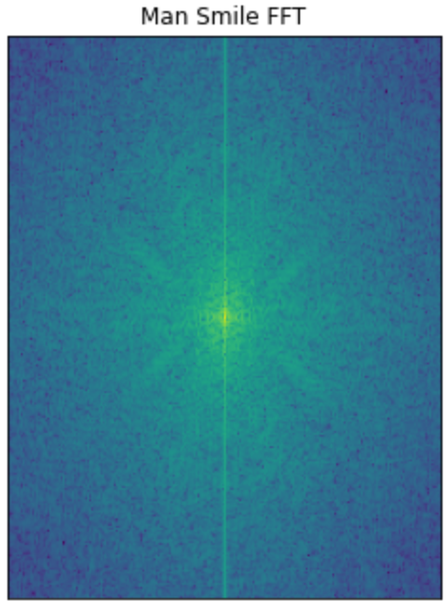
***Joe Morrissey (jrm15)***
**CS 445 - Project 1: Hybrid Images**

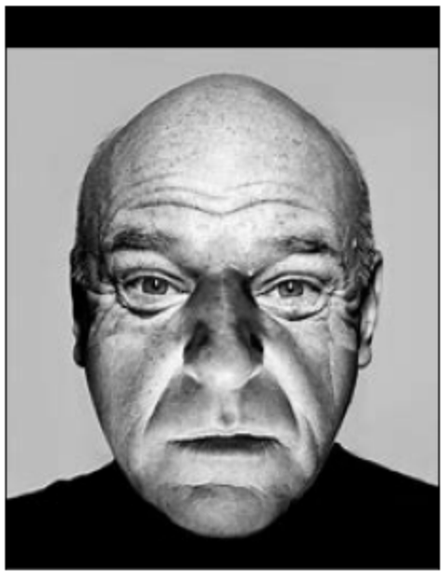
Complete the claimed points and sections below.

| | |
|---|---|
| **Total Points Claimed** | **[138] / 130** |
| 1. Hybrid image main result | |
| a. Main result and description | [44] / 45 |
| b. FFT images of main result | [10] / 15 |
| 2. Hybrid images: two additional results | [9.5] / 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 | [9.5] / 10 |
| 5. Color Hybrid Image w/ explanation (B&W) | [0] / 5 |
| 6. Gaussian / Laplacian Pyramids (B&W) | [0] / 15 |

