Homework 0 Questions

CSCI 1430

Template Instructions

This document is a template with specific answer regions and a fixed number of pages. Given large class sizes and limited TA time, the template helps the course staff to grade efficiently and still focus on the content of your submissions. Please help us in this task:

- Make this document anonymous.
- Questions are in the orange boxes. Provide answers in the green boxes.
- Use the footer to check for correct page alignment.
- **Do NOT remove the answer box.**
- **Do NOT change the size of the answer box.**
- **Extra pages are not permitted unless otherwise specified.**
- **Template edits or page misalignment will lead to a 10 point deduction.**

Gradescope Submission

- Compile this document to a PDF and submit it to Gradescope.
- Pages will be automatically assigned to the right questions on Gradescope.

This Homework

- 7 questions **[4 + 4 + 8 + 6 + 5 + 3 + 3 = 33 points]**.
- Include code, images, and equations where appropriate.

Q1 — [4 points]

For each of the following, complete the task and fill the box to mark it as done.

- ■ This is an example of a filled box.
- ☐ Create a GitHub account, as needed, and read the GitHub tutorial.
- ☐ Check that you are in the course's Gradescope.
- ☐ Check that you are in the course's Ed.
- ☐ If you're rusty, complete the numpy tutorial.
- ☐ Set up the course's python environment (csci1430).
- ☐ Set up an editing environment (e.g., VSCode), set it to use your python virtual environment, and know how to debug within it by setting breakpoints.

Only Q1 should be on this page

Q2 — [4 points]

Please find and read the course policies on the course website and check whether each of the following scenarios violates the policy.

(a) Search for Websites to help you clarify concepts, with citation in your submission.

- ☐ Acceptable
- ☐ Violation

(b) Ask AI about a concept in the course, without citation in your submission.

- ☐ Acceptable
- ☐ Violation

(c) Ask AI a question from the homework and submit its answer with minor changes, with citation in your submission.

- ☐ Acceptable
- ☐ Violation

(d) Look at another student's code before you write your own.

- ☐ Acceptable
- ☐ Violation

(e) Talk to another CSCI 1430 student about your code to help you debug, after you have spent time trying to tackle the bug or have come to TA office hours/Ed.

- ☐ Acceptable
- ☐ Violation

(f) Looking at code from a student who has previously taken the course to help you debug. The student is not a TA.

- ☐ Acceptable
- ☐ Violation

(g) Using the result images from another student's code because your code has a bug.

- ☐ Acceptable
- ☐ Violation

Q3 — [8 points]
Computer vision is all around us, sometimes in surprising ways.

Q3.1 — [2 points]

If you could have any computer vision related superpower, with no limitations, what would it be and how would you use it? **[2-3 sentences]**

Q3.2 — [6 points]
All students enter the course with different backgrounds in socially responsible computing. Please first review our ethics primer that introduces basic concepts to everyone.

Q3.2.1 — [3 points]

State one value that is affected if you use your superpower. Explain your reasoning. **[2–3 sentences]**

Q3.2.2 — [3 points]

State one value that is affected if your archenemy used your superpower. Explain your reasoning. **[2–3 sentences]**

Q4 — [6 points]
Here is an image: grizzlypeak-grayscale.png (in the images directory)

Q4.1:
Below is some code that sets pixels that have a value of 50 or less to 0. This removes some of the lower-intensity haze around the bright lights. However, the code only works on single-channel grayscale images.

Q4.1.1 — [2 points]

Using another for loop and other appropriate modifications, convert the following code to work on grizzlypeak-color.png (also in the images folder). You can paste the code in a Python file to run it. Make sure the csci1430 environment is activated.

TODO: Modify the following code:

```
from skimage import io

A = io.imread('grizzlypeak-grayscale.png')
height, width = A.shape

# TODO: introduce a for loop that allows this
# code to work on RGB images
for i in range(height):
    for j in range(width):
        if A[i,j] <= 50 :
            A[i,j] = 0
```

Let's find the time it takes to run this operation once, over one image. Because a single short task on multitasking computers often takes variable time, we can record how long code execution takes on a small number of images and then average the execution time.

Q4.1.2 — [1 point]

Record how long it takes to execute 10 times. Find the average time it takes to process a single image by dividing the total time by 10. Please include your code.

Note: When measuring the time, please ignore the file loading. You should only time the image modification itself.

```
# TODO: your code here
```

Report: How long does it take? TODO

Q4.2 — [3 points]
Let's say we want to run our image modification 1000 times. If we used our current implementation, it would be slow. Let's suppose we can find a faster solution.

Q4.2.1 — [2 points]

Use logical indexing as a faster solution to remove the haze. If you don't know what this is, please see the Python/numpy tutorial.

```
# TODO: Your code here
```

Q4.2.2 — [1 point]

Now, using your faster solution, record how long it takes to run our image modification over 1000 images. How long does the new code take to change one image? How much faster is the new version per image, as a multiplicative factor (e.g., 2x, 5x) compared to the naive solution in 4.1.2? Please include your code.

```
# TODO: your code here
```

Report: How long does it take? TODO

Q5 — [5 points]
Your friend tries to write code that reads an image into memory and displays it.

```
from skimage import io
import matplotlib.pyplot as plt
import numpy as np

I = io.imread('./images/gigi.jpg').astype(np.float32)
plt.imshow(I)
plt.show()
```

However, when they run the code, the image is almost entirely white. Confused, they come to you for help.

Q5.1 — [1 point]

Why is the image being displayed incorrectly?
Hint: What data values does plt.imshow() expect?

