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

Name: __Mert Alp Taytak____________________________ Program: __EECS_

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:

I expected to see a mostly single colored image with lines where edges and strong shadows lie orthogonal to direction of the derivative. Depending on the direction of transition from dark to bright we get a different black or white edge.

Size is different because we are convolving the image matrix by a kernel. Since it does not make sense to pad the image with 0s to calculate the derivative, we lose the outermost edges.

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

Answers:

It is easy to find a decent threshold that more or less shows thin, distinct edges. While increasing the threshold helps thin edges, it also causes disconnects where the contrast is not as high. Therefore optimizing the threshold is somewhat hard.

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

Answers:

Small amounts of smoothing to remove noise may be helpful but high amounts of smoothing reduces effectiveness overall. That is because we need contrast and sharp edges in order to select edges via thresholds. Smoothing makes it hard to find a proper threshold.

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

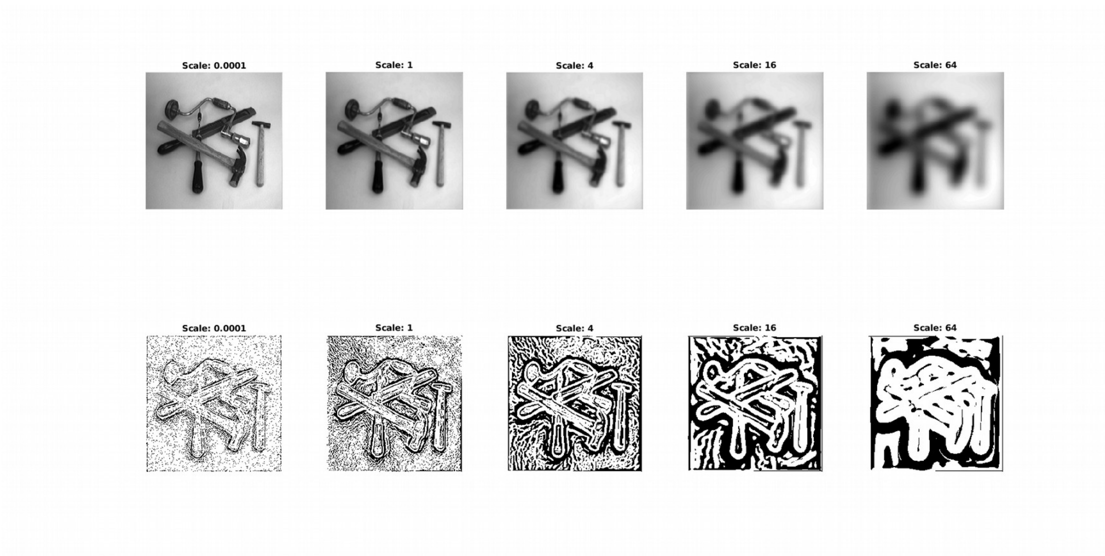
Answers:

I observe that smaller the scale higher the detail in the resulting images. With greater scales only the longest lines are present in the result and only after being distorted. On the other hand, very small scales allow for great detail to the points individual leaves or planks making up the walls of the house can be seen.

The method used in these contours make use of Gaussian filtering, hence higher the variance lower the detail.

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

Answers:



Sign conditions results in a binary, black and white, image. Per our construction, image is white where $L_{vvv} < 0$ and edges of black regions should have $L_{vv} = 0$. By our edge definition, edges of black regions are edges perceived in the base image. When we apply a Gaussian filter with higher variance we smooth the image, leading to fewer edges. Hence, there are less changes in resulting images. Leading to larger, connected areas of single color.

