CS 6476 Project 1

[Anh Dang]
[anh.dang@gatech.edu]
[adang40]
[903635962]

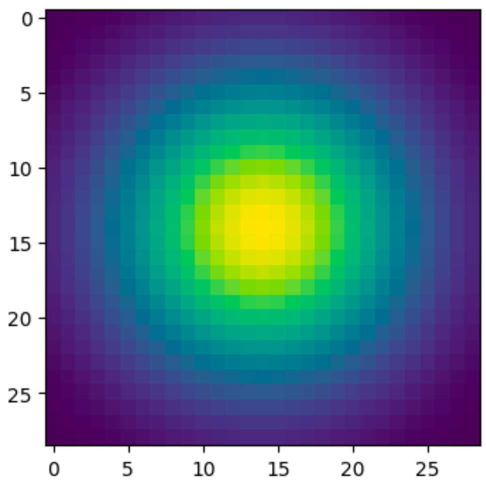
(1 pts) Part 1: Gaussian Kernels

Separated kernels can speed up the filtering operation because a 2D kernel can be decomposed into two 1D kernels, increasing the efficiency of its computation. Because the dimensions are being broken down into simpler operations.

(1 pts) Part 1: Image filtering

1D:

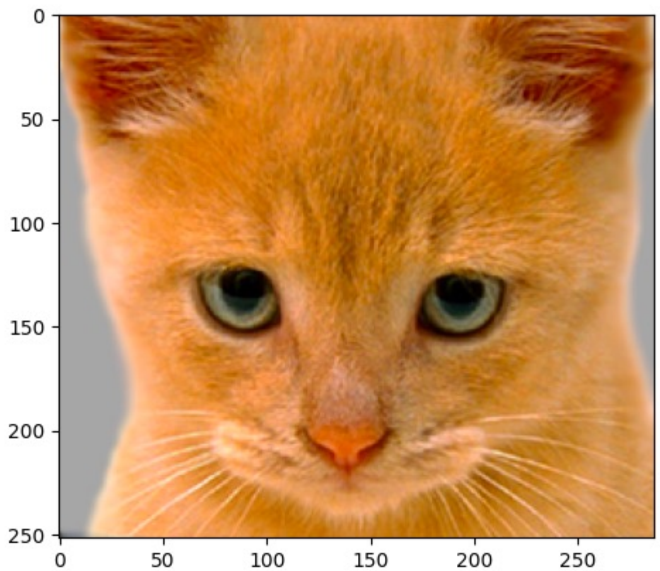
2D:



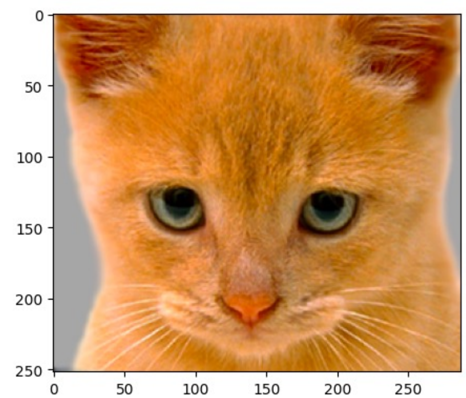
First, the shape of the image and filter are extracted. The image is padded with zero-padding to preserve the dimensions of the convolution by finding the center pixel. The image is symmetrically padded so the filter can be applied throughout the entire image. Then, the padding is applied to an empty image, which is then iterated through to apply the calculated padding. During each iteration in the X and Y dimension, element-wise multiplication is performed to the filtered region. This allows the calculation of the filter to be assigned to each position, which is done in an empty channel. Then, the resulting image is trimmed to de-pad the channel and added to the empty image.

