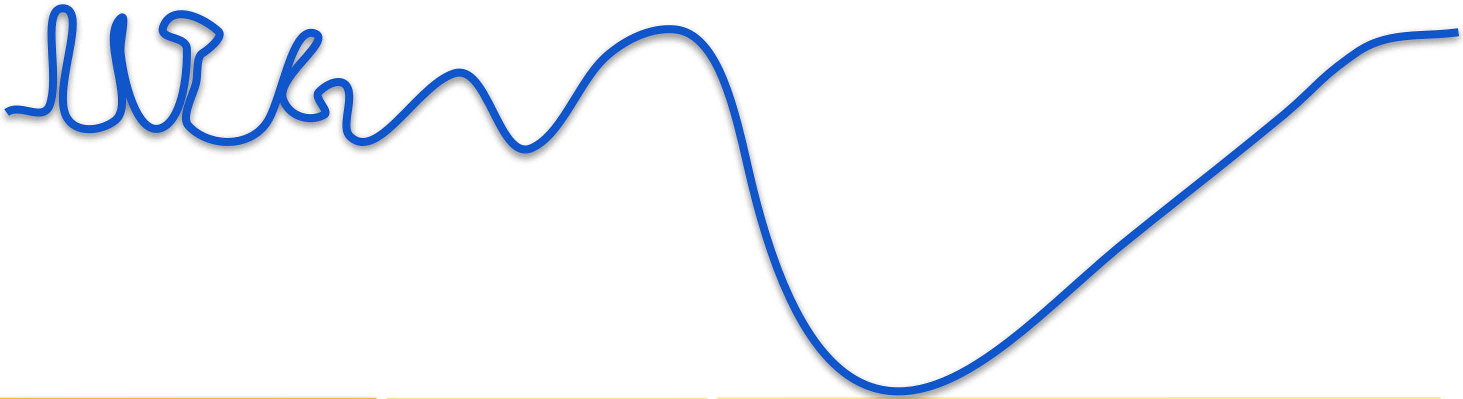
Computing with Signals



DA 623

Jan - May 2024

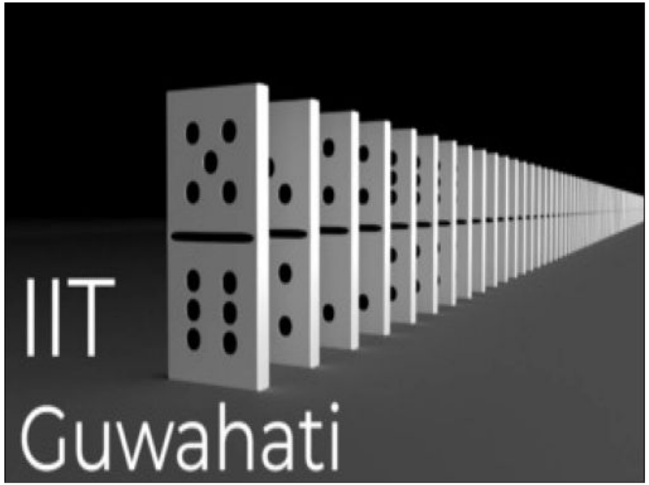
IIT Guwahati

Instructors: Neeraj Sharma

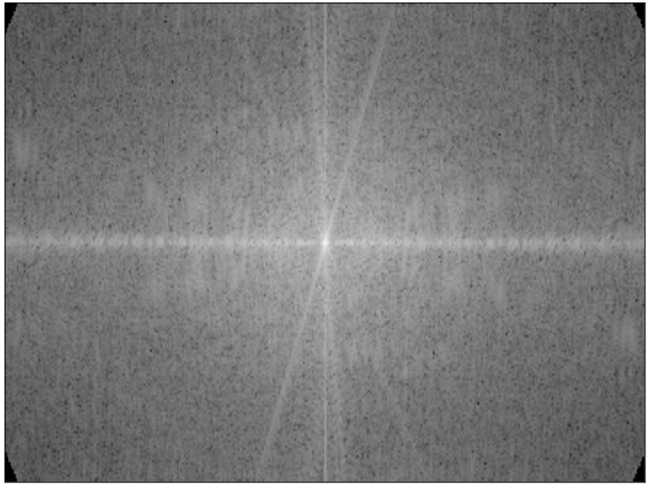
Lecture-18

Using Convolution to Design Linear Image Filters

Lowpass filtered image
(the cutoff frequency is gradually increasing)

