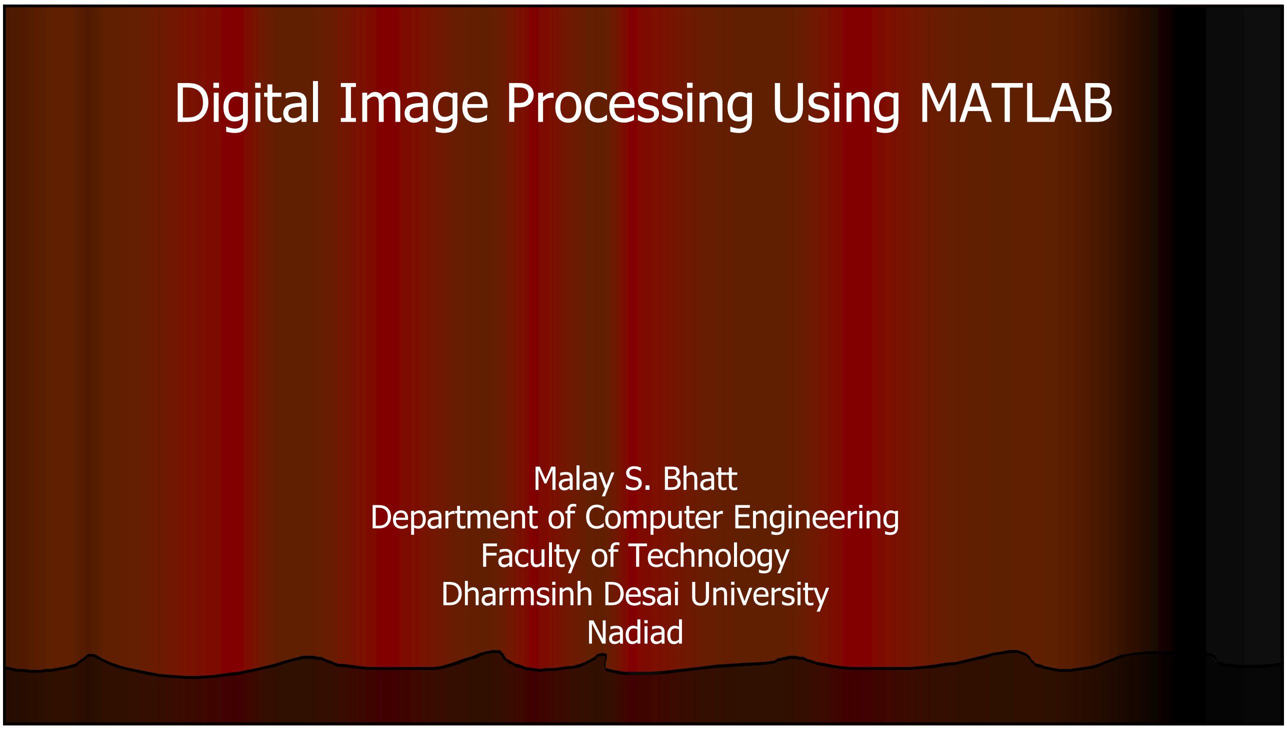


Digital Image Processing Using MATLAB

Malay S. Bhatt
Department of Computer Engineering
Faculty of Technology
Dharmsinh Desai University
Nadiad

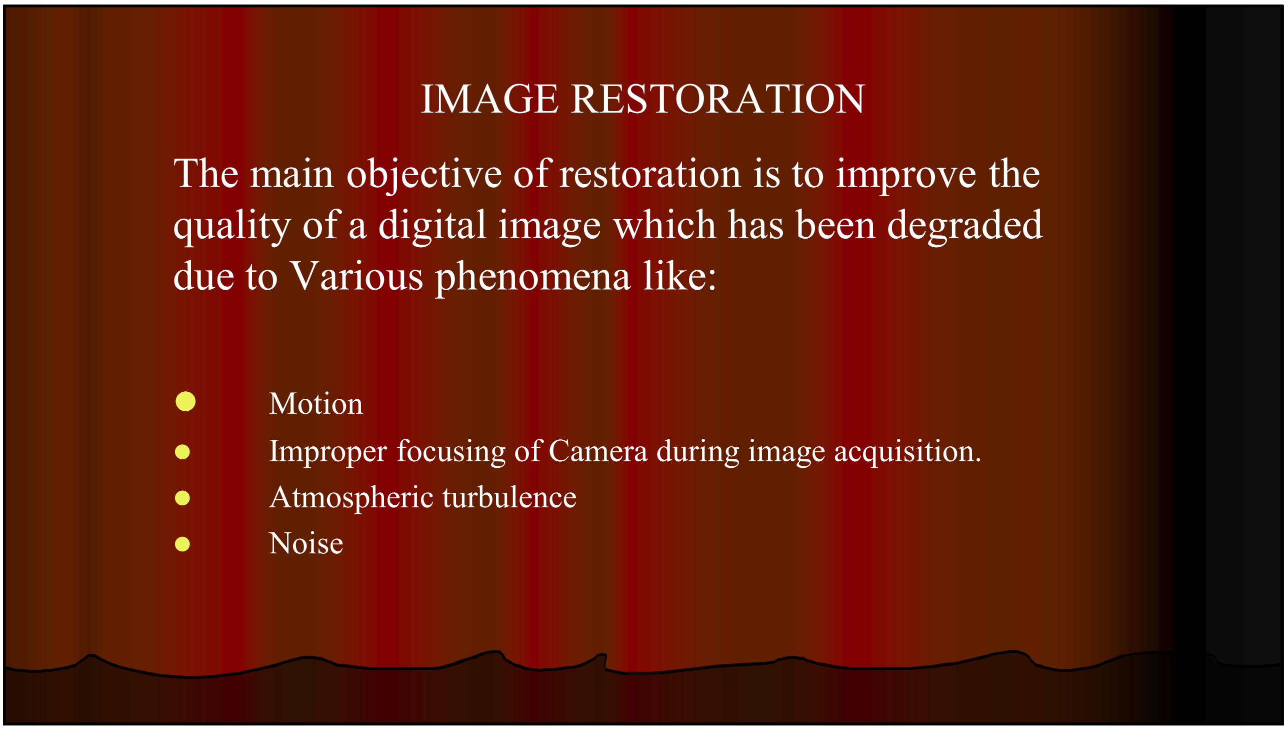
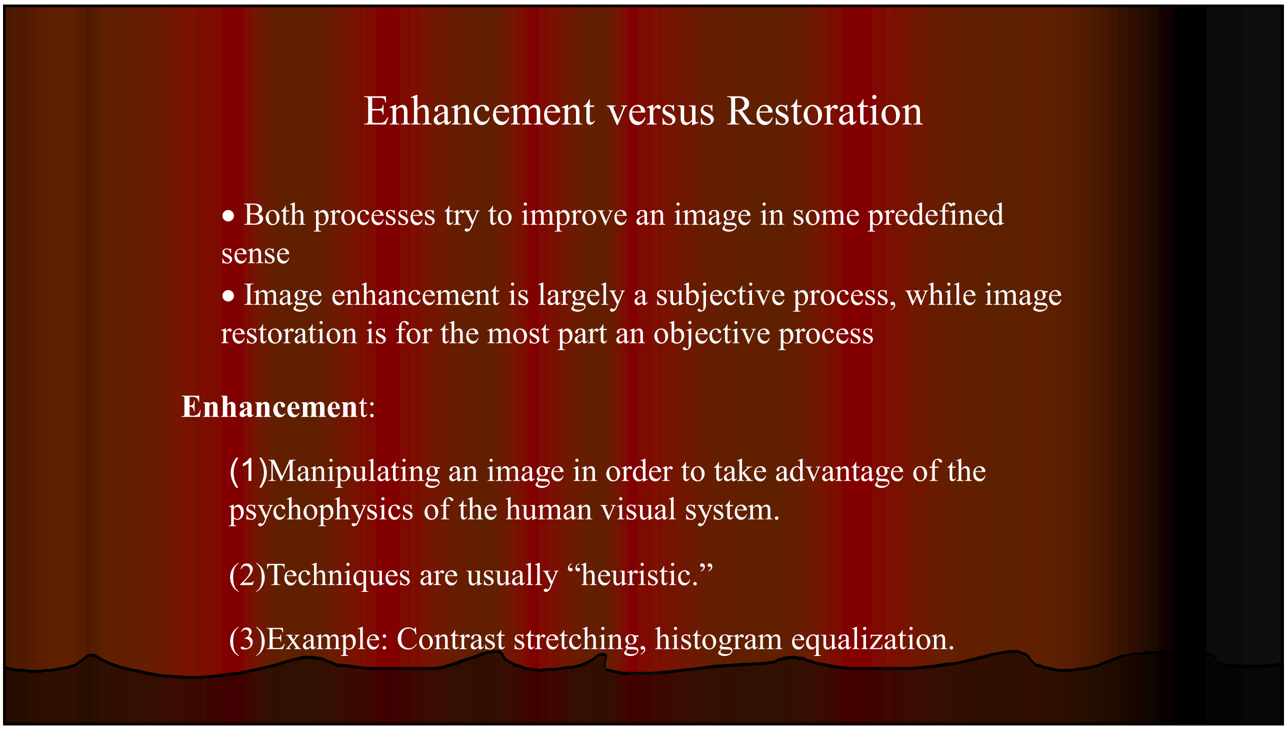


IMAGE RESTORATION

The main objective of restoration is to improve the quality of a digital image which has been degraded due to Various phenomena like:

- Motion
- Improper focusing of Camera during image acquisition.
- Atmospheric turbulence
- Noise