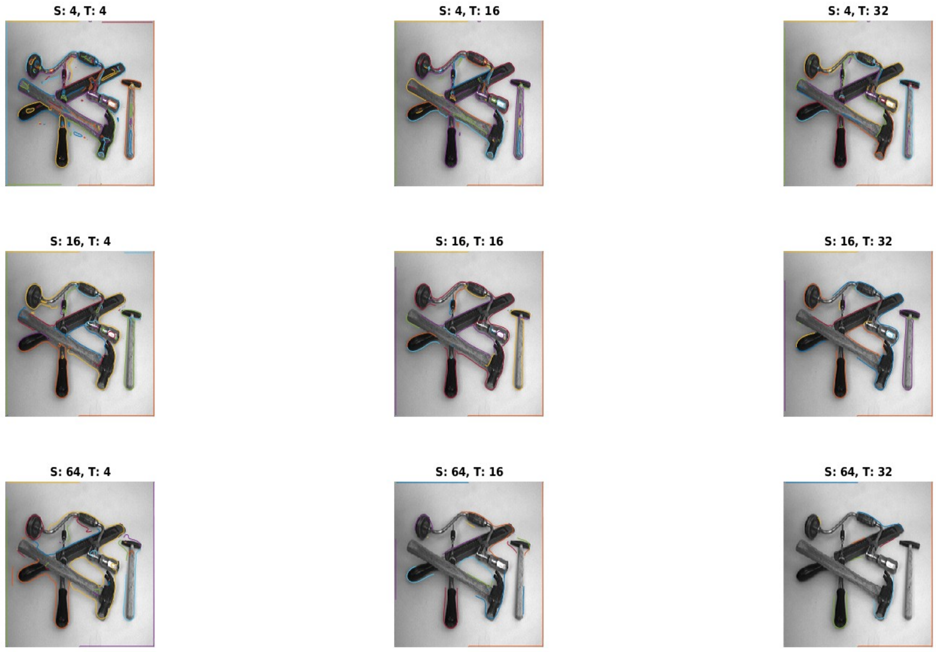
Question 6: How can you use the response from *Lvv* to detect edges, and how can you improve the result by using *Lvvv*?

Answers:

Per our definition of edges, points of interests are those with $L_{vv} = 0$. We extract those points from the image and then further improve upon them by selecting those who meet the criteria $L_{vvv} < 0$. Since computation is not always that exact, we can apply a small delta to select points where $|L_{vv}| <$ delta.

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

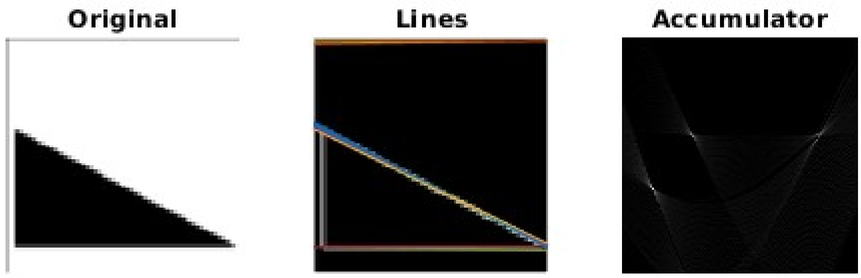
Answers:

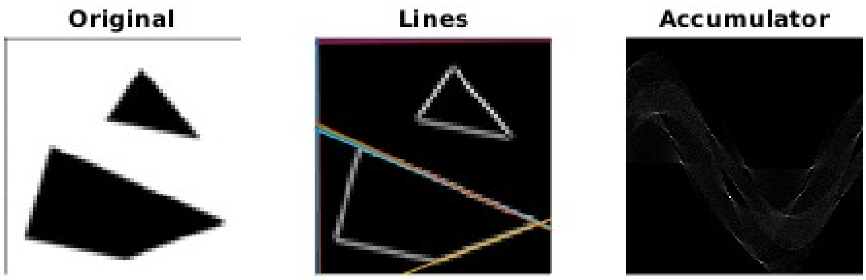


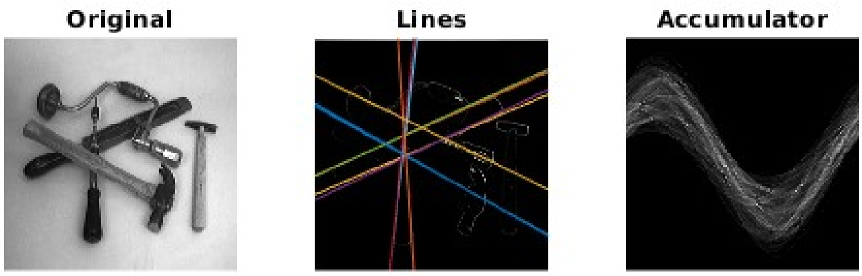
Question 8: Identify the correspondences between the strongest peaks in the accumulator and line segments in the output image. Doing so convince yourself that the implementation is correct. Summarize the results of in one or more figures.

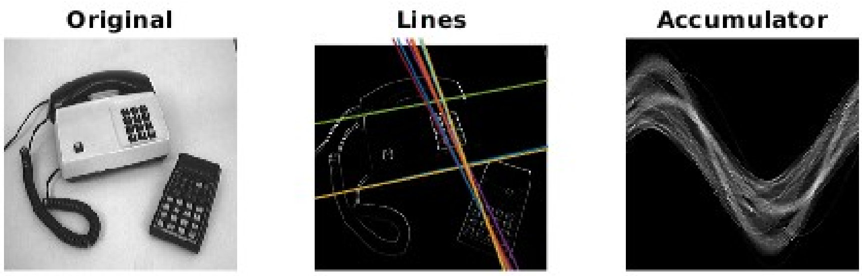
Answers:

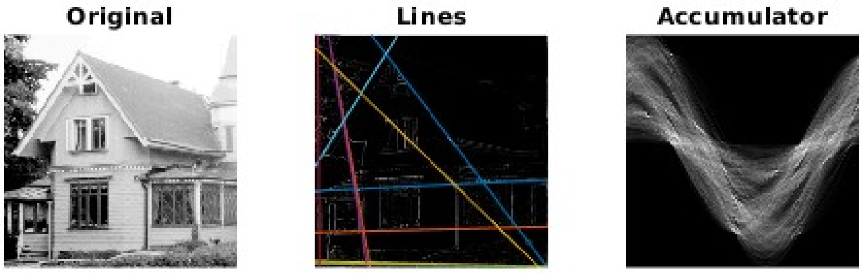
Following figures are the result of application of *houghedgeline*:











Bright points in the accumulator show points of intersection, which makes up the lines we see in the images in the middle. Vector from the center to an intersection in the accumulator image shows the corresponding distance from the center and orthogonal direction of the line in the middle image.

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

Answers:

I used *timeit* function to measure performance, number of cells will be investigated in both directions and results can be found in the following table:

nrho vs. ntheta	ntheta = 100	ntheta = 200	ntheta = 400
nrho = 100	0.192 sec	0.296 sec	0.492 sec
nrho = 200	0.199 sec	0.318 sec	0.562 sec
nrho = 400	0.239 sec	0.398 sec	0.711 sec

Obviously, computation time increases with increased resolution. Also, ntheta is more influential than nrho.

For the results, increased number of cells allows better results. Because increased resolution allows finding spatially small maxima.

