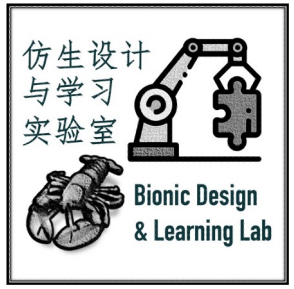
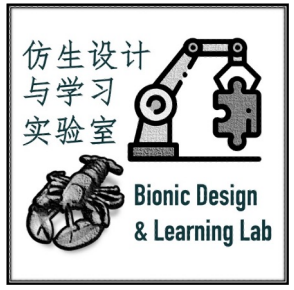
Lecture 07
Deep Networks II



AncoraSIR.com

SUSTech
Southern University
of Science and Technology

Convolutional Networks



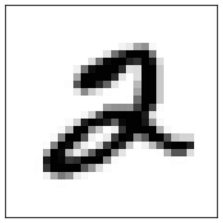
AncoraSIR.com

SUSTech
Southern University
of Science and Technology

A Design Challenge with Increasing Dimensions

Regular Neural Nets don't scale well to full images

$512 \times 512 \times 3 = 765,432$

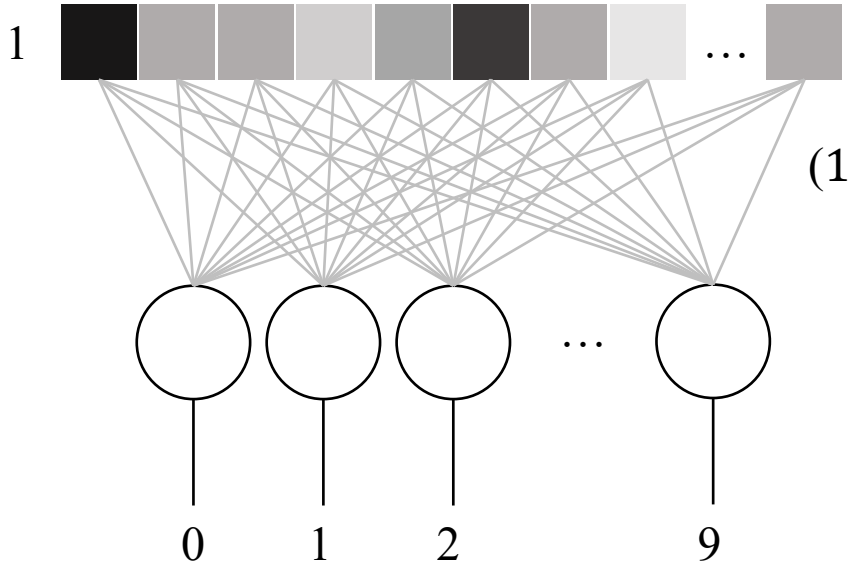
Gray-scaled  MNIST

CIFAR-10  Color (RGB)

$28 \times 28 \times 1 = 784$

Input *Features*

$32 \times 32 \times 3 = 3,072$



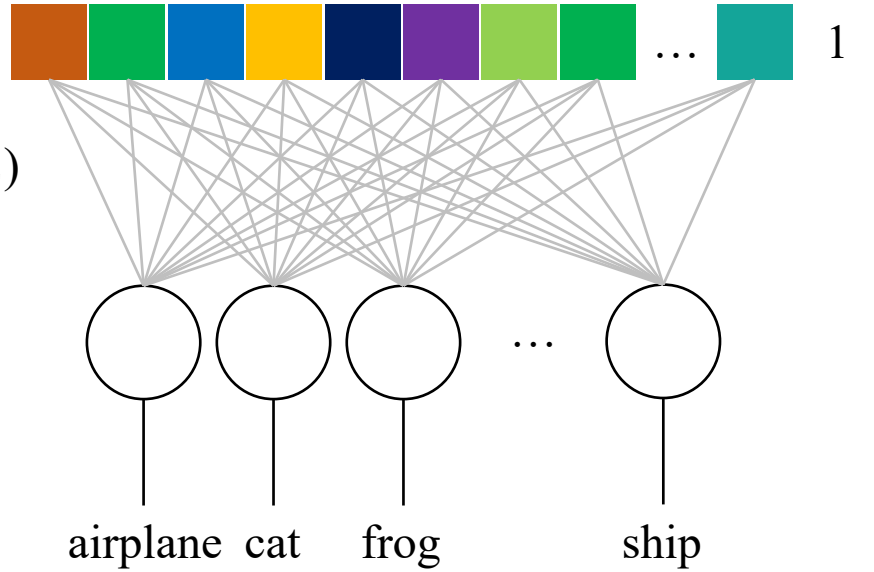
Weighted-Sum

$(10 \times 784 \leftarrow W \rightarrow 10 \times 3072)$

Activation

(?)

