

Open in app ↗

Medium

 Search

★ Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



Calculating Output dimensions in a CNN for Convolution and Pooling Layers with KERAS

Virajdatt Kohir · [Follow](#)

3 min read · Jun 25, 2020

Listen

Share

More

This article outlines how an input image changes as it passes through the Convolutional-Layers and Pooling layers in a Convolutional Neural Network (CNN) and as a bonus also has the calculation of the number of parameters. The article assumes that you are familiar with the fundamentals of KERAS and CNN's.

If you are new or just starting with CNN's I recommend these following sources:-

1. A great intro video on working of CNN
http://brohrer.github.io/how_convolutional_neural_networks_work.html
2. Official Keras book from its creator François Chollet
<https://www.manning.com/books/deep-learning-with-python>

Calculating the output when an image passes through a Convolutional layer:-

NOTE:- All matrices are square i.e the image width = height.

Parameters that influence the output shape :-

1. The input dimensions of the image — $\rightarrow I$ ($i \times i$)
2. The size of filter/kernel — $\rightarrow F$ ($f \times f$)
3. Strides — $\rightarrow S$ (integer)

4. Padding — > P (integer)

5. Depth/Number of feature maps/activation maps — > D (integer)

Convolution Output dimension = $[(I - F + 2 * P) / S] + 1 \times D$ > Formula1

NOTE:- The “x D” above doesn’t stand for multiplication operation but it depicts the depth or the number of activation maps.

Let us take a look at an example with python snippet: -

- An input image, I with dimensions (32x32x3) -An input image 32 pixel wide and 32 pixel in height with 3 channels i.e, (I =32),
- A filter size 3x3 (F=3)
- Stride is1 (S =1),
- Zero padding (P=3), and
- Depth /feature maps are 5 (D =5)

The output dimensions are = $[(32 - 3 + 2 * 0) / 1] + 1 \times 5 = (30 \times 30 \times 5)$

Keras Code snippet for the above example

```
import numpy as np
from tensorflow import keras
```

```
model = keras.models.Sequential()
```

```
#here in the snippet below
#D = 5 (first parameter)
#Stride= (1,1) by default
```

```
model.add(keras.layers.Conv2D(5, kernel_size=3, activation='relu',
input_shape=(32, 32, 3)))
```

```
model.summary()
```

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 30, 30, 5)	140
Total params: 140		
Trainable params: 140		
Non-trainable params: 0		

```
#No of Parameter calculation
#Kernel Size = (3x3) and we have three channles and the one bias term
#5 stands for the 5 filters
```

```
(3*3*3+1)*5
```

```
140
```

No of Parameter calculation, the kernel Size is (3x3) with 3 channels (RGB in the input), one bias term, and 5 filters.

Parameters = (FxF * number of channels + bias-term) * D

In our example Parameters = (3 * 3 * 3 + 1) * 5 = 140

Calculating the output when an image passes through a Pooling (Max) layer:-

For a pooling layer, one can specify only the filter/kernel size (F) and the strides (S).

Pooling Output dimension = [(I - F) / S] + 1 x D

Note Depth, D will be same as the previous layer (i.e the depth dimension remains unchanged, in our case D=5) —> Formula2

