XỬ LÝ ẢNH

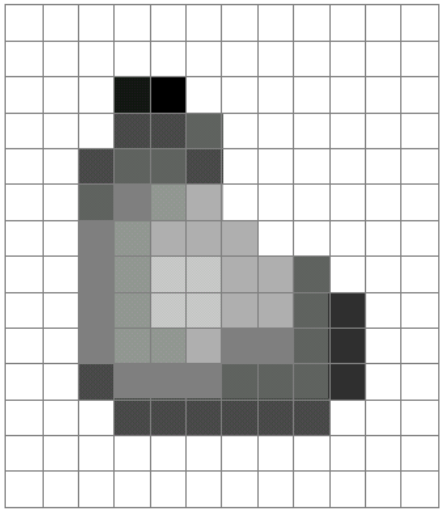
(Image Processing)

BÀI 4: CƠ SỞ TOÁN

Correlation vs Convolution

Review: What is an image?

- A grid (matrix) of intensity values



=

255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	20	0	255	255	255	255	255	255	255
255	255	255	75	75	75	255	255	255	255	255	255
255	255	75	95	95	75	255	255	255	255	255	255
255	255	96	127	145	175	255	255	255	255	255	255
255	255	127	145	175	175	175	255	255	255	255	255
255	255	127	145	200	200	175	175	95	255	255	255
255	255	127	145	200	200	175	175	95	47	255	255
255	255	127	145	145	175	127	127	95	47	255	255
255	255	74	127	127	127	95	95	95	47	255	255
255	255	255	74	74	74	74	74	74	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255

(common to use one byte per value: 0 = black, 255 = white)

Review: Images as functions

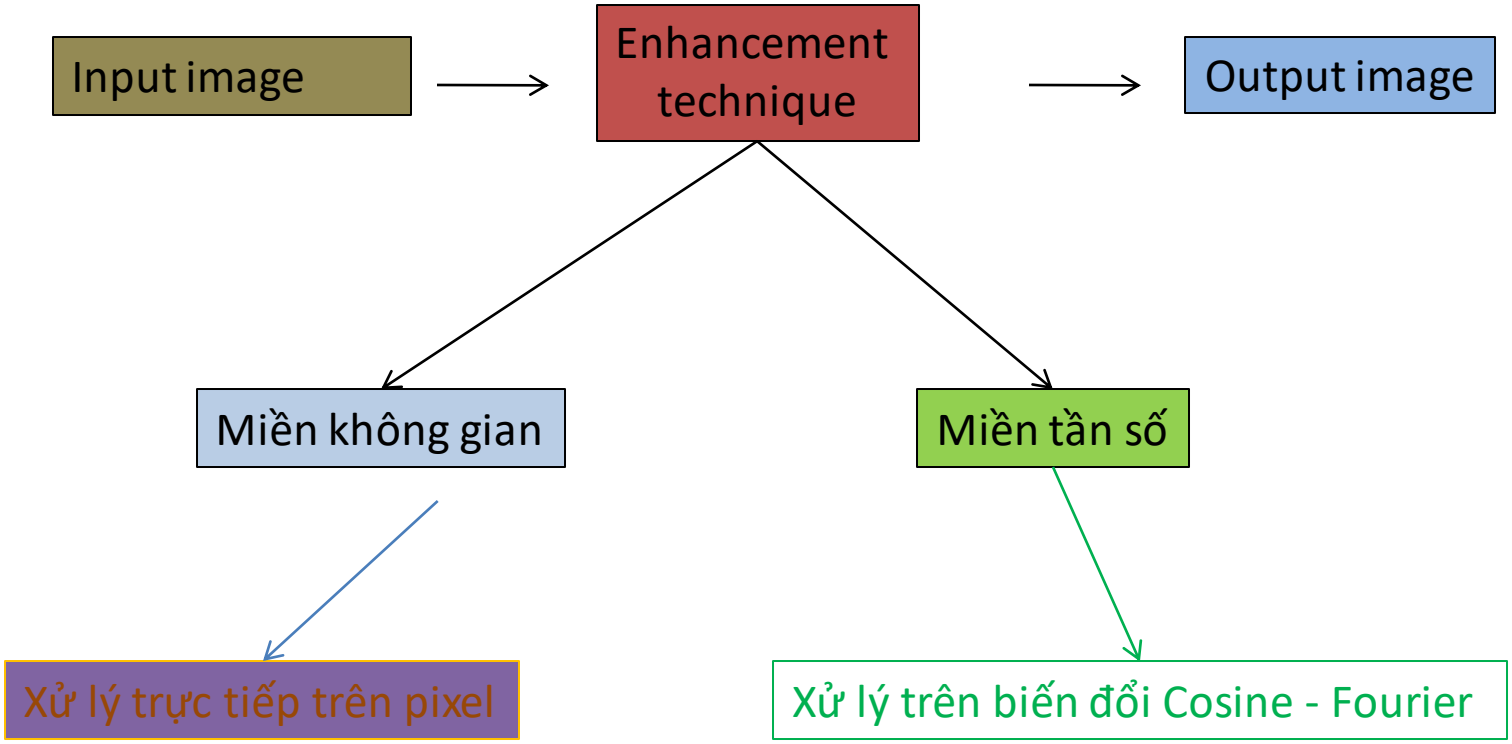
- An image contains discrete numbers of pixels

- Pixel value
  - grayscale/intensity
    - [0,255]
  - Color
    - RGB [R, G, B], where [0,255] per channel
    - Lab [L, a, b]: Lightness, a and b are color-opponent dimensions
    - HSV [H, S, V]: Hue, saturation, value

