

7-9 Monday – 309-GD2

# Xử lý ảnh

## INT3404 1

Giảng viên: TS. Nguyễn Thị Ngọc Diệp

Email: [ngocdiep@vnu.edu.vn](mailto:ngocdiep@vnu.edu.vn)Slide & code: [https://github.com/chupibk/INT3404\\_1](https://github.com/chupibk/INT3404_1)

## Lịch trình

Tuần	Nội dung	Yêu cầu đối với sinh viên
1	Giới thiệu môn học Làm quen với OpenCV + Python	Cài đặt môi trường: Python 3, OpenCV 3, Numpy, Jupyter Notebook
2	Phép toán điểm (Point operations) – Điều chỉnh độ tương phản – Ghép ảnh	Làm bài tập 1: điều chỉnh gamma tìm contrast hợp lý
3	Histogram - Histogram equalization - Phân loại ảnh dùng so sánh histogram	Thực hành ở nhà
4	Phép lọc trong không gian điểm ảnh (linear processing filtering) - làm mịn, làm sắc ảnh	Thực hành ở nhà Tìm hiểu thêm các phép lọc
5	Tìm cạnh (edge detection) Tìm features	Thực hành ở nhà
6	Các phép toán hình thái (Erosion, Dilation, Opening, Closing) - tìm biên số	Làm bài tập 2: tìm barcode
7	Chuyển đổi không gian - miền tần số (Fourier) - Hough transform	Thực hành ở nhà
8	Phân vùng (segmentation) - depth estimation - threshold-based - watershed/grabcut	Đăng ký thực hiện bài tập lớn
9	Mô hình màu Chuyển đổi giữa các mô hình màu	Làm bài tập 3: Chuyển đổi mô hình màu và thực hiện phân vùng
10	Mô hình nhiễu - Giảm nhiễu - Khôi phục ảnh - Giảm nhiễu chu kỳ - Ước lượng hàm Degration - Hàm lọc ngược, hàm lọc Wiener	Thực hành ở nhà
11	Template matching – Image Matching	Làm bài tập 4: puzzle
12	Nén ảnh	Thực hành ở nhà
13	Hướng dẫn thực hiện đồ án môn học	Trình bày đồ án môn học
14	Hướng dẫn thực hiện đồ án môn học	Trình bày đồ án môn học
15	Tổng kết cuối kỳ	Ôn tập

Xử lý ảnh - INT3404 1 - Diệping - 2019 UET.VNU

3

## Ôn lại tuần 2+3: Phép toán trên điểm ảnh

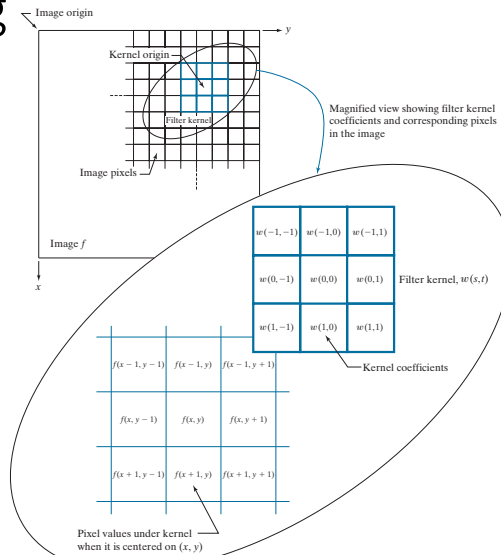
- Biến đổi trên điểm ảnh (e.g., gamma)
- Ghép ảnh (cộng, trừ, tính trung bình)
- Tính toán và sử dụng histogram
- Bài tập về nhà:
  - Deadline: 15/9/2019 23:59 → 45 – 5 (late submission) = 40
  - <https://forms.gle/ALrnAWc3c6h3dMqU9>
  - Bài “vớt vát”:
    - <https://forms.gle/bNTMBqKuuBg4V4o3A>

