Answers to questions in

Lab 2: Edge detection & Hough transform

Name: Baptiste Engel & Marie-Ange Stefanos Program: Machine Learning

**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:

The expected results are edges detection along one axis according to the mask used.

Using deltax (resp. deltay), edge detection along the x -horizontal- (resp. y -vertical-) axis gives vertical (resp. horizontal) lines.

*dxtools* and *dytools* have size (254, 254) whereas *tools* has size (256, 256) due to side effects.

The mode ‘valid’ only compute the convolution without padding. Thus, it is impossible to convolve the border pixels with a matrix of size (3, 3) since there is a lack of neighbor pixels for the ones that constitute the frame of the image.

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

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

Answers:

The gradient magnitude of images that we study have a dark background with shapes delimited by lighter pixels. Thus, looking at the histogram of the magnitude of the gradient before choosing a threshold can be very useful. Indeed, a suitable threshold will separate the background (darkest pixels) from shapes edges (lighter pixels).

No, it is not easy to find a threshold that results in thin edges because either (threshold too high) some edges are detected and are thin, but all edges are not detected or (threshold too low) all edges are correctly detected but edges are not thin.

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

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

Answers:

Yes, smoothing the image helps to find edges because smoothing the image, that is applying a gaussian filter, remove potential gaussian noise. This enables to avoid detecting artefacts of the images and noisy pixels as edges. The lower the variance, the thinner the edges.

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

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

Answers:

Une image contenant texte, équipement électronique

Description générée automatiquement

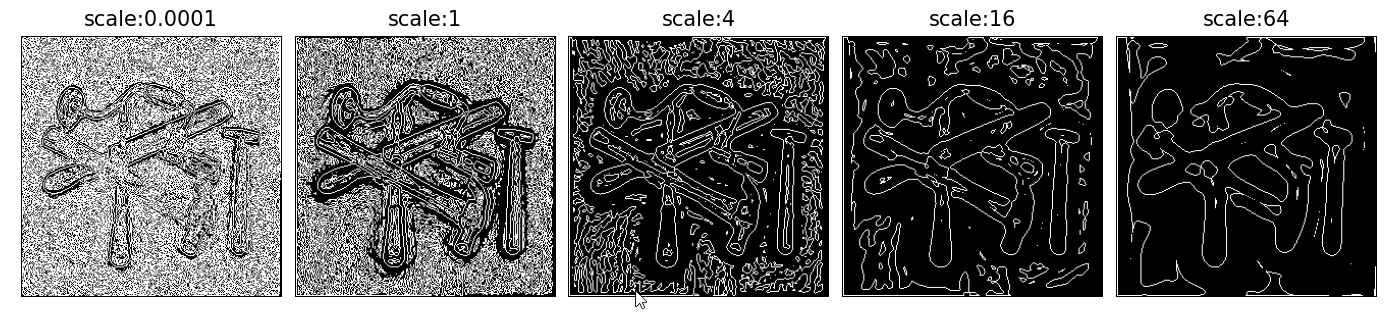


Fig 1: Magnitude of the results using 2nd criteria (Lvv = 0) (scales are the same on both lines)

1st line: house image

2nd line: tools image

The scale is the variance of the gaussian filter.

In this question we study the first criterion of edge detection, which is Lvv = 0. The lower the scale, the more edges are detected (including texture, details and noise if the scale is too low). The higher the scale, the less edges are detected since the image have been blurred, thus removing all details and gaussian noise from the image before computing its second derivative.

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

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

Answers:

Une image contenant texte, tapis

Description générée automatiquement

Une image contenant texte

Description générée automatiquement

Fig 2: Magnitude of the results using 2nd criteria (Lvvv < 0) (scales are the same on both lines)

1st line: house image

2nd line: tools image

About Lvvv sign: white part of the plot (fig. 2) are detected edges corresponding to Lvvv < 0, as defined by the criterion.

In this question we study the second criterion of edge detection, which is Lvvv < 0. The lower the scale, the thinner the edges and here as well the more edges are detected. Thus, if the scale is too high not only do some edges risk not to be detected, but detected edges will be very thick, which is not desired here since it brings a lack of precision.

**Conclusion**:

The variance of the gaussian filter has an impact of the number (for both criteria) and the thickness of detected edges (for Lvvv < 0).

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

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

Answers:

As we showed in questions 4 & 5, *Lvv* gives an accurate position of too many detected edges and *Lvvv* detects the main edges but those are thick. Therefore, combining both results leads to a precise and accurate detection of main edges of the image and thus best results than by using *Lvv* or *Lvvv* only.

