ECE-S 681:

Fundamentals of Computer Vision

Prof. Fernand Cohen

Zexi Liu

Feb 2nd, 2010

1. Using the various MATLAB resources draw the ROC curves and compute the areas under ROC Az for the Sobel and Canny edge detectors.
Fig. 1 shows the prototype. They are the true edges expect to be detected.

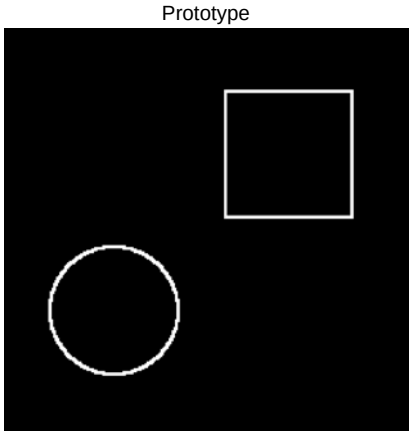

Fig. 1. Prototype

Fig. 2 shows the edge detection results using Sobel method. In the ROC figure, Pd is the detection rate, Pfa is the false alarm rate. The area under ROC curve is calculated and shown in each figure.

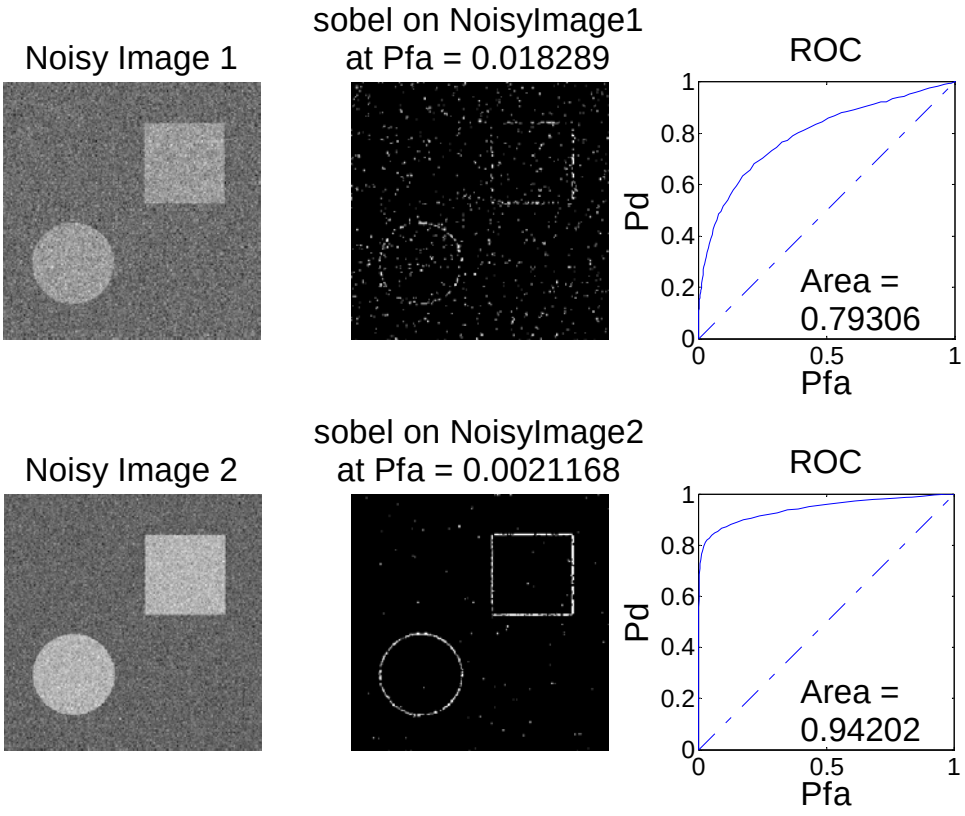

Fig. 2. Implement Sobel detector on two images

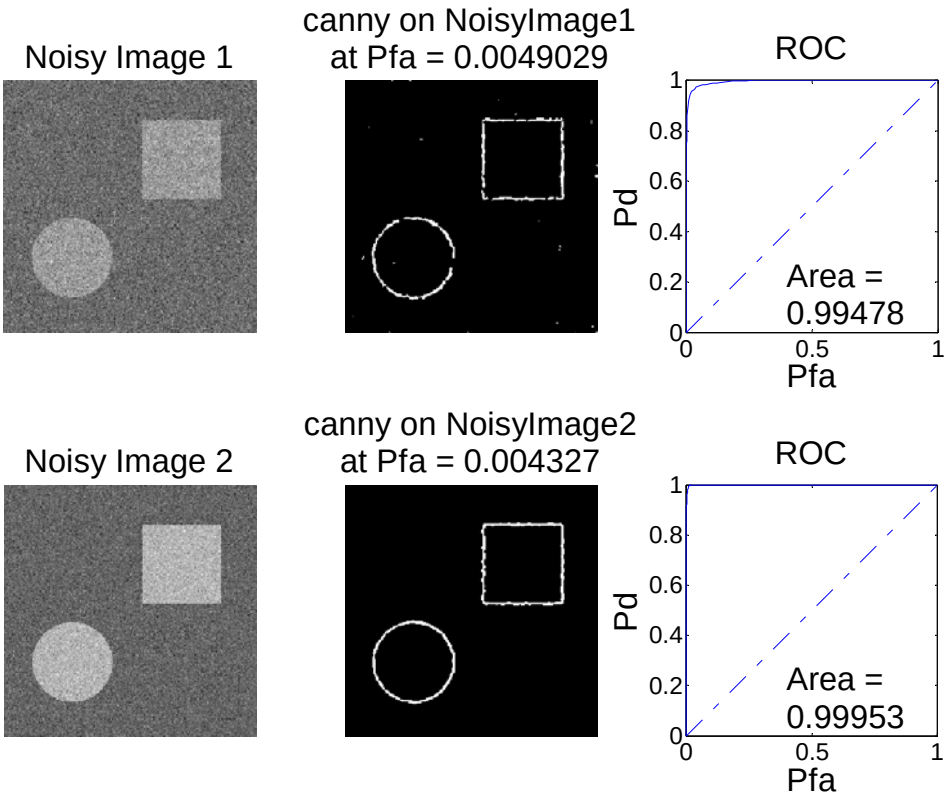

Fig. 3. Implement Canny detector on two images

Remarks: I'm not sure about the ROC curves for canny method. In order to have a complete ROC curves, the value of Pfa (false alarm rate) should be able to sweep from 0 to 1 and the value of Pd (detection rate) should also be able to sweep