Name (netid): Molly Yang (tyy2)
CS 445 - Project 1: Hybrid Images

Complete the claimed points and sections below.

Total Points Claimed	**[130] / 130**

1. Hybrid image main result
    a. Main result and description [45] / 45
    b. FFT images of main result [15] / 15
2. Hybrid images: two additional results [10] / 10
3. Image enhancement tasks (3rd is B&W)
    a. Contrast enhancement [10] / 10
    b. Color enhancement [10] / 10
    c. Color shift [10] / 10
4. Quality of results / report [10] / 10
5. Color Hybrid Image w/ explanation (B&W) [5] / 5
6. Gaussian / Laplacian Pyramids (B&W) [15] / 15

1. Hybrid image main result

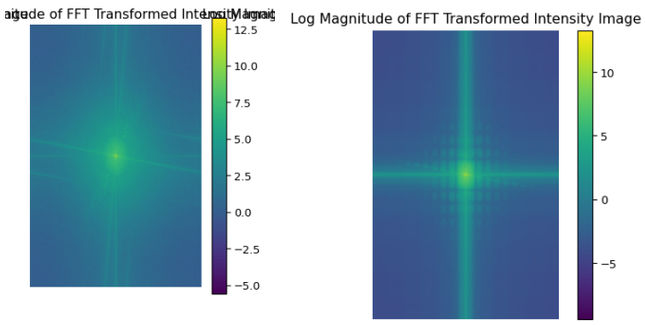
Include

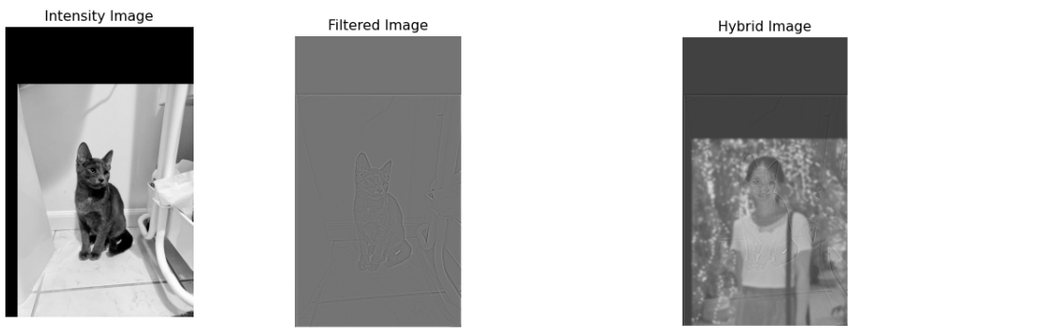
- ☑ ~~Original and filtered input images~~
- ☑ ~~Hybrid image result~~
- ☑ ~~FFT images of each original and filtered image and the hybrid image~~
    - log magnitude of the Fourier transform of the two input images, the filtered images, and the hybrid image
- ☑ ~~Description in a few sentences of how it works using the included images as illustrations. Explain parameter settings and any clever ideas that are incorporated.~~
- ☑ ~~All results must be based on your own images (can be from web with attribution, but not provided samples)~~

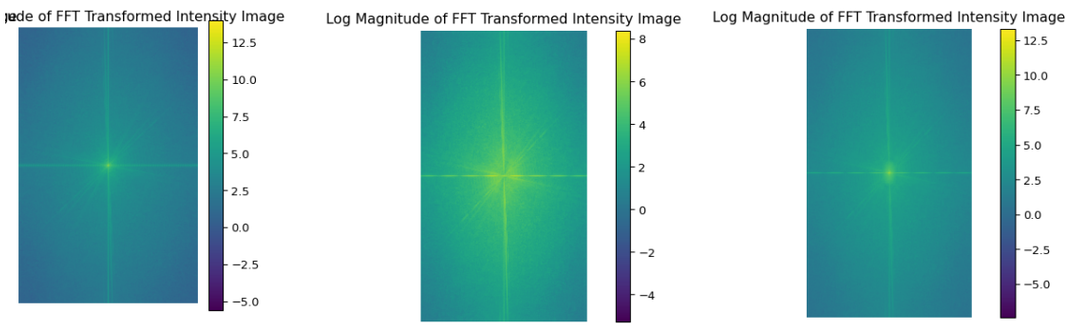
Original images and hybrid results

 +  = 

FFT images







Description

In this hybrid image of me and Misha, I was applied a low-pass (Gaussian) filter and Misha was applied a high-pass (Laplacian) filter computed by subtracting the Gaussian filtered image from the original. By experimenting and observing the hybrid results, the standard deviations or cut-off frequencies used in the two Gaussian filters are the following: sigma_low = 10 and sigma_high = 5. The hybrid results show Misha's face (high frequencies) when close up, and my face (low frequencies) when far away. It was not intentional, but Misha and I were looking in two slightly different directions in the original images, which at the right distance made the hybrid image seem a little like we are looking back and forth.

