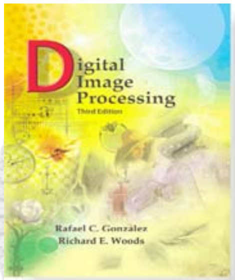
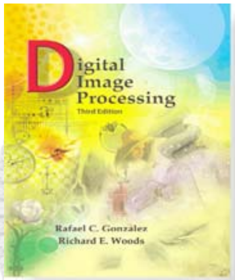


Lecture #14

- Color Gradient & Color Edges
- Noise in Color Images
- Image Degradation & Restoration
- Noise
- Noise Filters - mean, α-trimmed mean, ordering, contraharmonic

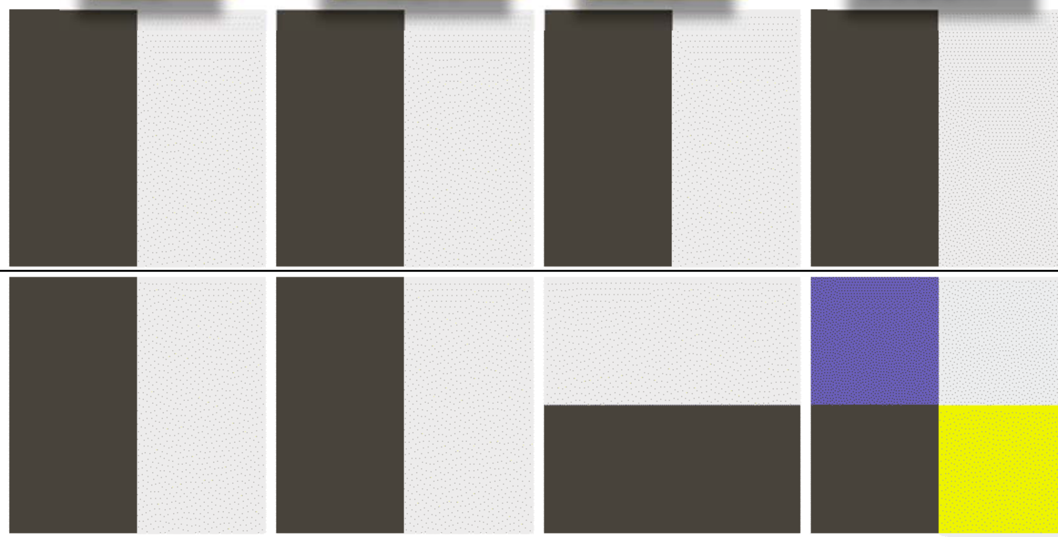


Color Edges

RED | GREEN | BLUE | COLOR

Color edges aligned

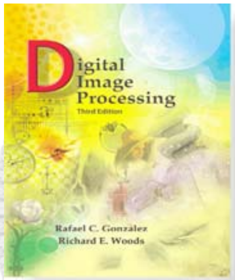
Color edges NOT aligned



If we simply added the magnitudes of the gradients they would be the same at the center. Intuitively they cannot be the same.

a	b	c	d
e	f	g	h

FIGURE 6.45 (a)–(c) *R*, *G*, and *B* component images and (d) resulting RGB color image. (e)–(g) *R*, *G*, and *B* component images and (h) resulting RGB color image.



(Vector) Color Gradient

See Section 6.6 and p. 563-564 of GWE. Implemented as colorgrad.

The maximum rate of change of a color vector c(x,y) at (x,y) is given by

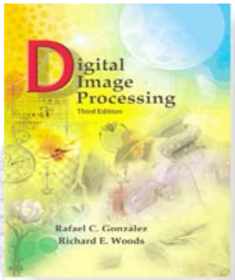
$$F(\theta)=\sqrt{\frac{1}{2}\left(g_{xx}+g_{yy}\right)+\left(g_{xx}-g_{yy}\right)\cos 2\theta+2g_{xy}\sin 2\theta}$$

in the direction

$$\theta(x,y)=\frac{1}{2}Tan^{-1}\left[\frac{2g_{xy}}{g_{xx}-g_{yy}}\right]$$

NOTE: This expression gives two directions. One is the direction of the maximum of F; the other is the direction of the minimum.

S.D.Zenzo, "A Note on the Gradient of a Multi-Image," Computer Vision, Graphics and Image Processing, Vol. 33, pp.116-125, 1986.



(Vector) Color Gradient

Let $\hat{r}, \hat{g}, \hat{b}$ be the unit vectors along the RGB axes of a RGB color space. Define

$$\vec{u} = \frac{\partial R}{\partial x}\hat{r} + \frac{\partial G}{\partial x}\hat{g} + \frac{\partial B}{\partial x}\hat{b} \qquad \vec{v} = \frac{\partial R}{\partial y}\hat{r} + \frac{\partial G}{\partial y}\hat{g} + \frac{\partial B}{\partial y}\hat{b}$$

Further define

$$g_{xx} = \vec{u} \bullet \vec{u} = \vec{u}^T \vec{u} = \left|\frac{\partial R}{\partial x}\right|^2 + \left|\frac{\partial G}{\partial x}\right|^2 + \left|\frac{\partial B}{\partial x}\right|^2$$

$$g_{yy} = \vec{v} \bullet \vec{v} = \vec{v}^T \vec{v} = \left|\frac{\partial R}{\partial y}\right|^2 + \left|\frac{\partial G}{\partial y}\right|^2 + \left|\frac{\partial B}{\partial y}\right|^2$$

$$g_{xy} = \vec{u} \bullet \vec{v} = \vec{u}^T \vec{v} = \frac{\partial R}{\partial x}\frac{\partial R}{\partial y} + \frac{\partial G}{\partial x}\frac{\partial G}{\partial y} + \frac{\partial B}{\partial x}\frac{\partial B}{\partial y}$$

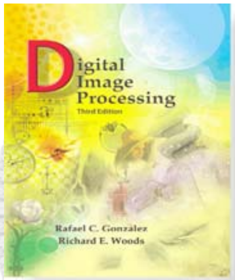
Color Edges

FIGURE 6.46 (a) RGB image. (b) Gradient computed in RGB color vector space. (c) Gradients computed on a per-image basis and then added. (d) Difference between (b) and (c).