1. Let F = 2 (2x2 window)
2. Stride, S = 2
3. Depth, D = 5 (depth from the previous layer)

In our example we have

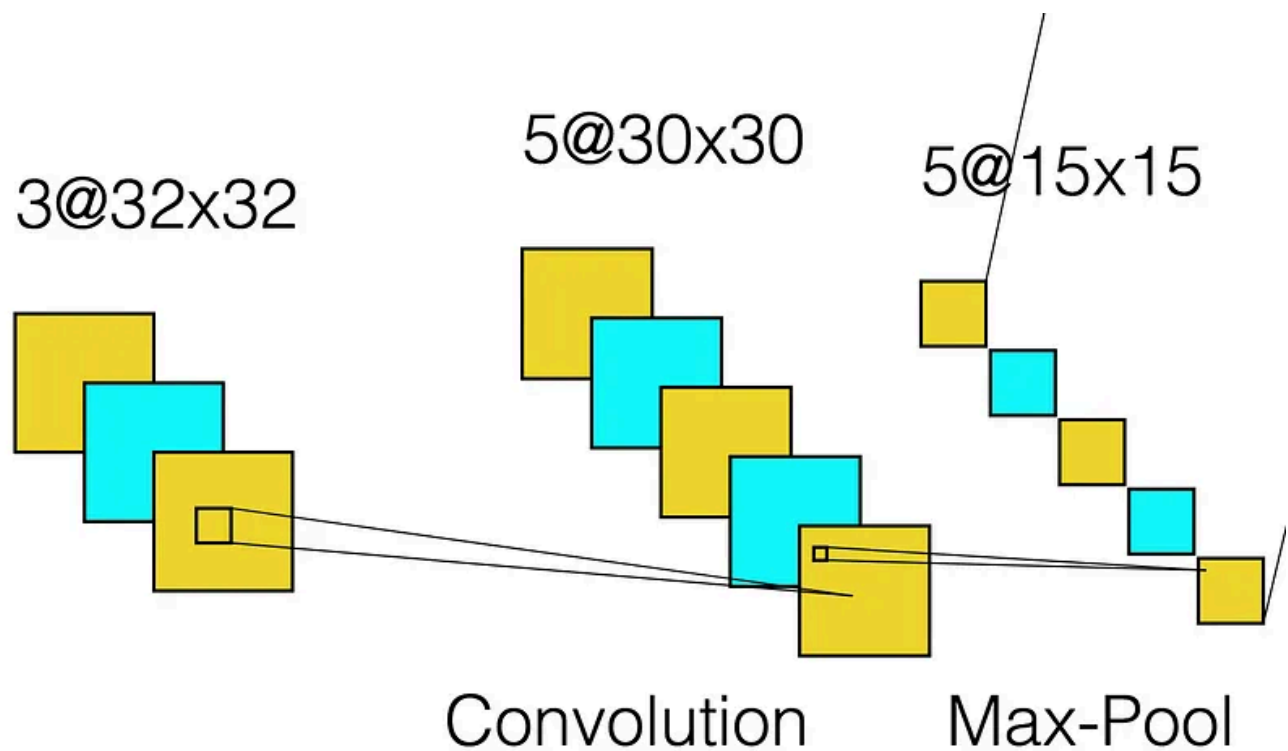
Output = [(30-2) / 2] + 1 x D = (15x15x5)

```
#default strides is 2 in pooling layer)
model.add(keras.layers.MaxPooling2D(pool_size=(2, 2)))

model.summary()
```

Layer (type)	Output Shape	Param #
conv2d_4 (Conv2D)	(None, 30, 30, 5)	140
max_pooling2d_3 (MaxPooling2D)	(None, 15, 15, 5)	0
Total params: 140		
Trainable params: 140		
Non-trainable params: 0		

Note:- Since pooling operation is a fixed function it introduces no additional parameters.



A visual summary of the 2 operations in our example.

The same formula1 and formula2 are applicable as the depth grows.

Thank you for reading this article, please let me know what your thoughts are down below in comments.

REFERENCES:-

1. Great Notes on CNN from Stanford <https://cs231n.github.io/convolutional-networks/>
2. Detailed coverage on Convolution Arithmetic <https://arxiv.org/pdf/1603.07285.pdf>
3. A great theoretical book for Deep Learning <https://www.deeplearningbook.org/>

Cnn

Convolution Neural Net

Deep Learning

Keras

Computer Vision

Public domain.

[Follow](#)

Written by Virajdatt Kohir

137 Followers · 4 Following

AI in health care with research focused on Deep Learning and LLM. I also love to talk about Machine Learning Engineering. A student for life.

Responses (2)



What are your thoughts?

[Respond](#)

Dev Nandan Anoop
over 1 year ago



What happens if our 'I' value is an odd integer? For example I have the I value to be 11 with a filter size of 2 and stride value 2. What does the dimension of the layer become then?

