# CSE 585/EE 555: Digital Image Processing II
# Computer Project #4
# Texture Segmentation

***Pavan Gurudath, Ming-Ju Li***

April 8 2018

---

## 1 Objectives

This project aims to implement segmentation of different textures in an image using Gabor filter in MATLAB.

## 2 Methods

In this project, we segment the different textures by applying Gabor filter and a smoothing filter (if necessary).

The following methodology details the algorithm to obtain the objectives of the project.

- Read the original image I(x,y).
- Compute the circularly symmetric Gaussian g(x,y) with the parametric values F, $\theta$ and sigma ($\sigma$).
  Compute g(x) and g(y) separately and then combine them together, g(x,y) = g(x)*g(y), where

$$g(x,y) = \frac{1}{2\pi\sigma^2} exp\left\{ -\frac{(x^2+y^2)}{2\sigma^2} \right\} \tag{1}$$

- Compute GEF h(x,y) using the formulae

$$h(x,y) = g(x,y) \cdot exp[j2\pi F(xcos\theta + ysin\theta)] \tag{2a}$$
$$= g(x,y) \cdot exp[j2\pi(Ux+Vy)] \tag{2b}$$

- Apply Gabor filter to the input image, I(x,y), following instructions on L18-4 and L18-6 as follows:
  - $i_1(x,y) = i(x,y) * h_1(x)$ convolution in x
    $i_1(x,y) = i(x,y) * g_1(x) \cdot exp\{j2\pi \mathrm{Fxcos}(\theta)\}$
  - $i_2(x,y) = i_1(x,y) * h_2(x,y)$ convolution in y
  - m(x,y) = $|i_2(x,y)|$

  where $i_1(x,y) = \sum_{x'=x-2\sigma}^{x+2\sigma} i(x-x',y)\hat{h}_1(x,y)$ and $i_2(x,y)$ is computed similarly.
  $m_(x,y)$ is the output image
- Compute smoothing filter g'(x,y) similar to equation 1 except with a different $\sigma$.
- Apply smoothing filter g'(x,y) to m(x,y) (if necessary) using the equation

$$m'(x,y) = m(x,y) * g'(x,y) \tag{3}$$

- Segment the different textures by choosing a threshold that produces a good segmentation of the different texture in m'(x,y).
- Superimpose the image segmentation result on the original image and display its result.

To run this project, run *main.m* and it'll call all the necessary functions to execute the program. The flowchart in Figure 1 shows the flow of the methodology. The images that are loaded are in the working directory namely texture1.gif, texture2.gif and d9d77.gif

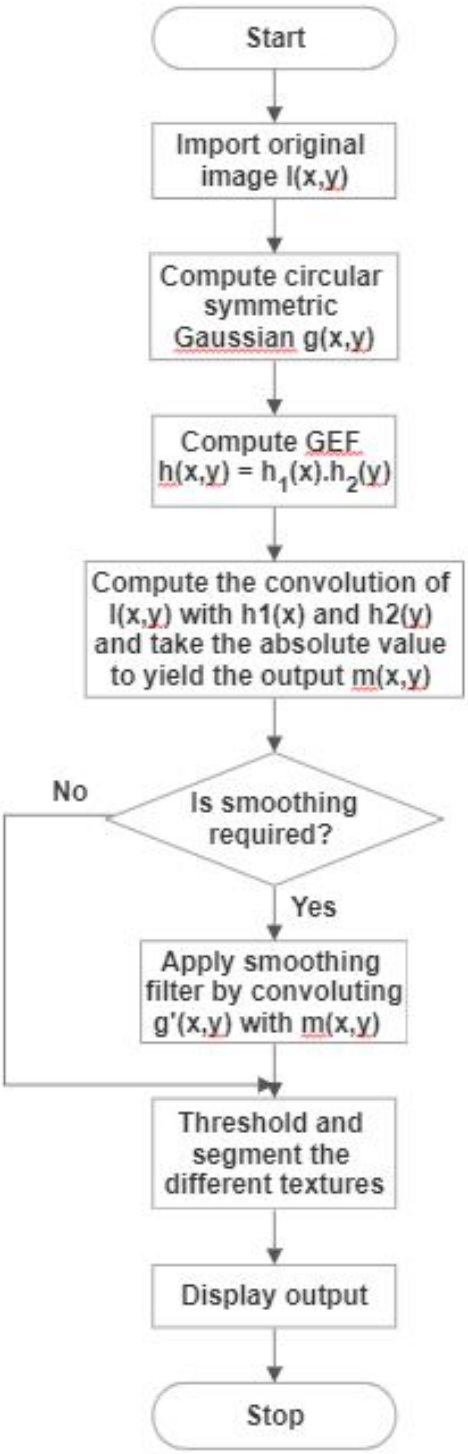


Figure 1: Flowchart

# 3 Results

This section shows the results of applying gabor filter and the smoothing filter to the input images texture2.gif, texture1.gif and d9d77.gif in Section 3.1, 3.2 and 3.3 respectively.

