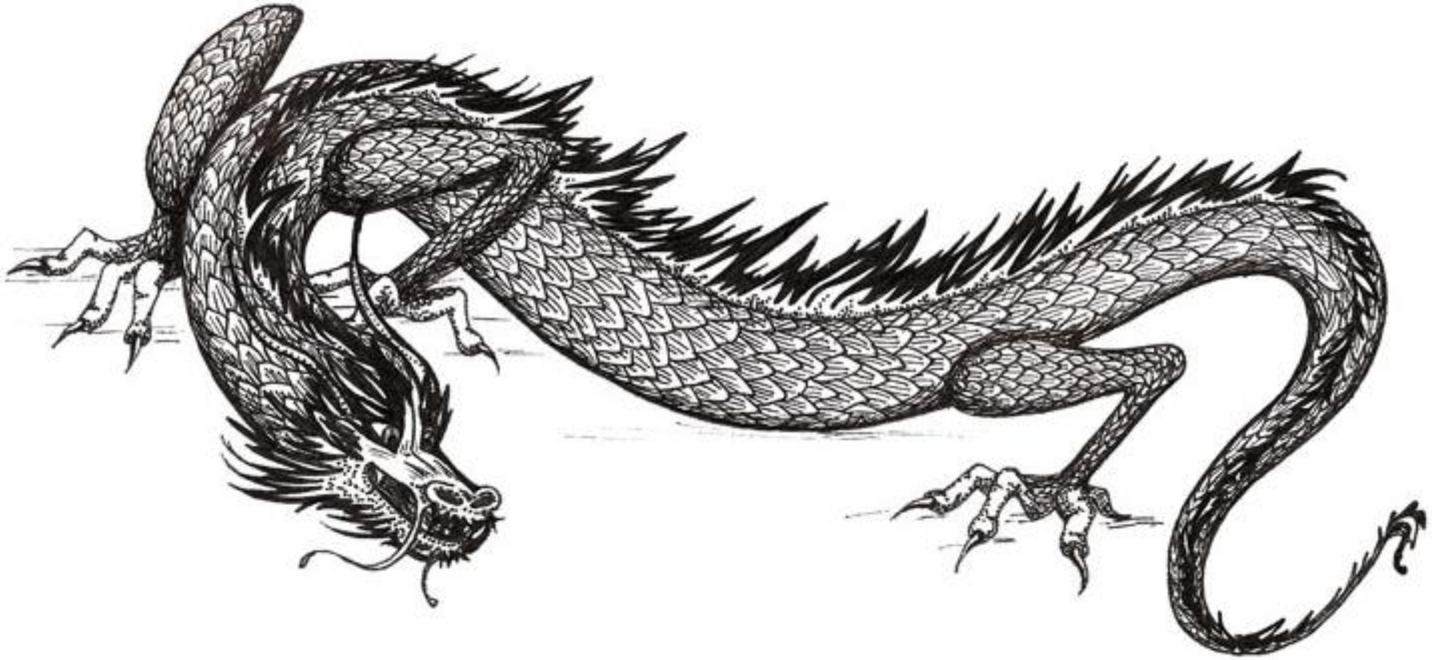
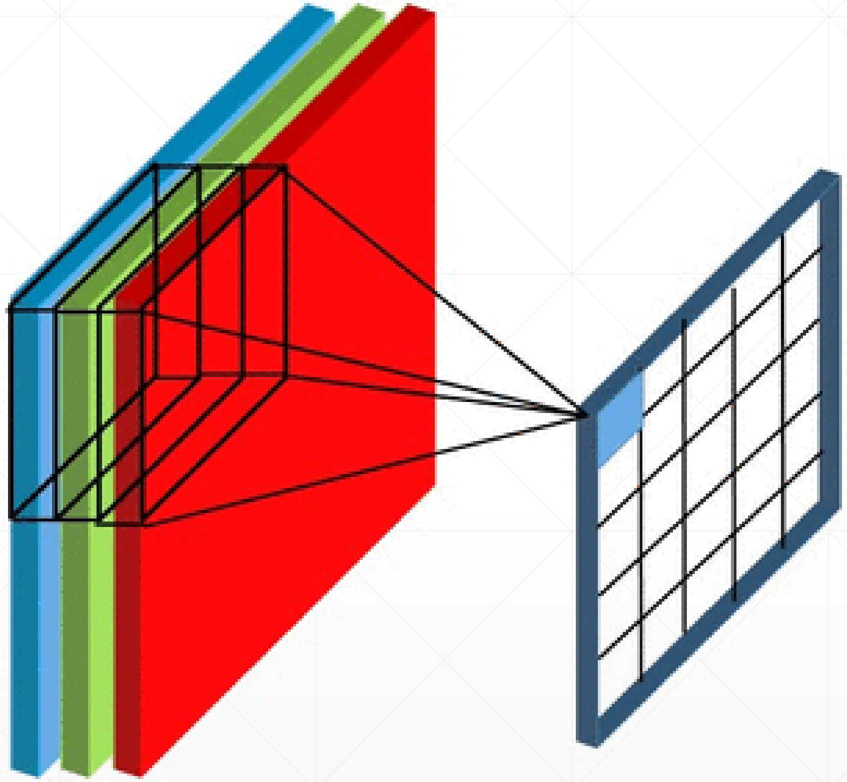


