

CHƯƠNG 4

MẠNG NEURAL TÍCH CHẬP (P1)

Khoa Khoa học và Kỹ thuật thông tin
Bộ môn Khoa học dữ liệu

NỘI DUNG

1. Dẫn nhập.
2. Tích chập (Convolutional).
3. Padding & Strides.
4. Tích chập trên nhiều lớp (Convolutional over volumes).

Dẫn nhập

Các bài toán trong Computer Vision

- Image classification.
- Object Detection.
- Object Tracking.
- Neural style transfer.
-

Image classification



64x64

→ Cat? (0/1)

x

y

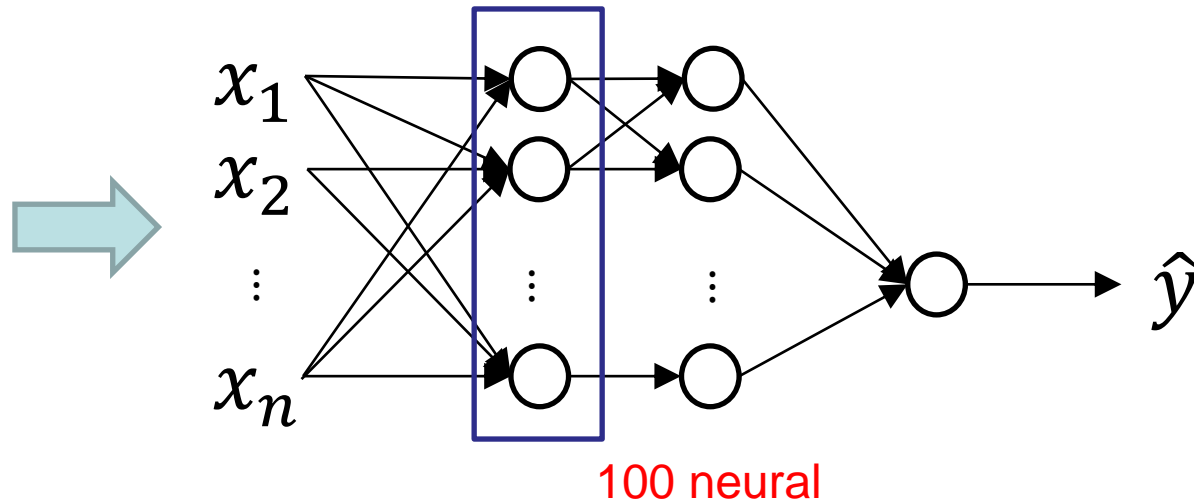
Object detection

Object detection

car



Deep neural network size

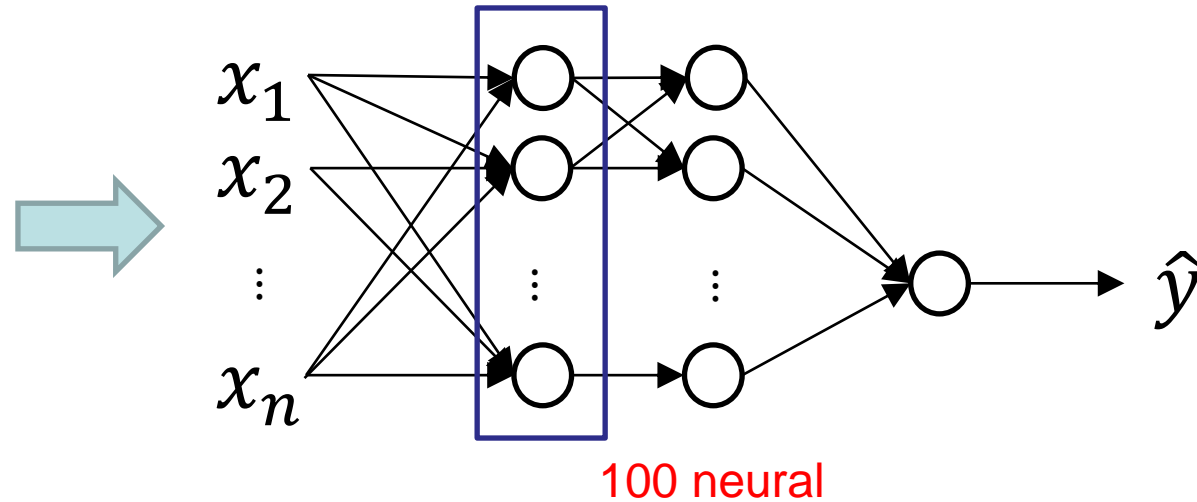


$$64 \times 64 \times 3 = 12288 \quad (1, 12288) \quad (100, 12288)$$

3 kênh màu cơ bản: RGB

*Bức ảnh 64x64 rất nhỏ
nhưng tốn khoảng 12 nghìn
trọng số.*

Deep neural network size



$1000 \times 1000 \times 3 = 3M$ $(1, 3M)$ $(100, 3M)$

Một bức ảnh 1000x1000 tốn gần 3 triệu trọng số !!

3 kênh màu cơ bản: RGB

Làm sao để giảm kích thước của 1 lớp ?

Tích chập (convolutional)

Giới thiệu

- Trong xử lý tín hiệu số, để đo độ tương quan (similarity measurement) của 2 tín hiệu, người ta dùng phép tính **cross-correlation** (tạm dịch là tương quan chéo).
- Bản chất của phép tính tương quan chéo là toán tử **dot product** – hoặc còn gọi là **scalar product**:
 - + *Input*: 2 dãy số có cùng độ dài (two equal-length sequence).
 - + *Output*: 1 con số (single number).
- Trong deep learning, phép tính **cross-correlation** được gọi là **tích chập – convolution operator**.

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

★

=

Output

4x4

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5			

4x4