## 3.1 texture2.gif

The original image that is considered for this question is as shown in Figure 2. The gabor filter is applied by using the parametric values, $F = 0.059, \theta = 135°$ and $\sigma = 8$. Figures 3 and 4 shows the results obtained after applying the gabor filter to the image in Figure 2, where Figure 4 is the 3-D plot. The output shown in figure 3 has been rescaled for the sake of display. Once this is done, the smoothing filter is applied and its output is as shown in Figure 5 and its 3-D plot is as shown in Figure 6.

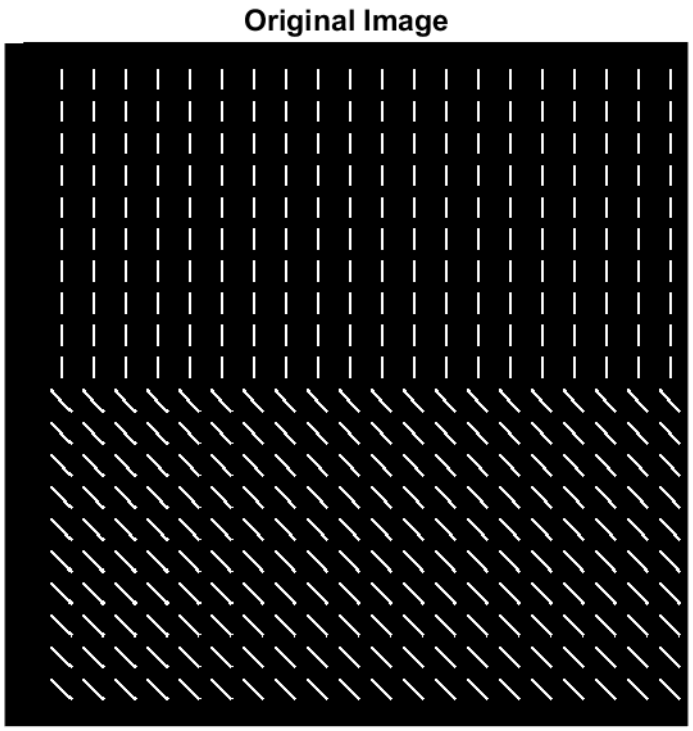