Enhancement versus Restoration

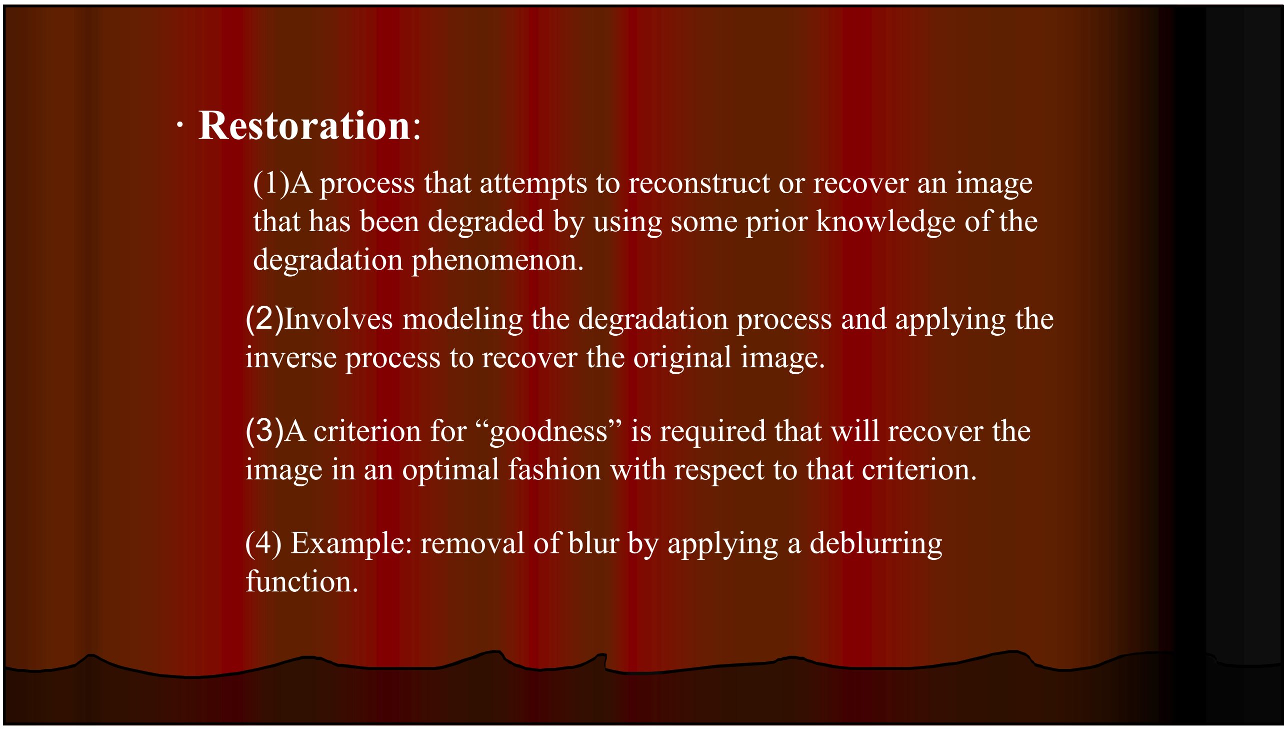
- Both processes try to improve an image in some predefined sense
- Image enhancement is largely a subjective process, while image restoration is for the most part an objective process

Enhancement:

(1)Manipulating an image in order to take advantage of the psychophysics of the human visual system.

(2)Techniques are usually "heuristic."

(3)Example: Contrast stretching, histogram equalization.



- **Restoration**:

(1)A process that attempts to reconstruct or recover an image that has been degraded by using some prior knowledge of the degradation phenomenon.

(2)Involves modeling the degradation process and applying the inverse process to recover the original image.

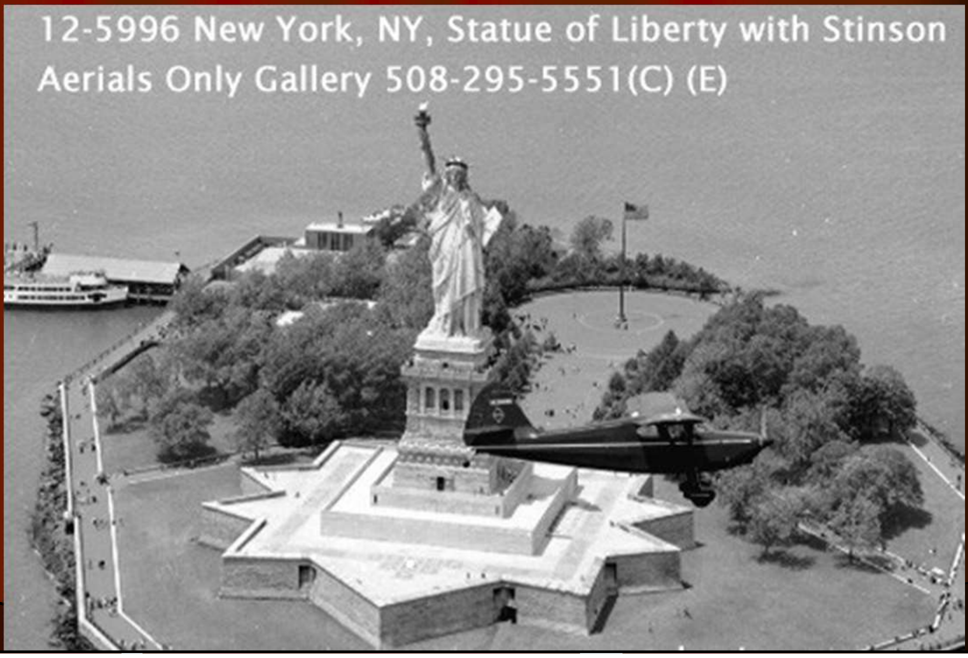
(3)A criterion for "goodness" is required that will recover the image in an optimal fashion with respect to that criterion.

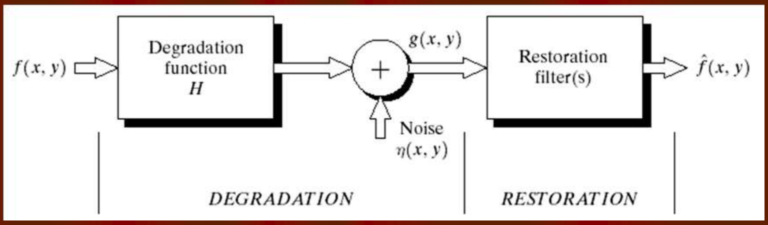
(4) Example: removal of blur by applying a deblurring function.

Problem:

- You want to know some image X.
- But you only have a corrupted version Y .
- How do you determine X from Y ?

Blurring due to uniform motion

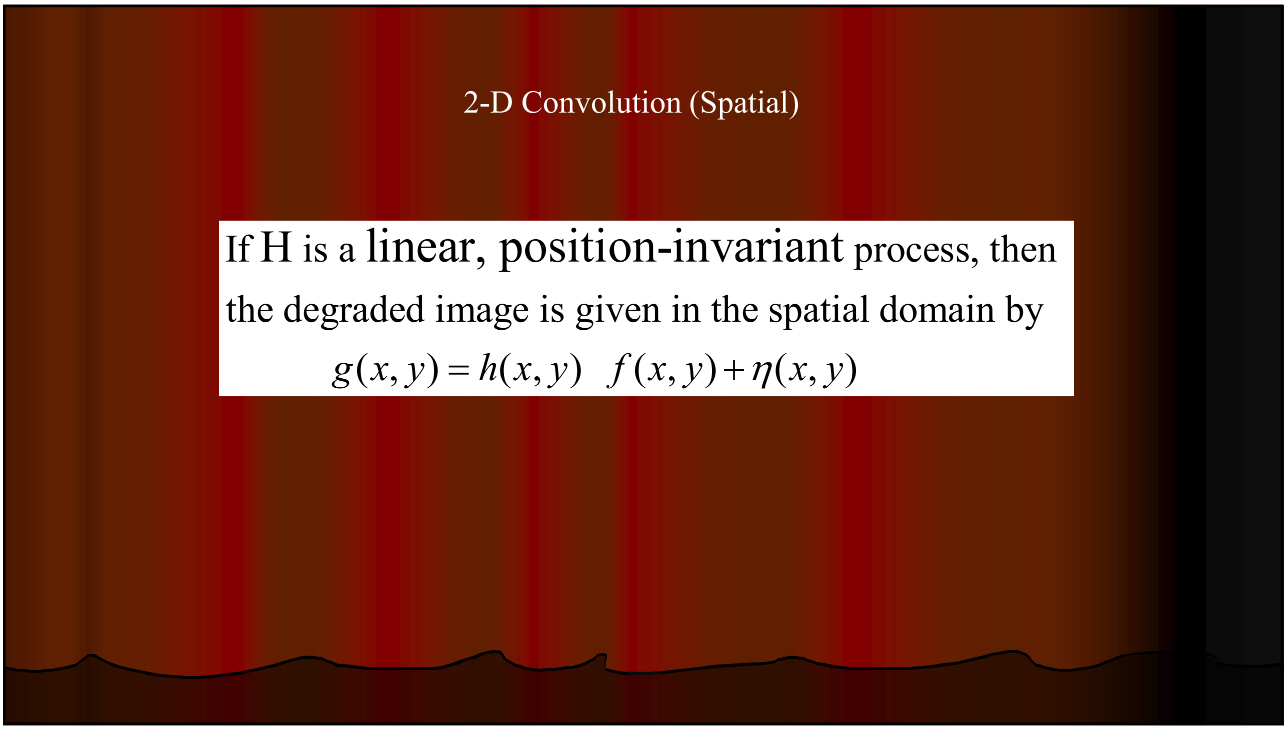

12-5996 New York, NY, Statue of Liberty with Stinson
Aerials Only Gallery 508-295-5551(C) (E)



Degradation model:

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

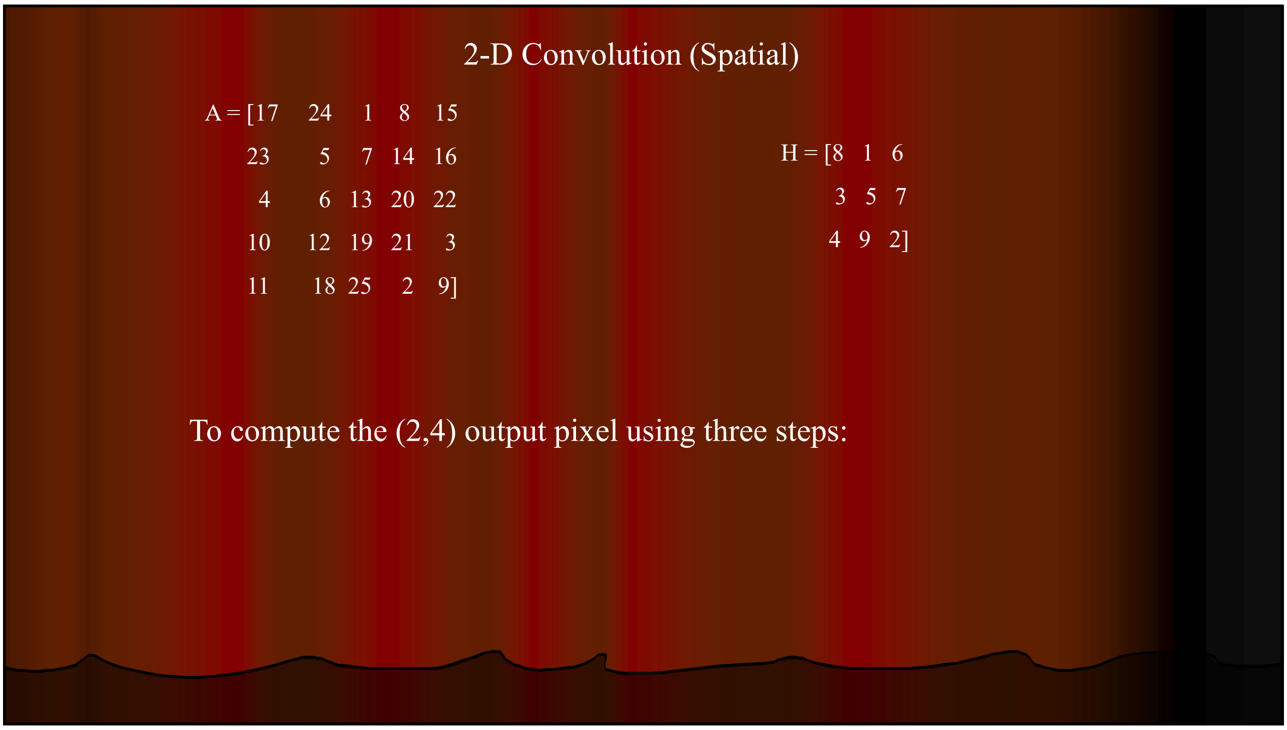
where $h(x,y)$ is a system that causes image distortion and $\eta(x,y)$ is noise.