Color Edges

a b c

FIGURE 6.47 Component gradient images of the color image in Fig. 6.46. (a) Red component, (b) green component, and (c) blue component. These three images were added and scaled to produce the image in Fig. 6.46(c).



Noise in RGB Color Image

Additive Gaussian noise (mean=0, variance=800) in EACH color component

Noisy RED

Noisy GREEN

Noisy BLUE

Noisy RGB image

a b
c d

FIGURE 6.48 (a)–(c) Red, green, and blue component images corrupted by additive Gaussian noise of mean 0 and variance 800. (d) Resulting RGB image. [Compare (d) with Fig. 6.46(a).]

Noise in HSI Color Image

Significantly degraded due to non-linearity of cosine and min in HSI transformations

I looks clean since 1/3*(R+G+B) averages noise

a b c

FIGURE 6.49 HSI components of the noisy color image in Fig. 6.48(d). (a) Hue. (b) Saturation. (c) Intensity.

Noise in Color Image

RGB image with salt&pepper noise in green component

HUE

SATURATION

INTENSITY

Noise spreads to all HSI components.

a b
c d

FIGURE 6.50 (a) RGB image with green plane corrupted by salt-and-pepper noise. (b) Hue component of HSI image. (c) Saturation component. (d) Intensity component.

Noise in Compression

Original image

a
b

FIGURE 6.51 Color image compression. (a) Original RGB image. (b) Result of compressing and decompressing the image in (a).

Original image compressed and decompressed using JPEG 2000. Slight blurring due to loss inherent in compression technique.

JPEG 2000 uses a color transformation similar to that used by the Y'CbCr (luminance-blue chroma-red chroma) color TV standard and wavelet compression.

Image Restoration

Additive noise

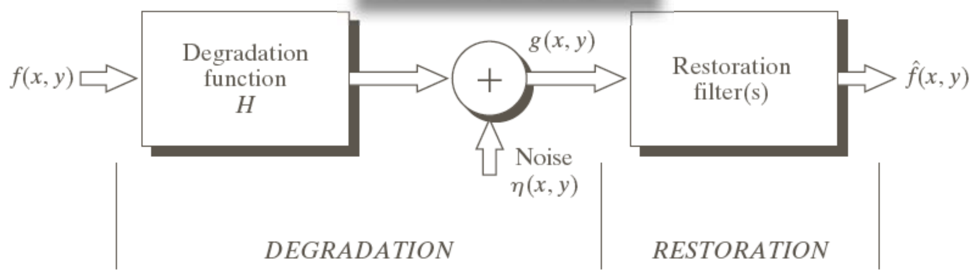


FIGURE 5.1 A model of the image degradation/restoration process.

Limits us to certain types of degradation

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

$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$

Noise PDFs

Used for skewed histograms

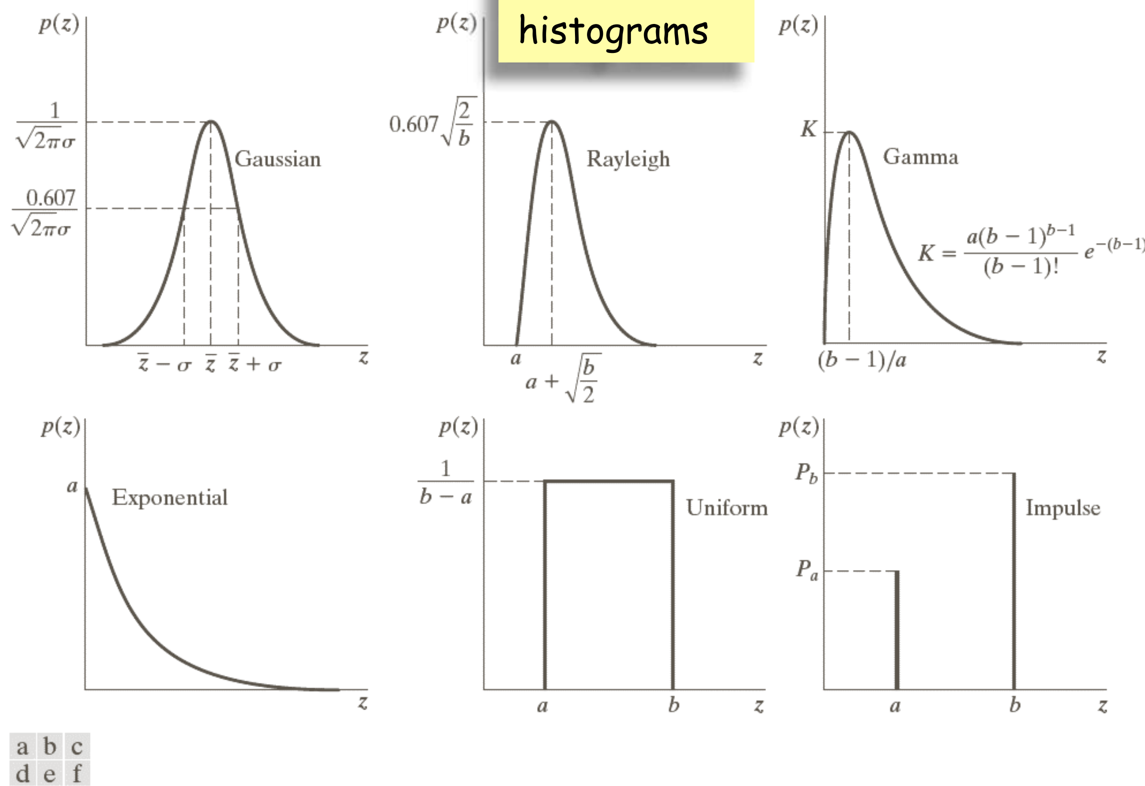
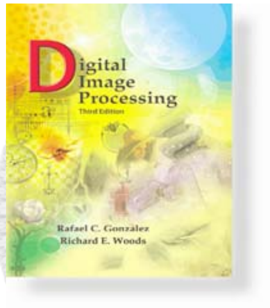