Figure 2: Input image of texture2.gif

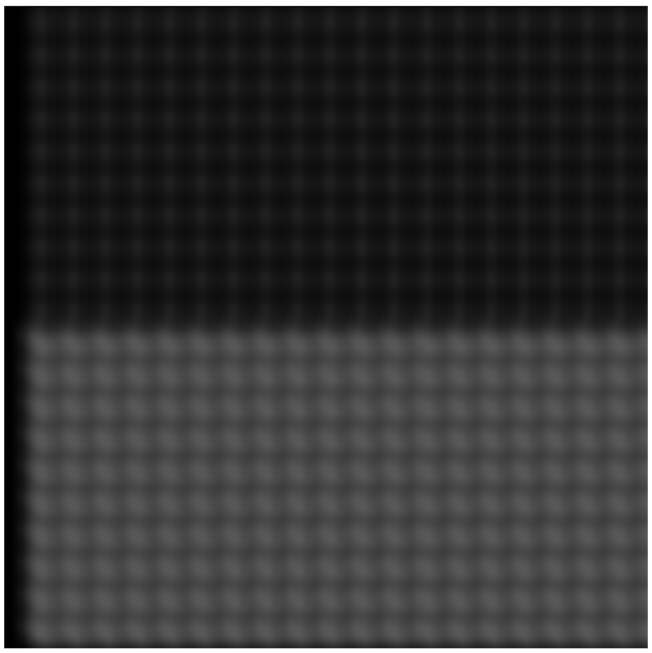

Figure 3: Gabor filtered image of texture2.gif, $\sigma = 8$

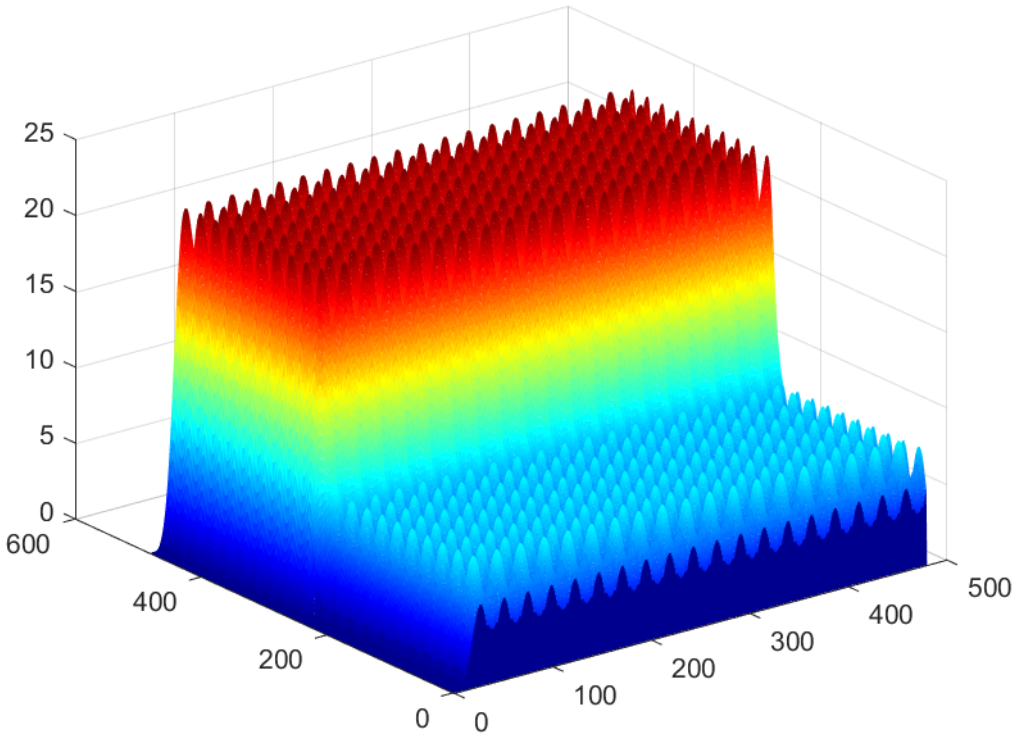


Figure 4: 3-D plot of gabor filtered image of texture2.gif, $\sigma = 8$

Figure 5: Smooth filtered image, $\sigma = 24$

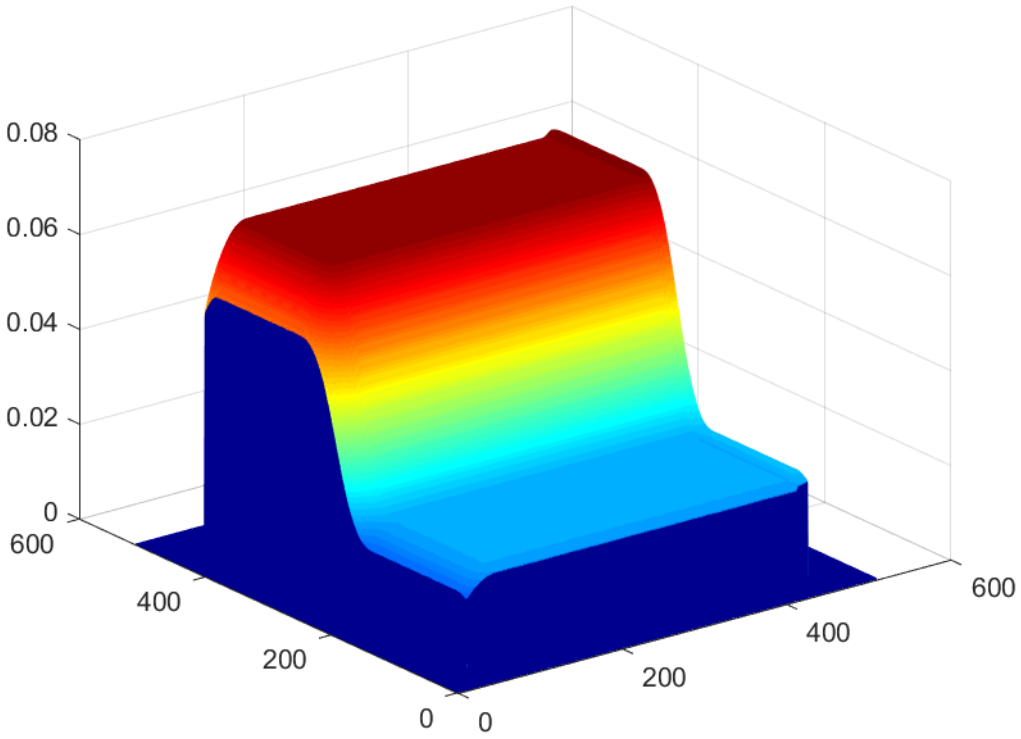


Figure 6: 3-D plot after smoothing filter $\sigma = 24$

Figure 7 shows the image after applying the threshold to the smoothened image. A threshold of 0.045 was used for this image. Figure 8 shows the segmentation of different textures.