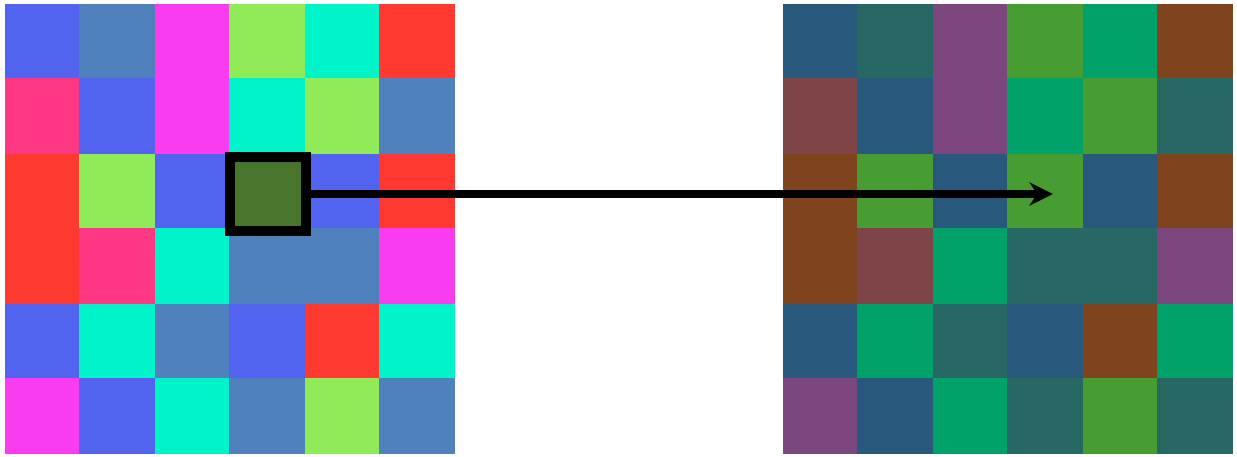
Review: Images as functions

- Can think of image as a **function**, f, from R^2 to R or R^M:
  - Grayscale: $f(x,y)$ gives **intensity** at position (x,y)
    - f: [a,b] x [c,d] ➔ [0,255]
  - Color: $f(x,y) = [r(x,y), g(x,y), b(x,y)]$

Image transformations



Review: Point Operations



Review: Point Operations

Original

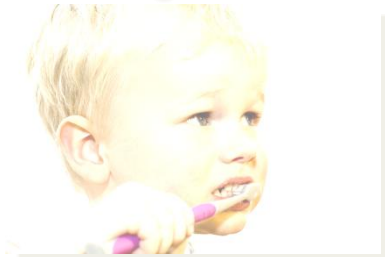
Darken

Lower Contrast

Nonlinear Lower Contrast

Invert

Lighten



Raise Contrast

Nonlinear Raise Contrast

Review: Point Operations

Original

`x`

Darken

`x - 128`

Lower Contrast

`x / 2`

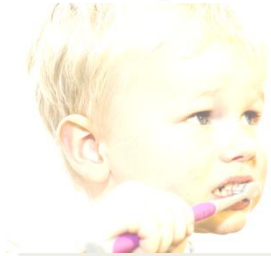
Nonlinear Lower Contrast

`((x / 255.0) ^ 0.33) * 255.0`

Invert

`255 - x`

Lighten



`x + 128`

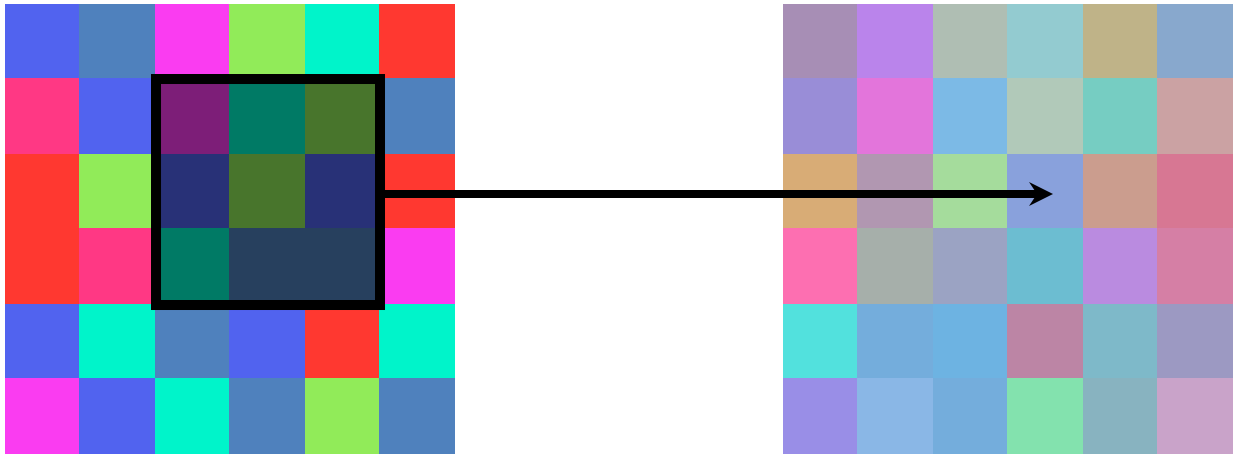
Raise Contrast

`x * 2`

Nonlinear Raise Contrast

`((x / 255.0) ^2) * 255.0`

Neighborhood Operations



Example: Tính Mean

0	0	0	0	0	0	0	0	0	0
0	0	0	10	10	10	0	0	0	0
0	0	10	20	20	20	10	40	0	0
0	10	20	30	0	20	10	0	0	0
0	10	0	30	40	30	20	10	0	0
0	10	20	30	40	30	20	10	0	0
0	10	20	10	40	30	20	10	0	0
0	10	20	30	30	20	10	0	0	0
0	0	10	20	20	0	10	0	20	0
0	0	0	10	10	10	0	0	0	0