2-D Convolution (Spatial)

If H is a linear, position-invariant process, then the degraded image is given in the spatial domain by

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

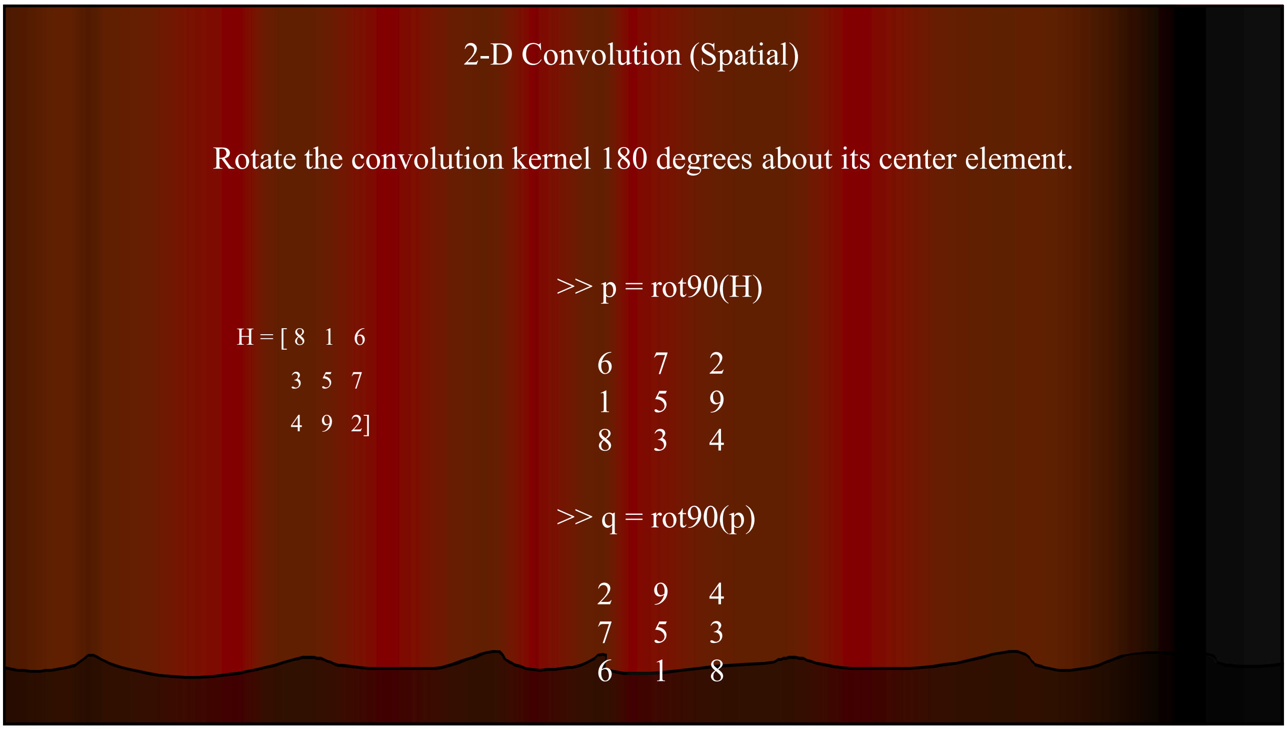


2-D Convolution (Spatial)

A = [17 24 1 8 15
23 5 7 14 16
4 6 13 20 22
10 12 19 21 3
11 18 25 2 9]

H = [8 1 6
3 5 7
4 9 2]

To compute the (2,4) output pixel using three steps:



2-D Convolution (Spatial)

Rotate the convolution kernel 180 degrees about its center element.

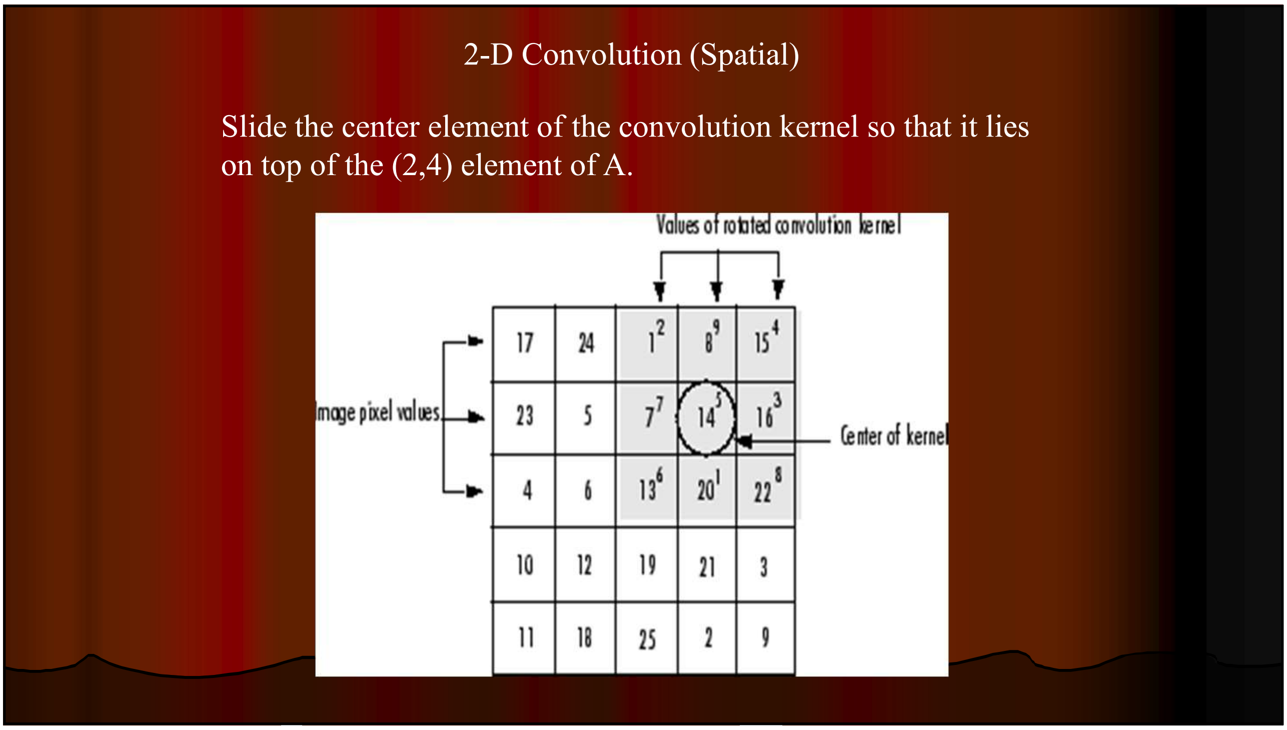
```
H = [ 8   1   6
      3   5   7
      4   9   2]
```

```
>> p = rot90(H)

   6    7    2
   1    5    9
   8    3    4

>> q = rot90(p)

   2    9    4
   7    5    3
   6    1    8
```



2-D Convolution (Spatial)

Slide the center element of the convolution kernel so that it lies on top of the (2,4) element of A.

17	24	1^2	8^9	15^4
23	5	7^7	14^5	16^3
4	6	13^6	20^1	22^8
10	12	19	21	3
11	18	25	2	9

2-D Convolution (Spatial)

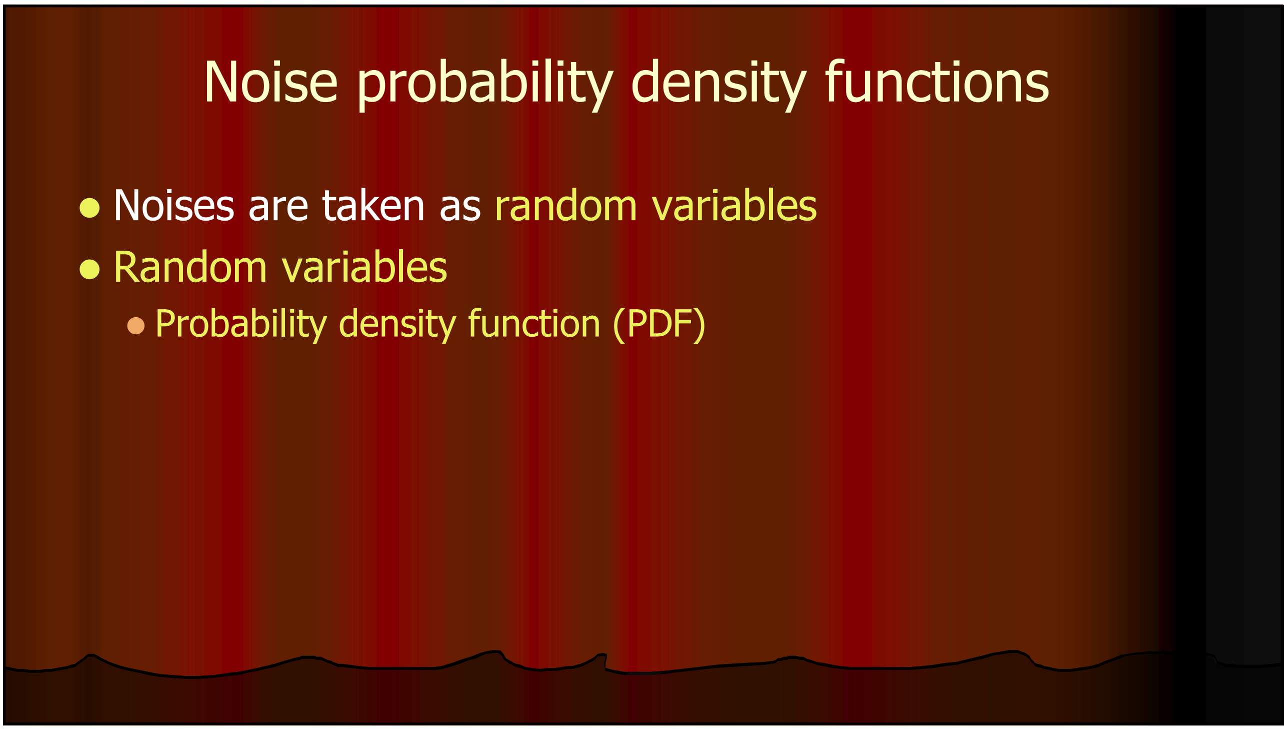
Multiply each weight in the rotated convolution kernel by the pixel of A underneath.