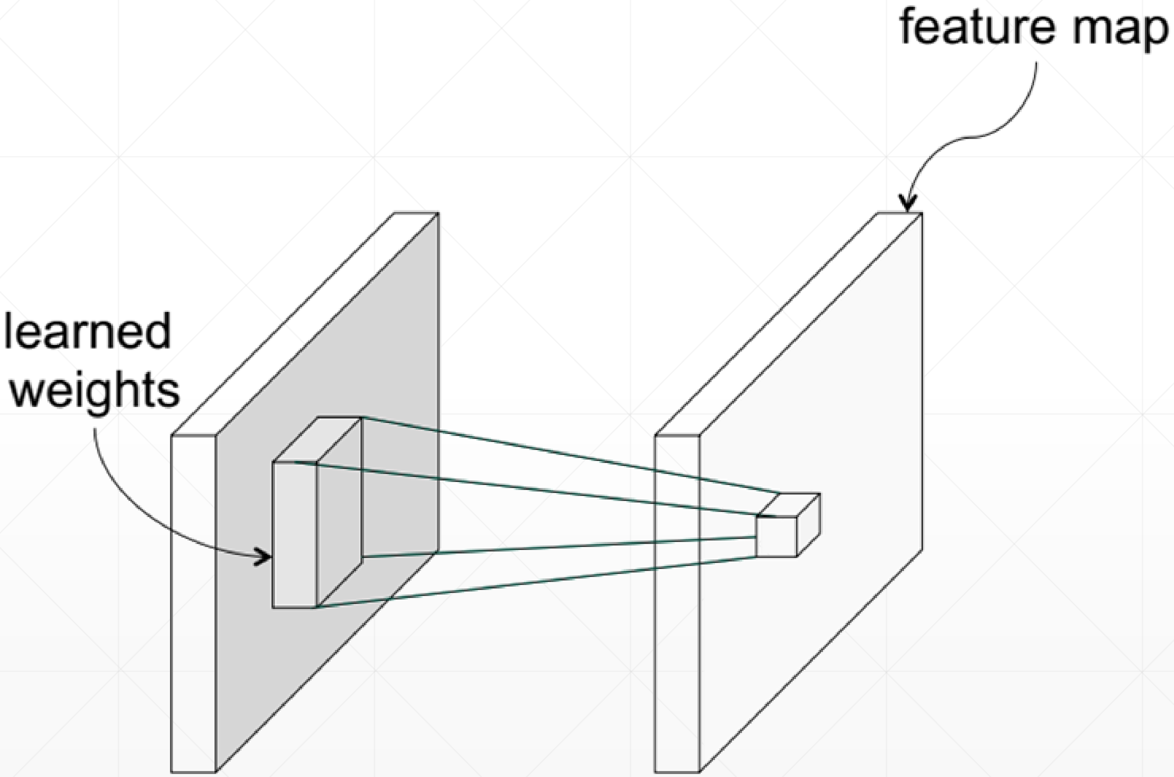
卷积神经网络

主讲：龙良曲

2D Convolution



Kernel size



1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

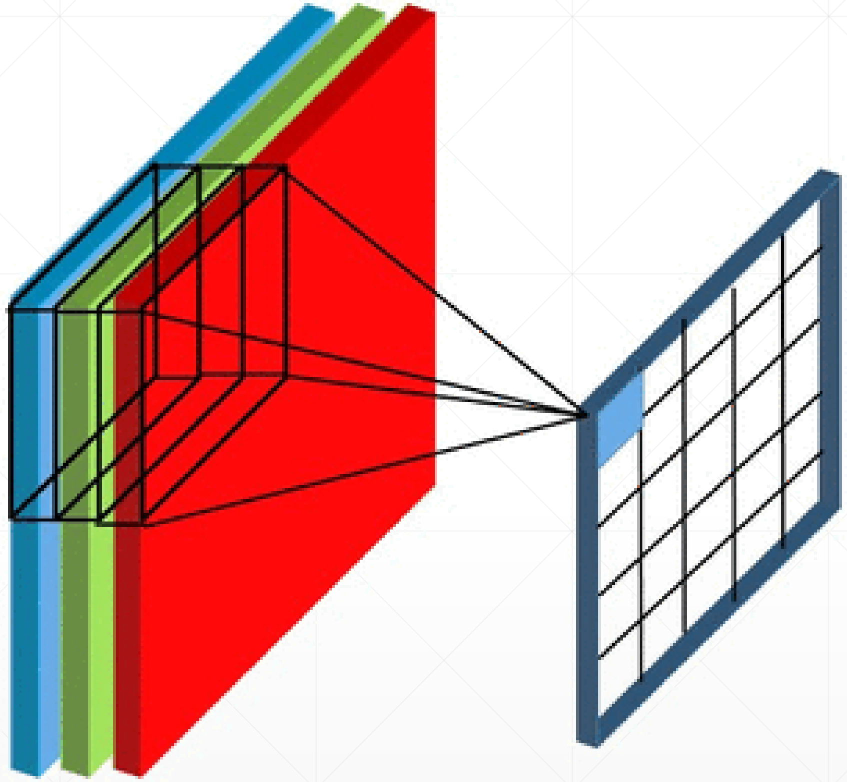
4		

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

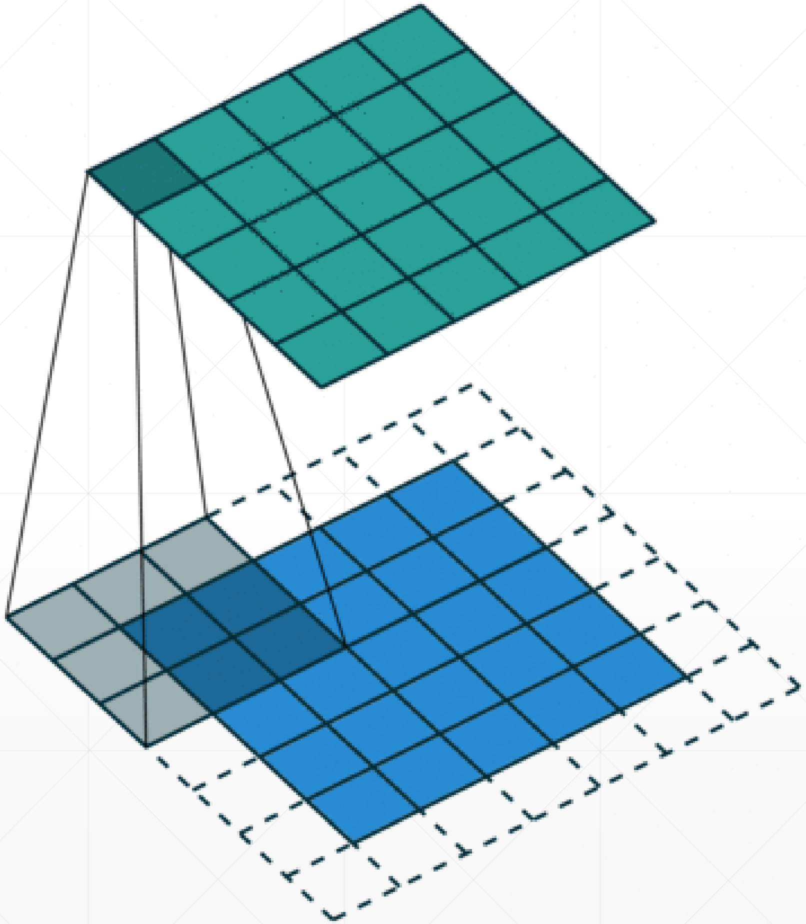