Output *Label*

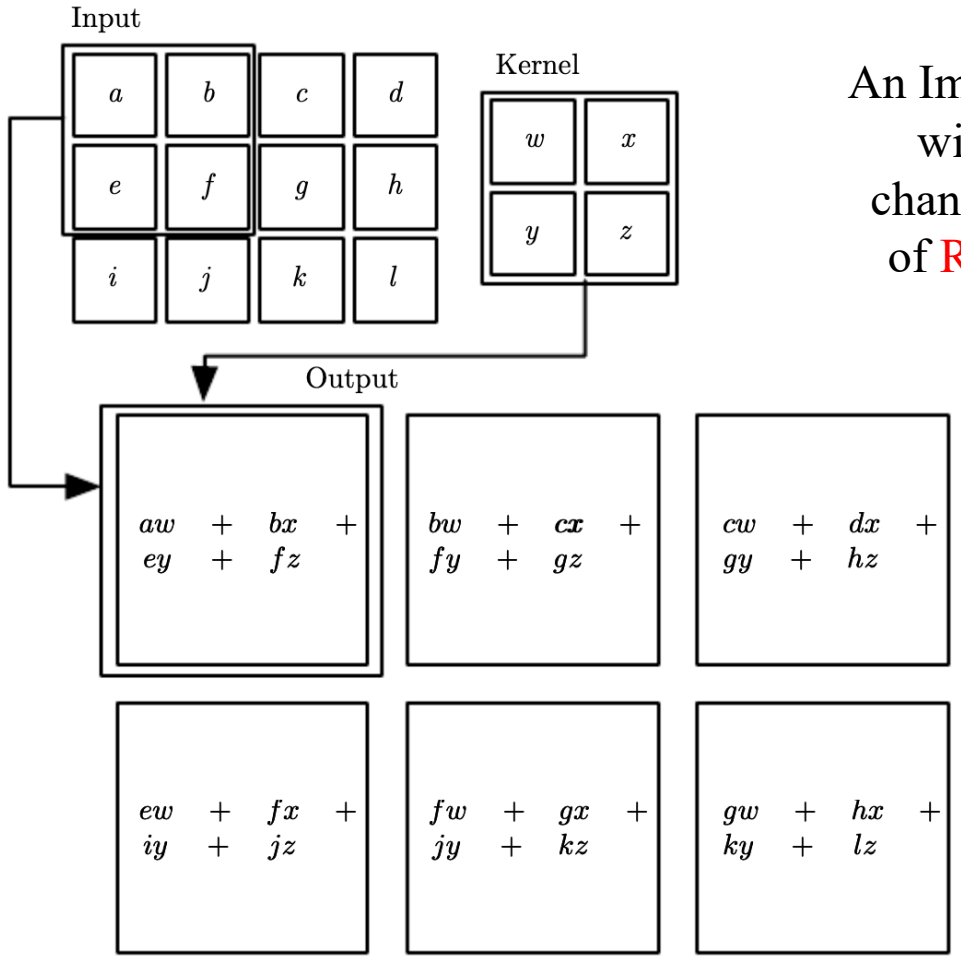


AncoraSIR.com

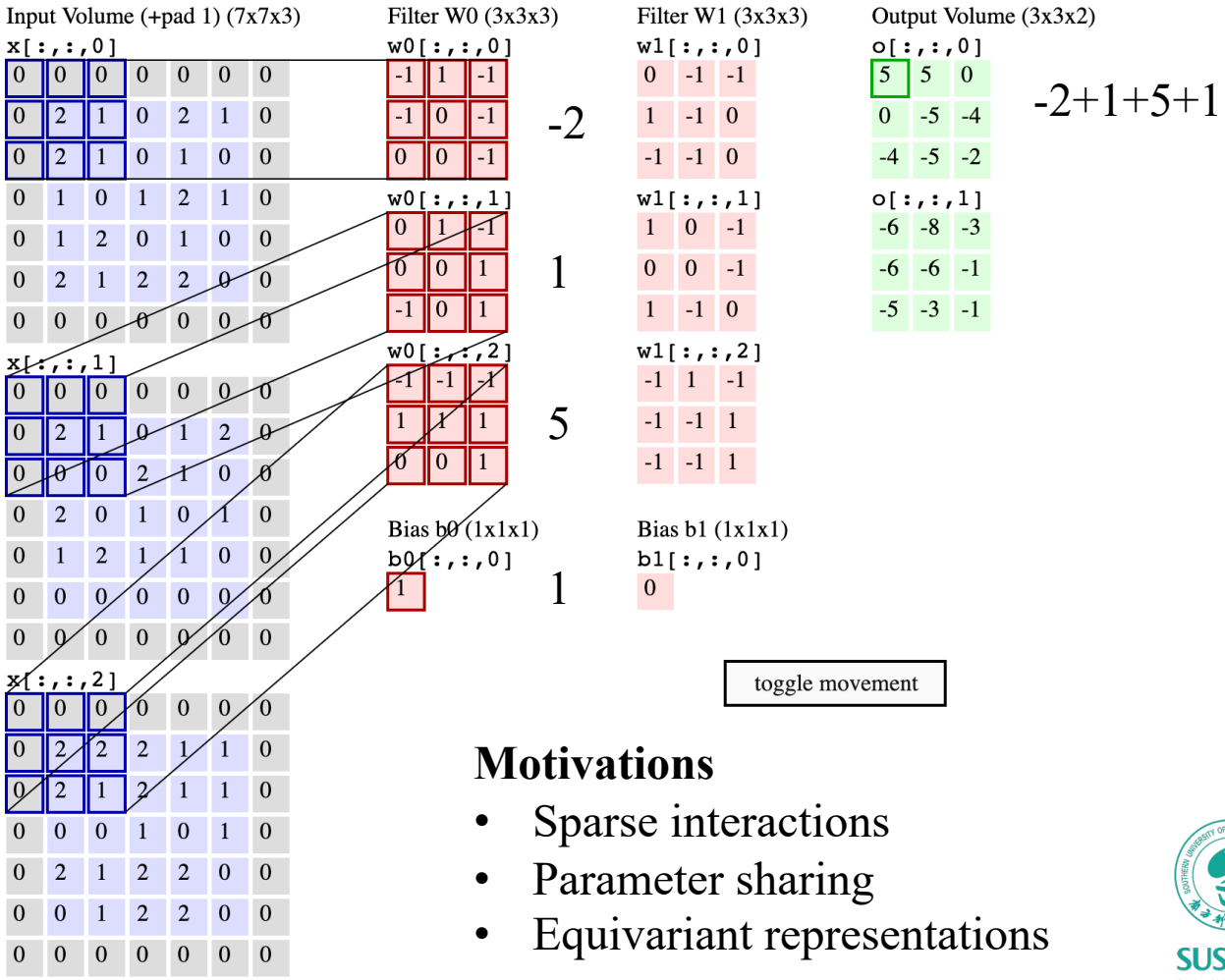
SUSTech Southern University of Science and Technology

Convolutional Operation

$$s(t) = \int x(a)w(t-a)da = (x * w)(t)$$



An Image with 3 channels of RGB



Motivations

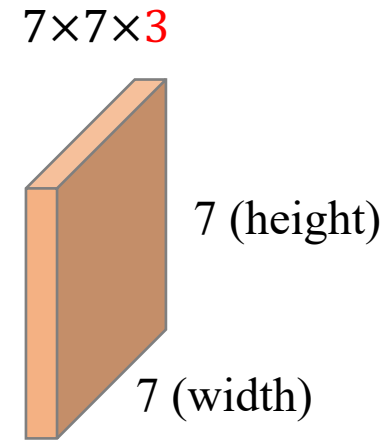
- Sparse interactions
- Parameter sharing
- Equivariant representations

