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

Name: Karen Jane Dsouza Program: MSc in Systems, Control and Robotics

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:
dxtools allows us to see the change in intensities across the x-wise pixels in the image. *dytools* allows us to see the change in intensities across the y-wise pixels in the image. When applying the Sobel operator, the results show the change in intensity in the image which corresponds to edges. The size of *dxtools* is different from the size of tools because the convolve2d function was called with the 'valid' argument. This argument causes the output to be smaller by the size of the kernel - 1 in each dimension.

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

Answers:
Strong edges in the image correspond to high gradient magnitudes while weak edges correspond to low gradient magnitudes. A high threshold may not include real edges that are affected by noise, while a low threshold may include too many noise-induced edges. If the threshold is set to be too low, the sharper edges get cut off and the edges become very thick. If the threshold is set to be too high, some weak edges might become more segmented.

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

Answers:
Smoothing removes the noises from the image. Noise can distort the edges which are at higher gradient magnitudes. Smoothing also makes the transition from low to high intensity more gradual making edge detection easier.

Question 4: What can you observe? Provide explanation based on the generated images.
Answers:
The scale determines the blurring of the image using the Gaussian filter. The larger the scale, the more the image is blurred. For lower scales less than 4, more noisy edges are visible, including the contours on the roof and the wooden panels making the house. At scale of 16, only the stronger edges are visible which include the general outline of the house. At a scale of 64, only the distorted edges of the house can be seen. Most of the edges within the image are reduced in this scale.

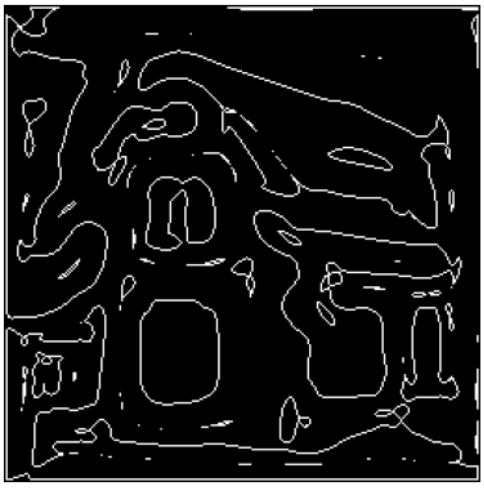
Scaled at 0.0001

Scaled at 1.0

Scaled at 4.0

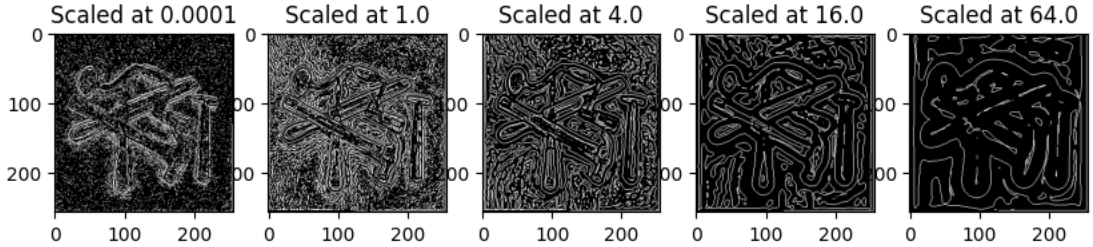
Scaled at 16.0

Scaled at 64.0



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

Answers:



$\widetilde{L_{vvv}}$ allows only the points corresponding to the local maxima to get passed. For lower scales, more details are visible but these diminish as the scale increases.

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

Answers:

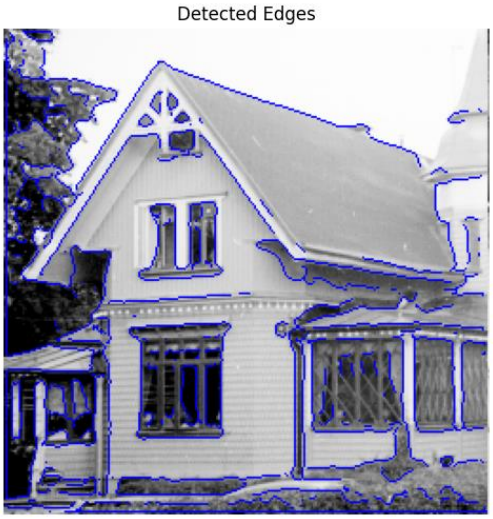
$\widetilde{L_{vv}}$ give us all the zero-crossings which coincide with the turning points (maxima and minima) of the image. $\widetilde{L_{vvv}}$ allow us to only consider the values of the maxima for the edge detection increasing the accuracy of the detector.

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

Answers:

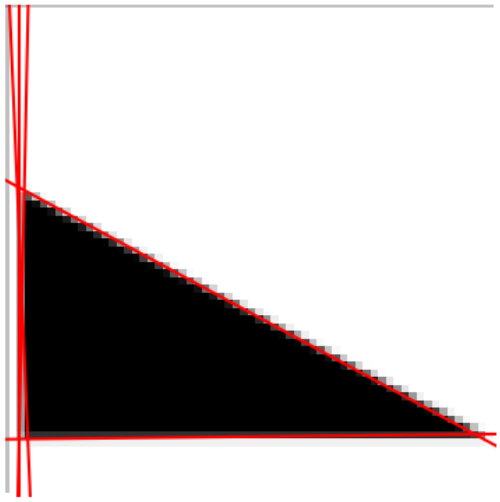
For the tools image, the best result was obtained for scale = 4.0 and threshold = 6.5.

For the house image, the best result was obtained for scale = 4.0 and threshold = 6.5.



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

Answers:
The strongest peaks are the points with the most overlaps. These overlaps correspond to the corners of the edges which are then used to interpolate and reproduce the appropriate edges.



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

Answers:
As the number of cells increases, the computational time increases, but the resolution also increases. If the number of nrho and ntheta are very low, multiple distinct lines would be detected as a single line and some values may also result in multiple edges for the same line.

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:
This accumulator allows the edge points with a higher gradient magnitude to be emphasized more. To implement this, the houghline method should have an extra argument called increment which allows the increment to be h($|\nabla L|$) or 1 to be passed for default if no argument is provided. Although theoretically the algorithm should give better results, it might not necessarily be true in all cases. This algorithm results in more responses from the edges.