4+4+4	3+3+3	4+4+4
6	12	9
6	9	12

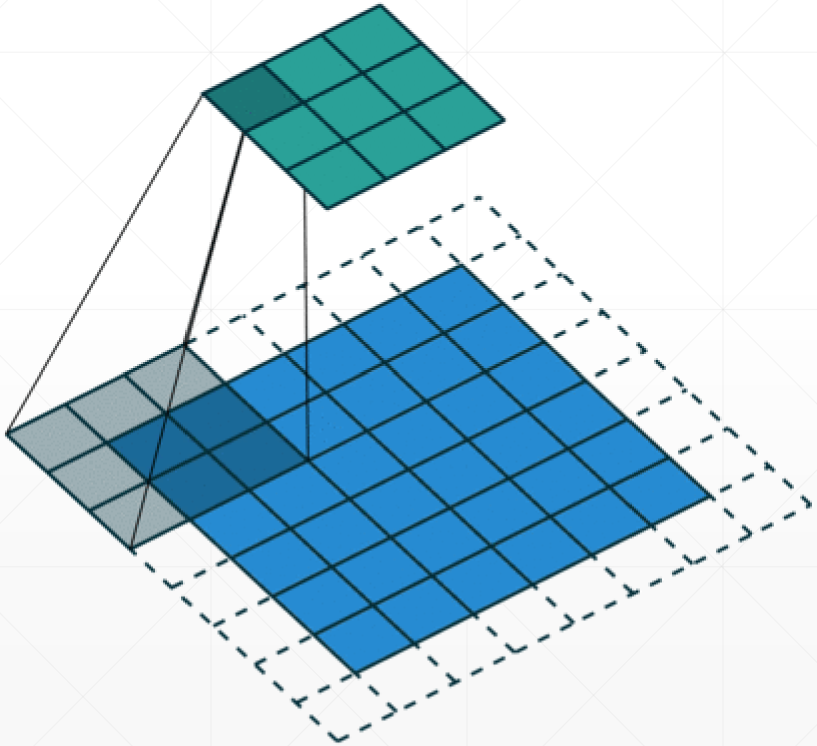
2D Convolution



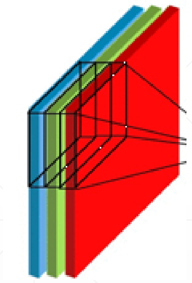
4+4+4	3+3+3	4+4+4
6	12	9
6	9	12

Padding & Stride

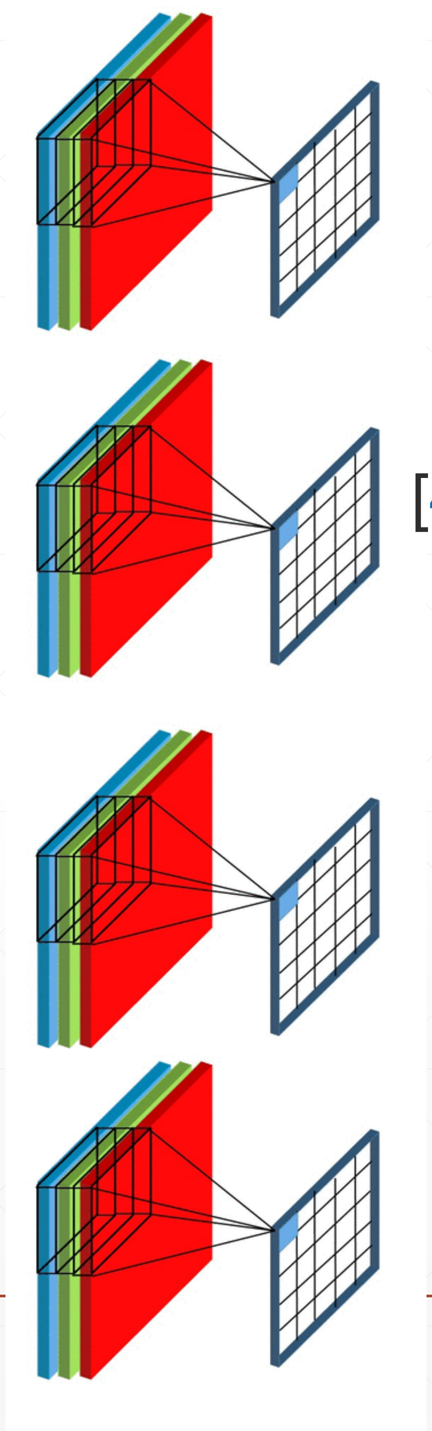




Channels



[1,32,32,3]



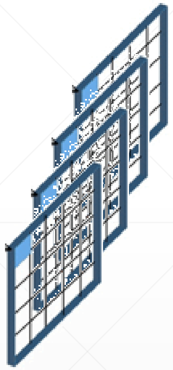
b_0 [4]

