Homework 1 Written Questions

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.**

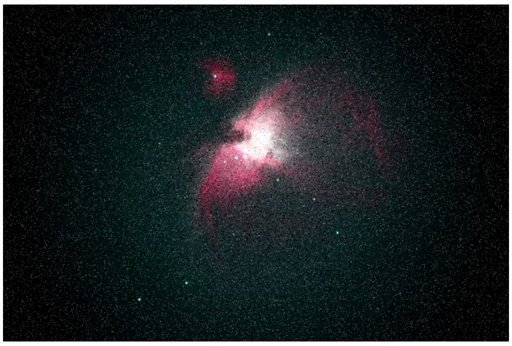
Gradescope Submission

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

This Homework

- 5 questions **[12 + 6 + 13 + 14 + 7 = 52]**.
- Include code, images, and equations where appropriate.

Q1: [12 points] We have been given special permission to use the telescope on the roof of Barus and Holley. Unfortunately, our fantastic image of the Orion nebula has noise caused by the imaging sensor: (orion-noise.png)



One way to deal with this noise is with image convolution. Convolution is a type of image filtering that is a fundamental image processing tool.

(a) *Explicitly describe* the input, transformation, and output components of 2D discrete convolution. Please be precise; define variables as need.

(i) **[2 points]** Input **[2–4 sentences]**

(ii) **[2 points]** Transformation (how is the image transformed?) **[2–4 sentences]**

(iii) **[2 points]** Output **[2–4 sentences]**

(b) **[4 points]**

Describe two filter kernels that we may use with convolution, and give an example computer vision application that each enables. **[4–8 sentences]**

(c) **[2 point]**

What kind of filter might we use to de-noise our image of the Orion nebula, and why? **[2–3 sentences]**

Q2: **[6 points]** Now that we've de-noised our image of the Orion nebula, let's explore filtering techniques more closely. Two kinds of linear filtering are correlation and convolution.

(a) **[3 points]**

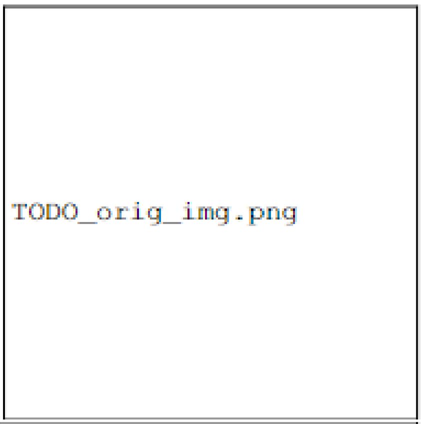
What is different between convolution and correlation? Include differences in their algebraic properties. When might we use each? **[5–6 sentences]**

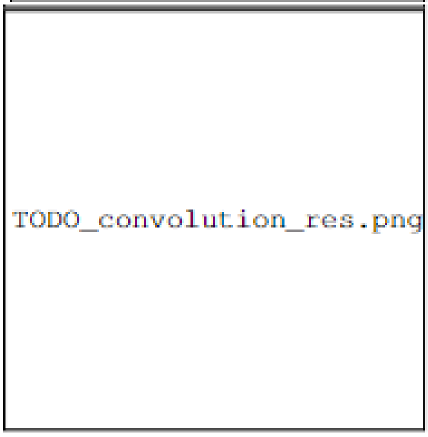
(b) **[3 points]** To solidify our understanding of the distinction between correlation and convolution, we will process another image.

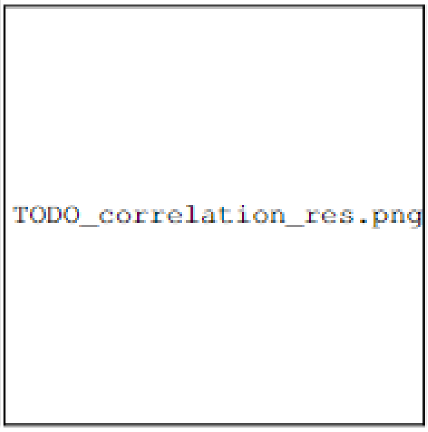
Devise a scenario in which the output of correlation and convolution differ. Write code that loads an image and produces two distinct images, one from convolution and one from correlation on some kernel of your choice. Specify your kernel, and provide the input image and two output results. Then, use your understanding of convolution and correlation to explain the outputs. **[2–4 sentences]**