Figure 7: Threshold image

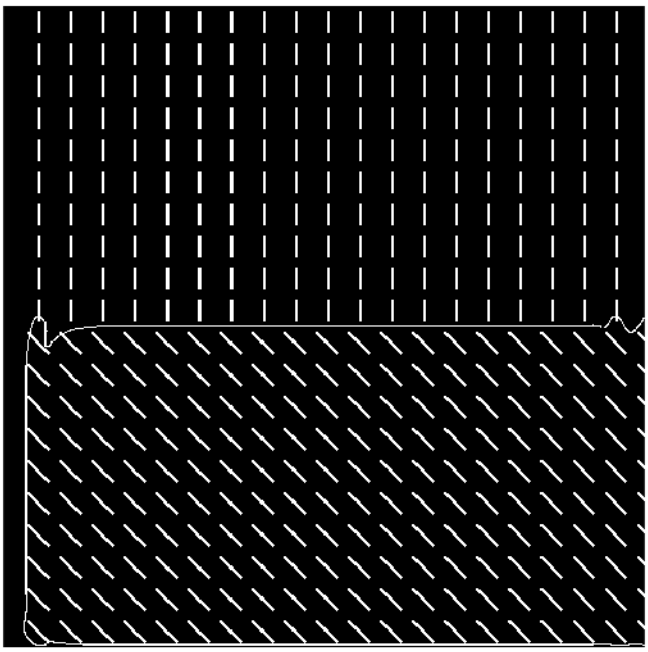

Figure 8: Segmented image

### 3.2 texture1.gif

The original image that is considered for this question is as shown in Figure 9. The gabor filter is applied by using the parametric values, $F = 0.042, \theta = 0^\circ$ and $\sigma = 24$. Figures 10 and 11 shows the results obtained after applying the gabor filter to the image in Figure 9, where Figure 11 is the 3-D plot. The output shown in figure 10 has been rescaled for the sake of display. Once this is done, the smoothing filter is applied and its output is as shown in Figure 12 and its 3-D plot is as shown in Figure 13.