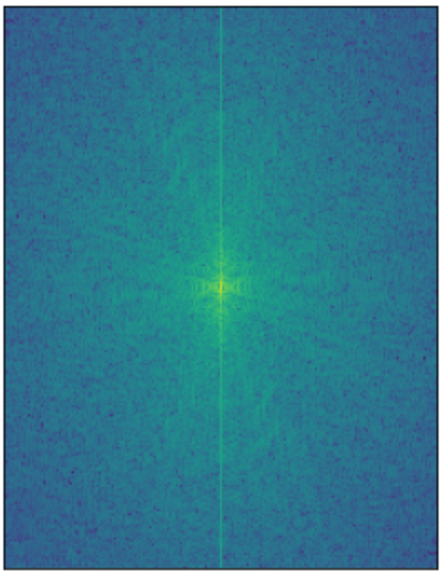
**1. Hybrid image main result**

Man Smirk

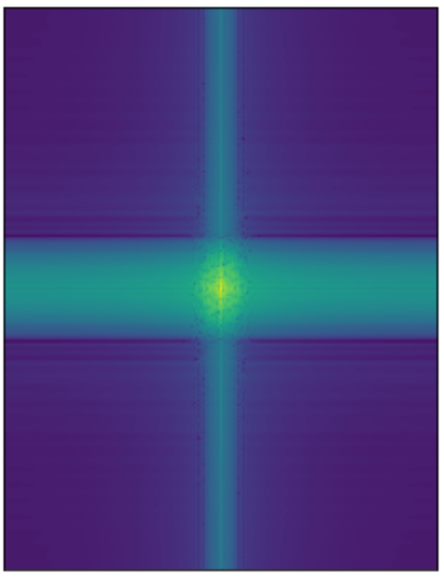


Man Smirk FFT

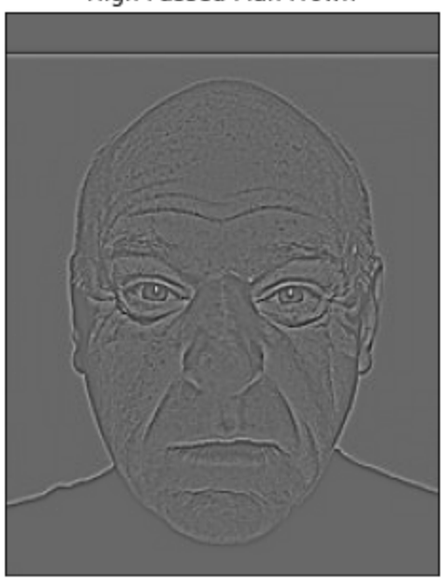


Low Passed Man Smile

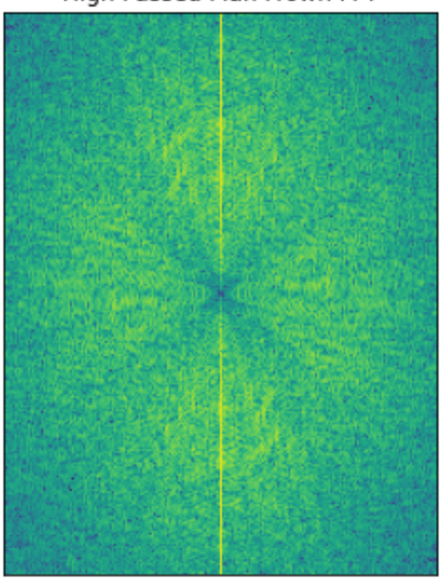
Low Passed Man Smile FFT

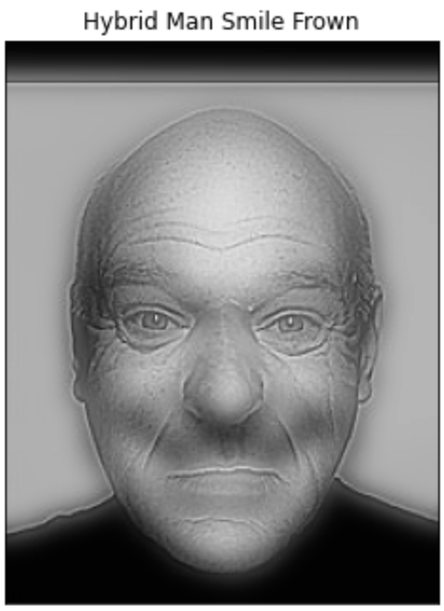


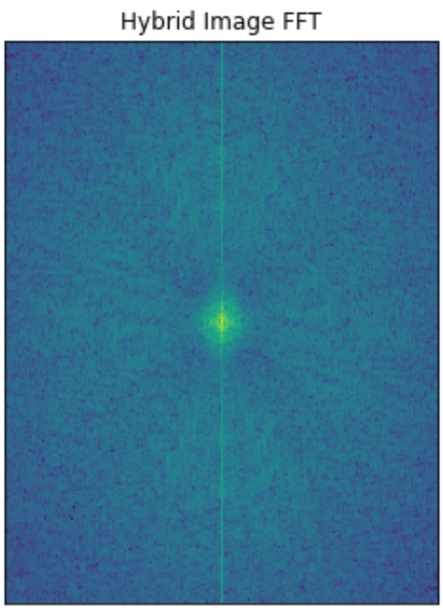
High Passed Man Frown



High Passed Man Frown FFT






*Description*

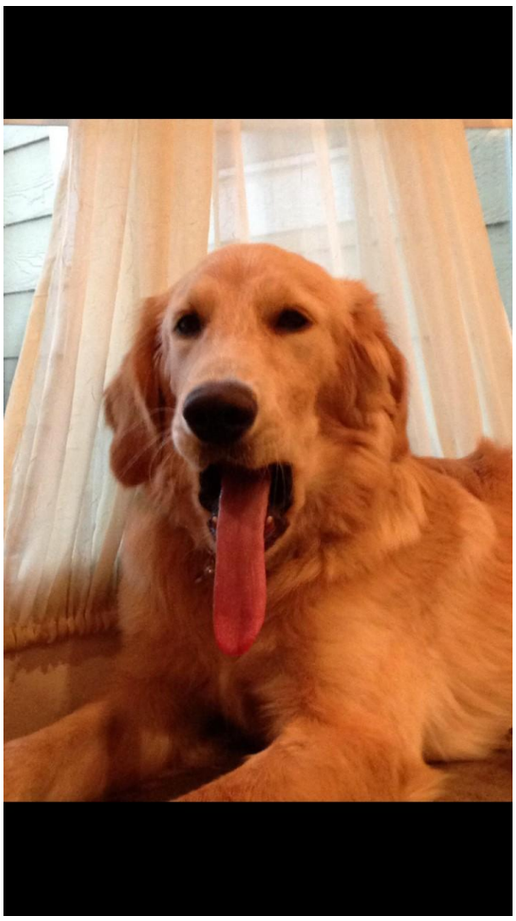
Hybrid Images works by taking two images. It applies a Gaussian, also known as a low pass filter, to the first image and applies a laplacian filter, also known as a high pass filter to the second image. The laplacian filter is obtained by first applying a gaussian filter to an image and then subtracting that filtered image from the original. Once both images