[4, 3, 5, 5]

b_1

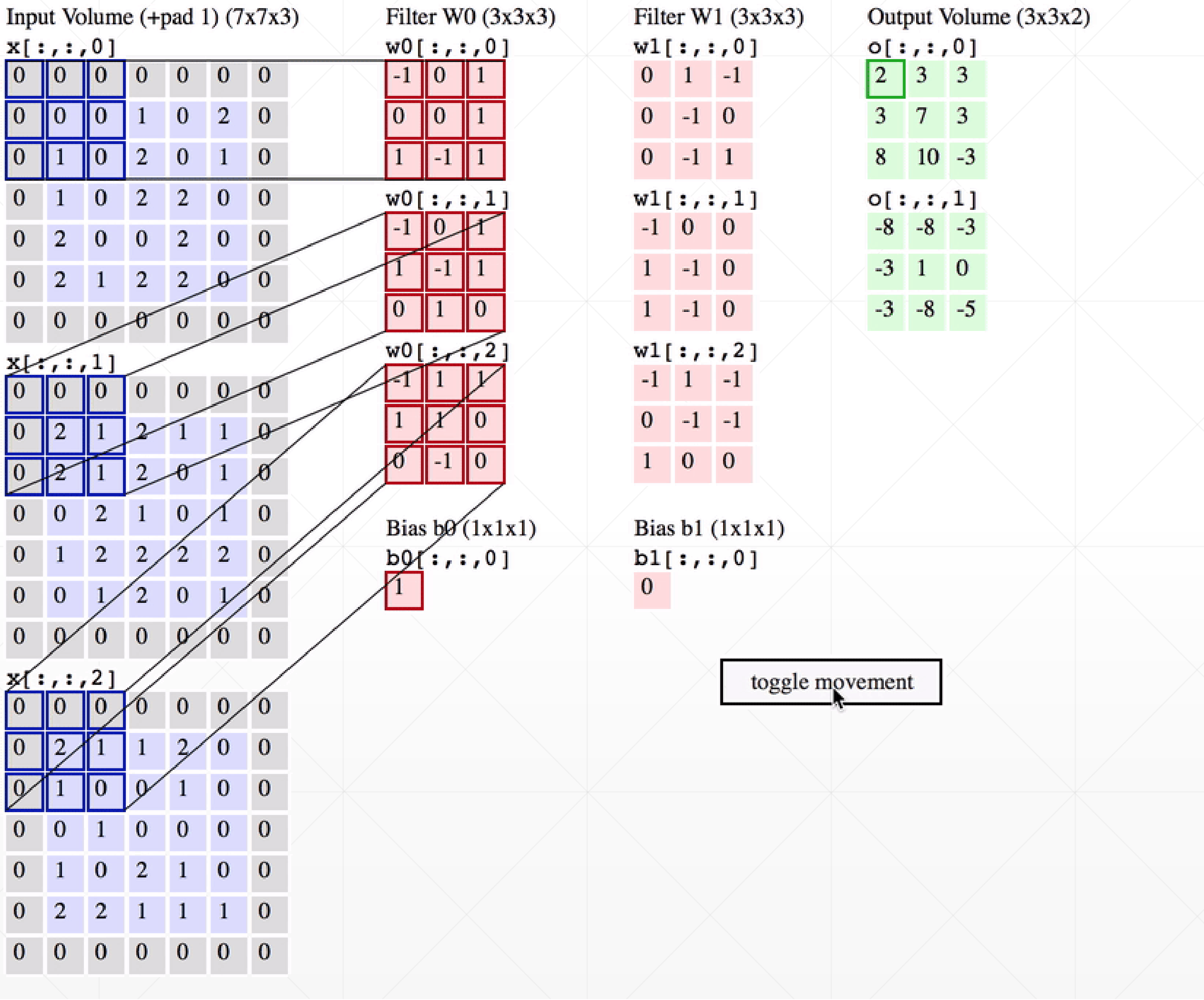
b_2

b_3



[1,30,30,4]

For instance



Input Volume (+pad 1) (7x7x3)

x[:,:,0]

0	0	0	0	0	0	0
0	0	0	1	0	2	0
0	1	0	2	0	1	0
0	1	0	2	2	0	0
0	2	0	0	2	0	0
0	2	1	2	2	0	0
0	0	0	0	0	0	0

x[:,:,1]

0	0	0	0	0	0	0
0	2	1	2	1	1	0
0	2	1	2	0	1	0
0	0	2	1	0	1	0
0	1	2	2	2	2	0
0	0	1	2	0	1	0
0	0	0	0	0	0	0

x[:,:,2]

0	0	0	0	0	0	0
0	2	1	1	2	0	0
0	1	0	0	1	0	0
0	0	1	0	0	0	0
0	1	0	2	1	0	0
0	2	2	1	1	1	0
0	0	0	0	0	0	0

Filter W0 (3x3x3)

w0[:,:,0]

-1	0	1
0	0	1
1	-1	1

w0[:,:,1]

-1	0	1
1	-1	1
0	1	0

w0[:,:,2]

-1	1	1
1	1	0
0	-1	0

Bias b0 (1x1x1)

b0[:,:,0]

1

Filter W1 (3x3x3)

w1[:,:,0]

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

w1[:,:,1]

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

w1[:,:,2]

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

Bias b1 (1x1x1)

b1[:,:,0]

0

Output Volume (3x3x2)

o[:,:,0]