Sum the individual products from step 3.

$$1\cdot 2+8\cdot 9+15\cdot 4+7\cdot 7+14\cdot 5+16\cdot 3+13\cdot 6+20\cdot 1+22\cdot 8=575$$

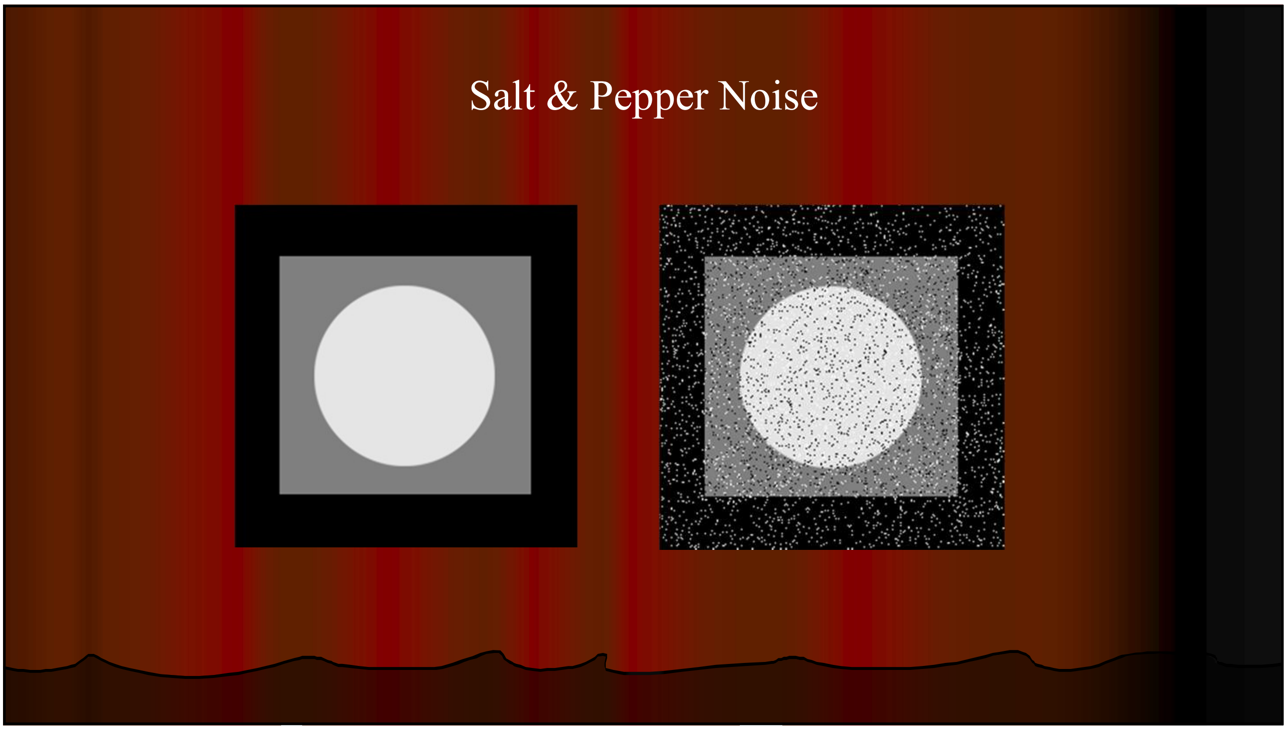
Noise Sources

- The principal sources of noise in digital images arise during **image acquisition and/or transmission**
- Image acquisition
  e.g., light levels, sensor temperature, etc.
- Transmission
  e.g., lightning or other atmospheric disturbance in wireless network

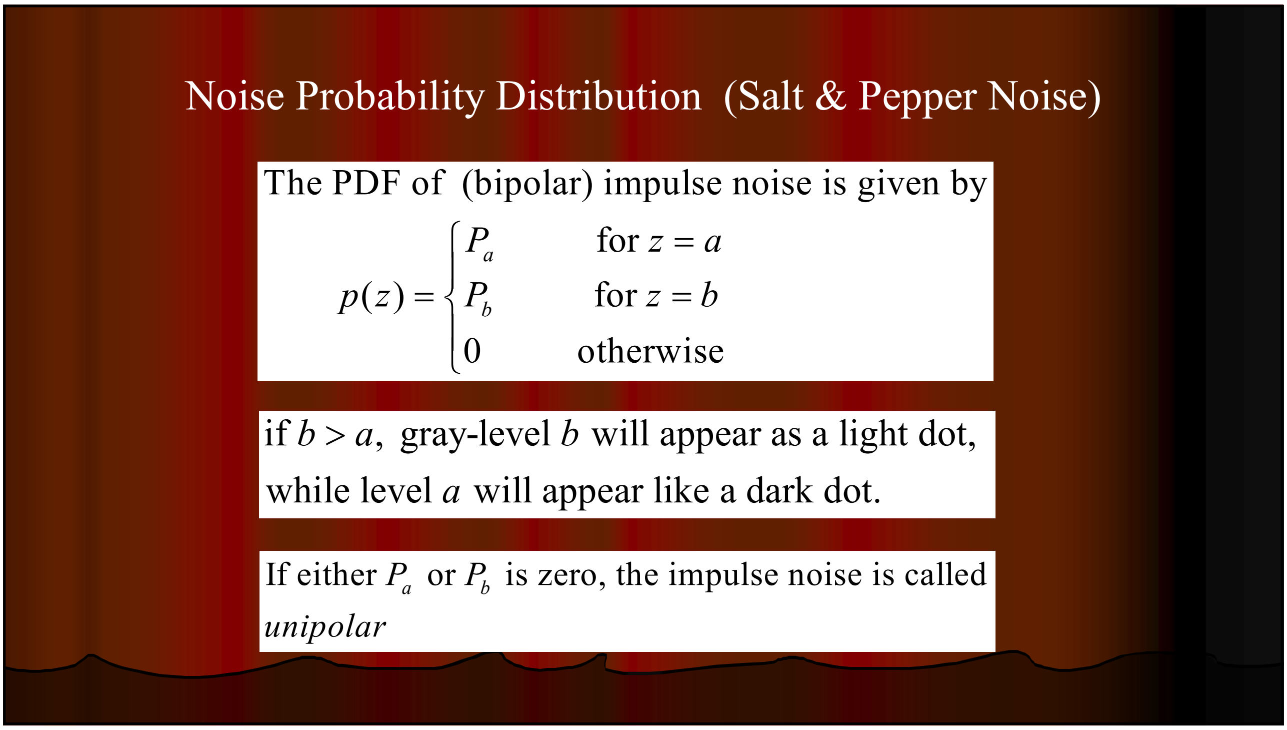


Noise probability density functions

- Noises are taken as random variables
- Random variables
  - Probability density function (PDF)



Salt & Pepper Noise



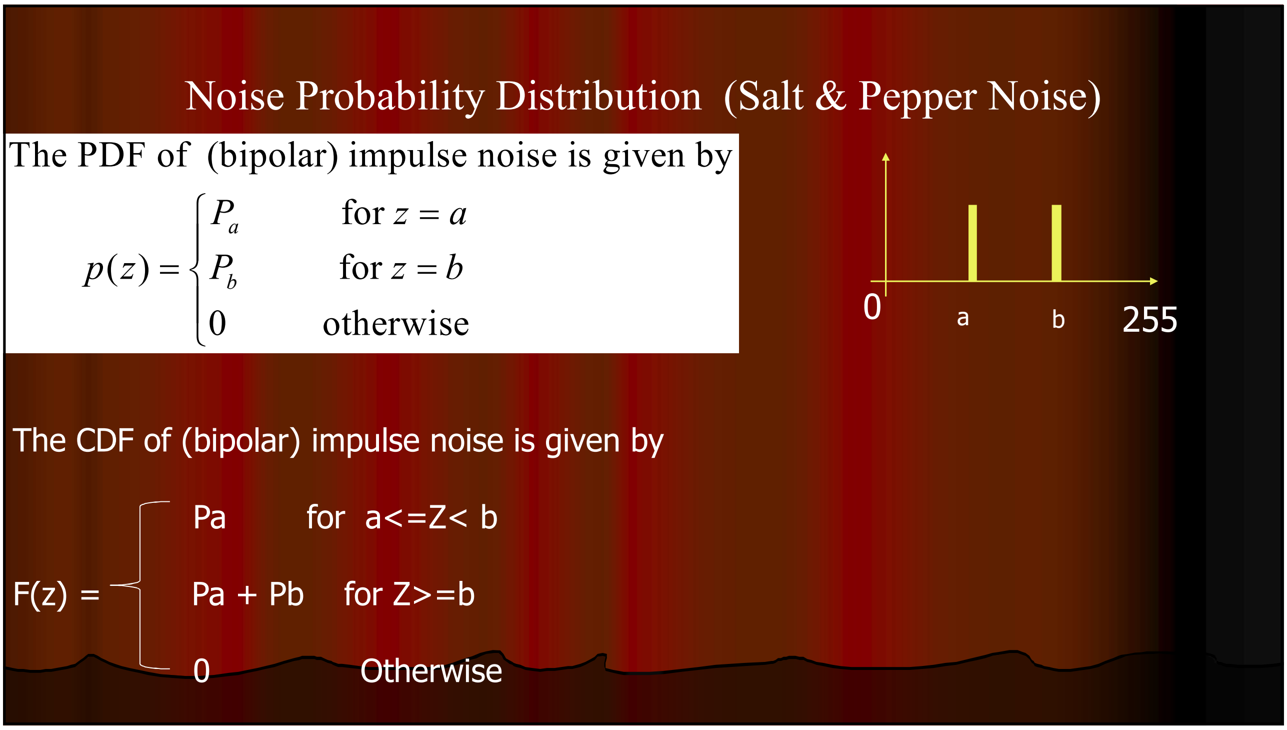
Noise Probability Distribution (Salt & Pepper Noise)

The PDF of (bipolar) impulse noise is given by

$$p(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise} \end{cases}$$

if $b > a$, gray-level b will appear as a light dot, while level a will appear like a dark dot.

