CS 376A
Digital Image Processing

01 / 30 / 2023

Instructor: Michael Eckmann

Today's Topics

- Questions/Comments?
- Code review of by RGBPixel.java, RGBImage.java classes and a program using them
- Start Cross-correlation of masks and images
  - Edge detection
  - smoothing

Java classes for the course

- Let's review code I will provide that reads and writes jpeg images and allows access to the 2d array of pixels. Each pixel of which contains a red, green and blue value in the 0..255 range inclusive.

Masks / Cross-Correlation / Edges

- Consider the problem of determining which pixels are (part of an) edge and which are not.
- Let's simply define an edge pixel to be where there is a significant difference in intensity near the pixel.
- Also let's consider vertical or horizontal edges only.

Let's look at a 1d example

- Let's consider this mask

  1 0 -1

- Applied to this 1 dimensional image:

  50 50 50 50 50 75 75 75 75 75

Let's look at an example

- Let's consider this mask

  1 0 -1

- Applied to this 1 dimensional image:

  50 50 50 50 50 125 125 125 125 125

- Resulting image:

  0 0 0 -75 -75 0 0 0

Let's look at an example

- Let's consider this mask

  1 0 -1

- Applied to this 1 dimensional image:

  50 50 50 50 50 125 125 125 125 125

- Resulting image:

  0 0 0 -75 -75 0 0 0

- Notice: in areas of constant values, the result is 0, but in areas where there are differences/high contrast, the result is non-zero (high absolute value).

Filters / Cross-correlation

- A mask (aka filter) is usually a small square grid of values
- Cross-correlation is a procedure that applies the mask to an image in the following way:
  - Given a mask of values, multiply each value in the mask by the corresponding pixel value in the image and add them up
  - Use the resulting value as the output pixel at the center

Derivative and Smoothing masks

- Derivative masks
  - values have opposite signs in order to obtain a high response in signal regions of high contrast
  - sum of mask values is 0 and a 0 results from applying it to constant regions
  - first derivative masks produce high absolute values at points of high contrast
- Smoothing masks
  - values are positive and sum to 1
  - resulting value is same as input in constant regions
    - so smoothing a constant region does nothing
  - in non constant regions the resulting value is a weighted average of the region's pixels

Smoothing examples

Smoothing examples

Smoothing examples

Homework

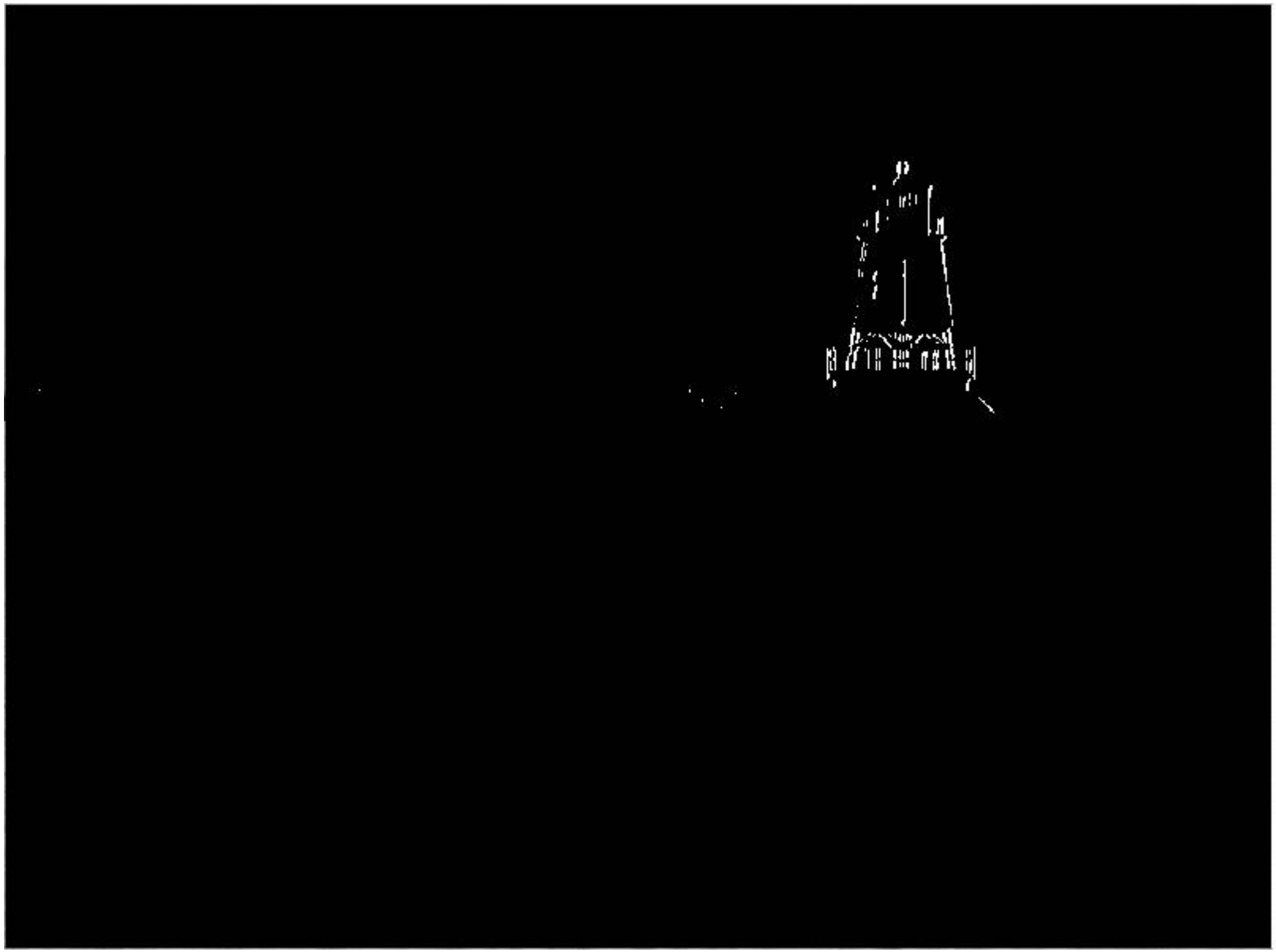
- Consider the name “derivative” mask/filter. Why are they called that do you think? Think about derivative of a continuous function. Are we working with continuous or discrete in this course?
- Read handouts.

