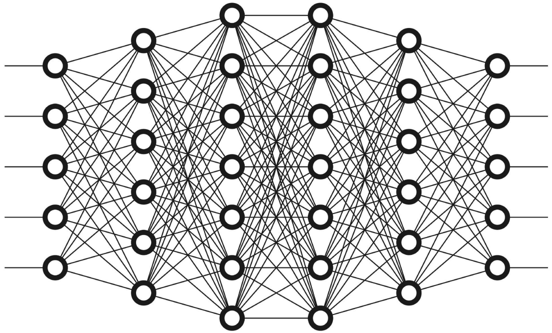


Introduction to Deep Learning

Fall 2018

Convolutional Neural Networks

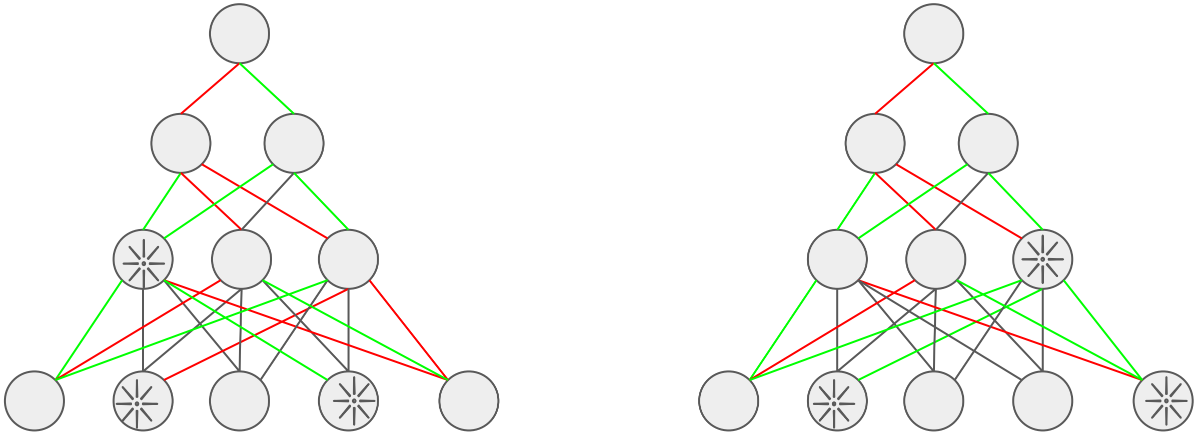
Elan Barenholtz

Feature Extraction in the Hidden Layers

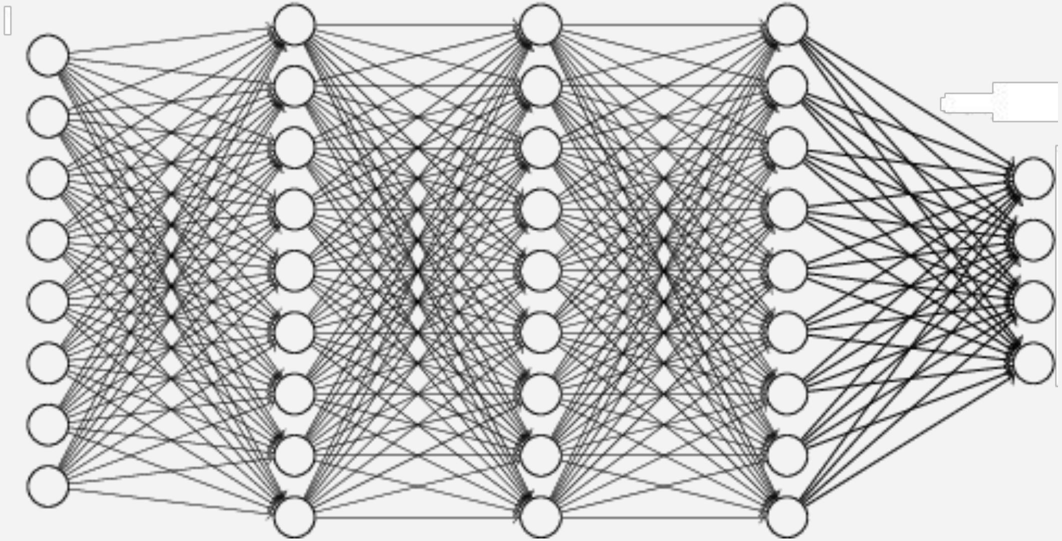
- The goal of each layer of an ANNs is to represent features/patterns present in the previous layer