are filtered, we can add the images together to obtain the hybrid image. Up close our eyes can make out the lower frequencies better than the high frequencies, so we primarily see the low pass filtered image. However, when we are far away, our eyes can make out higher frequencies better, so we primarily see the high pass filtered image. This phenomenon gives up the hybrid image effect. The hybrid image function also takes in sigma_high and sigma_low, which affect the distribution of the gaussian filter. A higher sigma_low, results in the lower frequency being much more transparent than high frequency. A higher sigma_high, results in the opposite since the laplacian filter is essentially the inverse of the gaussian filter.

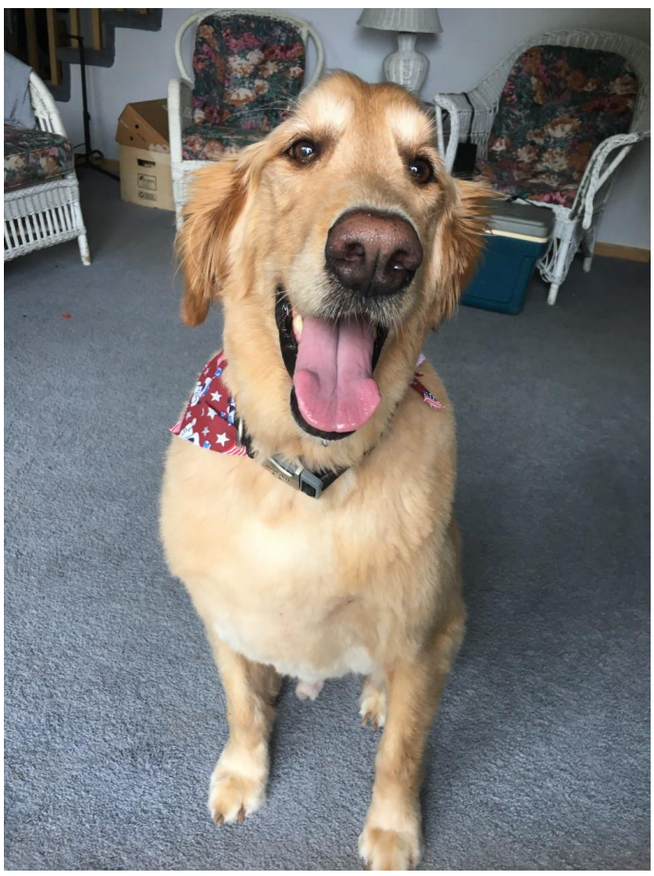
Frown Picture:
Dean Norris' Reaction | Know Your Meme

