Answers to questions in
Lab 2: Edge detection & Hough transform

Name:Rithika Harish Kumar   Program: TCSCM

Instructions: Complete the lab according to the instructions in the notes and respond to the questions stated below. Keep the answers short and focus on what is essential. Illustrate with figures only when explicitly requested.

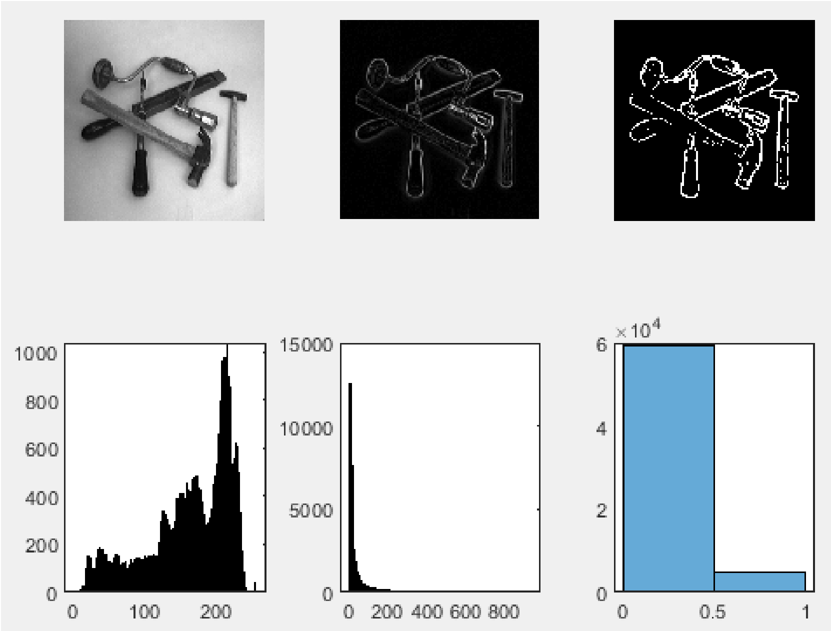
Good luck!

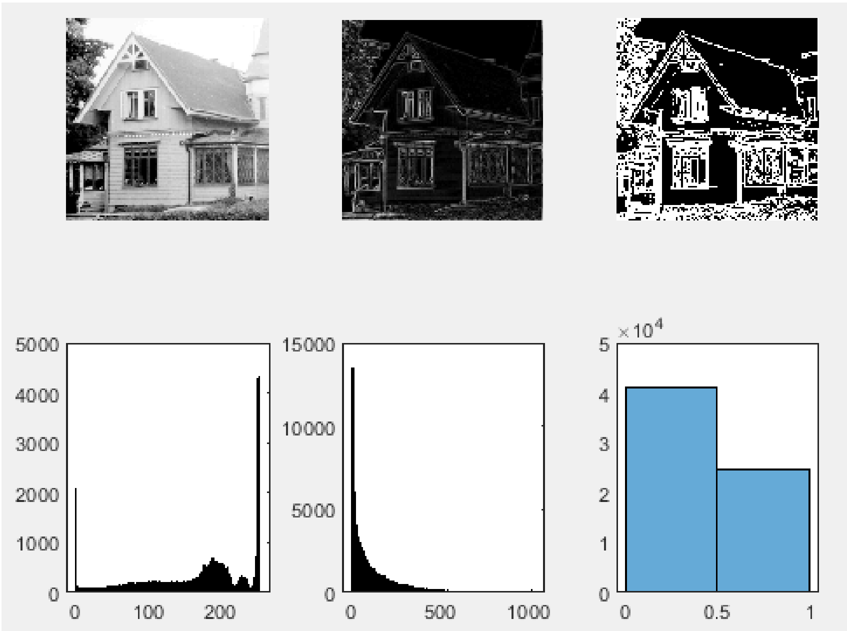
Question 1: What do you expect the results to look like and why? Compare the size of *dxtools* with the size of *tools*. Why are these sizes different?

Answers:
Results should be edge map images containing information about the gradient magnitude(intensity) of the original image. dxtools gives stronger response for vertical edges and dytools gives stronger response for horizonal edges. The zero response gradient areas have constant intensity.
The dxtools seem smaller. This is because we perform convolution on the image with a filter mask of 3x3 and set shape to valid. So, edges are trimmed by 2 to have valid pixels around centre of the kernel.

Question 2: Is it easy to find a threshold that results in thin edges? Explain why or why not!