$$(0 + 0 + 0 + 10 + 40 + 0 + 10 + 0 + 0)/9 = 6.66$$

Example: Tính Mean

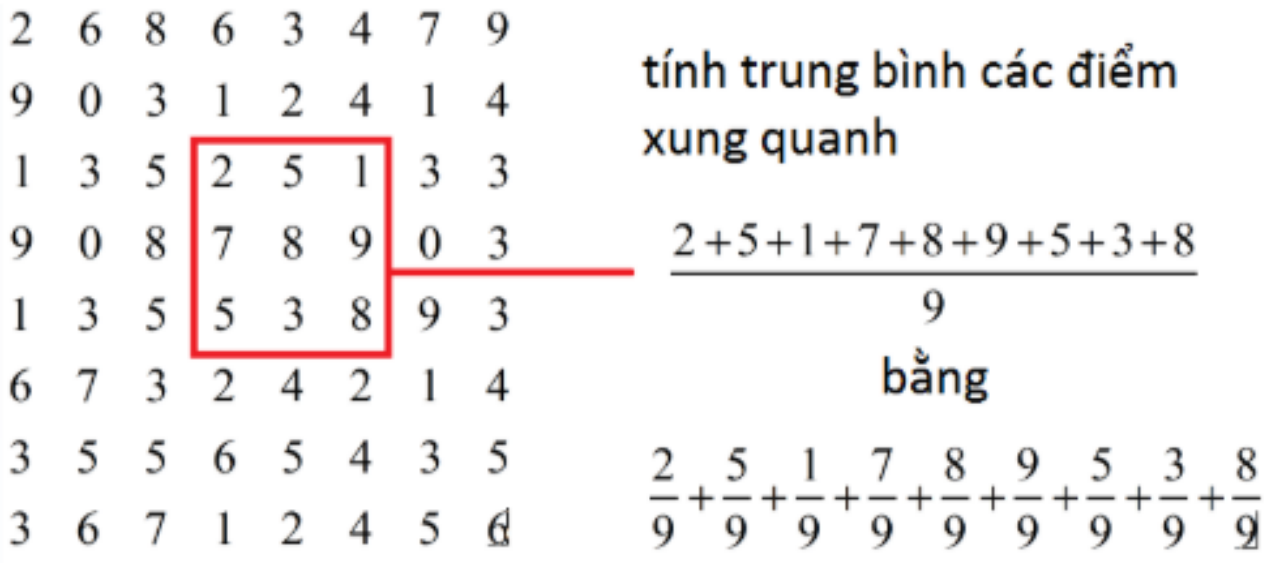
0	0	0	0	0	0	0	0	0	0
0	0	0	10	10	10	0	0	0	0
0	0	10	20	20	20	10	40	0	0
0	10	20	30	0	20	10	0	0	0
0	10	0	30	40	30	20	10	0	0
0	10	20	30	40	30	20	10	0	0
0	10	20	10	40	30	20	10	0	0
0	10	20	30	30	20	10	0	0	0
0	0	10	20	20	0	10	0	20	0
0	0	0	10	10	10	0	0	0	0

(0 + 0 + 0 + 0 + 0 + 10 + 0 + 0 + 0 + 0 + 20 + 10 + 40 + 0 + 0 + 20 + 10 + 0 + 0 + 0 + 30 + 20 + 10 + 0 + 0)/25 = 6.8

Noise reduction

Neighborhood Operations

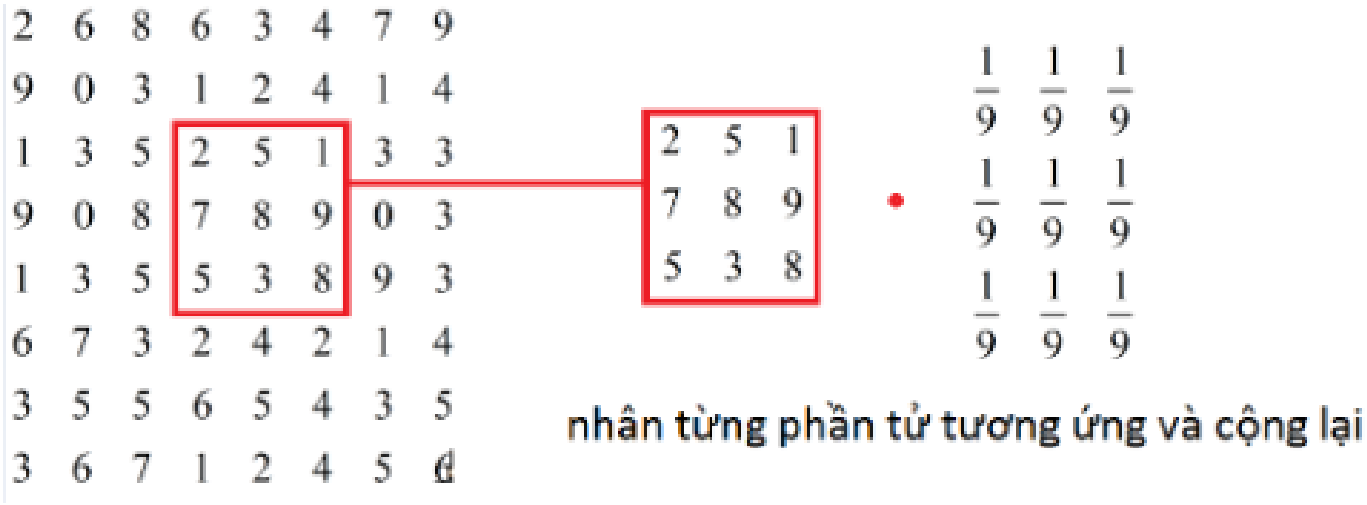
- Trung bình các giá trị xung quanh



- Phép tính trên cũng giống như nhân từng giá trị của các pixel lân cận với 1/9 sau đó cộng lại với nhau.