Consider $scipy.signal.convolve2d$ *and* $scipy.signal.correlate2d$ *to experiment!*

Only A2 (a) - (b) should be on this page

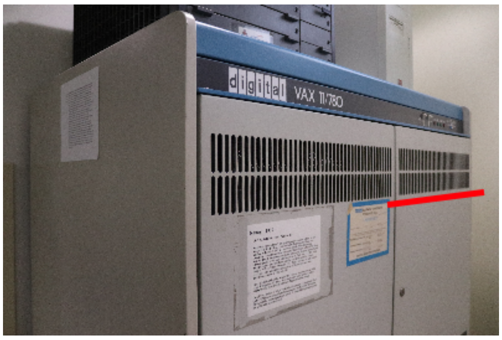






TODO: Your answer for (b) here

Q3: [13 points] While exploring Brown CS's history in the halls of CIT, we happen upon Nancy: a DEC VAX 11/780. So struck by its beauty, we decide to take an artful photo with our camera. Modern digital sensors have many megapixels, so we resize it to make the file smaller.



Resized image

Depiction of original (*not original file*)

(a) **[3 points]** Oh no! What happened to Nancy? There are weird artifacts in the vents (above red line)—these definitely weren't there in the original photo. Plus, if we look closely, the white label text is less smooth and there are jagged lines.

What is this phenomenon called, and why did it happen? **[2–4 sentences]**

TODO: Your answer for (a) here

(b) **[3 points]**

How might we fix this issue with filtering? Describe the process, and explain why it works. **[2–4 sentences]**

TODO: Your answer for (b) here

(c) **[3 points]**

Which of the following kernels is high pass, low pass, or neither?

Note: To fill in boxes, replace '\square' with '\blacksquare' for your answer.

(i) $\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$

TODO: Select the appropriate answer.
- ☐ High pass
- ☐ Low pass
- ☐ Neither

(ii) $\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}$

TODO: Select the appropriate answer.
- ☐ High pass
- ☐ Low pass
- ☐ Neither

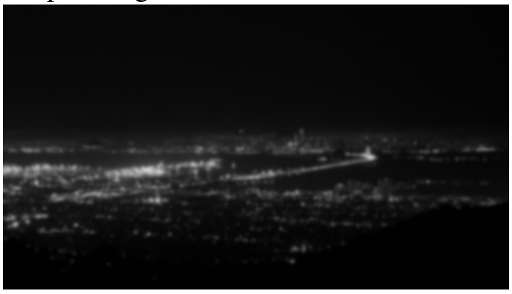
(iii) $\begin{bmatrix} -\frac{1}{9} & -\frac{1}{9} & -\frac{1}{9} \\ -\frac{1}{9} & \frac{8}{9} & -\frac{1}{9} \\ -\frac{1}{9} & -\frac{1}{9} & -\frac{1}{9} \end{bmatrix}$

TODO: Select the appropriate answer.
- ☐ High pass
- ☐ Low pass
- ☐ Neither

(d) **[2 points]** We decide to test if we can recognize which kind of filter has been used to achieve a target output image.

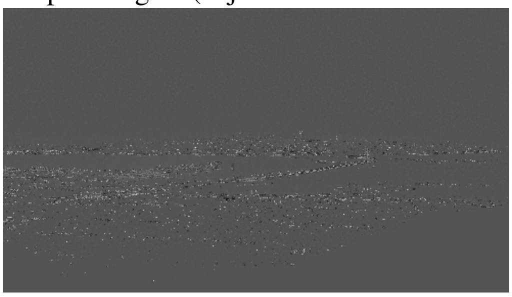
Given the input image below, identify the kind of filter.

(i) Output image 1:



TODO: Select the appropriate answer.
☐ High pass
☐ Low pass

(ii) Output image 2 (adjusted for easier visualization):



TODO: Select the appropriate answer.
☐ High pass
☐ Low pass

(e) **[2 points]**

Which of the following statements are true? (Check all that apply).

TODO: Select all that apply.

