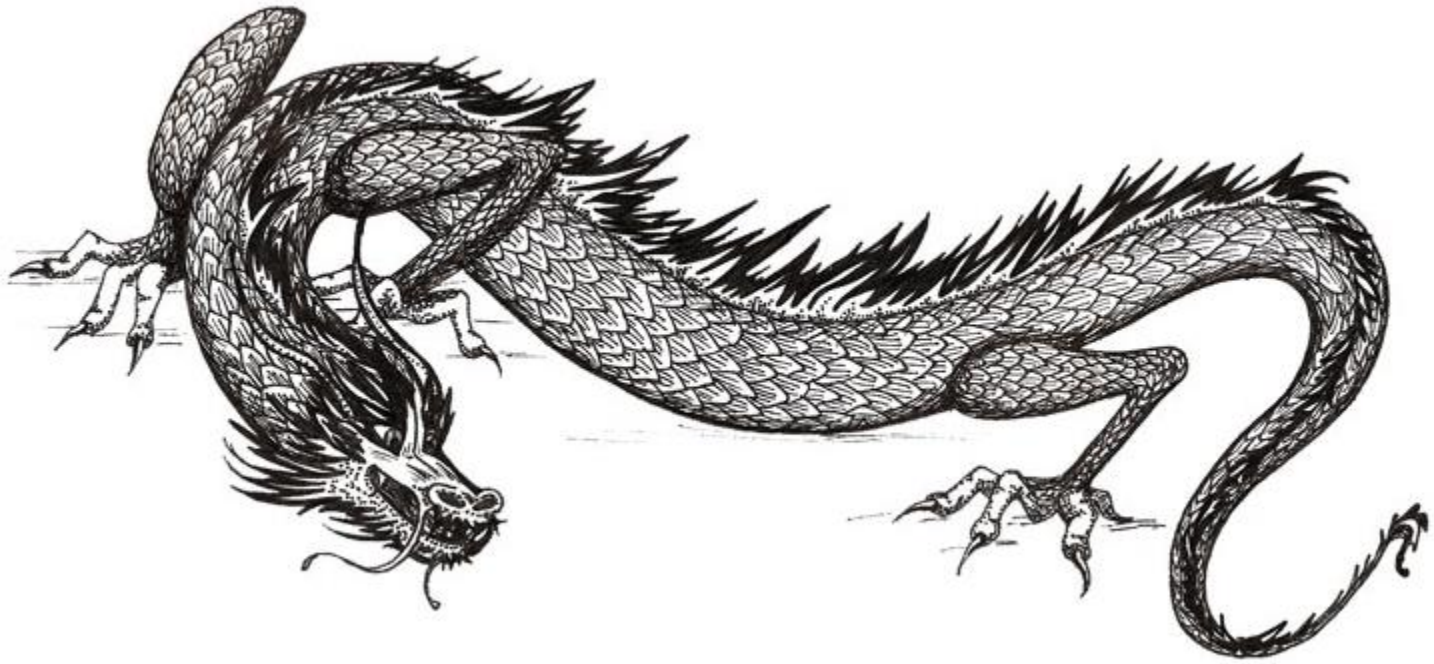


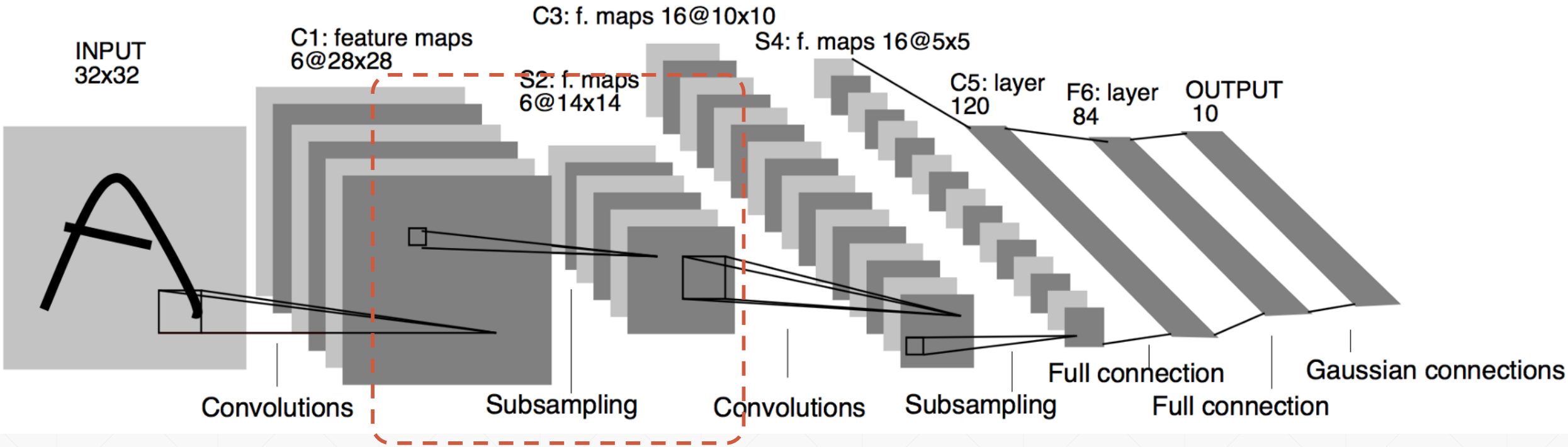
池化与采样

主讲：龙良曲

Outline

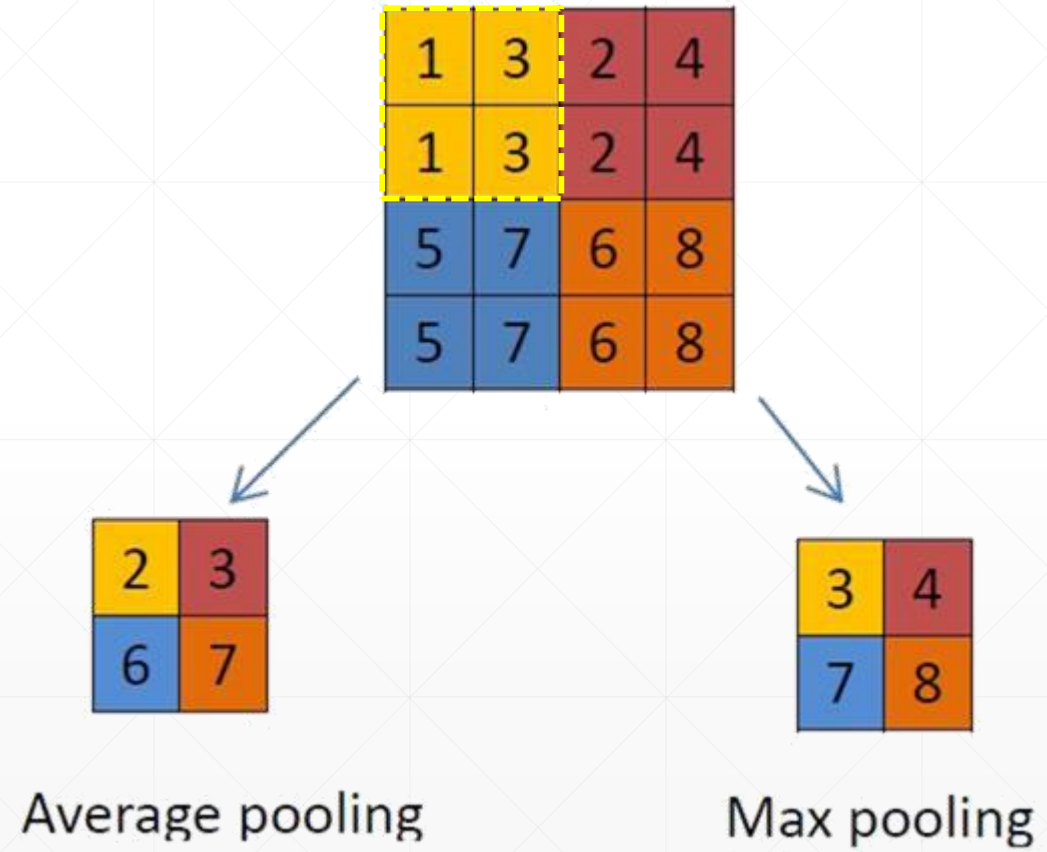
- Pooling
- upsample
- ReLU

Reduce Dim


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

Max/Avg pooling

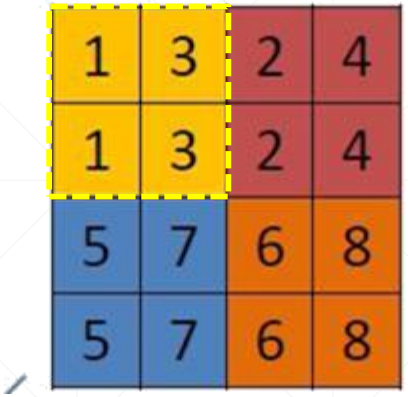
- stride=2



https://towardsdatascience.com/a-deeper-understanding-of-nnets-part-1-cnns-263a6e3ac61

