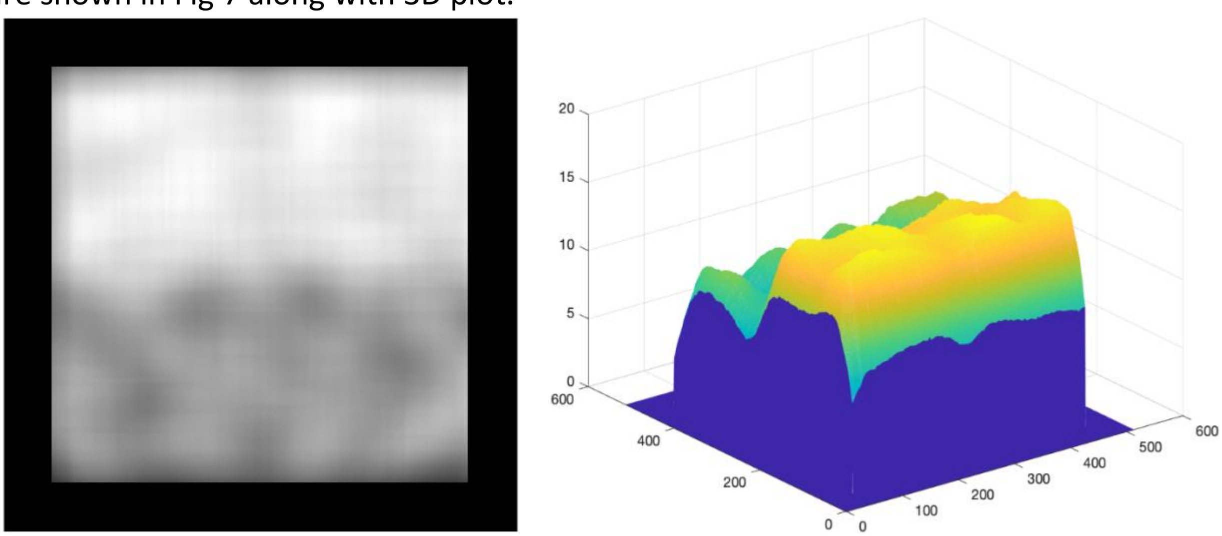
GEF is applied such that its width is $4\sigma + 1$.



(a) Original Image (texture1.gif) *(b) 3D- plot*

Figure 6: Original Images

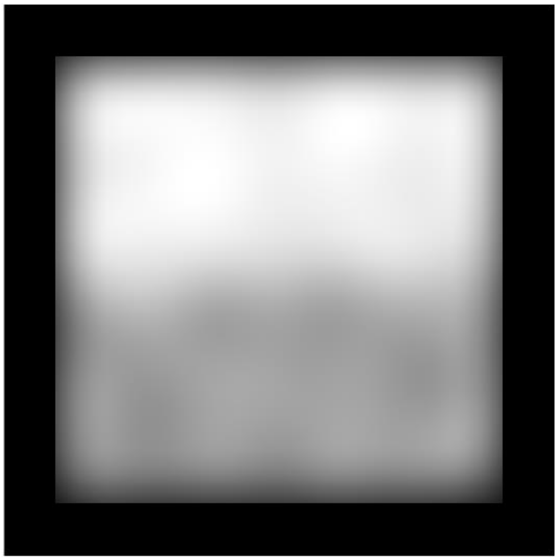
We applied gabor filter using the parameters given in the problem. The results obtained are shown in Fig 7 along with 3D plot.



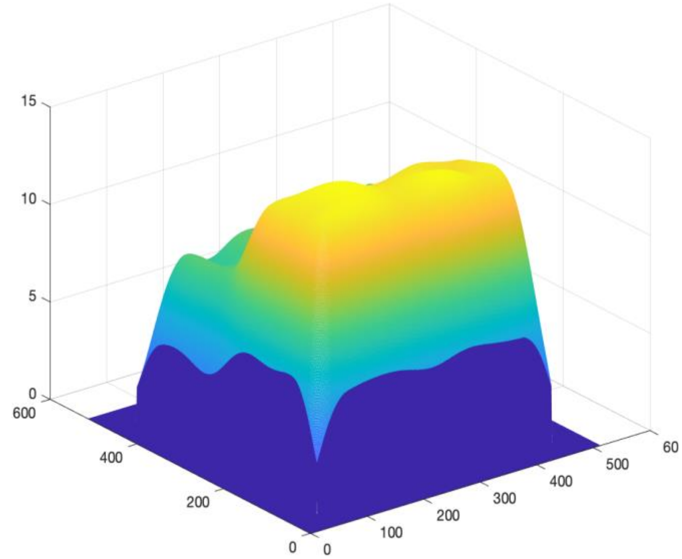
(a) Image after applying gabor filter *(b) 3D- plot*