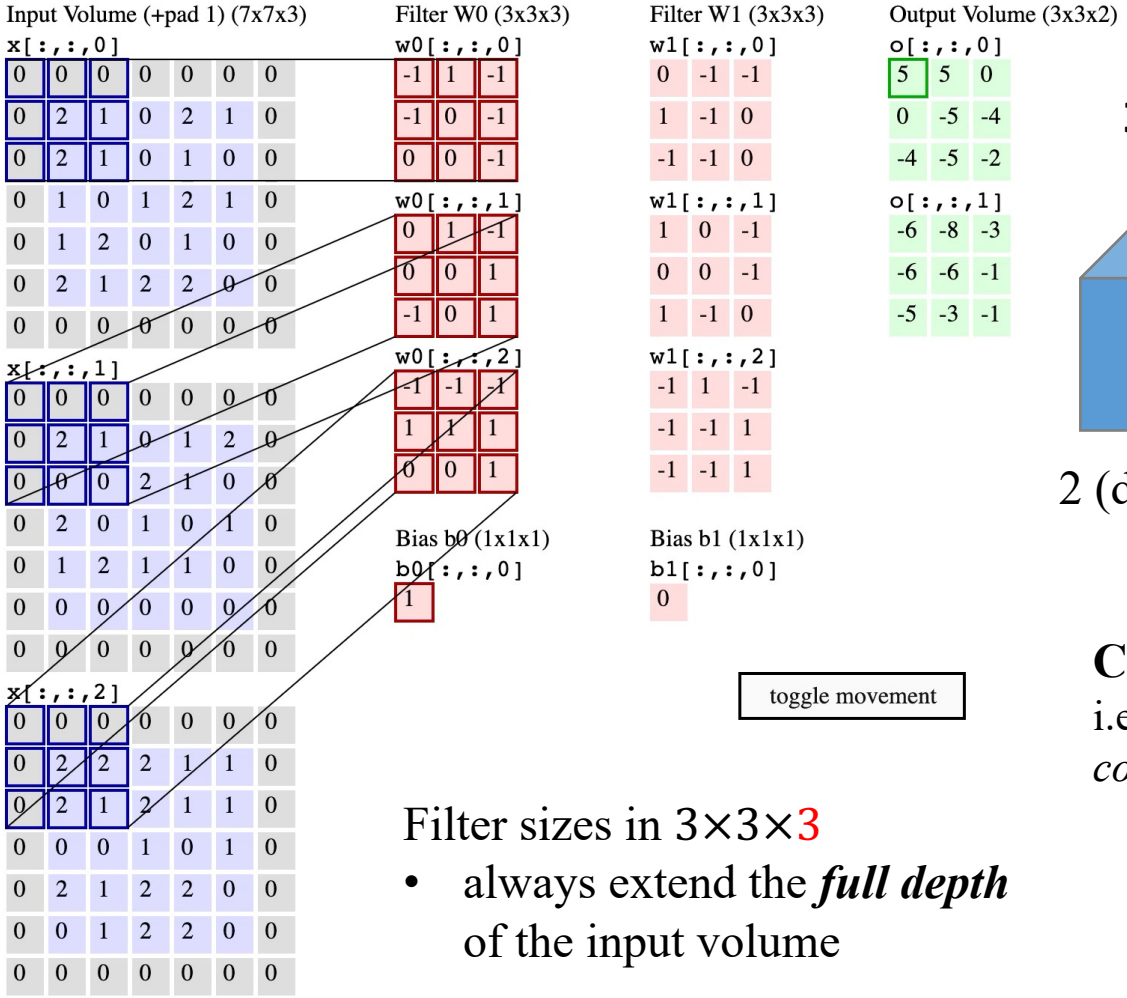
AncoraSIR.com

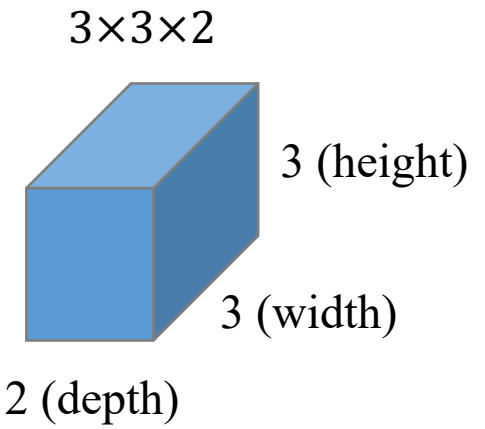
Convolution in 3D Volumes

Preserved spatial structure between the input and output volumes in width, height, number of channels

Input Volume in 3D







Output Volume in 3D

Layers in a ConvNet: Transform an input 3D volume to an output 3D volume with some differentiable function that may or may not have parameters.

Convolve the filter with the image i.e. "*slide over the image spatially, computing dot products*"

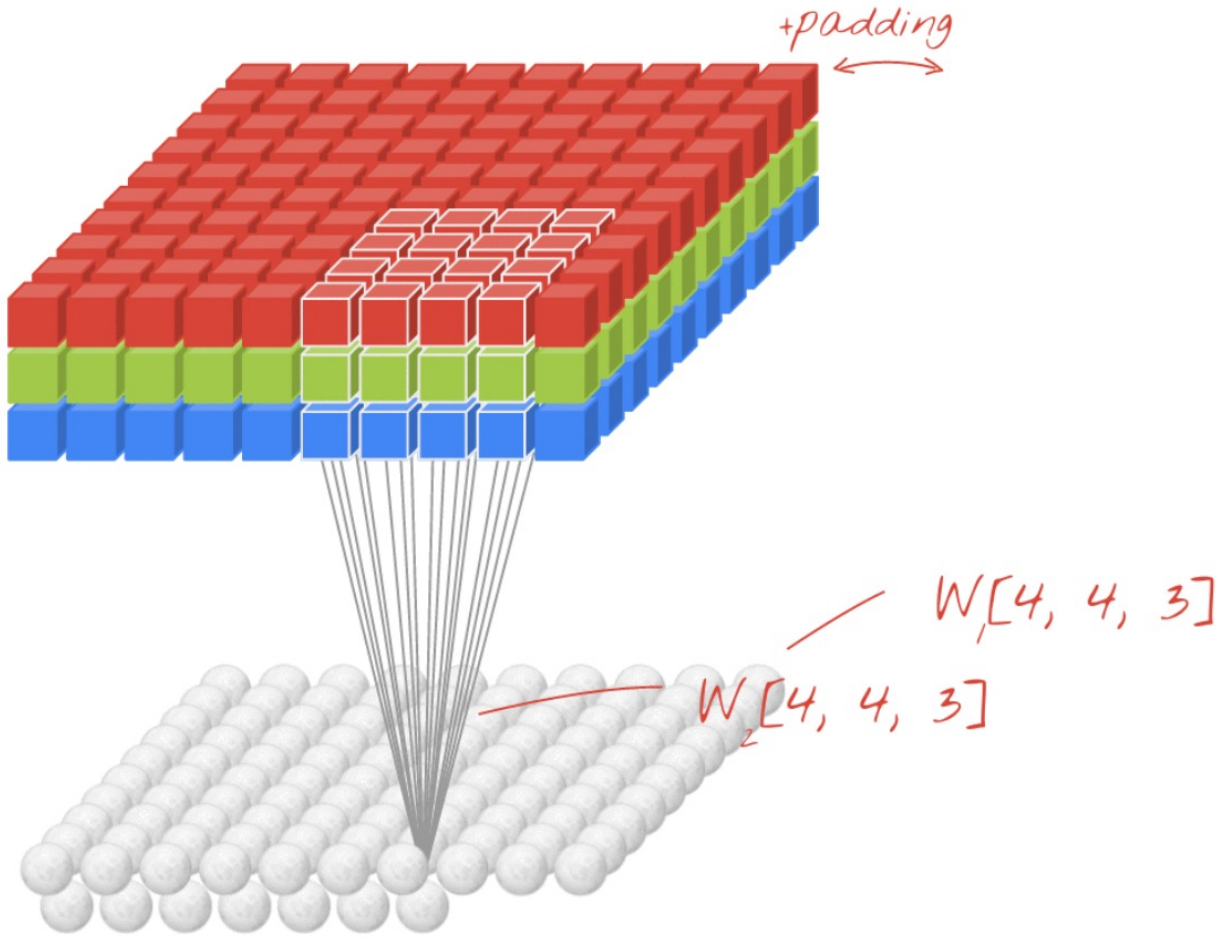
Filter sizes in 3×3×3

- always extend the ***full depth*** of the input volume

AncoraSIR.com

The Design of a Convolutional Layer

Defined by the filter (or kernel) size, the number of filters applied and the stride