[Reply](#)

Sai Rishab H S
over 2 years ago



At the begining u have missed padding . U have mentioned $p=3$ and while populating to the formula u have put 0 instead of . Might be because we were carried away by word zero padding ;)

great writing cheers!!



Reply

More from Virajdatt Kohir



Virajdatt Kohir

Pandas Equivalents in Pyspark

This year I have been using pyspark extensively at my job. It's been nearly 6 months and I have grown to love pyspark for it's efficiency...

Jul 13, 2023



21





 Virajdatt Kohir

Pandas, tips to deal with huge datasets!

I have been using pandas for more than 4 years now and have always done all my ETL using pandas before training a machine learning model...

Feb 20, 2022  104



 Virajdatt Kohir

Incremental Machine Learning for Streaming data with River: Part 1

The amount of data generated, processed and analyzed every day is increasing at a tremendous pace. Modern smart devices(ex: cell phones...

May 27, 2022 🖱 13



Virajdatt Kohir

Starting with TensorFlow Datasets -part 2; Intro to tfds and its methods

After discussing about the tf.data pipeline API's, in this article I want to talk about the TensorFlow Datasets library. If you are new to...

Jan 10, 2022 🖱 103

[See all from Virajdatt Kohir](#)

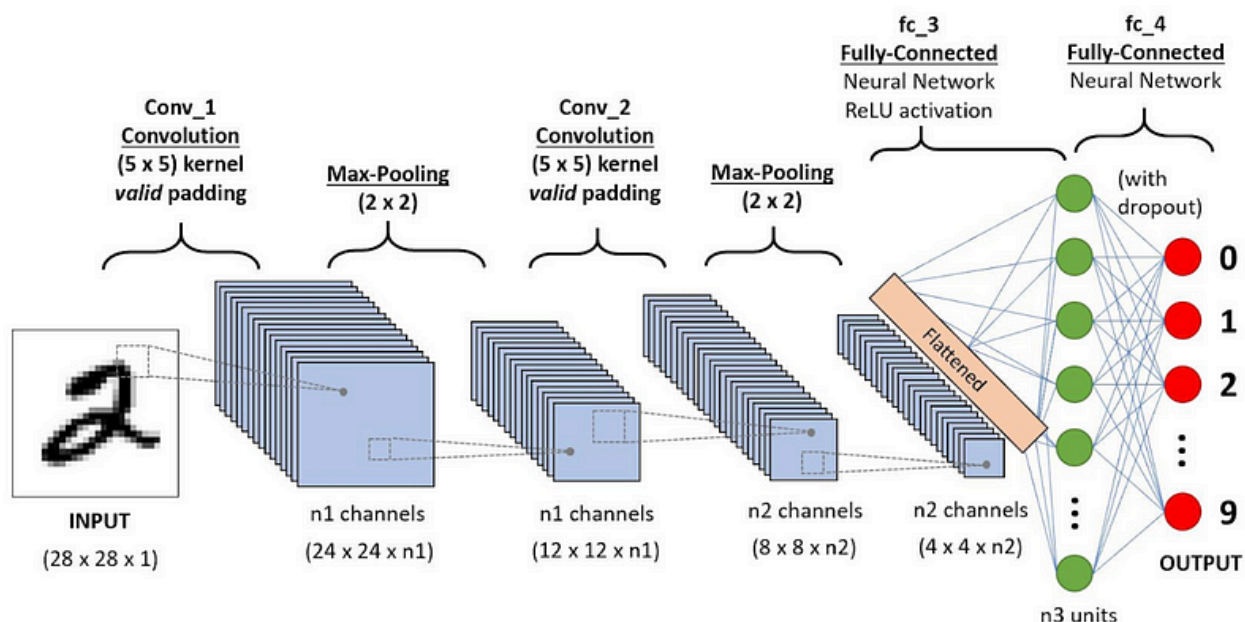
Recommended from Medium

J Jo Wang

Deep Learning Part 2—Neural Network and the critical Activation Functions

Neural Network Structure

★ Jun 29 🖱 3



In Lumos by Luqman Zaceria

Understanding Convolutional Neural Networks (CNNs)

Hey everyone! We're going to explore one of the most influential and powerful tools in the world of deep learning: Convolutional Neural...