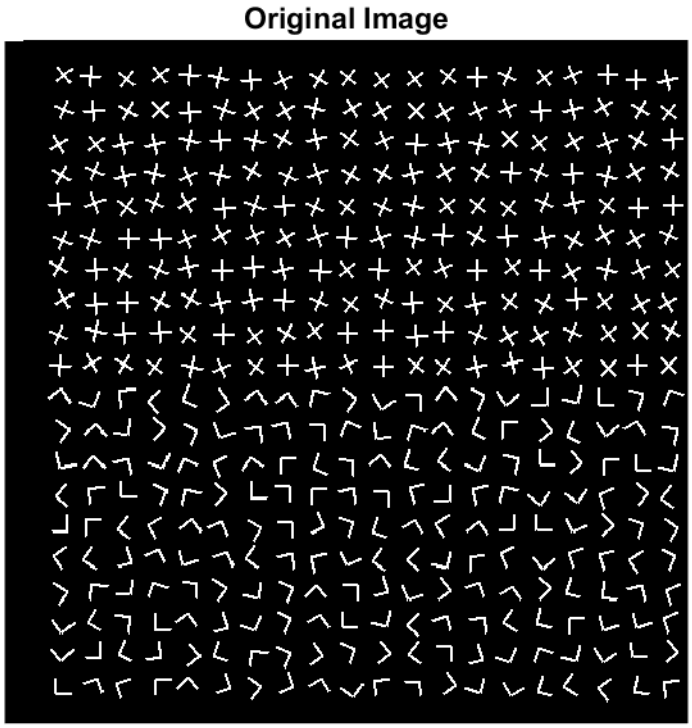


Figure 9: Input image of texture1.gif

Figure 10: Gabor filtered image of texture1.gif, $\sigma = 24$

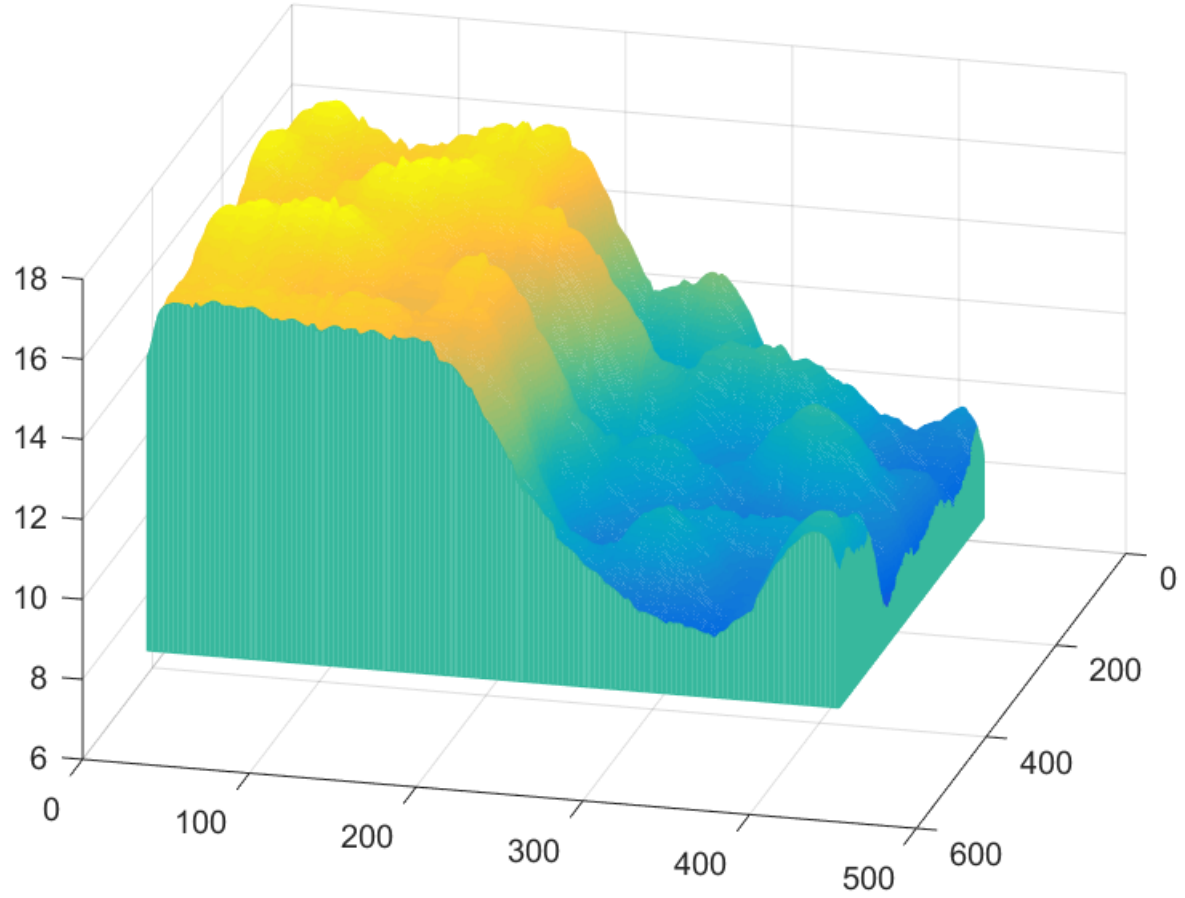


Figure 11: 3-D plot of gabor filtered image of texture1.gif, $\sigma = 8$

Figure 12: Smooth filtered image, $\sigma = 24$