- These are the *dimensions* of that layer

- They are *learned*

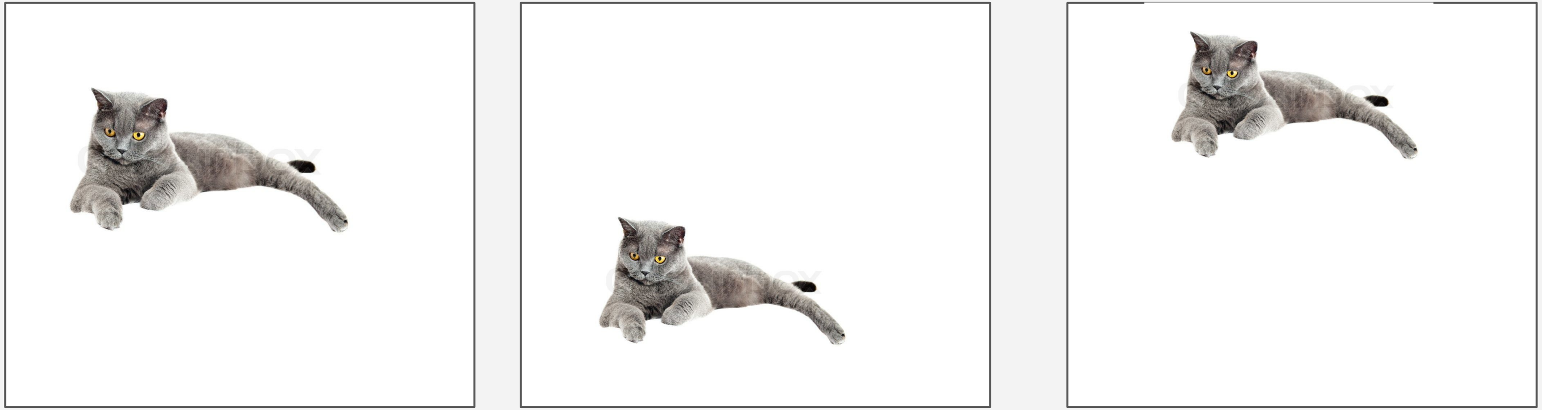


Fully connected multi-layer network



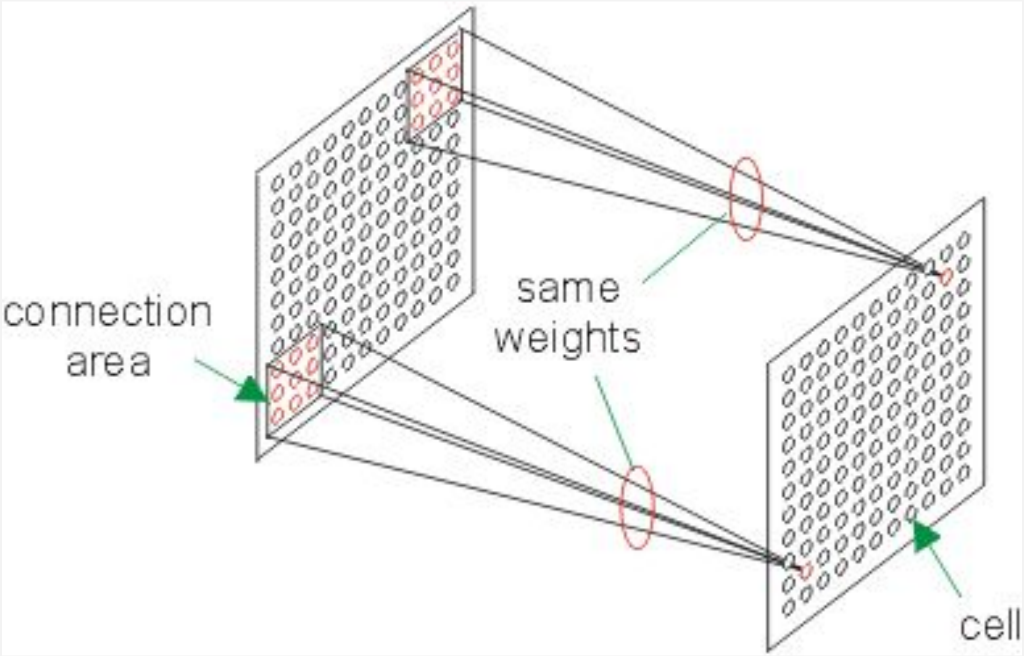
Large number of parameters
Computationally expensive to train