Strides

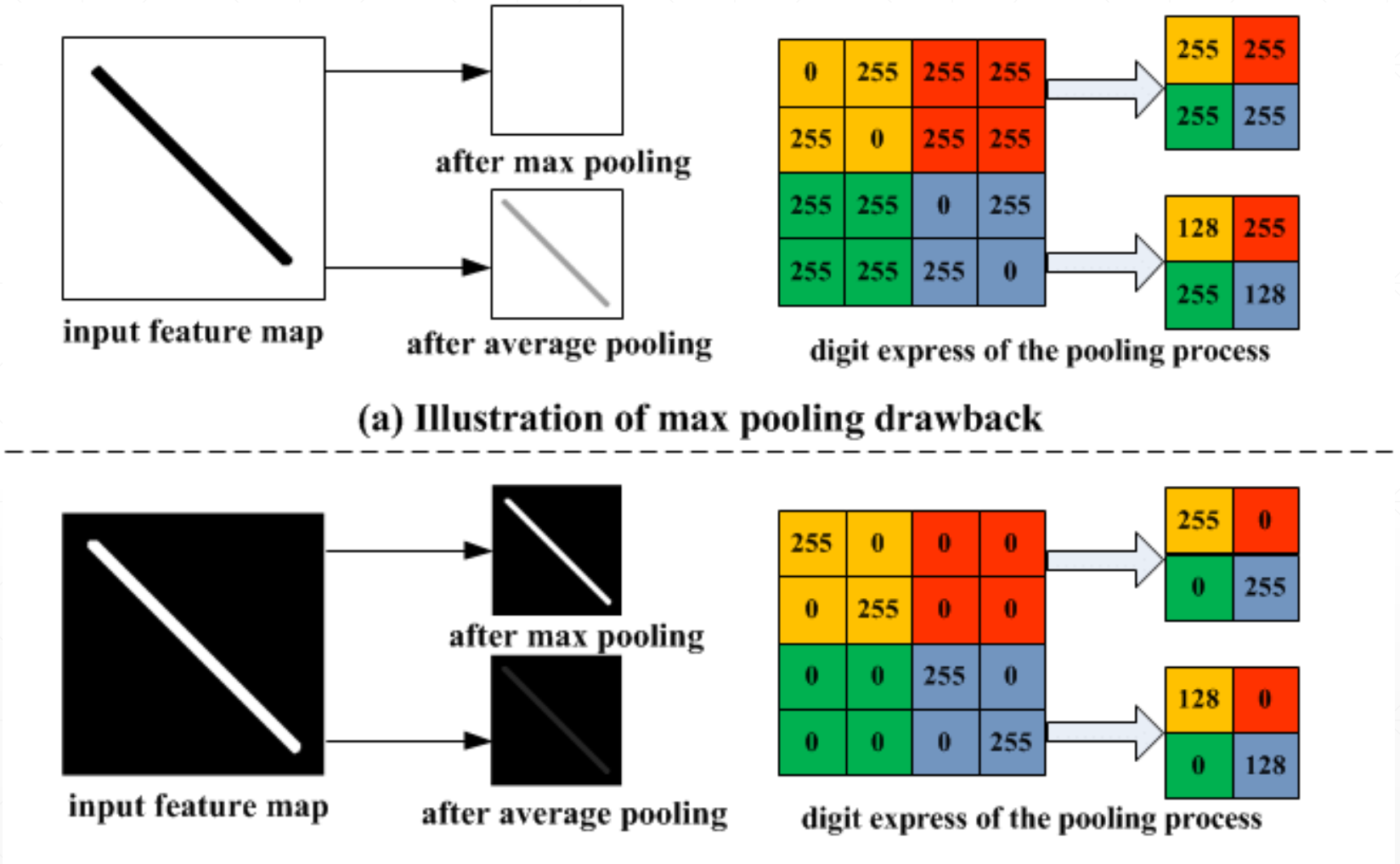
- stride=1



3	**3**	**4**
7	7	8
7	7	8

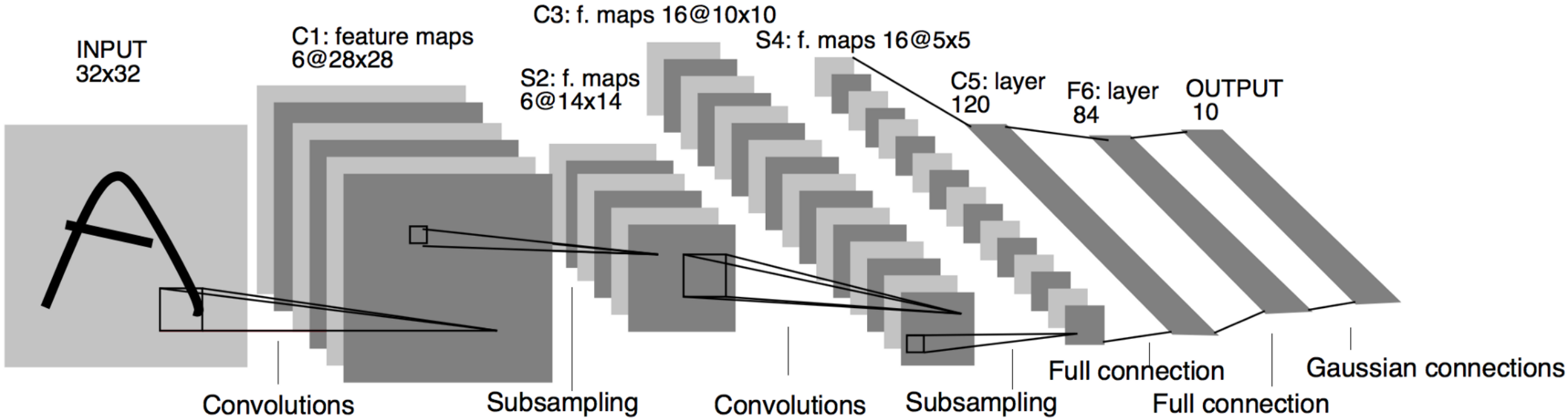
https://towardsdatascience.com/a-deeper-understanding-of-nnets-part-1-cnns-263a6e3ac61

For instance



https://www.researchgate.net/figure/Toy-example-illustrating-the-drawbacks-of-max-pooling-and-average-pooling_fig2_300020038

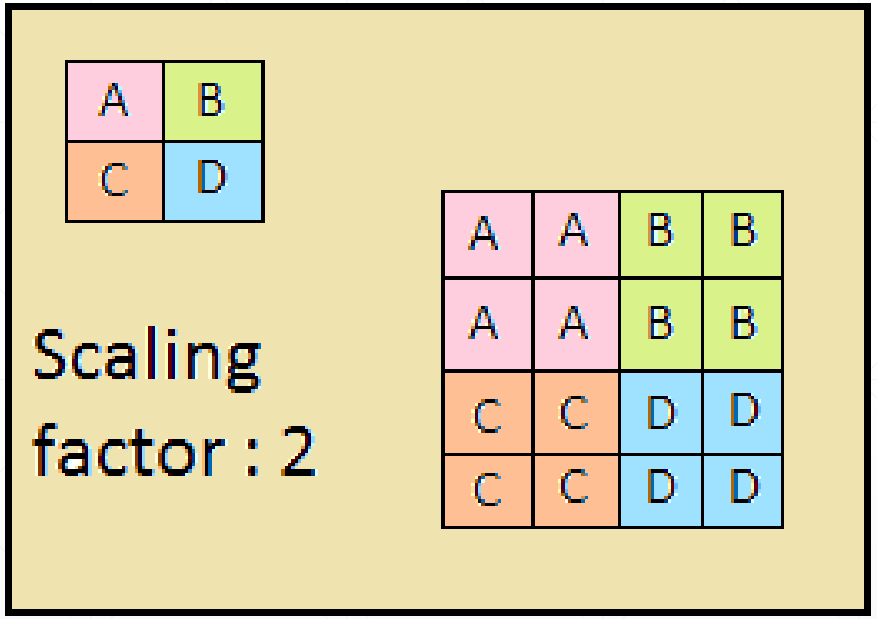
LeNet-5


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

```
In [36]: x # TensorShape([1, 14, 14, 4])

In [37]: pool=layers.MaxPool2D(2,strides=2)
In [38]: out=pool(x)
Out[39]: TensorShape([1, 7, 7, 4])

In [40]: pool=layers.MaxPool2D(3,strides=2)
In [41]: out=pool(x)
Out[42]: TensorShape([1, 6, 6, 4])

In [44]: out=tf.nn.max_pool2d(x, 2,strides=2,padding='VALID')
Out[45]: TensorShape([1, 7, 7, 4])
```

