CS GY 6643 - Computer Vision, Spring 2024 **Guido Gerig**

Homework 1: Pencil and Paper, should take about 1.5 hrs.
SOLUTIONS

Out: 02/12/2024
Due: 02/19/2024

Late submissions: Late submissions result in 10% deduction for each day. The assignment will no longer be accepted 3 days after the deadline.

Total 35

1. Distance Transform for path planning: Distance transforms are used for path planning in robotics, for example. The following figure shows a typical example. Given the 10x10 pixel grid and the boundary regions marked as shaded "0"s, we try to find the best pixel path (pixel line) from upper left to upper right that is as far away as possible from the "0" regions to allow a safe passage.

1a) Use **d_4** and **d_8** metrics to generate a distance transform for the 10x10 grid to quantify the distances to the nearest boundaries, here marked as shaded "0"s. Please note that a pixel may see several boundaries, so that with several possibilities we choose the smallest value which identifies the distance to the closest boundary. Draw your distance values into the grids shown below, or draw the grid on paper and fill in the values.

10

2	1	0	0	0	0	0	0	1	2
3	2	1	0	0	0	0	0	1	2
3	3	2	1	1	0	0	0	1	2
2	2	3	2	1	0	0	1	2	3
1	1	2	3	2	1	1	2	3	2
0	0	1	2	3	2	2	3	2	1
0	0	0	1	2	3	3	2	1	0
0	0	0	0	1	2	2	1	0	0
0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0

D_4 Metric

10

2	1	0	0	0	0	0	0	1	2
2	1	1	0	0	0	0	0	1	2
2 ~~3~~	2	1	1	1	0	0	0	1	2
2	2	2	2	1	0	0	1	1	2
1	1	1	2	1	1	1	1	2	2
0	0	1	1	2	2	2	2	1	1
0	0	0	1	1	2	2	1	1	0
0	0	0	0	1	2	2	1	0	0
0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0

D_8 Metric

1b) Describe your strategy/algorithm to obtain these results (just briefly):

2 Best is to follow the boundaries and then extend from there. For each pixel, use the metric d4 or d8 to calculate the distance from the closest gray-shaded boundary.

1c) Given your distance transforms as solved in a (copy them over to here and then draw a thin pixel path between upper left and upper right), select a pixel path between <u>upper left and upper right</u> that is <u>most distant from the two boundary regions</u>. Please note that for thin pixel lines, we can use the **d_8** distance metric where the path can be a combination of corner and also edge neighbors. Please also note that your path **is not unique** as there may be multiple equally good solutions, but a general strategy would be to select the shortest path. Mark your selected pixel path in your distance transform results copied over from above.

5

2	1	0	0	0	0	0	0	1	2
3	2	1	0	0	0	0	0	1	2
3	3	2	1	1	0	0	0	1	2
2	2	3	2	1	0	0	1	2	3
1	1	2	3	2	1	1	2	3	2
0	0	1	2	3	2	2	3	2	1
0	0	0	1	2	3	3	2	1	0
0	0	0	0	1	2	2	1	0	0
0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0
D_4	Path								

5

2	1	0	0	0	0	0	0	1	2
2	1	1	0	0	0	0	0	1	2
2 $\cancel{3}$	2	1	1	1	0	0	0	1	2
2	2	2	2	1	0	0	1	1	2
1	1	1	2	1	1	1	1	2	2
0	0	1	1	2	2	2	2	1	1
0	0	0	1	1	2	2	1	1	0
0	0	0	0	1	2	2	1	0	0
0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0
D_8	Path								

Observation, which metric would you prefer to find a path, and why?:

3

The path through the D4 distance transform seems to make it easier to find a path as far away as possible since it includes a lot of "3's". Further, this path keeps more distance from the boundary than the D8. The D8 works as well as the path does not touch any gray-shaded area. I personally prefer the D4 distance metric for path planning.

