

D12 : CNN分類器架構：步長、填充



簡報閱讀



範例與作業



問題討論



深度學習理論與實作



深度學習理論與實作

CNN原理：卷積、步長、填充

重要知識點

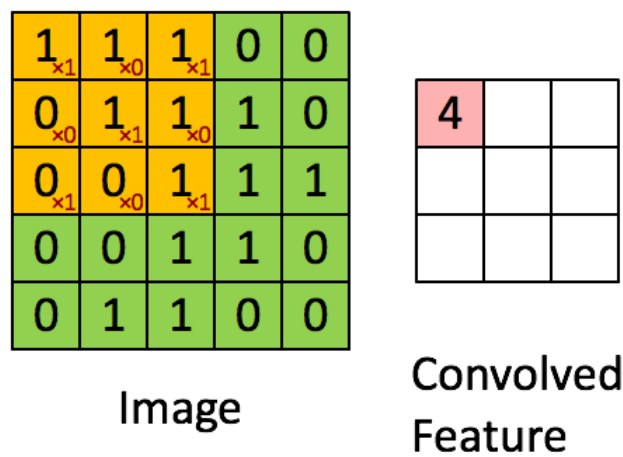
- 步長(Strides)與填充(Padding)的原理

步長與填充

我們先來看看下圖：

步長(Strides)：圖中 Kernel 的步長 (Strides) 在 height、width 均為 1，可以看到黃色 Kernel 往右、下都是一格像素。

填充 (Padding)：圖中並沒有使用任何 Padding，因此可以看到原圖周圍並沒有補 0 的像素，而輸出的 Feature map 長寬也下降。



參考來源：滑動卷積的原理

那步長與填充的運作原理是什麼呢？

步長(Strides)

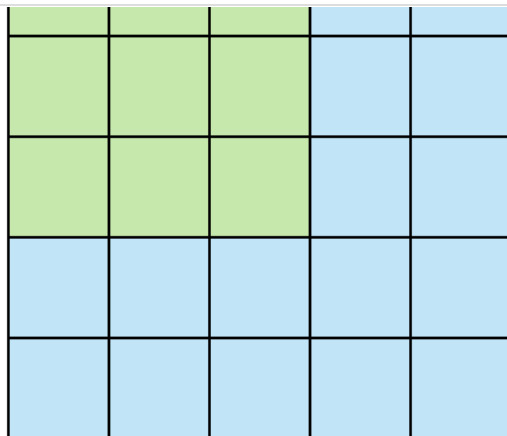
```
Init signature: Convolution2D(filters, kernel_size, strides=(1, 1), padding='valid', data_format=None, dilation_rate=(1, 1), activation=None, use_bias=True, kernel_initializer='glorot_uniform', bias_initializer='zeros', kernel_regularizer=None, bias_regularizer=None, activity_regularizer=None, kernel_constraint=None, bias_constraint=None, **kwargs)
Docstring:
2D convolution layer (e.g. spatial convolution over images).
```

Strides：控制 Kernel 在圖像不同 Dimention 上移動的距離。

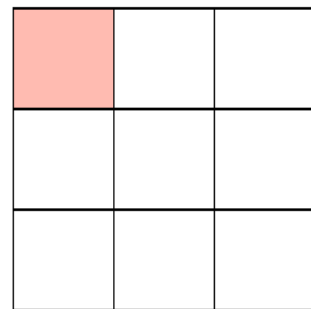
Keras Convolution2D：其中有一個可調參數為 Strides，可以針對Height、Width 賦予不同的值，藉此控制輸出 Feature map 高、寬的尺度變化。