2. Hybrid image additional results

Include

- ☑ ~~At least two additional results (may not use provided samples). For each, include the input and hybrid image (do not need to show filtered or FFT images)~~

Awake Saana + Asleep Saana = Half Awake Saana

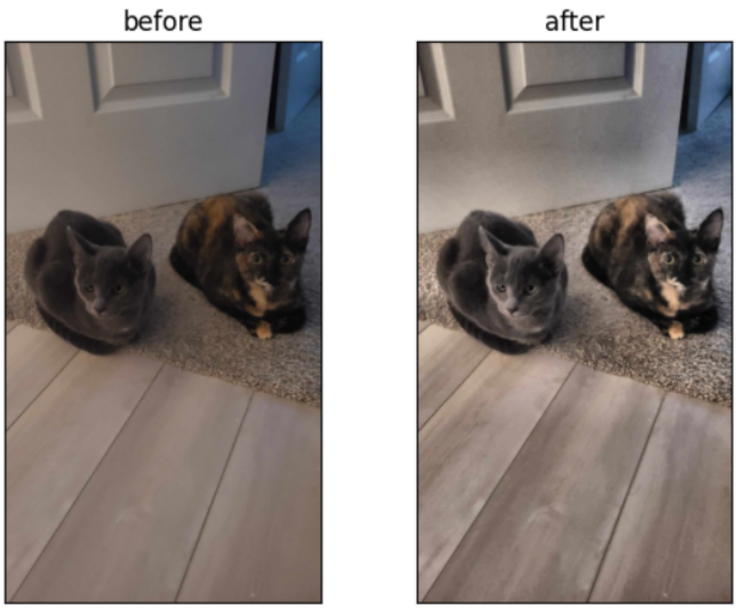
Kyoto Tower + Tokyo Skytree = Kyoto Skytree?

3. Image enhancement tasks (2 required, 3 for B&W)

Include

- ☑ ~~For at least two out of three enhancement tasks (each is worth 10 points), display original image, modified image, and explanation of how the image was modified~~

Contrast Enhancement



To enhance the contrast of the image, contrast limited adaptive histogram equalization (CLAHE) is applied. The image is first converted to LAB. By using the functions cv2.createCLAHE() to create a CLAHE object then clahe.apply() to apply to the image. Adaptive histogram equalization works by dividing the image to a 8x8 tile grid. Each of the tile grids are histogram equalized. The default contrast limit is also applied, where histogram bins with contrast above default threshold (40) are clipped and distributed uniformly before applying histogram equalization. After equalization, artifacts in the tile borders are removed by bilinear interpolation.

Color Enhancement

To enhance the color of the image, gamma correction is applied. In this case, gamma is set to 2.3 by trial and error. A nonlinear int array (look up table) scaled for the gamma value is applied to all channels of the original image by OpenCV function cv2.LUT().

Color Shift

To shift the color of the image to be more red and less yellow, the images are converted to LAB. Then, values of A and B are increased by 1.2 and decreased by 0.8 respectively. By using a table and cv2.LUT() similar to gamma correction used in color enhancement, the image's color is shifted to be more red and less yellow.

