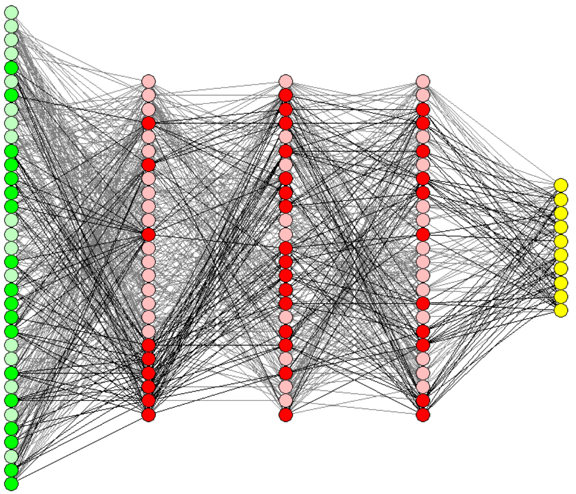


a soccer player is kicking a soccer ball

a street sign on a pole in front of a building

a couple of giraffe standing next to each other

Deep Learning

Dr. Shahid Mahmood Awan

shahid.awan@umt.edu.pk

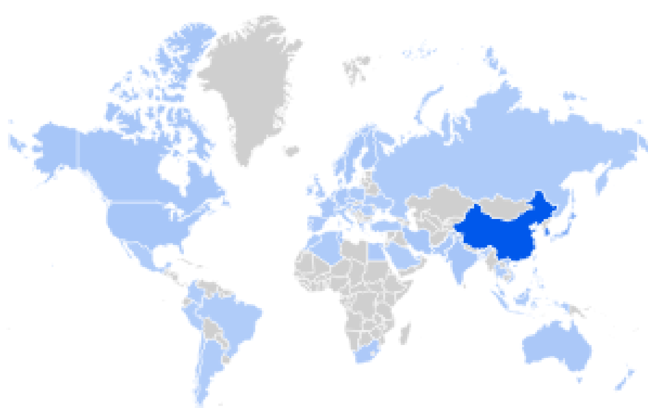
University of Management and Technology

Deep Learning attracts lots of attention.
Google Trends

Interest over time

Interest by region

Region



1	China	100
2	South Korea	64
3	Japan	23
4	Singapore	23
5	Taiwan	19

Outline

Part I: Introduction of Deep Learning

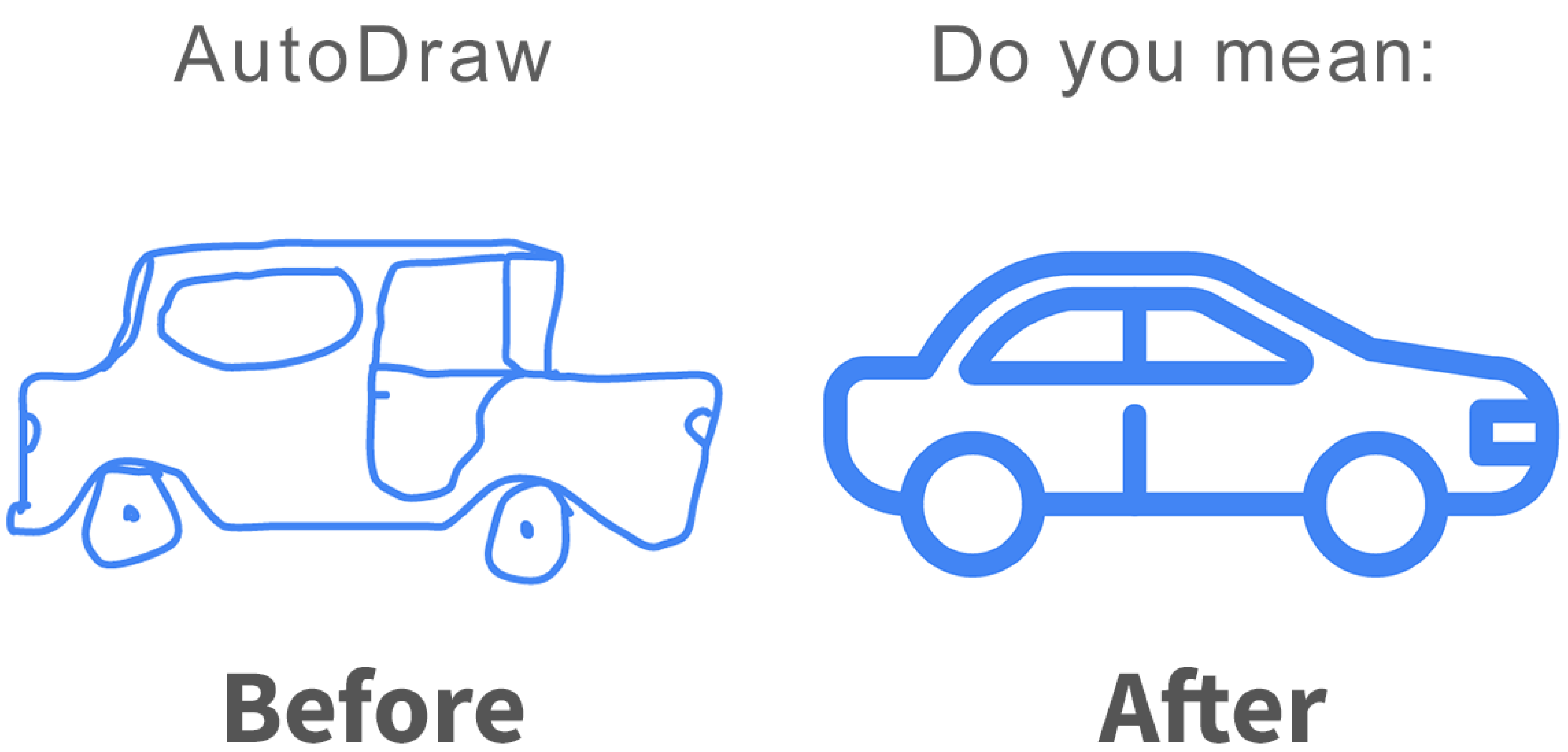
Part II: Why Deep Learning

Part III: Neural Networks and Deep Neural Networks

Part IV: Convolutional Neural Network (CNN)