Figure 7: Image after applying gabor filter with F=0.042, $\theta = 0°$ and $\sigma = 24$

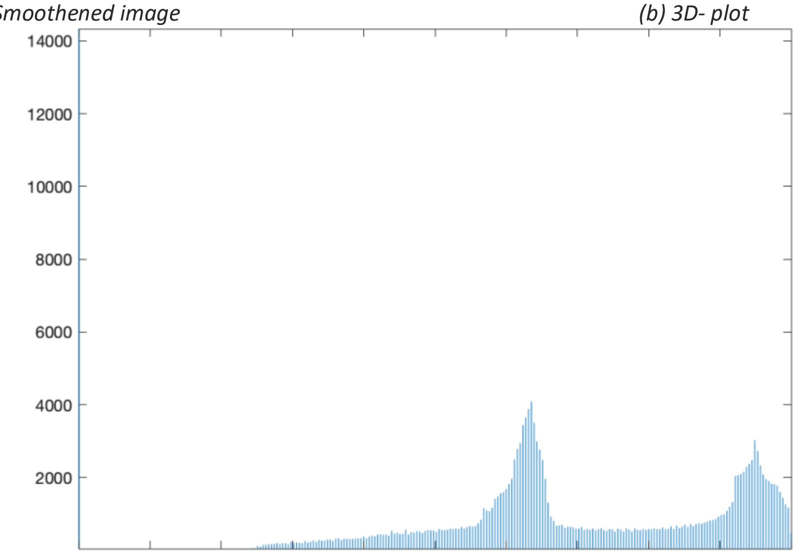
The image obtained is fine, but it can further be smoothened. It was done by convoluting it with another gaussian having $\sigma = 24$. The results after smoothening are shown in Fig. 8.



