Convolutional Neural Networks

Jiahui Chen
Department of Mathematical Sciences
University of Arkansas
Reference: Ming Li's notes

Introduction

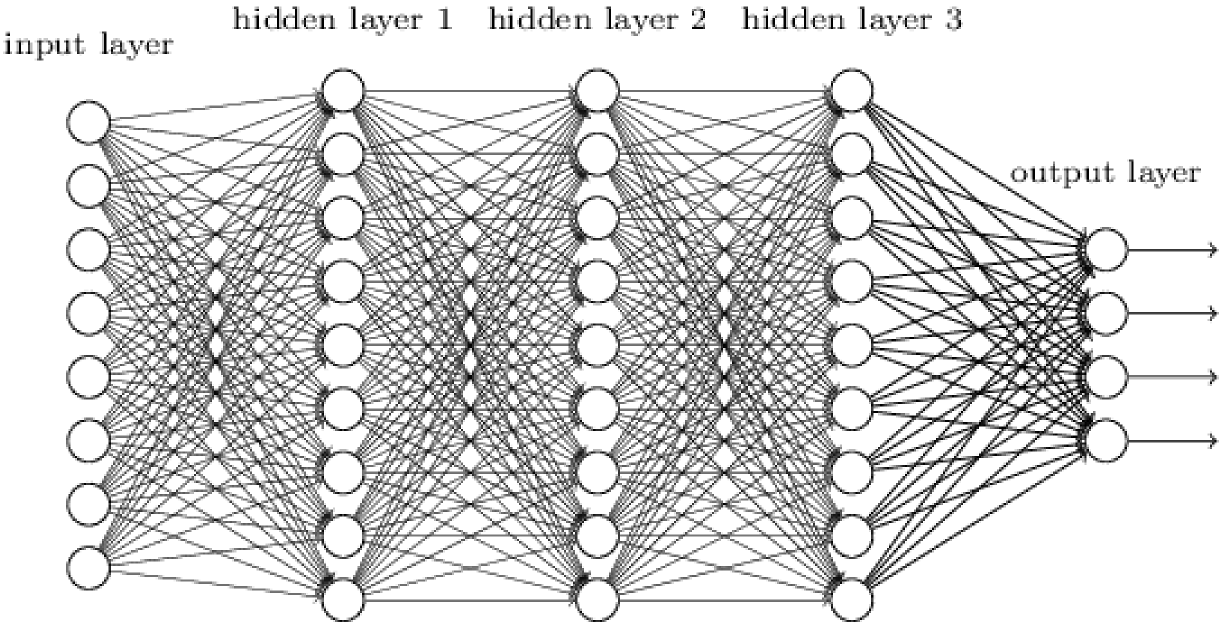
- Fukushima (1980) – Neo-Cognitron; LeCun (1998) – Convolutional Neural Networks (CNN, or ConvNet);...

Motivation:

Images typically have 1000^2 pixels, which give rise to 1000^2 data points or features, leading to an intractably high dimension of the weight space. However, not all of them are essential due to the spatial patterns. Many of them have shared properties. Therefore, the weight space dimension can be dramatically reduced if an appropriate pre-processing of the image data is carried out to analyze or extract spatial correlations or patterns in images.

Motivation

- We know it is good to learn a small model.
- From this fully connected model, do we really need all the edges?
- Can some of these be shared?



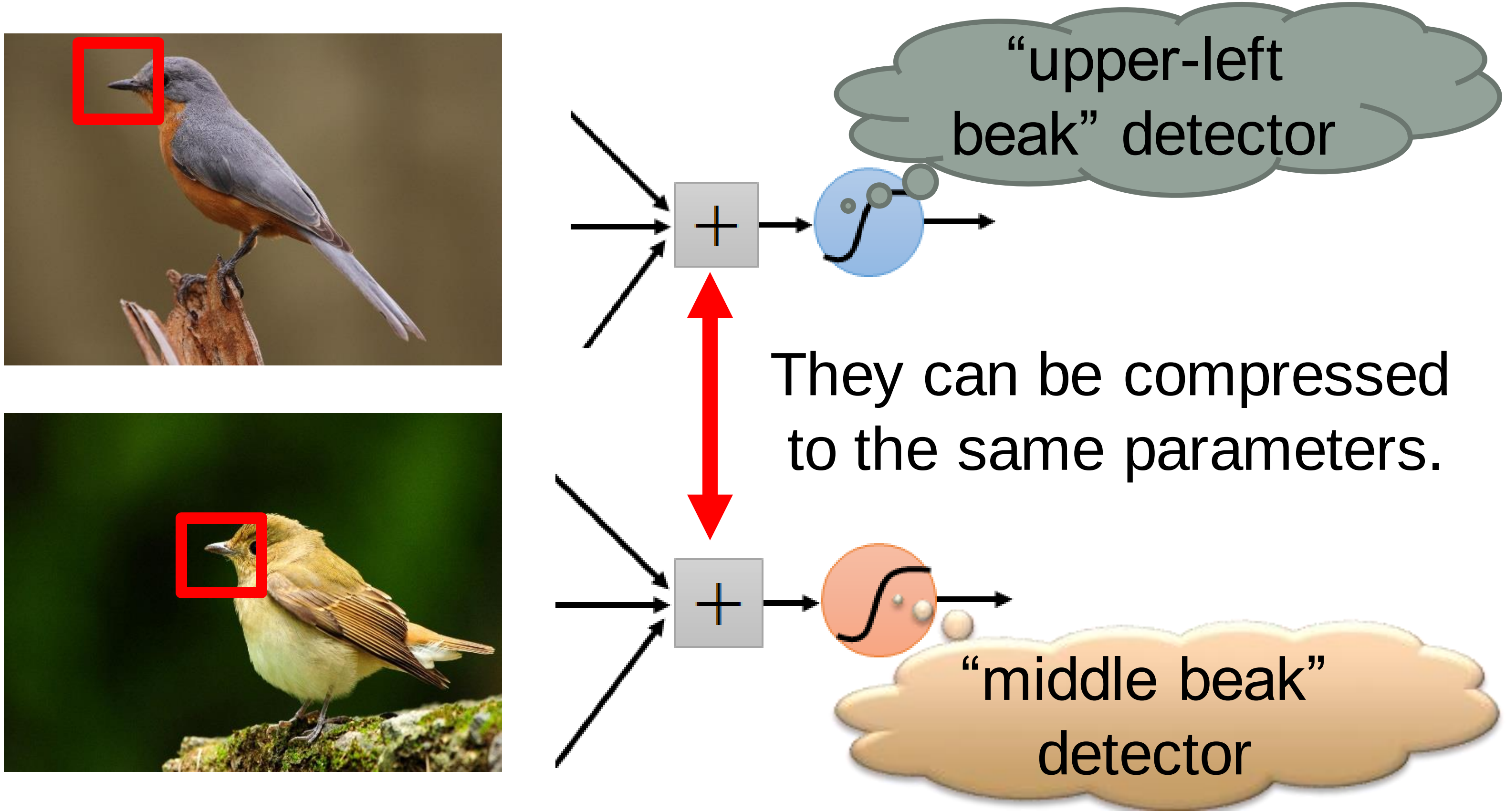
Motivation

Consider learning an image:

Some patterns are much smaller than the whole image

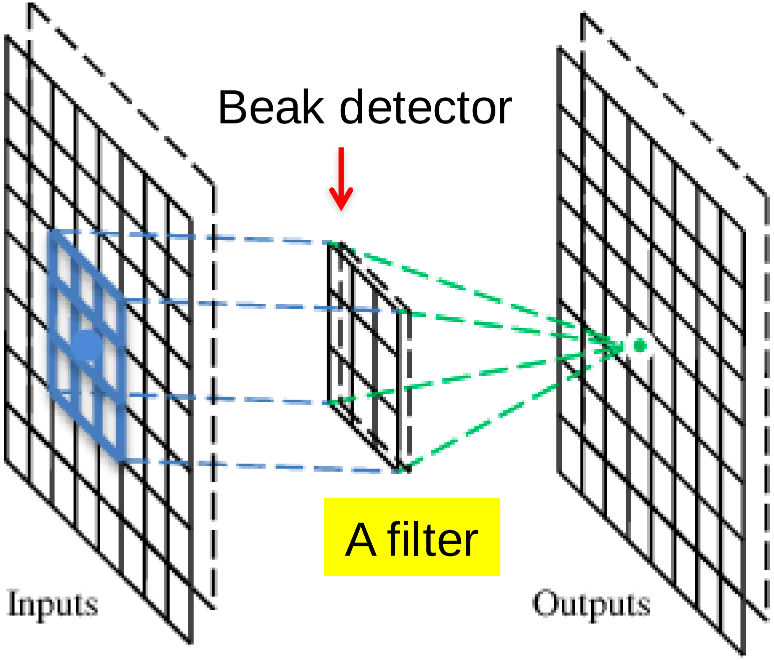
Can represent a small region with fewer parameters

Same pattern appears in different places:
They can be compressed!
What about training a lot of such "small" detectors and each detector must "move around".



