Lecture 7 Noise Removal & Contrast Enhancement

Guoxu Liu

Weifang University of Science and Technology

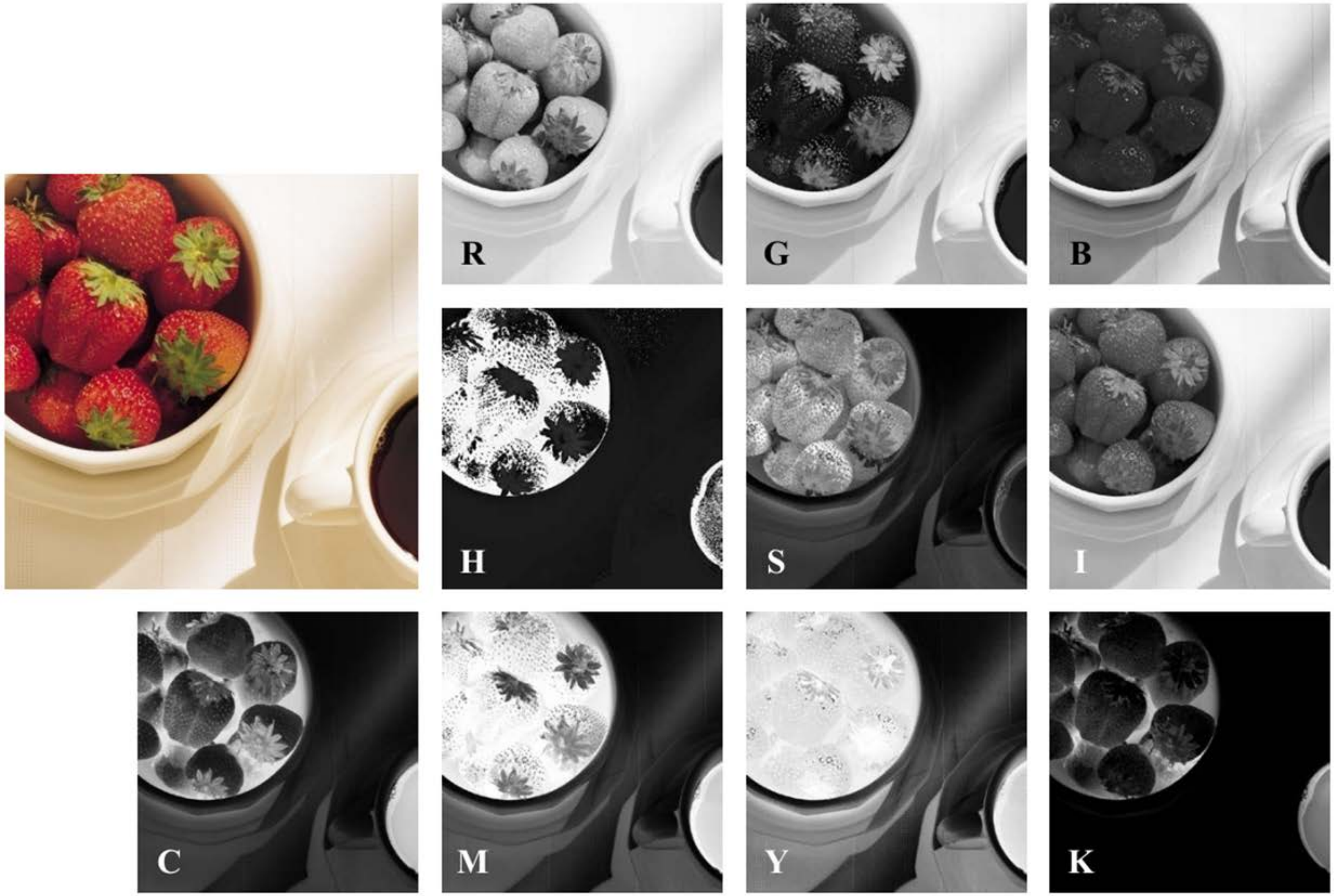
liuguoxu@wfust.edu.cn

November 13, 2020

Outline

- Noise Removal
  - Noise Models
  - Binomial Smoothing
  - Gaussian Smoothing
  - Median Filtering, etc
- Contrast Enhancement
  - Windowing
  - Histogram Equalization
  - Unsharp Masking
  - Contrast Variance Enhancement, etc

Color Channel Images

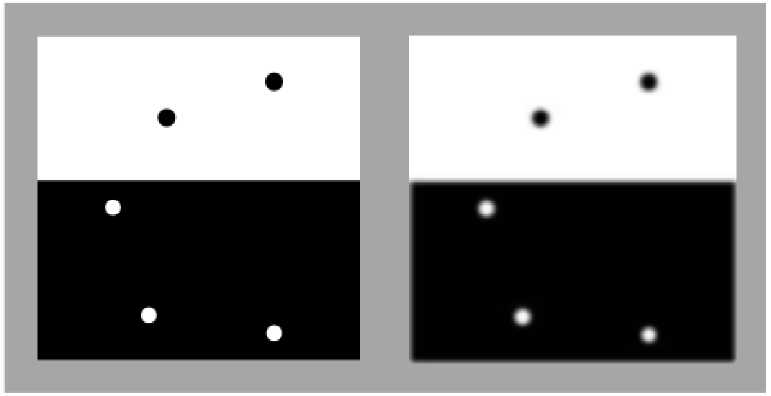


Introduction

Noise Removal

- Detect & remove unwanted noise
- Difficulty: what is noise? where is?
- Assumption: smoothness of intensity and color
- Local Averaging: common method of replacing anomalous pixels with values derived from nearby pixels
- Side Effect: blur output images

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$



❑ Contrast Enhancement

- To increase visibility of features of interest by amplifying local variations in color or intensity
- Side effect: amplifying also noise typically

❑ Challenges in Color Images

- Should retain chromatic information, particularly hue
- Approaches: manipulate only image intensity or work directly with vector-valued pixels