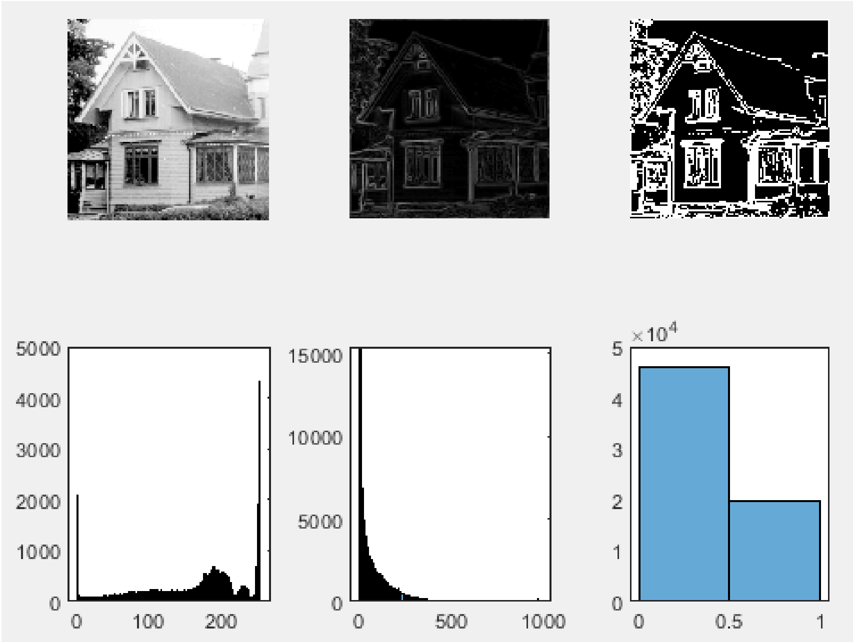
Answers:
When handling edges exported from high scales, the thresholding produces thick edges which could result in vanishing of thin edges.
When dealing with noisy images, connectivity between edges is lost.
Hence it is not very easy to find a threshold which results in thin edges because it depends on various factors.





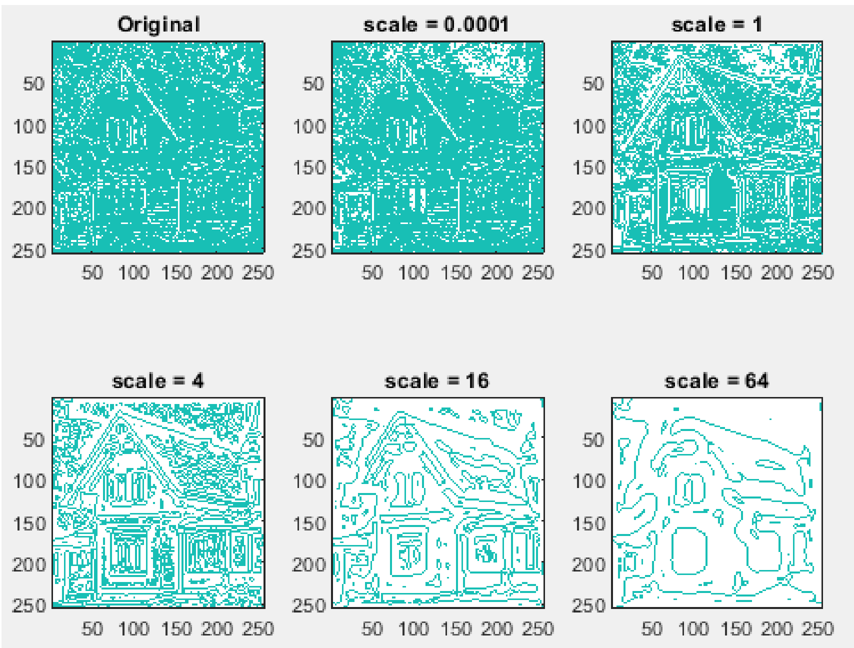
Question 3: Does smoothing the image help to find edges?

Answers:
Smoothing helps to some degree because it subtracts high frequency noise and the fine details in the image and keeps only the principle edges. But the smoothing should be done in such a way that the edge distortion is minimum. Smoothing with thresholding gives better results.