If either P_a or P_b is zero, the impulse noise is called *unipolar*



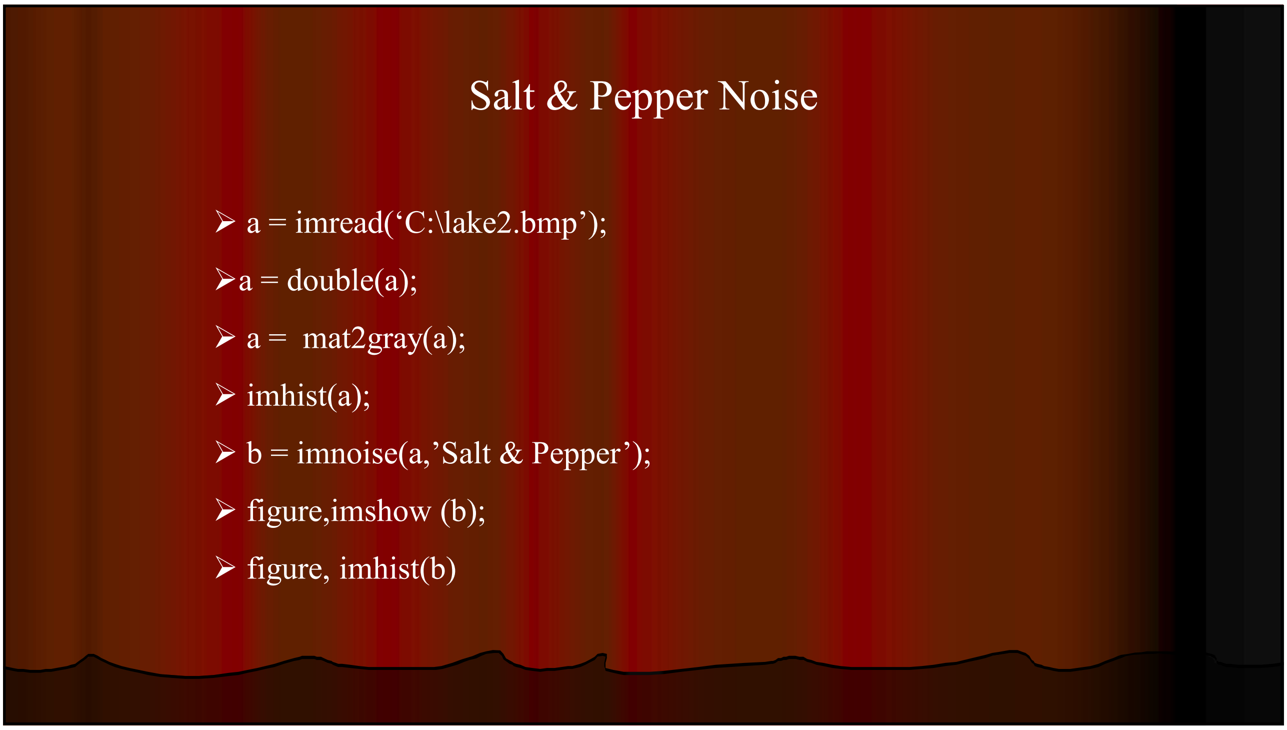
Noise Probability Distribution (Salt & Pepper Noise)

The PDF of (bipolar) impulse noise is given by

$$p(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise} \end{cases}$$

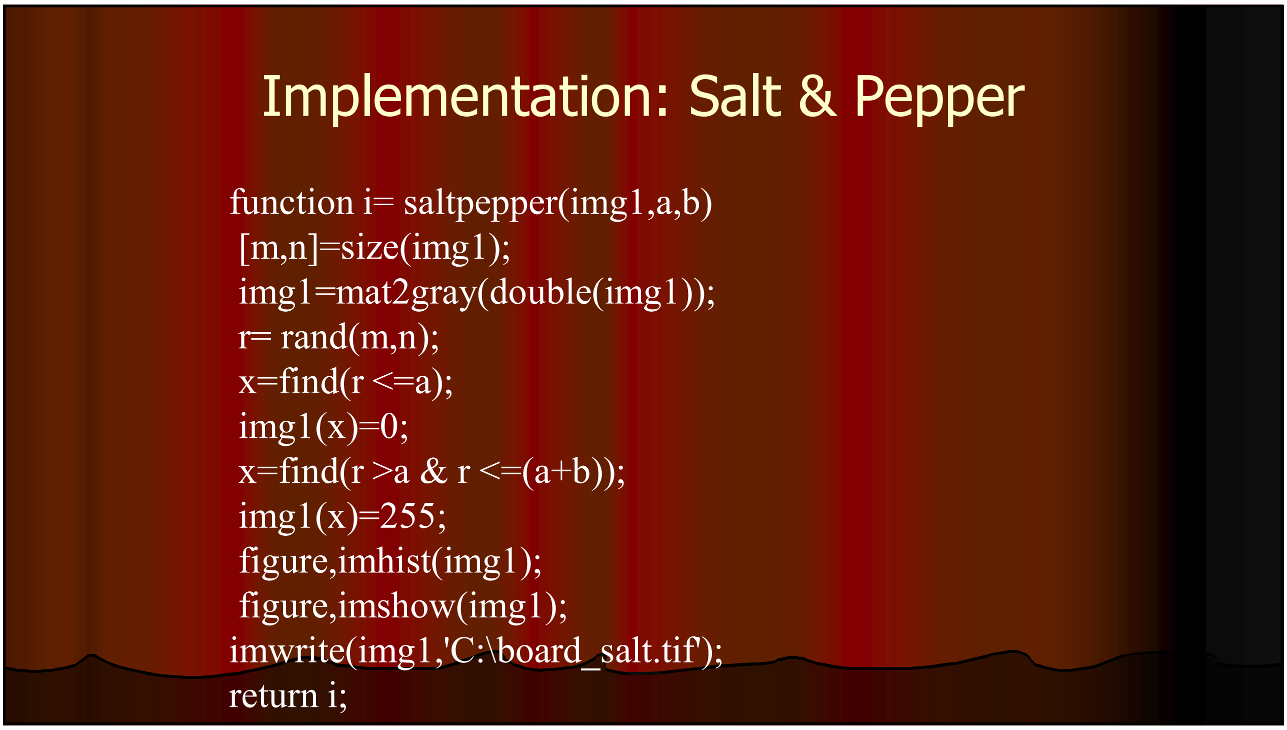
The CDF of (bipolar) impulse noise is given by

$$F(z) = \begin{cases} \text{Pa} & \text{for } \text{a<=Z< b} \\ \text{Pa + Pb} & \text{for Z>=b} \\ 0 & \text{Otherwise} \end{cases}$$



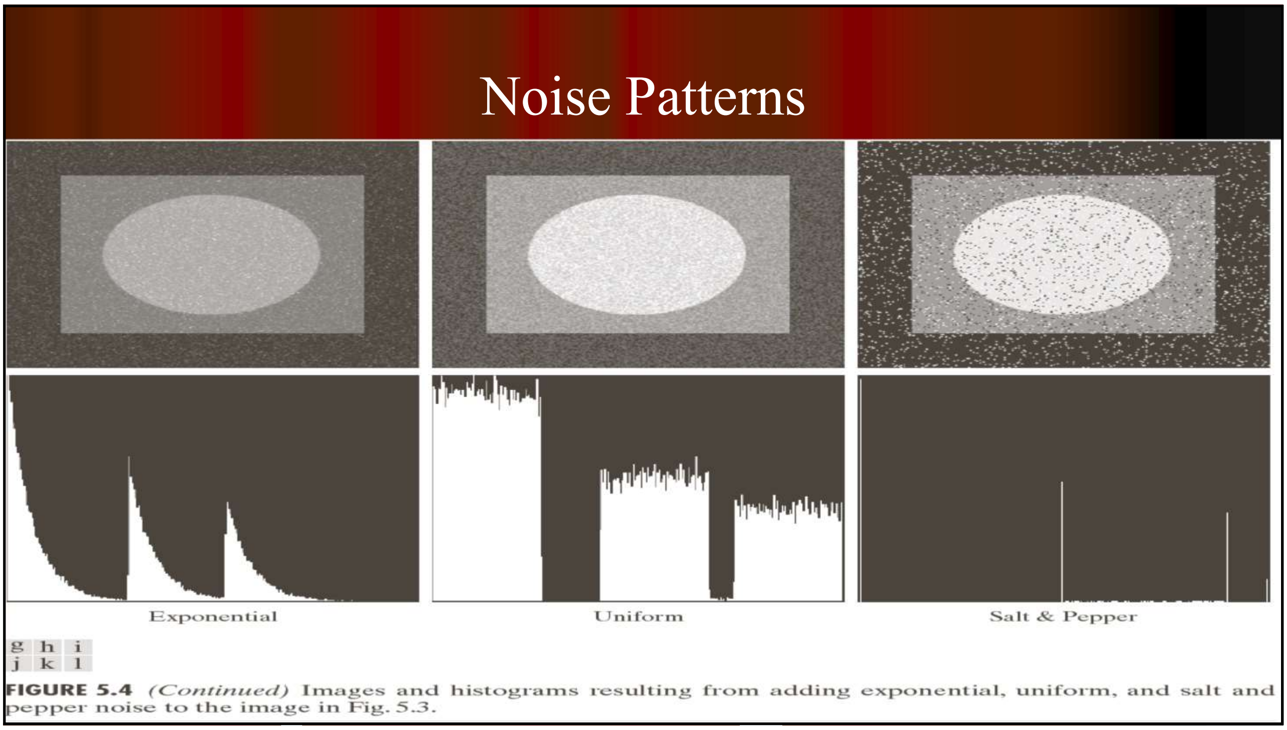
Salt & Pepper Noise

- a = imread(‘C:\lake2.bmp’);
- a = double(a);
- a = mat2gray(a);
- imhist(a);
- b = imnoise(a,’Salt & Pepper’);
- figure,imshow (b);
- figure, imhist(b)



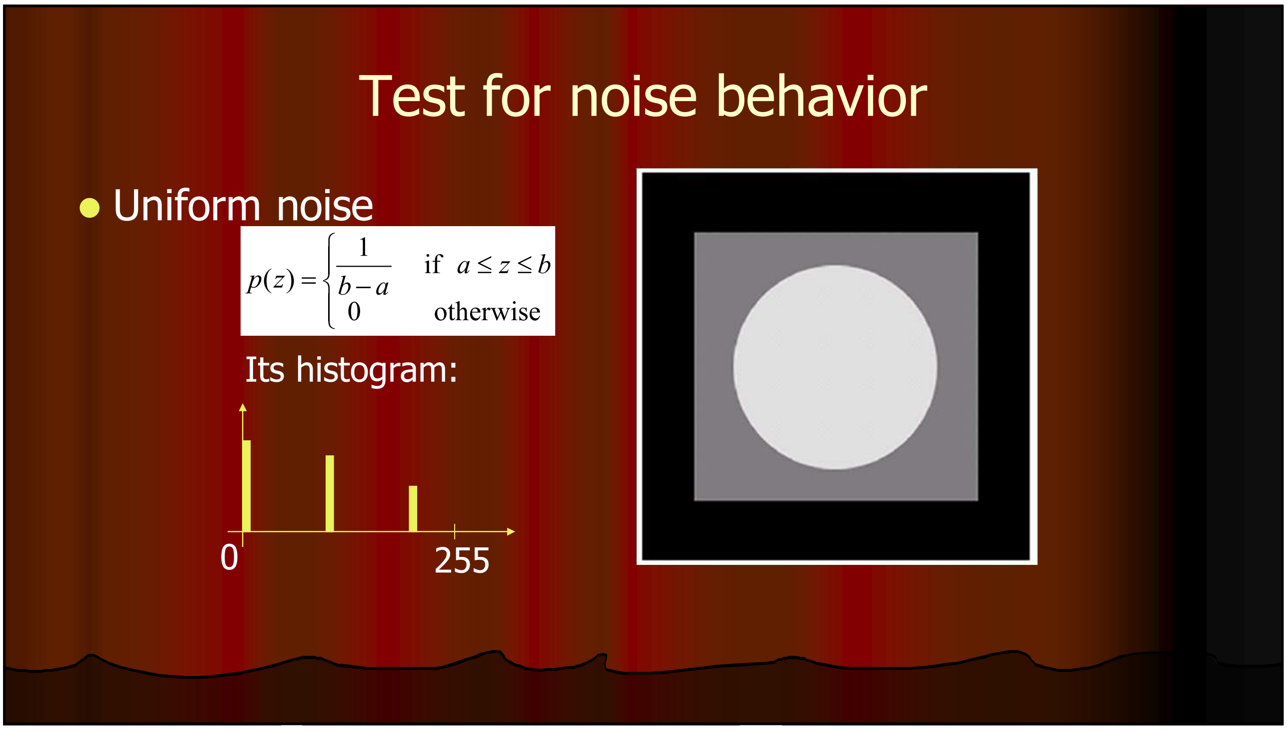
Implementation: Salt & Pepper

```
function i= saltpepper(img1,a,b)
 [m,n]=size(img1);
 img1=mat2gray(double(img1));
 r= rand(m,n);
 x=find(r <=a);
 img1(x)=0;
 x=find(r >a & r <=(a+b));
 img1(x)=255;
 figure,imhist(img1);
 figure,imshow(img1);
imwrite(img1,'C:\board_salt.tif');
return i;
```



Noise Patterns

FIGURE 5.4 *(Continued)* Images and histograms resulting from adding exponential, uniform, and salt and pepper noise to the image in Fig. 5.3.