Noise Removal

❑ Noise Models

- Additive noise model: the simplest one

$$g(x,y) = f(x,y) + n(x,y)$$

$g(x,y)$: observed image, $f(x,y)$: true image, $n(x,y)$: noise image

- $n(x,y)$ can be modeled
  - ✓ Gaussian distribution with zero mean
  - ✓ uniform, Poisson, or exponential distribution
  - ✓ impulse noise
- In RGB space, $g_r(x,y) = f_r(x,y) + n_r(x,y)$

$$g_g(x,y) = f_g(x,y) + n_g(x,y)$$

$$g_b(x,y) = f_b(x,y) + n_b(x,y)$$

❑ Signal-to-Noise Ratio

- To characterize noise in view of how additive noise is perceived
- SNR In decibels (dB)

$$SNR = 10\log_{10}\left(\frac{\sigma_f^2}{\sigma_n^2}\right)$$

σ_f^2, σ_n^2 : variance of image and noise, respectively

- PSNR: Peak SNR
  - ✓ The variance of image can be approximated by squaring the range of intensities

$$PSNR = 10\log_{10}\left(\frac{L^2}{\sigma_n^2}\right)$$

❑ Temporal Smoothing

- Averaging K independent observations of an image $f(x,y)$, corrupted with zero mean Gaussian noise with variance σ_n^2

$$g_k(x,y) = f(x,y) + n_k(x,y)$$

$$\bar{g}(x,y) = \frac{1}{K}\sum_{k=1}^{K} g_k(x,y) = f(x,y) + \bar{n}(x,y)$$

$$\text{where } \bar{n}(x,y) = \frac{1}{K}\sum_{k=1}^{K} n_k(x,y)$$

$$\boxed{\sigma_{\bar{n}}^2 = \frac{1}{K}\sigma_n^2} \quad \Rightarrow \quad SNR_{\bar{g}} = SNR_g + 10\log_{10} K$$

$$SNR_{\bar{g}} = 10\log_{10}\left(\frac{\sigma_f^2}{\sigma_n^2 / K}\right) = 10\log_{10}\left(\frac{\sigma_f^2}{\sigma_n^2}\right) + 10\log_{10} K = SNR_g + 10\log_{10} K$$

$$n_i\,,\ i = 1,\ldots,K \quad \textit{independent identically distributed random variables}$$

$$E[n_i] = \mu_n,\ \ Var(n_i) = E[(n_i - \mu_n)^2] = \sigma_n^2, \quad i = 1,\ldots,K$$

$$n_i\text{'}\textit{s are uncorrelated since, for } i \neq j,$$

$$E[(n_i - \mu_n)(n_j - \mu_n)] = E[n_i]E[n_j] - \mu_n(E[n_i] + E[n_j]) + \mu_n^2 = 0$$

$$\textit{Another random variable } \bar{n} = (1/K)\{n_1 + \cdots + n_K\}$$

$$E[\bar{n}] = E[(1/K)\{n_1 + \cdots + n_K\}] = (1/K)\{E[n_1] + \cdots + E[n_K]\} = (1/K)\{K\mu_n\} = \mu_n$$

$$\begin{aligned}
Var(\bar{n}) &= E[(\bar{n} - \mu_n)^2] = E[\{(1/K)(n_1 + \cdots + n_K) - \mu_n\}^2] \\
&= E[(1/K^2)\{(n_1 - \mu_n) + \cdots + (n_K - \mu_n)\}^2] \\
&= (1/K^2)\{E[(n_1 - \mu_n)^2] + \cdots + E[(n_K - \mu_n)^2]\} \quad \because \textit{uncorrelated} \\
&= (1/K^2)\{Var(n_1) + \cdots + Var(n_K)\} = (1/K^2)\{K\sigma_n^2\} = (1/K)\sigma_n^2
\end{aligned}$$

Noise Removal

or

$$
\begin{aligned}
Var(\overline{n}) &= E[(\overline{n}-\mu_n)^2] = E[\overline{n}^2]-\mu_n^2 = E[\{(1/K^2)(n_1+\cdots+n_K)^2]-\mu_n^2 \\
&= (1/K^2)E[(n_1+\cdots+n_K)^2]-\mu_n^2 \\
&= (1/K^2)\left\{\sum_{i=1}^{K} E[n_i^2]+2\sum_{i=1}^{K-1}\sum_{j=i+1}^{K} E[n_i]E[n_j]\right\}-\mu_n^2 \\
&= (1/K^2)\left\{\sum_{i=1}^{K} E[n_i^2]+2\sum_{i=1}^{K-1}\sum_{j=i+1}^{K} \mu_n^2\right\}-\mu_n^2 \\
&= (1/K^2)\sum_{i=1}^{K} E[n_i^2]+(1/K^2)2\frac{K(K-1)}{2}\mu_n^2-\mu_n^2 \\
&= (1/K^2)\sum_{i=1}^{K} E[n_i^2]-(1/K)\mu_n^2 \\
&= (1/K^2)\sum_{i=1}^{K}(E[n_i^2]-\mu_n^2) = (1/K^2)\sum_{i=1}^{K} E[(n_i-\mu_n)^2] \\
&= (1/K^2)\{K\sigma_n^2\} = (1/K)\sigma_n^2
\end{aligned}
$$

❑ Spatial Smoothing

- Applicable to when separate observations are not available
- Convolution

$$\bar{g}(x,y)=\sum_{i=0}^{M-1}\sum_{j=0}^{M-1} m(i,j)g(x-i,y-j)=m(x,y)*g(x,y)$$

$$m(x,y): M\times M \text{ convolution mask}$$

- Mask size ↑ Noise Removal ↑ Blurring ↑

$$1/16\begin{array}{|c|c|c|}\hline 1&2&1\\\hline 2&4&2\\\hline 1&2&1\\\hline\end{array} * \begin{array}{|c|c|c|}\hline 1&1&1\\\hline 1&4&1\\\hline 1&1&1\\\hline\end{array} = \boxed{1.75}$$

- Mask design
  - ✓ A single lobe shape: weighted averaging
  - ✓ Circularly symmetric: rotation invariant
  - ✓ Normalized form: to keep the range of values
- Associability of convolution

$$m(x,y)*[m(x,y)*g(x,y)]=[m(x,y)*m(x,y)]*g(x,y)$$

$$\underset{1/16}{\begin{array}{cccc}0 & & & \\ 0 & 1 & 2 & 1\\ 0 & 2 & 4 & 2\\ & 1 & 2 & 1\end{array}} * \underset{1/16}{\begin{array}{ccc}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{array}} = \underset{1/256}{\begin{array}{ccccc}1 & 4 & 6 & 4 & 1\\ 4 & 16 & 24 & 16 & 4\\ 6 & 24 & 36 & 24 & 6\\ 4 & 16 & 24 & 16 & 4\\ 1 & 4 & 6 & 4 & 1\end{array}}$$

- Mask design
  - ✓ A single lobe shape: weighted averaging
  - ✓ Circularly symmetric: rotation invariant
  - ✓ Normalized form: to keep the range of values
- Associability of convolution

$$m(x,y)*[m(x,y)*g(x,y)]=[m(x,y)*m(x,y)]*g(x,y)$$

$$\frac{1}{9}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} * \frac{1}{9}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} = \frac{1}{81}\begin{bmatrix}1&2&3&2&1\\2&4&6&4&1\\3&6&9&6&3\\2&4&6&4&1\\1&2&3&2&1\end{bmatrix}$$

