1. Tác dụng của việc sử dụng padding trong Convolutional Neural Networks

(CNN) ?

A. Giúp hạn chế việc giảm kích thước ảnh sau các phép tích chập

B. Bảo toàn thông tin ở các biên của ảnh

**C. Cả A và B đều đúng**

D. Cả A và B đều sai

2. Thứ tự cơ bản của các lớp trong Convolutional Neural Networks (CNN) ?

**A. Convolutional Layer, ReLu layer, Pooling Layer, Fully Connected Layer**

B. Pooling Layer, Convolutional Layer, ReLu layer, Fully Connected Layer

C. Pooling Layer, ReLu layer, Convolutional Layer, Fully Connected Layer

D. Convolutional Layer, Pooling Layer, ReLu layer, Fully Connected Layer

3. Convolutional Layer (Lớp tích chập) có tác dụng gì ?

A. Chuyển tất cả các giá trị âm thành 0, chỉ giữ lại các giá trị dương để tạo ra tính phi

tuyến tính.

B. Giảm số lượng tham số và tính toán, đồng thời tránh overfitting.

**C. Trích xuất các đặc trưng như cạnh, góc, hoặc chi tiết phức tạp**

D. Không có đáp án chính xác

4. Với một hình ảnh có kích thước 7x7 và một filter/kernel có kích thước 3x3 thì

sẽ cho ra ouptut có kích thước là bao nhiêu sau khi tích chập ( không sử dụng

padding và s(stride) = 1) ?

**A. 4x4**

B. 3x3

C. 5x5

D. 8x8

5. Với một hình ảnh có kích thước 7x7 và một filter/kernel có kích thước 3x3 thì

sẽ cho ra ouptut có kích thước là bao nhiêu sau khi tích chập với p(padding)

= 1 và s(stride) = 2 ?

**A. 4x4**

B. 3x3

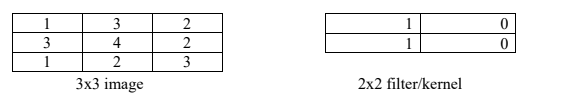
C. 5x5

D. 8x8

II. Tự luận

Để thực hiện phép tích chập giữa hình ảnh 3x3 và filter/kernel 2x2 với stride (s) = 1 và padding (p) = 0, chúng ta sẽ làm theo các bước sau:

Ta có:



Phép tính chập:

Với stride = 1 và padding = 0, chúng ta sẽ di chuyển kernel qua hình ảnh và tính toán giá trị đầu ra như sau:

1. Vị trí đầu tiên (top-left):

(1\*1 + 3\*0 + 3\*1 + 4\*0) = 1 + 0 + 3 + 0 = 4

1. Vị trí thứ hai (top-center):

(3\*1 + 2\*0 + 4\*1 + 2\*0) = 3 + 0 + 4 + 0 = 7

1. Vị trí thứ ba (top-right):

(2\*1 + 0\*0 + 2\*1 + 0\*0) = 2 + 0 + 2 + 0 = 4

1. Vị trí thứ tư (middle-left):

(3\*1 + 4\*0 + 1\*1 + 2\*0) = 3 + 0 + 1 + 0 = 4

1. Vị trí thứ năm (middle-center):

(4\*1 + 2\*0 + 2\*1 + 3\*0) = 4 + 0 + 2 + 0 = 6

1. Vị trí thứ sáu (middle-right):

(2\*1 + 0\*0 + 3\*1 + 0\*0) = 2 + 0 + 3 + 0 = 5

1. Vị trí thứ bảy (bottom-left):

(1\*1 + 2\*0 + 1\*1 + 2\*0) = 1 + 0 + 1 + 0 = 2

1. Vị trí thứ tám (bottom-center):

(2\*1 + 0\*0 + 2\*1 + 3\*0) = 2 + 0 + 2 + 0 = 4

1. Vị trí thứ chín (bottom-right):

(3\*1 + 0\*0 + 0\*1 + 0\*0) = 3 + 0 + 0 + 0 = 3

**Kết quả:**

Tổng hợp các giá trị đầu ra, chúng ta có ma trận đầu ra (output) 2x2:

|  |  |
| --- | --- |
| 4 | 7 |
| 4 | 6 |

**Kích thước đầu ra**

Dựa vào kích thước đầu vào và kích thước kernel, kích thước đầu ra sẽ là:

* Kích thước đầu ra = (W−F)/S+1(*W*−*F*)/*S*+1
* Với W=3*W*=3, F=2*F*=2, và S=1*S*=1:
* Kích thước đầu ra = (3−2)/1+1=2(3−2)/1+1=2

Vậy kích thước đầu ra sẽ là **2x2**