from 0 to 1. However, the non-maximum suppression step in canny detector will suppress any pixel value (sets it equal to 0) that is not considered to be an edge. Obviously, after this step, the value of Pd cannot be smaller than a certain value (around 0.4).
Canny detector can be divided into two stages. The filtering stage and the detection stage (non-maximum suppression and so on). My ROC curve for canny is based on the gradient matrices after filtering stage. But such ROC curve cannot really evaluate the canny detector since it's only based on the mid-result instead of final result.

2. Repeat the procedure for the coarser image where each pixel is now 2x2, then 4x4. Compute the ROC and the corresponding Az 's. Interpret the results.
Fig. 4 ~ 7 show the down-sampled images (128x128 and 64x63)
Fig. 8 ~11 show the up-sampled images (512x512 and 1024x1024)

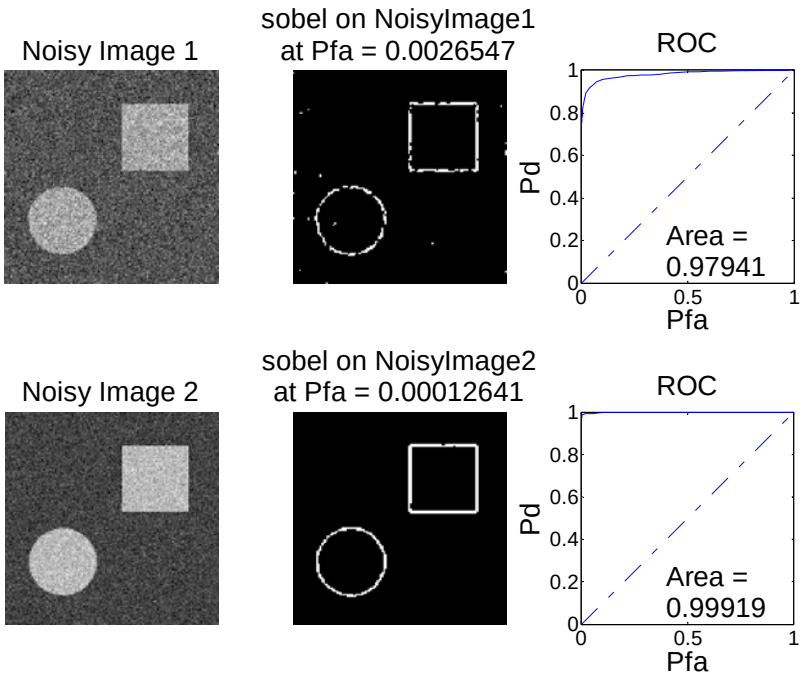

Fig. 5. Implement Sobel detector on two finer images (128x128)