2	3	3
3	7	3
8	10	-3

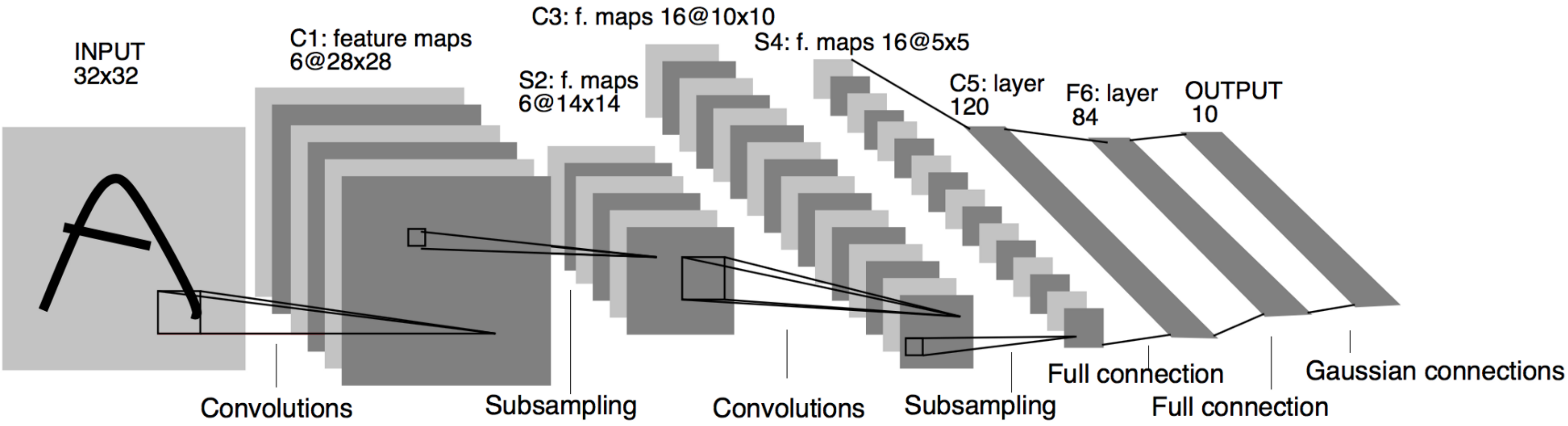
o[:,:,1]

-8	-8	-3
-3	1	0
-3	-8	-5

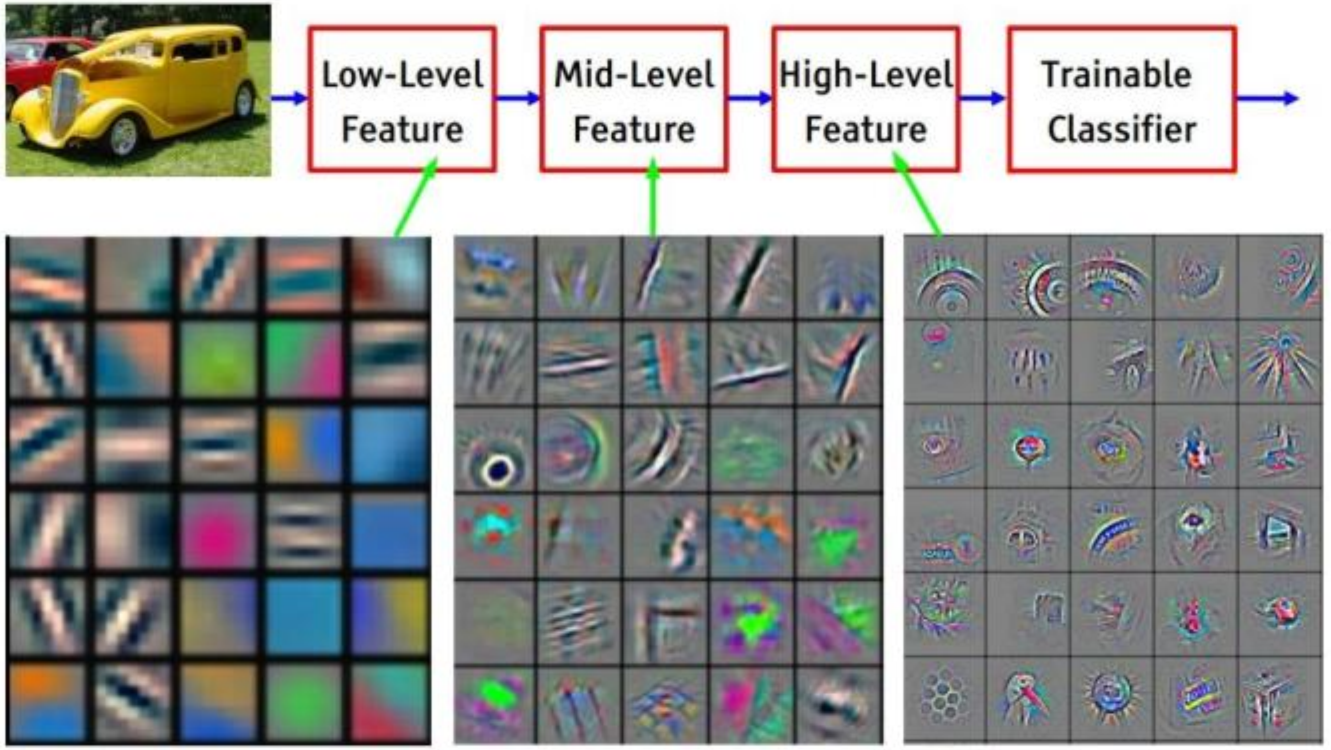
x:	[b, 28, 28, **3**]
one k:	[3, 3, 3]
multi-k:	[16, 3, 3, 3]
stride:	1
padding:	[1,1,1,1]
bias:	[16]
out:	[b, 28, 28, **16**]

https://skymind.ai/wiki/convolutional-network

LeNet-5


INPUT
32x32
C1: feature maps
6@28x28
S2: f. maps
6@14x14
C3: f. maps 16@10x10
S4: f. maps 16@5x5
C5: layer
120
F6: layer
84
OUTPUT
10
Convolutions
Subsampling
Convolutions
Subsampling
Full connection
Full connection
Gaussian connections