Structure in images is (often) local and repeated



Fully connected network would have to learn each of these independently

Weight Sharing



Convolutional Neural Networks

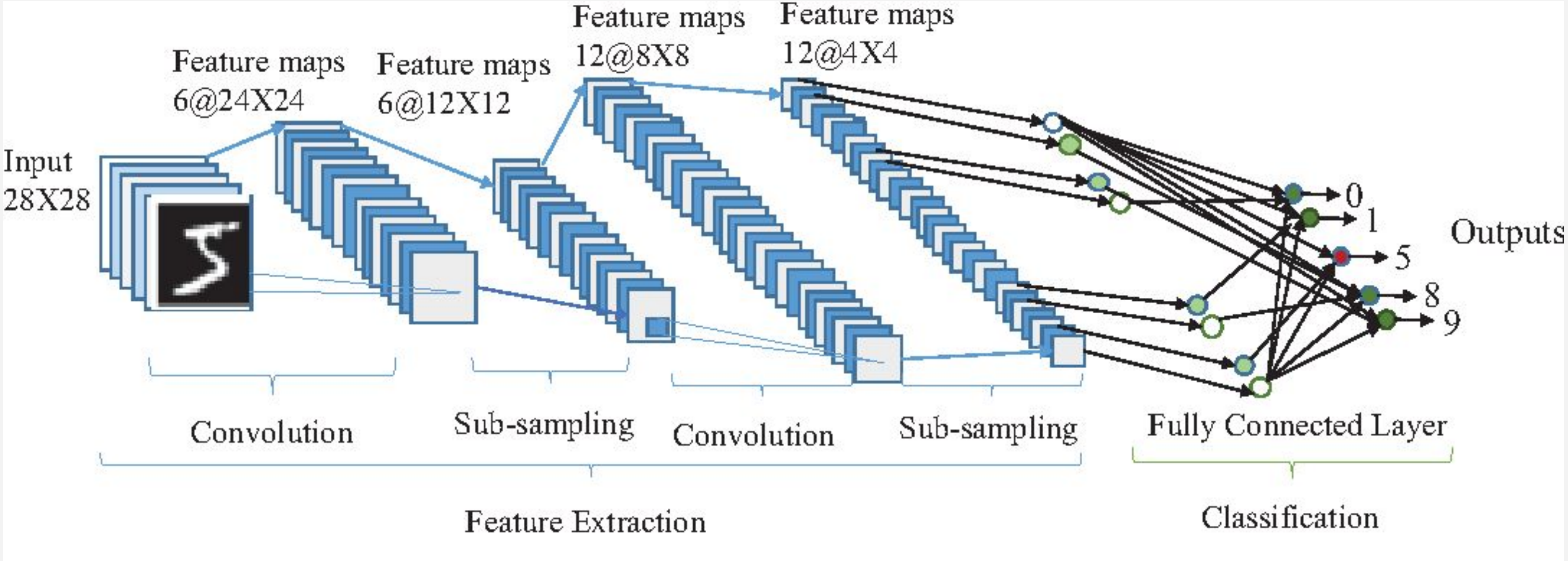
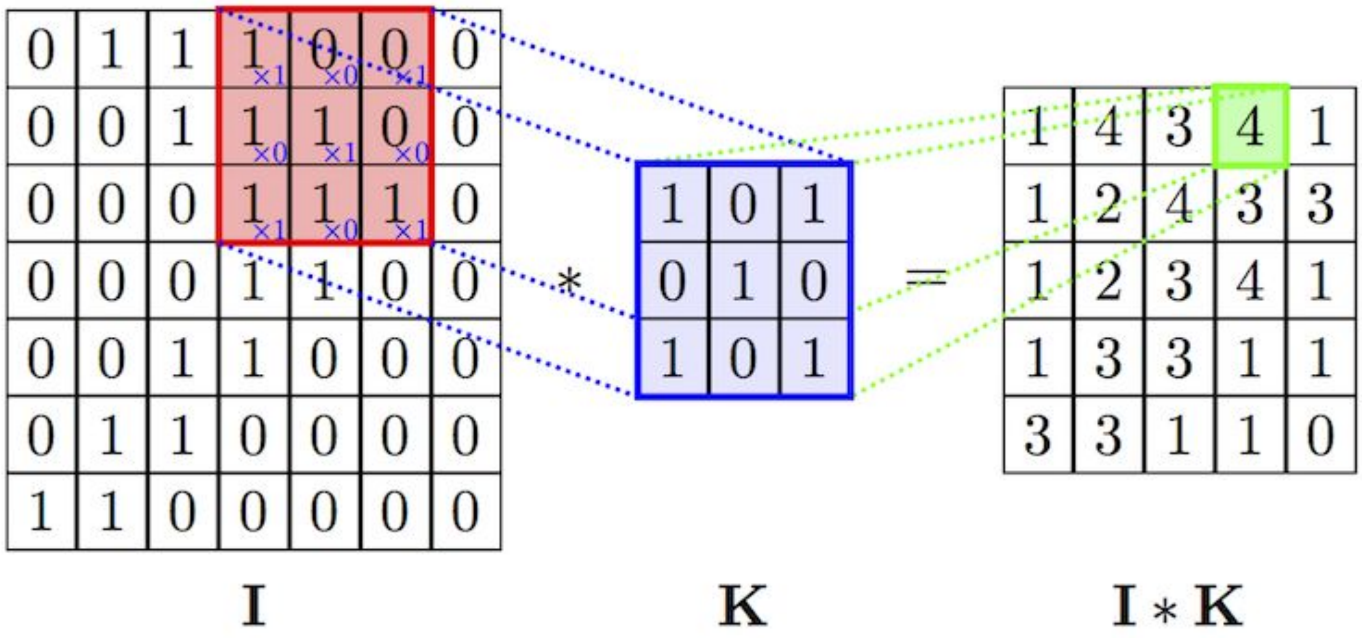


