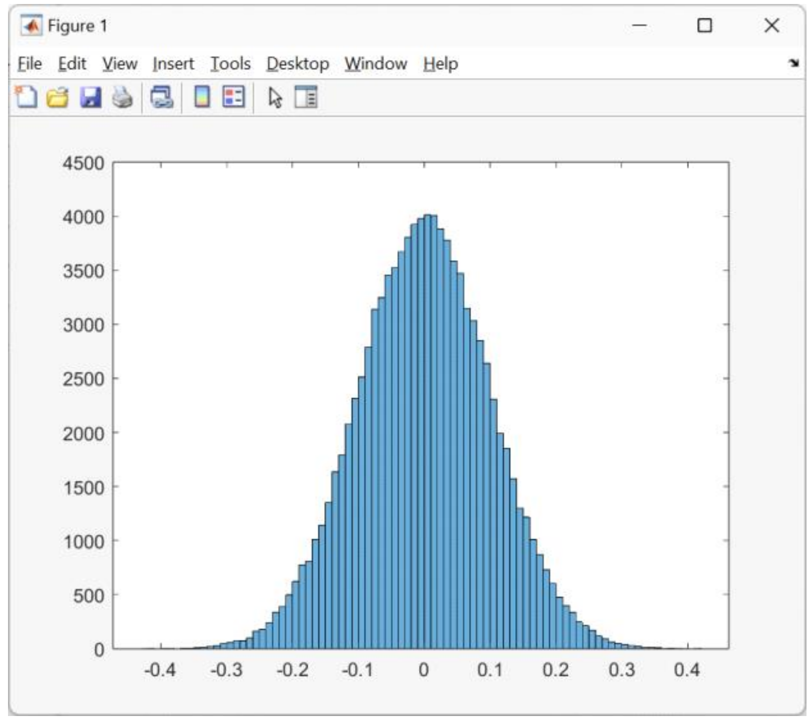
Lecture 7

De-Noising (cont'd)

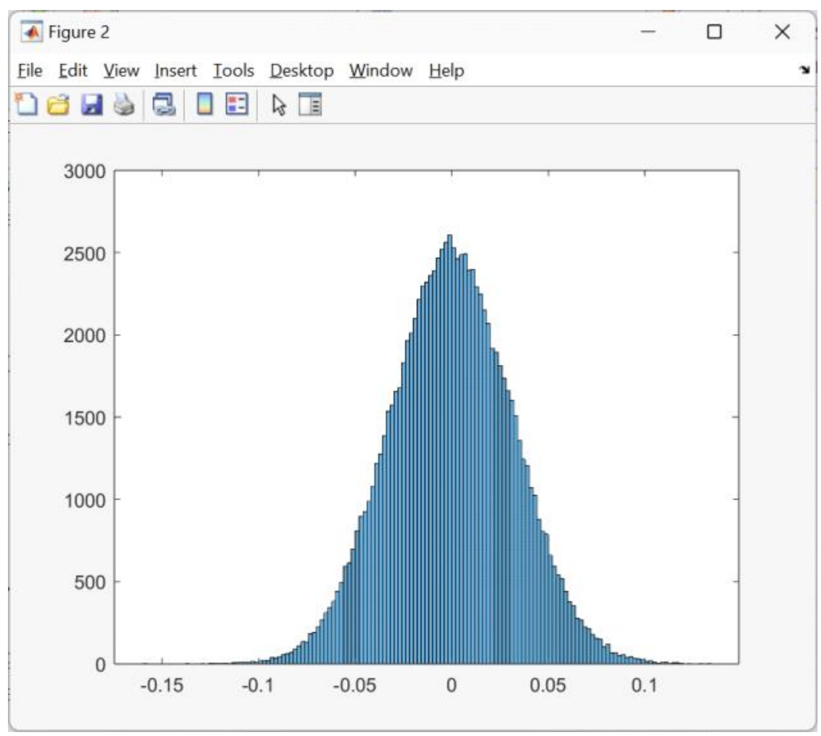
```
>> sample_size = 100;
>> run = 100000;
>> Xavg = zeros(1, run);
for i = 1: run
        X = randn(1, sample_size);
        Xavg(i) = mean(X);
    end
>> histogram(Xavg)

>> mean(Xavg)
ans =
  1.6542e-04

>> var(Xavg)
ans =
    0.0100
```