★ Aug 5 🖱 73



Lists



Practical Guides to Machine Learning

10 stories · 2095 saves



Natural Language Processing

1862 stories · 1489 saves



data science and AI

40 stories · 300 saves



Staff picks

789 stories · 1509 saves

Padding and Strides in CNN

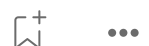
Minhaz Chowdhury

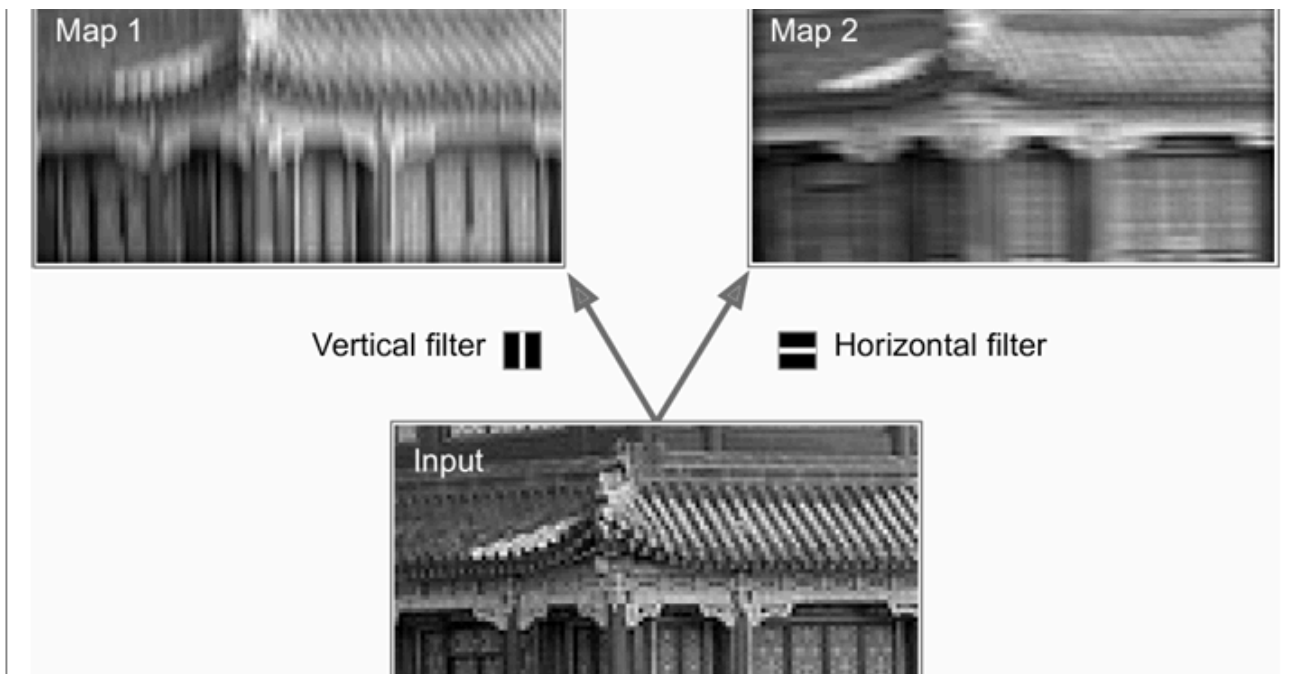
Minhaz Chowdhury

Padding and Strides in CNN

In Convolutional Neural Networks (CNNs), padding and strides are important concepts that determine how the convolution operation is applied...