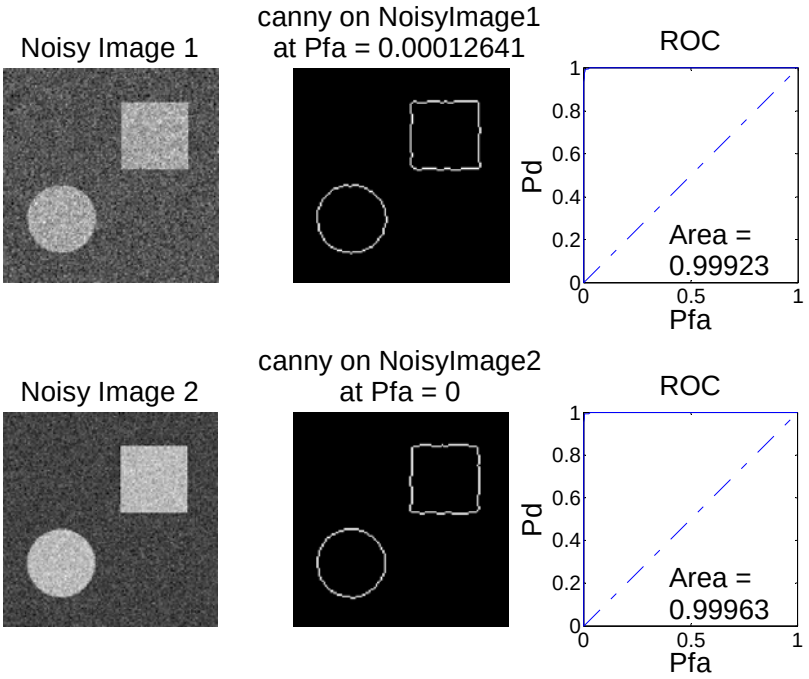

Fig. 4. Implement Canny detector on two finer images (128x128)

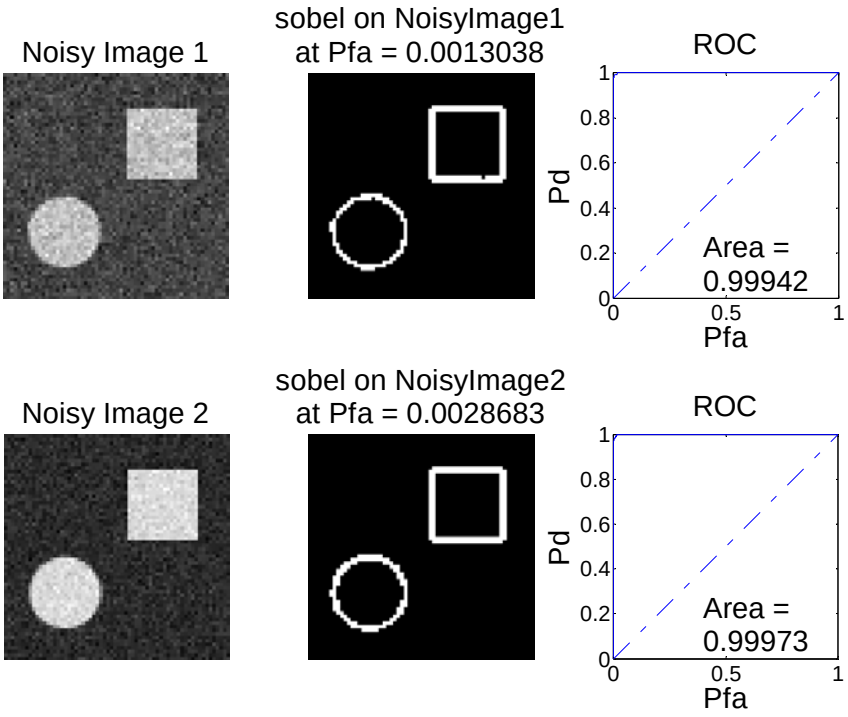

Fig. 6. Implement Sobel detector on two finer images (64x64)