下圖中運用不同的步長(1,1)與(2,2)，可以發現輸出的 Feature map 尺寸也有所不同。

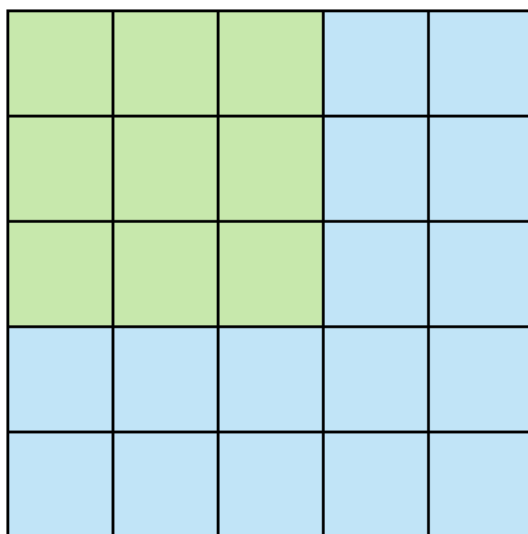




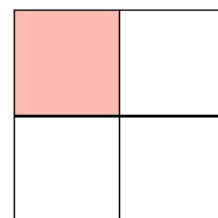
Stride 1



Feature Map



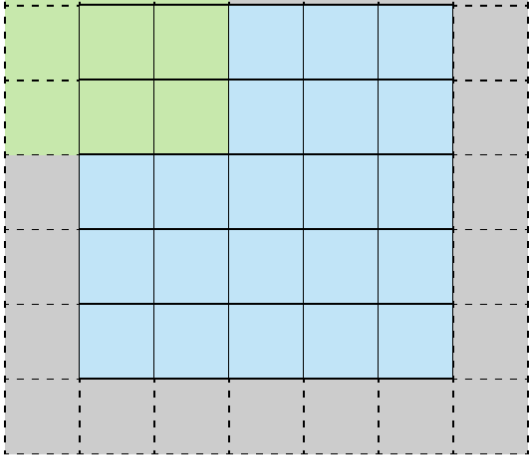
Stride 2



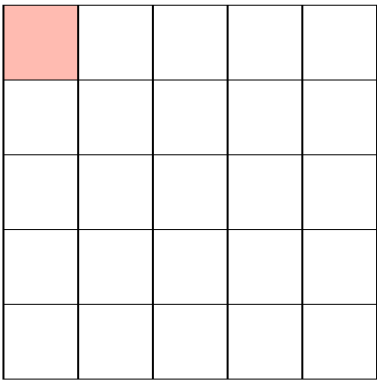
Feature Map

填充 (Padding)

Padding：Padding 的用途主要在於避免圖像尺寸下降，而為了避免干擾圖像資訊，通常 Padding 為補 0 的像素，而 **Padding=1** 就是在圖像周圍補一圈值為 0 的像素，也就是圖中灰色的區域。



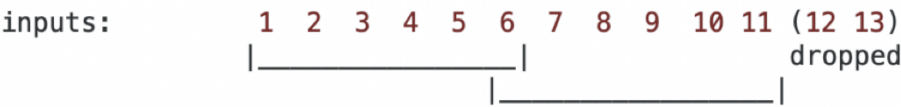
Stride 1 with Padding



Feature Map

而我們在操作 Keras 的 Convolution2D 時，會發現默認為『Valid』而一多數人會用『SAME』，這是什麼意思呢？

- "VALID" = without padding:



Valid：就是不去 Padding，多的像素直接捨去。像是上圖中，可以想像原圖為13*13，kernel 大小為6*6，步長為(5,5)，當Kernel要跨出第二步時，只剩下 **2 個像素**(12、13)，a就直接捨去。

SAME：透過補邊讓輸出長寬==原圖長寬/Strides，什麼意思呢？假如我們使用Strides=(1,1)，那麼不管使用多大的Kernel，輸出 Feature map 的寬、高等於輸入影像的寬高。

公式化如下：

『SAME』

$$\text{out_height} = \text{ceil}(\text{float}(\text{in_height}) / \text{float}(\text{strides}[1]))$$

$$\text{out_width} = \text{ceil}(\text{float}(\text{in_width}) / \text{float}(\text{strides}[2]))$$

『Valid』

$$\text{out_height} = \text{ceil}(\text{float}(\text{in_height} - \text{filter_height} + 1) / \text{float}(\text{strides}[1]))$$

$$\text{out_width} = \text{ceil}(\text{float}(\text{in_width} - \text{filter_width} + 1) / \text{float}(\text{strides}[2]))$$

輸出Feature map尺寸

$$\text{Output} = (\text{Input} + 2 * \text{padding} - \text{Kernel_Size}) / \text{Stride} + 1$$


推薦延伸閱讀

DeepLearning.ai- C4W1L01

吳恩達課程

https://cs230n.stanford.edu/media/mpeg-4...cs230.stanford.edu


C4W1L01 Computer Vision



Stanford-CS231n^[SEP]
史丹福大學課程

cs231n_2017_lecture1.pdf
http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture1.pdfcs231n.stanford.edu

Lecture 1 | Introduction to Convolutional Neural Networks for Visual Recognition



兩個課程中皆介紹許多DeepLearning CV領域相關知識，有些部分超過課程進度，學員們可以斟酌觀看，不懂的部分也可以直接在共學社團上詢問。

解題時間



[下一步：閱讀範例與完成作業](#)