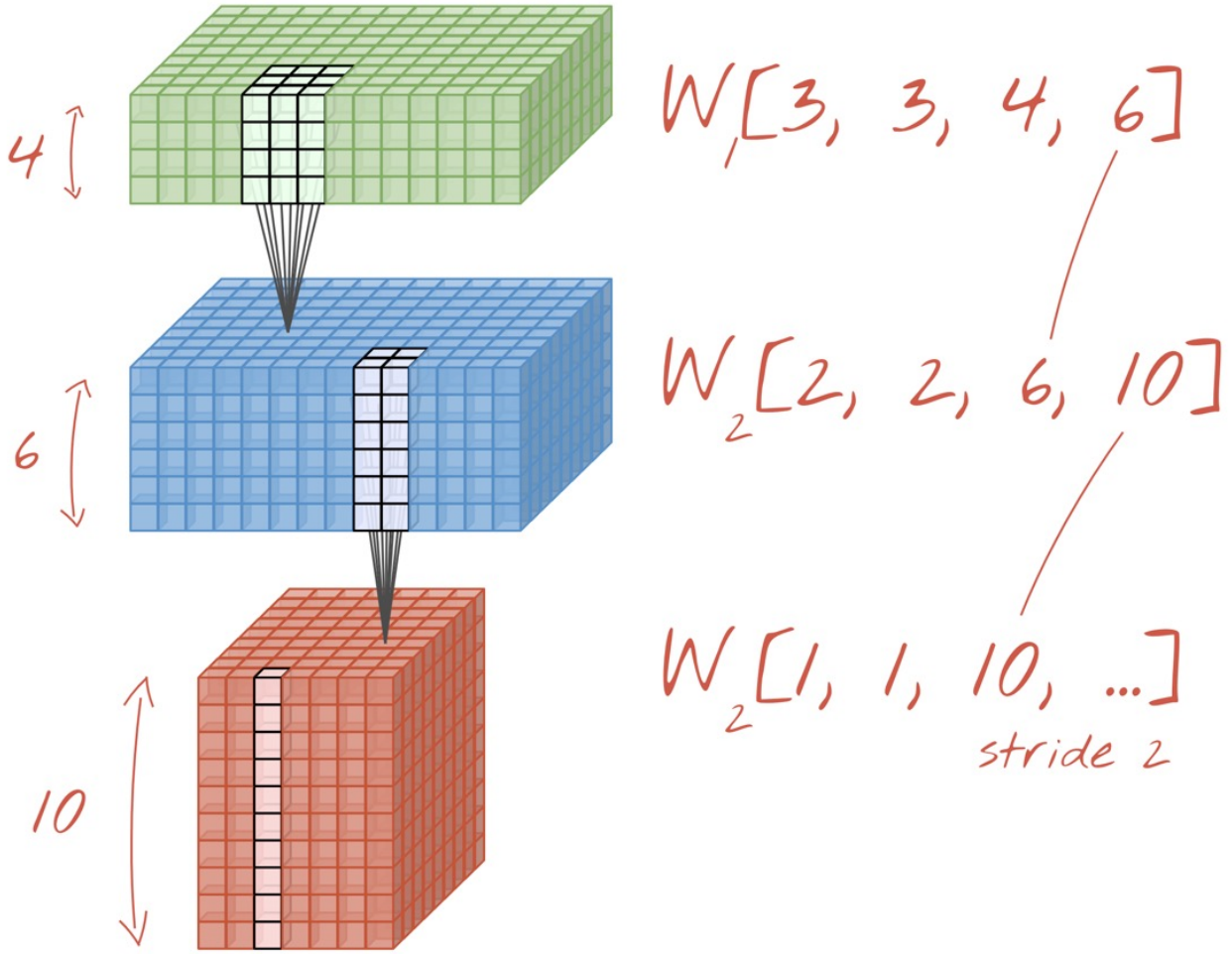
$W_1[4, 4, 3]$
$W_2[4, 4, 3]$
$W[4, 4, 3, 2]$

filter size — input channels — output channels

AncoraSIR.com

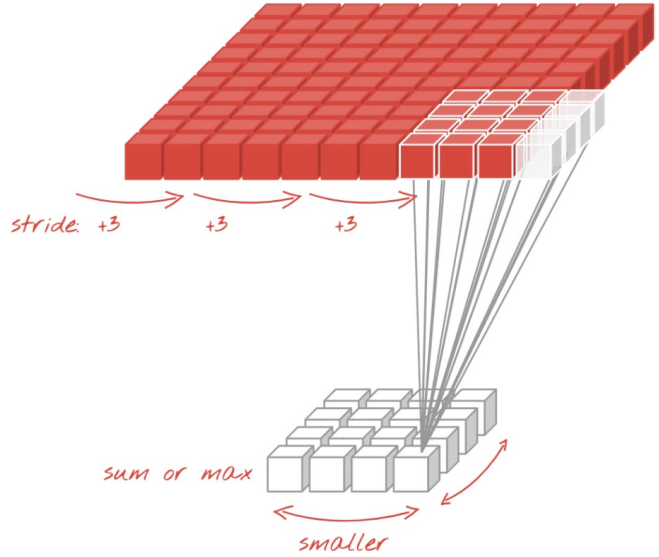
Output Volume Size

Defined by the filter (or kernel) size, the number of filters applied and the stride



AncoraSIR.com

- Depth (number of channels):
  - *adjusted by using more or fewer filters*
- Width & Height:
  - *adjusted by using a stride >1*
  - *(or with a max-pooling operation)*