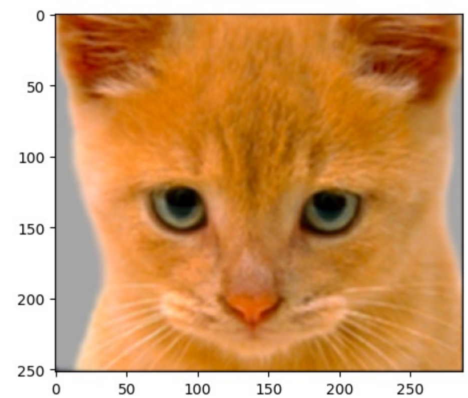
(1 pts) Part 1: Image filtering

Identity filter



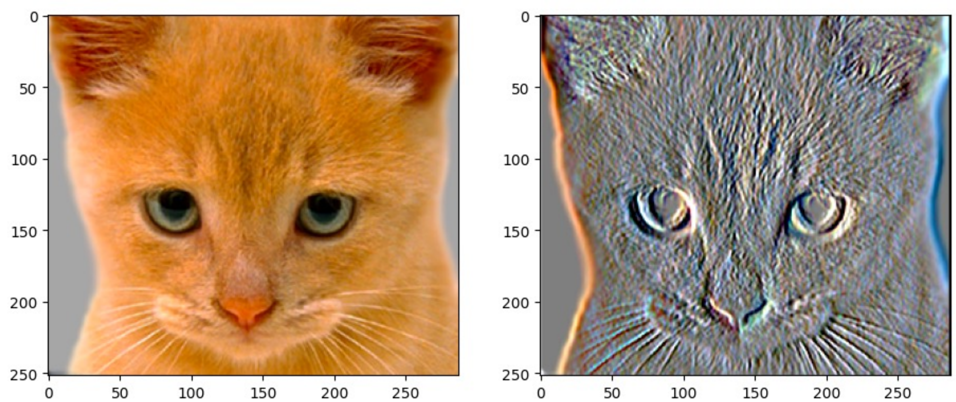
Small blur with a box filter



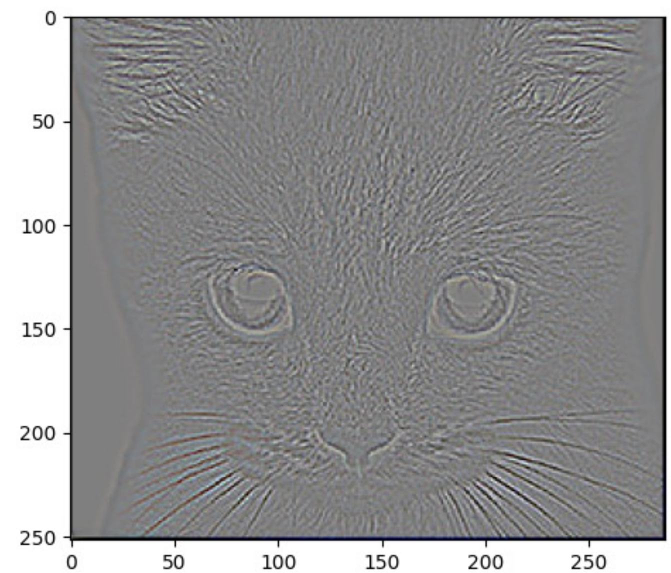


(1 pts) Part 1: Image filtering

Sobel filter



Discrete Laplacian filter



(1.5 pts) Part 1: Hybrid images

First, the first image is applied a low-pass filter using the `my_conv2d_numpy()` function, which results in the low frequency content of the first image. Then, the high frequency content for the second image is extracted by subtracting the low frequency content from itself. Then, the hybrid image is formed by adding the frequency content from the images to each other, and the operation is clipped using `np.clip()` to ensure the output values are within the appropriate range.

Cat + Dog

Cutoff frequency: 7

(1.5 pts) Part 1: Hybrid images

Motorcycle + Bicycle

Cutoff frequency: 4

Plane + Bird

Cutoff frequency: 3

(1.5 pts) Part 1: Hybrid images

Einstein + Marilyn

Cutoff frequency: 2

Submarine + Fish

Cutoff frequency: 4

(1.5 pts) Part 2: Hybrid images with PyTorch

Cat + Dog

Motorcycle + Bicycle

(1.5 pts) Part 2: Hybrid images with PyTorch

Plane + Bird

Einstein + Marilyn

(1.5 pts) Part 2: Hybrid images with PyTorch

Submarine + Fish

The runtime pf Part 1 is 2.429s, and the runtime of Part 2 is 0.355s. The runtime of Part 2 is approximately 85% faster than the runtime of Part 1.