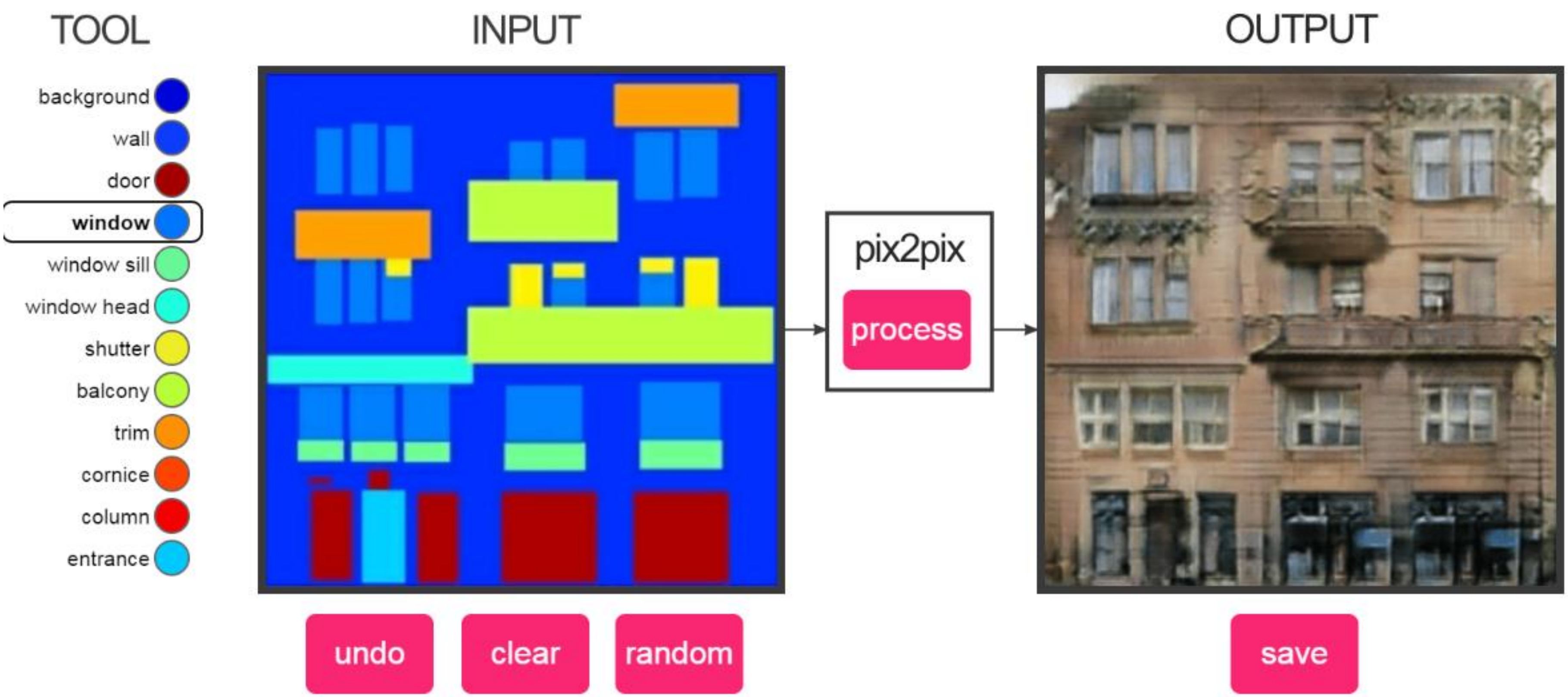
Google Auto Draw

https://aiexperiments.withgoogle.com/autodraw



Pix2Pix

https://affinelayer.com/pixsrv/



Object Classification and Detection in Photographs

Automatic Image Caption Generation

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"girl in pink dress is jumping in air."

"black and white dog jumps over bar."

"young girl in pink shirt is swinging on swing."