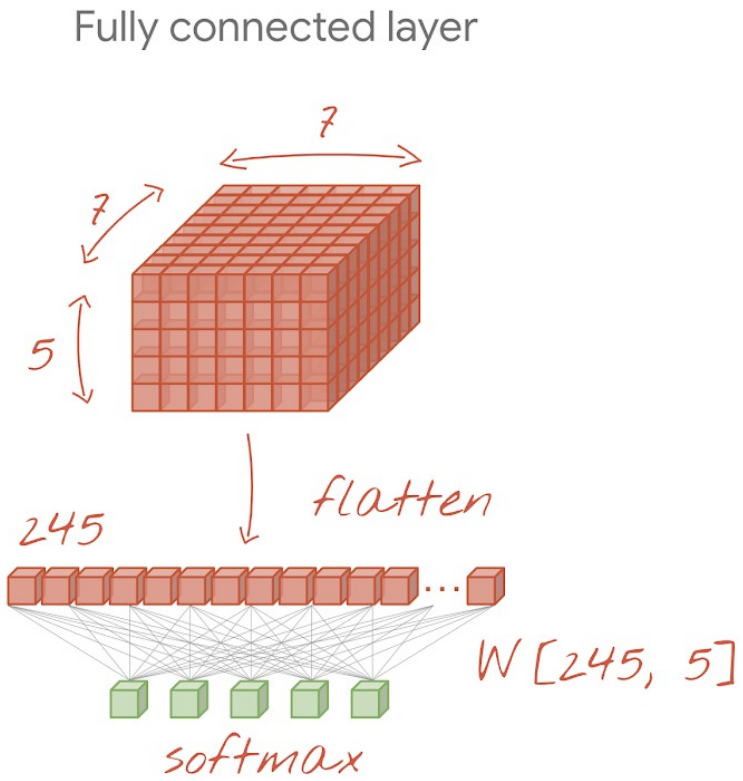
The Last Layer

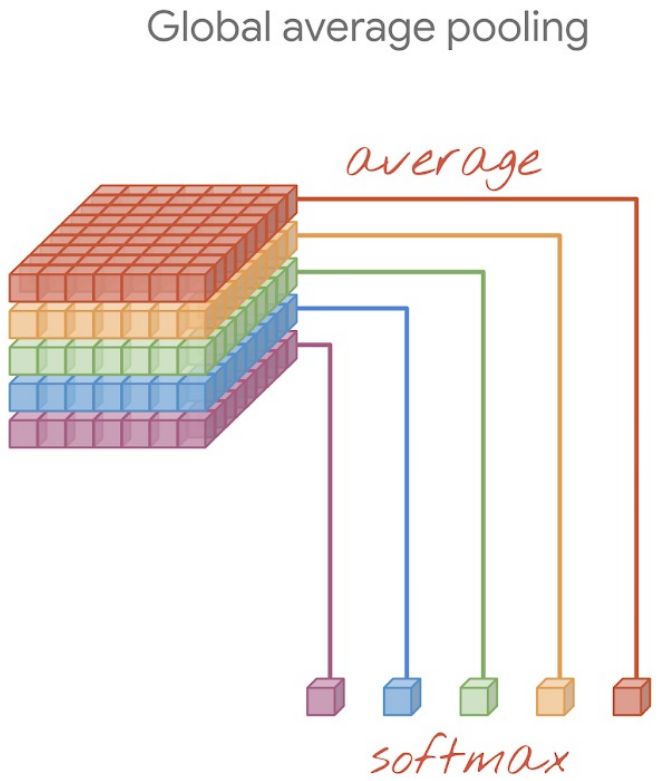
From a Cubic Volume in 3D to predicted labels

Similar like a normal neural network

Expensive in #weights

But preseves the location data (*x*, *y*)





Much lighter in calculation

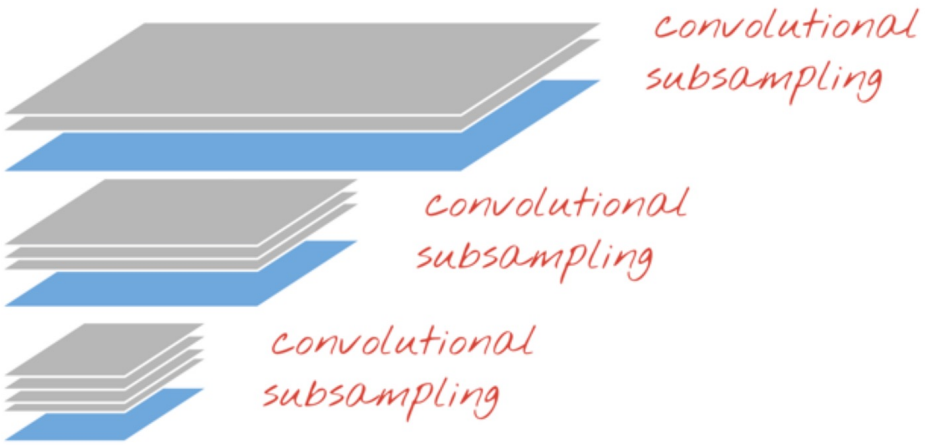
The average pooling explicitly discards all location data