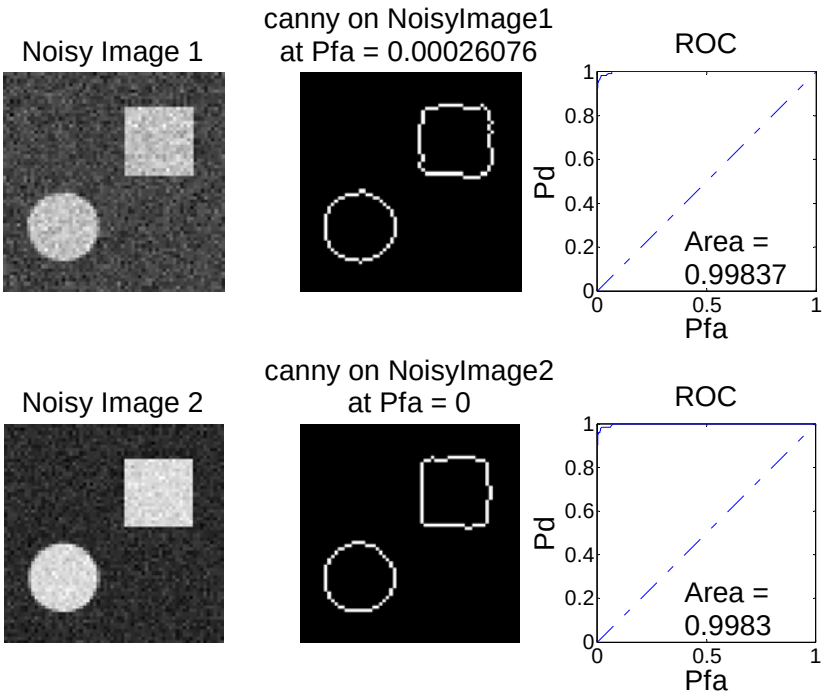

Fig. 7. Implement Canny detector on two finer images (64x64)

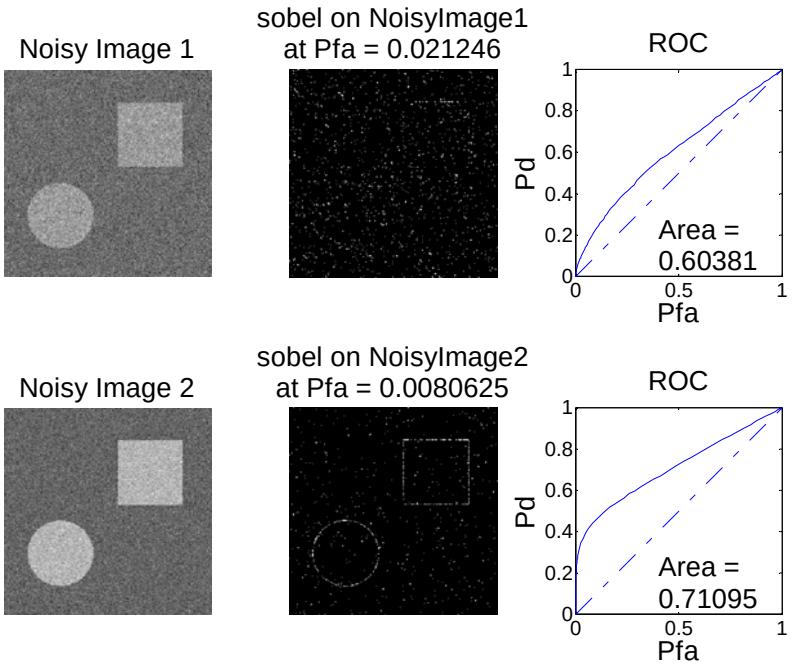

Fig. 8. Implement Sobel detector on two coarser images (512x512)

