Homework 2

Greg Dews | CS390S | 9/22/2019

Some Problems

Draft submission

# The Code

Goals:

* Read a photo and convert it to grayscale.
* Apply filters: Sobel, Laplacian, Median
* Using iris.bmp, detect edge of pupil
* Use thresholding and mask out noise in background

Python Libraries used:

* numpy – for making kernels
* cv2 – for convolution and thresholding

# The Process

Loaded a picture of my daughter and converted to greyscale.

A close up of a child wearing a hat

Description automatically generated

Applied an average filter with a 5x5 kernel of all ones, divided by 25.

A picture containing person, indoor, next, animal

Description automatically generated

Applied two different Sobel kernels for horizontal and vertical line detection.

sobelx\_weights = np.array([

    [ 1, 1, 1],

    [ 0, 0, 0],

    [-1,-1,-1]])

sobely\_weights = np.array([

    [-1, 0, 1],

    [-1, 0, 1],

    [-1, 0, 1]])

Added them together and got this as my resultant Sobel filter process.

A close up of a person

Description automatically generated

Then I applied a Laplacian kernel with a weight of 8 for the center, with the surrounding 8 neighbors all with a weight of -1.

A close up of a person

Description automatically generated

# Some Problems

After attempting to get things working through the ndimage library and having nothing work correctly with the convolve function. I then tried PIL.ImageFilter to use my kernels with all my pictures being blacked out. The result was me rewriting everything using the CV2 library with numpy arrays. I can’t figure out how to threshold without iterating through the whole image myself and am having trouble finishing the iris detection portion of the project. I am deciding to send a draft of what I have and continue the project after asking some questions in class.

# Reflection

The most difficult part of this homework was in finding the appropriate Python library to use and learning it’s API to achieve what I needed (Again).

It was harder to set my own kernels in python than it would have been to use a preset API command. This is possibly due to a lack of consistency between the libraries, but this gave me more experience working through the problems and knowing what to expect.

**Unfinished, submitting draft**

**still need:**

**to figure out how to threshold correctly in python**

**to create a correct median filter- not achieved through kernel weights**