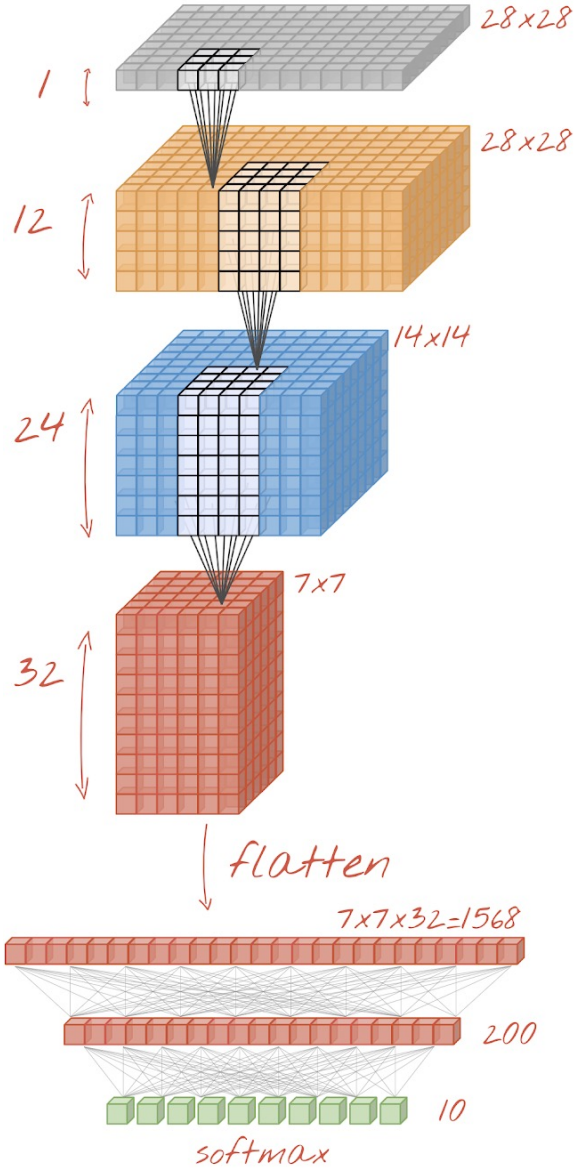
1225 weights cheaper **0** weights

AncoraSIR.com

Stacking Up a ConvNet

Layer-by-layer





Convolutional 3x3 filters=12

$W_1[3, 3, 1, 12]$

Convolutional 6x6 filters=24

$W_2[6, 6, 12, 24]$ stride 2

Convolutional 6x6 filters=32

$W_2[6, 6, 24, 32]$ stride 2

Dense layer

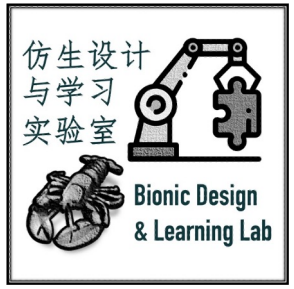
$W_4[1568, 200]$

Softmax dense layer

$W_5[200, 10]$

AncoraSIR.com

Layers in ConvNets

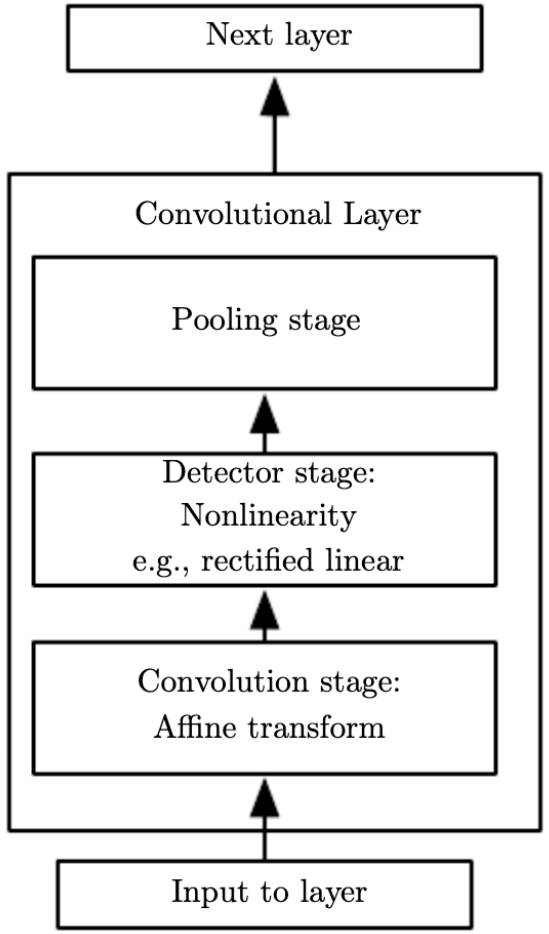


AncoraSIR.com

SUSTech
Southern University
of Science and Technology

The Three Stages of a Typical ConvNet Layer

The Convolution, Detector and Pooling Stages



AncoraSIR.com

- The maximum output within a rectangular neighborhood (max-pooling)
- The average of a rectangular neighborhood
- The L2 norm of a rectangular neighborhood
- A weighted average based on the distance from the central pixel

Replace the output of the net at a certain location with a summary statistic of the nearby outputs (can be viewed as a further abstraction of the learned features)

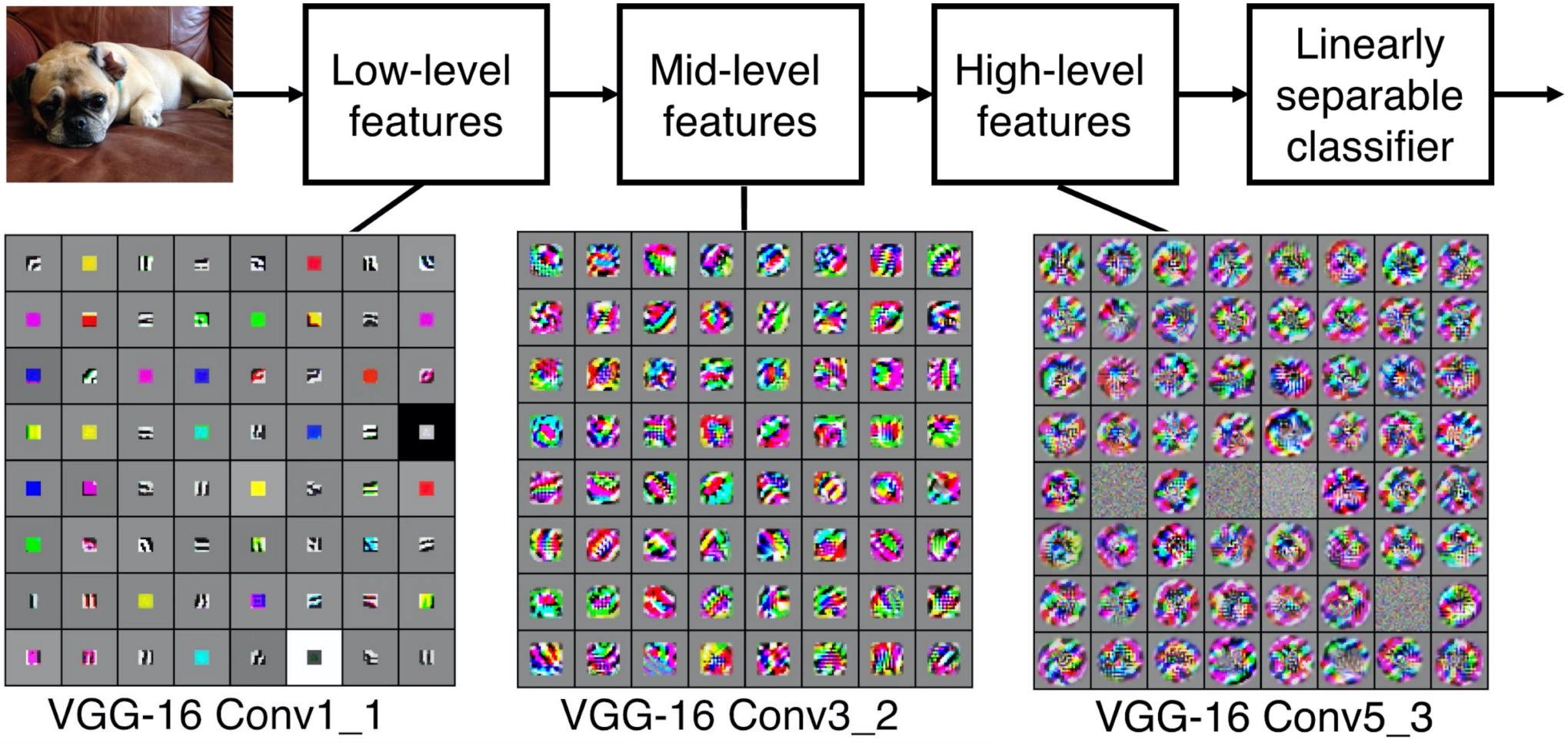
Each linear activation is run through a nonlinear activation function, such as ReLU (can be viewed as activation function)

Performs several convolutions in parallel to produce a set of linear activations (can be viewed as weighted-sum)

SUSTech
Southern University of Science and Technology

A Visualized Understanding of ConvNet

Multi-layered abstraction of 3D features towards a linerly separable classification