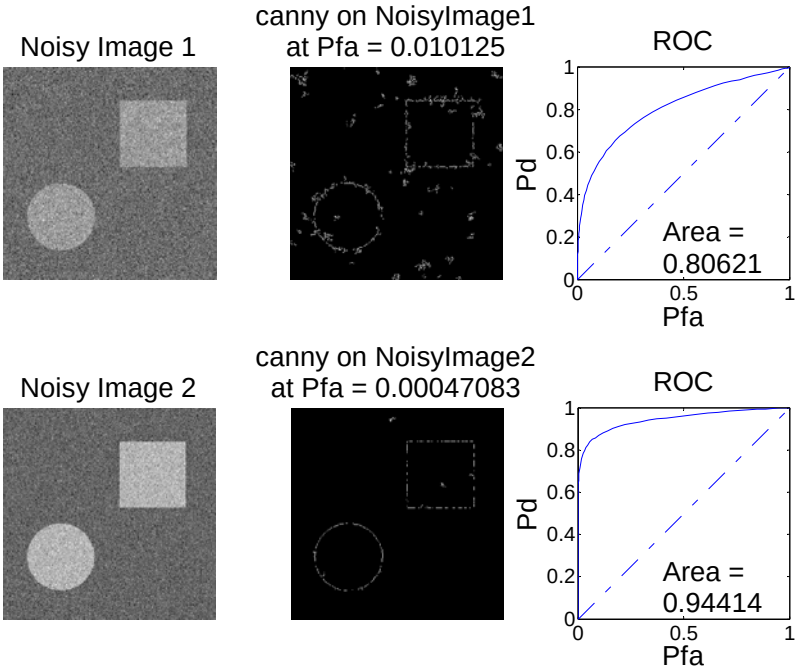

Fig. 9. Implement Canny detector on two coarser images (512x512)

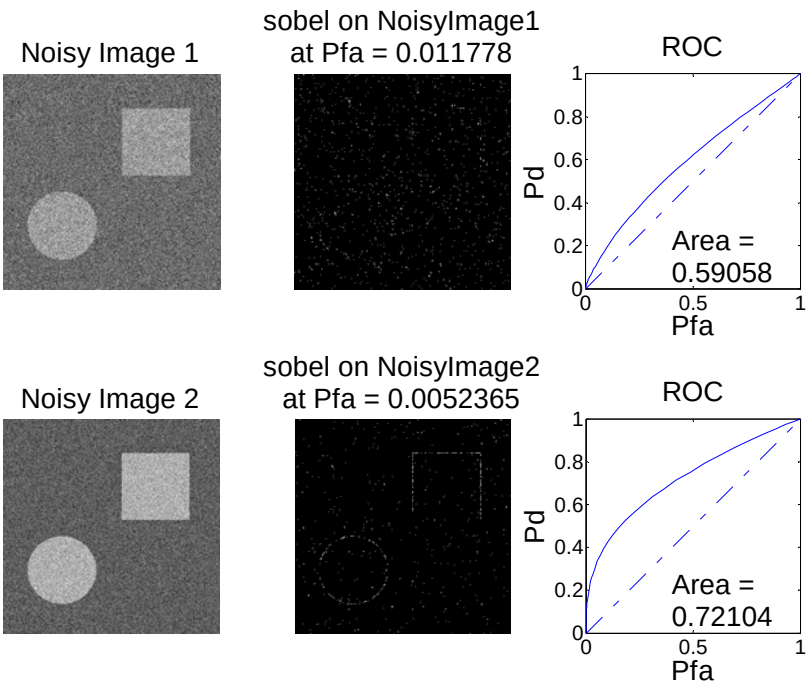

Fig. 10. Implement Sobel detector on two coarser images (1024x1024)

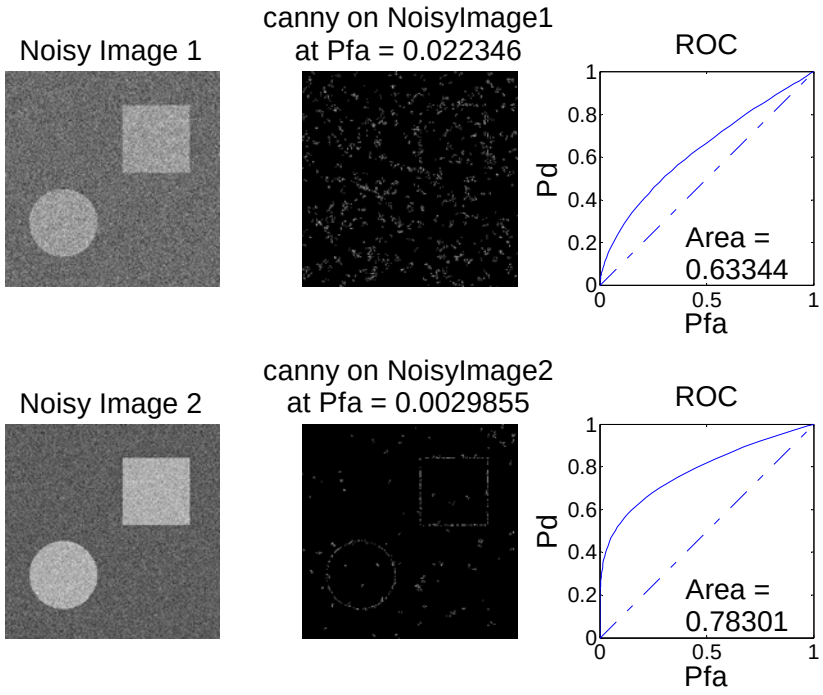

Fig. 11. Implement Canny detector on two coarser images (1024x1024)