Q5.2 — [2 points]
You and your friend think of two possible ways to display the image correctly.

Q5.2.1 — [1 point]

Write a possible fix that uses the float data type.

```
# TODO: your code here
```

Q5.2.2 — [1 point]

Write a possible fix that uses the 8-bit unsigned integer data type.

```
# TODO: your code here
```

Q5.3 — [2 points]

Your friend comes to you and wants help darkening an image.
Read in 'gigi.jpg', darken all pixels by an amount equivalent to 20% of the maximum brightness value of the *pixel data type*, and display the new image. Include your code and a screenshot of the darkened image.

```
# TODO: your code here - you can just write the critical lines
```

Q6 — [3 points]
The debugger within VSCode is an important tool you can use to discover potential bugs in the code that you write.

Imagine our task is to create a crop of an image that starts at the center of the image and extends to the lower right corner of the image. If all goes well, we should only see content from the lower right region of the original image.

Image: gigi.jpg (in images folder)

```
from skimage import io
import matplotlib.pyplot as plt

origImage = io.imread('./images/gigi.jpg')
(height, width, channels) = origImage.shape
startCropX = width % 2
startCropY = height % 2
croppedImage = origImage[startCropY:, startCropX:]

plt.imshow(croppedImage)
plt.show()
```

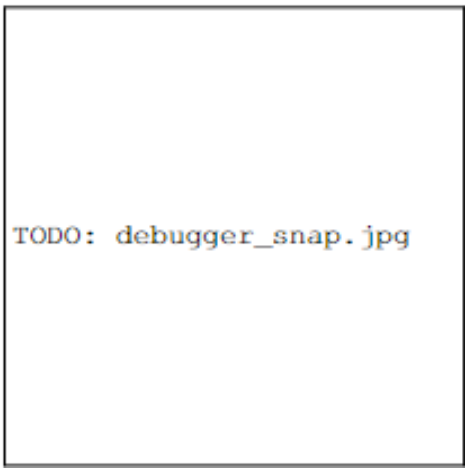
Create a new python file in the same directory as the image, and copy in the above code block. Then, open the file in VSCode, and execute the code within a debugging session by pressing F5 (or 'Run → Start Debugging'). At the prompt, we wish to 'Debug the currently active Python file'.
The output is not currently what we want, so let's stop execution and then identify the bug in this program:

1. First, set a breakpoint at line 7 and then re-execute the code for debugging.
2. Inspect the 'startCropX' variable either by looking at the left-hand Variables panel or by mouse hovering over the variable in the editor. What should it be?
3. Execute line 7 of code by 'stepping over' the current line (F10, or 'Run → Step Over). We should now be about to execute line 8.
4. Inspect 'startCropY' and verify its correctness.

At this point, you might have an idea of how to fix the code. But, before stopping execution and editing the file, let's test out our hypothesis in the 'Debug Console' during debugging.

1. Switch to the Debug Console by pressing CTRL-SHIFT-Y (or 'View → Debug Console') — you should see it in the bottom right of the display screen.
2. *This is an interactive Python console with access to working memory.* As a test, print out the value of 'width'. Perform a mathematical operation on 'width'.
3. Assign the right value to 'startCropX' within the Debug Console. Notice how the value updated in the Variables panel.
4. Do the same for 'startCropY'.
5. From this point, execute the rest of the code by Continuing beyond our current paused position in the code. Press F5 to Continue (or 'Run → Continue').

Re-execute the debugger, and capture a screenshot showing your use of the Debug Console and inspection of a variable. Also, write the correct code below.




```
    # TODO: paste your code here
```

Q7 — [3 points]
This program should print out the maximum value in the matrix obtained by multiplying a random non-square matrix with its transpose.

Here, we're using some numpy functions that may be new to us, but they each have self-explanatory names.

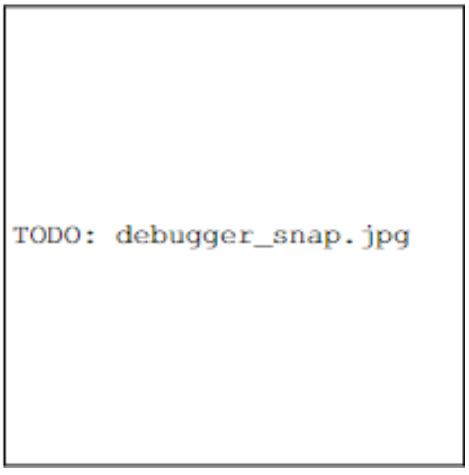
```
import numpy as np
from numpy import random as r

mat_1 = r.rand(200,150)
mat_2 = mat_1
np.transpose(mat_2)
mat_3 = np.matmul(mat_1, mat_2)
mat_max = np.max(mat_3)

print("Max value:", mat_max)
```

This time, when we execute the code, it will raise an exception.

> Run the code in a debugging session, note the exception, and inspect the variables. Form a hypothesis for the error, set a breakpoint before it, and use the Debug Console to test that it prevents the exception.

Hint: Remember rules about matrix multiplication. What should the dimensions of each matrix be? Use the debugger to notice how the shapes of the images do or do not change.

Capture a screenshot of your session showing us the issue and paste the correct code.



```
# TODO: paste your code here
```

Only Q7 should be on this page

Feedback? (Optional)

We appreciate your feedback on how to improve the course. You can provide anonymous feedback through this form which can be accessed using your Brown account (your identity will not be collected). If you have urgent non-anonymous comments/questions, please email the instructor.