FIGURE 5.2 Some important probability density functions.



Noise PDFs

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$$

Gaussian

$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-\frac{(z-a)^2}{b}} & z \geq a \\ 0 & z < a \end{cases}$$

Rayleigh

$$p(z) = \begin{cases} \frac{a^b z^{b+1}}{(b-1)!}(z-a)e^{-az} & z \geq 0 \\ 0 & z < 0 \end{cases}$$

Gamma

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

Exponential

$$p(z) = \begin{cases} \frac{1}{b-a} & a \leq z \leq b \\ 0 & otherwise \end{cases}$$

Uniform (white)

$$p(z) = \begin{cases} p_a & z = a \\ p_b & z = b \\ 0 & otherwise \end{cases}$$

Impulse (salt & pepper)

Test Image

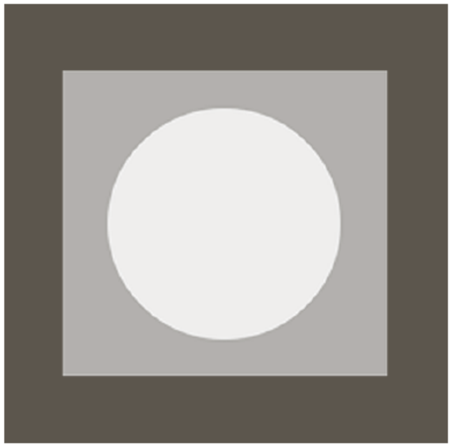
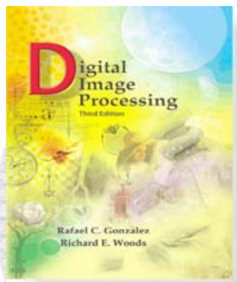


FIGURE 5.3 Test pattern used to illustrate the characteristics of the noise PDFs shown in Fig. 5.2.