Smile Picture:
Dean Norris' Reaction | Know Your Meme

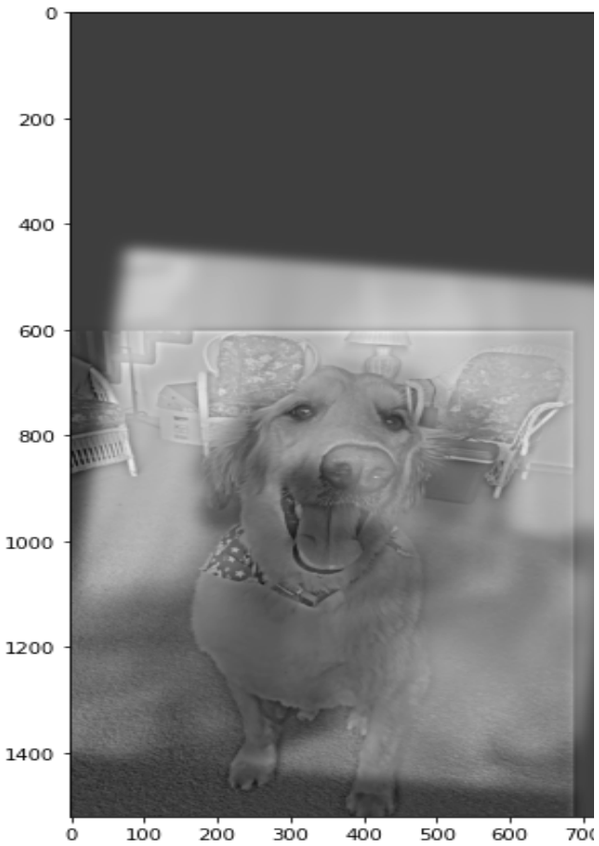
**2. Hybrid image additional results**

*Two input Images:*



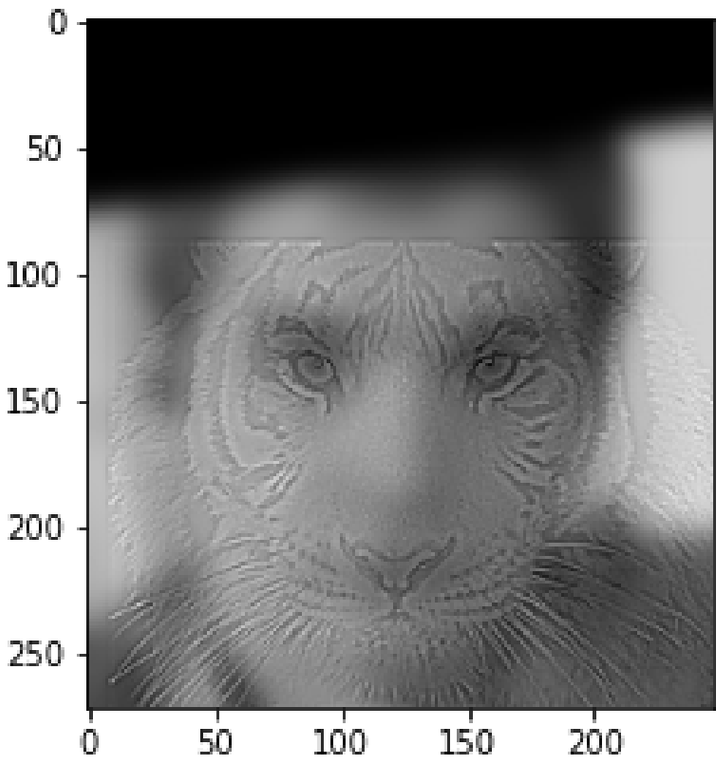


*Hybrid Image:*




*Two Input Images:*

*Hybrid Image:*



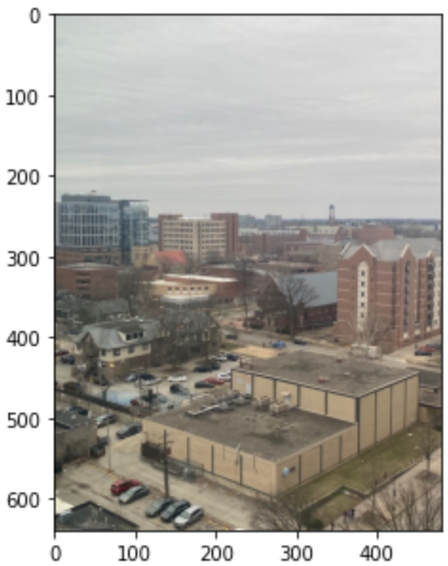

Close up man-
https://www.shutterstock.com/video/clip-5089364-close-up-serious-businessman-looking-camera-getting-down

