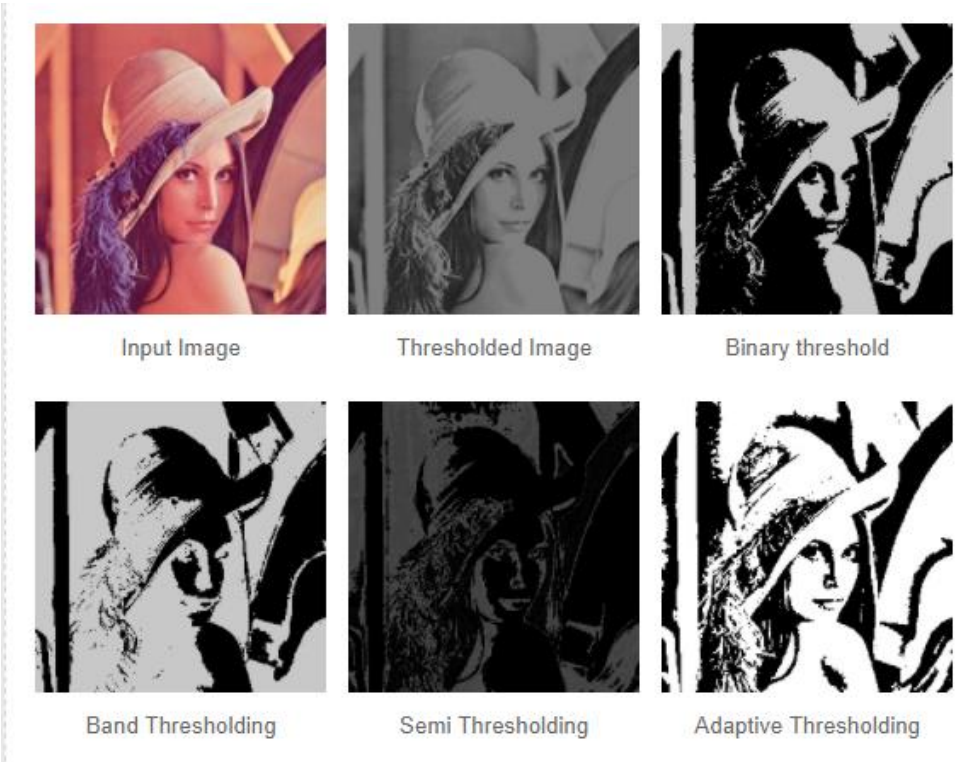
```
int threshold_type = 2;     // slider 1 [0, 4]
int threshold_value = 128;  // slider 2 [0 255]

threshold(gray, dst, threshold_value, 255, threshold_type );
  imshow("Thresholded Image", dst );
```

Threshold_Type = 2 = THRESH_TRUNC

THRESH_TRUNC is a threshold method which vary the result in an unique way.

Threshold_value is a defined threshold value. If a pixel value is beyond this value, it's value will be replaced by a value(in the above pic is 255).



Threshold Image comes from the gray level of input image, and the pixel value beyond threshold value will be replace by 255(255 represents white, 0 represents black).

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

THRESH_BINARY is a method which convert image only in white and black color(each pixel will only be 0 or 255). So, image Binary is only in black and white

```
// 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);
```

THRESH_BINARY_INV is reverse white to black, vice verse. Band thresholding is a image band image from THRESH_BINARY method and THRESH_BINARY_INV method

```
// 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);
```

THRESH_OTSU is a method that will find a threshold value when the origin threshold value is not suitable.

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

Function adaptiveThreshold() will extract a area and compare the mean value(or other methods) in this area with a pixel in order to define its value. Using this function means threshold value is not a const value but a

dynamic suitable value. That’s why Adaptive Threshold is useful(Question 3).

The disadvantages of binary threshold:

I think some pixel value near the threshold value will be(or not)replace by threshold value. So, the result maybe not precise. The contour of origin image can not be display precisely.