See Fig. 4 ~ 7. If the image is resized smaller i.e. lower resolution, the area under ROC curve will increase comparing to the original image. This is because the downsizing procedure is actually a filter procedure. Some noise was deleted from the image, which makes the detector work in a better condition. On the other hand (See Fig. 8 ~ 11), if the image is resized larger i.e. higher resolution, the area under ROC curve will decrease comparing to the original image. The reason for such decreasing is upsizing will introduce more noise into the image, in other words, make the image coarser. And this will lead to a lower detection rate.

Also, canny detector is superior to sobel detector in any conditions in our tests. Because canny detector has two stages, the filter stage and the detection stage, such that canny detector is more robust in the presents of noises.

3. If you were to compare the detection rates at various resolutions at fixed false alarm rates, what do you observe and why?

The detection rates at lower resolutions are smaller than the detection rates at higher resolutions at fixed false alarm rates. Moreover, the lower the resolution is, the higher the detection rate is obtained. This is because lower resolution indicates lower SNR and vice versa.

4. How would you go about using a multi-resolution scheme to combat the uncertainty principle? Implant your scheme and report on the results.

Based on Problem 2 and 3, my scheme is to make use of the lower resolution images. First, the original image is down-sampling to 128x128 and 64x64. Then Sobel or Canny detector is used on these down-sampled images. The detection results will of course in 128x128 and 64x64 resolution.

Next, these output images are up-sampled back to original resolution 256x256. (See Fig. 12 (a), (b), and(c)) The detected edges are thicker in image (a) and (b).

Then, since all the images are logical valued, the logical calculation, logical AND and OR can be used on these images. In Fig. 12, image (d) is produced as image (a) AND (b) AND (c) i.e. the edge pixels in image (d) is the common pixels of (a), (b), and (c). Matlab function “bwmorph” is used to thin the edges of (a) AND (b). This will give us image (e). In the end, image (d) OR (e) = (f) is the final result. (f) contains both (d) and (e)’s edge pixels.
The false alarm rate of image (c) is 0.0162 with detection rate 0.2329.
The false alarm rate of image (f) is 0.0013 with detection rate 0.4286.

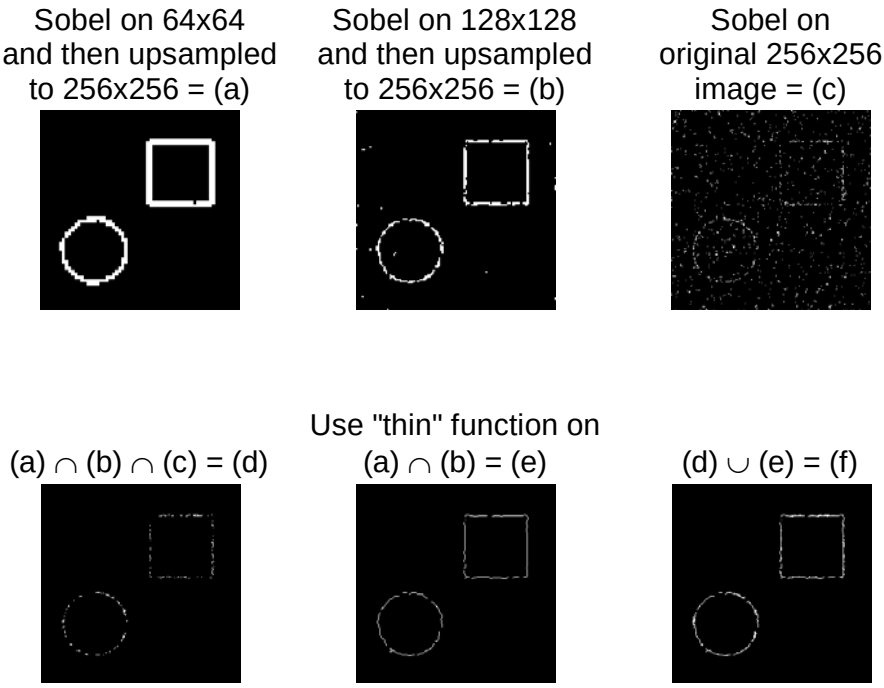

Fig. 12. Using multi-resolution scheme to combat uncertainty principle