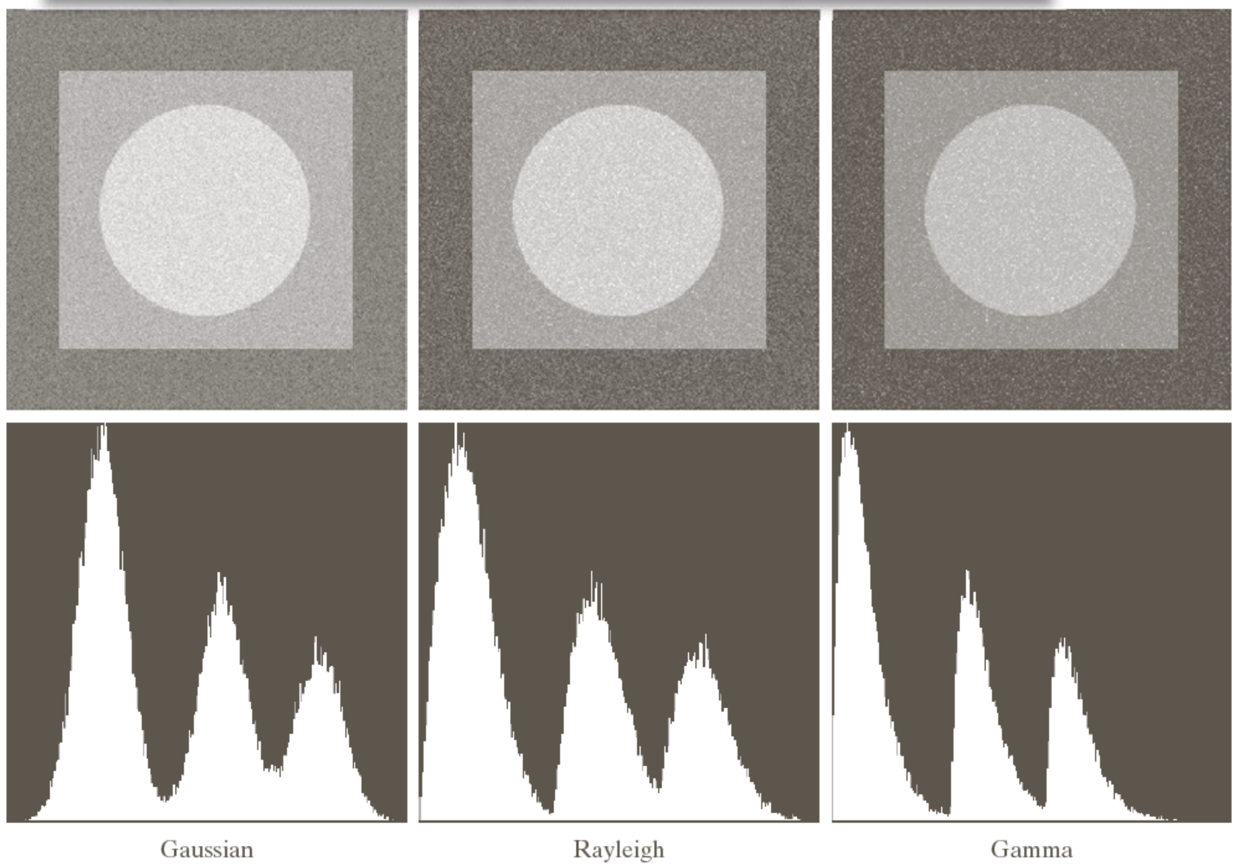


Noisy Images

Usually hard to identify type of noise in spatial domain

Test Image + Noise

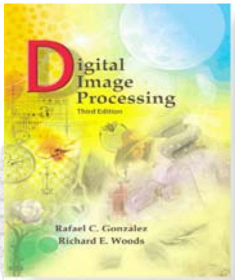
pdf



a b c
d e f

Noise spectra adjusted so they just overlap

FIGURE 5.4 Images and histograms resulting from adding Gaussian, Rayleigh, and gamma noise to the image in Fig. 5.3.



Noisy Images

Test Image + Noise

pdf

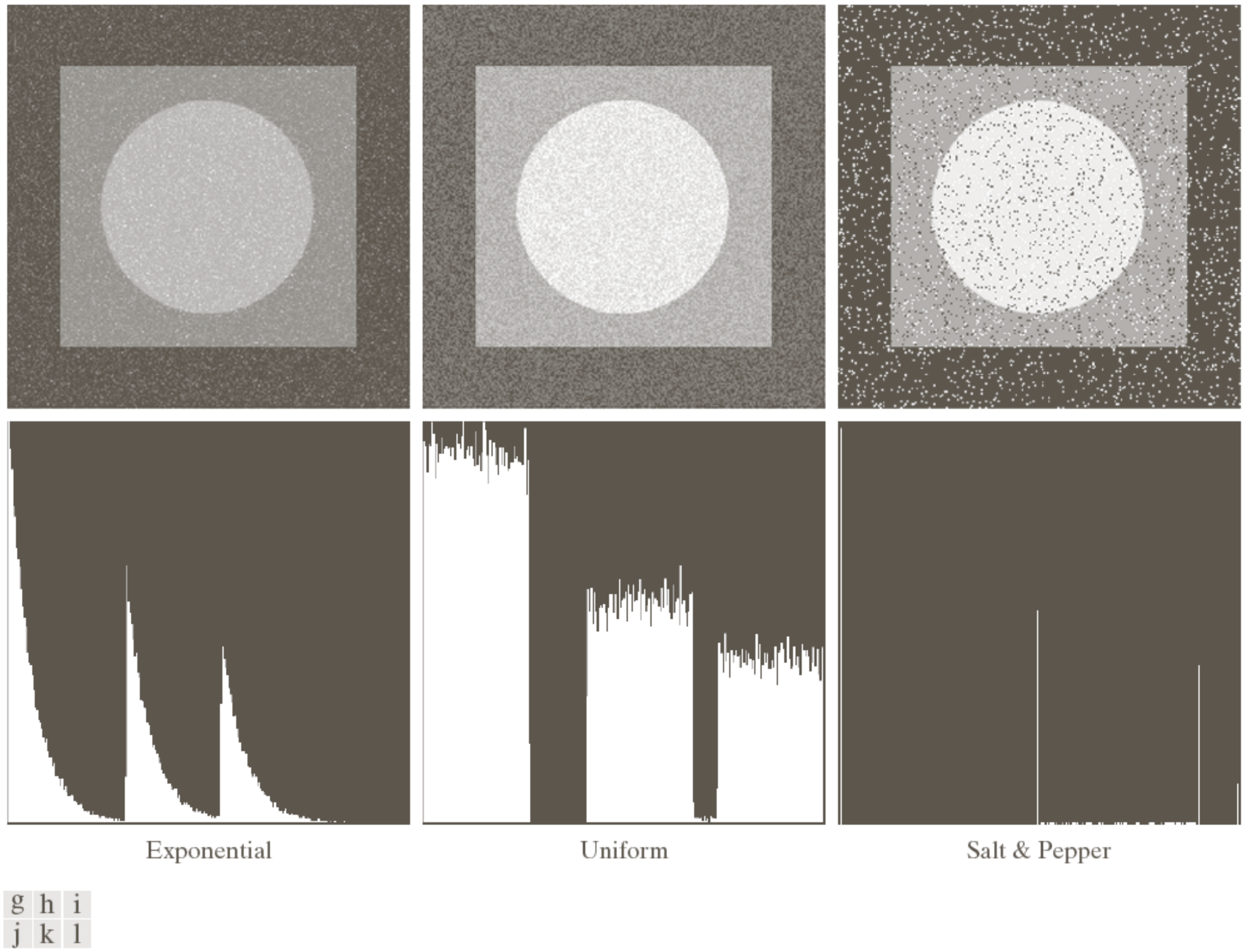
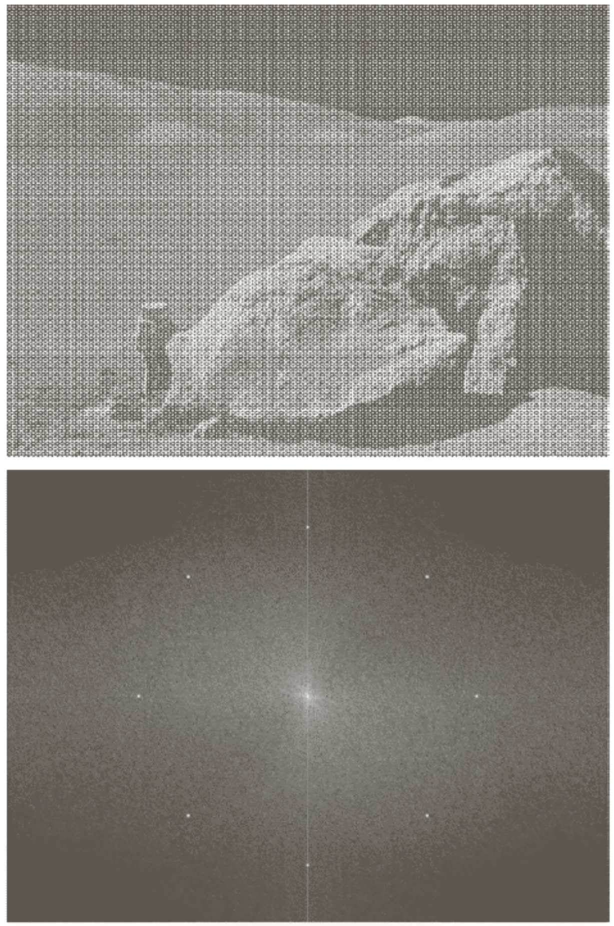