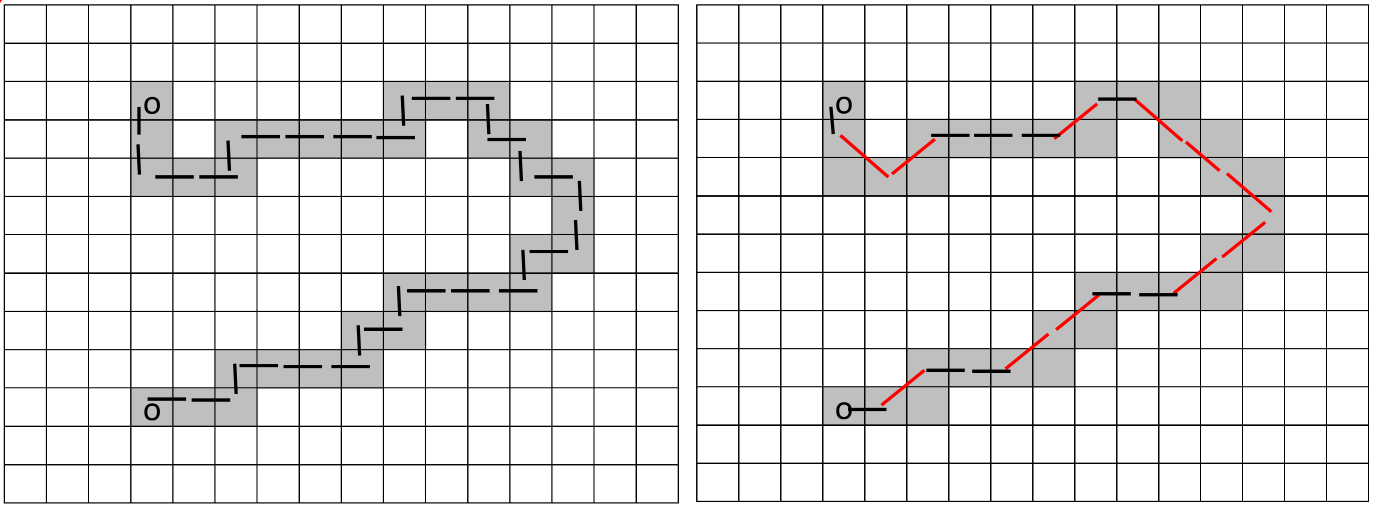
Total 20

2 Length of a discrete curve: We have discussed that the pixel grid brings challenges to measurements of distances and lengths, requiring us to use pre-defined grid connectivity and distance options. Given the following discrete pixel curve, calculate the length of the curve based on the metric used to describe the pixel-by-pixel moves from end to end. Choose the shortest path which you can find, which means to use "short-cuts" whenever you see possible.

2.1: How long is the curve if you can only use D_4 connections, i.e. only vertical and horizontal directions? Please note that pixel by pixel-by-pixel distances are defined in the metric of the distance definitions, which means that one move counts as "1". . Draw your moves into the graph to backup your result.

2.2: How long is the curve if you can use D_8 connections? Draw your moves into the graph to back up your result.

10



a) D_4 connections b) D_8 connections

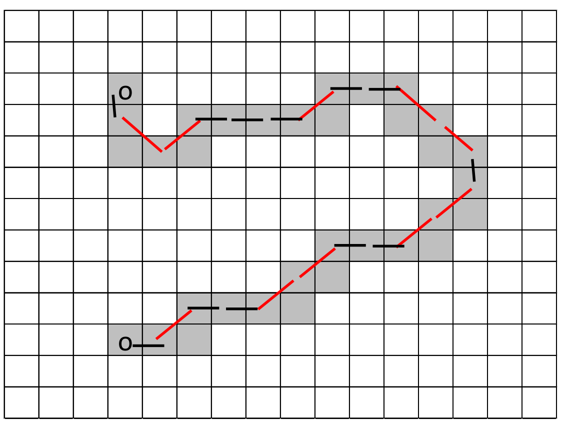
5

Solutions:

4-metric 32 moves, D_4 length: 32

8-metric: 11 diagonal moves plus 10 vertical/horizontal moves: 21 moves, D_8 length: 21

The 8-metric contour is not unique, there are other solutions which are slightly longer:



E.g. here, we have 10 diagonal and 12 vertical-horizontal moves, a total of 22 $D_{8,}$ length: 22

2.3: How long is the curve if you can use D_8 connections but then count diagonal moves with length of $\sqrt{2}$ rather than “1”?

8-metric with real lengths:

First version: 10 vertical or horizontal moves plus 11 diagonal moves: 10+11*1.41 ≈ 25.51.

Second version: 12 vertical or horizontal moves plus 10 diagonal moves: 12+10*1.41 ≈ 26.1.

5

1.3.3: Brief discussion of the three results: Would you have the three choices as above, which one would you prefer if you wanted to get a result which comes close to the real length of the curve before pixelation?