A Convolutional Layer

A CNN is a neural network with some convolutional layers (and some other layers). A convolutional layer has a number of filters that does convolutional operation.



UNIVERSITY OF ARKANSAS

Convolution

These are the network parameters to be learned.

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

-1	1	-1
-1	1	-1
-1	1	-1

Filter 2

⋮ ⋮

Each filter detects a small pattern (3 x 3).

Convolution

stride=1

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

Dot product →

3 -1

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

Convolution

If stride=2

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

3 -3

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

stride=1

Filter 1

1	-1	-1
-1	1	-1
-1	-1	1

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

3	-1	-3	-1
-3	1	0	-3
-3	-3	0	1
3	-2	-2	-1

stride=1

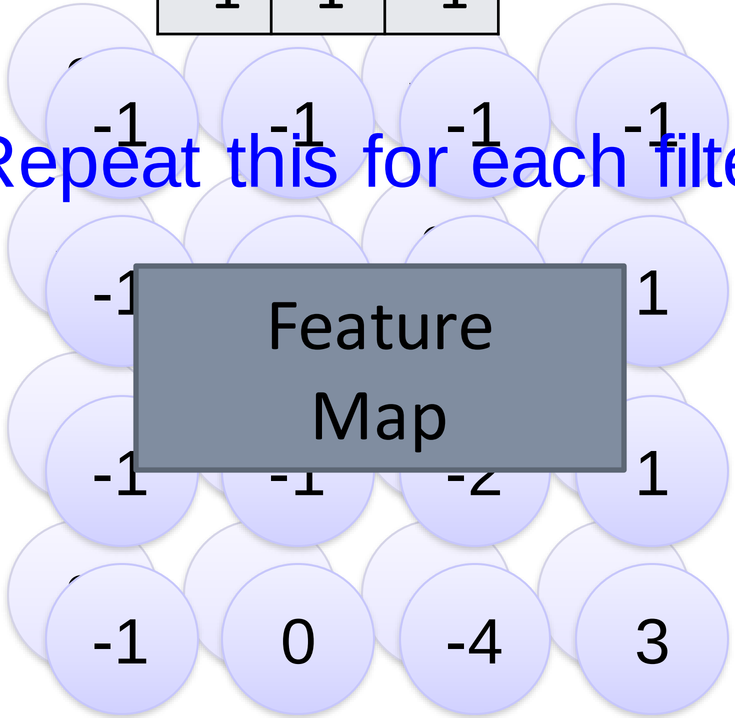
1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

-1	1	-1
-1	1	-1
-1	1	-1

Filter 2

Repeat this for each filter



Two 4 x 4 images Forming 2 x 4 x 4 matrix

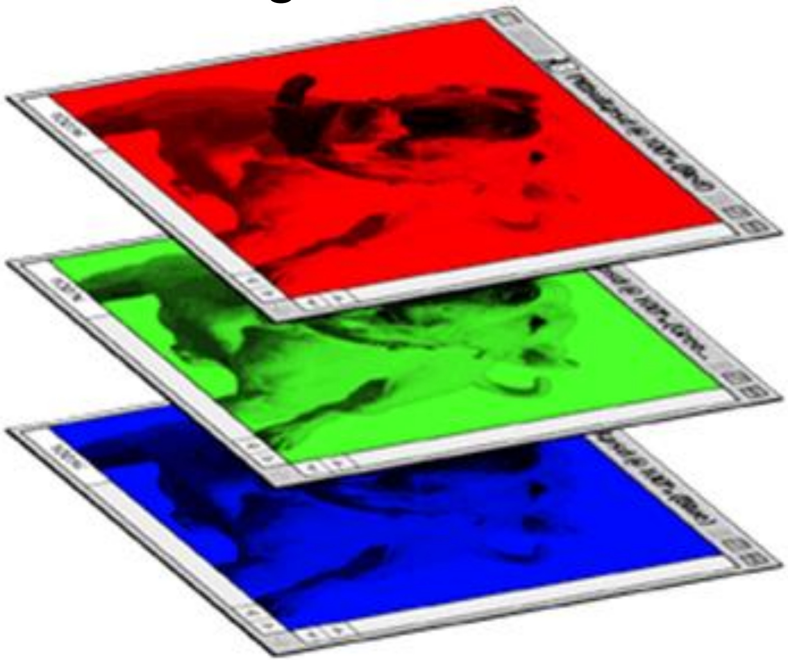
1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

-1	1	-1
-1	1	-1
-1	1	-1

Filter 2

Color image



1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

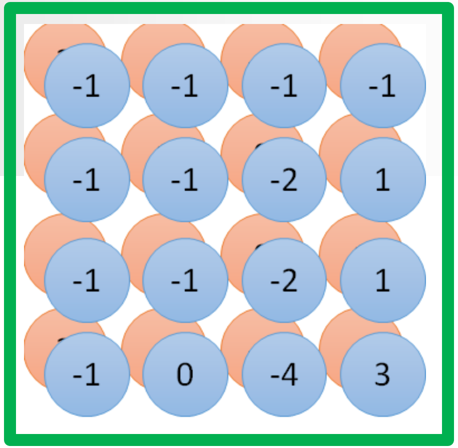
1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