(a) Smoothened image



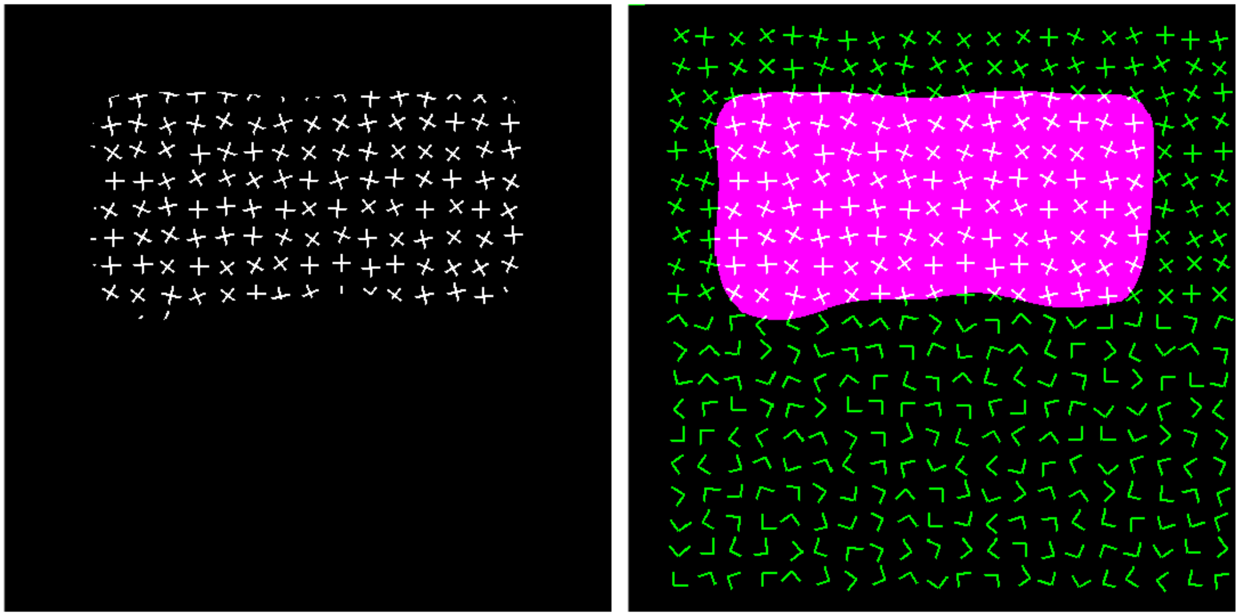
(b) 3D- plot



(c) Histogram of smoothened image

Figure 8: Image after applying smoothing filter with $\sigma = 24$

The histogram of the image shown Figure 8(c) was obtained and the **threshold value was found to be 0.8.** The segmented image was superimposed with the original image. The results are shown in Fig. 9.



(a) Superimposed segmented image with original image (b) Fused segmented and original image

Figure 9-: Superimposed original and segmented images

The bounding box for the segmented portion was found to have x and y limits as (74:442,74:266). The bounding box is shown on the original image in Fig 10.

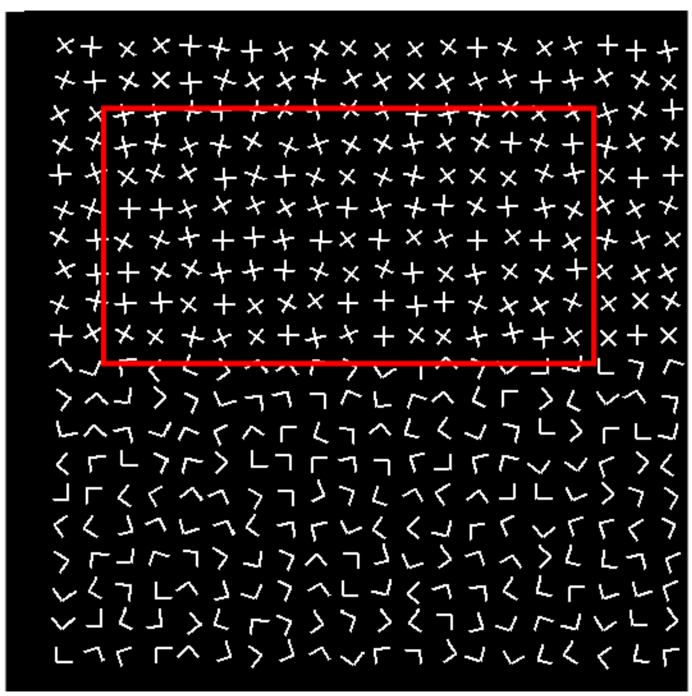


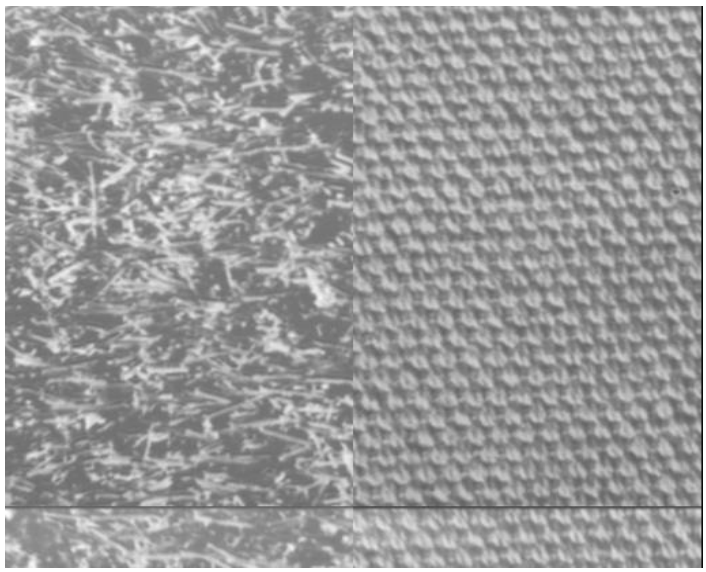
Figure 10: Segmented portion bounding box

Problem 3:

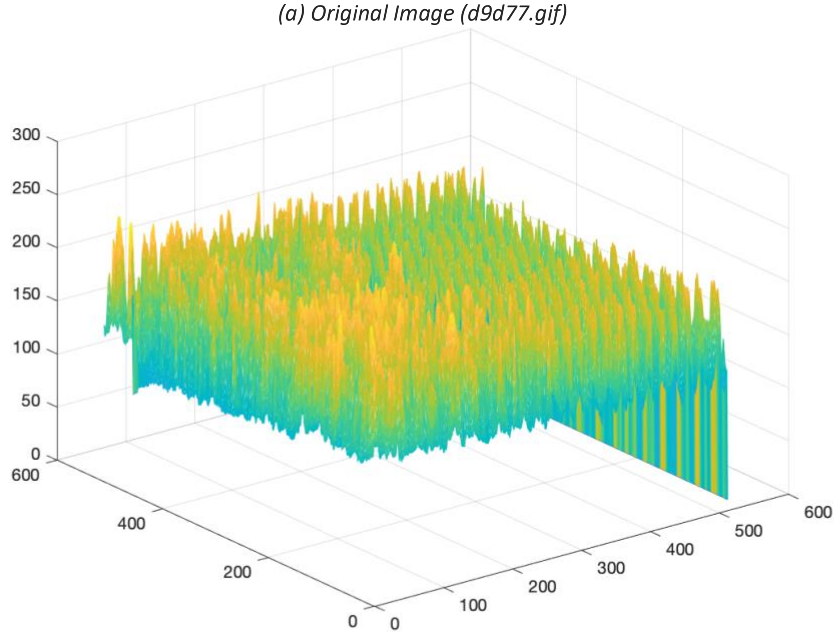
In this problem, we are given an image '$d9d77.gif$', shown in Fig 11(a). We need to apply the gabor filter. Following parameters were selected from L18-9:

1. F=0.063 cycles/pixel
2. $\theta = 60°$
3. $\sigma = 36$

GEF is applied such that its width is $4\sigma + 1$.



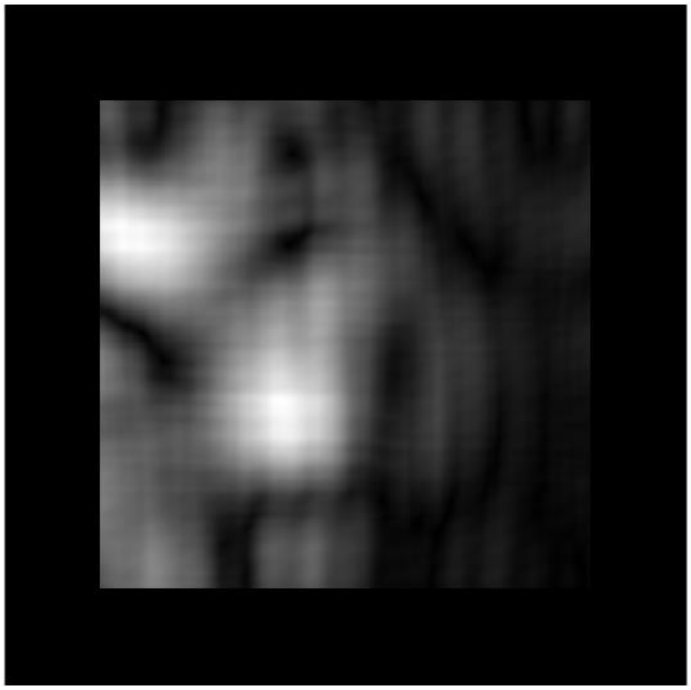
(a) Original Image (d9d77.gif)