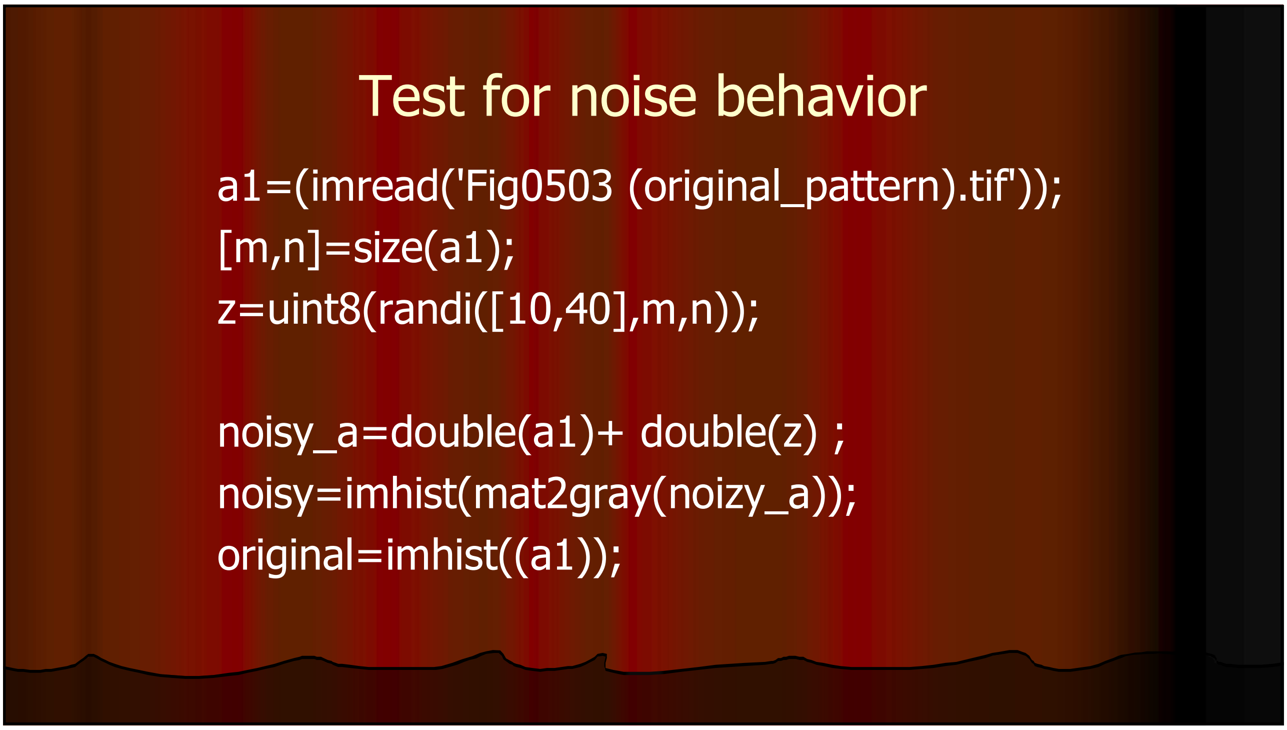


Test for noise behavior

- Uniform noise

$$p(z) = \begin{cases} \dfrac{1}{b-a} & \text{if } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$

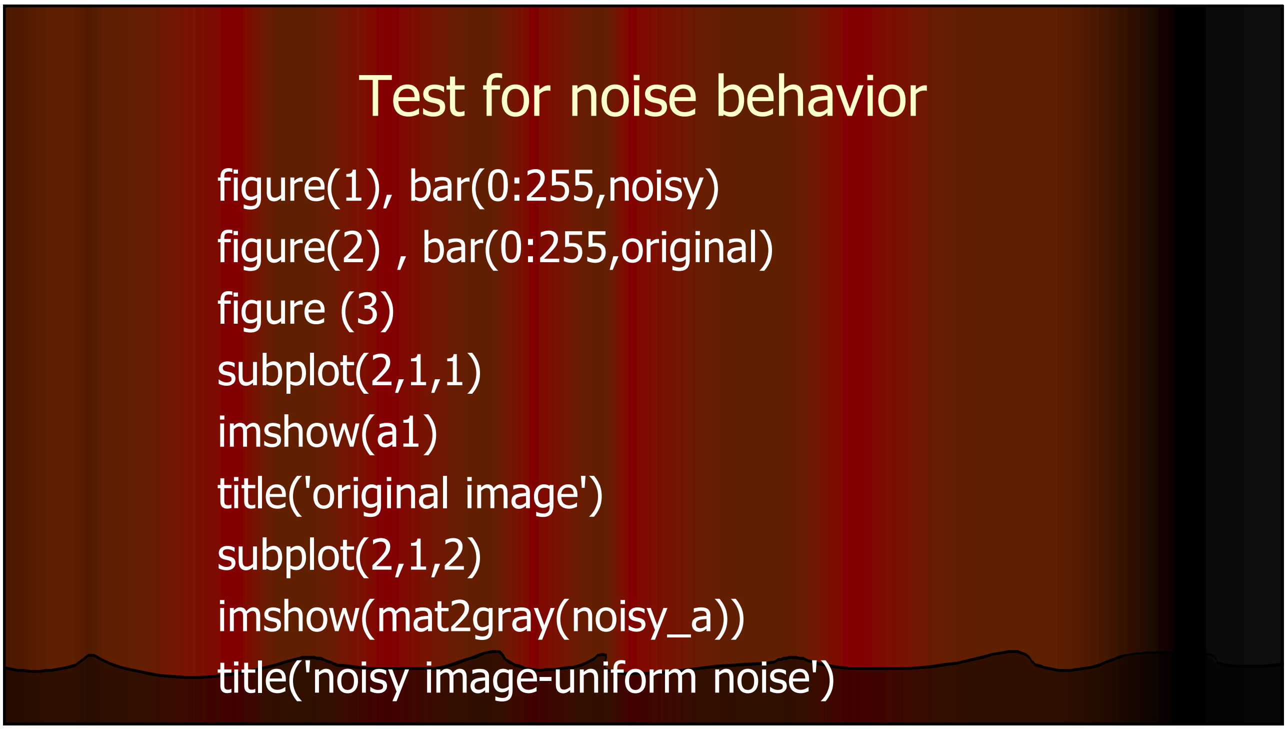
Its histogram:



Test for noise behavior

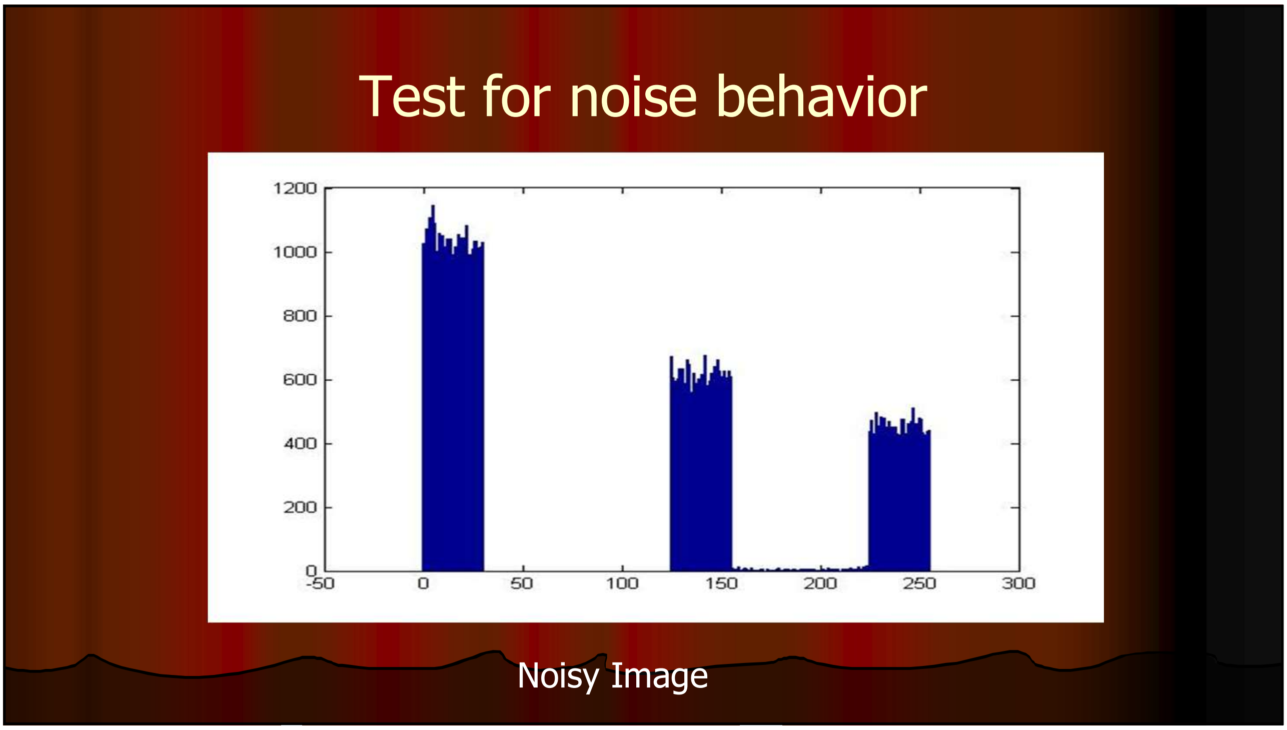
```
a1=(imread('Fig0503 (original_pattern).tif'));
[m,n]=size(a1);
z=uint8(randi([10,40],m,n));

noisy_a=double(a1)+ double(z) ;
noisy=imhist(mat2gray(noizy_a));
original=imhist((a1));
```