$$3*1 + 1*1 + 2*1 + 0*0 + 5*0 + 7*0 + 1*(-1) + 8*(-1) + 2*(-1) = -5$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

=

Output

-5	-4		

4x4

$$0*1 + 5*1 + 7*1 + 1*0 + 8*0 + 2*0 + 2*(-1) + 9*(-1) + 5*(-1) = -4$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

=

Output

-5	-4	0	

4x4

$$1*1 + 8*1 + 2*1 + 2*0 + 9*0 + 5*0 + 7*(-1) + 3*(-1) + 1*(-1) = 0$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

=

Output

-5	-4	0	8

4x4

$$2*1 + 9*1 + 5*1 + 7*0 + 3*0 + 1*0 + 4*(-1) + 1*(-1) + 3*(-1) = 8$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10			

4x4

$$1*1 + 2*1 + 0*1 + 5*0 + 7*0 + 1*0 + 8*(-1) + 2*(-1) + 3*(-1) = -10$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2		

4x4

$$5*1 + 7*1 + 1*1 + 8*0 + 2*0 + 3*0 + 9*(-1) + 5*(-1) + 1*(-1) = -2$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	

4x4

$$8*1 + 2*1 + 3*1 + 9*0 + 5*0 + 1*0 + 3*(-1) + 1*(-1) + 7*(-1) = 2$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	3

4x4

$$9*1 + 5*1 + 1*1 + 3*0 + 1*0 + 7*0 + 1*(-1) + 3*(-1) + 8*(-1) = 3$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	3
0			

4x4

$$2*1 + 0*1 + 4*1 + 7*0 + 1*0 + 2*0 + 2*(-1) + 3*(-1) + 1*(-1) = 0$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	3
0	-2		

4x4

$$7*1 + 1*1 + 2*1 + 2*0 + 3*0 + 1*0 + 5*(-1) + 1*(-1) + 6*(-1) = -2$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	3
0	-2	-4	

4x4

$$2*1 + 3*1 + 1*1 + 5*0 + 1*0 + 6*0 + 1*(-1) + 7*(-1) + 2*(-1) = -4$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7

4x4

$$5*1 + 1*1 + 6*1 + 1*0 + 7*0 + 2*0 + 3*(-1) + 8*(-1) + 8*(-1) = -7$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