image

1	-1	-1
-1	1	-1
-1	-1	1

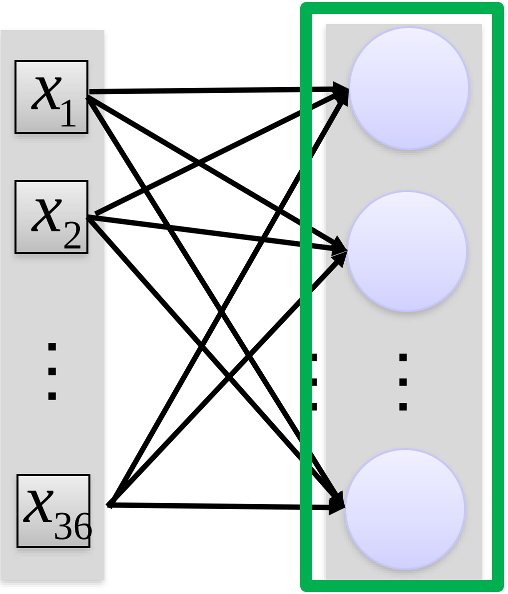
-1	1	-1
-1	1	-1
-1	1	-1

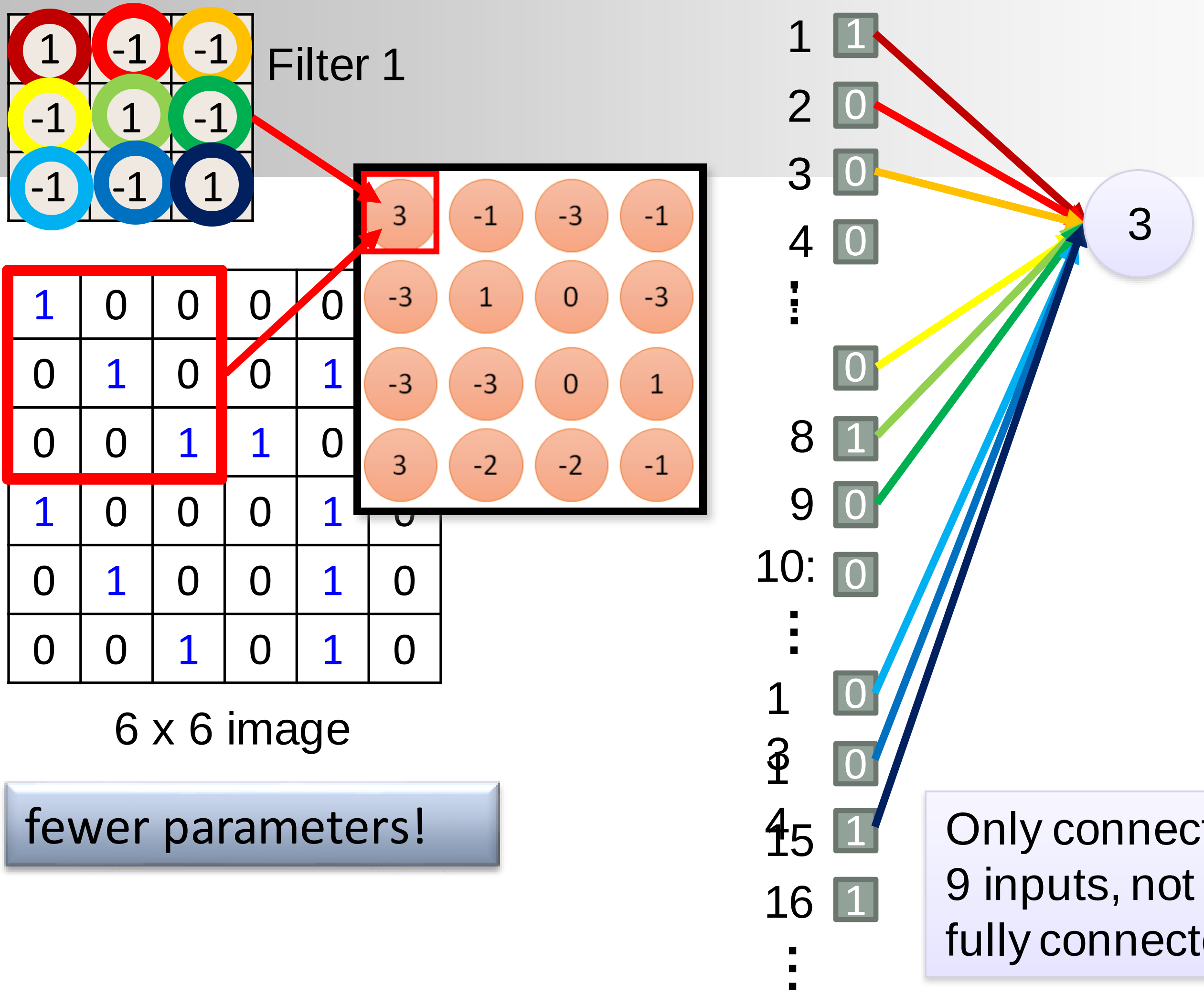
convolution



Fully-connected

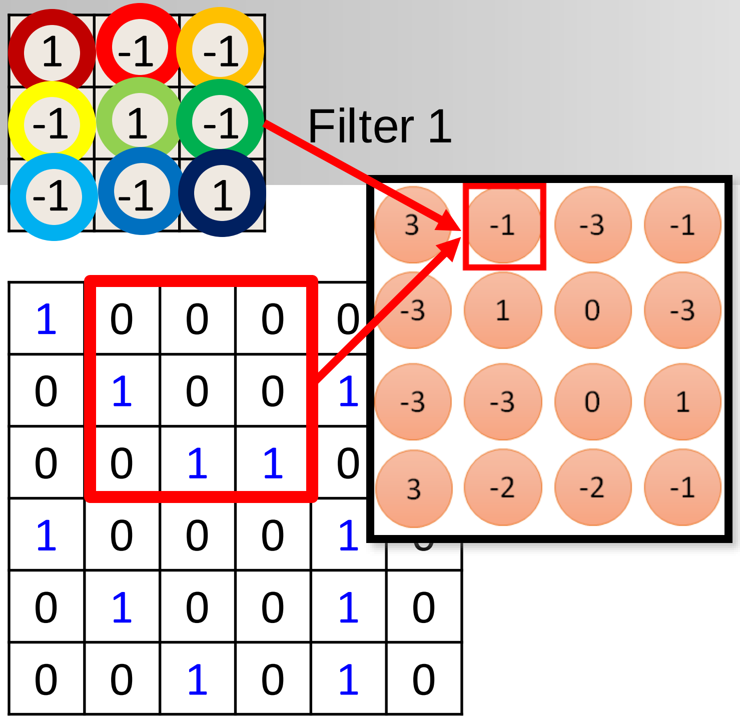
1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0





fewer parameters!

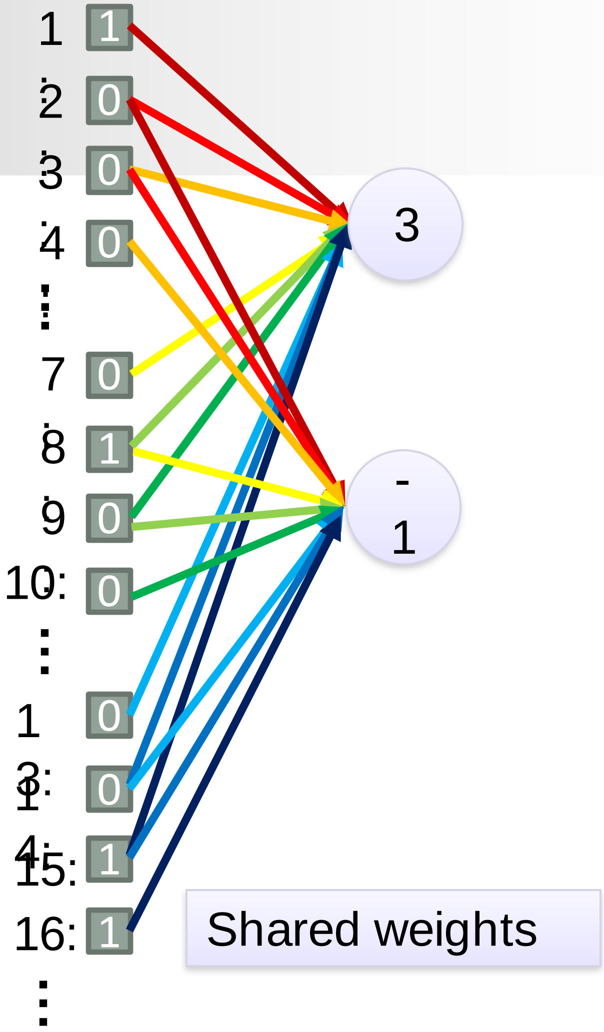
Only connect to 9 inputs, not fully connected