## Tuần 4: Các phép lọc

# Linear spatial filtering

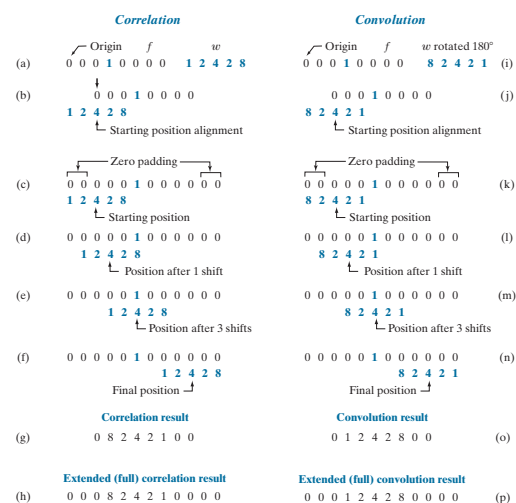
Kernel size:  $m \times n$   
 $m = 2a + 1$   
 $n = 2b + 1$   
 Image size:  $M \times N$

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)$$



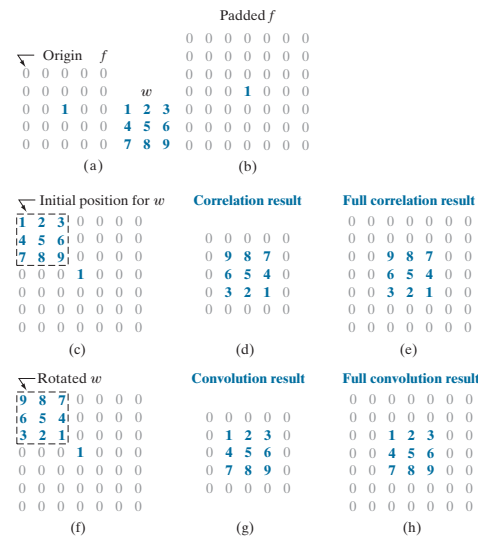
Source: Fig. 3.28, Gonzalez

## Correlation vs Convolution in 1D



Source: Fig. 3.29, Gonzalez

## Correlation vs Convolution in 2D



Source: Fig. 3.30, Gonzalez

## Correlation vs Convolution

Correlation

$$(w \star f)(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)$$

Convolution

$$(w \star f)(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x - s, y - t)$$

## Properties of convolution

**TABLE 3.5**  
Some fundamental properties of convolution and correlation. A dash means that the property does not hold.

Property	Convolution	Correlation
Commutative	$f \star g = g \star f$	—
Associative	$f \star (g \star h) = (f \star g) \star h$	—
Distributive	$f \star (g + h) = (f \star g) + (f \star h)$	$f \star (g + h) = (f \star g) + (f \star h)$

Differentiation:  $\frac{\partial}{\partial x}(f \star g) = \frac{\partial f}{\partial x} \star g$

## Separability

$$w = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

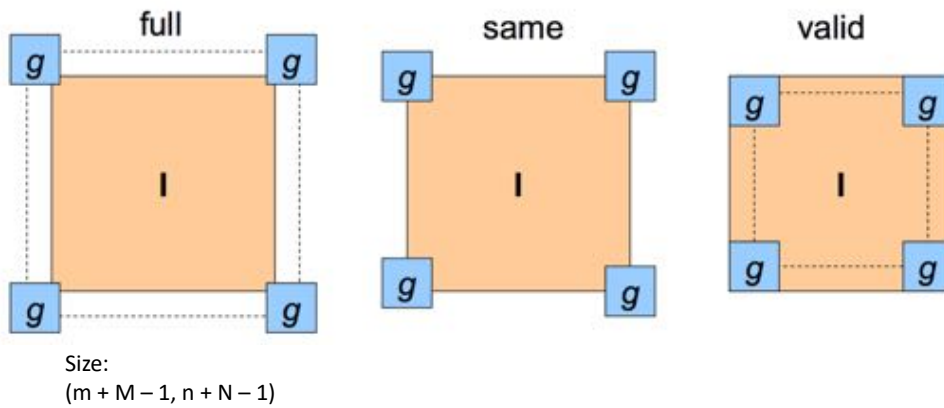
is separable because it can be expressed as the outer product of the vectors

$$\mathbf{c} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{and} \quad \mathbf{r} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