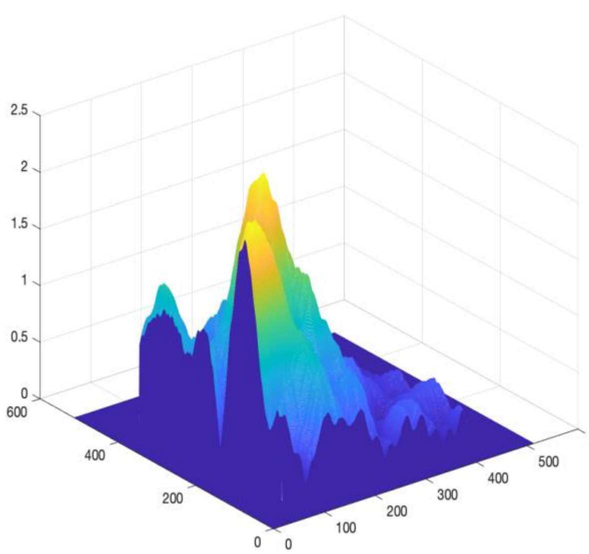
(b) 3D Plot

Figure 11: Original Image

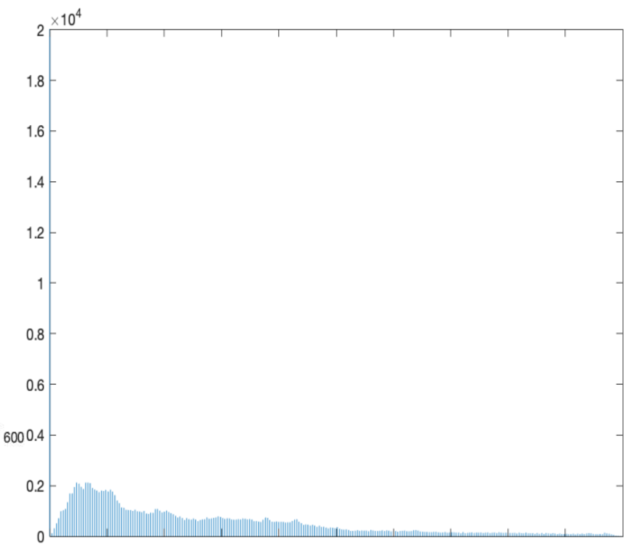
We applied gabor filter using the parameters mentioned. The results obtained are shown in Fig 12 along with 3D plot.



(a) Image after applying gabor filter



(b) 3D Plot

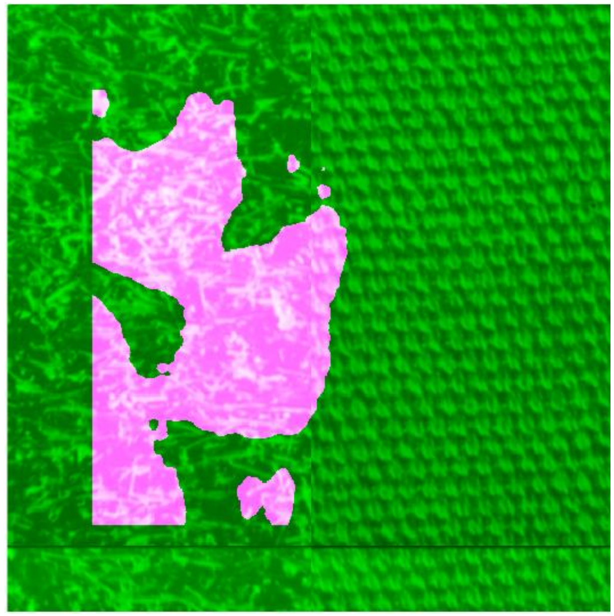


(c) Histogram

Figure 12: Image after applying gabor filter with F=0.063, $\theta = 60°$ and $\sigma = 36$

The histogram of the image shown Figure 10(c) was obtained and the **threshold value was found to be 0.3.** The segmented image was superimposed with the original image. The results are shown in Fig. 13.

(a) Segmentation mask



(b) Fused segmented and original image

Figure 13: Superimposed original and segmented images

No smoothing was done in this problem. The bounding box for the segmented portion was found to have x and y limits as (73:287,73:439). The bounding box is shown on the original image in Fig 14.

