Homework 1 - Basic Convolution

1 Convolution (2 points)

Recall the definition of convolution,

$$g = I \otimes f \tag{1}$$

where ***I*** and ***f*** represents the image and kernel respectively.

Typically, when kernel f is a 1-D vector, we get

$$g(i) = \sum_{m} I(i-m) f(m) \tag{2}$$

where i is the index in the 1-D dimension.

If the kernel f is a 2-D kernel, we have

$$g(i,j) = \sum_{m,n} I(i-m, j-n) f(m,n) \tag{3}$$

where i and j are the row and column indices respectively.

In this section, you need to perform the convolution **by hand**, get familiar with convolution in both 1-D and 2-D as well as its corresponding properties.

Note: All convolution operations in this section follow except additional notifications: 1. Zero-Padding, 2. Same Output Size, 3. An addition or multiplication with 0 will count as one operation.

For this problem, we will use the following 3×3 image:

$$I = \begin{bmatrix} 0.0 & 1.0 & -1.0 \\ 2.0 & 1.0 & 0.0 \\ 0.0 & 3.0 & -1.0 \end{bmatrix} \tag{4}$$

You are given two 1-D vectors for convolution:

$$f_x = \begin{bmatrix} -1.0 & 0.0 & 1.0 \end{bmatrix} \tag{5}$$

$$f_y = \begin{bmatrix} 1.0 & 1.0 & 1.0 \end{bmatrix}^T \tag{6}$$

Let $g_1 = I \otimes f_x \otimes f_y$, $f_{xy} = f_x \otimes f_y$ and $g_2 = I \otimes f_{xy}$.

Note: f_{xy} should be of full output size.

- **Question 1.1:** Compute g_1 and g_2 (At least show two steps for each convolution operation and intermediate results), and verify the associative property of convolution.

  Answer 1.1: $g_1 = I \otimes f_x \otimes f_y$

$$g_1 = I \otimes f_x \otimes f_y = \begin{bmatrix} 0.0 & 1.0 & -1.0 \\ 2.0 & 1.0 & 0.0 \\ 0.0 & 3.0 & -1.0 \end{bmatrix} \otimes \begin{bmatrix} -1.0 & 0.0 & 1.0 \end{bmatrix} \otimes \begin{bmatrix} 1.0 & 1.0 & 1.0 \end{bmatrix}^T \tag{7}$$

$$= \begin{bmatrix} 0.0 & 1.0 & -1.0 \\ 2.0 & 1.0 & 0.0 \\ 0.0 & 3.0 & -1.0 \end{bmatrix} \otimes \begin{bmatrix} -1.0 & 0.0 & 1.0 \end{bmatrix} \otimes \begin{bmatrix} 1.0 \\ 1.0 \\ 1.0 \end{bmatrix} \tag{8}$$

$$= \begin{bmatrix} 0.0+0.0-1.0 & 0.0+0.0+1.0 & 1.0 \\ 0.0+0.0-1.0 & 2.0+0.0+0.0 & 1.0+0.0+0.0 \\ 0.0+0.0-3.0 & 0.0+1.0+0.0 & 3.0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{9}$$

$$= \begin{bmatrix} -1.0 & 1.0 & 1.0 \\ -1.0 & 2.0 & 1.0 \\ -3.0 & 1.0 & 3.0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{10}$$

$$g_1 = \begin{bmatrix} -1.0-1.0 & 1.0+2.0 & 2.0 \\ -1.0-1.0-3.0 & 1.0+2.0+1.0 & 5.0 \\ -1.0-3.0 & 2.0+1.0 & 4.0 \end{bmatrix} \tag{11}$$

$$g_1 = \begin{bmatrix} -2.0 & 3.0 & 2.0 \\ -5.0 & 4.0 & 5.0 \\ -4.0 & 3.0 & 4.0 \end{bmatrix} \tag{12}$$

$$f_{xy} = f_x \otimes f_y \tag{13}$$

$$= \begin{bmatrix} 0-1 & 0+0 & 1 \\ 0-1+0 & 0+0+0 & 1 \\ -1+0 & 0+0 & 1 \end{bmatrix} \tag{14}$$

$$= \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \tag{15}$$

$$g_2 = I \otimes f_{xy} \tag{16}$$

$$= \begin{bmatrix} 0 & 1 & -1 \\ 2 & 1 & 0 \\ 0 & 3 & -1 \end{bmatrix} (convolution) \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \tag{17}$$

