Answers to questions in

Lab 2: Edge detection & Hough transform

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_PIETRO ALOVISI\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Program: \_\_\_\_\_\_\_\_\_CDATE\_\_\_\_\_\_\_\_\_\_

**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 was expecting that the result in *dxtools* would enhance the change in intensity only on the horizontal axis, while *dytools* would do the same but in the vertical direction.

The different sizes are due to the fact that when we convolve we loose the the pixels on the edge because the convolution centered in those pixel would require to know the pixels outside the image.

In my case, with a filter deltax=[-1/2,0,1/2], I loose a the full leftmost and rightmost columns, leading to a new image size of 256X254.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Answers:

It is not easy using this method, because there is a trade off between the resulting thickness of the edge and the amount of edges I can detect. This is due to the fact that if I use a high threshold I get only the sharpest changes in the intensity in the image, which only take 1 or 2 pixels in the image, obtaining thin edges. But, since most of the real edges are blurred and their intensity grows as a ramp, choosing a high threshold leads to few edges detected, but if I choose a lower threshold I would get a thicker edge, because I will not filter the blurred part.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Answers:

Short answer: yes. It can help to reduce noise, and so to avoid some “fake” edges, but the most important effect is that it helps in selecting the “*scale”* of the edges.

For example, blurring out fine details of the house in godthem256 leads to not recognize no more the wooden tiles of the house and the leaves in the tree and bushes.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Answers:

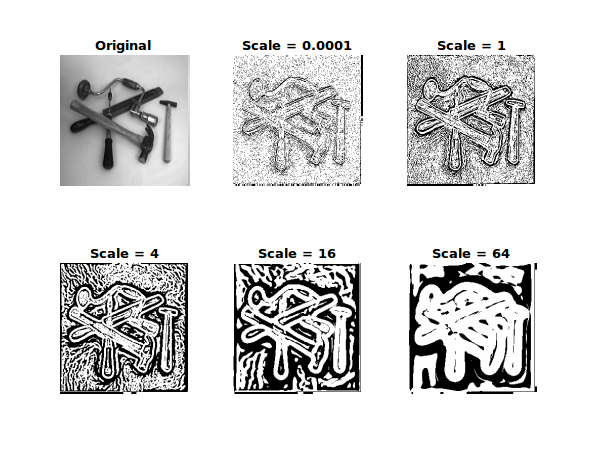
I can observe that if I choose a lower scale value I get almost all the edges in the image, even the small one like leaves and windows details. While for a higher value of the scale I get the coarser details, like the house and tree silouette.

As I said before, the scale is very important because it lets us choose the size of the object we want to recognize. Basically the blur is a filter in the domain of the scale, it blurs out the smaller edges and leaves the image only with the coarser ones. So its value must be carefully selected.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Answers:



We observe that if we choose a greater scale we get wider white areas around the 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:

I can detect edges by finding values where *Lvv* is 0, or better, interpolate those points from adjecent pixel having opposite sign values.

We can improve the result by looking at both *Lvv* and  *Lvvv* at the same time. As highlighted in the theory part, we want *Lvv* to be 0 to have a single pixel in the border, while we choose *Lvvv* to be negative to make sure that the point is a maximum point in the magnitude of the magnitude of the first derivative (in the direction of the gradient).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**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 accu-mulator 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:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Answers:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**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:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_