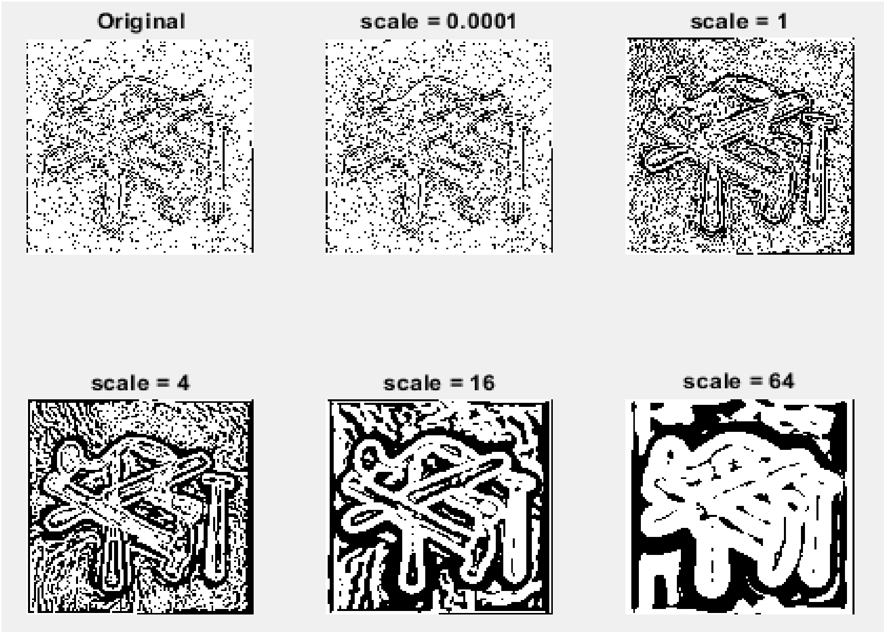
Question 4: What can you observe? Provide explanation based on the generated images.

Answers:
Shows where the second order derivative of the images is zero.
Different scales give different results. If scale is small, the noise is not suppressed enough and fine details are there. This leads to lot of false positives.
The higher the scale, the higher the variance of the gaussian filter used, the higher blurring in image, the higher suppression of noise, the lower the accuracy at approximating true position of the edges.



Question 5: Assemble the results of the experiment above into an illustrative collage with the *subplot* command. Which are your observations and conclusions?

Answers:
Shows the points where third order derivative of images is negative.(white area)
A scale of 3-5 should give proper results. Also thresholding to an extent may collect more principle edges.

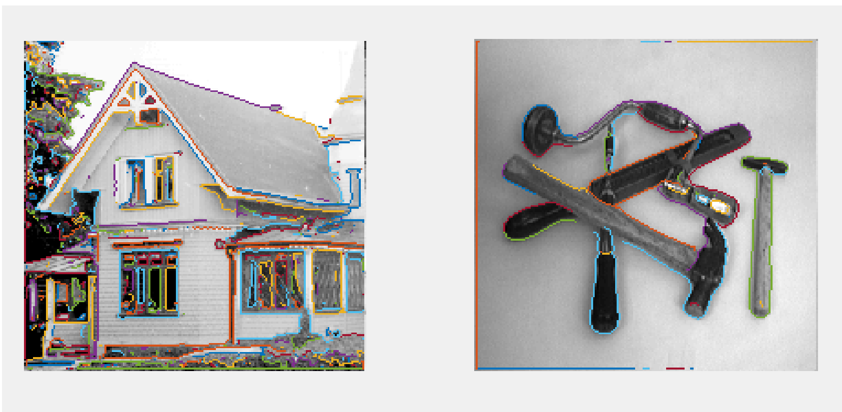


Question 6: How can you use the response from *Lvv* to detect edges, and how can you improve the result by using *Lvvv*?

Answers:
The first order derivative gives information about the changes in slope. In the second order derivative the curvety of the function will change at zero crossings. Thus, it gives the edges of the image. But to see that it is a local maxima, we have to give information of the derivative of that.
"The gradient magnitude reaches a local maximum where the second order derivative is zero and third order derivative is less than zero."

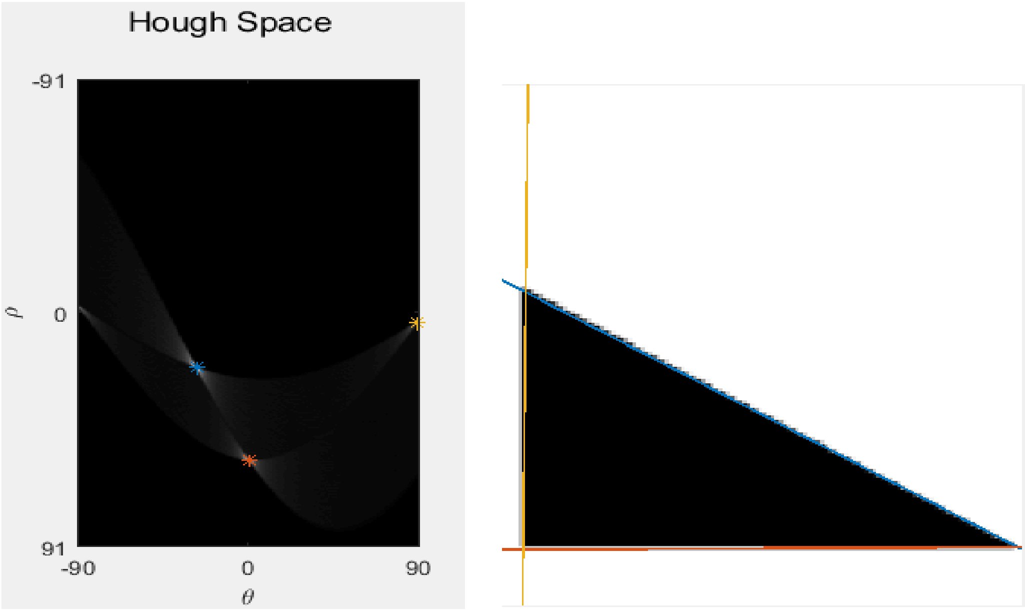
Question 7: Present your best results obtained with *extractedge* for *house* and *tools.*