Close up Tiger-
http://ayay.co.uk/background/animals/tigers/tiger-face-close-up/

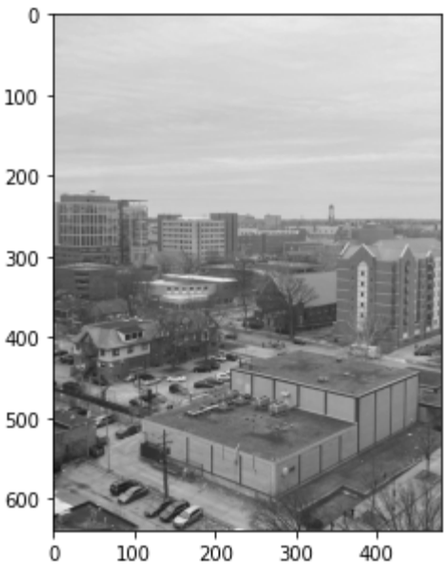
**3. Image enhancement tasks (2 required, 3 for B&W)**

*Contrast Enhancement*

The Original Image




The Original Image in GrayScale



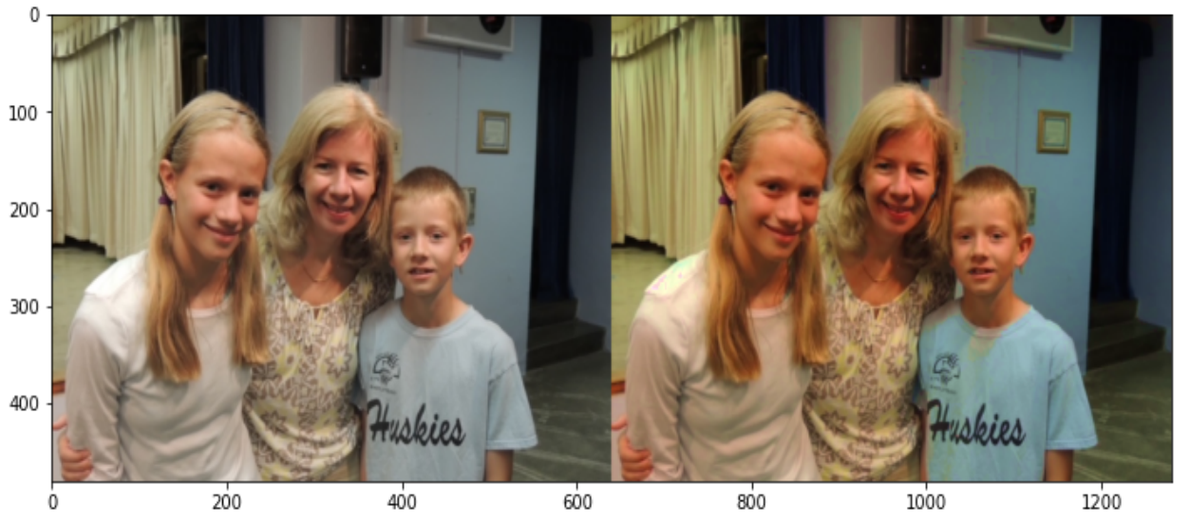


I used histogram equalization to enhance this image. Histogram equalization enhances the higher intensity areas and mutes the lower intensity areas, creating more contrast in the image. I accomplished this by converting to the YCrCb color space, since it separates color from intensity. Since it was a color image,  I then passed only the Y channel into a built-in Histogram Equalization function and finally converted back to the RGB color space.