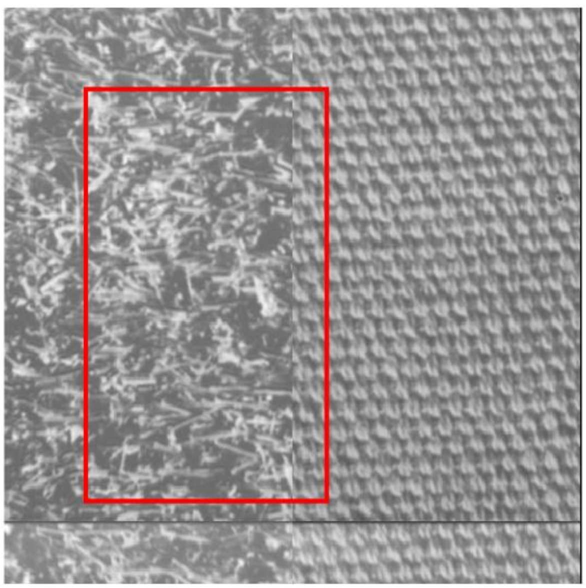


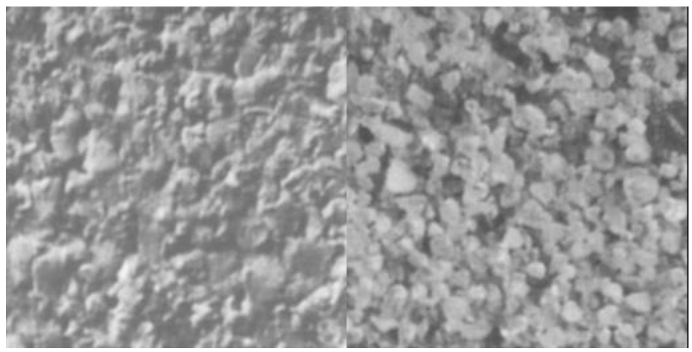
Figure 14: Bounding box of segmented portion

Problem 4:

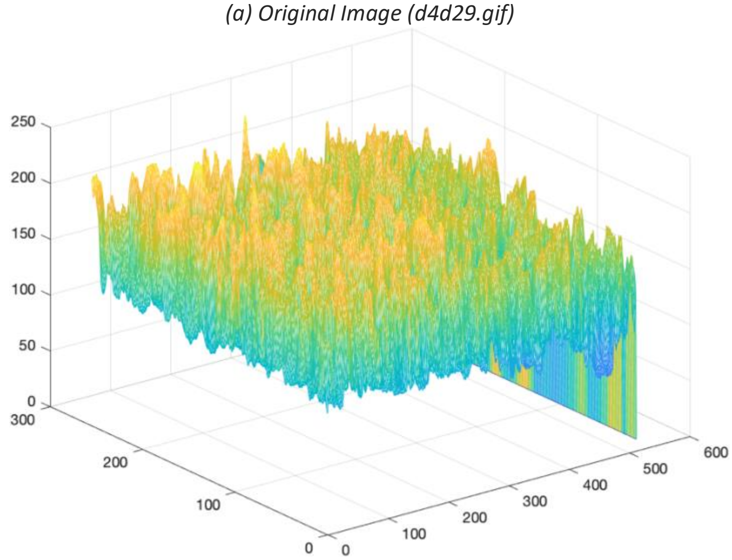
In this problem, we are given an image '$d4d29.gif$', shown in Fig 15(a). We need to apply the gabor filter. Following parameters were selected from L18-8:

1. F= 0.6030 cycles/pixel
2. $\theta = -50.5°$
3. $\sigma = 8$

GEF is applied such that its width is $4\sigma + 1$.



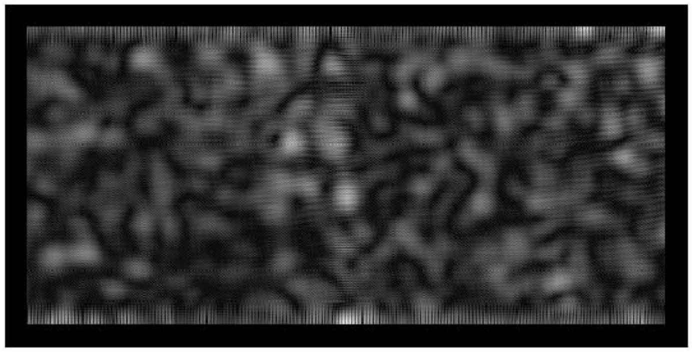
(a) Original Image (d4d29.gif)