(2 pts) Part 3: Understanding input/output shapes in PyTorch

[Consider a 1-channel 5x5 image and a 3x3 filter. What are the output dimensions of a convolution with the following parameters?

Stride = 1, padding = 0	**3x3**
Stride = 2, padding = 0	**2x2**
Stride = 1, padding = 1	**5x5**
Stride = 2, padding = 1	**3x3**

[What are the input & output dimensions of the convolutions of the dog image and a 3x3 filter with the following parameters:

Stride = 1, padding = 0	**359x408**
Stride = 2, padding = 0	**180x204**
Stride = 1, padding = 1	**361x410**
Stride = 2, padding = 1	**181x205**

(1 pts) Part 3: Understanding input/output shapes in PyTorch

$$\mathbf{M} = \begin{array}{|c|c|c|}\hline 4 & 4 & 7 \\ \hline 6 & 4 & 7 \\ \hline 6 & 4 & 4 \\ \hline \end{array}$$

$$\mathbf{K} = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline -1 & 1 \\ \hline \end{array}$$

As mentioned in the instruction, use the kernel K to filter through the image M with stride 1 and padding 1. For padding, simply append 0s to each side of the image. You can insert tables here as matrices to show the process

$$M_{padded} = \begin{array}{|c|c|c|c|c|}\hline 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 4 & 4 & 7 & 0 \\ \hline 0 & 6 & 4 & 7 & 0 \\ \hline 0 & 6 & 4 & 4 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline \end{array} \ \text{x} \ \begin{array}{|c|c|}\hline 1 & 2 \\ \hline -1 & 1 \\ \hline \end{array} \ \text{->} \ \begin{array}{|c|c|c|c|}\hline 4 & 12 & 15 & 14 \\ \hline 2 & 16 & 12 & 21 \\ \hline 0 & 18 & 9 & 15 \\ \hline -6 & 2 & 0 & 4 \\ \hline \end{array}$$

(1 pts) Part 3: Understanding input/output shapes in PyTorch

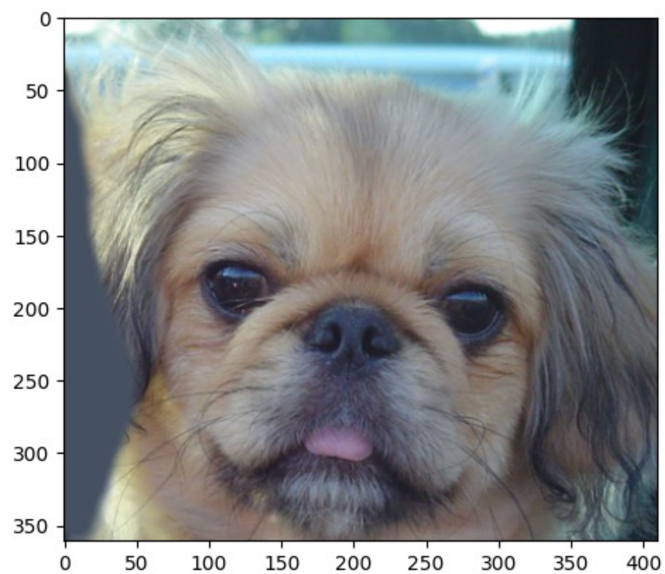
[How many filters did we apply to the dog image?]

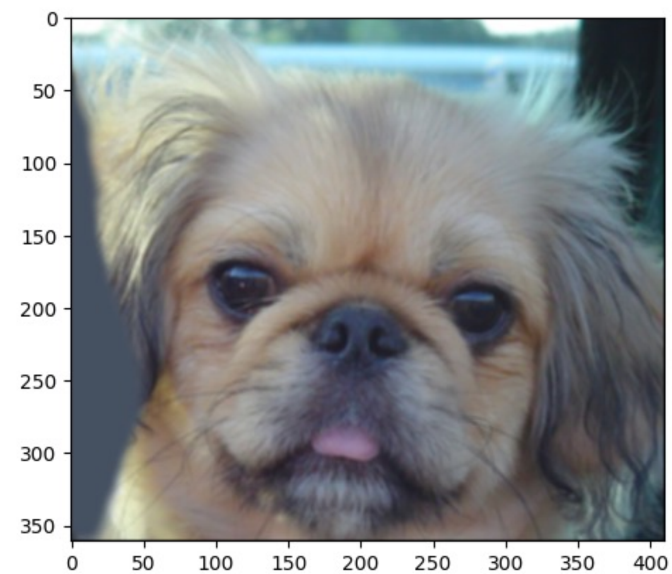
12 filters were applied to the dog image.