digital image

- Notice that a digital image is a 2d discrete function that relates to a 2d continuous function, sampled regularly.

Horizontal Edges

Vertical Edges



Let's do an example on the board

- Those last images used a threshold of 150 to determine edge or not
- Let's create a mask for edges to be cross-correlated with a 2d image (as opposed to that 1 dimensional example)
- We can create masks for vertical and horizontal edges.
- We also could create masks for the 2 different diagonal edges.
- Present Prewitt and Sobel masks.

Enhancing Images

- Removal of salt-and-pepper noise from greyscale images is typically done with a median filter
  - consider a neighborhood around each pixel,
  - sort the grey values and choose the median value (the one that appears in the middle of the sorted list)
  - e.g. a 5x5 neighborhood will have 25 pixels, the median is the 13th highest value
  - example on the board w/ a 3x3 median filter

Enhancing Images

- Contrast the median filter with an averaging (aka smoothing) filter
- A smoothing filter computes a new value of a pixel by the following
  - given a mask of values, multiply each value in the mask by the corresponding pixel value in the image and add them up
  - then divide by the total of the values in the mask
  - if the total of the values in the mask is 1 (each value is some fraction --- the weight) --- no need to divide by the total
- A box filter is a simple smoothing filter which uses an equally weighted rectangular neighborhood.
- A gaussian filter is a smoothing filter that gives most weight to center pixel and gradually tapers off the weight the further from the center.
- example of a few masks on the board (search 2d Gaussian filter)

Cross-correlation

- The process described on the last slide is similar to cross-correlation of an image and a mask.
- Cross-correlation involves aligning the origin of the mask with a pixel in question and multiplying the corresponding mask pixel values with the pixels in the original image and summing them all up. This summed up value is the value of the new pixel. No divide takes place.
- Typical masks though will either sum up to 0 or 1. A smoothing mask will sum up to 1. A derivative mask will sum up to 0.
- examples on the board of square masks with these properties

Considerations

- resulting value of output image pixels can be much larger than the input image pixels due to the summation, it is also possible for the output image pixels to yield negative values if the mask contains negatives --- so the type of the new image needs to be considered as possibly being different than the input image type
- what to do at borders of the image when cross-correlating or convolving and image and a mask and the mask overlaps (doesn't have corresponding pixels for one or more mask values.)
  - no processing -> make all the border pixels 0 in output image; copy the border pixels from input to output; make the output image smaller; reduce the size of the mask at the border locations; reflected indexing; circular indexing
    - note: this list comes from “Digital Image Processing” by Nick Efford.

Gaussian masks

- Program to generate gaussian masks
  - specify the width/height of the mask
  - specify the variance of the gaussian
  - let's examine some masks that have different variances
    - e.g. 7x7 mask w/ variance 0.5, 1, 1.5