$= \frac{1}{100}$

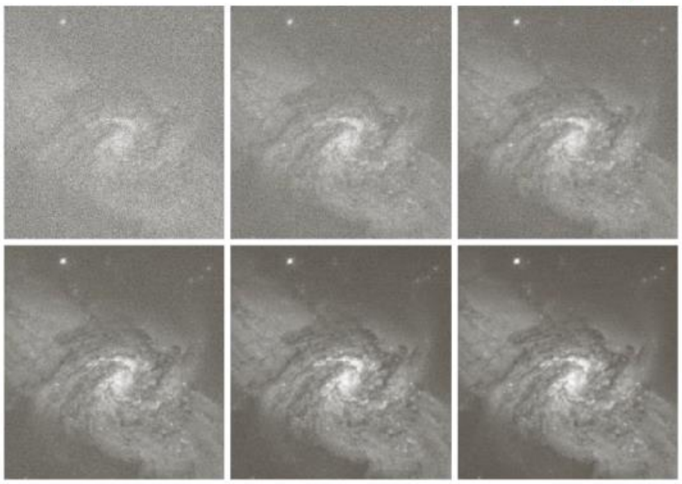



```
>> sample_size = 1000;
>> Xavg = zeros(1, run);
for i = 1: run
        X = randn(1, sample_size);
        Xavg(i) = mean(X);
    end
>> mean(Xavg)
ans =
 -1.0231e-04

>> var(Xavg)
ans =
    0.0010
```

$= \frac{1}{1000}$



De-Noising



a b c
d e f

FIGURE 2.26 (a) Image of Galaxy Pair NGC 3314 corrupted by additive Gaussian noise. (b)–(f) Results of averaging 5, 10, 20, 50, and 100 noisy images, respectively. (Original image courtesy of NASA.)

$$g(x, y) = f(x, y) + \eta(x, y)$$

$$\downarrow$$

$$\bar{g}(x, y) = \frac{1}{K}\sum_{i=1}^{K} g_i(x, y)$$

$$E\{\bar{g}(x, y)\} = f(x, y)$$

$$\sigma^2_{\bar{g}(x,y)} = \frac{1}{K}\sigma^2_{\eta(x,y)}$$

Assumption: Noise is uncorrelated to image and has zero mean.

Define Sample Mean $\hat{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \Rightarrow E[\hat{X}] = E\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] = \frac{1}{n}\sum_{i=1}^{n} E[X_i] = \underbrace{\bar{X}}_{\text{True Mean } E[X]}$

$$\mathrm{Var}(\hat{X}) = \underbrace{E\left[(\hat{X})^2\right]}_{?} - \underbrace{E^2[\bar{X}]}_{E^2[X]}$$

where

$$E\left\{\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)\left(\frac{1}{n}\sum_{j=1}^{n} X_j\right)\right\} = E\left[\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(X_i X_j)\right] = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\underbrace{E[X_i X_j]}$$

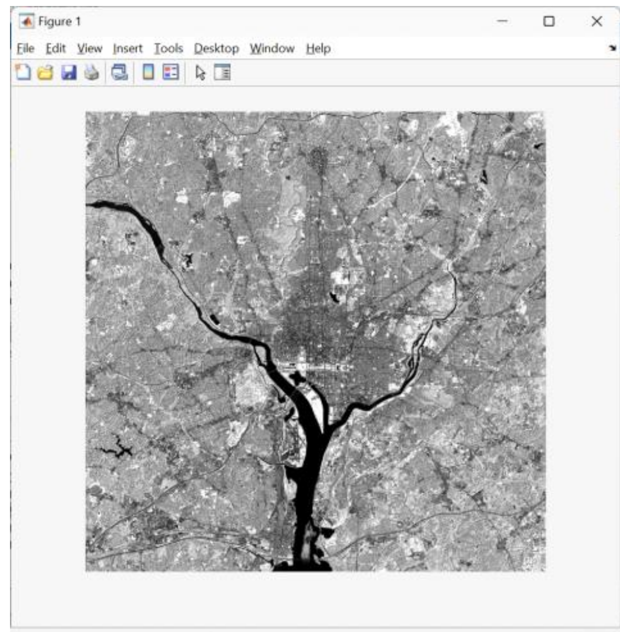
and

$$E[X_i X_j] = \begin{cases} E[X^2] & \text{, if } i=j \\ \underbrace{E[X_i]}_{E[X]\cdot} \underbrace{E[X_j]}_{E[X]} & \text{, if } i \neq j \end{cases}$$ since X_i and X_j are uncorrelated