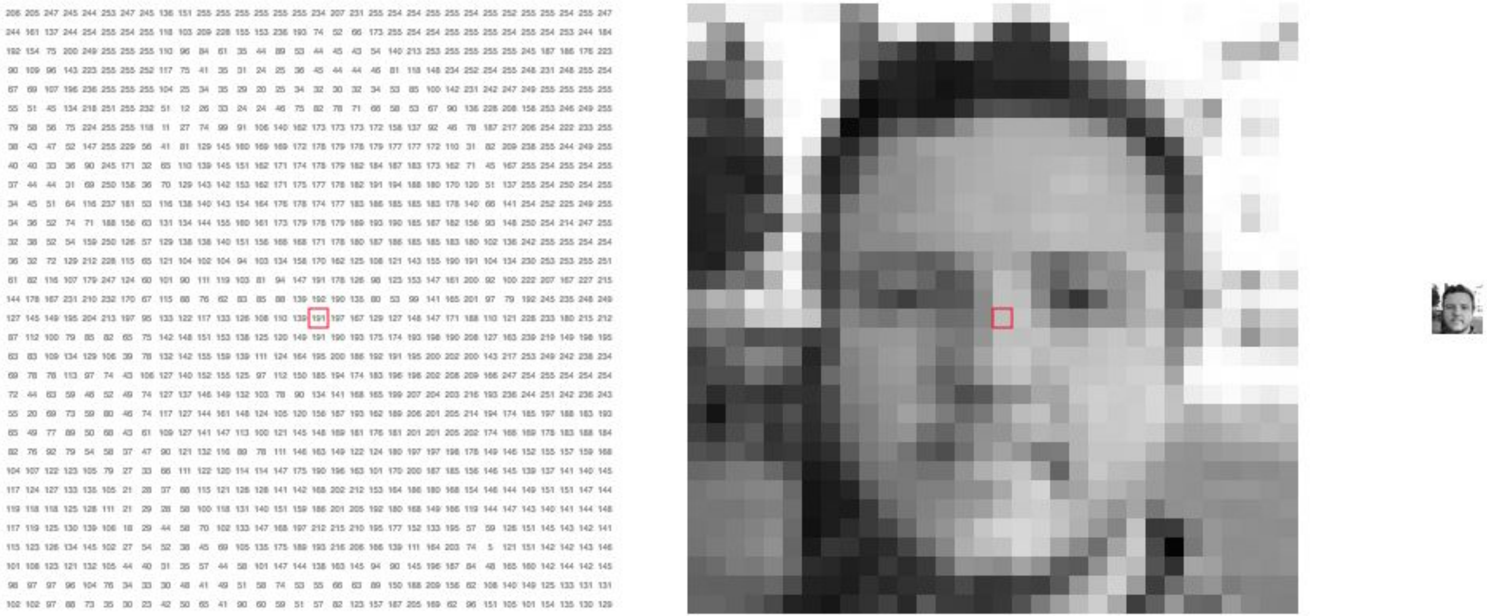
Fig. 1. CNN Block Diagram

Convolutions: Multiplication and Summation

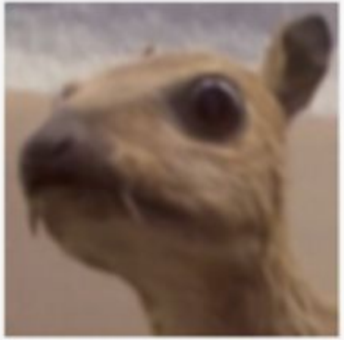


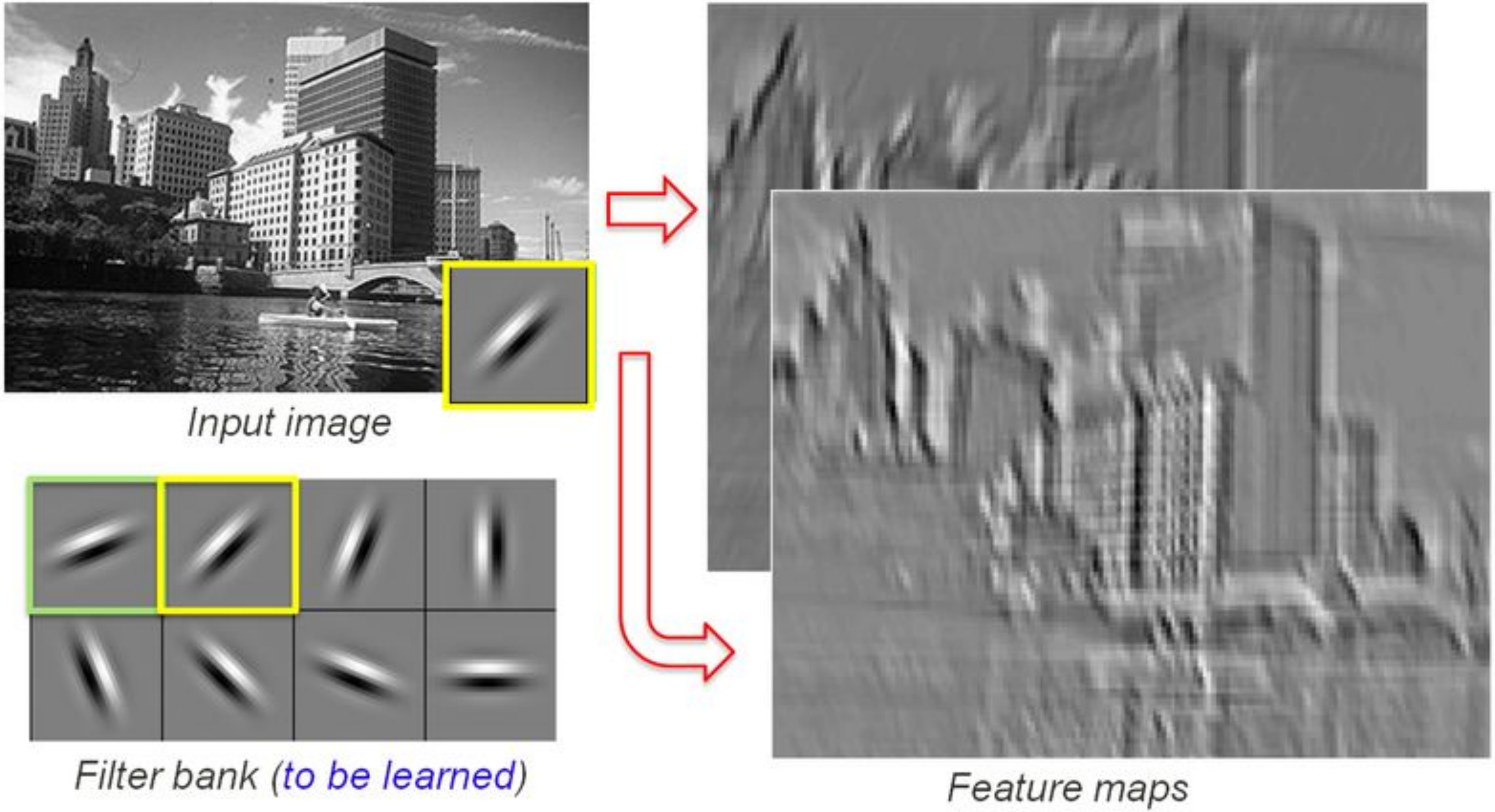
$$\begin{bmatrix} 7 & 2 & 3 & 3 & 8 \\ 4 & 5 & 3 & 8 & 4 \\ 3 & 3 & 2 & 8 & 4 \\ 2 & 8 & 7 & 2 & 7 \\ 5 & 4 & 4 & 5 & 4 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 6 & & \\ & & \\ & & \end{bmatrix}$$

7x1+4x1+3x1+
2x0+5x0+3x0+
3x-1+3x-1+2x-1
= 6



http://setosa.io/ev/image-kernels/

Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	


Input image
Filter bank (to be learned)
Feature maps

Input

Image Size: 5 X 5
Padding Size: 1
Kernel Size: 3 X 3
Stride: 2
Feature Map: 3X3

Image Size: 5 X 5
Padding Size: 1
Kernel Size: 3 X 3
Stride: 1
Feature Map: 5 X 5