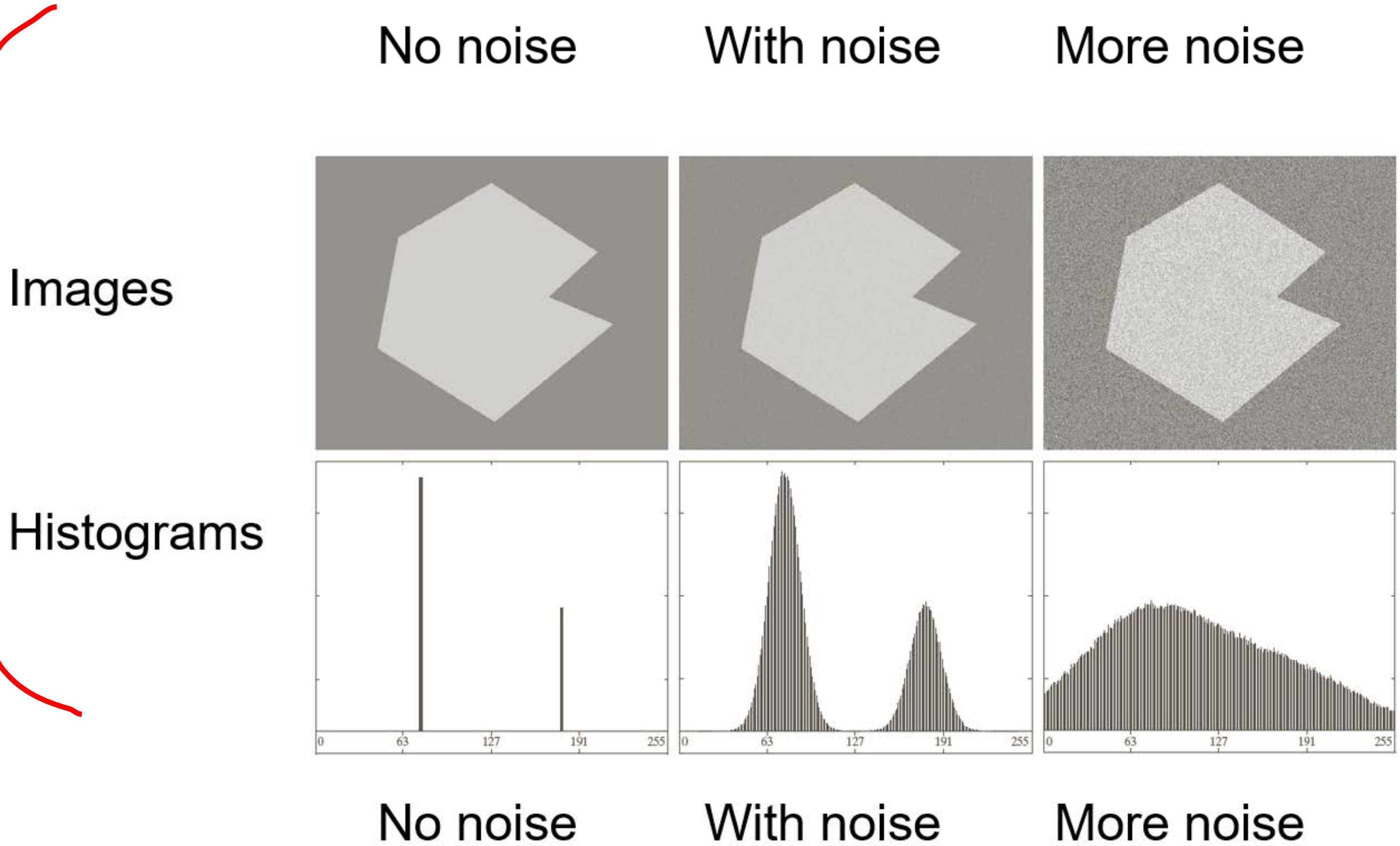
Would I have a customer who want to measure the lengths of digital contours as close as possible to a real length, I would choose the 8-metric with $\sqrt{2}$ diagonal measure (1.2.3) . It seems this comes closest to what we can measure (e.g. putting a string into the discrete image) unless we would apply a curve interpolation. Wrt the different, non-unique solutions, we may write a program that chooses the shortest path.

3) Histograms Total 25

3a) Effect of noise on histograms

Sketch the histograms for the following 3 images corrupted by different noise levels.

10



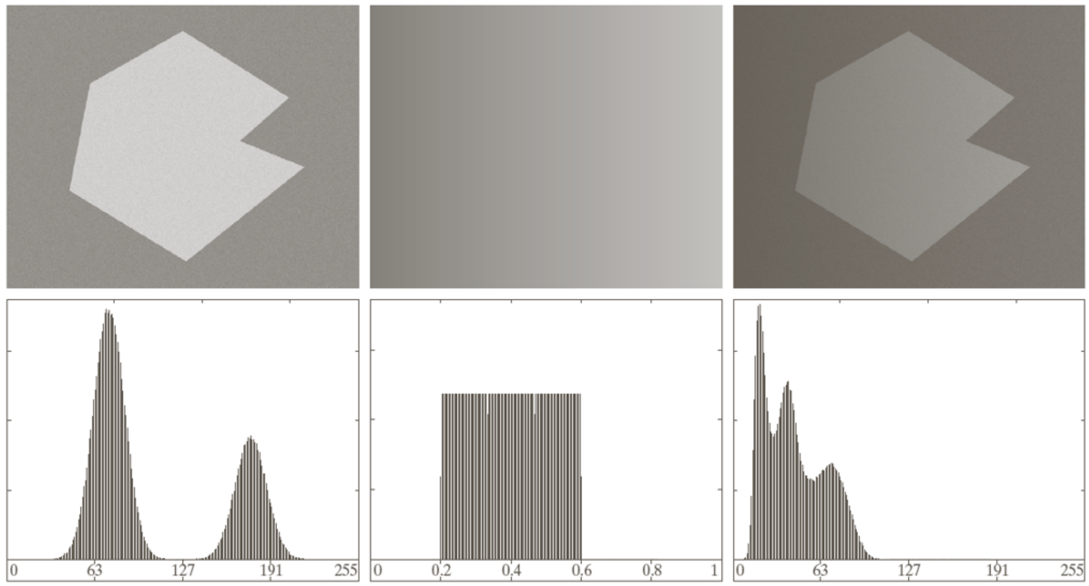
3b) Effect on illumination inhomogeneities on histograms