Thus $E\left[(\hat{X})^2\right] = \frac{1}{n^2}\left\{ n \cdot E[X^2] + (n^2 - n)\, E^2[X] \right\}$

$$= \frac{Var[X]}{n} + E^2[X], \quad \text{Since } Var[X] = E[X^2] - E^2[X]$$

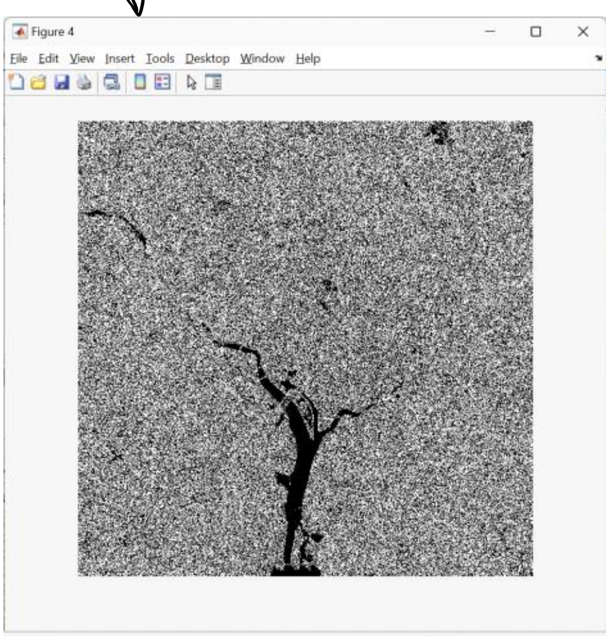
Therefore, $\boxed{Var(\hat{X}) = \frac{Var[X]}{n}}$



- Bitplane Processing

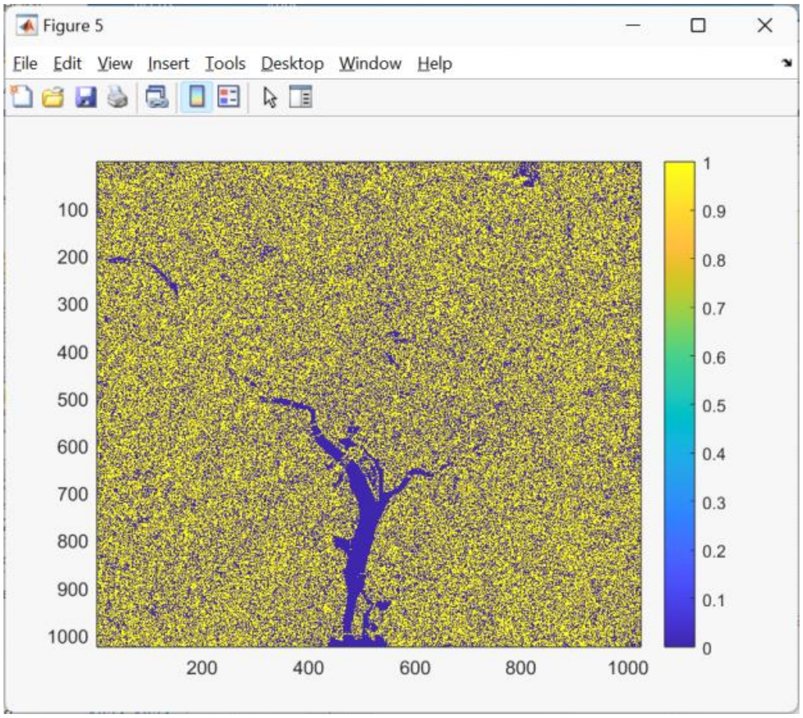
http://www.ece.uah.edu/~dwpan/course/ee604/code/ch2/

```
% Fig. 2.27
% Setting the least significant bit to zero, then scale
% the difference to [0,255] for clarity

I = imread('Fig0227(a)(washington_infrared).tif');
J = bitset(I, 1, 0);
Diff = I - J;
D = 255*(I - J);
imshow(D)
```




```
>> A = uint8(3)

A =

  uint8

   3
```

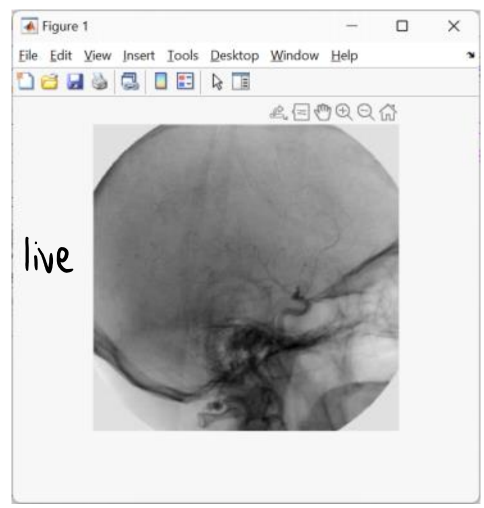
```
>> B = bitset(A, 1, 0)

B =

  uint8

   2
```