That is,

$$\mathbf{c} \mathbf{r}^T = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = w$$

## Boundary issues



## What to do around the edge?

- Pad a constant value (black)
- Wrap around (circulate the image)
- Copy edge (replicate the edges' pixels)
- Reflect across edges (symmetric)



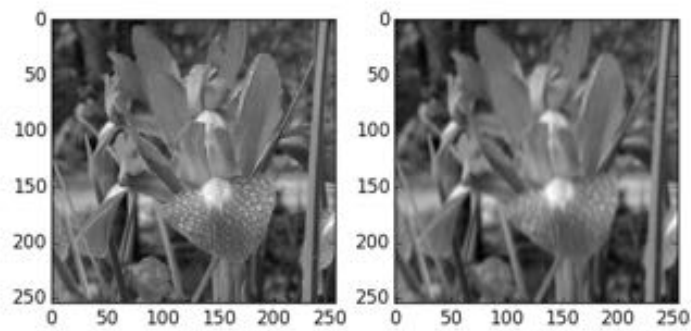
## Các phép lọc thường gặp

### Construct a filter

- A filter that computes the average of pixels in a neighborhood blurs an image
- A filter that computes the local derivative of an image sharpens the image

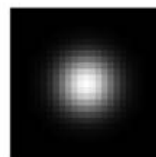
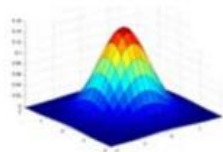
## Mean filter

$$M = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



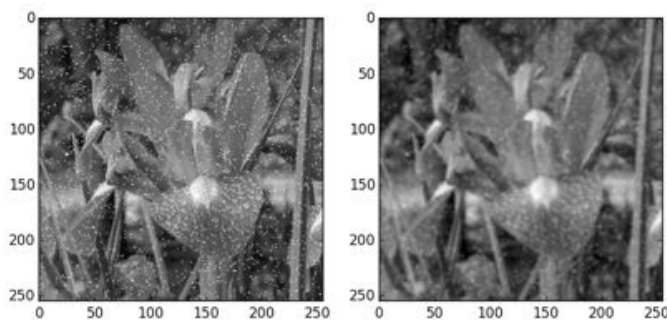
## Gaussian filter

$$G_{\sigma} = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$



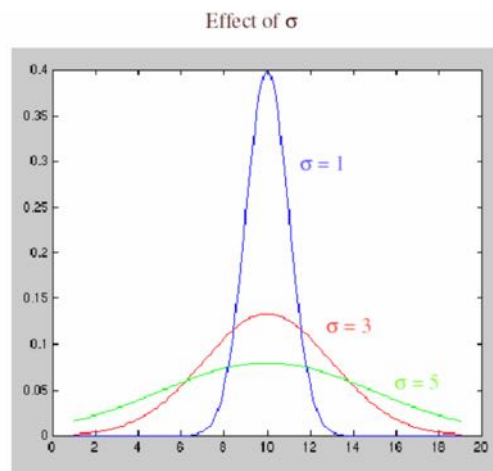
0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