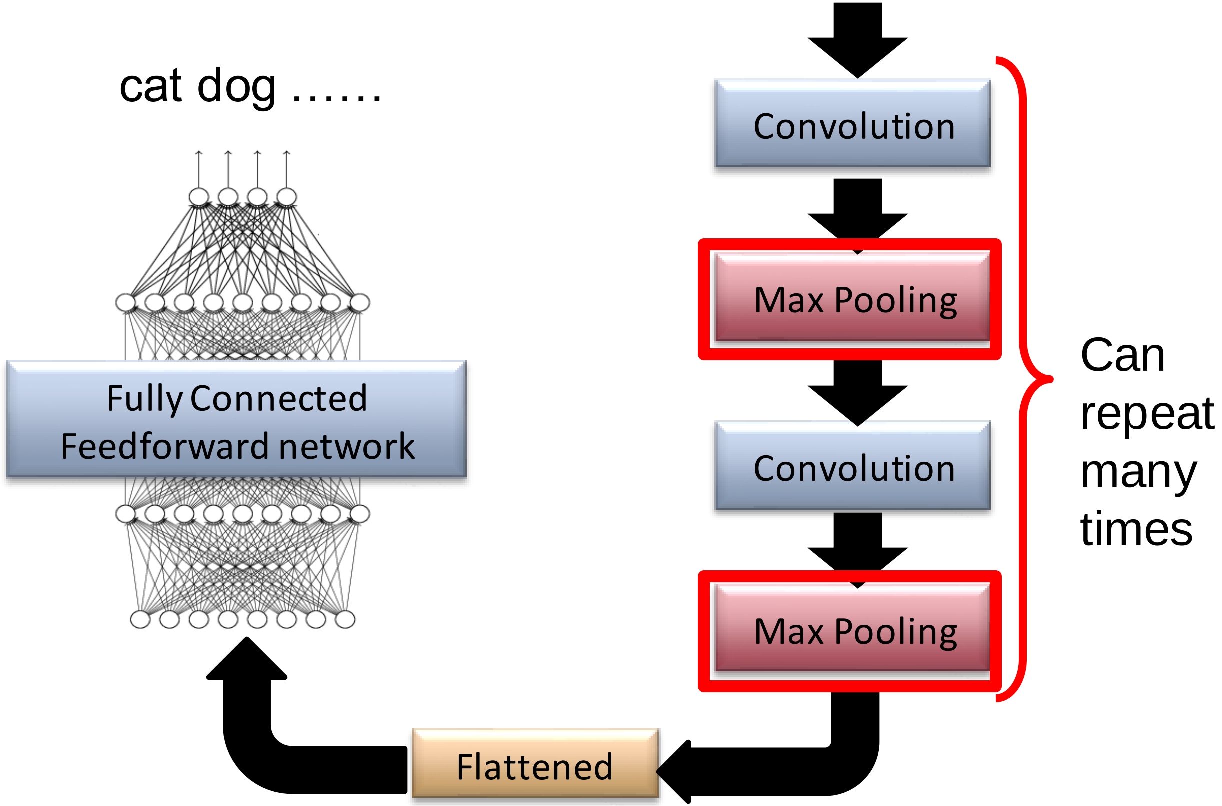
6 x 6 image

Fewer parameters

Even fewer parameters



The whole CNN



Max Pooling

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

-1	1	-1
-1	1	-1
-1	1	-1

Filter 2

3	-1	-3	-1
-3	1	0	-3
-3	-3	0	1
3	-2	-2	-1

-1	-1	-1	-1
-1	-1	-2	1
-1	-1	-2	1
-1	0	-4	3

Why Pooling

- Subsampling pixels will not change the object

bird

Subsampling

bird

We can subsample the pixels to make image smaller

fewer parameters to characterize the image

A CNN compresses a fully connected network in two ways:

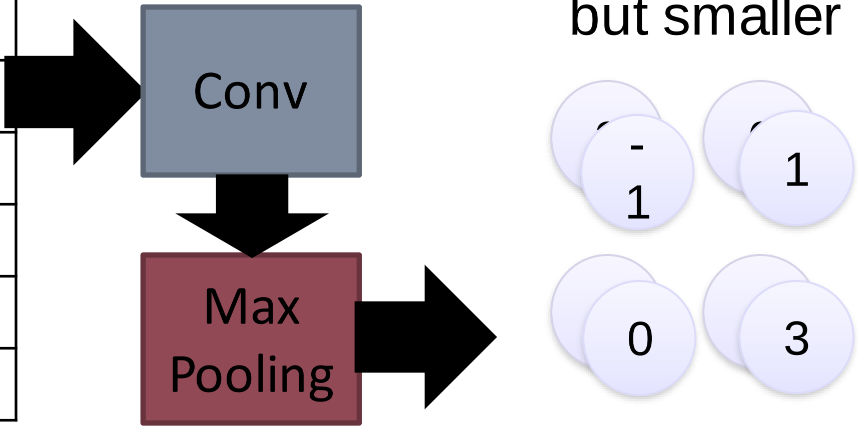
- Reducing number of connections
- Shared weights on the edges
- Max pooling further reduces the complexity