FIGURE 5.4 *(Continued)* Images and histograms resulting from adding exponential, uniform, and salt and pepper noise to the image in Fig. 5.3.

Noisy Images

Note regular pattern of noise in image.



a
b

FIGURE 5.5 (a) Image corrupted by sinusoidal noise. (b) Spectrum (each pair of conjugate impulses corresponds to one sine wave). (Original image courtesy of NASA.)

Noise Characterization

Test strips

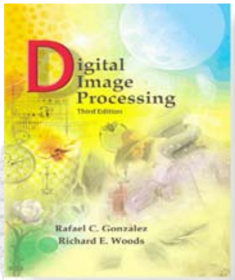
a b c

FIGURE 5.6 Histograms computed using small strips (shown as inserts) from (a) the Gaussian, (b) the Rayleigh, and (c) the uniform noisy images in Fig. 5.4.

Compute mean and variance for histograms and relate them to distribution parameters.

$$\mu_z = \sum_{z_i} z_i p(z_i) \qquad \sigma_z = \sum_{z_i} (z_i - \mu_z)^2 p(z_i)$$

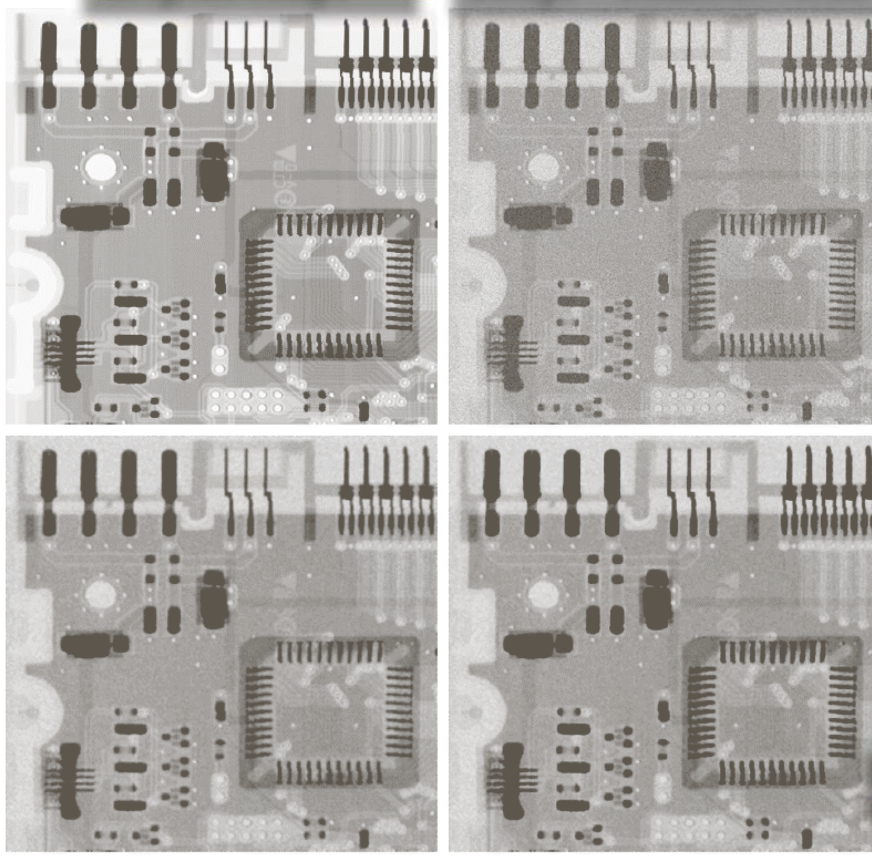
z_i is the gray level and $p(z_i)$ is the histogram



Mean Filters

Original image

Image + Gaussian noise



a b
c d

FIGURE 5.7 (a) X-ray image. (b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size 3 × 3. (d) Result of filtering with a geometric mean filter of the same size. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

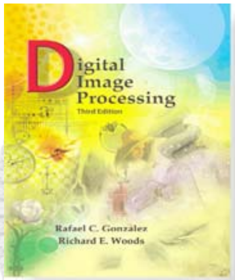
Geometric mean filter gives smoothing comparable to arithmetic mean filter without losing as much detail.