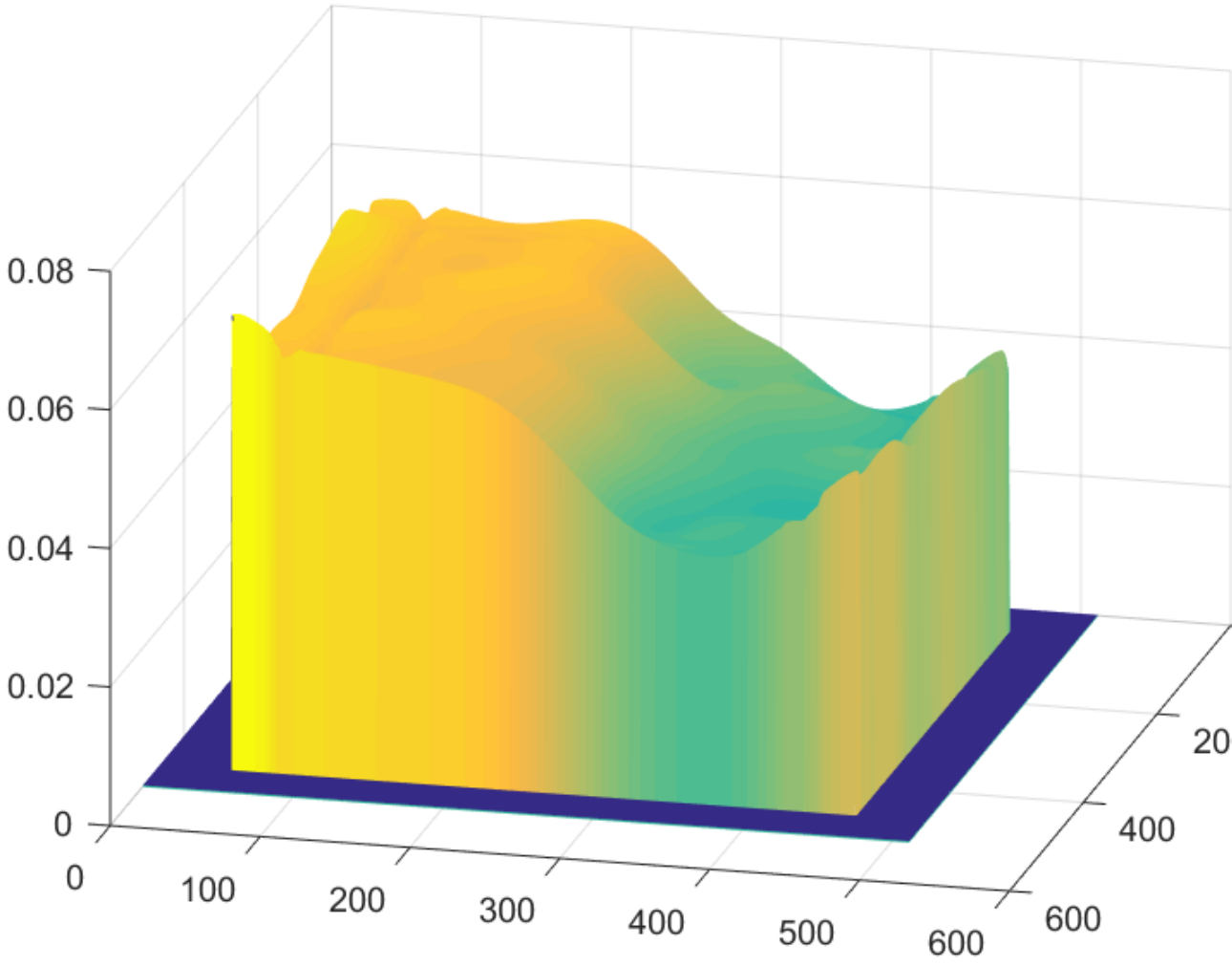


Figure 13: 3-D plot after smoothing filter $\sigma = 24$

Figure 14 shows the image after applying the threshold to the smoothened image. A threshold of 0.045 was used for this image. Figure 15 shows the segmentation of different textures.