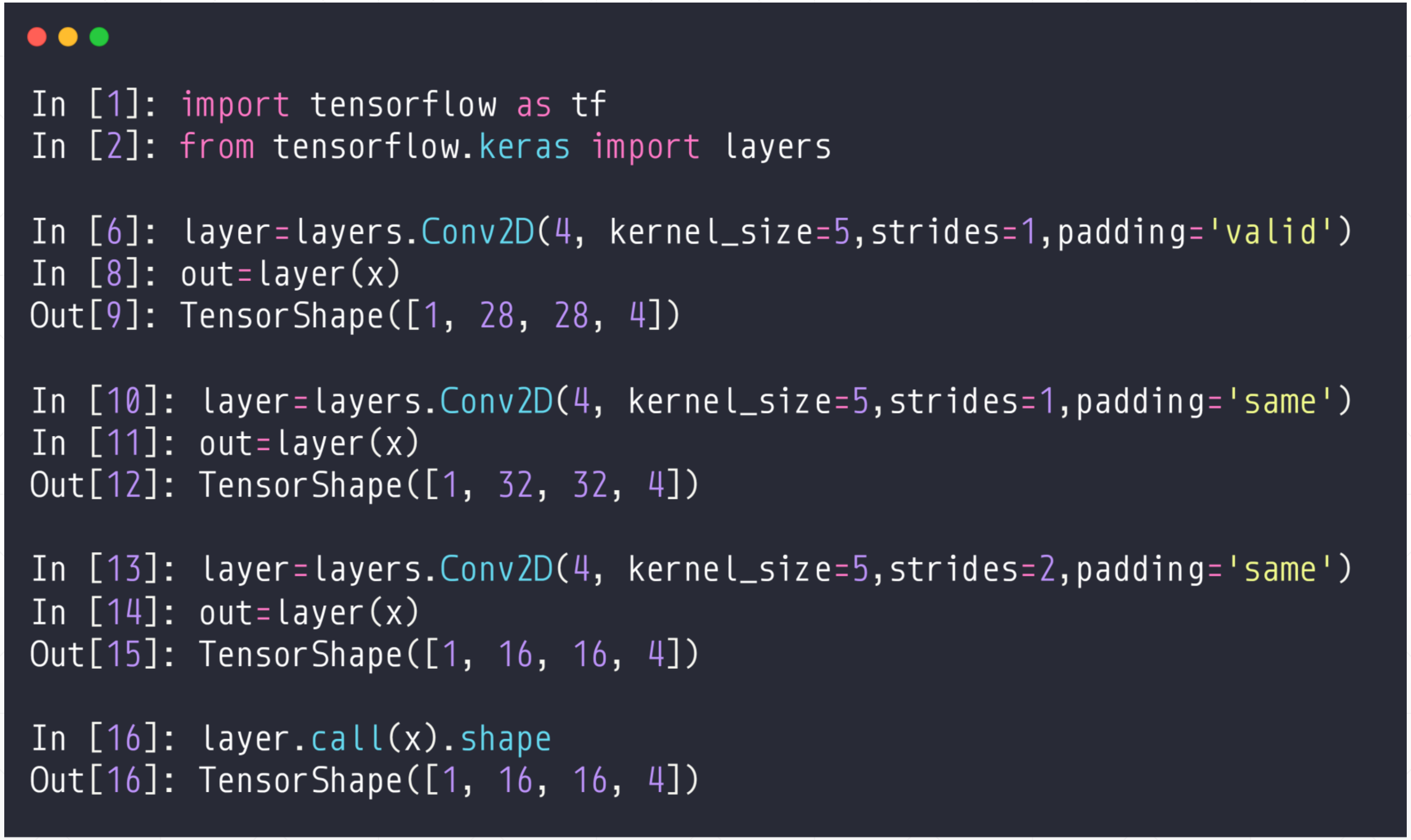
Pyramid Architecture



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

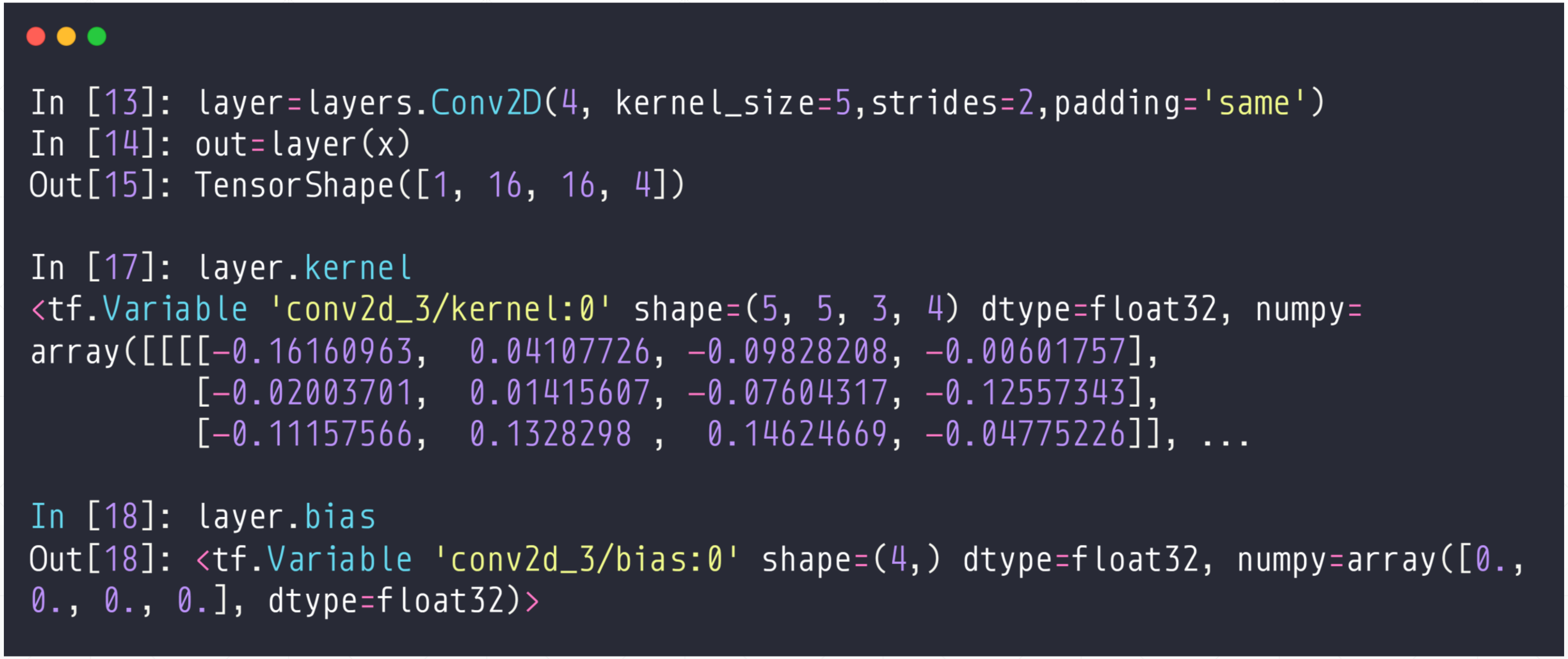
http://cs231n.stanford.edu/slides/2016/winter1516_lecture7.pdf