Arithmetic mean filter

Geometric mean filter

$$\hat{f}(x,y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} g(s,t)$$

$$\hat{f}(x,y) = \prod_{(s,t)\in S_{xy}} g(s,t)^{\frac{1}{mn}}$$



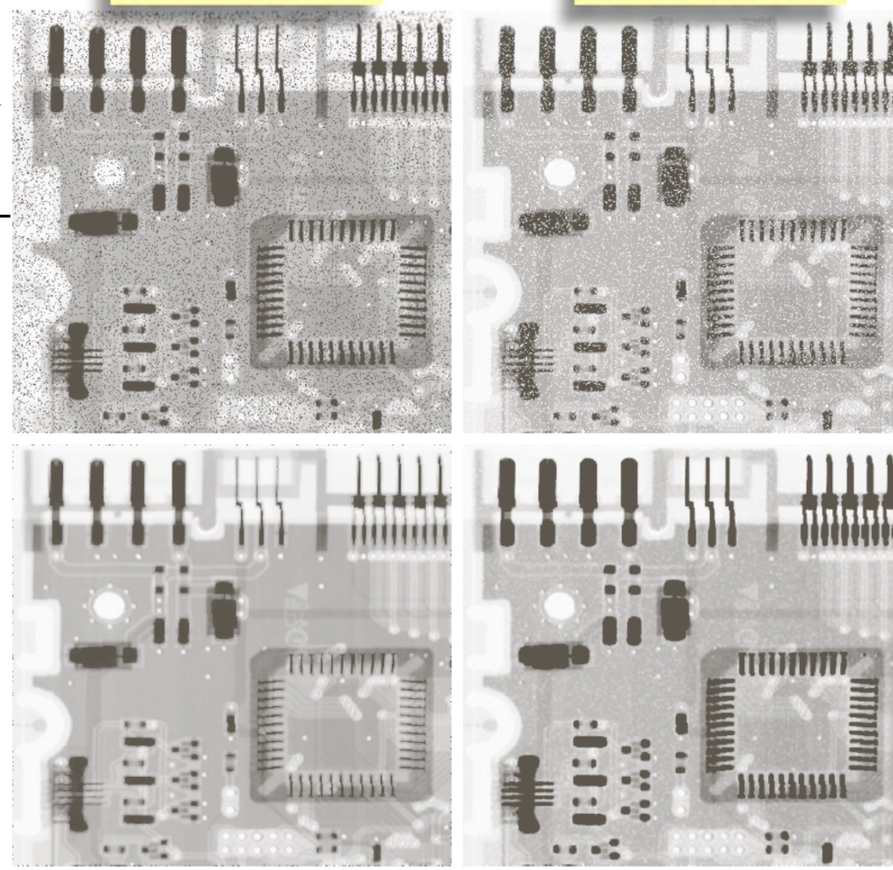
Contraharmonic Filters

$$\hat{f}(x,y)=\frac{\sum_{(s,t)\in S_{xy}}\left[g(s,t)\right]^{Q+1}}{\sum_{(s,t)\in S_{xy}}\left[g(s,t)\right]^{Q}}$$

Contraharmonic filter reduces to arithmetic mean filter if Q=1

Pepper noise (random 0's)

Salt noise (random 1's)



a b
c d

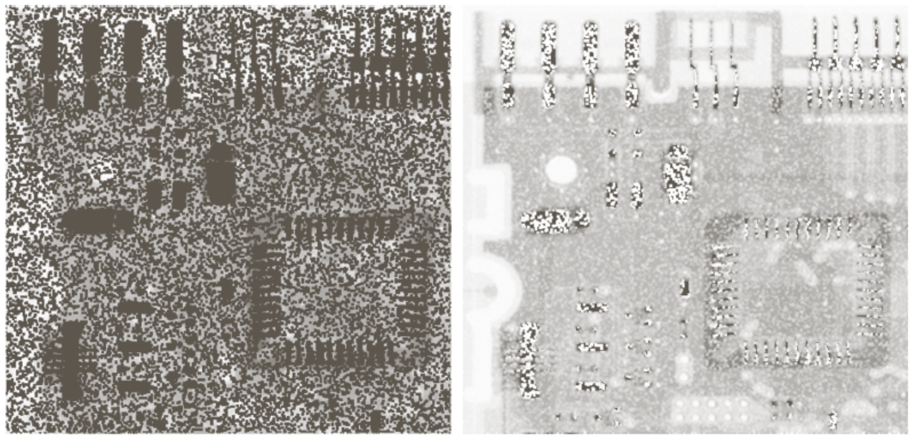
FIGURE 5.8 (a) Image corrupted by pepper noise with a probability of 0.1. (b) Image corrupted by salt noise with the same probability. (c) Result of filtering (a) with a 3 × 3 contraharmonic filter of order 1.5. (d) Result of filtering (b) with $Q = -1.5$.