=

Output

-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3			

4x4

$$0*1 + 4*1 + 2*1 + 1*0 + 2*0 + 4*0 + 3*(-1) + 1*(-1) + 5*(-1) = -3$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3	-2		

4x4

$$1*1 + 2*1 + 4*1 + 3*0 + 1*0 + 5*0 + 1*(-1) + 6*(-1) + 2*(-1) = -2$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

*

=

Output

-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3	-2	-3	

4x4

$$3*1 + 1*1 + 5*1 + 1*0 + 6*0 + 2*0 + 7*(-1) + 2*(-1) + 3*(-1) = -3$$

Ví dụ

Gray scale image

3	0	1	2	7	4
1	5	8	9	3	1
2	7	2	5	1	3
0	1	3	1	7	8
4	2	1	6	2	8
2	4	5	2	3	9

6x6

Filter

1	0	-1
1	0	-1
1	0	-1

3x3

=

Output

-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3	-2	-3	-16

4x4

$$1*1 + 6*1 + 2*1 + 7*0 + 2*0 + 3*0 + 8*(-1) + 8*(-1) + 9*(-1) = -16$$

Ảnh hưởng của việc tích chập

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

★

1	0	-1
1	0	-1
1	0	-1

3x3

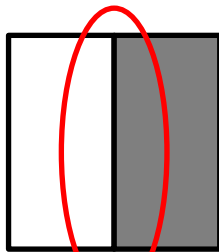
=

0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0

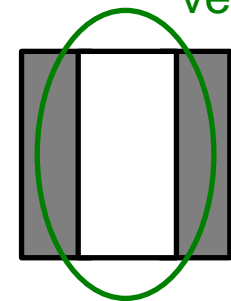
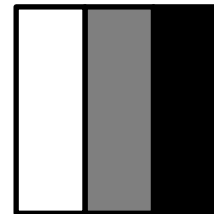
4x4

vertical edge

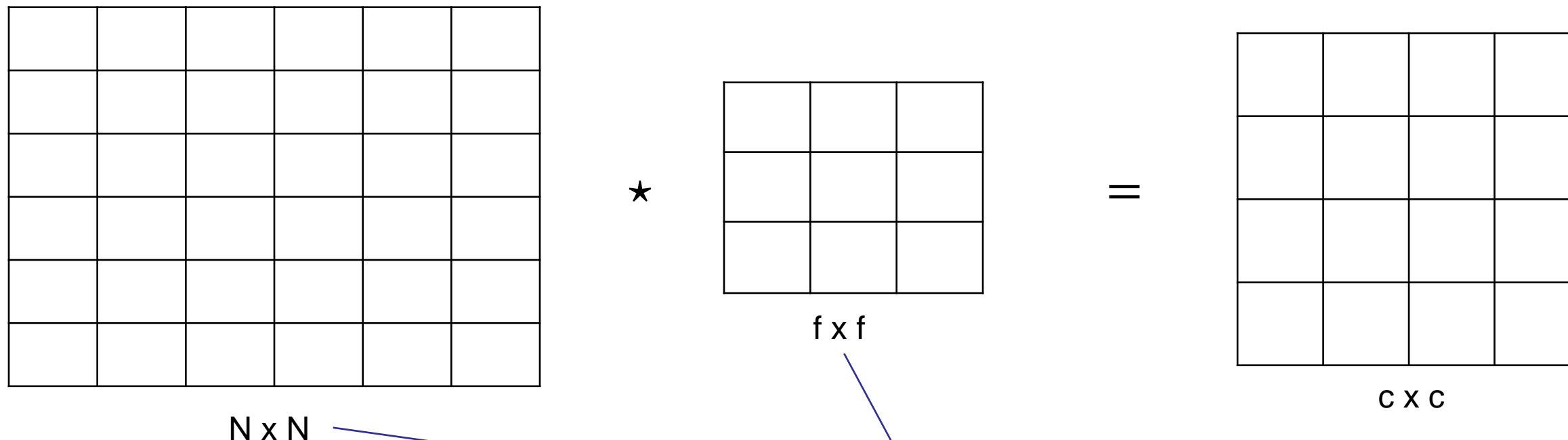
Hình thật:



vertical edge



Công thức tính ma trận tích chập



$$C = N - F + 1$$

VD: $N = 6$ (6x6)