layers.Conv2D



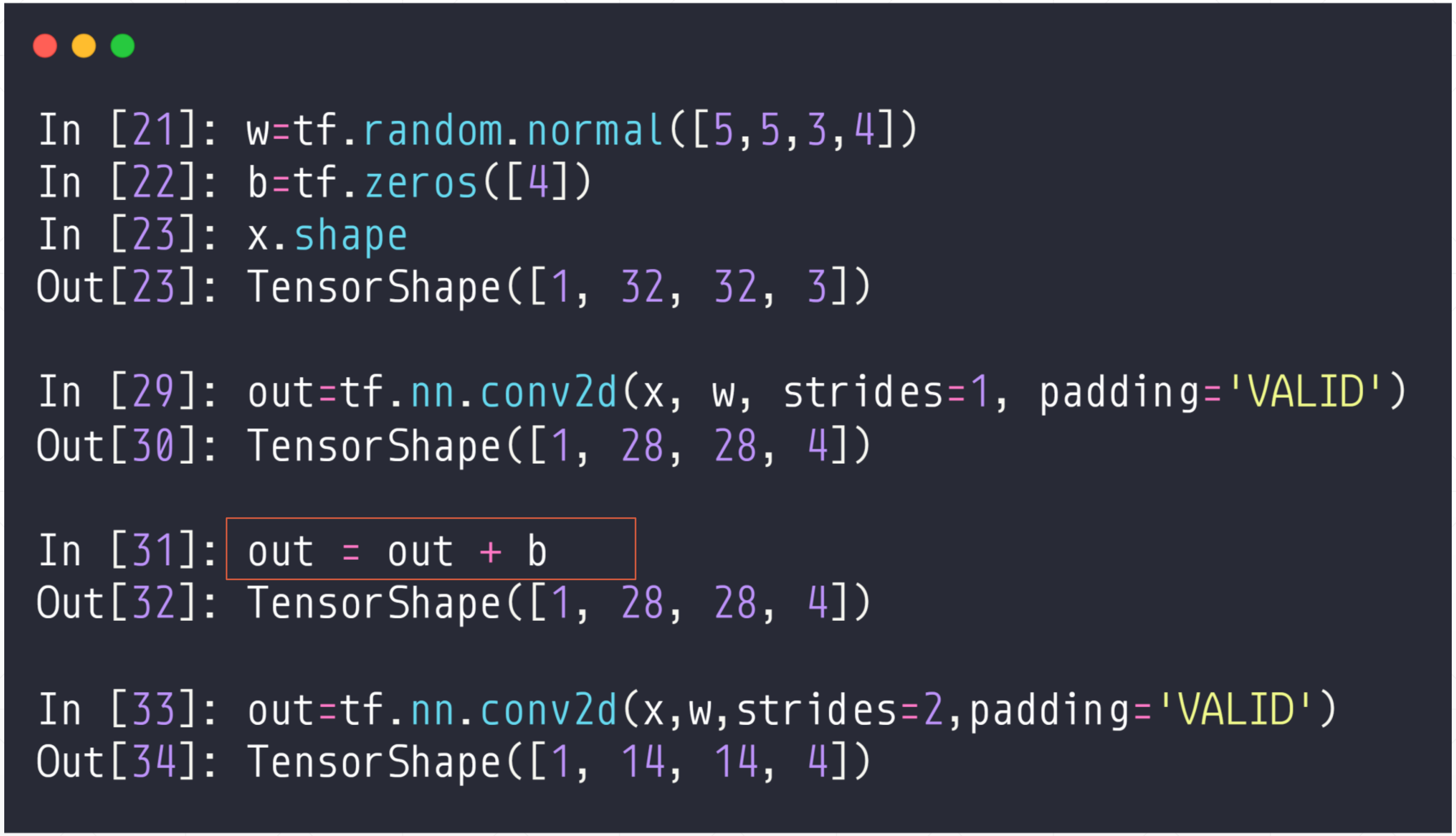
```
In [1]: import tensorflow as tf
In [2]: from tensorflow.keras import layers

In [6]: layer=layers.Conv2D(4, kernel_size=5,strides=1,padding='valid')
In [8]: out=layer(x)
Out[9]: TensorShape([1, 28, 28, 4])

In [10]: layer=layers.Conv2D(4, kernel_size=5,strides=1,padding='same')
In [11]: out=layer(x)
Out[12]: TensorShape([1, 32, 32, 4])

In [13]: layer=layers.Conv2D(4, kernel_size=5,strides=2,padding='same')
In [14]: out=layer(x)
Out[15]: TensorShape([1, 16, 16, 4])

In [16]: layer.call(x).shape
Out[16]: TensorShape([1, 16, 16, 4])
```