Contraharmonic (Q=1.5) eliminates pepper noise

Contraharmonic (Q=-1.5) eliminates salt noise

Contraharmonic filter can reduce salt or pepper noise but not both at the same time.

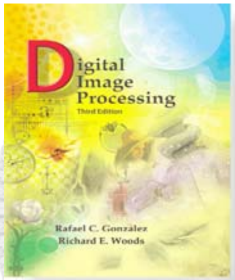
Contraharmonic Filters



a b

FIGURE 5.9 Results of selecting the wrong sign in contraharmonic filtering. (a) Result of filtering Fig. 5.8(a) with a contraharmonic filter of size 3×3 and $Q = -1.5$. (b) Result of filtering 5.8(b) with $Q = 1.5$.

Using the wrong sign in a contraharmonic filter will significantly degrade the image.

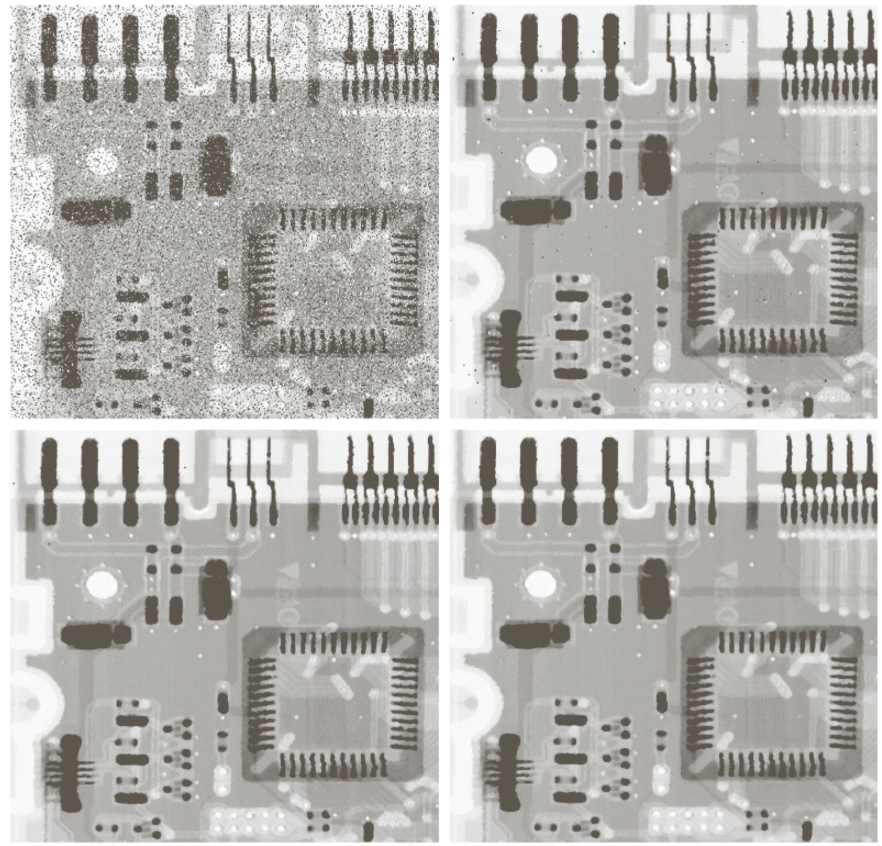


Order Statistics Filter

Order statistics filters are based upon ordering (ranking) pixels in a neighborhood.

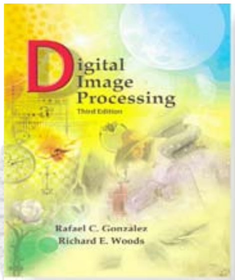
The median filter selects the middle element in the list.

$$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{median}\left\{g(s,t)\right\}$$



a b
c d

FIGURE 5.10 (a) Image corrupted by salt-and-pepper noise with probabilities $P_a = P_b = 0.1$. (b) Result of one pass with a median filter of size 3×3. (c) Result of processing (b) with this filter. (d) Result of processing (c) with the same filter.



Order Statistics Filters

Other order statistics filters

Median works best for impulse noise.

$$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{median}\{g(s,t)\}$$

Max works best for finding bright points.

$$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}}\{g(s,t)\}$$

Min works best for finding dark points.

$$\hat{f}(x,y) = \min_{(s,t)\in S_{xy}}\{g(s,t)\}$$

Midpoint works best for Gaussian or uniform noise

$$\hat{f}(x,y) = \frac{1}{2}\left[\min_{(s,t)\in S_{xy}}\{g(s,t)\} + \max_{(s,t)\in S_{xy}}\{g(s,t)\}\right]$$