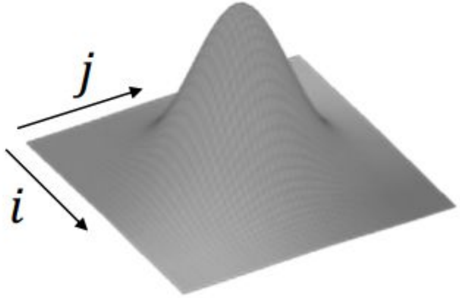


2-D Fourier Spectrum
(in dB, post circular lowpass filtering)

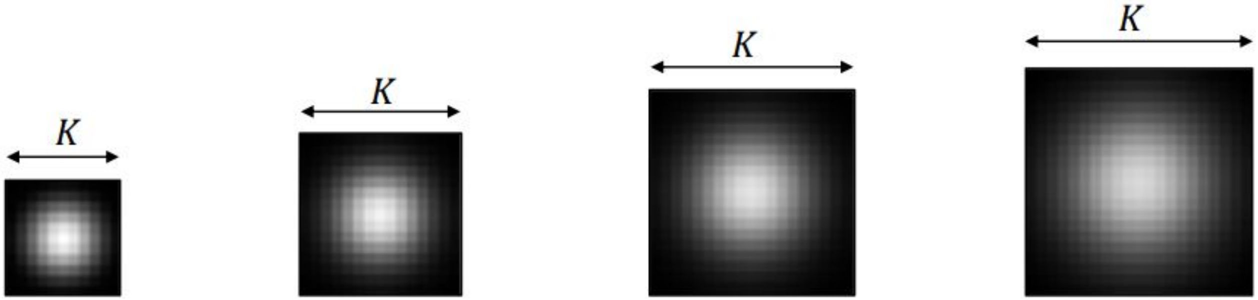


Gaussian Kernel

$$n_\sigma[i,j] = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\left(\frac{i^2+j^2}{\sigma^2}\right)} \qquad \boxed{1}$$



σ^2: Variance



$\sigma = 2$ $\sigma = 3$ $\sigma = 4$ $\sigma = 5$

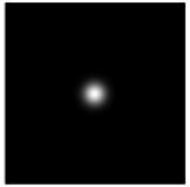
Rule of thumb: Set kernel size $K \approx 2\pi\sigma$

Reference:
https://fpcv.cs.columbia.edu/Monographs

Gaussian Smoothing

Input

$*$



$\sigma = 4$

$=$

Output

$f(x, y)$ $\quad$ $n_4(x, y)$ $\quad$ $g(x, y)$

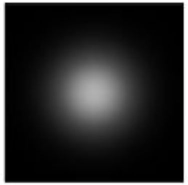
Larger the kernel (or σ), more the blurring

Reference:
https://fpcv.cs.columbia.edu/Monographs

Gaussian Smoothing

Input

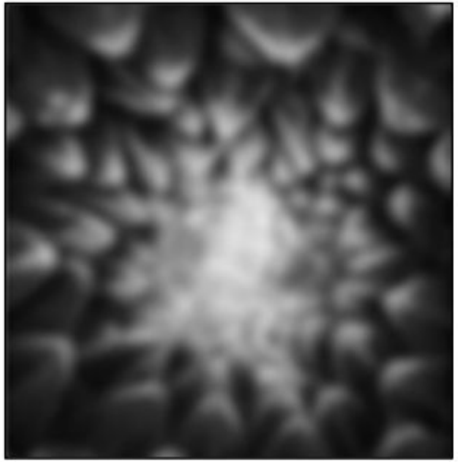
$*$



$\sigma = 16$

$=$

Output



$f(x, y)$ $\quad n_{16}(x, y)$ $\quad g(x, y)$

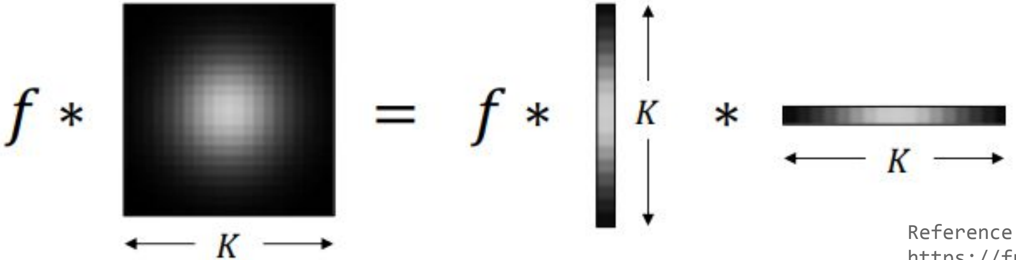
Larger the kernel (or σ), more the blurring