AncoraSIR.com

https://arxiv.org/pdf/1311.2901.pdf

SUSTech Southern University of Science and Technology

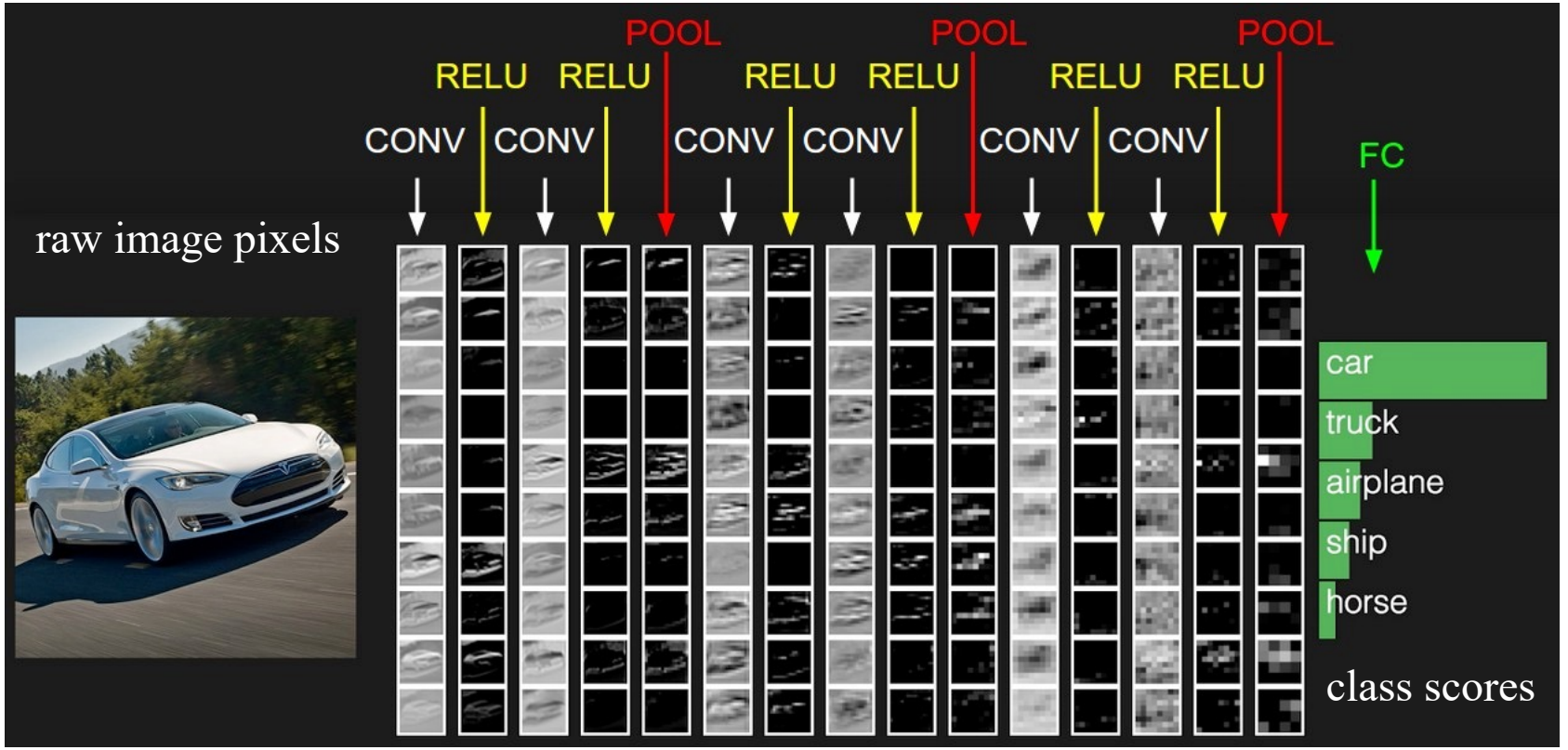
A Simple ConvNet for CIFAR-10 Classification

[INPUT - CONV - RELU - POOL - FC]

CONV layer compute the output of neurons that are connected to local regions in the input, i.e. [32x32x12] with 12 filters.

RELU layer will apply an elementwise activation function, such as the max(0,x) thresholding at zero. This leaves the size of the volume unchanged ([32x32x12]).

POOL layer will perform a downsampling operation along the spatial dimensions (width, height), resulting in volume such as [16x16x12].



INPUT layer [32x32x3] will hold the raw pixel values of the image

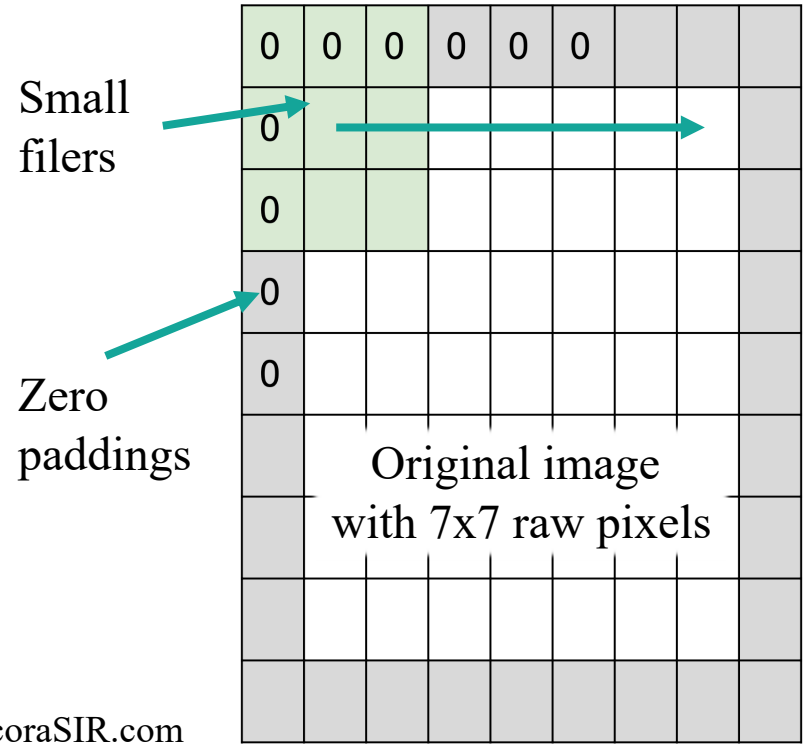
FC (i.e. fully-connected) layer will compute the class scores, resulting in volume of size [1x1x10], where each of the 10 numbers correspond to a class score

AncoraSIR.com

Convolutional Layer

Small filters that slide across the input volume

- Small-size filers
  - e.g. 3x3 or at most 5x5, using a stride of S=1,
  - Padding the input volume with zeros to avoid altering the spatial dimensions of the input.



INPUT features: 7x7

Filer size: 3x3

Stride: 1 (move step-by-step)

Padding: 1 pixel of 0 on all borders

OUTPUT features: 7x7

What if without paddings on the border?