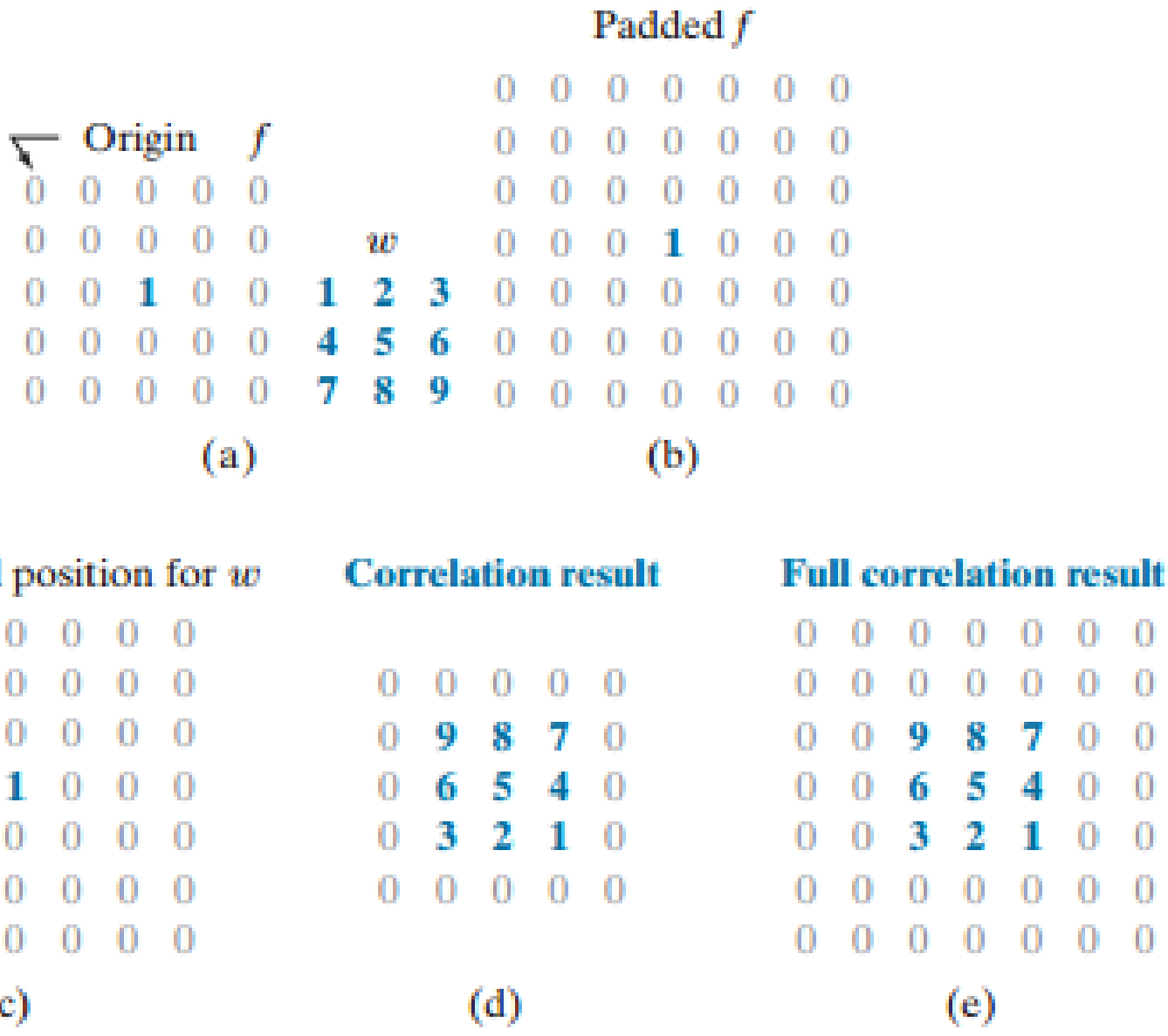
Neighborhood Operations

- Tương đương:

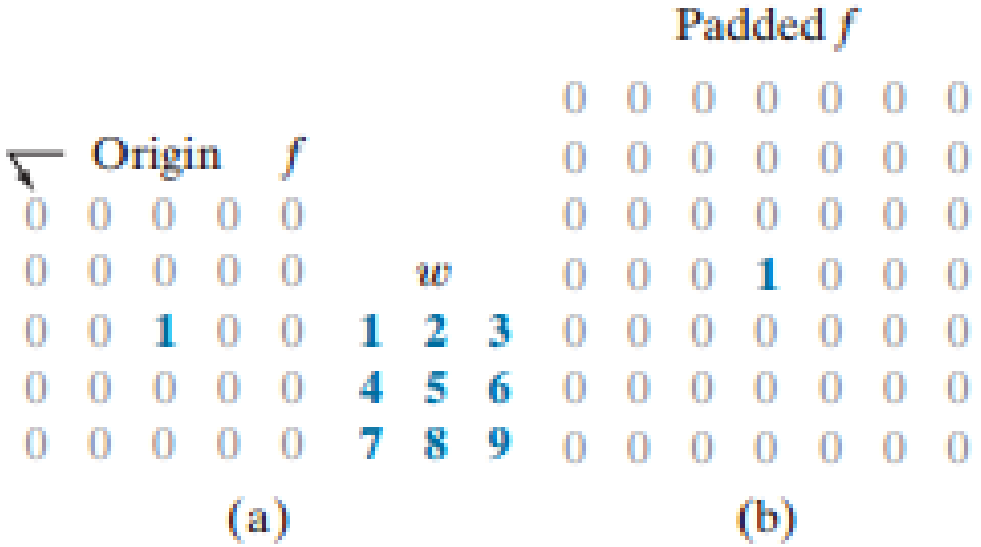


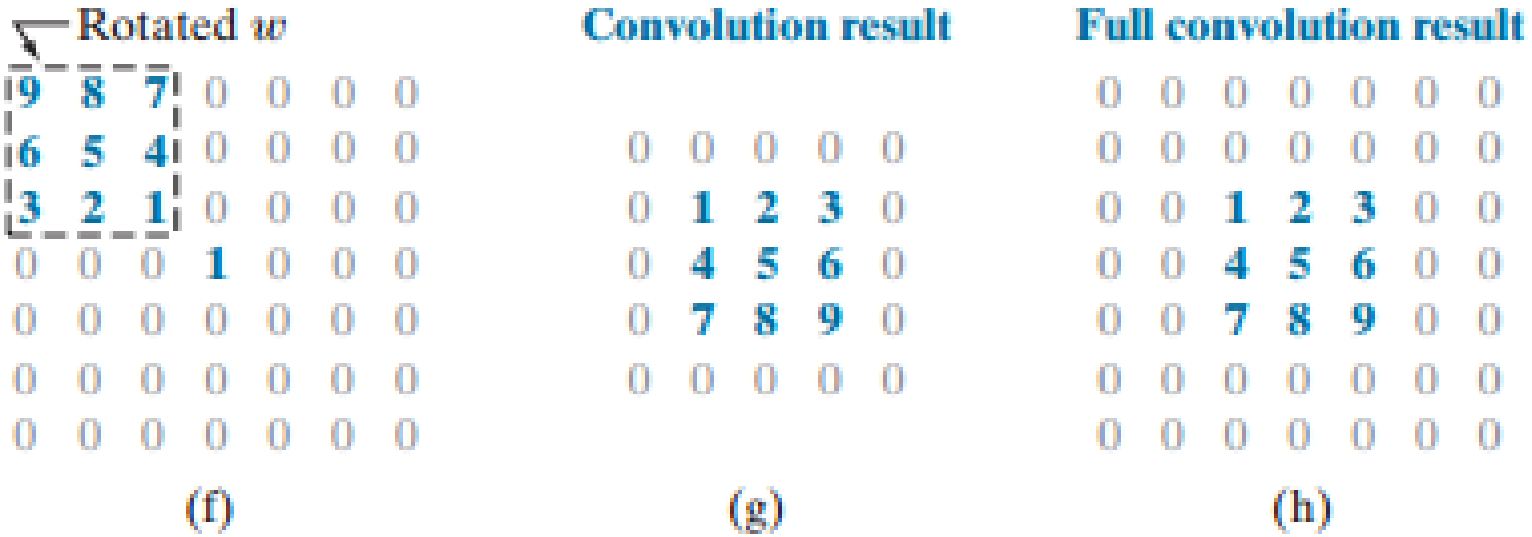
- Phép tính trên được gọi là phép tích chập (Convolution).
- Ma trận $\begin{matrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{matrix}$ được gọi là kernel.

Correlation vs Convolution in 2D



Correlation vs Convolution in 2D





Correlation – Tương quan

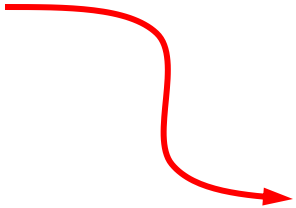
$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} \omega(s,t) f(x+s, y+t)$$

$$g = \omega \circ f$$

Correlation

correlation kernel, ω

1	-1	-1
1	2	-1
1	1	1



Input Image f

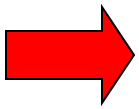
2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

Correlation

1	-1	-1
1	2	-1
1	1	1

1	-1	-1		
1	4	-2	2	3
1	2	1	3	3
	2	2	1	2
	1	3	2	2

Input Image, f



5			

output Image, *g*

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

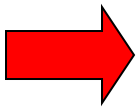
Correlation

1	-1	-1
1	2	-1
1	1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

1	-1	-1	
2	4	-2	3
2	1	3	3
2	2	1	2
1	3	2	2

Input Image, f



5	10		

output Image, *g*

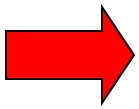
Correlation

1	-1	-1
1	2	-1
1	1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

	1	-1	-1
2	2	4	-3
2	1	3	3
2	2	1	2
1	3	2	2

Input Image, f



5	10	10	

output Image, *g*

Correlation

1	-1	-1
1	2	-1
1	1	1

		1	-1	-1
2	2	2	6	-1
2	1	3	3	1
2	2	1	2	
1	3	2	2	

Input Image, f

5	10	10	15

output Image, *g*

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

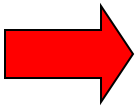
Correlation

1	-1	-1
1	2	-1
1	1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

1	-2	-2	2	3
1	4	-1	3	3
1	2	2	1	2
	1	3	2	2

Input Image, f



5	10	10	15
3			

output Image, *g*

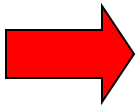
Correlation

1	-1	-1
1	2	-1
1	1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

2	-2	-2	3
2	2	-3	3
2	2	1	2
1	3	2	2

Input Image, f



5	10	10	15
3	4		

output Image, *g*

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

Correlation

5	10	10	15
3	4	6	11
7	11	4	9
-5	4	4	5

Final output Image, *g*

Ý nghĩa của correlation

- Tìm **sự tương quan** của kernel và ảnh gốc. Vùng pixel trên ảnh gốc có mẫu (pattern) **càng giống** với cửa sổ (kernel) thì giá trị tại điểm tương ứng của ảnh đầu ra **càng lớn**.