Gaussian Smoothing is Separable

$$g[i,j] = \frac{1}{2\pi\sigma^2} \sum_{m=1}^{K} \sum_{n=1}^{K} e^{-\frac{1}{2}\left(\frac{m^2+n^2}{\sigma^2}\right)} f[i-m, j-n]$$

$$g[i,j] = \frac{1}{2\pi\sigma^2} \sum_{m=1}^{K} e^{-\frac{1}{2}\left(\frac{m^2}{\sigma^2}\right)} . \sum_{n=1}^{K} e^{-\frac{1}{2}\left(\frac{n^2}{\sigma^2}\right)} f[i-m, j-n]$$

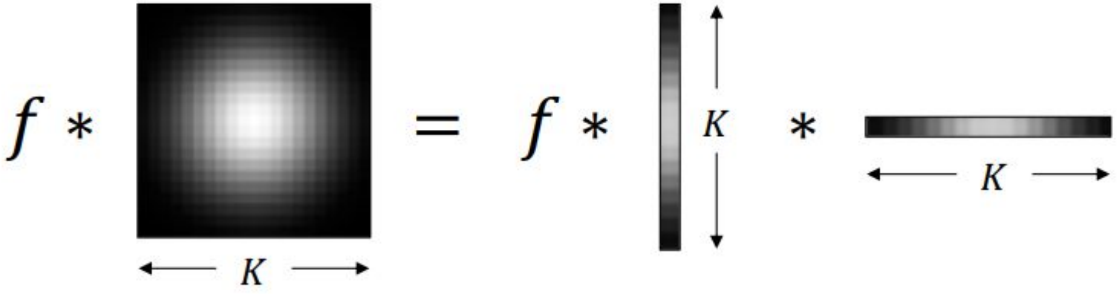
Using One 2D Gaussian Filter ≡ Using Two 1D Gaussian Filters



Reference:
https://fpcv.cs.columbia.edu/Monographs

Gaussian Smoothing is Separable

Using One 2D Gaussian Filter ≡ Using Two 1D Gaussian Filters



Which one is faster? Why?