*Source:* OpenCV Python equalizeHist colored image - Stack Overflow

## *Color Enhancement*

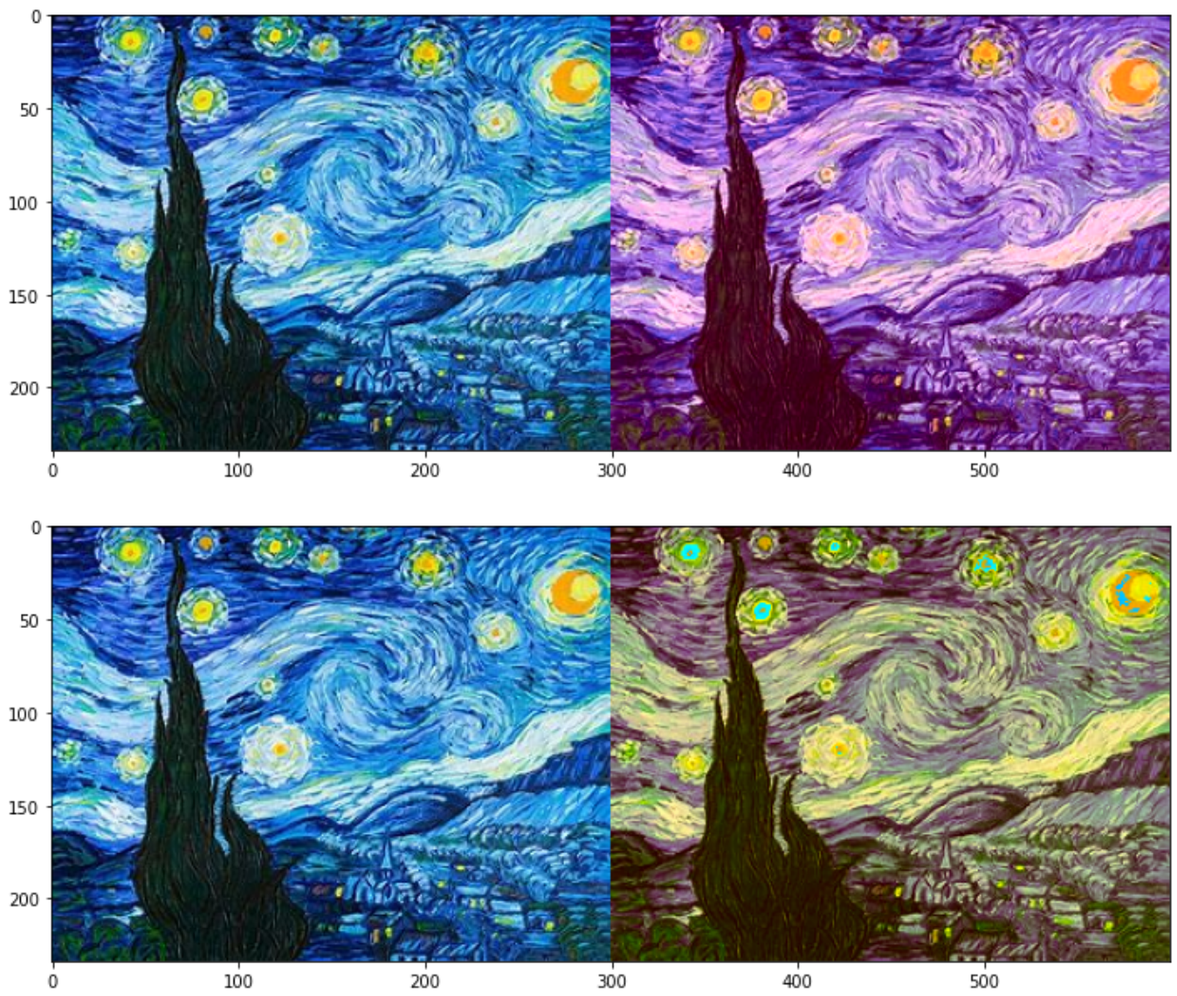
***Original Picture on Left, Enhanced Picture on Right*




To enhance the color without affecting the intensity,  I first converted to the HSV color space. This allowed me to only increase the saturation channel, which enhanced the colors without affecting intensity. Finally, I converted back to the RGB color space.

## *Color Shift*

***Original Picture on Left, Enhanced Picture on Right*



To make the image more red and yellow, I first converted to the LAB color space, since this separates luminance from colors. To make the image more red, I increased the value of the A channel, since higher A values correspond to redder images. To make the image more yellow, I increased the value of the B channel, since higher B values correspond to more yellow images. Since the luminance is its own channel, this method didn't affect the brightness of the image, and could individually change certain colors. Finally, I converted back to the RGB color space.

*Source:* https://i.pinimg.com/originals/42/fc/07/42fc0770ab99e52b41dcac3555f47b8e.jpg

**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?

**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

Close up man:
https://www.shutterstock.com/video/clip-5089364-close-up-serious-businessman-looking-camera-getting-down

Close up tiger: http://ayay.co.uk/background/animals/tigers/tiger-face-close-up/

Starry Night:
https://i.pinimg.com/originals/42/fc/07/42fc0770ab99e52b41dcac3555f47b8e.jpg

Frown Picture:
Dean Norris' Reaction | Know Your Meme

Smile Picture:
Dean Norris' Reaction | Know Your Meme

Passing one channel into cv2.equalizeHist

OpenCV Python equalizeHist colored image - Stack Overflow