$$g_2 = \begin{bmatrix} -1.0-1.0 & 1.0+2.0 & 2.0 \\ -1.0-1.0-3.0 & 2.0+2.0 & 5.0 \\ -1.0-3.0 & 1.0+2.0 & 4.0 \end{bmatrix} \tag{18}$$

$$g_2 = \begin{bmatrix} -2.0 & 3.0 & 2.0 \\ -5.0 & 4.0 & 5.0 \\ -4.0 & 3.0 & 4.0 \end{bmatrix} \tag{19}$$

$$g_1(convolution)g_2 = \tag{20}$$

$$= \begin{bmatrix} -2.0 & 3.0 & 2.0 \\ -5.0 & 4.0 & 5.0 \\ -4.0 & 3.0 & 4.0 \end{bmatrix} (convolution) \begin{bmatrix} -2.0 & 3.0 & 2.0 \\ -5.0 & 4.0 & 5.0 \\ -4.0 & 3.0 & 4.0 \end{bmatrix} \tag{21}$$

$$= \begin{bmatrix} -8-15-8-15 & -10+12-10-10+12-10 & 15+8+8+15 \\ -76 & -100 & 76 \\ -62 & -56 & 62 \end{bmatrix} \tag{22}$$

$$= \begin{bmatrix} -46 & -16 & 46 \\ -76 & -100 & 76 \\ -62 & -56 & 62 \end{bmatrix} \tag{23}$$

$$g_2(convolution)g_1 = \tag{24}$$

$$\begin{bmatrix} -2.0 & 3.0 & 2.0 \\ -5.0 & 4.0 & 5.0 \\ -4.0 & 3.0 & 4.0 \end{bmatrix} (convolution) \begin{bmatrix} -2.0 & 3.0 & 2.0 \\ -5.0 & 4.0 & 5.0 \\ -4.0 & 3.0 & 4.0 \end{bmatrix} \tag{25}$$

$$= \begin{bmatrix} -8-15-8-15 & -10+12-10-10+12-10 & 15+8+8+15 \\ -76 & -100 & 76 \\ -62 & -56 & 62 \end{bmatrix} \tag{26}$$

$$= \begin{bmatrix} -46 & -15 & 46 \\ -76 & -100 & 76 \\ -62 & -56 & 62 \end{bmatrix} \tag{27}$$

- **Question 1.2:** How many operations are required for computing g_1 and g_2 respectively? addition and multiplication times in your result.

- Answer 1.2:
- When we convolute I with f_x, it consists of 45 operations.
- When we convolute the result above with f_x, it consists of 45 operations once again.
- This makes a total of 90 operations to compute g_1.
- g_1:90
- If we compute G_2 by convoluting I and f_{xy}, it involves 17 operations of multiplication and division for one element of the output because it includes the operations for zero. It will be the same value for all the elements and the total number of operations for this is 153. If we include the operations required for fxy,i.e., 45, the total number of operations for g_2 will be 198.
- g_2:198 operations
- **Question 1.3:** What does convolution do to this image?

When convolution is done on the image with fx and fy subsequently,the output image is flipped on its brightness and the bright parts are brighter and the dark parts are darker.

The image was very bright on the first column, normally bright on the second column and dark in the third column. The values of the brightness increased, darkness decreased(darkness intensity increased or light intensity decreased) and it was flipped horizontally. The output image has its left side much darker compared to the right side of the input image and the right side much brighter than the left side of the input image. The middle column of the output image is just brighter than the normal brightness in the middle column of the image.

We can also see that the output image is symmetric from left to right between the first and third column and from top to bottom in the second column.

2 Kernel Estimation (2 points)

Recall the special case of convolution discussed in class: The Impulse function. Using an impulse function, it is possible to 'shift' (and sometimes also 'scale') an image in a particular direction.

For example, when the following image

$$I = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \tag{28}$$

is convolved with the kernel,

$$f = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{29}$$

it results in the output:

$$g = \begin{bmatrix} e & f & 0 \\ h & i & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{30}$$

Another useful trick to keep in mind is the decomposition of a convolution kernel into scaled impulse kernels. For example, a kernel

$$f = \begin{bmatrix} 0 & 0 & 7 \\ 0 & 0 & 0 \\ 0 & 4 & 0 \end{bmatrix} \tag{31}$$

can be decomposed into

$$f_1 = 7 * \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } f_2 = 4 * \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

- **Question:** Using the two tricks listed above, estimate the kernel f **by hand** which when convolved with an image

$$I = \begin{bmatrix} 1 & 5 & 2 \\ 7 & 8 & 6 \\ 3 & 9 & 4 \end{bmatrix} \tag{32}$$

results in the output image

$$g = \begin{bmatrix} 29 & 43 & 10 \\ 62 & 52 & 30 \\ 15 & 45 & 20 \end{bmatrix} \tag{33}$$

Hint: Look at the relationship between corresponding elements in g and I.