Convolution

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

$$g = \omega * f$$

Convolution

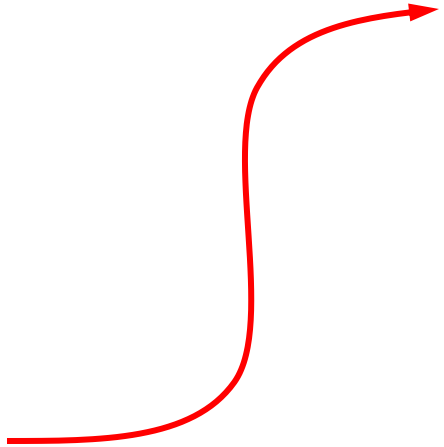
Correlation kernel, ω

1	-1	-1
1	2	-1
1	1	1

Rotate 180°

1	1	1
-1	2	1
-1	-1	1

Convolution kernel, ω



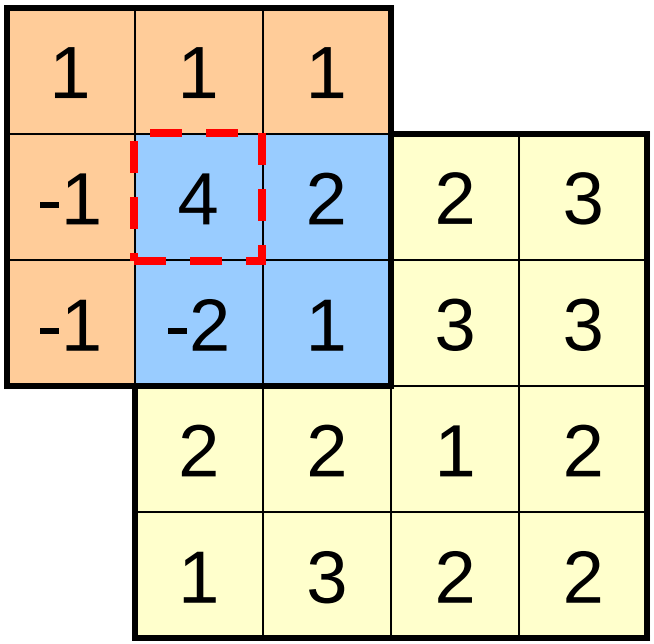
Input Image, f

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

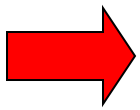
Convolution

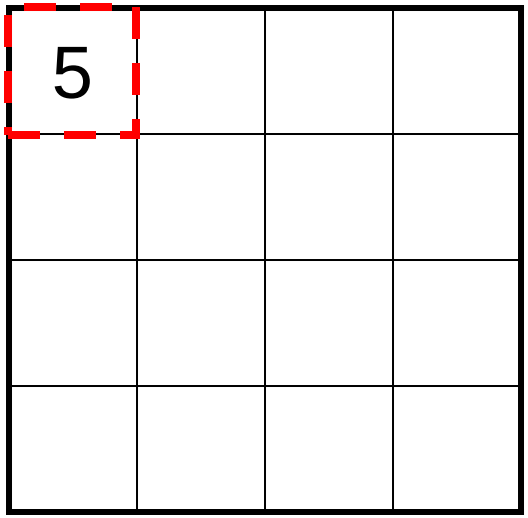
1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2