$F = 3$ (3x3)

$\rightarrow C = 6 - 3 + 1 = 4$ (4x4)

Padding & Strides

Padding

- Padding (tạm dịch là đệm) là kỹ thuật thêm vào một hàng ảo và một cột ảo trong ma trận feature hiện tại.
- Mục tiêu là tránh các trường hợp sau:
 - + Shrinking output (làm co ma trận tích chập lại).
VD: Bức ảnh 6x6 sau khi thực hiện tích chập co lại còn 4x4
 - + Bỏ qua các thông tin về biên cạnh (edge).

Ví dụ

0	0	0	0	0	0	0	0
0	10	10	10	0	0	0	0
0	10	10	10	0	0	0	0
0	10	10	10	0	0	0	0
0	10	10	10	0	0	0	0
0	10	10	10	0	0	0	0
0	10	10	10	0	0	0	0
0	0	0	0	0	0	0	0

★

1	0	-1
1	0	-1
1	0	-1

3x3

=

	0	30	30	0	
	0	30	30	0	
	0	30	30	0	
	0	30	30	0	

Padding

8x8

(ban đầu là 6x6)

Bức ảnh ban đầu được giữ
nguyên kích thước sau khi thực
hiện tích chập nhờ padding

6x6

$$C = 8 - 3 + 1 = 6$$

Công thức tính ma trận tích chập khi có padding P

$N \times N$

★

$f \times f$

=

$C \times C$

$$C = N - F + 2P + 1$$

VD: $N = 6$ (6x6)

$F = 3$ (3x3)

$P = 1$

→ $C = 6 - 3 + 2 + 1 = 6$ (6x6)

Valid and same convolution

- **Valid:** Không dùng padding

$$C = N - F + 1$$

- **Same:** Sử dụng padding sao cho ma trận tích chập bằng với ảnh ban đầu

$$C = N - F + 2P + 1$$

Để tìm P sao cho $C = N$:

$$C = N \text{ (same)} \Leftrightarrow N - F + 2P + 1 = N$$

$$\Leftrightarrow -F + 2P + 1 = 0 \Leftrightarrow P = \frac{F-1}{2}$$

Lưu ý

- Kích thước các bộ lọc thường là các số lẻ (odd number), mục đích là tạo sự đối xứng.
- VD: 3x3, 5x5, 7x7,

Strides

- Strides là kỹ thuật “**dịch chuyển**” filter đi **N đơn vị** trên ma trận đầu vào (input data).
- Việc dịch chuyển bao nhiêu đơn vị sẽ **ảnh hưởng** đến **kích thước** của ma trận tích chập đầu ra.

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

★

=

Output

3x3

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91		

3x3

$$2*3 + 6*1 + 3*(-1) + 3*4 + 6*0 + 4*0 + 7*4 + 9*2 + 8*3 = 91$$

Strides = 2

Ví dụ

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	

3x3

$$7*3 + 9*1 + 8*(-1) + 4*4 + 8*0 + 3*0 + 6*4 + 7*2 + 8*3 = 100$$

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	88

3x3

$$6*3 + 7*1 + 8*(-1) + 2*4 + 4*0 + 9*0 + 9*4 + 3*2 + 7*3 = 88$$

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	88
69		

3x3

$$3*3 + 7*1 + 4*(-1) + 4*4 + 8*0 + 2*0 + 8*4 + 3*2 + 1*3 = 69$$

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	88
69	91	

3x3

$$8 \cdot 3 + 3 \cdot 1 + 1 \cdot (-1) + 3 \cdot 4 + 6 \cdot 0 + 8 \cdot 0 + 8 \cdot 4 + 6 \cdot 2 + 3 \cdot 3 = 91$$