Answer:-

I have assumed the kernel matrix after flipping to be $\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$

When I did the convolution for the image matrix with the assumed kernel, I got a system of equations,

$$5f + 8i + 7h + e = 29 \tag{34}$$

$$d + 7g + 8h + 6i + 2f + 5e = 43 \tag{35}$$

$$5d + 8g + 6h + 2e = 10 \tag{36}$$

$$b + 5c + 8f + 9i + 3h + 7e = 62 \tag{37}$$

$$a + 5b + 2c + 7d + 8e + 6f + 3g + 9h + 4i = 52 \tag{38}$$

$$2b + 5a + 8d + 9g + 4h + 6e = 30 \tag{39}$$

$$7b + 8c + 9f + 3e = 15 \tag{40}$$

$$7a + 8b + 6c + 3d + 4f + 9e = 45 \tag{41}$$

$$8a+6b+9d+4e=20 \tag{42}$$

Upon solving this system of equations, we can obtain the values of the variables.
e=5,i=3 and the rest were zero.
We flip the answer to get the actual kernel,

$$\begin{bmatrix} 3 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{43}$$

However, I actually got the answer by continuous guessing.

3 Edge Moving (2 points)

Object Recognition is one of the most popular applications in Computer Vision. The goal is to identify the object based on a template or a specific pattern of the object that has been learnt from a training dataset. Suppose we have a standard template for a "barrel" which is a 3×3 rectangle block in a 4×4 image. We also have an input 4×4 query image. Now, your task is to verify if the image in question contains a barrel. After preprocessing and feature extraction, the query image is simplified as I_Q and the barrel template is I_T.

$$I_Q = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, I_T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

Instinctively, the human eye can automatically detect a potential barrel in the top left corner of the query image but a computer can't do that right away. Basically, if the computer finds that the difference between query image's features and the template's features are minute, it will prompt with high confidence: 'Aha! I have found a barrel in the image'. However, in our circumstance, if we directly compute the pixel wise distance D between I_Q and I_T where

$$D(I_Q, I_T) = \sum_{i,j} (I_Q(i,j) - I_T(i,j))^2 \tag{44}$$

we get $D = 10$ which implies that there's a huge difference between the query image and our template. To fix this problem, we can utilize the power of the convolution. Let's define the 'mean shape' image I_M which is the blurred version of I_Q and I_T.

$$I_M = \begin{bmatrix} 0.25 & 0.5 & 0.5 & 0.25 \\ 0.5 & 1 & 1 & 0.5 \\ 0.5 & 1 & 1 & 0.5 \\ 0.25 & 0.5 & 0.5 & 0.25 \end{bmatrix}$$

- **Question 3.1:** Compute two 3×3 convolution kernels f_1, f_2 **by hand** such that $I_Q \otimes f_1 = I_M$ and $I_T \otimes f_2 = I_M$ where $\otimes$ denotes the convolution operation. (Assume zero-padding)
- **Question 3.2:** For a convolution kernel $f = (f_1 + f_2)/2$, we define $I'_Q = I_Q \otimes f$ and $I'_T = I_T \otimes f$. Compute I'_Q, I'_T and $D(I'_Q, I'_T)$ **by hand**. Compare it with $D(I_Q, I_T)$ and briefly explain what you find.