Input Image, *f*





Output Image, *g*

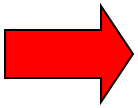
Convolution

1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

1	1	1	
-2	4	2	3
-2	-1	3	3
2	2	1	2
1	3	2	2

Input Image, *f*



5	4		

Output Image, *g*

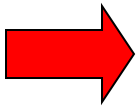
Convolution

1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

	1	1	1
2	-2	4	3
2	-1	-3	3
2	2	1	2
1	3	2	2

Input Image, *f*



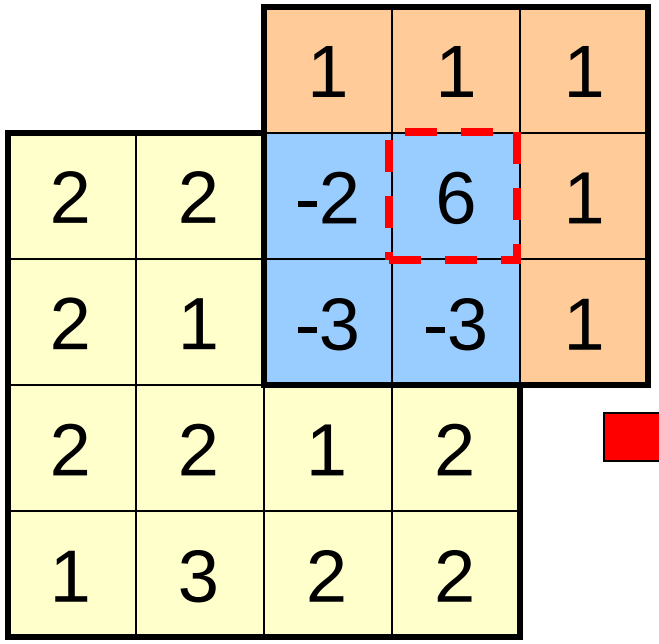
5	4	4	

Output Image, *g*

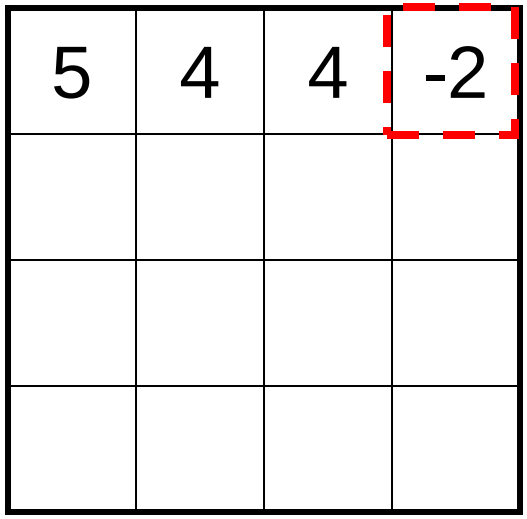
Convolution

1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2



Input Image, *f*



Output Image, *g*

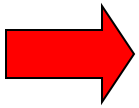
Convolution

1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

1	2	2	2	3
-1	4	1	3	3
-1	-2	2	1	2
	1	3	2	2

Input Image, *f*



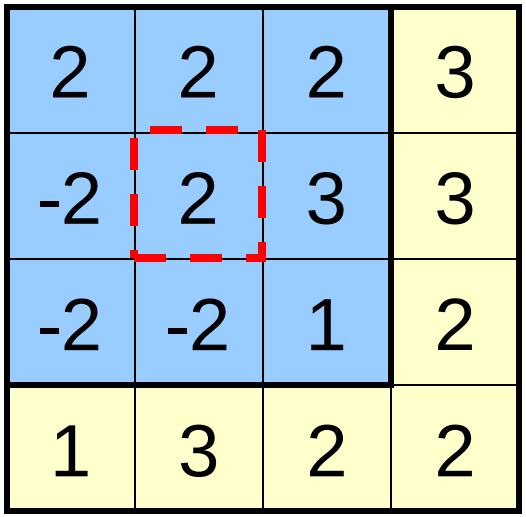
5	4	4	-2
9			

Output Image, *g*

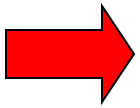
Convolution

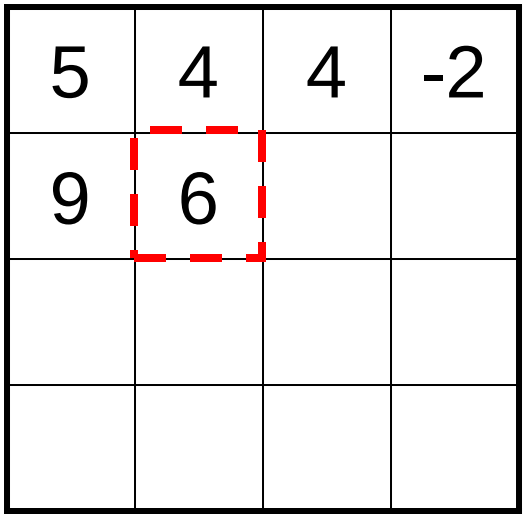
1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2



Input Image, *f*





Output Image, *g*

Convolution

5	4	4	-2
9	6	14	5
11	7	6	5
9	12	8	5

Final output Image, *g*

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	$f \star (g + h) = (f \star g) + (f \star h)$	$f \star\!\!\star (g + h) = (f \star\!\!\star g) + (f \star\!\!\star h)$

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

Ý nghĩa của Convolution

- **Convolution** có tính chất kết hợp:

  (K2*(K1*I) = (K2*K1)*I)
- Thay vì lấy ảnh I **convolve** với kernel K1, sau đó lấy ảnh kết quả **convolve** với K2 thì ta thực hiện lấy kernel K1 **convole** với K2 thành 1 kernel nào đó, sau đó lấy kernel kết quả này áp dụng cho ảnh gốc I.
- Nhờ tính chất này mà khi thiết kế kernel, thay vì thiết kế nhiều phép **convolve** tuần tự ta có thể kết hợp chúng lại thành 1 kernel duy nhất.
- Chi phí tính toán của **convolution / correlation** là tương đối lớn.

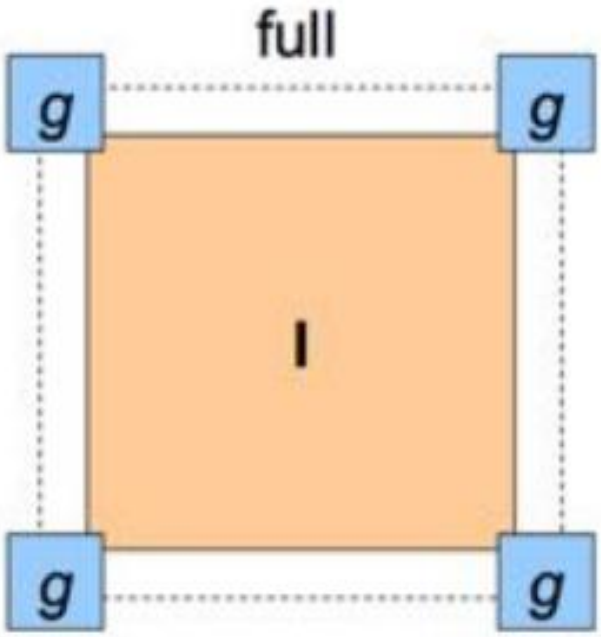
Convolution

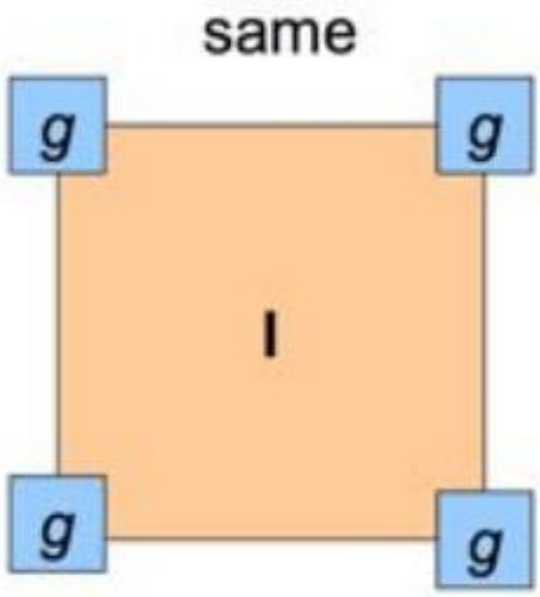
- Thông thường, kernel có dạng vuông ***(m = n), với m và n là số lẻ***.
- **Chú ý:**
  - **Tâm của nhân chập ở giữa cửa sổ chập.**
  - **Tại một vị trí bất kỳ, pixel output bằng tổng các tích giữa pixel input với các phần tử tương ứng trong kernel**