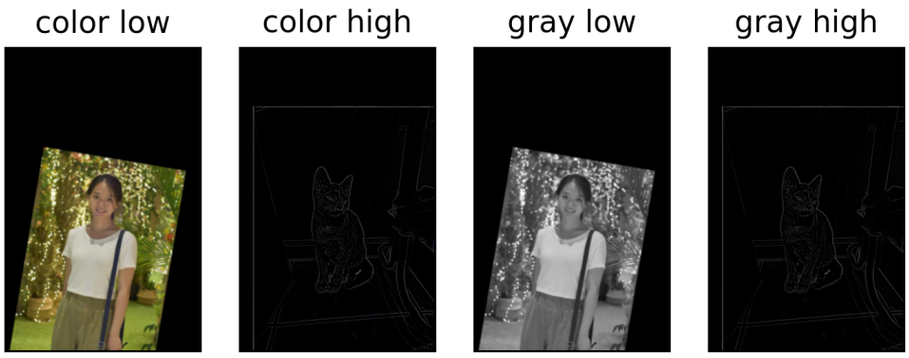
4. Quality of results and report

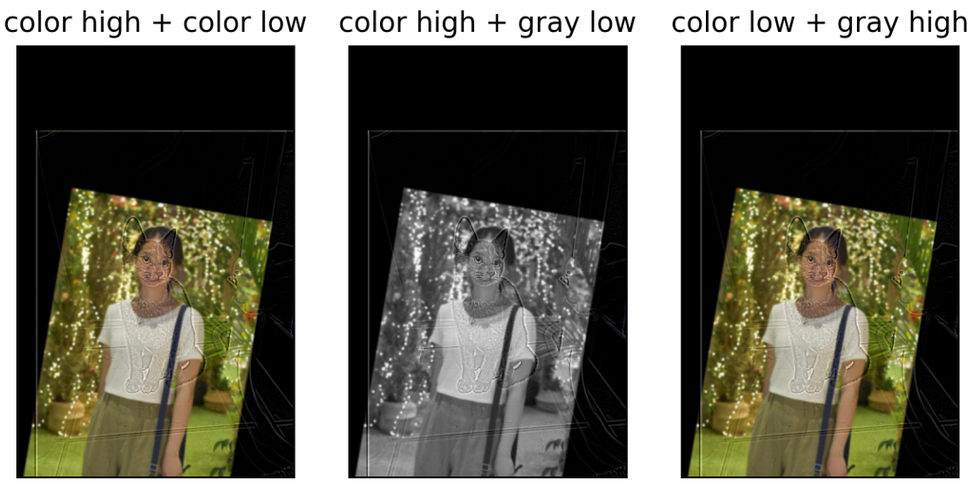
Nothing extra to include.

5. Color hybrid result (B&W)

Include

- ☑ ~~Original images, hybrid image~~
- ☑ ~~Explanation of method: Is it better to use color for the low-pass, the high-pass, or both?~~



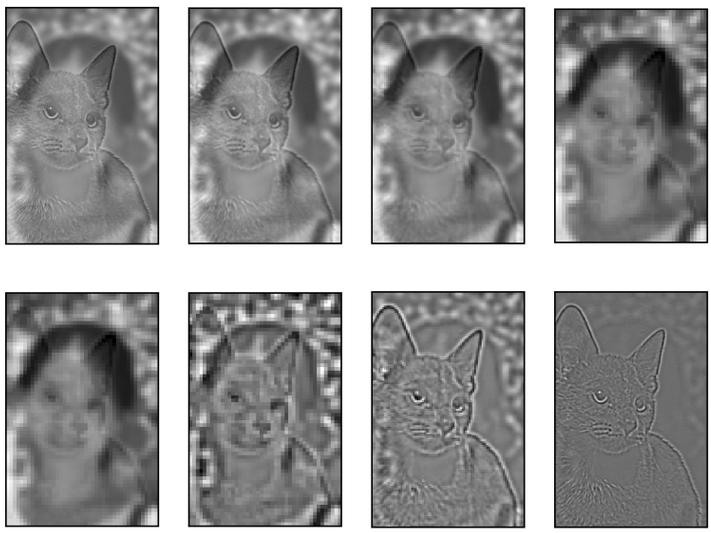


Color does not show up well in high pass filtered images. Using color for low pass and using color for both high and low pass has comparable results. However, this could also be due to Misha being a gray cat.

6. Gaussian and Laplacian Pyramids (B&W)

Include

- ☑ ~~Gaussian pyramid of main hybrid image result (can be one row of images)~~
- ☑ ~~Laplacian pyramid of main hybrid image result (another row of images)~~



Acknowledgments / Attribution

List any sources for code or images from outside sources

OpenCV histogram equalization tutorial
https://docs.opencv.org/4.x/d5/daf/tutorial_py_histogram_equalization.html
Explaining adaptive histogram equalization
https://en.wikipedia.org/wiki/Adaptive_histogram_equalization
StackOverflow article about applying clahe on rgb color images
https://stackoverflow.com/questions/25008458/how-to-apply-clahe-on-rgb-color-images
Gamma correction tutorial
https://pyimagesearch.com/2015/10/05/opencv-gamma-correction/
Week 2 demos
https://www.coursera.org/learn/cs-445/supplement/3Ctfs/week-2-overview
Course website - project 1
https://courses.engr.illinois.edu/cs445/fa2023/projects/hybrid/ComputationalPhotography_ProjectHybrid.html
Project 1 tips
https://docs.google.com/document/d/1x-MPS1bfmSkmacnsX8xIvyN-qJSsxkHWIoYSEERjn_g/edit
Siggraph paper
http://olivalab.mit.edu/publications/OlivaTorralb_Hybrid_Siggraph06.pdf
Image pyramids
https://docs.opencv.org/3.4/d4/d1f/tutorial_pyramids.html
How to find Laplacian pyramids
https://www.tutorialspoint.com/how-to-find-laplassian-pyramids-for-an-image-using-opencv-in-python
Unsupported type
https://stackoverflow.com/questions/55128386/python-opencv-depth-of-image-unsupported-cv-64f