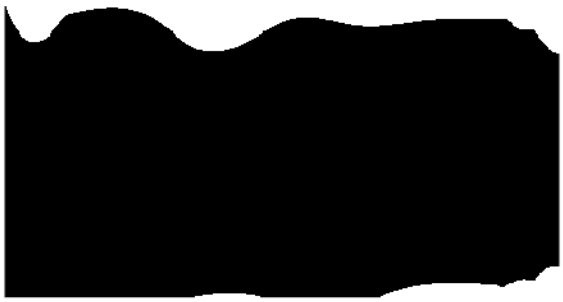

Figure 14: Threshold image

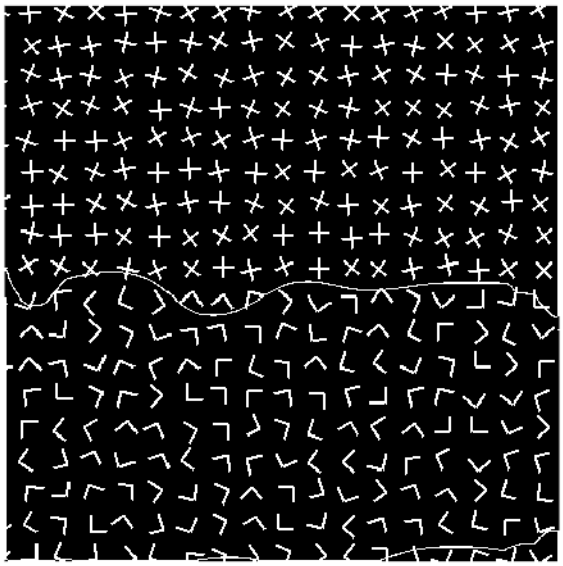

Figure 15: Segmented image

### 3.3 d9d77.gif

The original image that is considered for this question is as shown in Figure 16. The size of this image is 512 x 513 and the last column is removed to make it 512 x 512. This image is converted to black and white by using a threshold value of 150. All pixel values above 150 are converted to white and the remaining are converted to black. The resultant image is shown in Fig 17. The gabor filter is applied by using the parametric values, $F = 0.062, \theta = 63^{\circ}$ and $\sigma = 38$. Figures 18 and 19 shows the results obtained after applying the gabor filter to the image in Figure 17, where Figure 19 is the 3-D plot. The output shown in figure 18 has been rescaled for the sake of display.