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	88
69	91	117

3x3

$$8*3 + 6*1 + 3*(-1) + 9*4 + 3*0 + 4*0 + 7*4 + 4*2 + 6*3 = 117$$

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	88
69	91	117
44		

3x3

$$4*3 + 3*1 + 0*(-1) + 2*4 + 2*0 + 1*0 + 1*4 + 4*2 + 3*3 = 44$$

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	88
69	91	117
44	72	

3x3

$$1*3 + 4*1 + 3*(-1) + 8*4 + 1*0 + 9*0 + 3*4 + 9*2 + 2*3 = 72$$

Ví dụ

Gray scale image

2	3	7	4	6	2	9
6	6	9	8	7	4	3
3	4	8	3	8	9	7
7	8	3	6	6	3	4
4	2	1	8	3	4	6
3	2	4	1	9	8	3
0	1	3	9	2	1	4

7x7

Filter

3	4	4
1	0	2
-1	0	3

3x3

*

=

Output

91	100	88
69	91	117
44	72	74

3x3

$$3 \cdot 3 + 9 \cdot 1 + 2 \cdot (-1) + 4 \cdot 4 + 8 \cdot 0 + 1 \cdot 0 + 6 \cdot 4 + 3 \cdot 2 + 4 \cdot 3 = 74$$

Công thức tính ma trận tích chập khi có padding P và Stride S

$N \times N$

★

$f \times f$

=

$C \times C$

$$C = \{(N - F + 2P)/S\} + 1$$

VD: $N = 7$ (7x7)

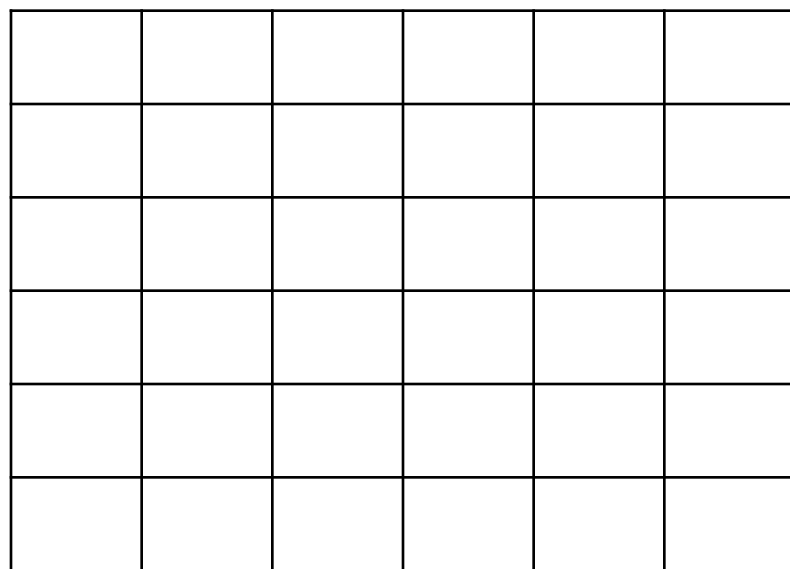
$F = 3$ (3x3)