- ☐ High pass filter kernels will always contain at least one negative number
- ☐ A Gaussian filter is an example of a low pass filter
- ☐ A high pass filter is the basis for most smoothing methods
- ☐ In a high pass filter, the center of the kernel must have the highest value

Q4: **[14 points]** With filtering, we can create *hybrid images* that depict different objects when viewed at different distances. They are inauthentic images of the natural world.

As technology advances, evaluating the authenticity of images becomes increasingly difficult. Please read this article by photography critic Andy Grundberg in the *New York Times* from August 1990.

Grundberg stated that: "In the future, readers of newspapers and magazines will probably view news pictures more as illustrations than as reportage, since they can no longer distinguish between a genuine image and one that has been manipulated."

(a) **[4 points]**

When is Grundberg's future, and why? **[4–6 sentences]**

(b) **[4 points]**

For a news picture, are any digital manipulations permissible? If so, how do we decide which ones? **[4–6 sentences]**

(c) **[4 points]** The Coalition for Content Provenance and Authenticity (C2PA) has designed a technical specification to attach history to an image. Please watch this video for an overview; stop at 4 minutes and 40 seconds.

Describe one situation in which the C2PA system helps us in determining authenticity, and one situation in which it does not. Please explain why in each case. **[4–6 sentences]**

(d) **[2 point]** Grundberg's article is titled "Ask It No Questions: The Camera Can Lie."

Does the C2PA system weaken Grundberg's argument? **[1–2 sentences]**

Q5: Technical practice — [7 points] In computer vision, each image is a matrix of pixels. The `numpy` library provides fast computation with large multi-dimensional vectors and matrices.

To familiarize yourself with the library, read through the following scenarios and complete the exercises. Write *one* `numpy` function to complete each of the following tasks.

With `numpy` imported as

```
import numpy as np
```

we can call functions with

```
np.function_name(<arguments>)
```

Test out your answers by creating your own python program. Some functions you might find useful are np.squeeze, np.expand_dims, np.clip, np.pad, and np.zeros.

Use operators like `[]` and `:`, but remember that each prompt can be completed with only *one* function/operator shorthand.

(a) **[1 point]** Create a black image `img` with all values in this matrix equal to 0.

> Create `img` where `np.shape(img) == (320,640)`.

```
# TODO: Your expression here
```

(b) **[1 point]** A filtering operation seems to mess up the output dimensions, producing a variable `img_out` where `np.shape(img_out) == (1, 1, 320, 640)`.

> Remove all 1-sized dimensions. Convert `img_out` to a new matrix `img_fixed` where `np.shape(img_fixed) == (320, 640)`.

```
# TODO: Your expression here
```

(c) **[1 point]** Color images are represented with three dimensions. The first dimension represents the number of color channels, and could help us to identify whether an image is color (RGB) or grayscale.

Say you have a grayscale image `img` where `np.shape(img) == (320, 640)`. Convert this to a new image `img_expanded` where `np.shape(img_expanded) == (1, 320, 640)`. In other words, add a 1-sized dimension to `img`.

```
# TODO: Your expression here
```

(d) **[1 point]** Assume we have a 2D matrix `img` with values rage [-1.0, 1.0].

Clip `img` so that all its values lie within the range [-0.5, 0.5].

```
# TODO: Your expression here
```

(e) **[1 point]** Suppose we have an RGB image matrix, `img`, of shape (320, 640, 3). We call each color matrix a channel of information.

Retrieve the third blue channel of the image while preserving all of `img`'s dimensions and intensity values.

```
# TODO: Your expression here
```

(f) **[1 point]** Suppose we have a second RGB image matrix, `img2`, also of shape (320, 640, 3).

Retrieve the red and blue channels of `img2` while preserving all of `img2`'s dimensions and intensity values.

```
# TODO: Your expression here
```

(g) **[1 point]** Padding is a useful operation to help us produce a convolved image equal in size to an input image.

Given an RGB image, `img`, pad it with two columns of zeros on the left and right edges of the image, and three rows of zeros on the top and bottom edges of the image. Do not add zeros to the color dimension.

```
# TODO: Your expression here
```

Feedback? (Optional)

Please help us make the course better. If you have any feedback for this assignment, we'd love to hear it!