K^2 Multiplications	$2K$ Multiplications
$K^2 - 1$ Additions	$2(K-1)$ Additions

Reference: https://fpcv.cs.columbia.edu/Monographs

Non-Linear Image Filters

Smoothing to Remove Image Noise

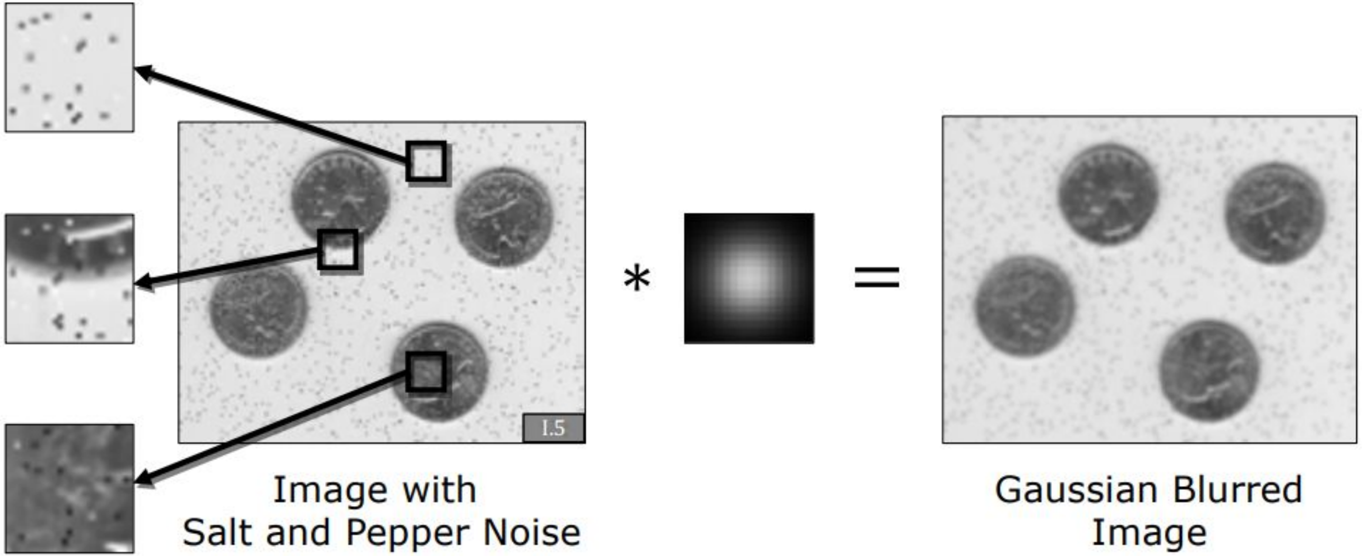


Image with
Salt and Pepper Noise

Gaussian Blurred
Image

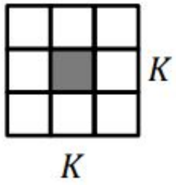
Problem with Smoothing:

- Does not remove outliers (Noise)
- Smooths edges (Blur)

Reference:
https://fpcv.cs.columbia.edu/Monographs

Median Filtering

1. Sort the K^2 values in window centered at the pixel
2. Assign the Middle Value (Median) to pixel



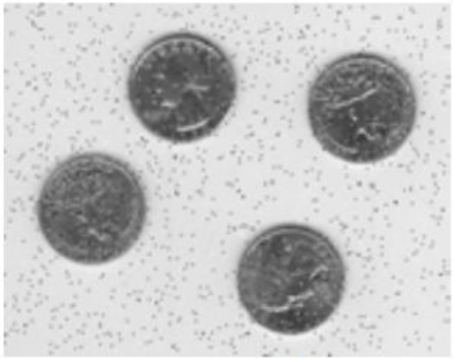


Image with
Salt and Pepper Noise

Median Filtered
Image ($K = 3$)

Non-linear Operation

(Cannot be implemented using convolution)

Median Filtering

Not Effective when Image Noise is not a Simple Salt and Pepper Noise.

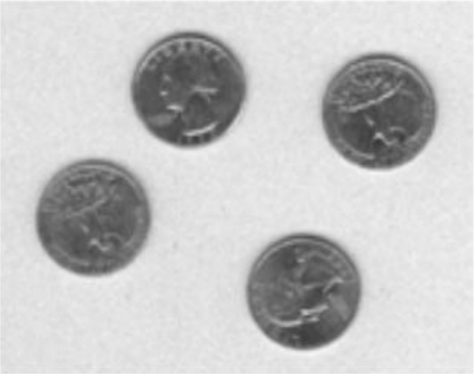
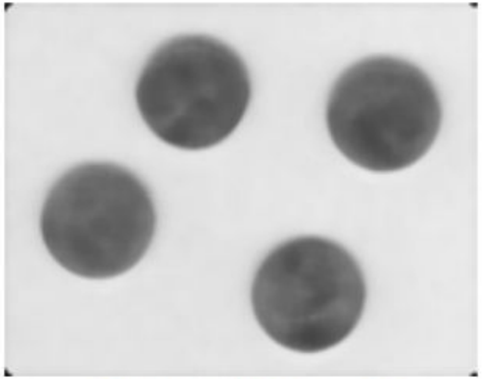


Image with Noise



Median Filtered Image ($K = 11$)

Larger K causes blurring of image detail

Template Matching

Template Matching

Template

How do we locate the template in the image?

Minimize:

$$E[i,j] = \sum_m \sum_n (f[m,n] - t[m-i,n-j])^2$$

Reference:
https://fpcv.cs.columbia.edu/Monographs

Template Matching

Template

How do we locate the template in the image?