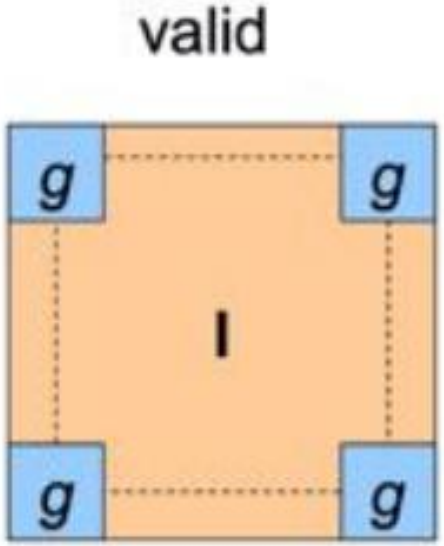
Tính chất

- Tích chập giữa $f(M_1 x N_1)$ và nhân $h(M_2 x N_2)$ có thể tạo ra các ma trận có kích thước như sau:
  - Giữ nguyên kích thước: $M_1 x N_1$ (same convolution)
  - Tăng kích thước : $(M_1+M_2-1)x(N_1+N_2-1)$ (full convolution)
  - Giảm kích thước: $(M_1-M_2+1)x(N_1-N_2+1)$ (valid convolution)

Boundary issues







Size:
(m + M – 1, n + N – 1)

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 bước thực hiện

1. Consider a 5x5 matrix A = $\begin{bmatrix} 1 & 6 & 11 & 16 & 21 \\ 2 & 7 & 12 & 17 & 22 \\ 3 & 8 & 13 & 18 & 23 \\ 4 & 9 & 14 & 19 & 24 \\ 5 & 10 & 15 & 20 & 25 \end{bmatrix}$

2. Define a 3x3 Kernel to perform spatial averaging

$$avg3 = \begin{bmatrix} 0.1111 & 0.1111 & 0.1111 \\ 0.1111 & 0.1111 & 0.1111 \\ 0.1111 & 0.1111 & 0.1111 \end{bmatrix}$$

3. Pad the matrix A with zeros

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 6 & 11 & 16 & 21 & 0 \\ 0 & 2 & 7 & 12 & 17 & 22 & 0 \\ 0 & 3 & 8 & 13 & 18 & 23 & 0 \\ 0 & 4 & 9 & 14 & 19 & 24 & 0 \\ 0 & 5 & 10 & 15 & 20 & 25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Các bước thực hiện

4. Place a 3x3 window on B and fetch the data.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 6 & 11 & 16 & 21 & 0 \\ 0 & 2 & 7 & 12 & 17 & 22 & 0 \\ 0 & 3 & 8 & 13 & 18 & 23 & 0 \\ 0 & 4 & 9 & 14 & 19 & 24 & 0 \\ 0 & 5 & 10 & 15 & 20 & 25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

5. Multiply the window with the kernel.

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 6 \\ 0 & 2 & 7 \end{bmatrix} \mathbf{x} \begin{bmatrix} 0.1111 & 0.1111 & 0.1111 \\ 0.1111 & 0.1111 & 0.1111 \\ 0.1111 & 0.1111 & 0.1111 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.111 & 0.667 \\ 0 & 0.222 & 0.778 \end{bmatrix}$$

6. Find the sum of the result obtained in step 5 and update the result. [0+0+0+0+0.1111+0.222+0+0.667+0.778]=1.778

Các bước thực hiện

7. Output Matrix (5x5) with updated value.

$$\text{Output}=\begin{bmatrix} 1.778 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

8. Now slide the window to the next position on B and fetch the data.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 6 & 11 & 16 & 21 & 0 \\ 0 & 2 & 7 & 12 & 17 & 22 & 0 \\ 0 & 3 & 8 & 13 & 18 & 23 & 0 \\ 0 & 4 & 9 & 14 & 19 & 24 & 0 \\ 0 & 5 & 10 & 15 & 20 & 25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Các bước thực hiện

9. Repeat the process of multiplying it with the kernel (step 5), finding the sum (step 6) and update the result. (Step 7)

$$\text{Output} = \begin{bmatrix} 1.778 & 4.333 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

10. Similarly, perform the steps 5 through 7 by sliding the window on the whole matrix.
11. Final updated matrix, Output =

$$\begin{bmatrix} 1.7778 & 4.3333 & 7.6667 & 11.000 & 8.4444 \\ 3.000 & 7.000 & 12.000 & 17.000 & 13.000 \\ 3.6667 & 8.000 & 13.000 & 18.000 & 13.6667 \\ 4.3333 & 9.000 & 14.000 & 19.000 & 14.3333 \\ 3.1111 & 6.3333 & 9.6667 & 13.000 & 9.7778 \end{bmatrix}$$

Exercises

1. The array below represents a small greyscale image. Compute the images that result when the image is convolved with each of the masks (a) to (h) shown. At the edge of the image use a restricted mask. (In other words, pad the image with zeroes.)

20	20	20	10	10	10	10	10	10
20	20	20	20	20	20	20	20	10
20	20	20	10	10	10	10	20	10
20	20	10	10	10	10	10	20	10
20	10	10	10	10	10	10	20	10
10	10	10	10	20	10	10	20	10
10	10	10	10	10	10	10	10	10
20	10	20	20	10	10	10	20	20
20	10	10	20	10	10	20	10	20

(a)
-1	-1	0
-1	0	1
0	1	1

(b)
0	-1	-1
1	0	-1
1	1	0

(c)
-1	-1	-1
2	2	2
-1	-1	-1

(d)
-1	2	-1
-1	2	-1
-1	2	-1

(e)
-1	-1	-1
-1	8	-1
-1	-1	-1

(f)
1	1	1
1	1	1
1	1	1

(g)
-1	0	1
-1	0	1
-1	0	1

(h)
0	-1	0
-1	4	-1
0	-1	0

2. Cho ảnh 3 bit, kích thước 5x5. Tính ảnh cường độ gradient.

IMAGE

y \ x=	0	1	2	3	4
0	3	7	6	2	0
1	2	4	6	1	1
2	4	7	2	5	4
3	3	0	6	2	1
4	5	7	5	1	2

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Image gradient

f

$\frac{\partial f}{\partial x}$

$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$

$\frac{\partial f}{\partial y}$