Max Pooling

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image

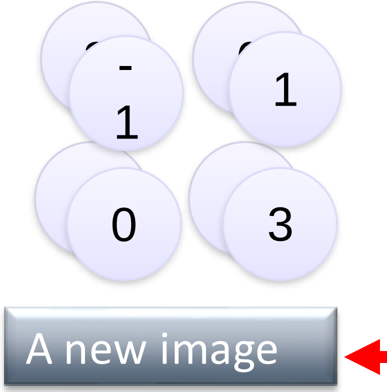
New image but smaller

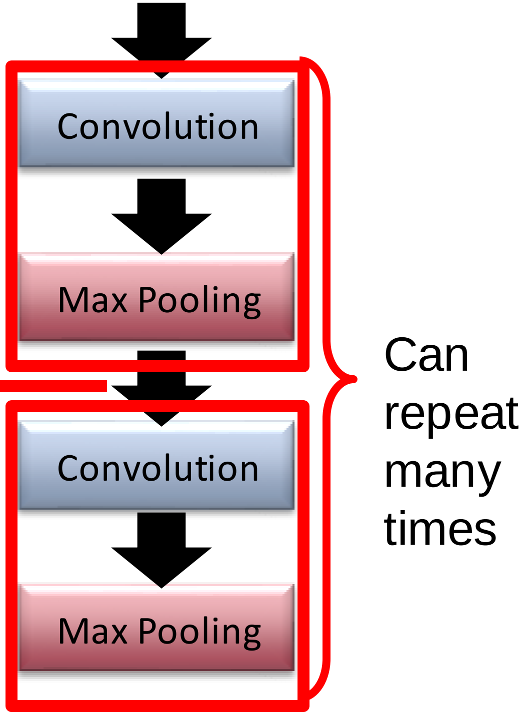


2 x 2 image

Each filter is a channel

The whole CNN

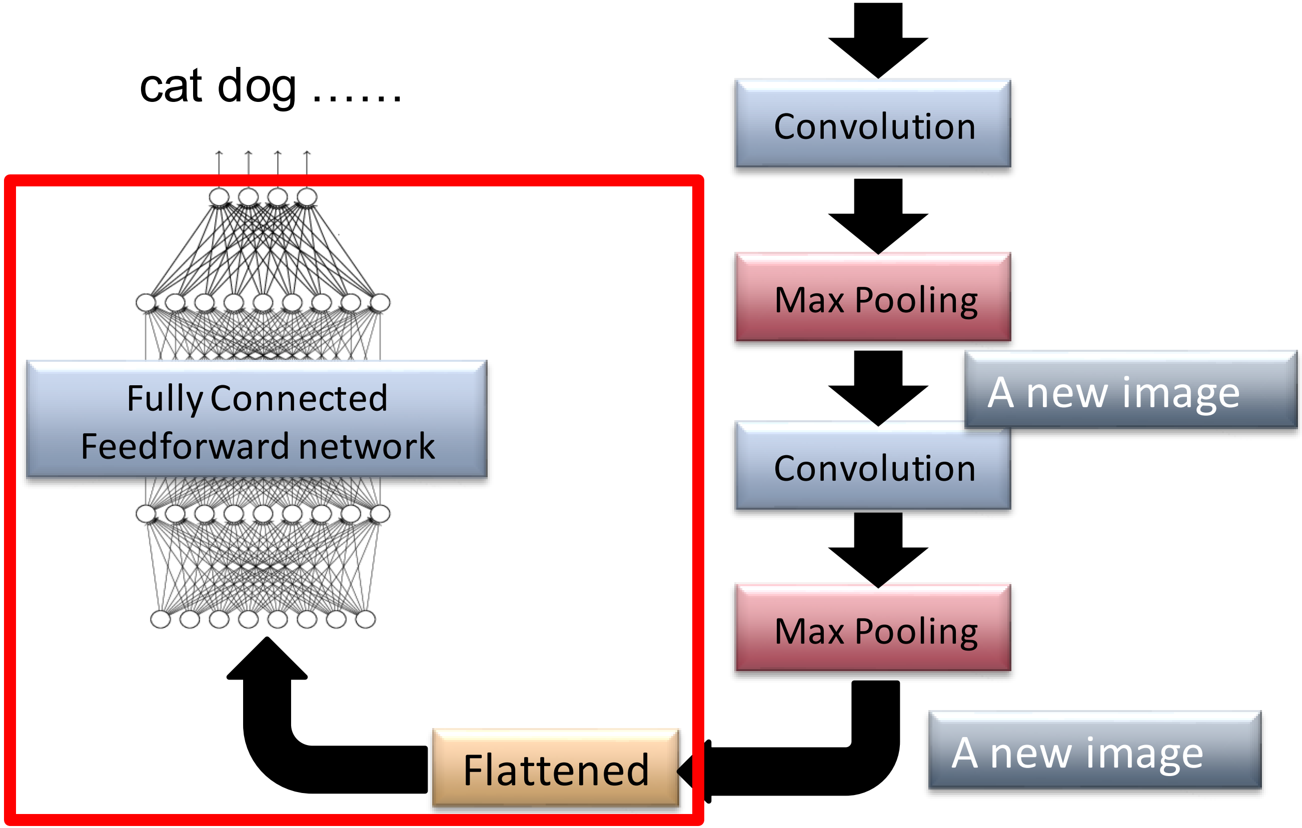




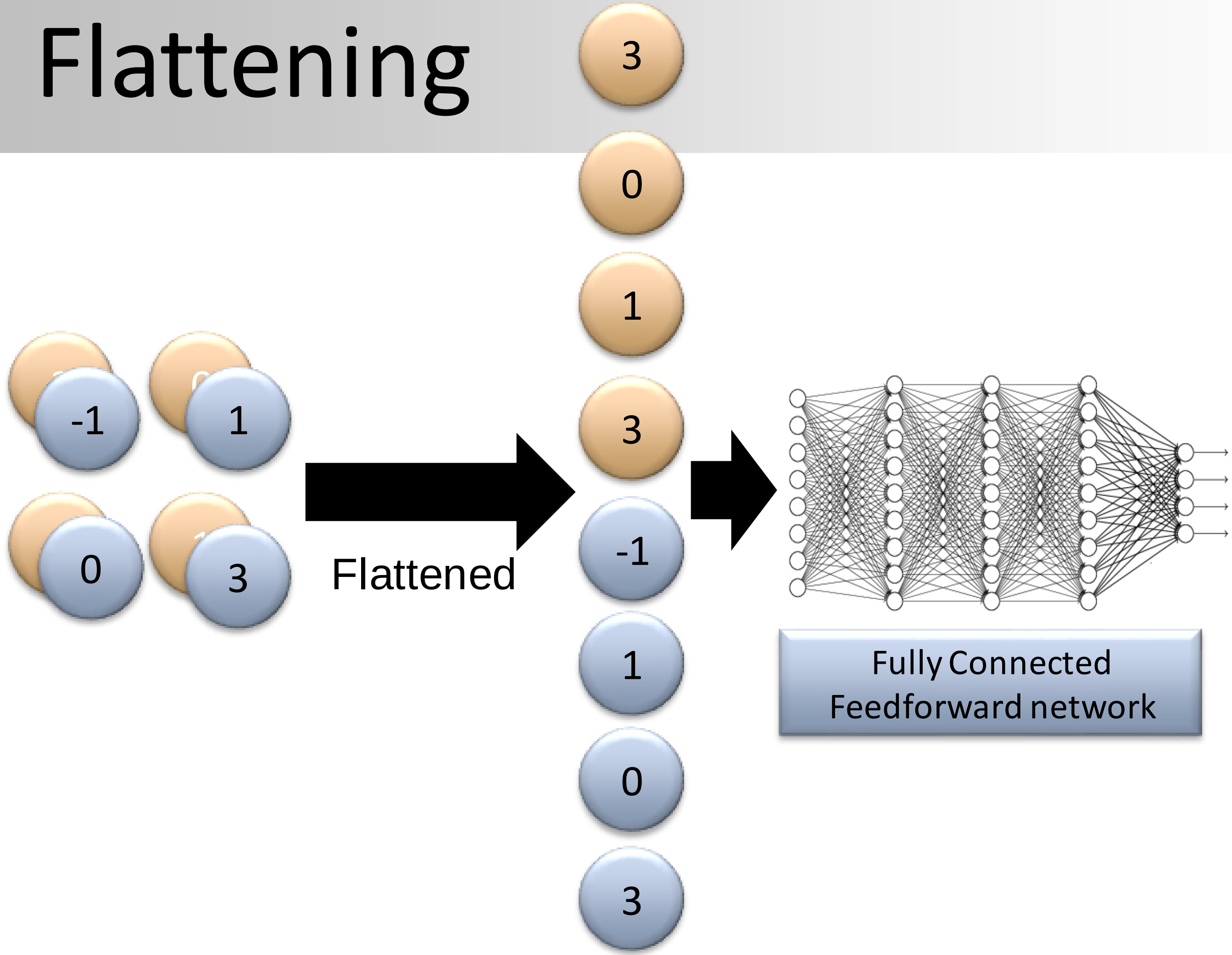
Smaller than the original image

The number of channels is the number of filters

The whole CNN



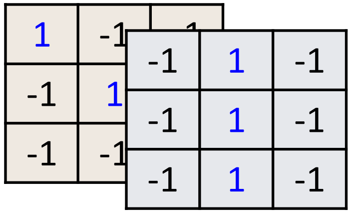
Flattening


-1
1
0
3
Flattened
3
0
1
3
-1
1
0
3
Fully Connected
Feedforward network

CNN in Keras

Only modified the ***network structure*** and ***input format (vector -> 3-D tensor)***

```
model2.add( Convolution2D( 25,3,3,
            input_shape=(28,28,1)) )
```



…

…

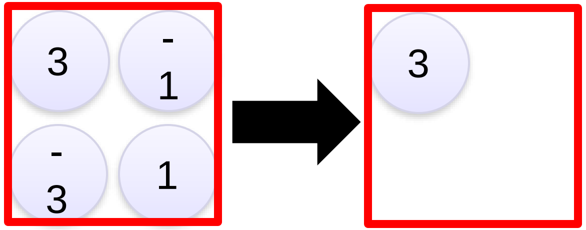
There are 25 3x3 filters.

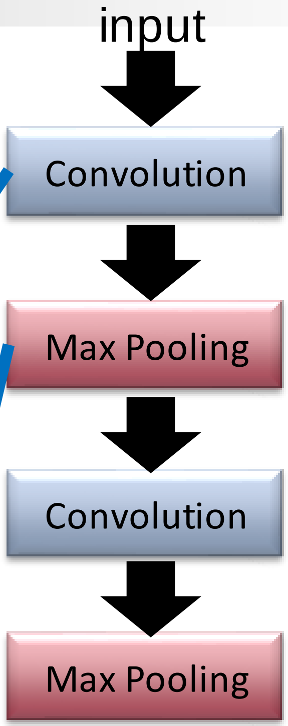
Input_shape = (28 , 28 , 1)

28 x 28 pixels

1: black/white, 3: RGB

```
model2.add(MaxPooling2D((2,2)))
```





CNN in Keras

Only modified the ***network structure*** and ***input format (vector -> 3-D array)***

Input

1 x 28 x 28

Convolution

```
model2.add( Convolution2D( 25,3,3,
            input_shape=(28,28,1)) )
```

How many parameters for each filter? 9

25 x 26 x 26

Max Pooling

```
model2.add(MaxPooling2D((2,2)))
```

25 x 13 x 13

Convolution

```
model2.add(Convolution2D(50,3,3))
```

How many parameters for each filter? 225= 25x9

50 x 11 x 11

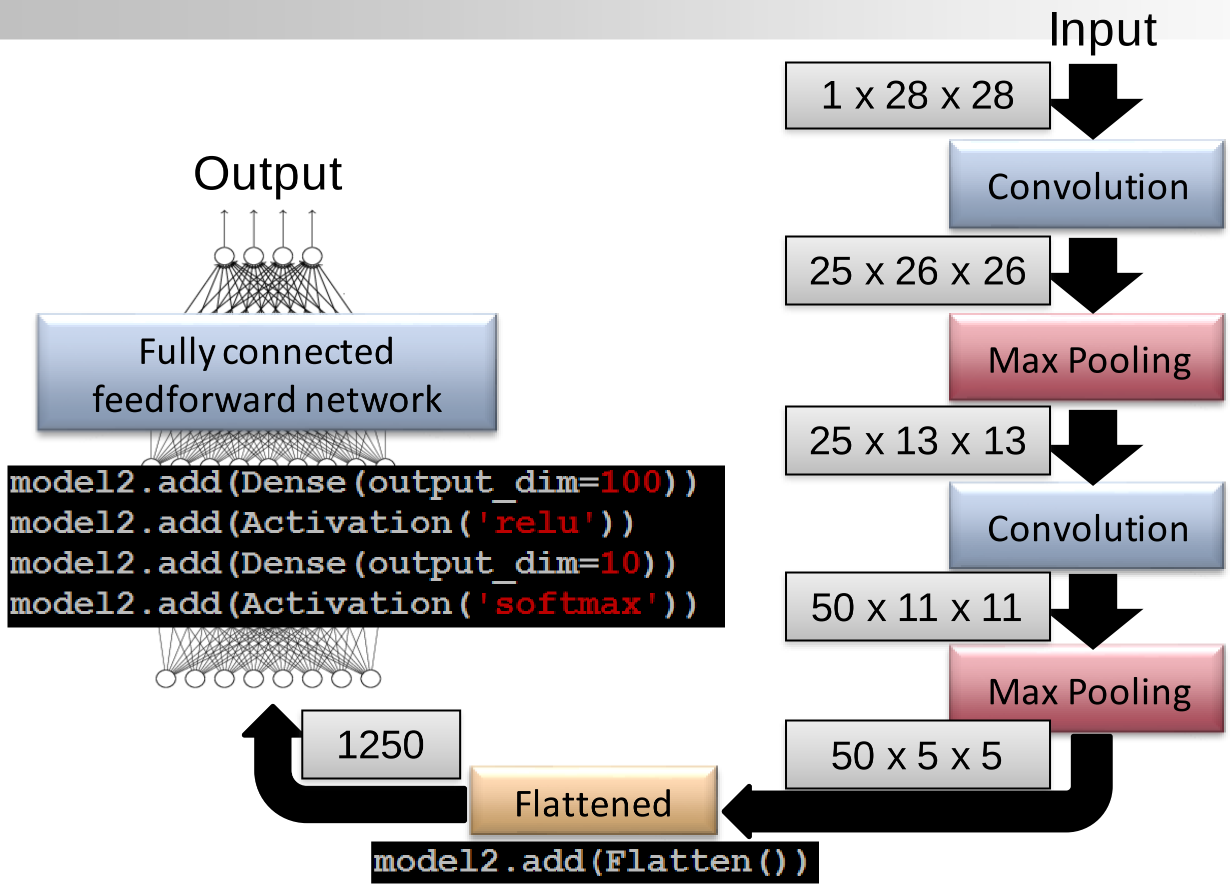
Max Pooling

```
model2.add(MaxPooling2D((2,2)))
```