❑ Convolution Theorem

$$m(x,y)*g(x,y) \Leftrightarrow M(u,v)G(u,v)$$

$$m(x,y)g(x,y) \Leftrightarrow M(u,v)*G(u,v)$$

$$\overline{g}(x,y) = m(x,y)*g(x,y) = \Im^{-1}\{M(u,v)G(u,v)\}$$

- Convolution operation → Linear operation
- Work in spatial domain vs. frequency domain
  - ✓ Convolution of an $N \times N$ image with $M \times M$ mask: $O(N^2M^2)$
  - ✓ Fast Fourier Transformation (FFT): $O(N^2 \log_2 N)$
  - ✓ Break-even point commonly $10 \leq M \leq 15$

❑ Gaussian Smoothing

- Repeated convolution with any smoothing mask

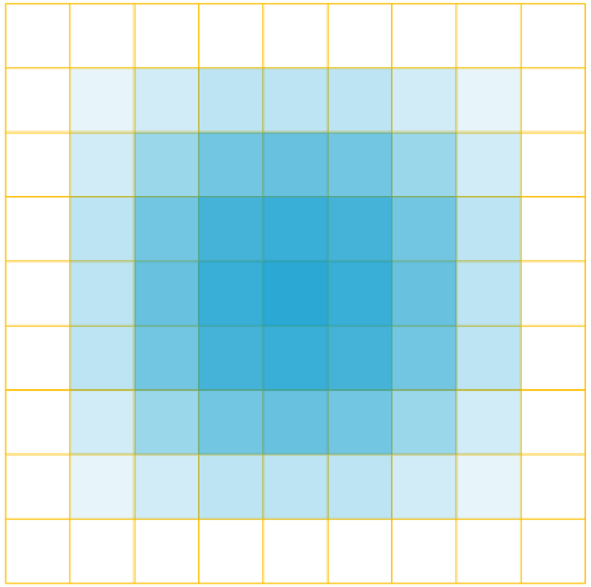
  → Converge to a Gaussian function

$$\exp\left(\frac{x^2+y^2}{-2\sigma^2}\right) \Leftrightarrow \sqrt{2\pi}\sigma \exp\left(\frac{u^2+v^2}{-2\alpha^2}\right), \alpha = \frac{1}{2\pi\sigma}$$

$$m(x,y) = G(x,y;\sigma) \Leftrightarrow M(u,v) = \sqrt{2\pi}\sigma\, G(u,v;\frac{1}{2\pi\sigma})$$

- Different amount of smoothing by varying standard deviation
- Rotationally symmetric, single lobe
- Separable: successive convolution of two 1D Gaussian

Repeated convolution with any smoothing mask converges to a Gaussian function ???



$$\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} * \begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} \Rightarrow \begin{bmatrix}1&2&3&2&1\\2&4&6&4&2\\3&6&9&6&3\\2&4&6&4&2\\1&2&3&2&1\end{bmatrix}$$

$$\begin{bmatrix}1&2&3&2&1\\2&4&6&4&2\\3&6&9&6&3\\2&4&6&4&2\\1&2&3&2&1\end{bmatrix} * \begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} \Rightarrow \begin{bmatrix}1&3&6&7&6&3&1\\3&9&18&21&18&9&3\\6&18&36&42&36&18&6\\7&21&42&49&42&21&7\\6&18&36&42&36&18&6\\3&9&18&21&18&9&3\\1&3&6&7&6&3&1\end{bmatrix}$$

Successive convolution of two 1D Gaussian ???

$$\underbrace{\begin{array}{ccc} 0 & 0 & 1 \\ & & 4 \\ & & 6 \\ & & 4 \\ & & 1 \end{array}}_{1/16} * \underbrace{\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}}_{1/16} * f(x,y) = \underbrace{\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}}_{1/256} * f(x,y)$$

❑ Binomial *vs*. Gaussian Filter

- Gaussian filter can be excellently approximated by the coefficients of binomial expansion

$$(1+x)^n = {}_nC_0 + {}_nC_1 x + {}_nC_2 x^2 + \cdots + {}_nC_n x^n$$

Pascal's Triangle

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1

	1	2	1
1	1	2	1
2	2	4	2
1	1	2	1

1/16

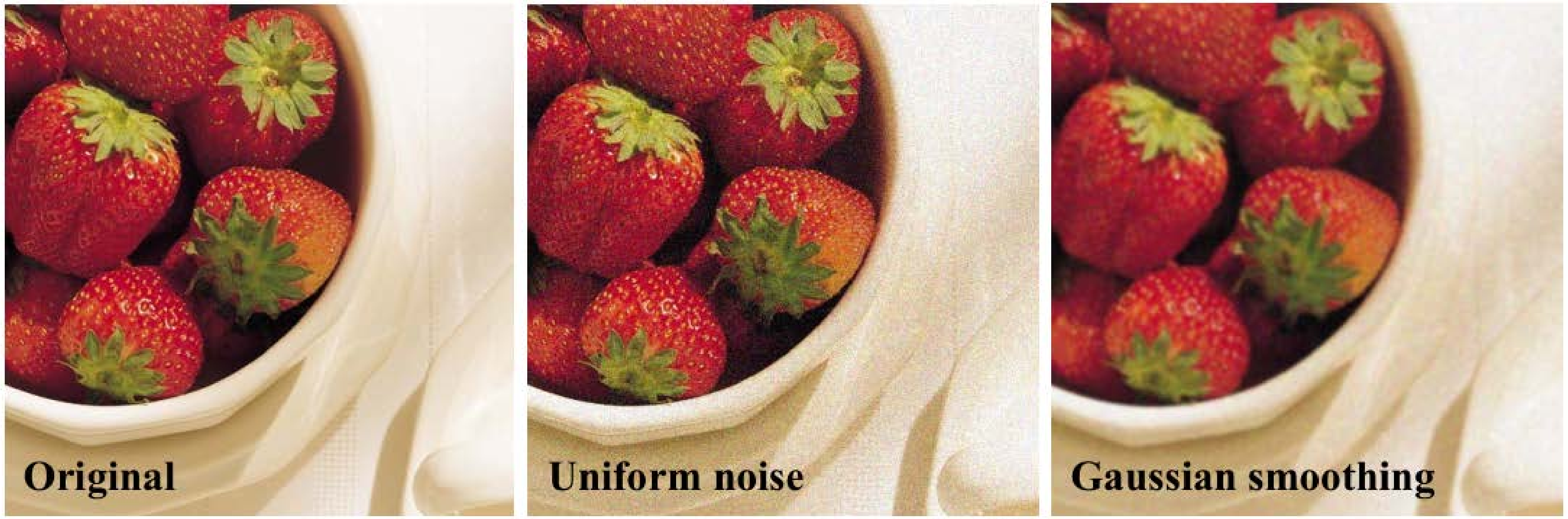
1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1