[Section 3 of the handout gives equations to calculate output dimensions given filter size, stride, and padding. What is the intuition behind this equation?]

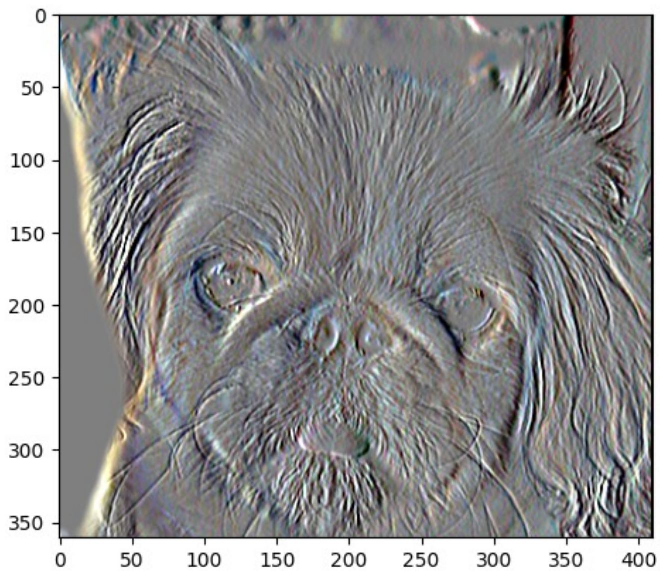
The intuition behind the equation is that the first half of the equation (`h1 or w1 – k + 2 * padding`) is to fit the filter size to the image size, which is done through subtraction. Padding is them added to each size of the image, which is multiplied by 2 to achieve both the top and bottom/left and right side of the image. Everything is then divided by the stride as pixels are lost if the filter by a stride of more than 1 pixel is moved.

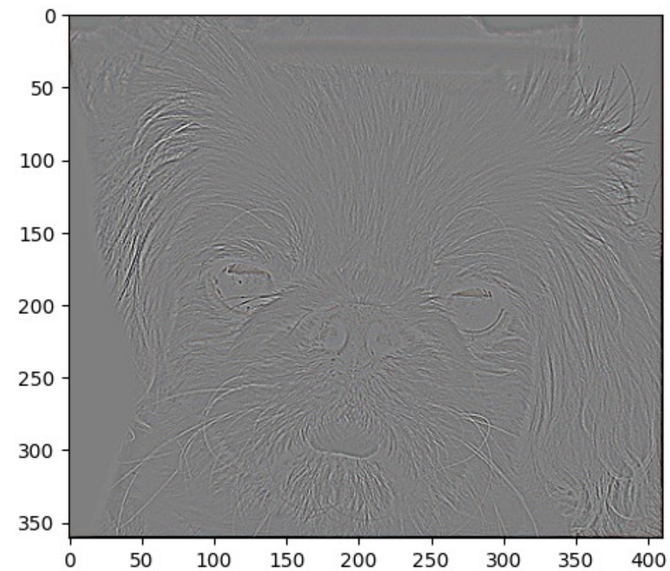
(1 pts) Part 3: Understanding input/output shapes in PyTorch





(1 pts) Part 3: Understanding input/output shapes in PyTorch





(1 pts) Conclusion

[How does varying the cutoff frequency value or swapping images within a pair influences the resulting hybrid image?]

As the cutoff frequency decreases, more of the low-frequency content from Image A will be captured in the hybrid image. This is an inversely proportionally relationship as a higher cutoff frequency will resume in more high-frequency content from Image B.

Swapping images will cause the low frequencies in Image B and high frequencies in Image A to be captured.