50 x 5 x 5

CNN in Keras

Only modified the ***network structure*** and ***input format (vector -> 3-D array)***



AlphaGo

19 x 19 matrix

Black: 1

white: -1

none: 0

Neural Network

Next move

(19 x 19 positions)

Fully-connected feedforward network can be used

But CNN performs much better

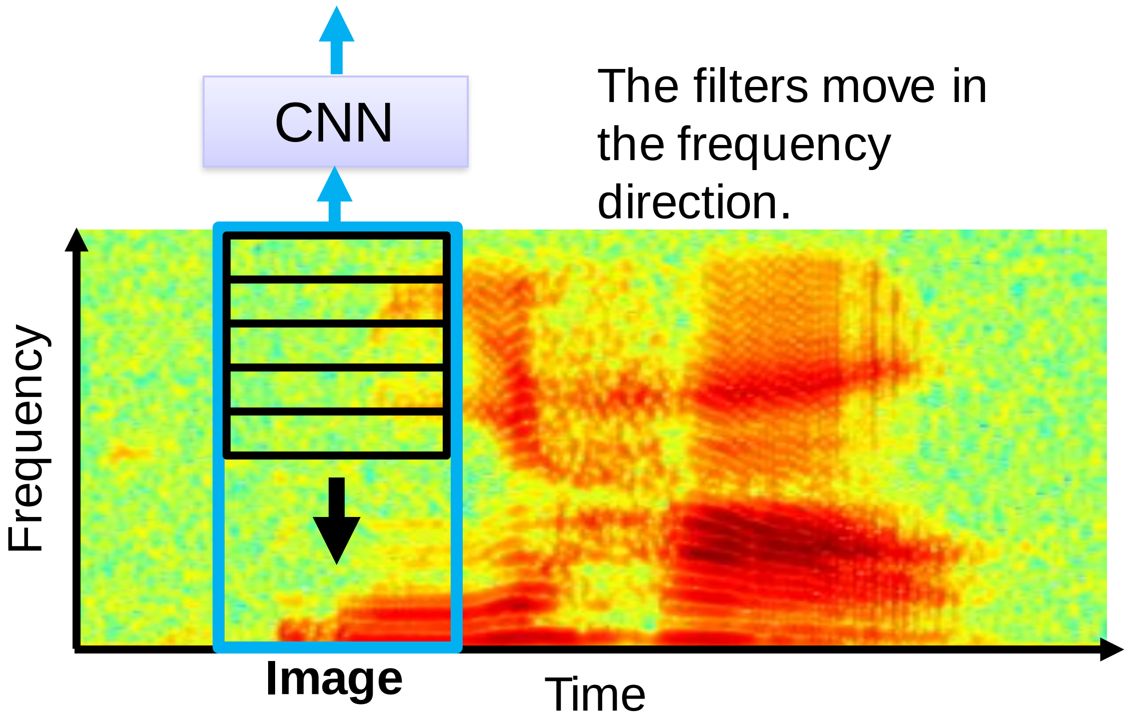
AlphaGo's policy network

The following is quotation from their Nature article:

Note: AlphaGo does not use Max Pooling.

Neural network architecture. The input to the policy network is a $19 \times 19 \times 48$ image stack consisting of 48 feature planes. The first hidden layer zero pads the input into a 23×23 image, then convolves k filters of kernel size 5×5 with stride 1 with the input image and applies a rectifier nonlinearity. Each of the subsequent hidden layers 2 to 12 zero pads the respective previous hidden layer into a 21×21 image, then convolves k filters of kernel size 3×3 with stride 1, again followed by a rectifier nonlinearity. The final layer convolves 1 filter of kernel size 1×1 with stride 1, with a different bias for each position, and applies a softmax function. The match version of AlphaGo used $k = 192$ filters; Fig. 2b and Extended Data Table 3 additionally show the results of training with $k = 128$, 256 and 384 filters.

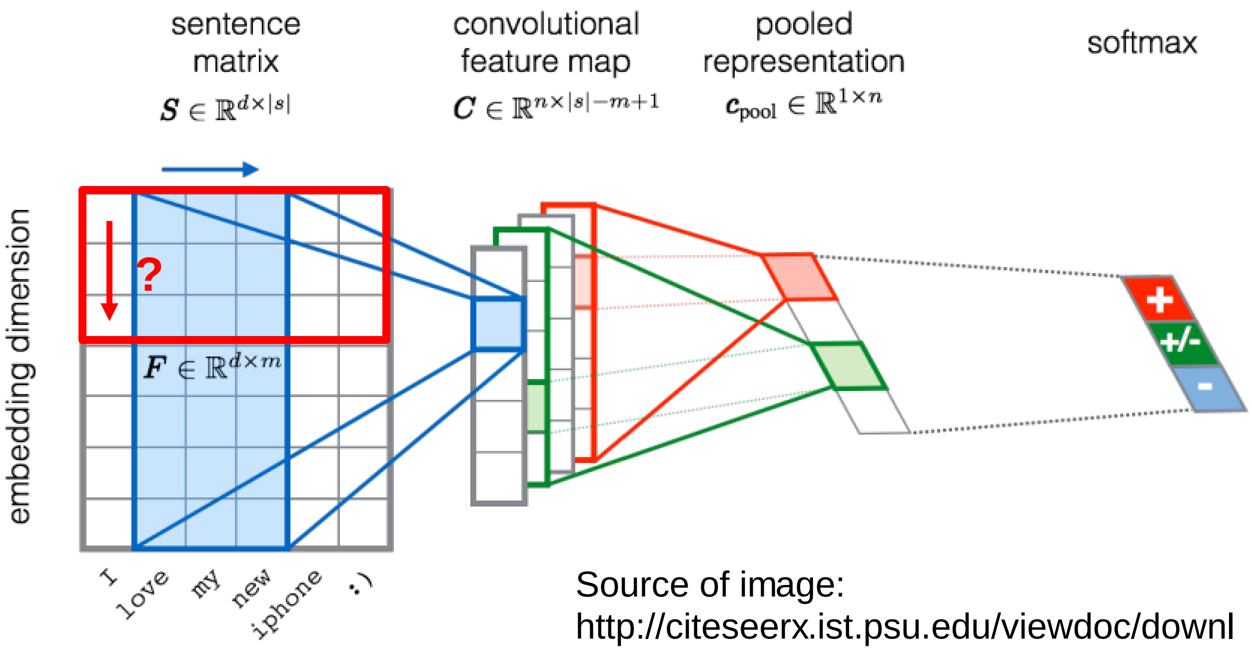
CNN in speech recognition



The filters move in the frequency direction.