$P = 1$; $S = 2$

$$\rightarrow C = ((7 - 3 + 2)/2) + 1 = 3 \text{ (3x3)}$$

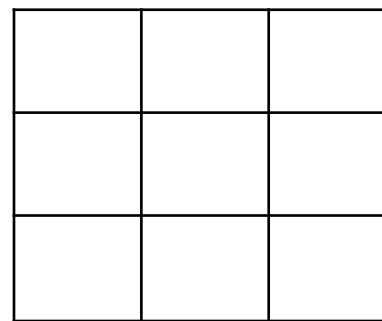
Tích chập trên nhiều lớp

Một lớp (1-layer) vs Một filter



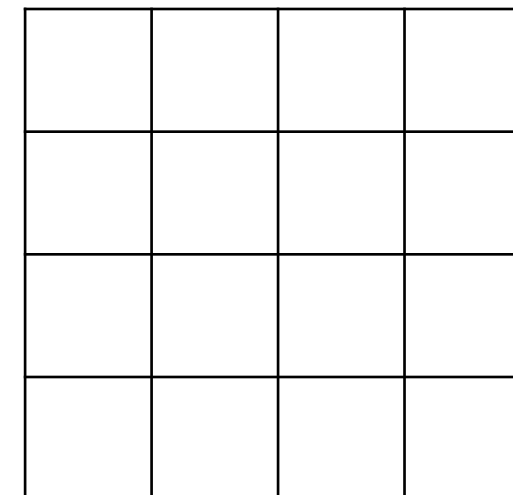
$N \times N$

★



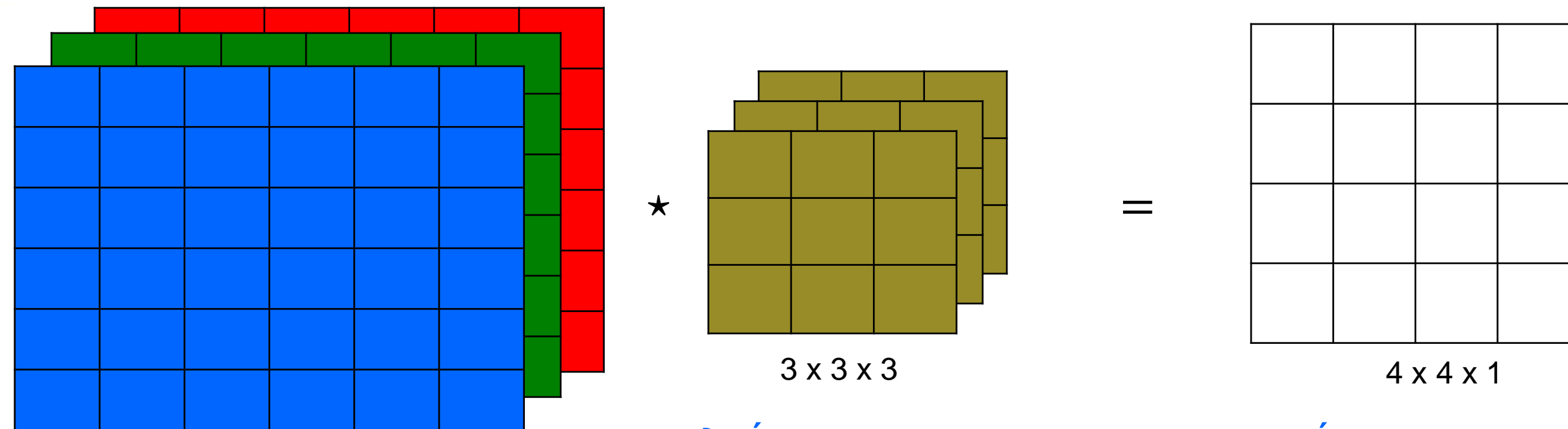
$f \times f$

=



$C \times C$

Ba lớp (3-layer) – 1 filter

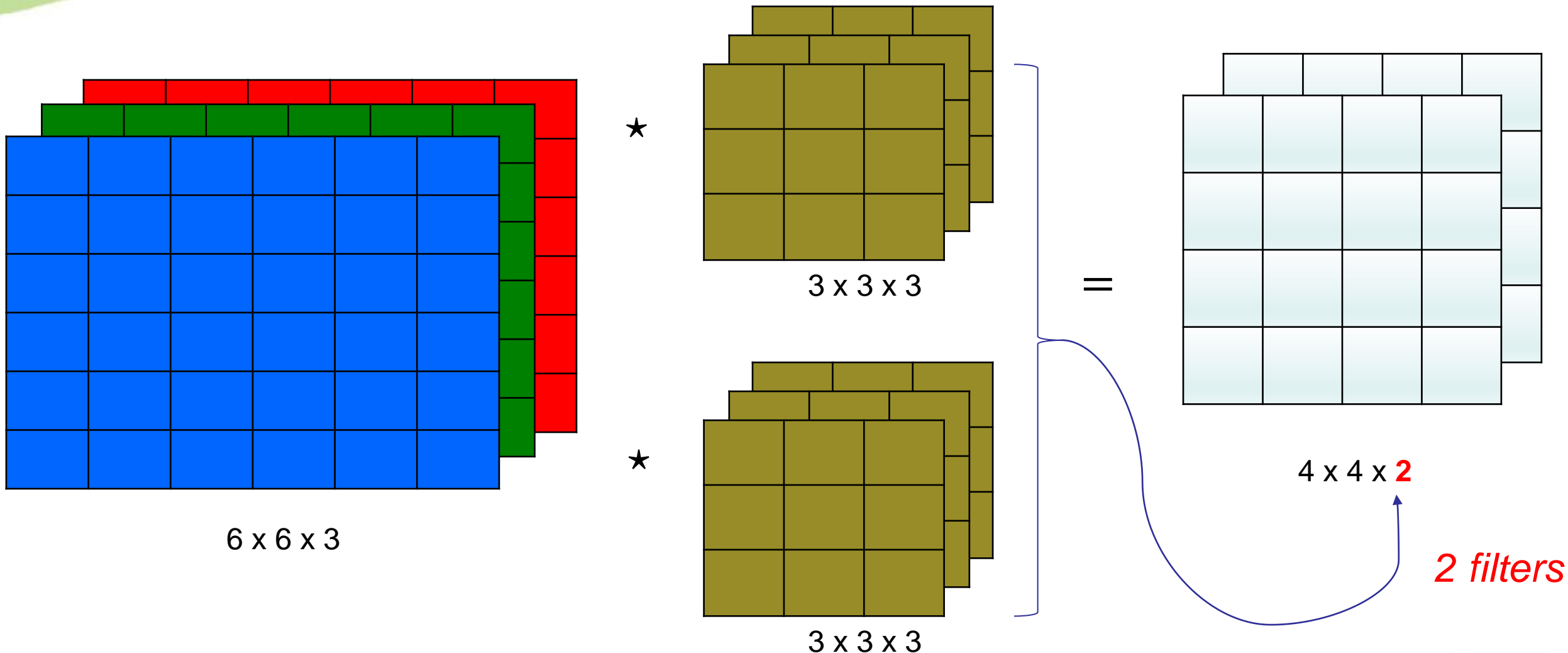


6 x 6 x 3

Chỉ số này đôi khi còn được gọi là số kênh
(Number of chanel)

3 ứng với 3 kênh màu cơ bản: RGB (Red,
Green, Blue)

Ba lớp (3-layer) – 2 filters



Tổng quát

— Đầu vào:

+ Bức ảnh kích thước $N \times N \times N_c$, với N_c là số kênh (Number of chanel).

+ N_f bộ lọc kích thước: $F \times F \times N_c$.

+ Các thông số khác như: Padding (P), Strides (S).

— Ma trận tích chập đầu ra: $C \times C \times N_f$, với N_f là số lượng bộ lọc sử dụng ($N_f = \text{Number of filters}$), trong đó:

$$C = \{(N - F + 2P)/S\} + 1$$

Ví dụ

- Ảnh đầu vào: 6x6x3.
- 2 Bộ lọc có kích thước: 3x3x3.
- Padding: $S = 0$ và Stride là 1.
- Ma trận đầu ra: $C = ((N - F + 2P) / S) + 1 = ((6 - 3 + 0) / 1) + 1 = 4$.
- Có 2 bộ lọc.
- Kích thước ma trận là: 4x4x2

TÀI LIỆU THAM KHẢO

1. Khoá học *Neural Network and Deep learning*, deeplearning.ai.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep learning*, MIT Press, 2016.
3. Andrew Ng., *Machine Learning Yearning*. Link:
<https://www.deeplearning.ai/machine-learning-yearning/>
4. Vũ Hữu Tiệp, *Machine Learning cơ bản*, NXB Khoa học và Kỹ thuật, 2018.