Answer 3.1:

Upon assuming a 3x3 kernel with variables a to i as f1 and guessing the values appropriately by convoluting with Iq and having Im as the output image, the variables are,

$$a = 0.25 \tag{45}$$
$$b = 0.25 \tag{46}$$
$$c = 0 \tag{47}$$
$$d = 0.25 \tag{48}$$
$$e = 0.25 \tag{49}$$
$$f = 0 \tag{50}$$
$$g = 0 \tag{51}$$
$$h = 0 \tag{52}$$
$$i = 0 \tag{53}$$

The f1 convolution kernel is obtained after flipping,

$$f1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.25 & 0.25 \\ 0 & 0.25 & 0.25 \end{bmatrix} \tag{54}$$

Keeping symmetry in mind, I tried the following 3x3 convolution kernel for f2,

$$f2 = \begin{bmatrix} 0.25 & 0.25 & 0 \\ 0.25 & 0.25 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{55}$$

I found that it does work and this is the solution for f2.
Answer 3.2:

$$f = (f1 + f2)/2 = \begin{bmatrix} 0.125 & 0.125 & 0 \\ 0.125 & 0.125 & 0.125 \\ 0 & 0.125 & 0.125 \end{bmatrix} \tag{56}$$

$$Iq' = Iq \otimes f = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 0.125 & 0.125 & 0 \\ 0.125 & 0.125 & 0.125 \\ 0 & 0.125 & 0.125 \end{bmatrix} \tag{57}$$

$$= \begin{bmatrix} 0.125 + 0.125 + 0.125 + 0.125 & 0.625 & 0.375 & 0.125 \\ 0.625 & 0.875 & 0.625 & 0.25 \\ 0.375 & 0.625 & 0.5 & 0.25 \\ 0.125 + 0 + 0 + 0 & 0.25 & 0.25 & 0.125 \end{bmatrix} \tag{58}$$

$$= \begin{bmatrix} 0.5 & 0.625 & 0.375 & 0.125 \\ 0.625 & 0.875 & 0.625 & 0.25 \\ 0.375 & 0.625 & 0.5 & 0.25 \\ 0.125 & 0.25 & 0.25 & 0.125 \end{bmatrix} \tag{59}$$

$$It' = It \otimes f = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 0.125 & 0.125 & 0 \\ 0.125 & 0.125 & 0.125 \\ 0 & 0.125 & 0.125 \end{bmatrix} \tag{60}$$

$$= \begin{bmatrix} 0 + 0 + 0 + 0.125 & 0.25 & 0.25 & 0.125 \\ 0.25 & 0.5 & 0.625 & 0.375 \\ 0 + 0 + 0 + 0 + 0.125 + 0.125 & 0.625 & 0.875 & 0.625 \\ 0.125 & 0.375 & 0.625 & 0.5 \end{bmatrix} \tag{61}$$

$$= \begin{bmatrix} 0.125 & 0.25 & 0.25 & 0.125 \\ 0.25 & 0.5 & 0.625 & 0.375 \\ 0.25 & 0.625 & 0.875 & 0.625 \\ 0.125 & 0.375 & 0.625 & 0.5 \end{bmatrix} \tag{62}$$

$$D(Iq', It') = \sum_{i,j} (I_Q(i,j) - I_T(i,j))^2 \tag{63}$$

$$= 1.1875 \tag{64}$$

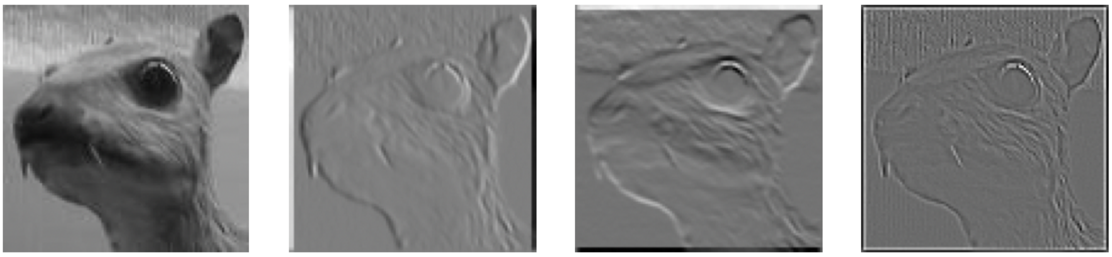
From the calculations I did above, I could find that the difference between the query image and the template

image has been drastically reduced from 10 to around 1.
I basically took a blurred image of Iq and It combined and had that as an output image for Iq $\otimes f1 and It \otimes f2$.

I calculated f1 and f2, found its average,i.e.,(f1+f2)/2 and assigned it to "f". I found the output image-Iq' and It' by convoluting f with Iq and It respectively.
Now, I found D(Iq',It') and compared it with D(Iq,It) obtained earlier and found the former was way smaller than the latter. This happens because the kernel chosen to obtain Iq' and It' was obtained by taking the average of the kernels that would form the output Im, when convoluted with Iq and It. Im is the blurred version of Iq and It combined.
We can notice that Iq and It has pixels only in the top left or the bottom right corner with every other pixel as zero but Iq' and It' has all the pixels filled. We can see with a cursory glance that the difference between the pixels is much smaller.