Spectrogram

CNN in text classification



Source of image: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.703.6858&rep=rep1&type=pdf

Convolution Backward Propagation

- Horizontal and vertical stride = 1
- Convolution between Input X and Filter F, gives the output O

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \otimes \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} = \begin{bmatrix} o_{11} & o_{12} \\ o_{21} & o_{22} \end{bmatrix}$$

Input X

Filter F

Output O

$$o_{11} = x_{11}f_{11} + x_{12}f_{12} + x_{21}f_{21} + x_{22}f_{22}$$

Convolution Backward Propagation

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \otimes \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} = \begin{bmatrix} o_{11} & o_{12} \\ o_{21} & o_{22} \end{bmatrix}$$

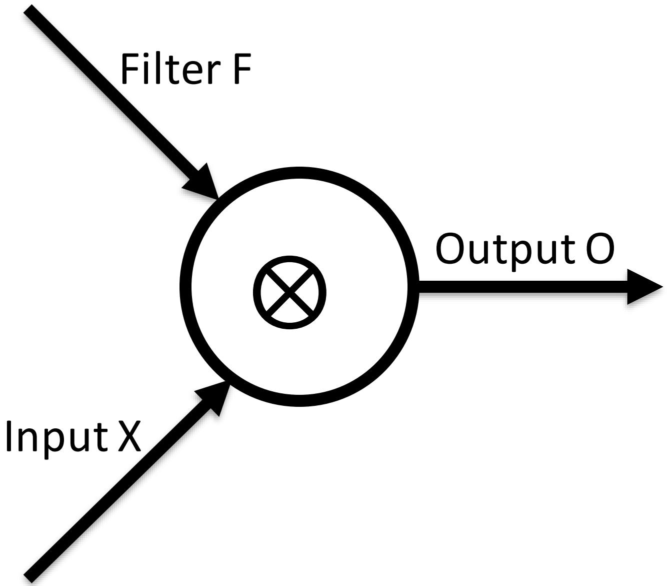
Input X

Filter F

Output O

$$o_{11} = x_{11}f_{11} + x_{12}f_{12} + x_{21}f_{21} + x_{22}f_{22}$$
$$o_{12} = x_{12}f_{11} + x_{13}f_{12} + x_{22}f_{21} + x_{23}f_{22}$$
$$o_{21} = x_{21}f_{11} + x_{22}f_{12} + x_{31}f_{21} + x_{32}f_{22}$$
$$o_{22} = x_{22}f_{11} + x_{23}f_{12} + x_{32}f_{21} + x_{33}f_{22}$$

Loss Gradient



$$\frac{\partial L}{\partial F} = \frac{\partial L}{\partial O} * \frac{\partial O}{\partial F}$$

Gradient to update Filter F

Loss Gradient from previous layer

Local gradient

For every element of F

$$\frac{\partial L}{\partial f_{ij}} = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\partial L}{\partial o_{nm}} * \frac{\partial o_{nm}}{\partial f_{ij}}$$

Loss Gradient w.r.t the Filter

- $\frac{\partial L}{\partial f_{ij}} = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\partial L}{\partial o_{nm}} * \frac{\partial o_{nm}}{\partial f_{ij}}$

We can expand the chain rule summation as:

- $\frac{\partial L}{\partial f_{11}} = \frac{\partial L}{\partial o_{11}} * \frac{\partial o_{11}}{\partial f_{11}} + \frac{\partial L}{\partial o_{12}} * \frac{\partial o_{12}}{\partial f_{11}} + \frac{\partial L}{\partial o_{21}} * \frac{\partial o_{21}}{\partial f_{11}} + \frac{\partial L}{\partial o_{22}} * \frac{\partial o_{22}}{\partial f_{11}}$
- $\frac{\partial L}{\partial f_{12}} = \frac{\partial L}{\partial o_{11}} * \frac{\partial o_{11}}{\partial f_{12}} + \frac{\partial L}{\partial o_{12}} * \frac{\partial o_{12}}{\partial f_{12}} + \frac{\partial L}{\partial o_{21}} * \frac{\partial o_{21}}{\partial f_{12}} + \frac{\partial L}{\partial o_{22}} * \frac{\partial o_{22}}{\partial f_{12}}$
- $\frac{\partial L}{\partial f_{21}} = \frac{\partial L}{\partial o_{11}} * \frac{\partial o_{11}}{\partial f_{21}} + \frac{\partial L}{\partial o_{12}} * \frac{\partial o_{12}}{\partial f_{21}} + \frac{\partial L}{\partial o_{21}} * \frac{\partial o_{21}}{\partial f_{21}} + \frac{\partial L}{\partial o_{22}} * \frac{\partial o_{22}}{\partial f_{21}}$
- $\frac{\partial L}{\partial f_{22}} = \frac{\partial L}{\partial o_{11}} * \frac{\partial o_{11}}{\partial f_{22}} + \frac{\partial L}{\partial o_{12}} * \frac{\partial o_{12}}{\partial f_{22}} + \frac{\partial L}{\partial o_{21}} * \frac{\partial o_{21}}{\partial f_{22}} + \frac{\partial L}{\partial o_{22}} * \frac{\partial o_{22}}{\partial f_{22}}$

Loss Gradient w.r.t the Filter

$$\frac{\partial L}{\partial f_{ij}} = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\partial L}{\partial o_{nm}} * \frac{\partial o_{nm}}{\partial f_{ij}}$$

$$\frac{\partial L}{\partial f_{ij}}$$

$$\begin{bmatrix} \frac{\partial L}{\partial f_{11}} & \frac{\partial L}{\partial f_{12}} \\ \frac{\partial L}{\partial f_{21}} & \frac{\partial L}{\partial f_{22}} \end{bmatrix} = \begin{bmatrix} \frac{\partial L}{\partial o_{11}} & \frac{\partial L}{\partial o_{12}} \\ \frac{\partial L}{\partial o_{21}} & \frac{\partial L}{\partial o_{22}} \end{bmatrix} \otimes \begin{bmatrix} \frac{\partial o_{11}}{\partial f_{11}} & \frac{\partial o_{12}}{\partial f_{12}} \\ \frac{\partial o_{21}}{\partial f_{21}} & \frac{\partial o_{22}}{\partial f_{22}} \end{bmatrix}$$

$$\begin{bmatrix} \frac{\partial L}{\partial f_{11}} & \frac{\partial L}{\partial f_{12}} \\ \frac{\partial L}{\partial f_{21}} & \frac{\partial L}{\partial f_{22}} \end{bmatrix} = \begin{bmatrix} \frac{\partial L}{\partial o_{11}} & \frac{\partial L}{\partial o_{12}} \\ \frac{\partial L}{\partial o_{21}} & \frac{\partial L}{\partial o_{22}} \end{bmatrix} \otimes \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}$$

Loss Gradient w.r.t the Filter

$$\begin{bmatrix} \frac{\partial L}{\partial f_{11}} & \frac{\partial L}{\partial f_{12}} \\ \frac{\partial L}{\partial f_{21}} & \frac{\partial L}{\partial f_{22}} \end{bmatrix} = \begin{bmatrix} \frac{\partial L}{\partial o_{11}} & \frac{\partial L}{\partial o_{12}} \\ \frac{\partial L}{\partial o_{21}} & \frac{\partial L}{\partial o_{22}} \end{bmatrix} \otimes \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}$$

$$\frac{\partial L}{\partial f_{11}} = \frac{\partial L}{\partial o_{11}} * x_{11} + \frac{\partial L}{\partial o_{12}} * x_{12} + \frac{\partial L}{\partial o_{21}} * x_{21} + \frac{\partial L}{\partial o_{22}} * x_{22}$$

$$\frac{\partial L}{\partial f_{12}} = \frac{\partial L}{\partial o_{11}} * x_{12} + \frac{\partial L}{\partial o_{12}} * x_{13} + \frac{\partial L}{\partial o_{21}} * x_{22} + \frac{\partial L}{\partial o_{22}} * x_{23}$$

$$\frac{\partial L}{\partial f_{21}} = \frac{\partial L}{\partial o_{11}} * x_{21} + \frac{\partial L}{\partial o_{12}} * x_{22} + \frac{\partial L}{\partial o_{21}} * x_{31} + \frac{\partial L}{\partial o_{22}} * x_{32}$$

$$\frac{\partial L}{\partial f_{22}} = \frac{\partial L}{\partial o_{11}} * x_{22} + \frac{\partial L}{\partial o_{12}} * x_{23} + \frac{\partial L}{\partial o_{21}} * x_{32} + \frac{\partial L}{\partial o_{22}} * x_{33}$$

Loss Gradient w.r.t the Input

- Why calculate the gradient w.r.t the input?