1/256

❑ Noise Removal in Each Color Component

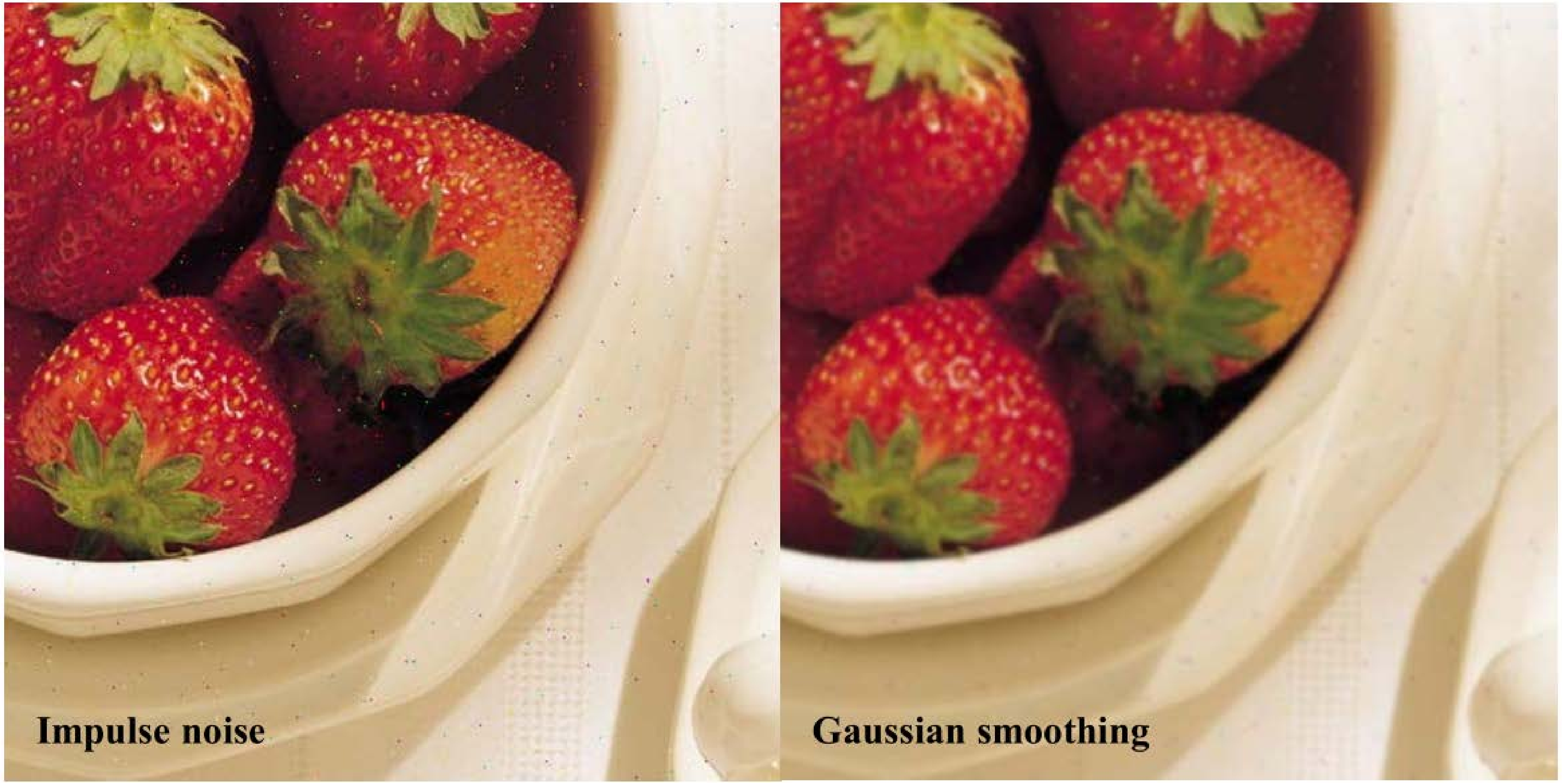
- Effective when additive noise can be modeled using a uniform or Gaussian distribution
- Less successful for impulse noise