Answers:

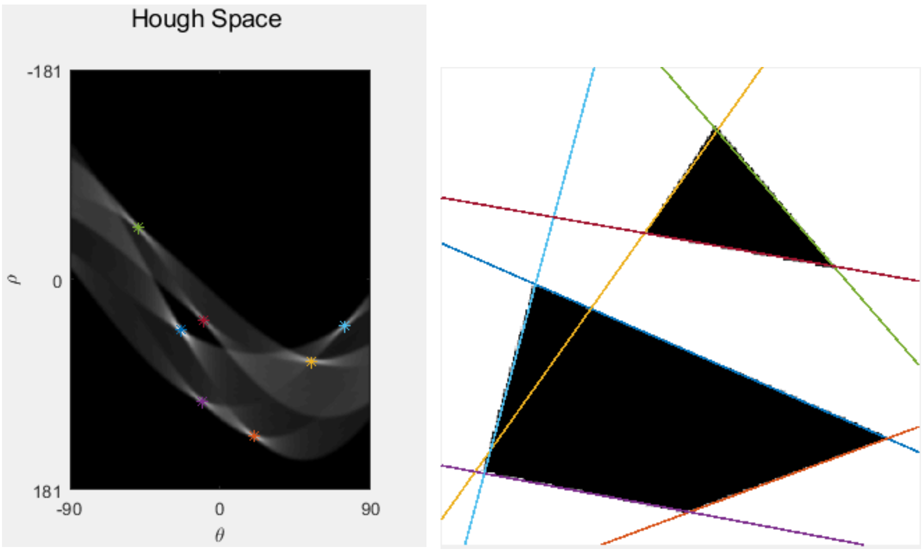


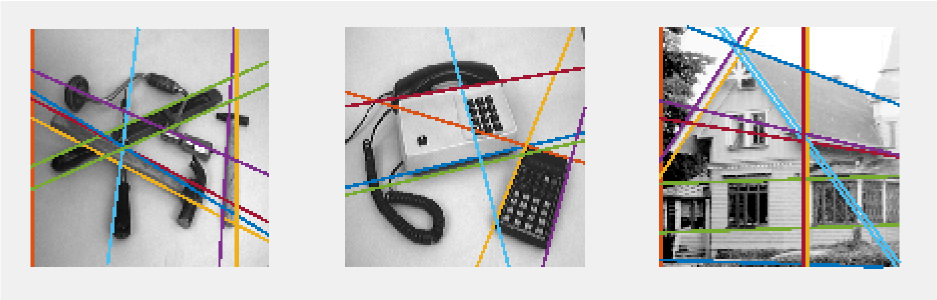
Question 8: Identify the correspondences between the strongest peaks in the accumulator and line segments in the output image. Doing so convince yourself that the implementation is correct. Summarize the results of in one or more figures.

Answers:
The parameters used for simple triangle: scale = 0.5, threshold = 40, nrho= 500, ntheta = 300, nlines = 3, smoothing = 0.1 and accumulator inc function is f(x).



The parameters used for the triangle and polygon: scale = 0.5, threshold = 60, nrho = 700, ntheta = 500, nlines = 7, smoothing = 0.7 and accumulator inc function is f(x^2).





Question 9: How do the results and computational time depend on the number of cells in the accumulator?

Answers:
The bigger the number of rhos and thetas the computational time increases. If the accumulator matrix has high resolution, the computational time is also high. Double looping increases proportionally to increment in number of cells. But the accuracy of line approximation increases if the resolution is high mainly in complex structures. But if the resolution is high it could also give several responses for the same line because many neighbour local maximas are present.

Increasing cells – more lines near sharp edges, local maxima near edge line, more computational time.

Decreasing cells – less computation time, edges not detected accurately.

Question 10: How do you propose to do this? Try out a function that you would suggest and see if it improves the results. Does it?

Answers:
When accumulator increment function is set to 1, it means all edge lines have equal strength. But different edge lines are stronger than others. So, we should let the strongest edges perform accumulator increment based on magnitude strength because gradient magnitude has this information. But in some cases there are small responses alongside strong lines.

Big slopes gives sudden changes in accumulator. The edges with medium responses vanishes, and it saturates around a strong pick. Hence, logarithmic function(increases the possibility of edges with lower values of gradient magnitude) would provide better results.