For every element of X

$$\frac{\partial L}{\partial x_{ij}} = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\partial L}{\partial o_{nm}} * \frac{\partial o_{nm}}{\partial x_{ij}}$$

$\frac{\partial L}{\partial x_{11}}$	$\frac{\partial L}{\partial x_{12}}$	$\frac{\partial L}{\partial x_{13}}$
$\frac{\partial L}{\partial x_{21}}$	$\frac{\partial L}{\partial x_{22}}$	$\frac{\partial L}{\partial x_{23}}$
$\frac{\partial L}{\partial x_{31}}$	$\frac{\partial L}{\partial x_{32}}$	$\frac{\partial L}{\partial x_{33}}$

$$\frac{\partial o_{11}}{\partial x_{11}} = \frac{\partial}{\partial x_{11}}(x_{11}f_{11} + x_{12}f_{12} + x_{21}f_{21} + x_{22}f_{22}) = f_{11}$$

$$\frac{\partial o_{12}}{\partial x_{11}} = \frac{\partial}{\partial x_{11}}(x_{12}f_{11} + x_{13}f_{12} + x_{22}f_{21} + x_{23}f_{22}) = 0$$

$$\frac{\partial o_{21}}{\partial x_{11}} = \frac{\partial}{\partial x_{11}}(x_{21}f_{11} + x_{22}f_{12} + x_{31}f_{21} + x_{32}f_{22}) = 0$$

$$\frac{\partial o_{22}}{\partial x_{11}} = \frac{\partial}{\partial x_{11}}(x_{22}f_{11} + x_{23}f_{12} + x_{32}f_{21} + x_{33}f_{22}) = 0$$

Loss Gradient w.r.t the Input

For every element of X, $\frac{\partial L}{\partial x_{ij}} = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\partial L}{\partial o_{nm}} * \frac{\partial o_{nm}}{\partial x_{ij}}$

$$\frac{\partial L}{\partial x_{11}} = \frac{\partial L}{\partial o_{11}} f_{11}, \frac{\partial L}{\partial x_{12}} = \frac{\partial L}{\partial o_{11}} f_{12} + \frac{\partial L}{\partial o_{12}} f_{11}, \ldots$$

$\frac{\partial L}{\partial x_{11}}$	$\frac{\partial L}{\partial x_{12}}$	$\frac{\partial L}{\partial x_{13}}$
$\frac{\partial L}{\partial x_{21}}$	$\frac{\partial L}{\partial x_{22}}$	$\frac{\partial L}{\partial x_{23}}$
$\frac{\partial L}{\partial x_{31}}$	$\frac{\partial L}{\partial x_{32}}$	$\frac{\partial L}{\partial x_{33}}$

$$\frac{\partial o_{11}}{\partial x_{12}} = \frac{\partial}{\partial x_{12}}(x_{11}f_{11} + x_{12}f_{12} + x_{21}f_{21} + x_{22}f_{22}) = f_{12}$$

$$\frac{\partial o_{12}}{\partial x_{12}} = \frac{\partial}{\partial x_{12}}(x_{12}f_{11} + x_{13}f_{12} + x_{22}f_{21} + x_{23}f_{22}) = f_{11}$$

$$\frac{\partial o_{21}}{\partial x_{12}} = \frac{\partial}{\partial x_{12}}(x_{21}f_{11} + x_{22}f_{12} + x_{31}f_{21} + x_{32}f_{22}) = 0$$

$$\frac{\partial o_{22}}{\partial x_{12}} = \frac{\partial}{\partial x_{12}}(x_{22}f_{11} + x_{23}f_{12} + x_{32}f_{21} + x_{33}f_{22}) = 0$$

Loss Gradient w.r.t the Input

For every element of X, $\frac{\partial L}{\partial x_{ij}} = \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\partial L}{\partial o_{nm}} * \frac{\partial o_{nm}}{\partial x_{ij}}$

$$\frac{\partial L}{\partial x_{11}} = \frac{\partial L}{\partial o_{11}} f_{11},$$

$$\frac{\partial L}{\partial x_{12}} = \frac{\partial L}{\partial o_{11}} f_{12} + \frac{\partial L}{\partial o_{12}} f_{11},$$

$$\frac{\partial L}{\partial x_{13}} = \frac{\partial L}{\partial o_{12}} f_{12},$$

$$\frac{\partial L}{\partial x_{21}} = \frac{\partial L}{\partial o_{11}} f_{21} + \frac{\partial L}{\partial o_{21}} f_{11},$$

$$\frac{\partial L}{\partial x_{22}} = \frac{\partial L}{\partial o_{11}} f_{22} + \frac{\partial L}{\partial o_{21}} f_{21} + \frac{\partial L}{\partial o_{21}} f_{12} + \frac{\partial L}{\partial o_{22}} f_{11},$$

$$\frac{\partial L}{\partial x_{23}} = \frac{\partial L}{\partial o_{12}} f_{22} + \frac{\partial L}{\partial o_{22}} f_{12},$$

$$\frac{\partial L}{\partial x_{31}} = \frac{\partial L}{\partial o_{21}} f_{21},$$

$$\frac{\partial L}{\partial x_{32}} = \frac{\partial L}{\partial o_{21}} f_{22} + \frac{\partial L}{\partial o_{22}} f_{21},$$

$$\frac{\partial L}{\partial x_{33}} = \frac{\partial L}{\partial o_{22}} f_{22}$$

$$\begin{array}{|c|c|c|c|} \hline 0 & 0 & 0 & 0 \\ \hline 0 & \frac{\partial L}{\partial o_{11}} & \frac{\partial L}{\partial o_{12}} & 0 \\ \hline 0 & \frac{\partial L}{\partial o_{21}} & \frac{\partial L}{\partial o_{22}} & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array}$$

$$\otimes$$

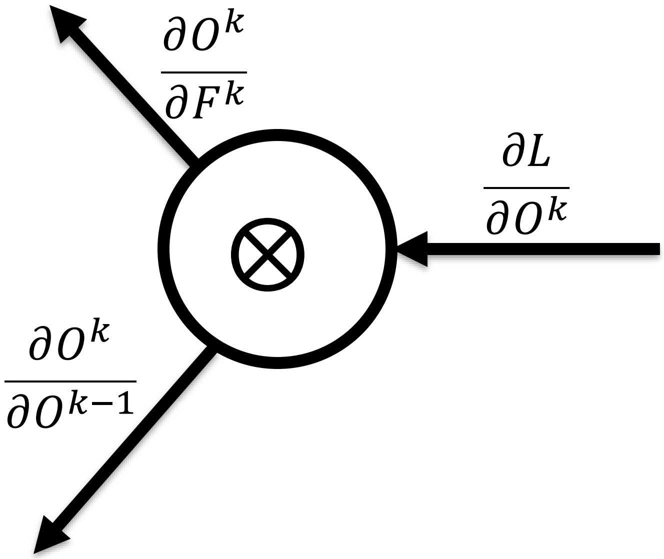
$$\begin{array}{|c|c|} \hline f_{22} & f_{21} \\ \hline f_{12} & f_{11} \\ \hline \end{array}$$

Loss Gradient w.r.t the Input II

$$\begin{bmatrix} \frac{\partial L}{\partial x_{11}} & \frac{\partial L}{\partial x_{12}} & \frac{\partial L}{\partial x_{13}} \\ \frac{\partial L}{\partial x_{21}} & \frac{\partial L}{\partial x_{22}} & \frac{\partial L}{\partial x_{23}} \\ \frac{\partial L}{\partial x_{31}} & \frac{\partial L}{\partial x_{32}} & \frac{\partial L}{\partial x_{33}} \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} & 0 \\ f_{21} & f_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \frac{\partial L}{\partial o_{11}} + \begin{bmatrix} 0 & f_{11} & f_{12} \\ 0 & f_{21} & f_{22} \\ 0 & 0 & 0 \end{bmatrix} \times \frac{\partial L}{\partial o_{12}}$$

$$+ \begin{bmatrix} 0 & 0 & 0 \\ f_{11} & f_{12} & 0 \\ f_{21} & f_{22} & 0 \end{bmatrix} \times \frac{\partial L}{\partial o_{21}} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & f_{11} & f_{12} \\ 0 & f_{21} & f_{22} \end{bmatrix} \times \frac{\partial L}{\partial o_{22}}$$

Backward Propagation



$$\frac{\partial L}{\partial F^k} = \frac{\partial L}{\partial O^k} * \frac{\partial O^k}{\partial F^k}$$

$$F^k = F^k - lr * \frac{\partial L}{\partial F^k}$$

$$\frac{\partial L}{\partial O^{k-1}} = \frac{\partial L}{\partial O^k} * \frac{\partial O^k}{\partial O^{k-1}}$$