  → corrupt color and intensity of adjacent points

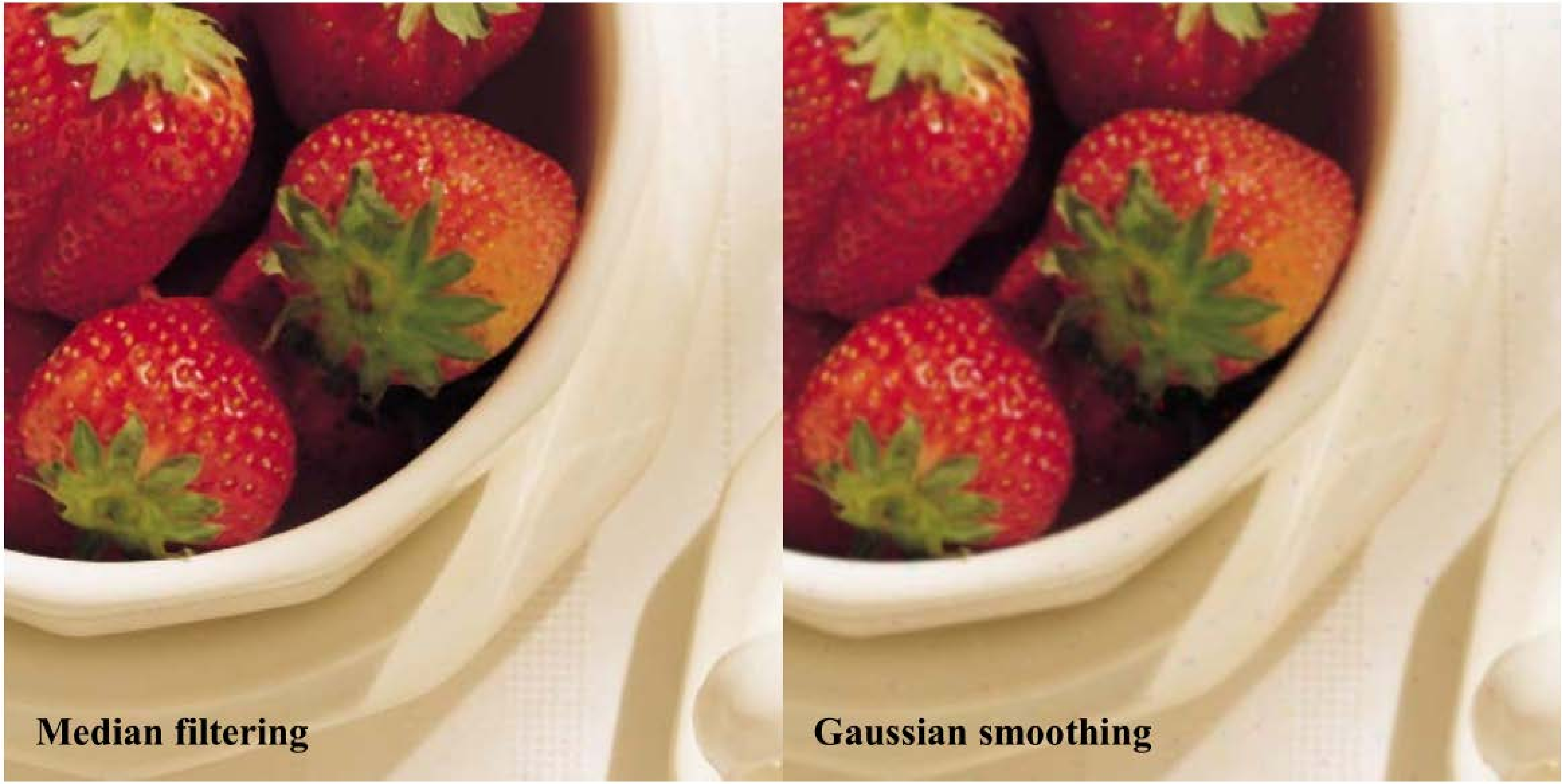


Noise Removal

Noise Removal

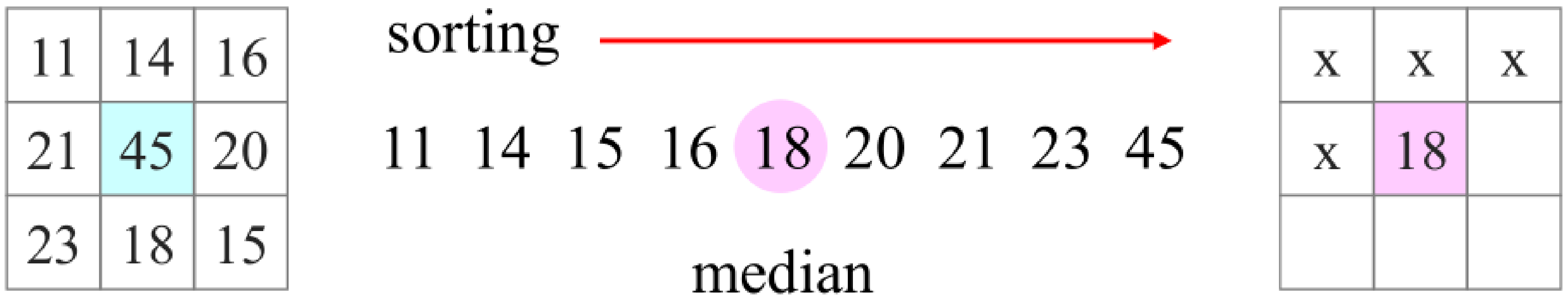


Noise Removal



❑ Median Filtering

- Replace a pixel value with the median value of neighbor pixels
- Weighted averaging effect + impulse removal
- Retain edges well
- Its application to each color component causes often chromatic shifts, particularly near edges



Noise Removal

❑ Vector Median

- Treat RGB values as vectors and calculate vector median
- Not natural way to sort vectors in RGB space
- Property: sum of distances between all vectors and the median is less than sum of distances to any other vector

$$S_m = \sum_{i=1}^{K} \left\| v_m - v_i \right\| < \sum_{i=1}^{K} \left\| v_j - v_i \right\|, \ \text{for} \ \forall\, j \neq m$$

- K^2 distance calculations are relatively time consuming

$$\rightarrow \quad S_m^{'} = \left\| v_m - \bar{v} \right\| < \left\| v_j - \bar{v} \right\|, \ \text{for} \ \forall\, j \neq m$$

- ‘plus sign’ shaped neighborhoods are more effective than circular neighborhoods

❑ Anisotropic Diffusion Approach

- Vary size and shape of smoothing neighborhoods in different parts of image based on image content
- (e.g.) Apply smoothing parallel to edges and not perpendicular to edges

5	4	6	5	25
4	5	6	26	28
6	4	29	24	30
5	23	24	25	29
27	24	26	27	31

5	4	6	5	25
4	5	5	26	28
6	5	29	24	30
5	23	24	25	29
27	24	26	27	31

$$\frac{1}{3}\begin{array}{|c|c|c|}\hline & & 1 \\ \hline & 1 & \\ \hline 1 & & \\ \hline\end{array}$$

5 *vs.* 12 $(\cong 109 / 9)$

Contrast Enhancement

❑ Contrast

$$\text{contrast} \propto \frac{\text{standard deviation}}{\text{mean}} = \frac{\sigma}{\mu}$$

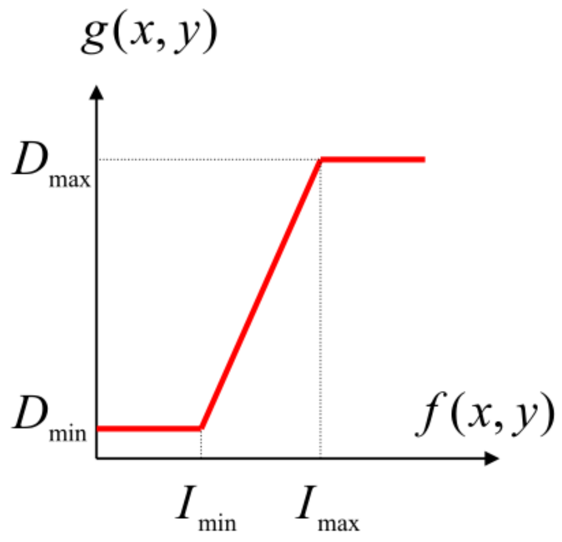
- A measure of sharpness
- Intensity or color variations in a local area
- High contrast: easy to locate object boundaries and distinctive features within objects
- Contrast enhancement