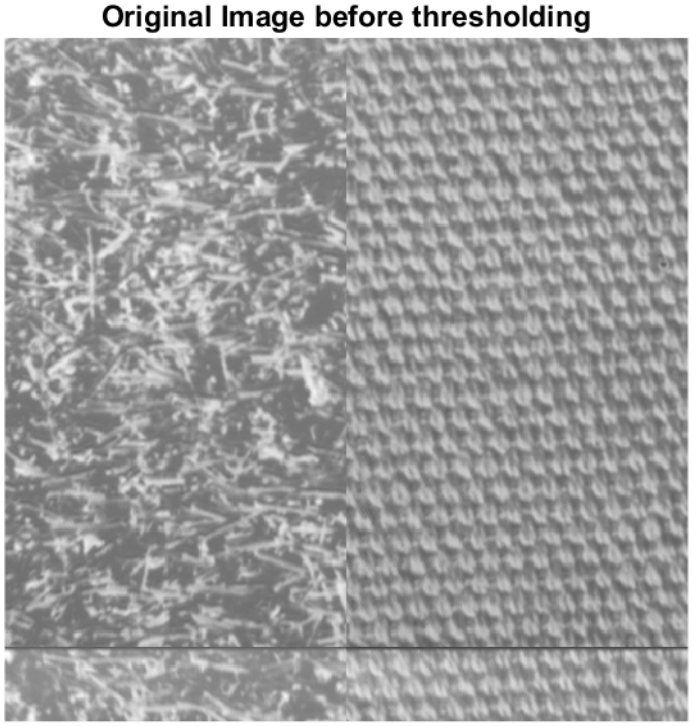


Figure 16: Input image of d9d77.gif

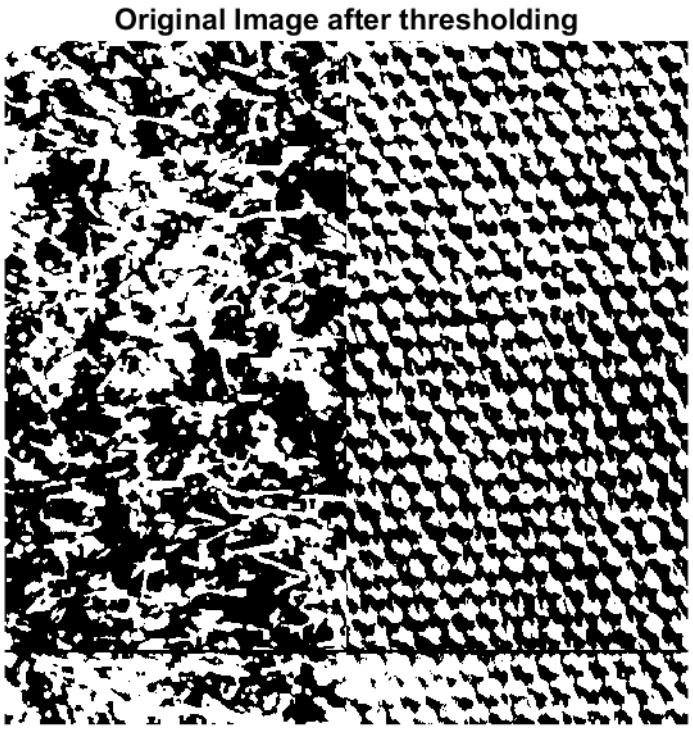


Figure 17: Binary Input image of d9d77.gif

Figure 18: Gabor filtered image of d9d77.gif, $\sigma = 38$

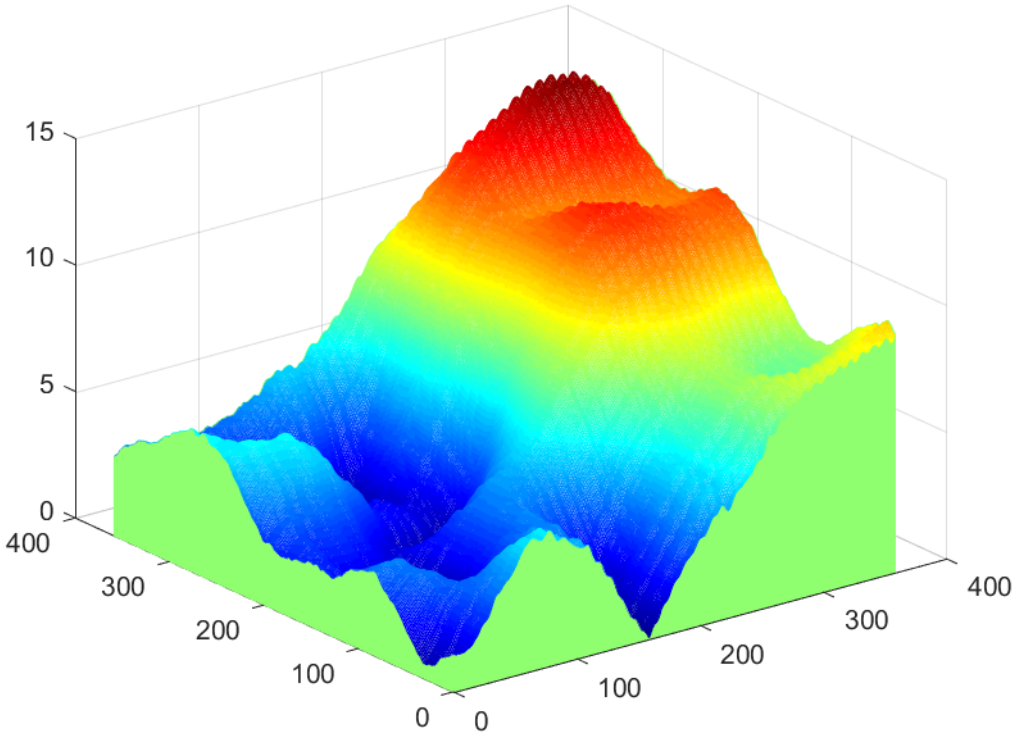


Figure 19: 3-D plot of gabor filtered image of d9d77.gif, $\sigma = 38$

Figure 20 shows the image after applying the threshold to the gabor filtered image. A threshold of 0.021 was used for this image after several tries. Figure 21 shows the segmentation of the different textures.

Figure 20: Threshold image

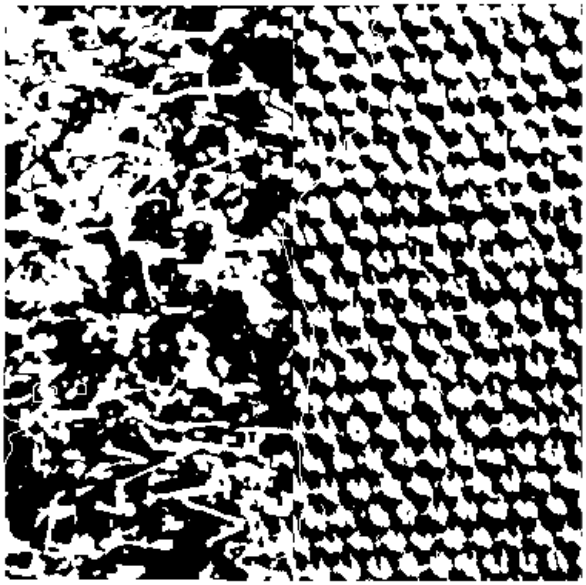

Figure 21: Segmented image

# 4 Conclusion

The following conclusions can be drawn after analyzing the results:

- Gabor filter works reasonably well with respect to segmenting textures. While there aren't any issues with texture1.gif and texture2.gif, however, finding the right values for d9d77.gif was a little bit of an issue. It could be concluded that the more distinct and organized the texels are, the better the segmentation result.
- From the 3D plots, the effect of smoothing filter can be seen and it helps in reducing error when we use a threshold to segment 2 textures.
- With every filter applied, the image gets smaller in size. This is because the images were not zero-padded. However, this will have to be taken into effect for small sized images.
- The choosing of the parameter values is very important to achieve desirable results.