5 x 5,  $\sigma = 1$

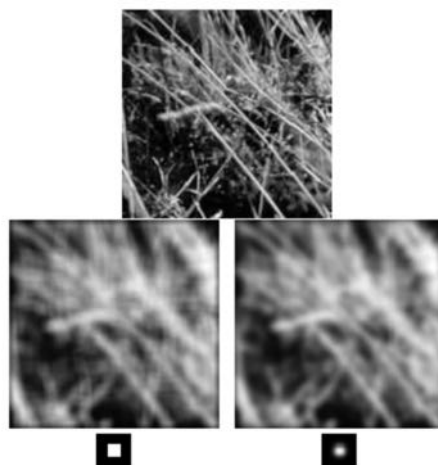




## Gaussian's sigma

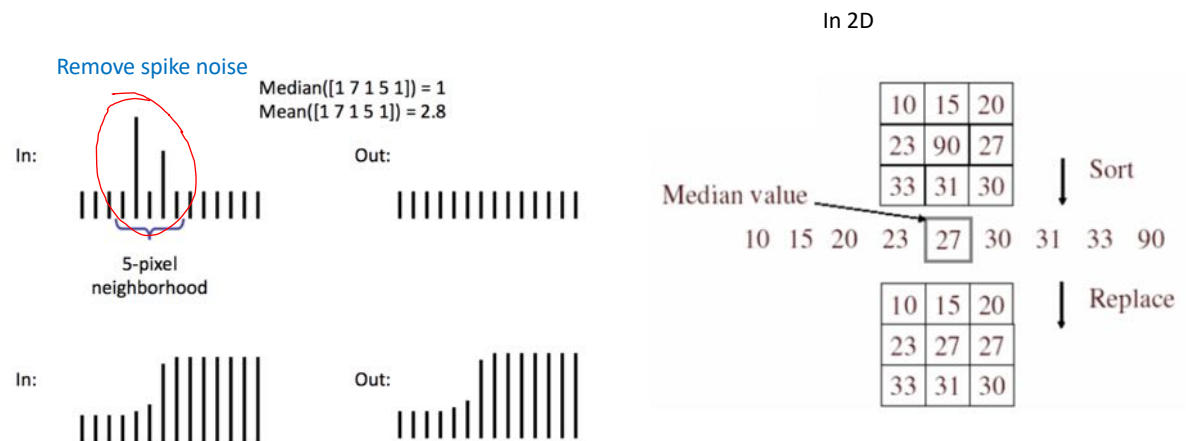


## Mean vs Gaussian filtering

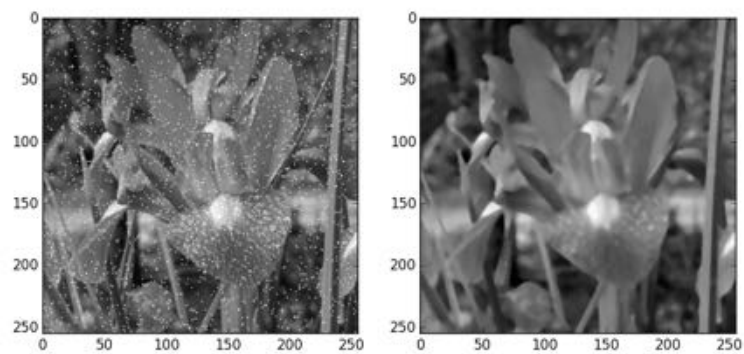


Source: Grauman

## Median filter



## Median filtering

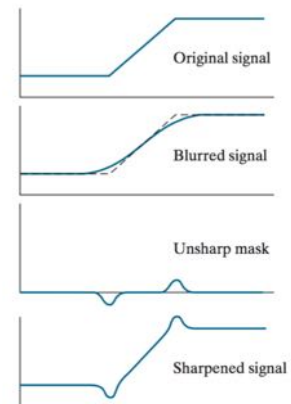


# Unsharp masking

$$g_{\text{mask}}(x, y) = f(x, y) - \overset{\text{blurred}}{\bar{f}}(x, y)$$

$$g(x, y) = f(x, y) + \overset{\text{add a weighted portion of the mask}}{k g_{\text{mask}}(x, y)}$$

$k = 1 \rightarrow$  unsharp masking  
 $k > 1 \rightarrow$  highboost filtering



a b c  
d e

**FIGURE 3.49** (a) Original image of size  $600 \times 259$  pixels. (b) Image blurred using a  $31 \times 31$  Gaussian lowpass filter with  $\sigma = 5$ . (c) Mask. (d) Result of unsharp masking using Eq. (3-56) with  $k = 1$ . (e) Result of highboost filtering with  $k = 4.5$ .

Source: Gonzalez