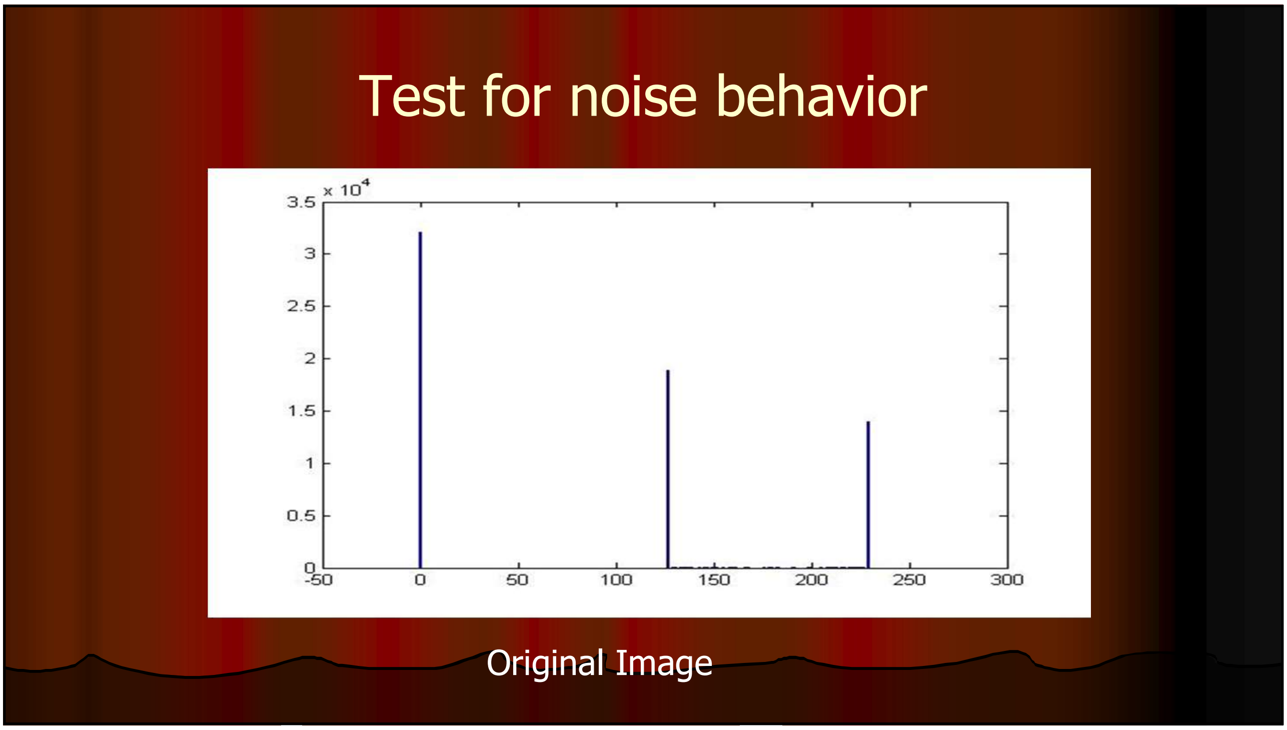
Test for noise behavior

```
figure(1), bar(0:255,noisy)
figure(2) , bar(0:255,original)
figure (3)
subplot(2,1,1)
imshow(a1)
title('original image')
subplot(2,1,2)
imshow(mat2gray(noisy_a))
title('noisy image-uniform noise')
```



Test for noise behavior

Noisy Image



Test for noise behavior

Original Image

Noise Probability Distribution (Uniform Noise)

The PDF of uniform noise is given by

$$p(z) = \begin{cases} \dfrac{1}{b-a} & \text{for } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$

The mean and variance of this density are given by

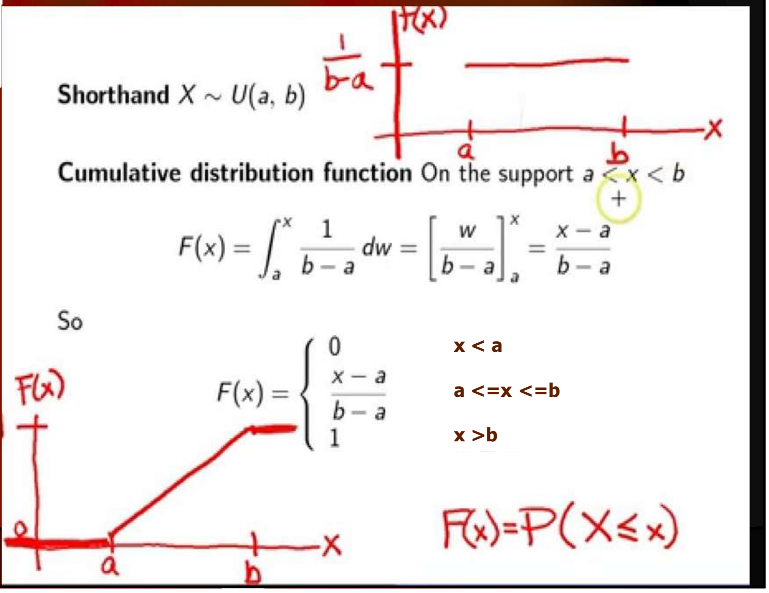
$$\bar{z} = (a+b)/2$$

$$\sigma^2 = (b-a)^2/12$$

Uniform Noise

The PDF of uniform noise is given by

$$p(z) = \begin{cases} \dfrac{1}{b-a} & \text{for } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$



Shorthand $X \sim U(a, b)$

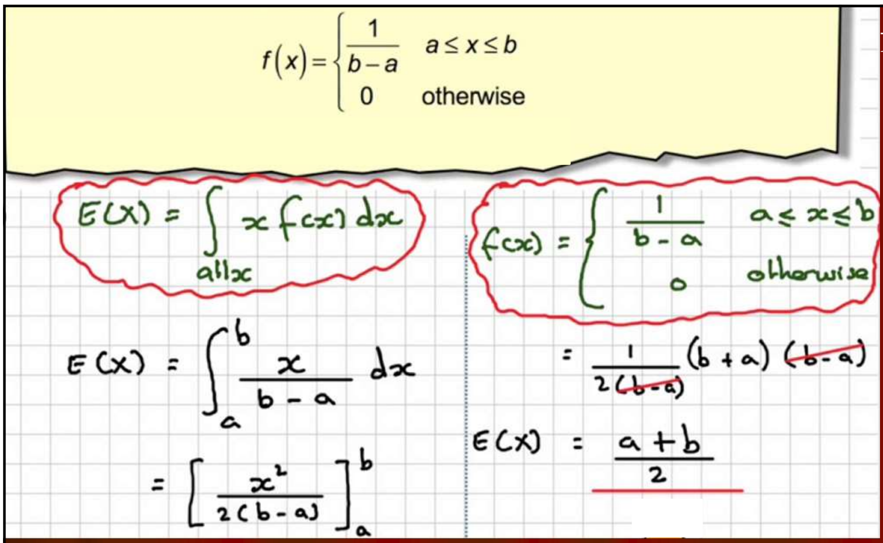
Cumulative distribution function On the support $a \le x < b$

$$F(x) = \int_a^x \frac{1}{b-a}\,dw = \left[\frac{w}{b-a}\right]_a^x = \frac{x-a}{b-a}$$

So

$$F(x) = \begin{cases} 0 & x < a \\ \dfrac{x-a}{b-a} & a <= x <= b \\ 1 & x > b \end{cases}$$

$F(x) = P(X \le x)$



$$f(x)=\begin{cases}\dfrac{1}{b-a} & a\le x\le b\\ 0 & \text{otherwise}\end{cases}$$

$$E(X) = \int_{\text{all}x} x\,f(x)\,dx$$

$$E(X) = \int_a^b \frac{x}{b-a}\,dx$$

$$= \left[\frac{x^2}{2(b-a)}\right]_a^b$$

$$f(x) = \begin{cases}\dfrac{1}{b-a} & a\le x\le b\\ 0 & \text{otherwise}\end{cases}$$

$$= \frac{1}{2\cancel{(b-a)}}(b+a)\cancel{(b-a)}$$

$$E(X) = \frac{a+b}{2}$$

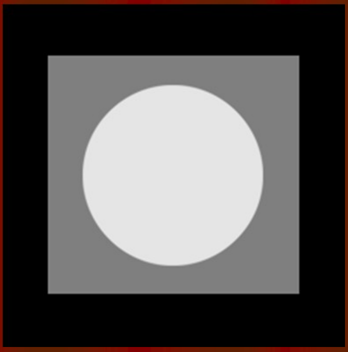
form Noise

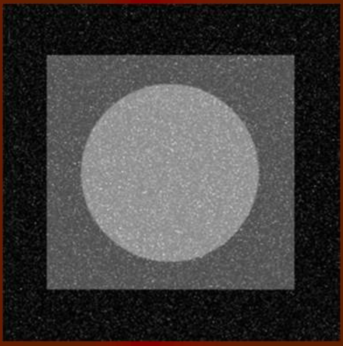
The mean and variance of this density are given by

$$\bar{z} = (a+b)/2$$

$$\sigma^2 = (b-a)^2/12$$

Exponential Noise





Noise Patterns

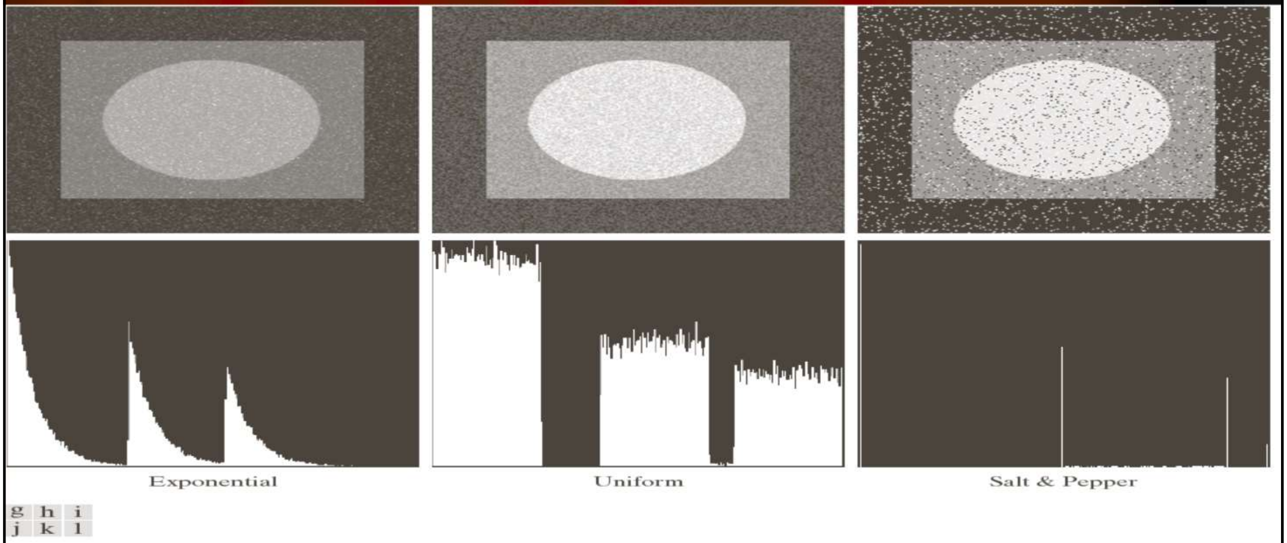


FIGURE 5.4 *(Continued)* Images and histograms resulting from adding exponential, uniform, and salt and pepper noise to the image in Fig. 5.3.