  → Amplifying local intensity or color variations within an image, thereby increasing feature visibility

Windowing

- A point operator: $g(x,y) = m\big(f(x,y)\big)$
- Stretch the range of interest $[I_{\min}, I_{\max}]$ to appropriate display range $[D_{\min}, D_{\max}]$
- Clip values out of the interest range
- Color windowing on each of color channels separately

  → color shift & unnatural looking



Contrast Enhancement

Only R-channel   Only B-channel

❑ Histogram Equalization

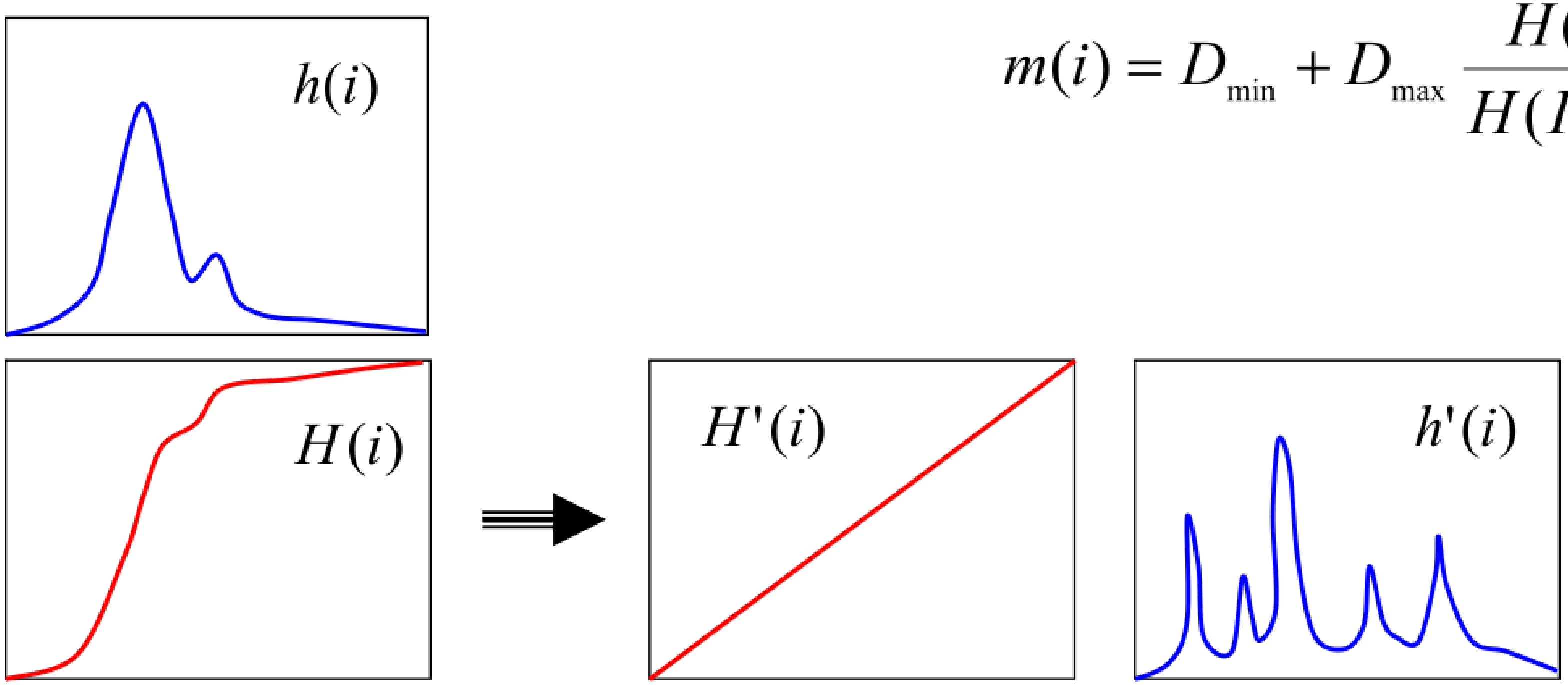
- Histogram $h(i)$
- Cumulative histogram $H(i)$
- Equalization function $m(i)$

$$h(i) = \sum_{(x,y)} \begin{cases} 1 & \text{if } f(x,y) = i \\ 0 & \text{otherwise} \end{cases}$$