Sketch the histograms of the following images where the original, low-noise image is corrupted by a multiplicative inhomogeneity (here left to right change of intensity as shown in the middle image.

10

Images

Histograms



f x g = h

Original image | Illumination image | Final image

3c) Describe how noise in 3a and illumination in 3b may affect segmentation by simple image thresholding/binarization.

3a) Small noise still allows to find a threshold to segment fore- and background. With high level of noise (right image), the histogram does not show a bimodal distribution so that any threshold would result in pixels from background and foreground included into each category.

5

3b) The corruption with the illumination inhomogeneity changes intensity values within fore- and background. Qualitatively, it appears that the corrupted image cannot be segmented by a single threshold.

4: 1D image filtering Total 20

20 Given the following 1-D filter and 1-D pixel image:

4.1 Apply a 3pixel average box filter and add the resulting values into the empty boxes below.

(1/3)	1	1	1

Box Filter

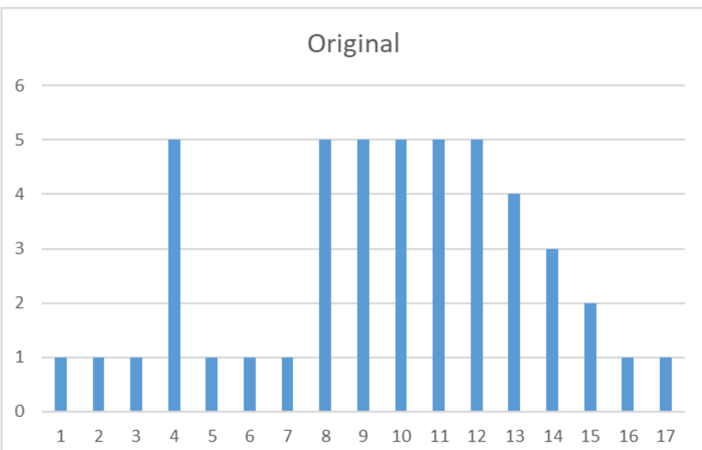
Image Signal

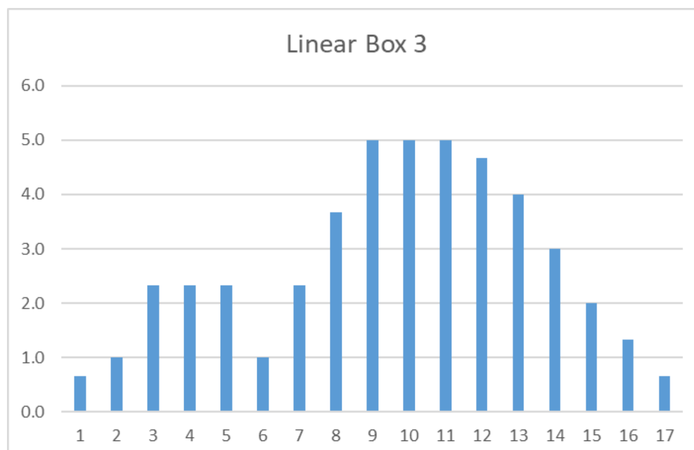
0	1	1	1	5	1	1	1	5	5	5	5	5	4	3	2	1	1	0

Filtered Signal

	0.7	1.0	2.3	2.3	2.3	1.0	2.3	3.7	5.0	5.0	5.0	4.7	4.0	3.0	2.0	1.3	0.7	

4.2 Plot the original image signal into the left graph, and the filtered signal into the right one. The horizontal axis is the pixel location, and the vertical axis the intensity value.





4.3 Briefly discuss the result after linear filtering when comparing the two plots.

Box filtering results in smoothing of the original signal. The outlier at position 4 is smoothed but it also affects neighboring pixels getting higher values. Also, the step edge at position 8 gets blurred..