4 Match the Kernels (2 points)

- **Question 4.1 Match the corresponding kernels for the output images.**



Input Image (a) (b) (c)

1	2	1
0	0	0
-1	-2	-1

(6)

1	0	-1
2	0	-2
1	0	-1

(2)

-1	-1	-1
-1	8	-1
-1	-1	-1

(3)

- **Question 4.2 Match the corresponding kernels for the output images.**

Input Image (d) (e) (f) (g)

(1/256 *

1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1

(5)

0	0	0
0	1	0
0	0	0

(7)

0	-1	0
-1	5	-1
0	-1	0

(4)

(1/16) *

1	2	1
2	4	2
1	2	1

(1)

Answer 4.1:
2-a
6-b
3-c

Answer 4.2:
1-f
5-e

4-d
7-g

5 Boundary Conditions (2 points)

For this problem, we will use the following 3×3 image:

$$I = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 3 & 2 \\ 1 & 1 & 1 \end{bmatrix} \tag{65}$$

You are given 2-D convolution filter:

$$f = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{66}$$

Let $g = I \underbrace{\otimes f \cdots \otimes f}_{\text{infinity times}}$. The output image g has the same size as I.

- **Question 5.1:** When zero-padding is used, what's the output image g. (Give the verification process)

Answer 5.1:
Since,

$$f = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{67}$$

When we convolute I $\otimes f, each pixel decreases.$

$$I(convolution)f = g_1 = \begin{bmatrix} 0.66 & 1.22 & 0.77 \\ 0.88 & 1.33 & 1 \\ 0.66 & 1 & 0.77 \end{bmatrix} \tag{68}$$

$$g_1(convolution)f = g_2 = \begin{bmatrix} 0.46 & 0.65 & 0.48 \\ 0.64 & 0.93 & 0.68 \\ 0.43 & 0.63 & 0.46 \end{bmatrix} \tag{69}$$

$$g_2(convolution)f = g_3 = \begin{bmatrix} 0.30 & 0.43 & 0.30 \\ 0.42 & 0.60 & 0.42 \\ 0.29 & 0.42 & 0.30 \end{bmatrix} \tag{70}$$

A simple verification process would be to focus on the element at the centre at (2,2). Since the kernel is 1/9* a matrix with all elements as one, the output image would result in the pixel at (2,2) to have the maximum value as it adds all the elements in the image and divides it by 9. The sum of all the elements in the first output image when we convolute I with f for the first time, it would result in the 12/9 which is more than 1.
However, if we convolute the result again with f, the value at the centre is 8.33 which will be divided by 9 to give a value smaller than 1. The values of the other pixels will also be less than this and subsequently less than 1 as well. If we convolute this output again with f, we add all the nine elements with values less than 1, so its output will definitely be less than 9. So, if we divide by 9, we get a value much smaller.

$$I(convolution)f = g_1 = 1/9 * \begin{bmatrix} 6 & 11 & 7 \\ 8 & 12 & 9 \\ 6 & 9 & 7 \end{bmatrix} \tag{71}$$

Centre value = 1.33

$$g_1(convolution)f = g_2 = (1/(9*9)) * \begin{bmatrix} 37 & 53 & 39 \\ 52 & 75 & 55 \\ 35 & 51 & 37 \end{bmatrix} \tag{72}$$

Centre value = 0.93

$$g_2(convolution)f = g_3 = (1/(9*9*9)) * \begin{bmatrix} 217 & 311 & 222 \\ 303 & 434 & 310 \\ 213 & 305 & 218 \end{bmatrix} \tag{73}$$

Centre value = 0.6
If we just take the sum of the values of the elements in g3, we get 3.48.
If we divide that by 9, we get 0.38 which is the central value of the new output image. The other elements are smaller than that. If we assume that all the elements are equal to that, we can just multiply it by 9 to get the sum of the elements. And then we divide it again by 9 and we get the same number.

So, if we take the values of the other elements less than 0.38 and add all the elements and divide it by 9, the value of the central element/pixel will be less than 0.38.

As we keep doing it till infinity, the pixels keep decreasing and decreasing and tends to zero.

The output image will be,

$$g = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{74}$$