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

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

Answers:

Une image contenant texte

Description générée automatiquement Une image contenant texte, intérieur, outil, arrangé

Description générée automatiquement

Fig 3: Results of *extractedge* function on house and tools images

**House**: With these parameters (scale=4, threshold=6), main edges are very well detected. Some edges are incomplete, and others are completely missing though.

**Tools**: With (scale=4, threshold=9) parameters, we have even better results compared to the previous image since this one is simpler, with way less details. Edges are correctly detected even though some are incomplete. No edge has been missed by the detection. However, some points are detected as edges whereas they are not. Still, there are fewer mistakes on this image than on the *house* one.

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

**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:

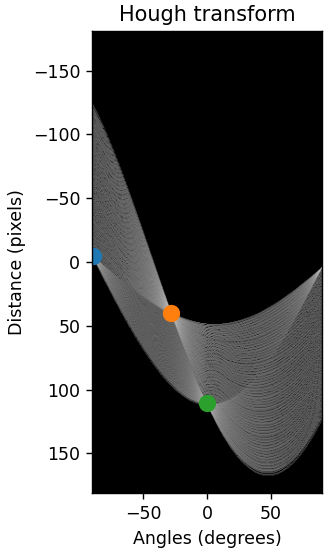
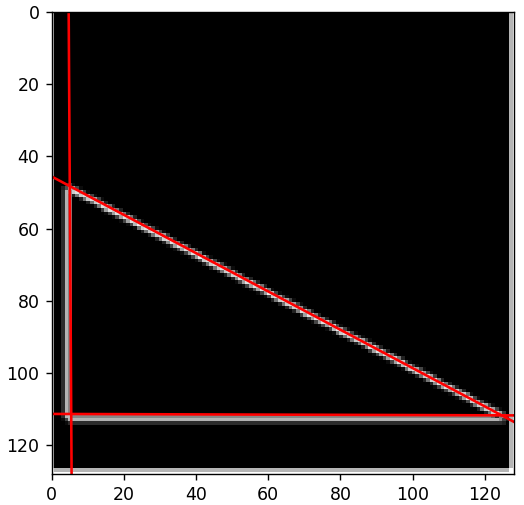
 

Fig 4: The right-hand panel is the accumulator with the 3 strongest peaks and the left-hand panel shows the detected line segments in the output image

As shows the images in the figure 4, Hough transform gives 3 local maxima in the accumulator (blue, orange and green points on the left image) that correspond to 3 red lines in the normal space (right image).

First, the lines suits to the 3 sides of the triangle so the Hough transform gives correct results.

Then, we can associate each point to a side.

The green point has an angle of 0° so it is associated to the horizontal line of the triangle. The blue point is situated at -90° so it can be linked to the vertical line. The orange point is located at a negative angle as the slope of the triangle hypotenuse.

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

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

Answers:

The algorithm that we have implemented in this lab has a computational time of O(n\*m\*ntheta) where (n, m) is the image shape and ntheta the number of theta values in the accumulator. Indeed, it first loops on each point of the image and then on each theta value of the generated interval.

Besides, the higher ntheta, the more accurate and precise the detection.

That is why there is a trade-off concerning ntheta value and accuracy.

Une image contenant texte

Description générée automatiquement

Fig 5: Results of *houghedgeline* function obtained on *tools* image

Une image contenant texte, bâtiment, maison, extérieur

Description générée automatiquement

Fig 6: Results of *houghedgeline* function obtained on *house* image

Une image contenant texte

Description générée automatiquement

Fig 7: Results of *houghedgeline* function obtained on *house* image

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

**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:

Firstly, *h* is used as the identity function. Then, another test has been done using the *log* function. The results are different but not drastically improved. Some edges that were not detected using the identity function has been detected when using the *log* and vice versa. The choice of the function therefore depends on the expectations of the user.

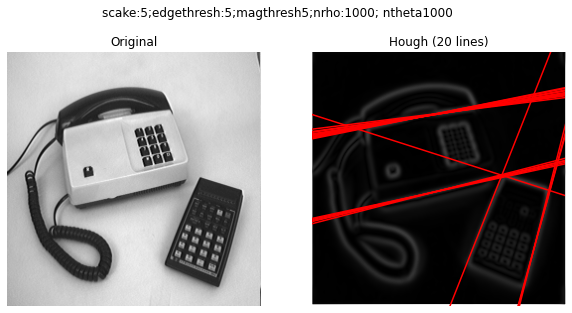


Fig 8: Results obtained with h(x) = x

Une image contenant texte

Description générée automatiquement

Fig 9: Results obtained with h(x) = log(x)

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