weight & bias



```
In [13]: layer=layers.Conv2D(4, kernel_size=5,strides=2,padding='same')
In [14]: out=layer(x)
Out[15]: TensorShape([1, 16, 16, 4])

In [17]: layer.kernel
<tf.Variable 'conv2d_3/kernel:0' shape=(5, 5, 3, 4) dtype=float32, numpy=
array([[[[-0.16160963,  0.04107726, -0.09828208, -0.00601757],
         [-0.02003701,  0.01415607, -0.07604317, -0.12557343],
         [-0.11157566,  0.1328298 ,  0.14624669, -0.04775226]], ...

In [18]: layer.bias
Out[18]: <tf.Variable 'conv2d_3/bias:0' shape=(4,) dtype=float32, numpy=array([0.,
0., 0., 0.], dtype=float32)>
```

nn.conv2d



```
In [21]: w=tf.random.normal([5,5,3,4])
In [22]: b=tf.zeros([4])
In [23]: x.shape
Out[23]: TensorShape([1, 32, 32, 3])

In [29]: out=tf.nn.conv2d(x, w, strides=1, padding='VALID')
Out[30]: TensorShape([1, 28, 28, 4])

In [31]: out = out + b
Out[32]: TensorShape([1, 28, 28, 4])

In [33]: out=tf.nn.conv2d(x,w,strides=2,padding='VALID')
Out[34]: TensorShape([1, 14, 14, 4])
```

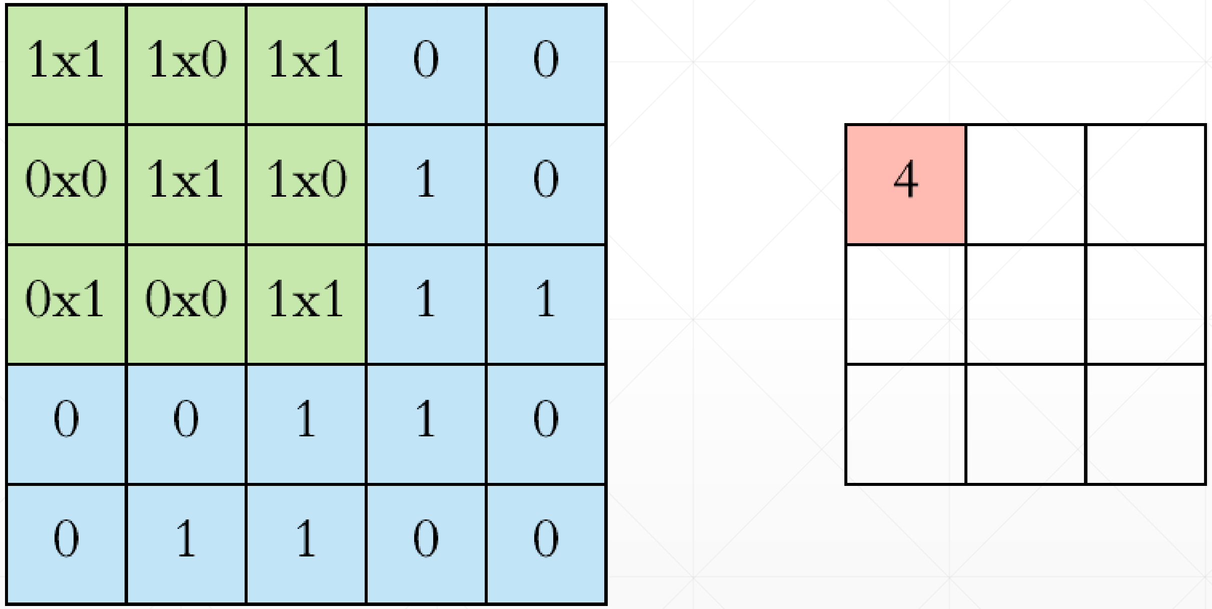
One more thing

∇ The Gradient

https://thegradient.pub/

Gradient?

- $\frac{\partial Loss}{\partial w}$



1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

For instance

x_{00}	x_{01}	x_{02}
x_{10}	x_{11}	x_{12}
x_{20}	x_{21}	x_{22}

w_{00}	w_{01}
w_{10}	w_{11}

$$O_{00} = x_{00}{}^{*}w_{00} + x_{01}{}^{*}w_{01} + x_{10}{}^{*}w_{10} + x_{11}{}^{*}w_{11} + b$$

$$O_{01} = x_{01}{}^{*}w_{00} + x_{02}{}^{*}w_{01} + x_{11}{}^{*}w_{10} + x_{12}{}^{*}w_{11} + b$$

$$O_{10} = x_{10}{}^{*}w_{00} + x_{11}{}^{*}w_{01} + x_{20}{}^{*}w_{10} + x_{21}{}^{*}w_{11} + b$$

$$O_{11} = x_{11}{}^{*}w_{00} + x_{12}{}^{*}w_{01} + x_{21}{}^{*}w_{10} + x_{22}{}^{*}w_{11} + b$$

O_{00}	O_{01}
O_{10}	O_{11}

↔

0
1
0
0

$$\frac{\partial Loss}{\partial w_{00}} = \sum_{i\in\{00,01,10,11\}} \frac{\partial Loss}{\partial O_i}\frac{\partial O_i}{\partial w_i}$$

$$\frac{\partial O_{00}}{\partial w_{00}} = \frac{\partial(x_{00}*w_{00} + x_{01}*w_{01} + x_{10}*w_{10} + x_{11}*w_{11} + b)}{w_{00}}$$

$$= \boldsymbol{x_{00}}$$

下一课时

池化与采样

Thank You.