Noise Probability Distribution (Exponential Noise)

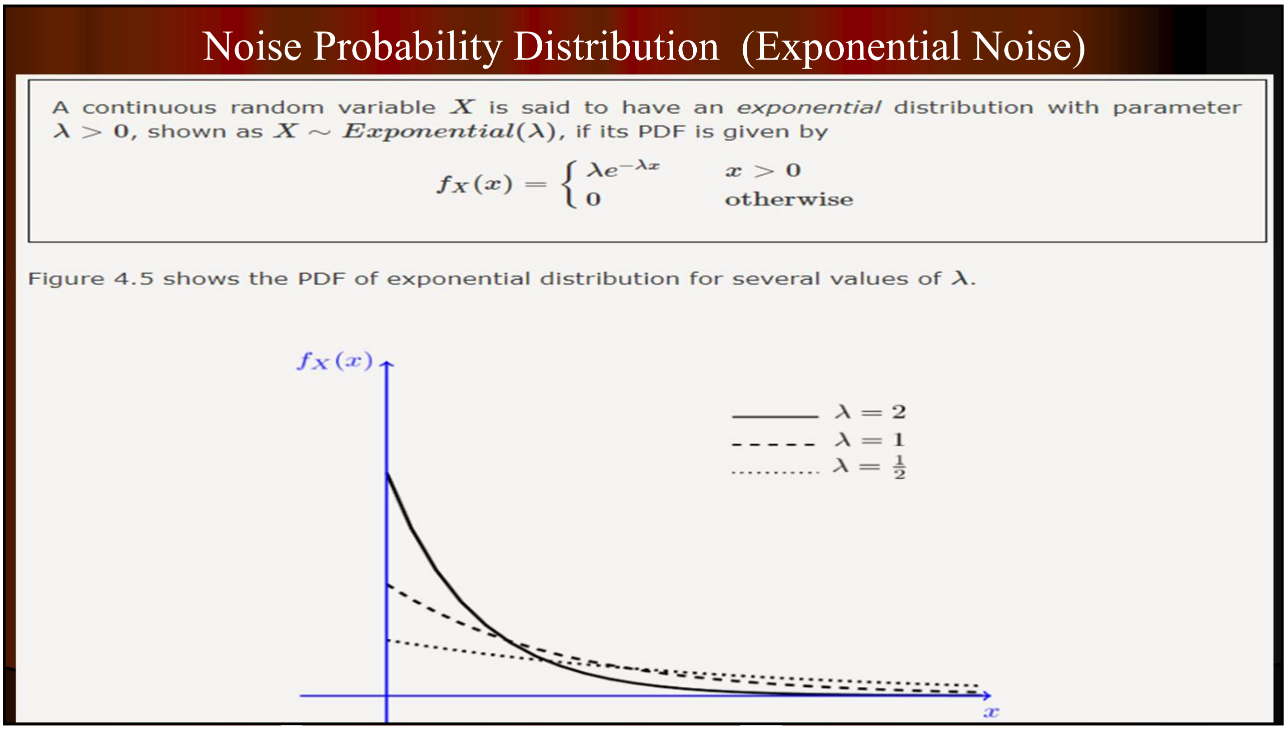
The PDF of exponential noise is given by

$$p(z) = \begin{cases} ae^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases}$$

The mean and variance of this density are given by

$$\bar{z} = 1/a$$

$$\sigma^2 = 1/a^2$$

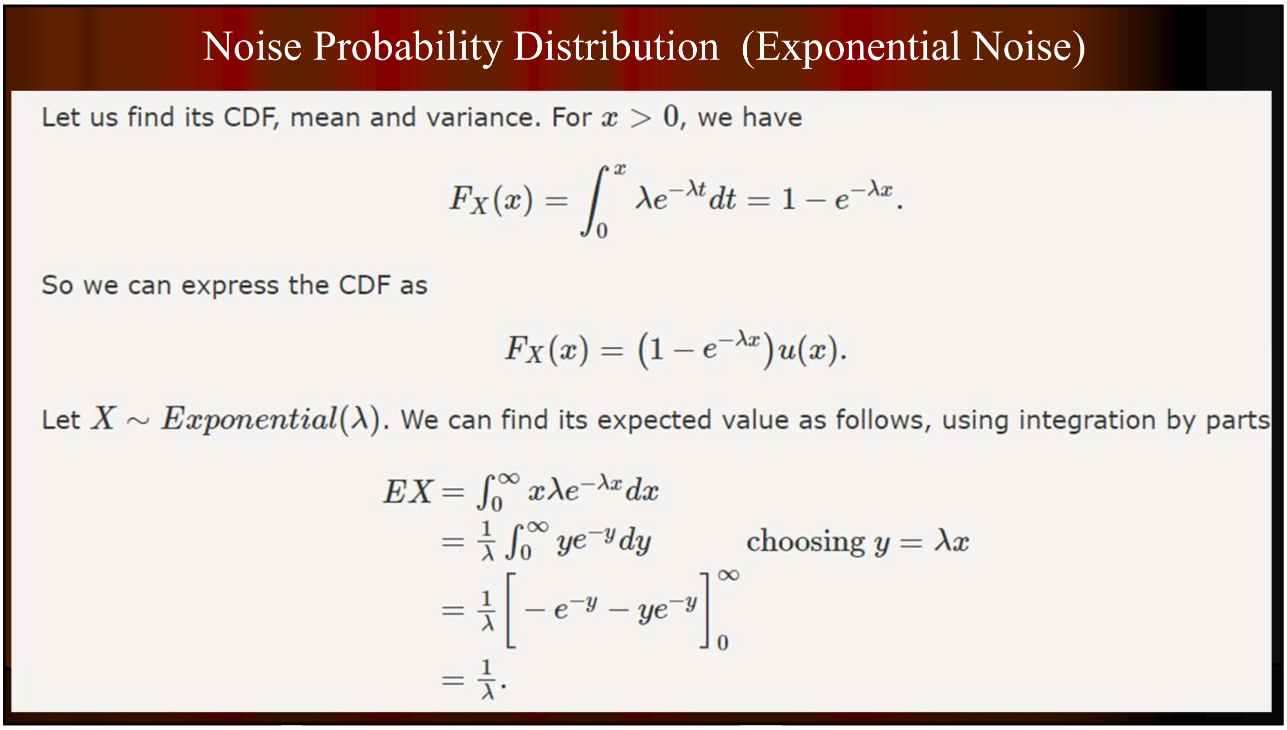


Noise Probability Distribution (Exponential Noise)

A continuous random variable X is said to have an *exponential* distribution with parameter $\lambda > 0$, shown as $X \sim Exponential(\lambda)$, if its PDF is given by

$$f_X(x) = \begin{cases} \lambda e^{-\lambda x} & x > 0 \\ 0 & \text{otherwise} \end{cases}$$

Figure 4.5 shows the PDF of exponential distribution for several values of λ.



Noise Probability Distribution (Exponential Noise)

Let us find its CDF, mean and variance. For $x > 0$, we have

$$F_X(x) = \int_0^x \lambda e^{-\lambda t} dt = 1 - e^{-\lambda x}.$$

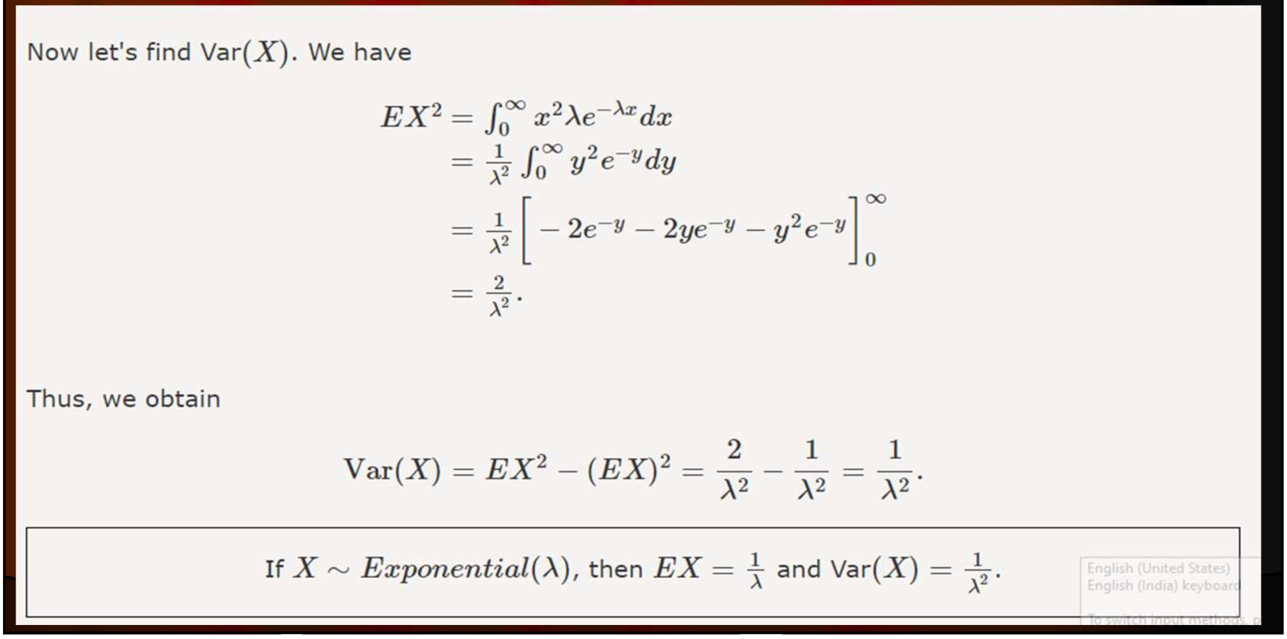
So we can express the CDF as

$$F_X(x) = \big(1 - e^{-\lambda x}\big) u(x).$$

Let $X \sim Exponential(\lambda)$. We can find its expected value as follows, using integration by parts

$$\begin{aligned} EX &= \int_0^\infty x \lambda e^{-\lambda x} dx \\ &= \frac{1}{\lambda} \int_0^\infty y e^{-y} dy \qquad \text{choosing } y = \lambda x \\ &= \frac{1}{\lambda} \Big[-e^{-y} - y e^{-y} \Big]_0^\infty \\ &= \frac{1}{\lambda}. \end{aligned}$$

Noise Probability Distribution (Exponential Noise)



Now let's find Var(X). We have

$$\begin{aligned} EX^2 &= \int_0^\infty x^2 \lambda e^{-\lambda x} dx \\ &= \frac{1}{\lambda^2} \int_0^\infty y^2 e^{-y} dy \\ &= \frac{1}{\lambda^2} \bigg[-2e^{-y} - 2ye^{-y} - y^2 e^{-y} \bigg]_0^\infty \\ &= \frac{2}{\lambda^2}. \end{aligned}$$

Thus, we obtain

$$\mathrm{Var}(X) = EX^2 - (EX)^2 = \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}.$$

If $X \sim Exponential(\lambda)$, then $EX = \frac{1}{\lambda}$ and Var$(X) = \frac{1}{\lambda^2}$.