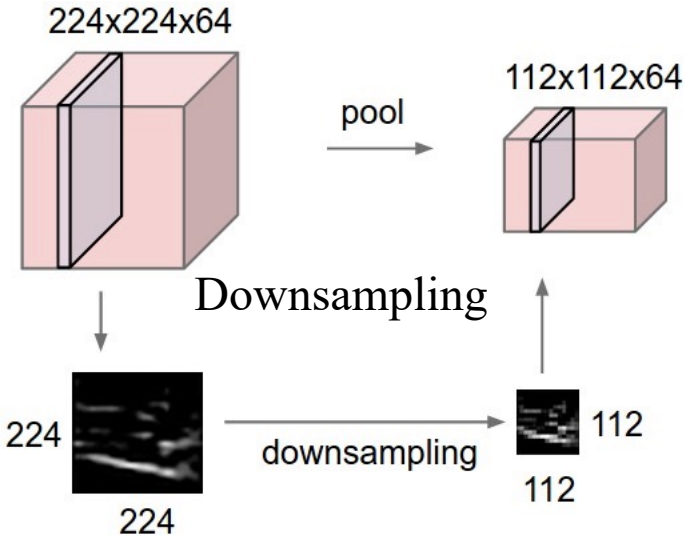
- *The spatial dimensions of the input will be changed, causing information loss on the border*

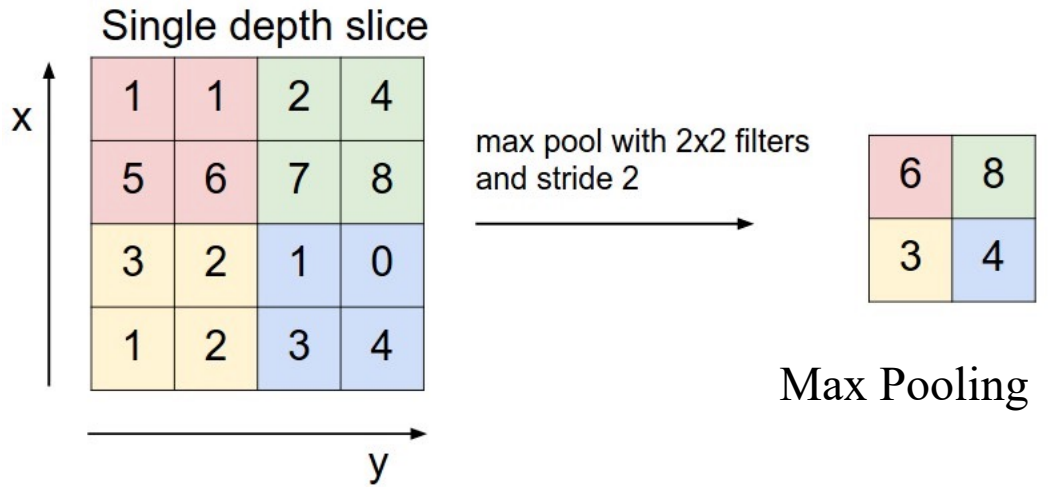
AncoraSIR.com

Pooling Layer

Downsampling the spatial dimensions of the input volume

- A network-wise regularization
  - Progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network, and hence to also control overfitting
  - Operates over each activation map independently
  - Usually, no need to zero padding (no convolutional operations)





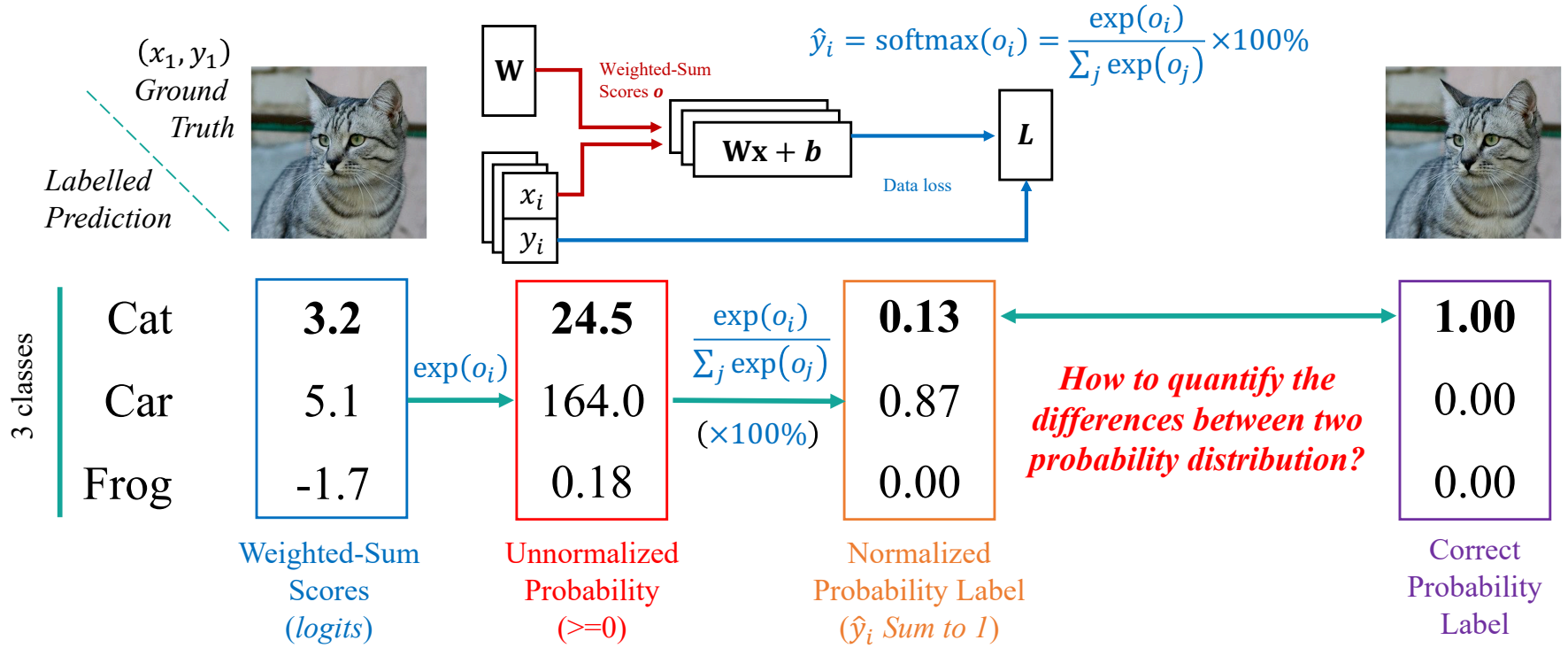
Max Pooling

AncoraSIR.com

Fully-Connected Layer

Full connections to all activations in the previous layer, as seen in regular Neural Networks

- Contains neurons that connect to the entire input volume
- Softmax is a common choice



AncoraSIR.com

ConvNet Architectures

Common choice of hyperparameters of ConvNet designs

- $\text{INPUT} \rightarrow [[\text{CONV} \rightarrow \text{RELU}] * \text{N} \rightarrow \text{POOL?}] * \text{M} \rightarrow [\text{FC} - \text{RELU}] * \text{K} \rightarrow \text{FC}$
  - the * indicates repetition,
  - the POOL? indicates an optional pooling layer.
  - N >= 0 (and usually N <= 3), M >= 0, K >= 0 (and usually K < 3)

- **INPUT** (that contains the image) should be divisible by 2 many times
  - 32 (e.g. CIFAR-10), 64, 96 (e.g. STL-10), or 224 (e.g. ImageNet), 384, and 512
- **CONV** should be using small filters using a stride of S=1
  - 3x3 or at most 5x5 with zero padding of the input volume
- **POOL** downsamples the spatial dimensions of the input
  - Common setting is to use max-pooling with 2x2 receptive fields with a stride of 2

AncoraSIR.com

SUSTech Southern University of Science and Technology

Thank you~

songcy@sustech.edu.cn