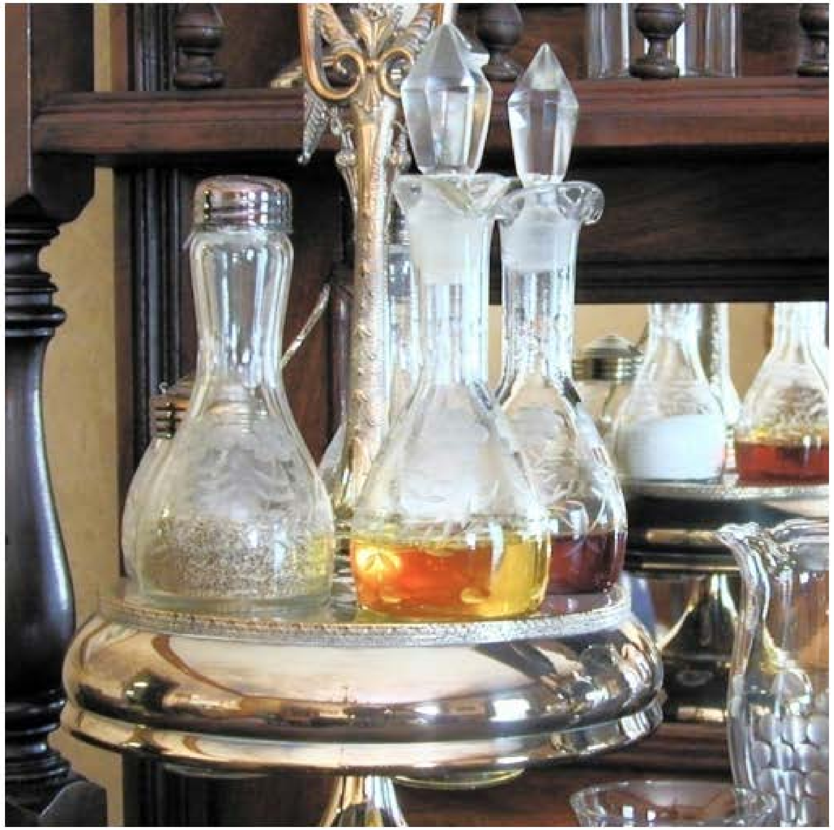
$$H(i) = \sum_{j=0}^{i} h(i)$$

$$g(x,y) = m\big(f(x,y)\big)$$

$$m(i) = D_{\min} + D_{\max} \frac{H(i)}{H(I_{\max})}$$



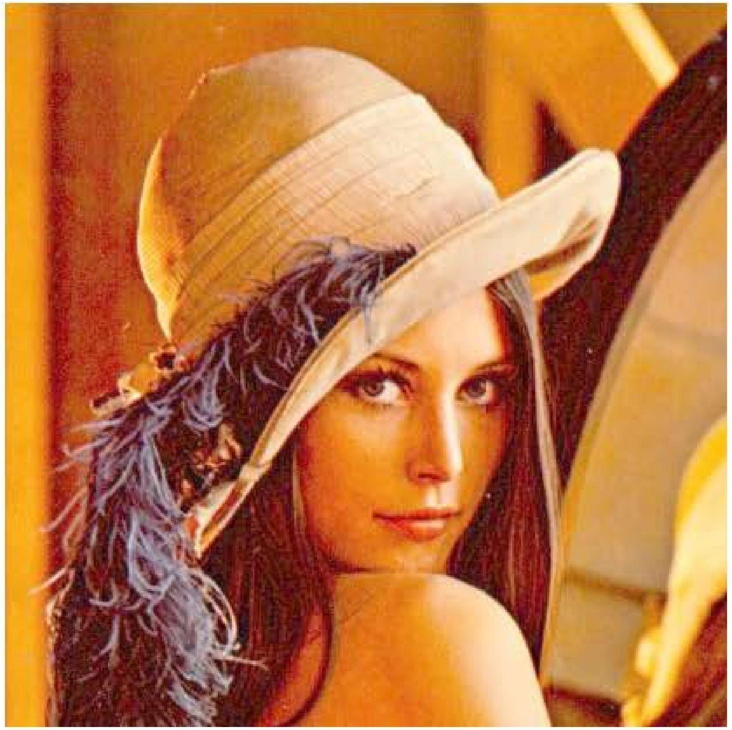
- Enhance only intensity in color images
  - ✓ Prevent from distorting chromatic information



Contrast Enhancement

Original

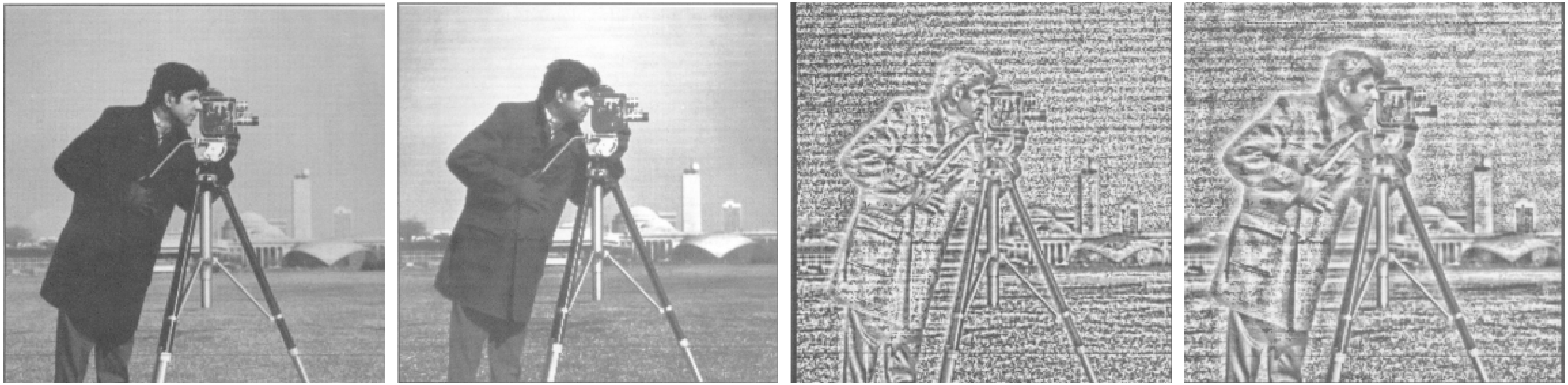
Only L-channel



All channels

Contrast Enhancement

- Adaptive histogram equalization
  - ✓ Local enhancement
  - ✓ Histogram equalization in each small block separately

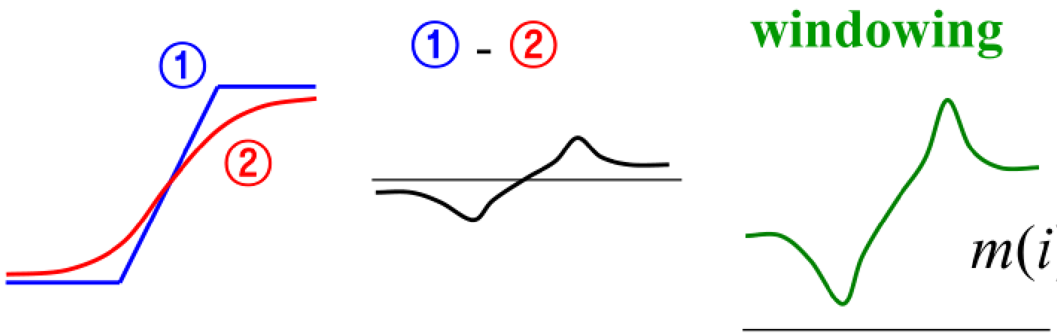


❑ Unsharp Masking

Windowing (Original Image – Unsharp Image)

