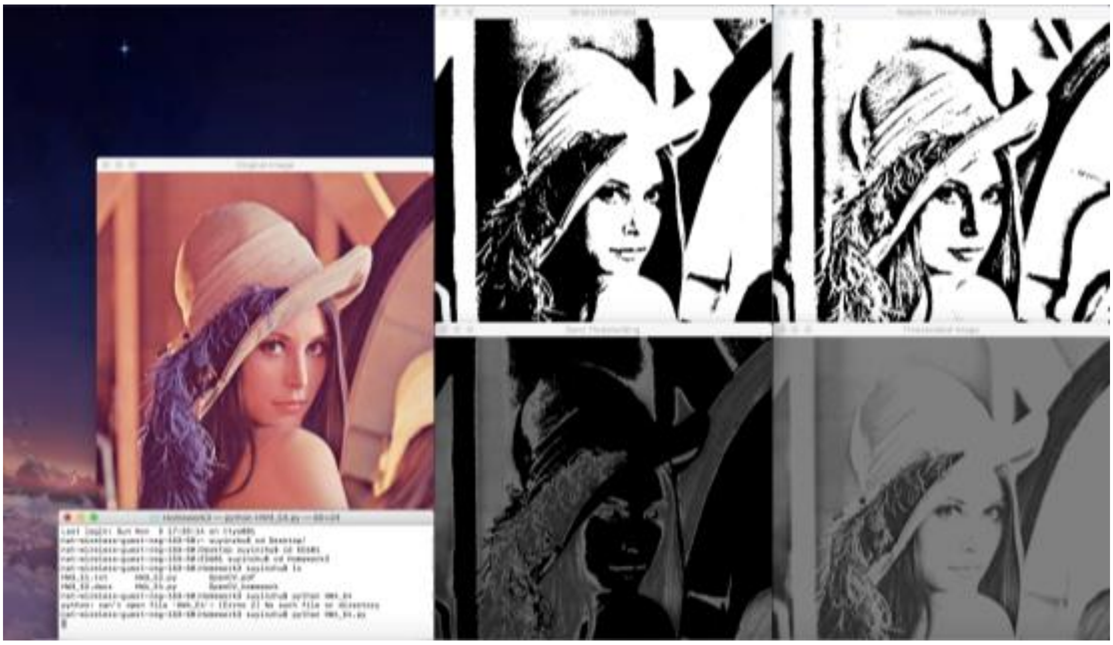
1. Look at Threshold.cpp and implement the code in Python, and observe the results for different threshold values. Comment on the results.



```
int threshold_type = 2;        // slider 1 [0, 4]
int threshold_value = 128;    // slider 2 [0 255]
threshold(gray, dst, threshold_value, 255, threshold_type );
imshow("Thresholded Image", dst );
```

The program firstly set Threshold_Type as 2 and set Threshold_Value as 128. Threshold_Type=2 means the program is using Truncate method to process the image. That is to firstly detect if the pixel value is above the threshold. If the value is smaller than the threshold, the pixel will stay its original value. If the value is bigger than the threshold, the pixel will be changed into another value. In this program, the threshold value is set as 128, and any pixels in the image whose value is bigger than 128 will be changed into 255, which means become white.

```
//Binary Threshold
Mat thresholded;
threshold(gray, thresholded, current_threshold, max_threshold,
THRESH_BINARY);imshow("Binary threshold",thresholded);
```

This part of program is to create a method to change the image into two kind of colors: black and white. Any pixels in the image whose value is bigger than the threshold value will become white, and others will become black. And this is the reason why this method's name includes "binary".

```
// Band thresholding
Mat binary_image1,binary_image2,band_thresholded_image;
int threshold1 = 27, threshold2 = 125;
threshold(gray, binary_image1, threshold1, max_threshold, THRESH_BINARY);
threshold(gray, binary_image2, threshold2, max_threshold,
THRESH_BINARY_INV);
bitwise_and( binary_image1, binary_image2, band_thresholded_image );
imshow("Band Thresholding", band_thresholded_image);
```

This piece of program is to reverse the color in the image. For example, change the white pixel into black and change the black pixel into white.

```
// Semi thresholding
Mat semi_thresholded_image;
threshold(gray, semi_thresholded_image, current_threshold, max_threshold,
THRESH_BINARY_INV | THRESH_OTSU);
bitwise_and( gray, semi_thresholded_image, semi_thresholded_image );
imshow("Semi Thresholding", semi_thresholded_image);
```

This piece of program is to find a new threshold value to into a new one.

```
// Adaptive thresholding
Mat adaptive_thresh;
adaptiveThreshold(gray, adaptive_thresh, 255.0,
ADAPTIVE_THRESH_GAUSSIAN_C, THRESH_BINARY, 101, 10 );
imshow("Adaptive Thresholding", adaptive_thresh);
```

This piece of program is to automatically pick up an area and calculate the average value of the pixel value to decide a specific pixel's value. So the new pixel's value will be a function of other pixels' value in a specific area.

2. What are the disadvantages of binary threshold?

There will only have two colors in the result, black and white. All the pixel values bigger than the threshold will be changed into white, no matter what value the pixel is, and vise versa. For example, if the threshold value is 128, then the pixel with the value of 129 and the pixel with the value of 254 will be like the same, all become white.

3. When is Adaptive Threshold useful?

When an area of the image is missing, and the missing area's pixels have some connections with other areas. Then we can use the Adaptive Threshold to fix the missing area and make the whole image looks better.