Minimize:

$$E[i,j] = \sum_m \sum_n (f[m,n] - t[m-i,n-j])^2$$

$$E[i,j] = \sum_m \sum_n (f^2[m,n] + t^2[m-i,n-j] - \underbrace{\overset{\boxed{1}}{2f[m,n]t[m-i,n-j]}}_{\text{Maximize}})$$

Reference:
https://fpcv.cs.columbia.edu/Monographs

Template Matching

Template

How do we locate the template in the image?

Maximize:

$$\boxed{R_{tf}[i,j] = \sum_{m}\sum_{n} f[m,n]t[m-i,n-j] = t \otimes f}$$

(Cross-Correlation)

Convolution vs. Correlation

Convolution:

$$g[i,j] = \sum_{m}\sum_{n} f[m,n]\underline{t[i-m,j-n]} = t * f$$

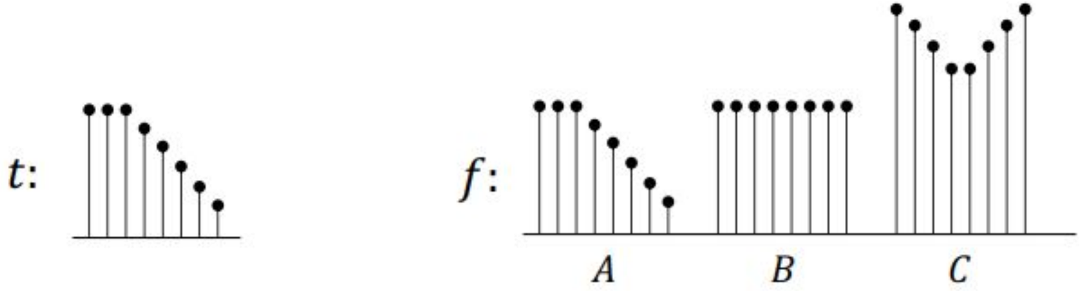
Correlation:

$$R_{tf}[i,j] = \sum_{m}\sum_{n} f[m,n]\underline{t[m-i,n-j]} = t \otimes f$$

No Flipping in Correlation

Problem with Cross-Correlation

$$R_{tf}[i,j] = \sum_m \sum_n f[m,n]t[m-i,n-j] = t \otimes f$$



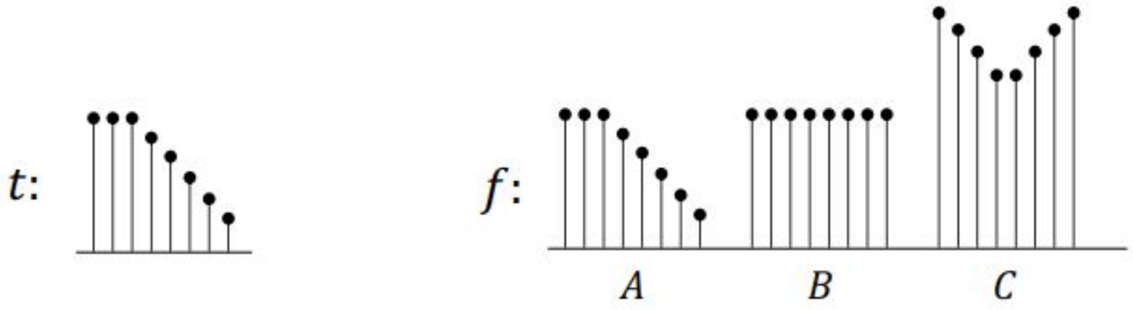
$$R_{tf}(C) > R_{tf}(B) > R_{tf}(A)$$

We need $R_{tf}(A)$ to be the maximum!

Normalized Cross-Correlation

Account for energy differences

$$N_{tf}[i,j] = \frac{\sum_m \sum_n f[m,n]t[m-i,n-j]}{\sqrt{\sum_m \sum_n f^2[m,n]}\sqrt{\sum_m \sum_n t^2[m-i,n-j]}}$$



$$N_{tf}(A) > N_{tf}(B) > N_{tf}(C)$$

Normalized Cross-Correlation

Account for energy differences

$$N_{tf}[i,j] = \frac{\sum_m \sum_n f[m,n]t[m-i,n-j]}{\sqrt{\sum_m \sum_n f^2[m,n]}\sqrt{\sum_m \sum_n t^2[m-i,n-j]}}$$

$\otimes$

$=$