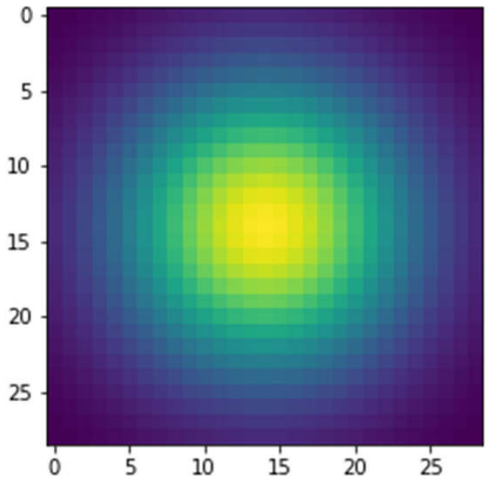
CS 4476 Project 1

Kok Jian Yu
jkok7@gatech.edu
jkok7
903550380

Part 1: Image filtering



Using the cutoff_frequency, calculate k, mean and std.

Calculate range of x, which is from mean – k/2 to mean + k/2.

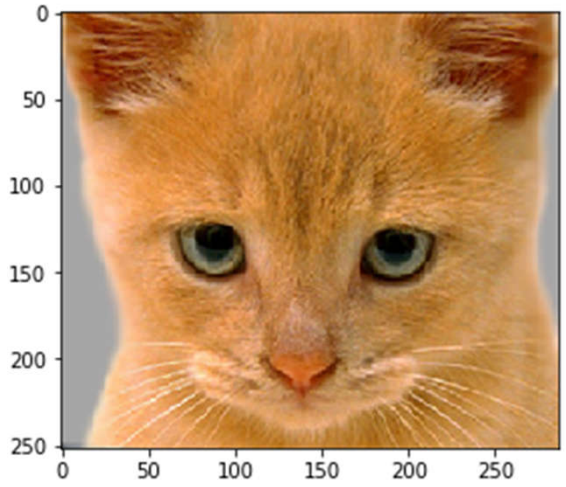
Create 1D gaussian kernel using the formula.

Do an outer product of 1D gaussian kernel with 1D gaussian kernel to get a 2D gaussian kernel.

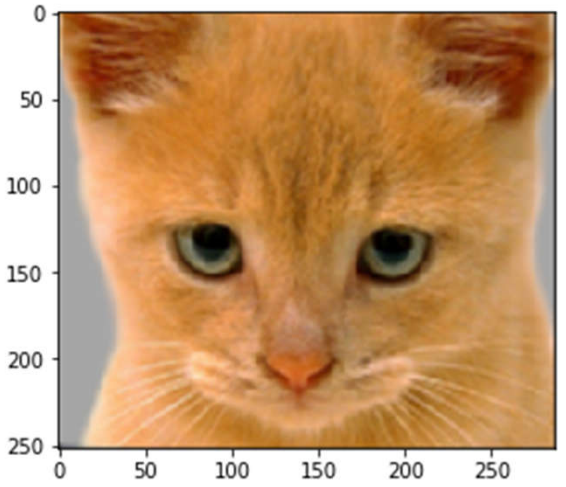
Normalize the 2D gaussian kernel to sum up to 1.

Part 1: Image filtering

Identity filter

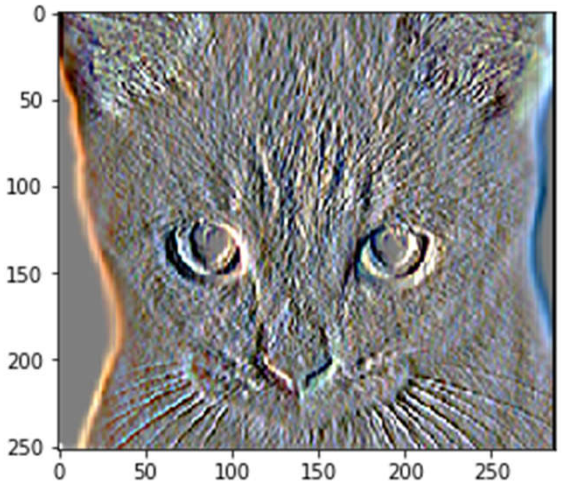


Small blur with a box filter

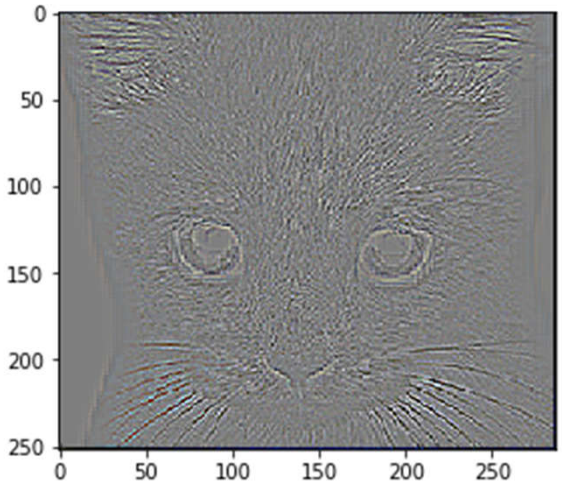


Part 1: Image filtering

Sobel filter



Discrete Laplacian filter



Part 1: Hybrid images

Calculate low frequency for image1 by calling my_imfilter method with image1 and the filter.