Aug 14 🖱 11



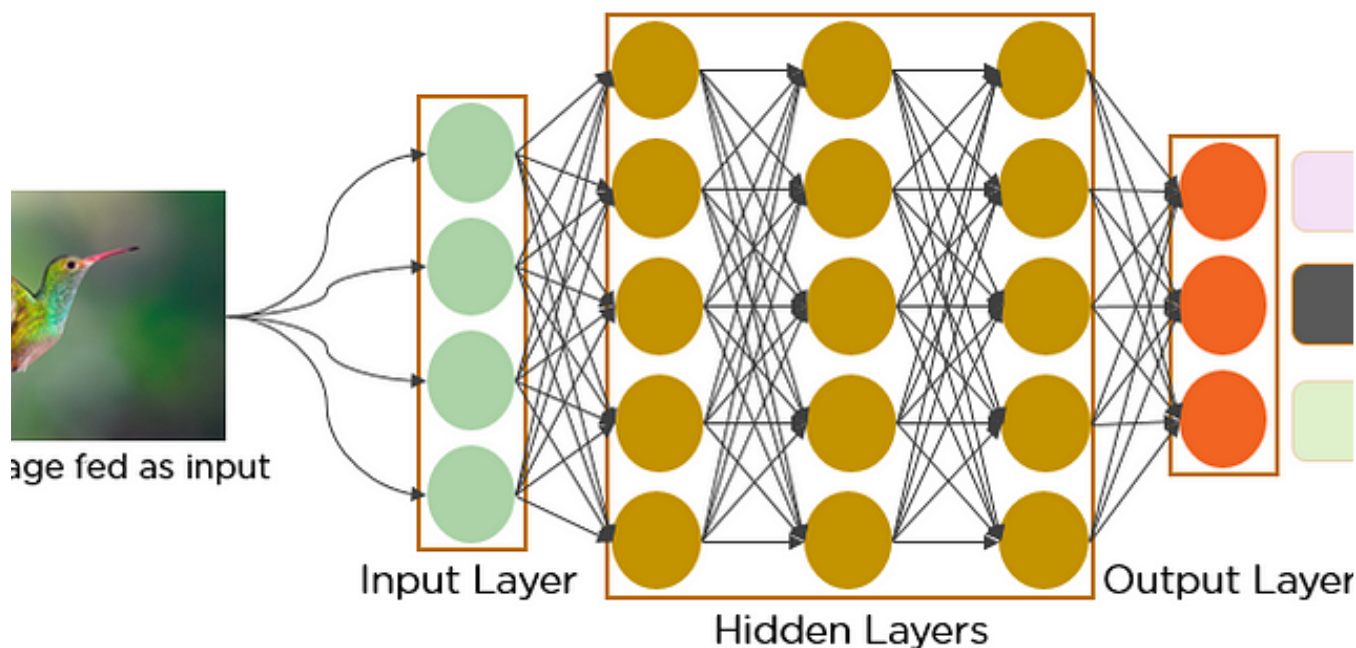



 Sanjay Dutta

Exploring Filters and Feature Maps in Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) have emerged as a powerful tool in the field of computer vision, excelling in tasks like image...

Oct 7  11

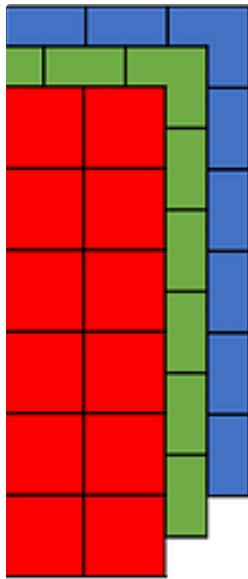


 Rishabh Singh

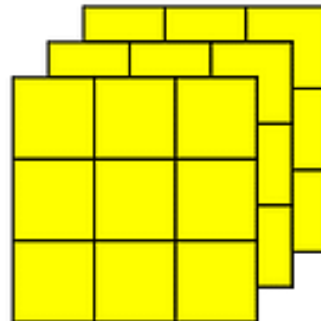
Convolutional Neural Network (CNN)—Part 1

When classifying images, traditional neural networks struggle because each pixel is treated as an independent feature, which misses...

Nov 6  100  2



*



=

$3 \times 3 \times 3$



Abhishek Jain

Detailed understanding about convolution operation in a coloured image with the help of 3D kernels

REMEMBER : Convolution operation only happens if the input image depth and kernels depth is same

Jul 3  2



See more recommendations