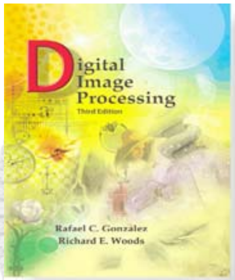
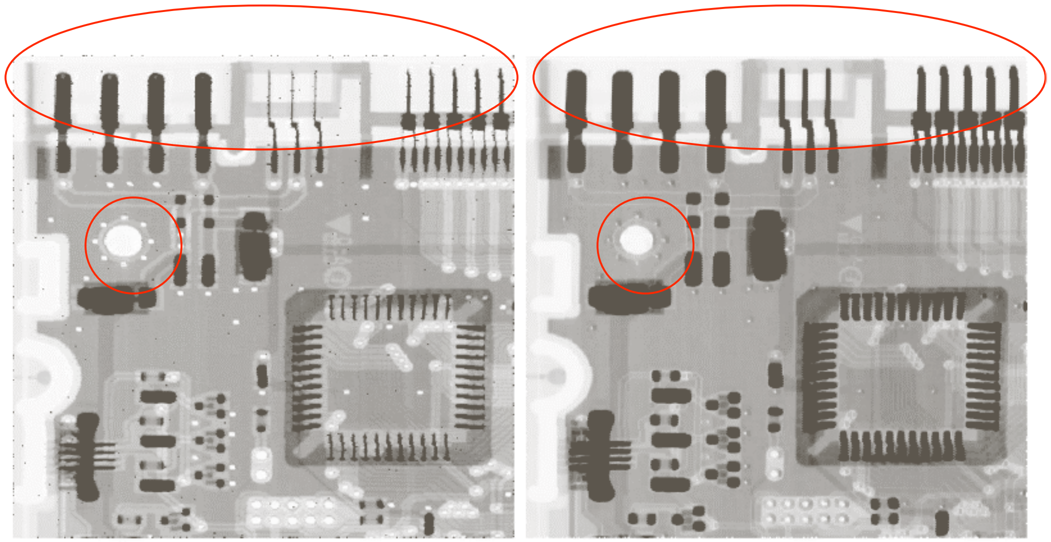


Image Restoration

a b

FIGURE 5.11 (a) Result of filtering Fig. 5.8(a) with a max filter of size 3 × 3. (b) Result of filtering 5.8(b) with a min filter of the same size.



Note differences between images

Max filter

Min filter

Max and min filters do a reasonable job of removing impulsive noise but:

- Max filter removed dark pixels from the borders of dark objects
- Min filter removed light pixels from the borders of light objects

Yet Another Order Statistics Filter

Alpha-trimmed mean filter

removes the d/2 highest and d/2 lowest intensity values. The average of these remaining mn-d values is called an alpha-trimmed mean filter.

$$\hat{f}(x,y) = \frac{1}{mn-d} \sum_{(s,t)\in S_{xy}} g_r(s,t)$$

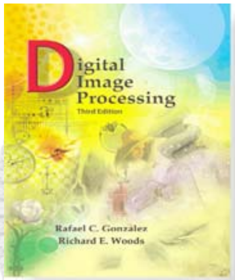


Image Restoration

Image
+white noise

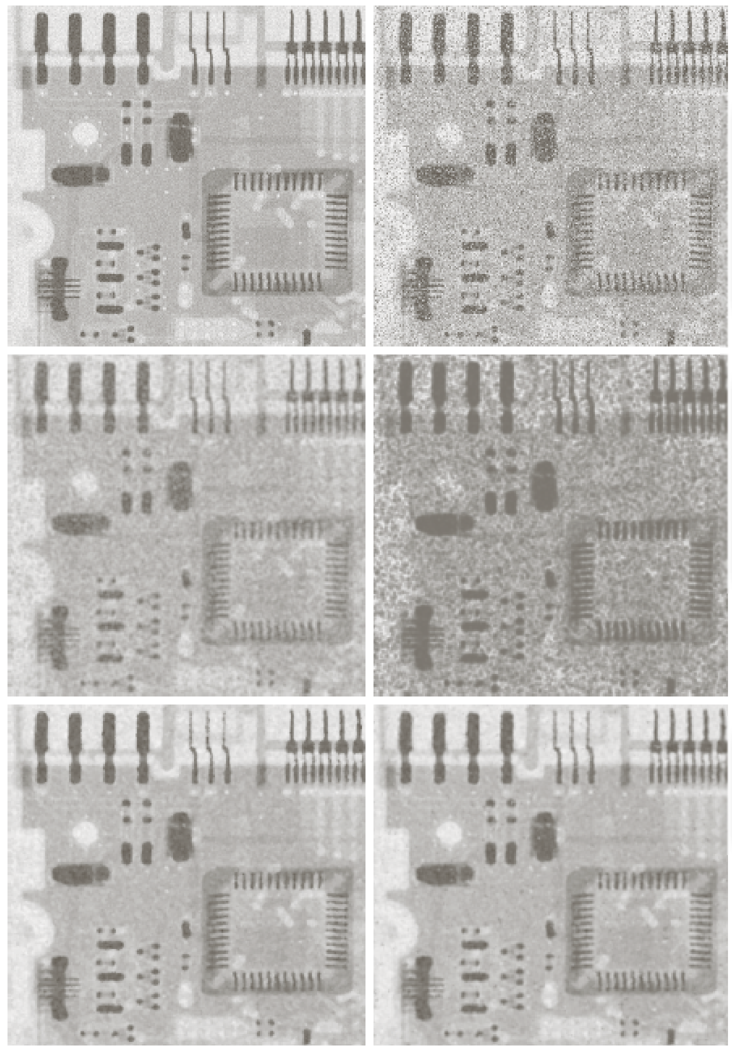
Image
+white noise
+salt&pepper
noise

5x5
arithmetic
mean
filter

5x5
geometric
mean
filter

5x5
median
filter

5x5 alpha-trimmed
(d=5) mean filter



a b
c d
e f

FIGURE 5.12 (a) Image corrupted by additive uniform noise. (b) Image additionally corrupted by additive salt-and-pepper noise. Image (b) filtered with a 5×5; (c) arithmetic mean filter; (d) geometric mean filter; (e) median filter; and (f) alpha-trimmed mean filter with $d = 5$.

Approaches performance of median filter as d increases but also smooths