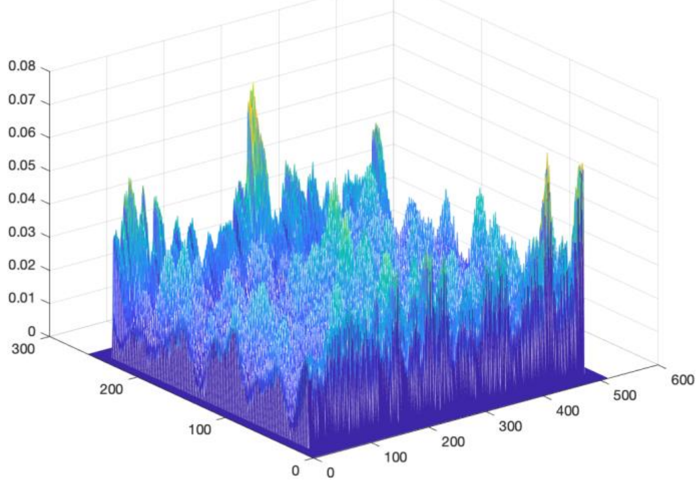
(b) 3D Plot

Figure 15: Original Image

The results obtained after applying Gabor filter are shown in Fig. 16 on next page.



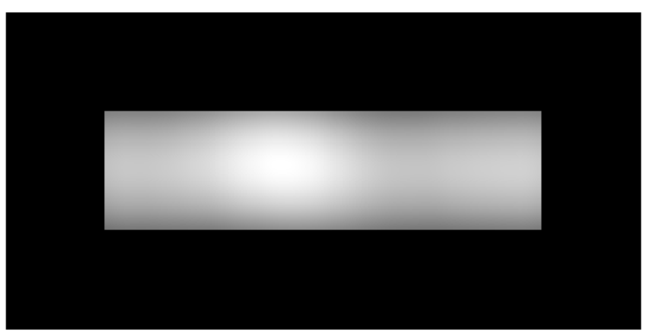
(a) Image after applying Gabor Filter



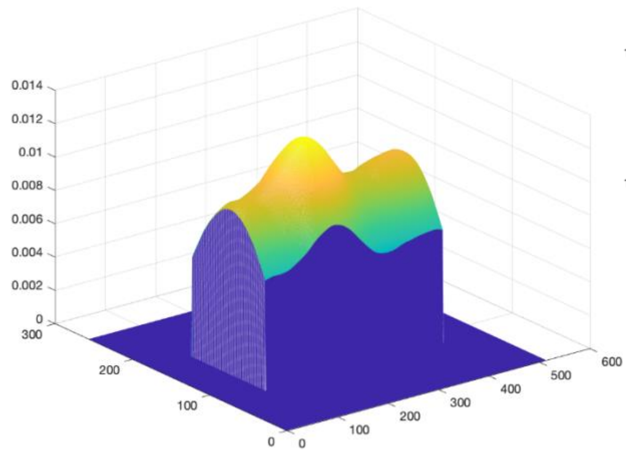
(b) 3D Plot

Figure 16: Image after applying gabor filter with F=0.6038, $\theta = -50.5°$ and $\sigma = 8$

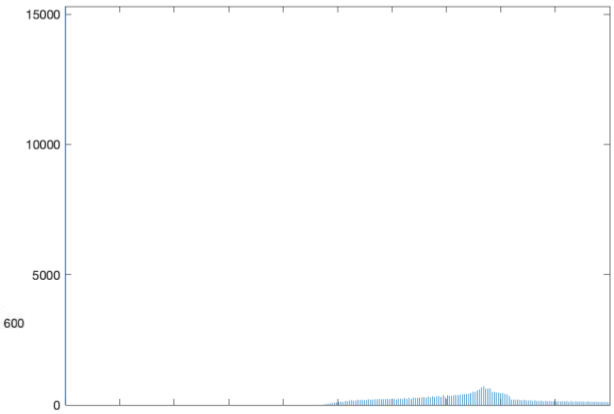
As seen from Fig 16 (b), it is pretty evident that a smooth graph is not obtained. There is lot of spikes. Thus, we further smoothen it with another gaussian having $\sigma = 40$. The results are shown in Fig. 17.