upsample

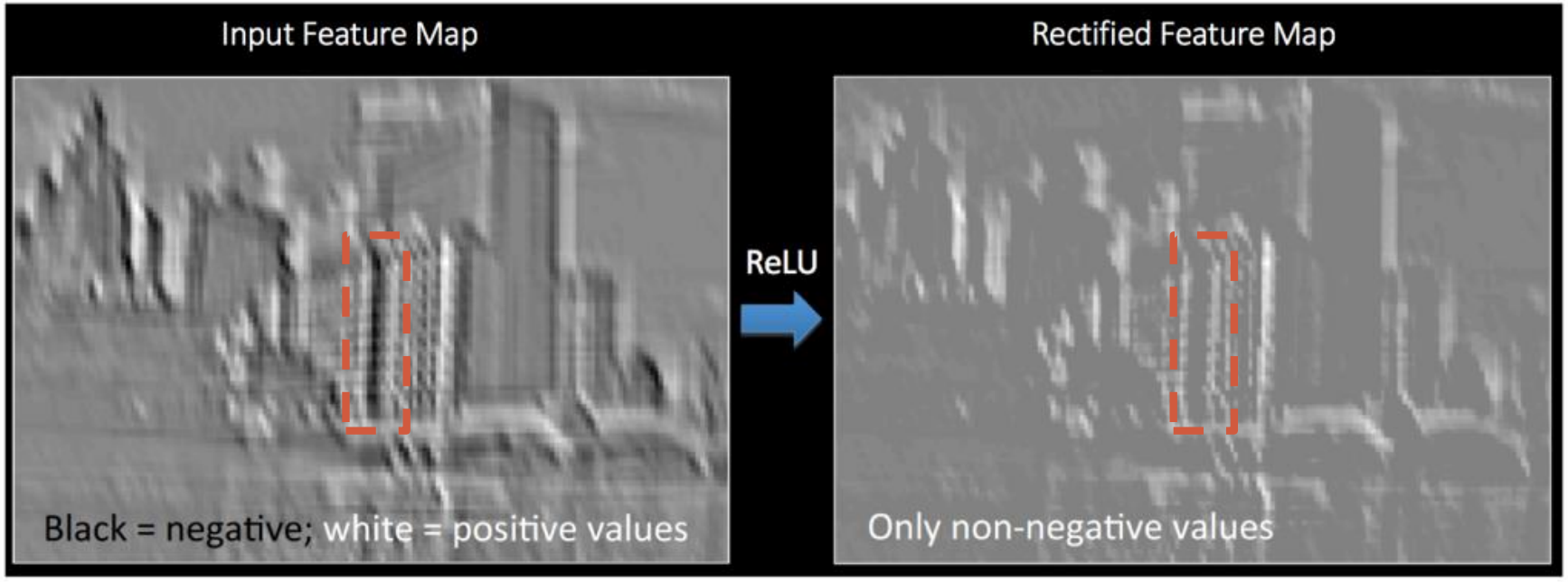
- nearest
- bilinear



UpSampling2D

```
In [47]: x=tf.random.normal([1,7,7,4])

In [48]: layer=layers.UpSampling2D(size=3)
In [49]: out=layer(x)
Out[50]: TensorShape([1, 21, 21, 4])

In [51]: layer=layers.UpSampling2D(size=2)
In [52]: out=layer(x)
Out[53]: TensorShape([1, 14, 14, 4])
```

ReLU


Input Feature Map
Rectified Feature Map
ReLU
Black = negative; white = positive values
Only non-negative values

```
In [55]: x=tf.random.normal([2,3])
<tf.Tensor: id=154, shape=(2, 3), dtype=float32, numpy=
array([[-1.533682  , -2.7053335 ,  0.36354962],
       [ 0.00713745,  0.69756126,  0.8053344 ]], dtype=float32)>

In [57]: tf.nn.relu(x)
<tf.Tensor: id=156, shape=(2, 3), dtype=float32, numpy=
array([[0.        , 0.        , 0.36354962],
       [0.00713745, 0.69756126, 0.8053344 ]], dtype=float32)>

In [59]: layers.ReLU()(x)
<tf.Tensor: id=158, shape=(2, 3), dtype=float32, numpy=
array([[0.        , 0.        , 0.36354962],
       [0.00713745, 0.69756126, 0.8053344 ]], dtype=float32)>
```

下一课时

CIFAR100
实战

Thank You.