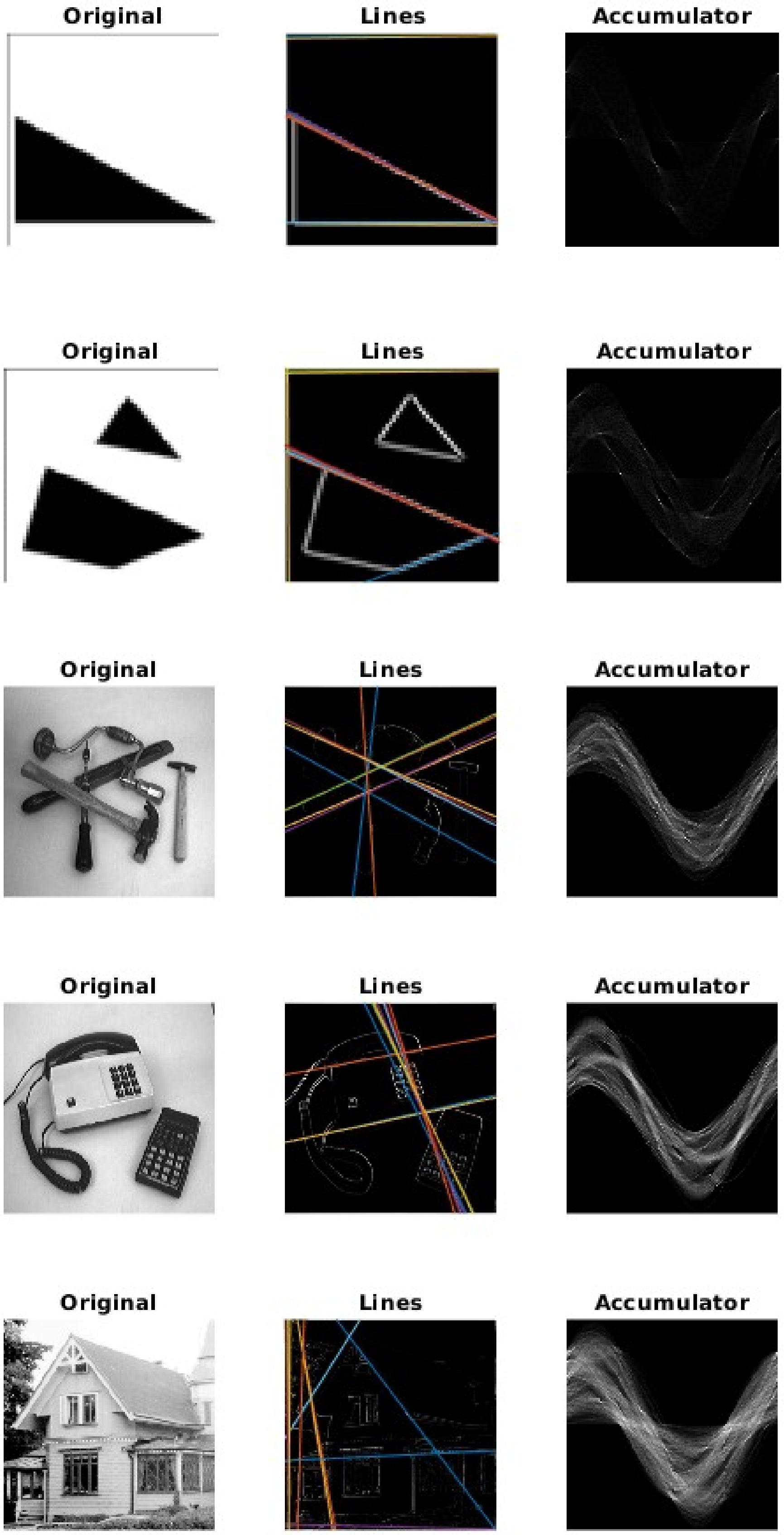
Question 10: How do you propose to do this? Try out a function that you would suggest and see if it improves the results. Does it?

Answers:

As a monotonically increasing function that depends on the gradient magnitude I used the following:

$$acc(x, y) = acc(x, y) + \log(1 + |magnitude|)$$

Then I repeated the experiment from question 8, whose results are in the following:


Original
Lines
Accumulator
Original
Lines
Accumulator
Original
Lines
Accumulator
Original
Lines
Accumulator
Original
Lines
Accumulator

With the exception of the last image, there is not a particularly noticeable difference. However, I would consider the changes to the last image an improvement. It is possible that by optimizing the parameters the updated increment can yield better results.