(a) Image after applying smoothing Filter with $\sigma = 40$



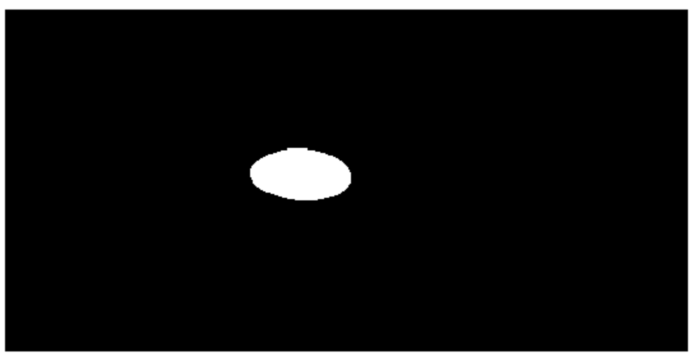
(b) 3D Plot



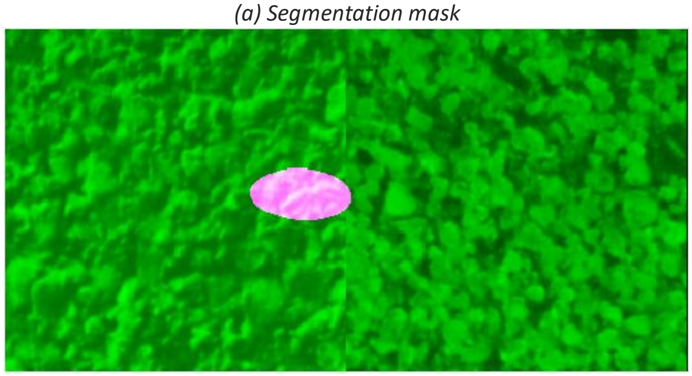
(c) Histogram after scaling the values from [0,1]

Figure 17: Image and its #D representation after applying smoothing filter with $\sigma = 40$

The smoothened image (shown in Fig. 17(a)) obtained was segmented using its histogram (Fig. 17(c))**. The threshold value was found out to be 0.92754**. The segmented image was further superimposed with the original image. The results are shown in Fig 18



(a) Segmentation mask



(b) Segmented Image Fused with Original Image.

Figure 18: Superimposed original and segmented images

The bounding box for the segmented portion was found to have x and y limits as (185:260,105:143). The bounding box is shown on the original image in Fig 19.

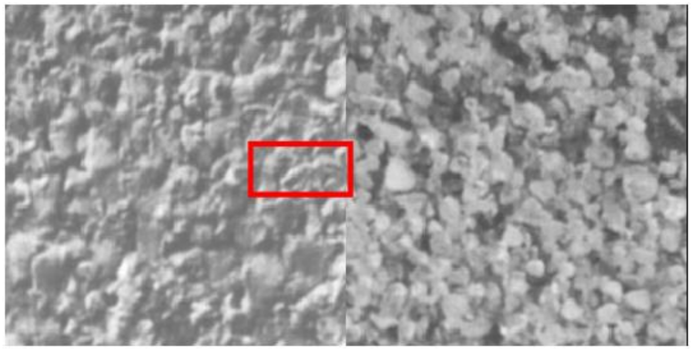


Figure 19: Segmented portion bounding box

Comments:

The results show that Gabor filter works reasonably fine with respect to segmenting textures. It has worked best for problem 1 and problem 2. However, the results for problem 3 and problem 4 were not up to the mark. In-fact, the results for problem 4 were the worst. It shows that the more distinct the textures, the better the segmentation result. And **the more repetitive patterns in the textures, we are likely to get nice and well separable distinct image after applying Gabor filter.** This is evident from the texture1.gif with I and \ patterns which resulted in a nicely separable image after smoothening the Gabor filtered output, but this is not the case in the other textures.

From the 3D plots, it is pretty much evident that smoothening helps in reducing the error when we are trying to segment the threshold values. However, smoothening is blurring the images and introducing another error.

Only the pixels within the width of GEF and gaussian are modified and rest of the pixels were zeroed out.

D. Conclusion

This project shows the power of MATLAB. The complex numbers in GEF were very easy to compute in MATLAB. Also, Gabor filter detect the textures better when it is followed by a smoothening filter. However, it is very much dependent on the parameters we have selected. Thus, parameter tuning is very crucial in the success of Gabor filters.

From our findings from this project we also conclude that the **Gabor filter doesn’t work well on its own with naturally occurring textures which has little repeating patterns in it**.