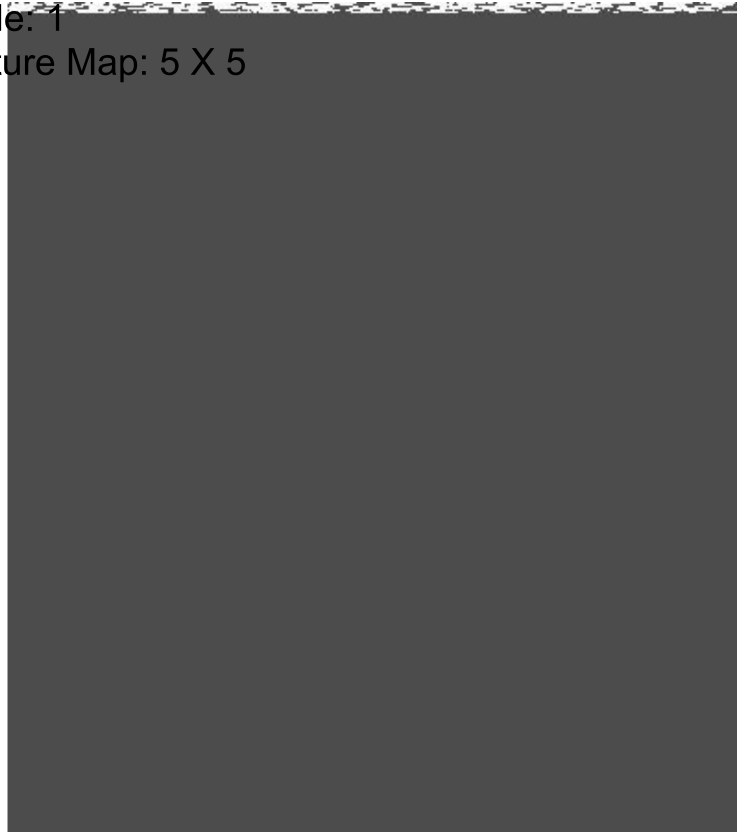
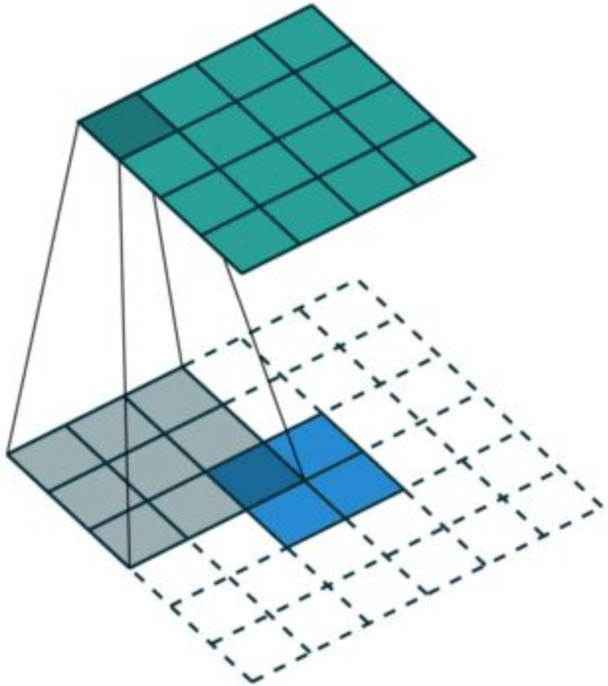
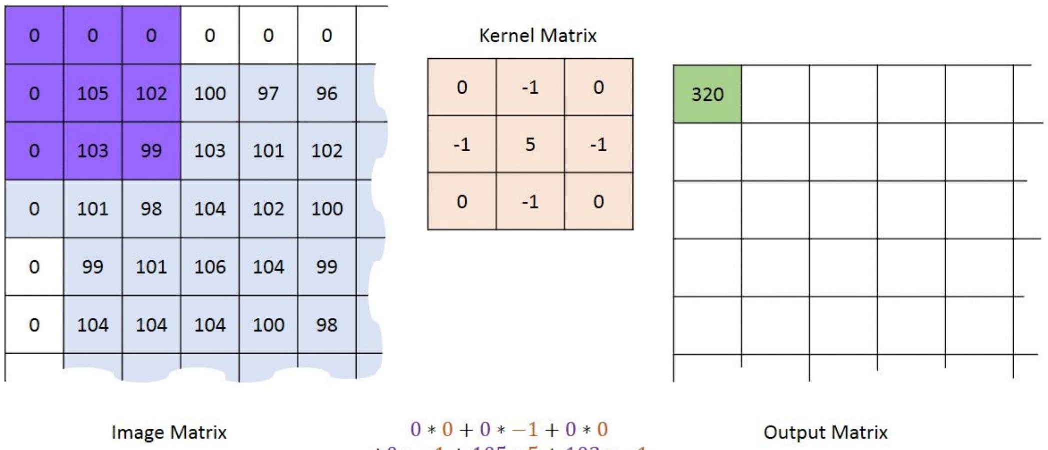