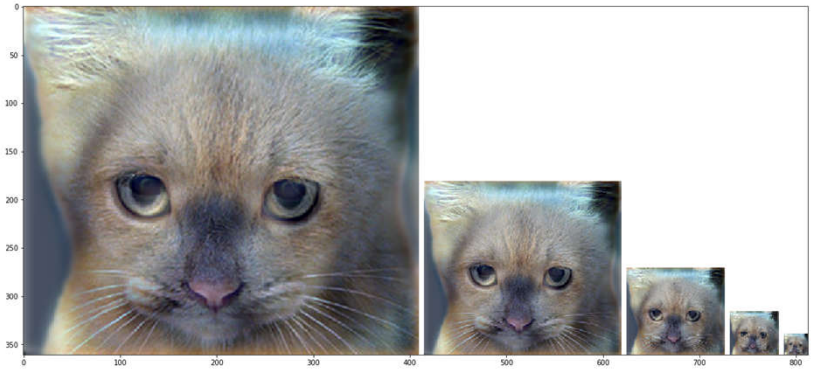
Calculate low frequency for image2 by calling my_imfilter method with image2 and the filter.

Calculate high frequency for image 2 by subtracting each value for image 2, with the low frequency value for image 2.

Calculate the hybrid image by summing up the low frequency for image1, and high frequency for image2, and performing clip to ensure values are from 0 to 1.

Cat + Dog

<insert your hybrid image here>



Cutoff frequency: 7

Part 1: Hybrid images

Motorcycle + Bicycle

Cutoff frequency: 4

Plane + Bird

Cutoff frequency: 9

Part 1: Hybrid images

Einstein + Marilyn

nage here>

Cutoff frequency: 5

Submarine + Fish

re>

Cutoff frequency: 4

Part 2: Hybrid images with PyTorch

Cat + Dog

Motorcycle + Bicycle

Part 2: Hybrid images with PyTorch

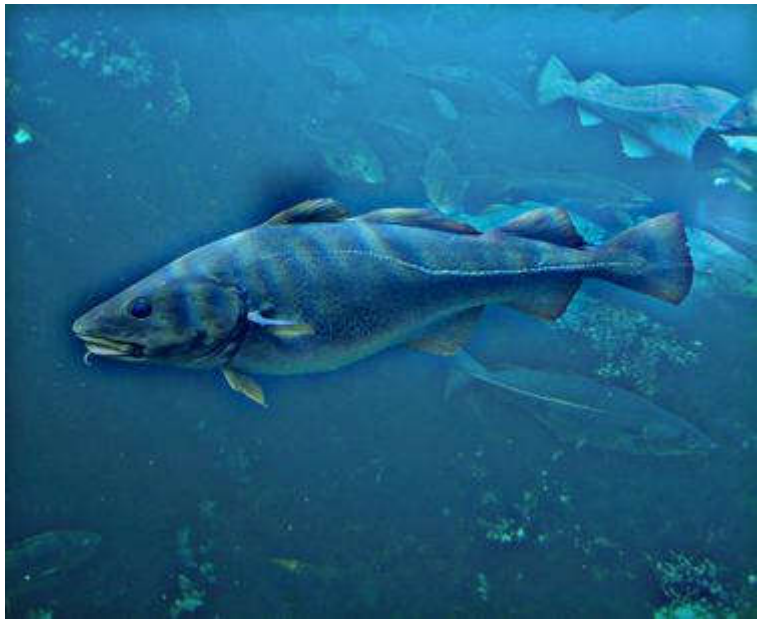
Plane + Bird

Einstein + Marilyn

<insert your hybrid image here>

Part 2: Hybrid images with PyTorch

Submarine + Fish



Part 1 vs. Part 2

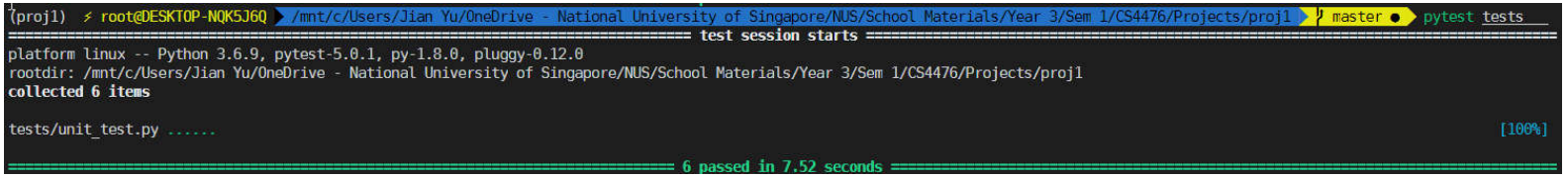
<Compare the run-times of Parts 1 and 2 here, as calculated in proj1.ipynb. What can you say about the two methods?>

Part 1: 6.090s
Part 2: 3.544s

From this, we can say that pytorch is more optimized for such operations compared to simply using numpy.

Tests

<Provide a screenshot of the results when you run `pytest tests` on your final code implementation (note: we will re-run these tests).>



Conclusions

I have learned that for hybrid image, the images to be merged, and the cutoff frequency must be chosen carefully for the results to look decent. For example, for the cat and dog example, simply swapping the image (cat to low freq, dog to high freq) causes the image to look extremely weird.

It is also hard to get some hybrid images to be as good as the cat and dog hybrid image despite tweaking the cutoff frequency due to the background differences.