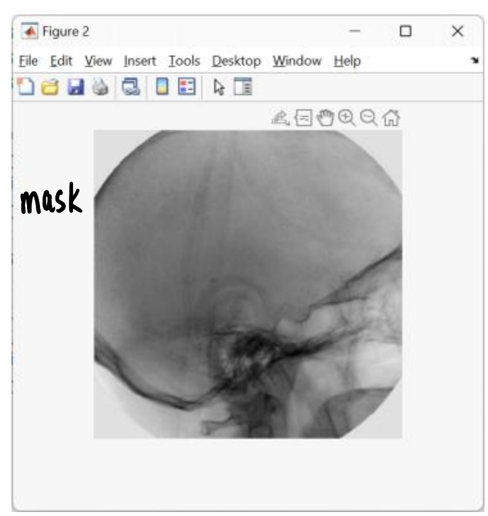


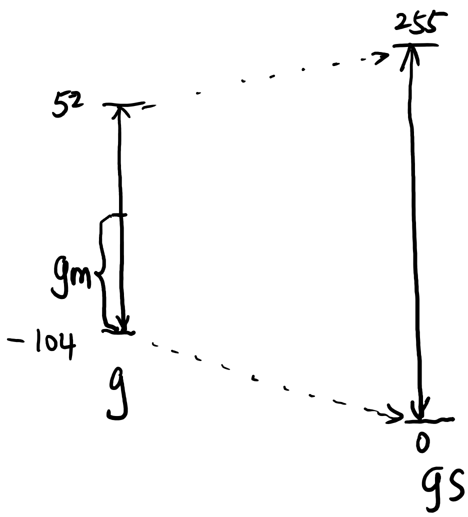
>> figure; imagesc(Diff)
>> colorbar

```
% Fig. 2.28
% g(x,y) = f(x,y) - h(x,y), where h(x,y) is the mask
% Enhance g(x,y)
f = imread('Fig0228(b)(angiography_live_ image).tif');
h = imread('Fig0228(a)(angiography_mask_image).tif');
imshow(f);
figure;
imshow(h);
g = double(f) - double(h);
% Enhance difference
gm = g - min(min(g));
gm_max = max(max(gm));
gs = 255*(gm/gm_max);
figure;
imshow(uint8(gs))
```

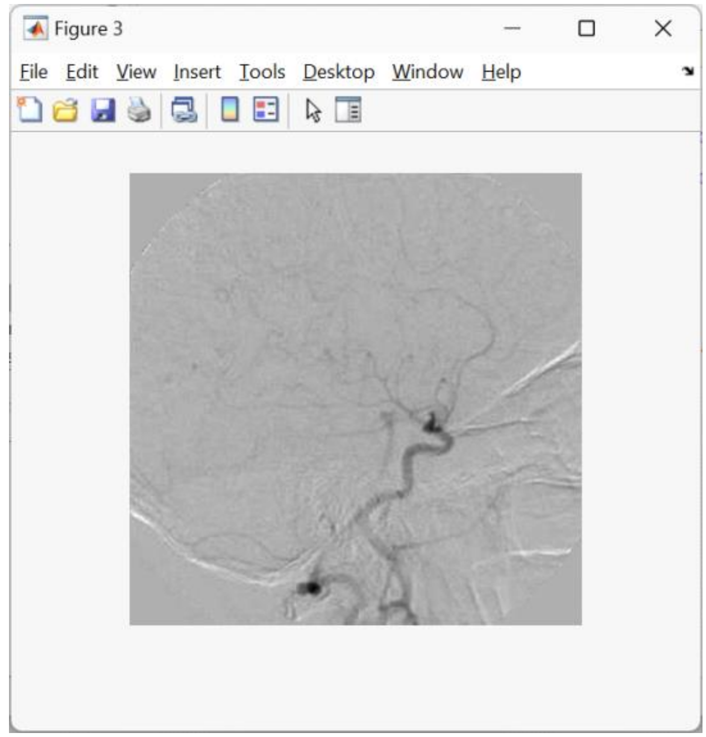
gm_max = 156
= 52 - (-104)







gs



- Output images should be normalized to the range of $[0, 255]$.

$$f_m = f - \min(f)$$
$$f_s = K[f_m / \max(f_m)]$$