Image Size: 2 X 2
Padding Size: 2
Kernel Size: 3 X 3
Stride: 2
Feature Map: 4X4

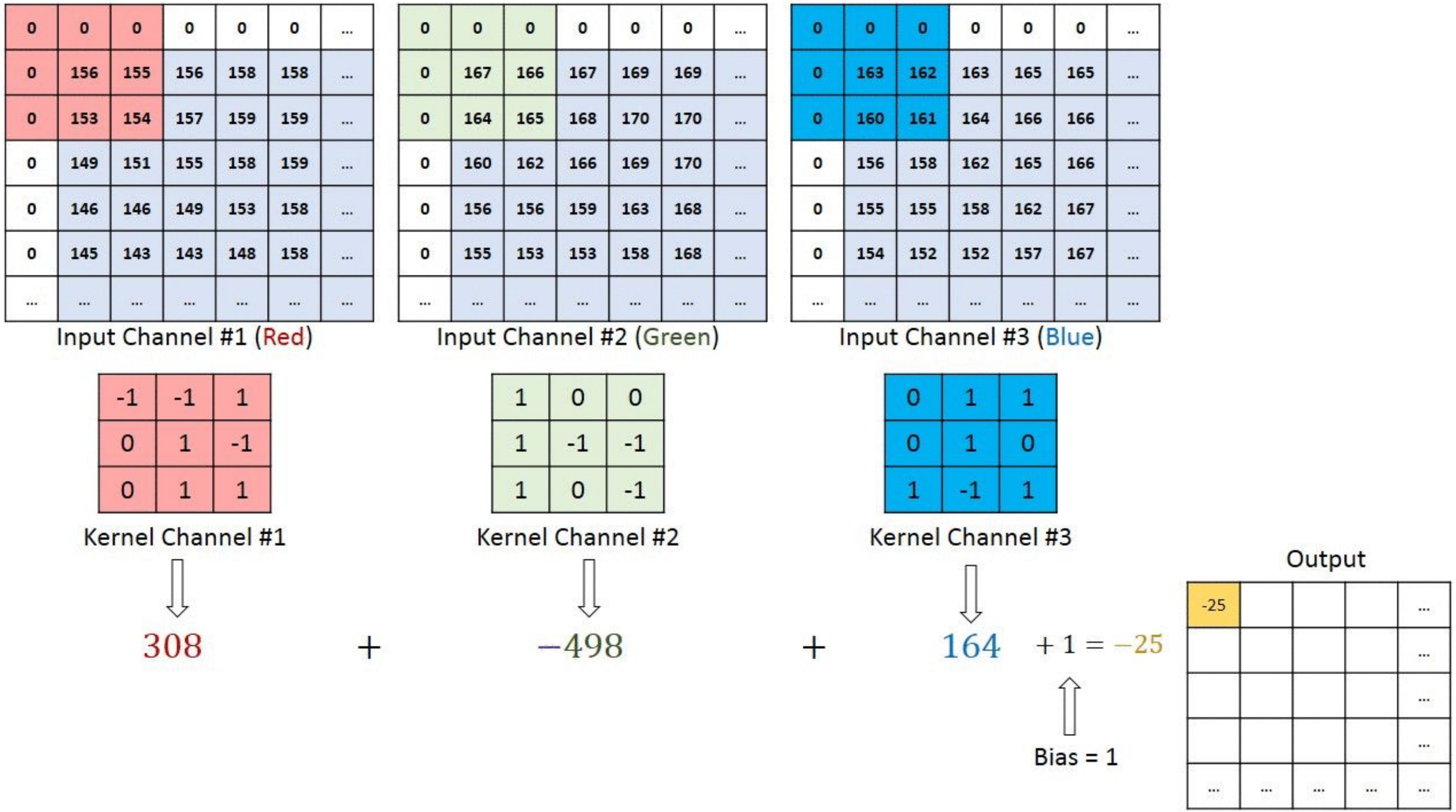


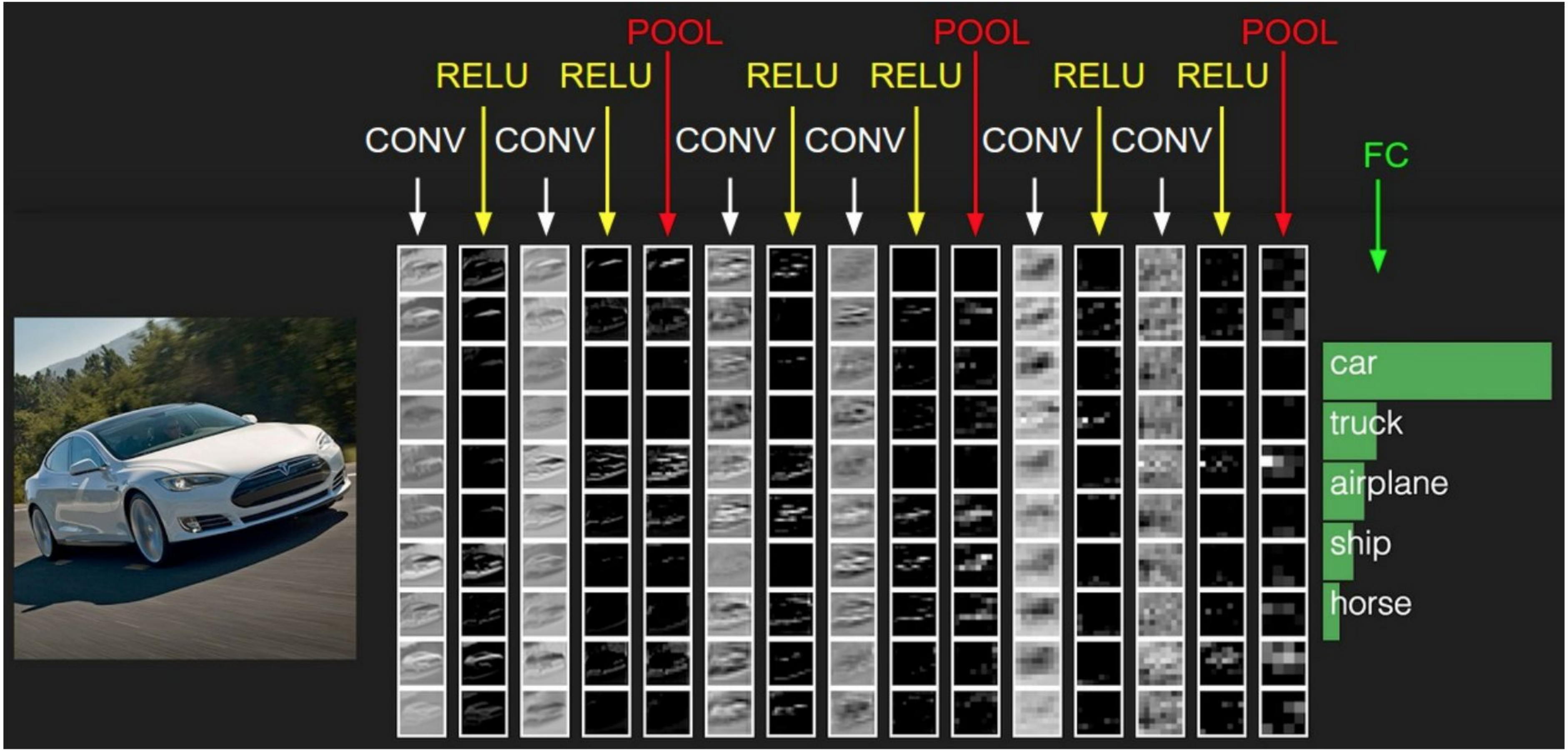


$$
\begin{aligned}
&0*0+0*-1+0*0\\
&+0*-1+105*5+102*-1\\
&+0*0+103*-1+99*0 \quad =320
\end{aligned}
$$