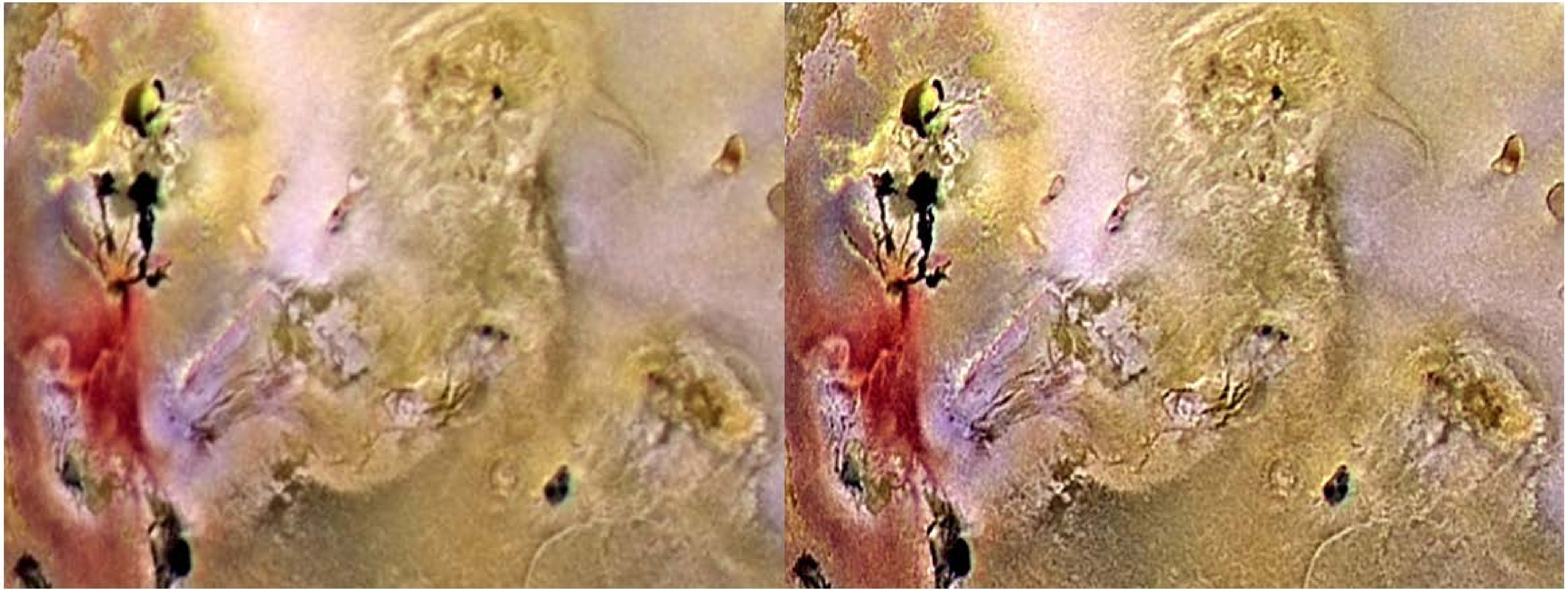
= Windowing (Original Image – Low-Pass Filtered Image)

$$
\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}
- \frac{1}{16}
\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}
= \frac{1}{16}
\begin{array}{|c|c|c|}\hline -1 & -2 & -1 \\ \hline -2 & 12 & -2 \\ \hline -1 & -2 & -1 \\ \hline \end{array}
$$



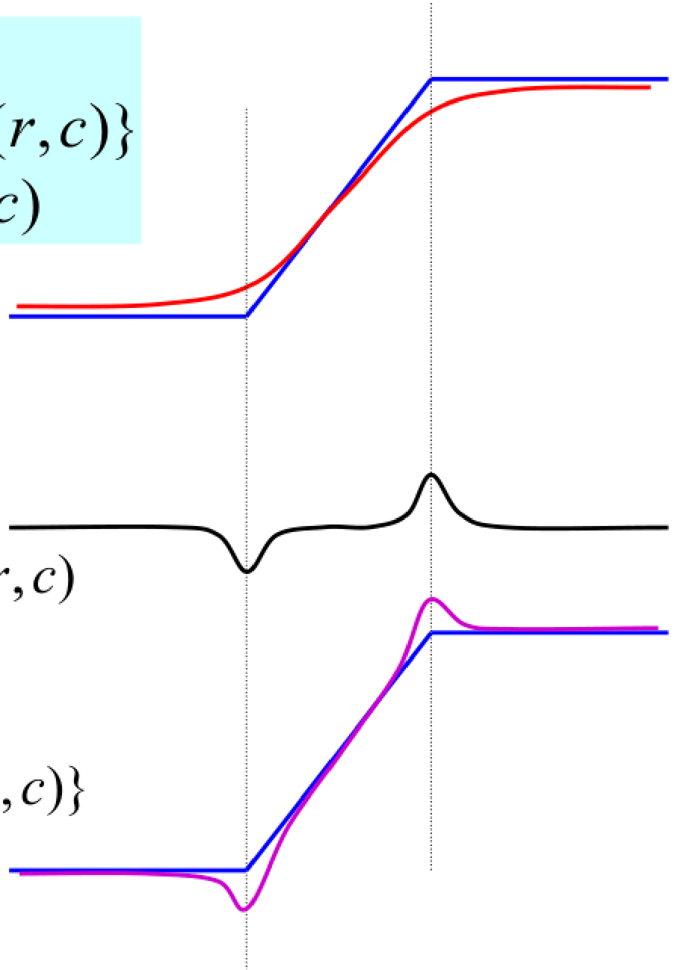
$$m(i) = D_{\min} + (i - I_{\min}) \frac{D_{\max} - D_{\min}}{I_{\max} - I_{\min}}$$

- Unsharp Masking for Color Images
  - ✓ Unsharp masking for each channel separately
  - ✓ The same windowing for color channels to avoid hue shift



❑ Sharpening Concept

$$
\begin{aligned}
E(r,c) &= I(r,c) + \lambda\, H(r,c) \\
&= I(r,c) + \lambda\{I(r,c) - L(r,c)\} \\
&= (1+\lambda) I(r,c) - \lambda\, L(r,c)
\end{aligned}
$$



Contrast Enhancement

❑ Constant Variance Enhancement

- Enhance visual significance of local changes

$$g(x,y)=\frac{f(x,y)-\overline{f}(x,y)}{\sigma(x,y)}+k\overline{f}(x,y),\quad k\in[0..1]$$

$$\text{where }\ \sigma(x,y)=\sum_{(i,j)}\left(f(x-i,y-j)-\overline{f}(x,y)\right)^2$$

$$g(x,y)=\frac{f(x,y)-\overline{f}(x,y)}{\sigma(x,y)}+k\mu(x,y),\quad k\in[0..1]$$

$$\text{where }\ \mu(x,y)=\text{global mean of input image}$$

- For color images, apply the same contrast boost to each color channel: use average of local deviations for $\sigma(x,y)$

- Other enhancement methods
  - $m(i) = \log i$ for emphasizing low values
  - $m(i) = i^p$ with $p > 1$ for emphasizing high values
  - High-pass or band-pass filtering in frequency domain
  - Homomorphic filtering
  - etc
- Enhancement of Color Images
  - Treat each color channel separately
  - Need special care for avoiding color shift
    - ✓ Use the same amount of enhancement in each channel
    - ✓ Enhance only the intensity

Summary

- Noise Removal
  - Noise Models
  - Binomial Smoothing
  - Gaussian Smoothing
  - Median Filtering, etc
- Contrast Enhancement
  - Windowing
  - Histogram Equalization
  - Unsharp Masking
  - Contrast Variance Enhancement, etc

Q & A

Thank You!