Convolution with horizontal and vertical strides = 1

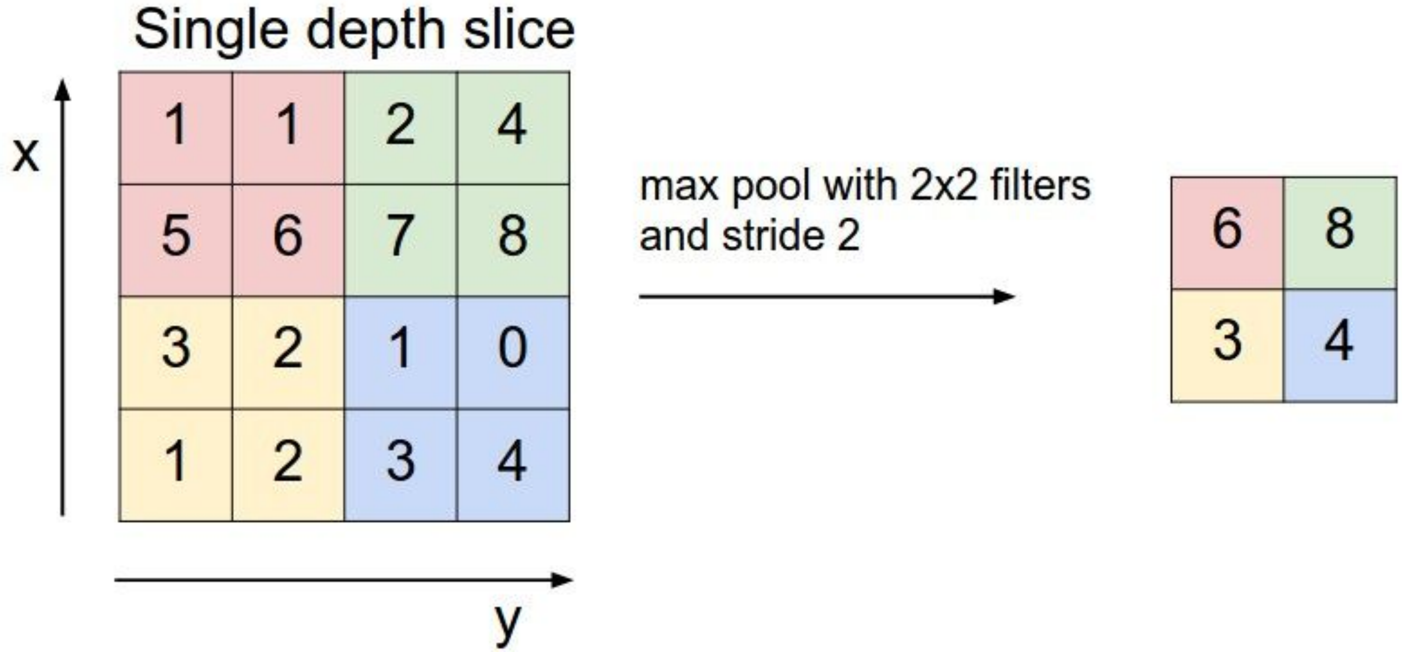
3d (RGB) Input → 1D output





POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

Pooling to reduce dimensionality



Convolutional Neural Networks

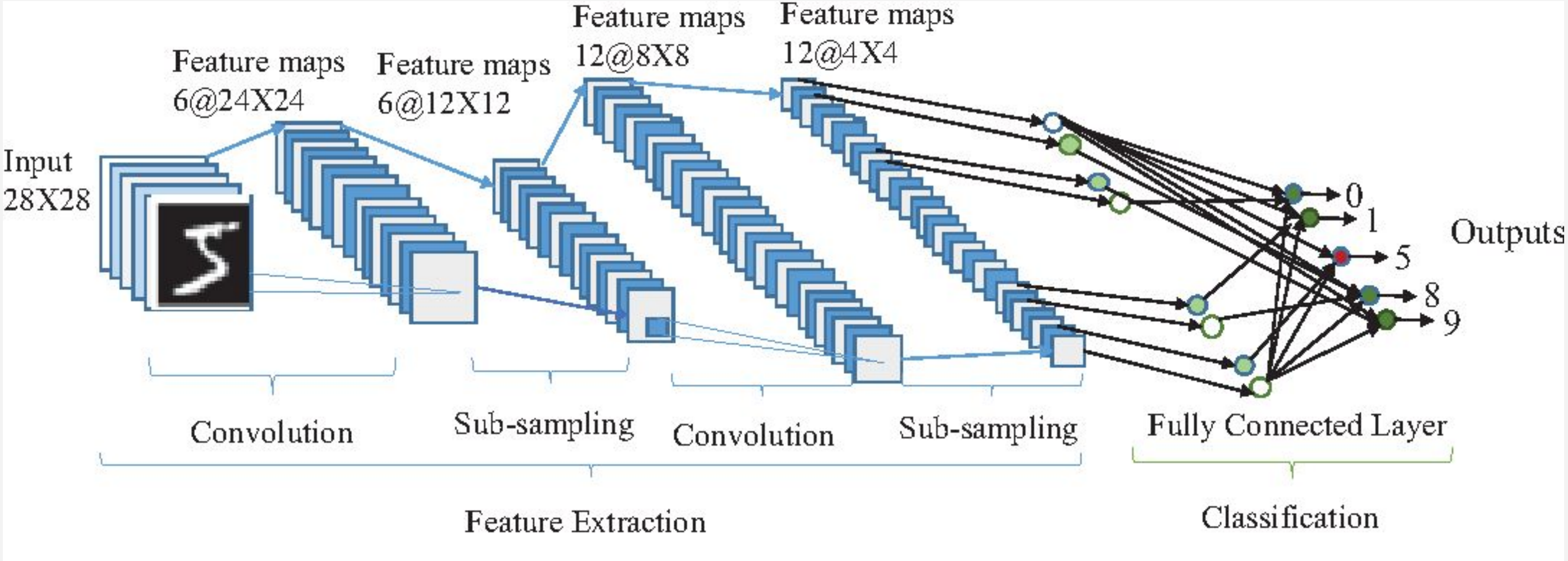
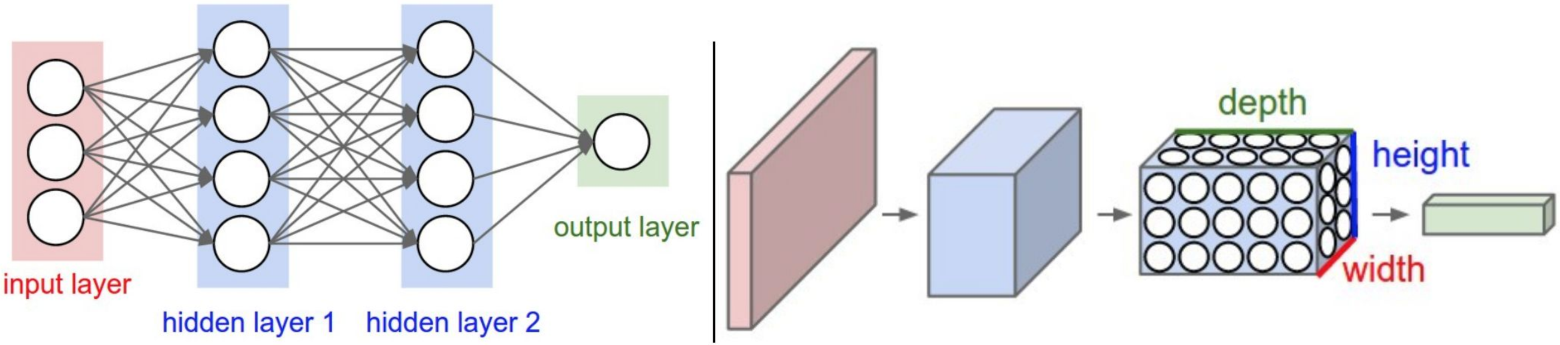
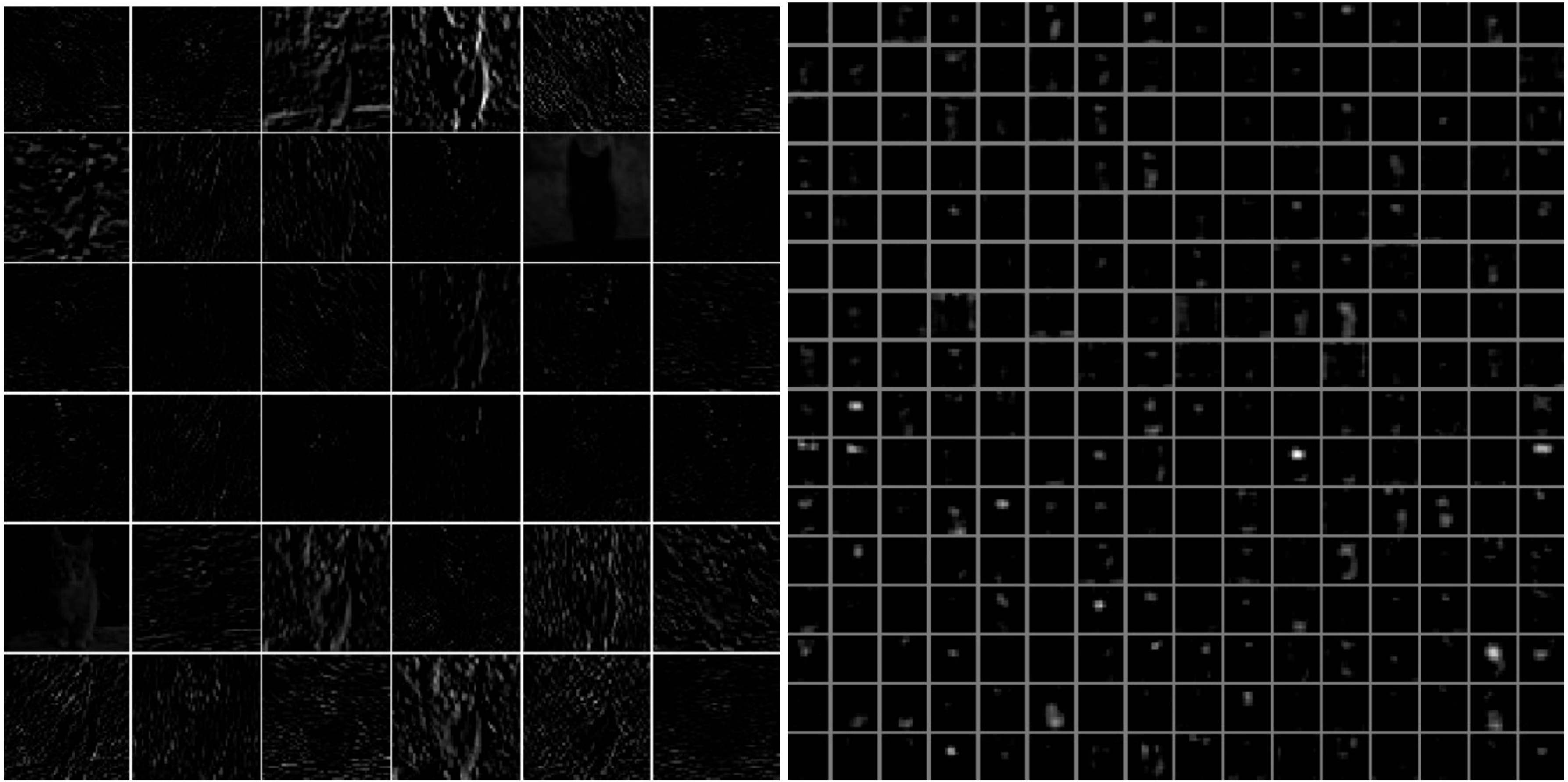


Fig. 1. CNN Block Diagram


input layer
hidden layer 1
hidden layer 2
output layer
depth
height
width


conv1 p1 n1 conv2 p2 n2 conv3 conv4 conv5 p5 fc6 fc7 fc8 prob
fwd conv5_151 | Back: deconv (from conv5_151, disp raw) | Boost: 0/1



Typical-looking activations on the first CONV layer (left), and the 5th CONV layer (right) of a trained AlexNet looking at a picture of a cat. Every box shows an activation map corresponding to some filter. Notice that the activations are sparse (most values are zero, in this visualization shown in black) and mostly local.