Automatic Handwriting Generation

http://www.cs.toronto.edu/~graves/handwriting.html

Machine Learning Mastery

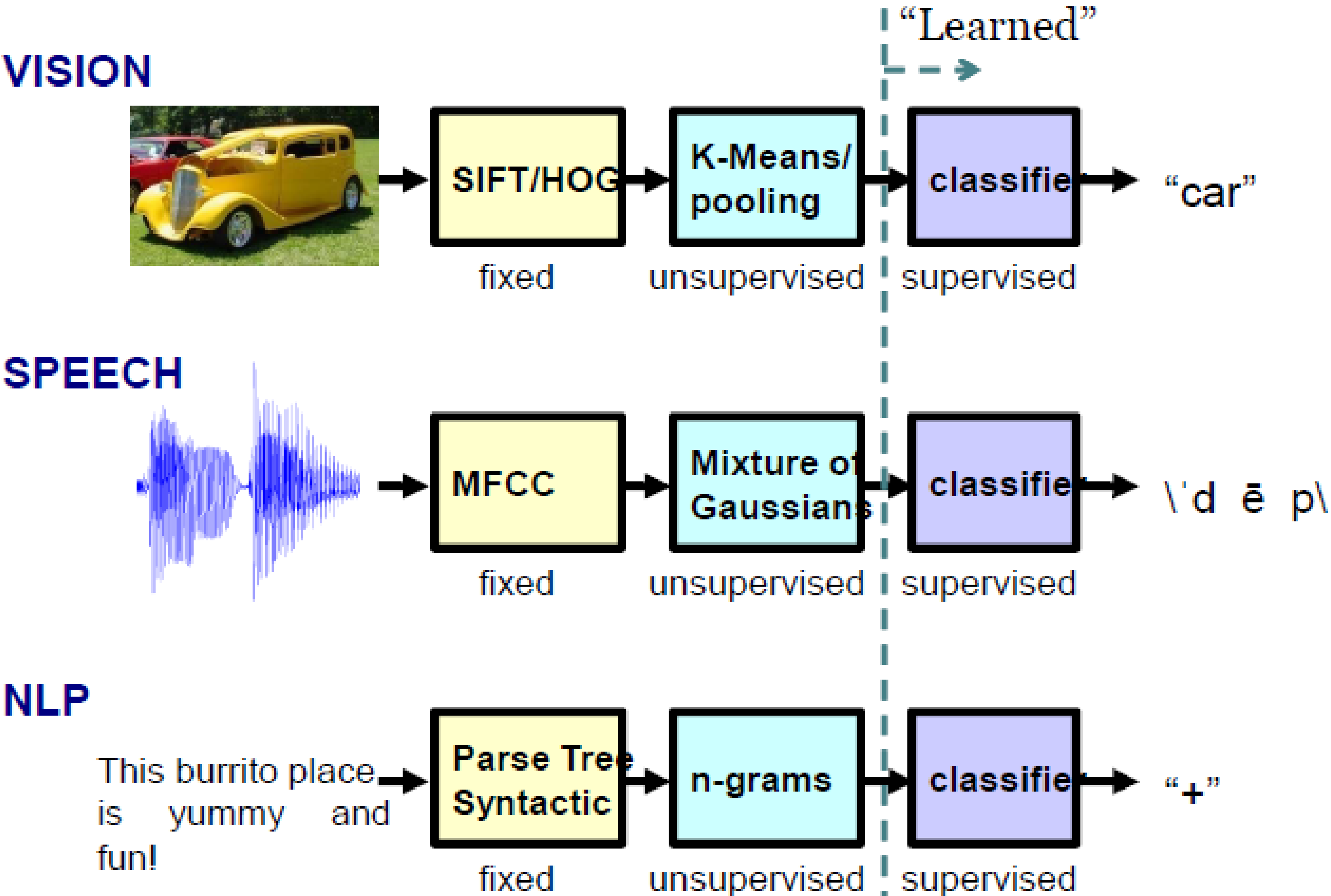
Machine Learning Mastery

Machine Learning Mastery

Start Playing with Deep learning

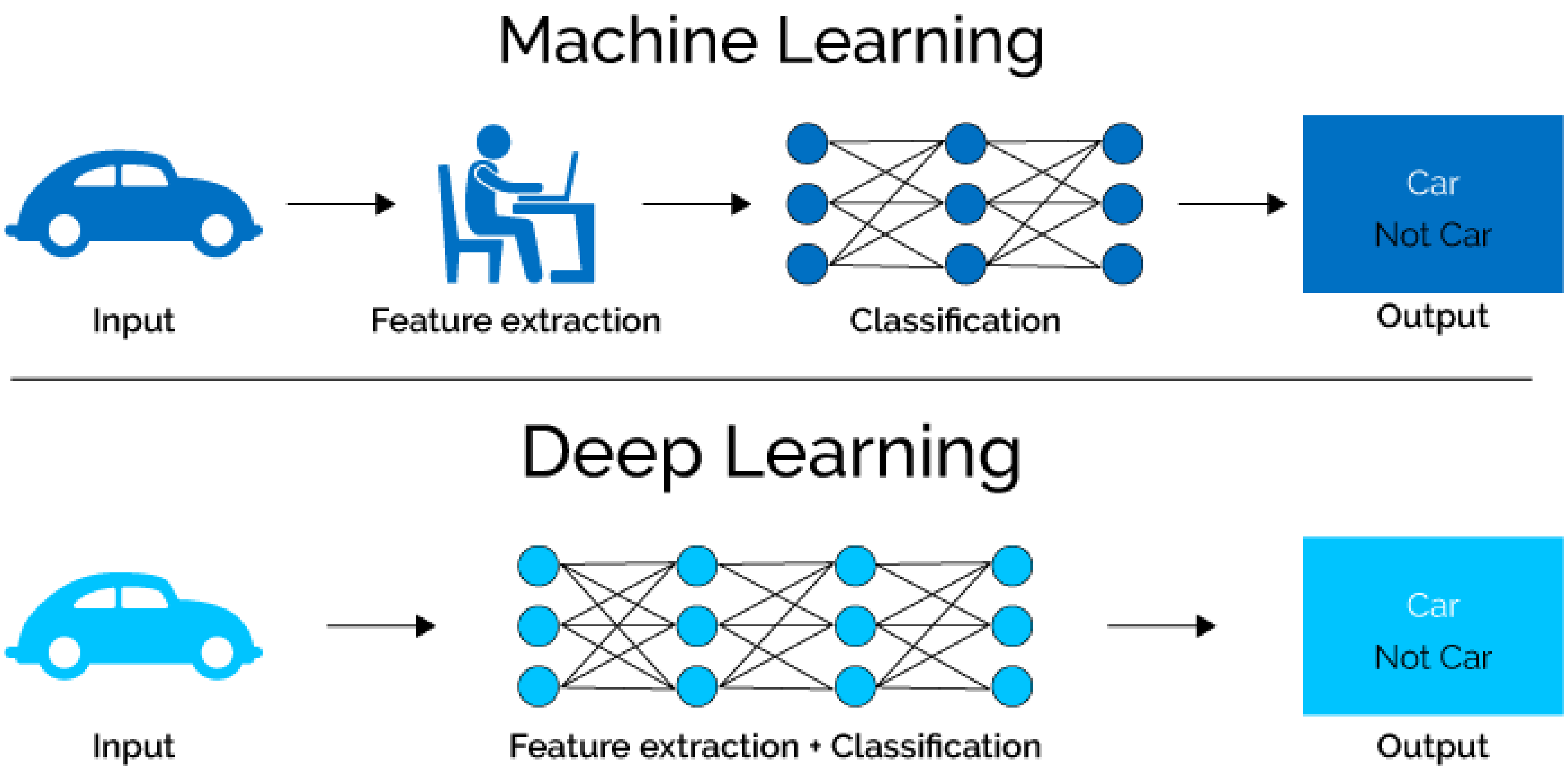
- **http://deeplearning.net/demos/**
- **https://experiments.withgoogle.com/ai**
- **http://playground.tensorflow.org/**

Traditional Machine Learning (more accurately)



Slide Credit: Marc'Aurelio Ranzato, Yann LeCun

Deep Learning = End-to-End Learning



The Deep Learning Way

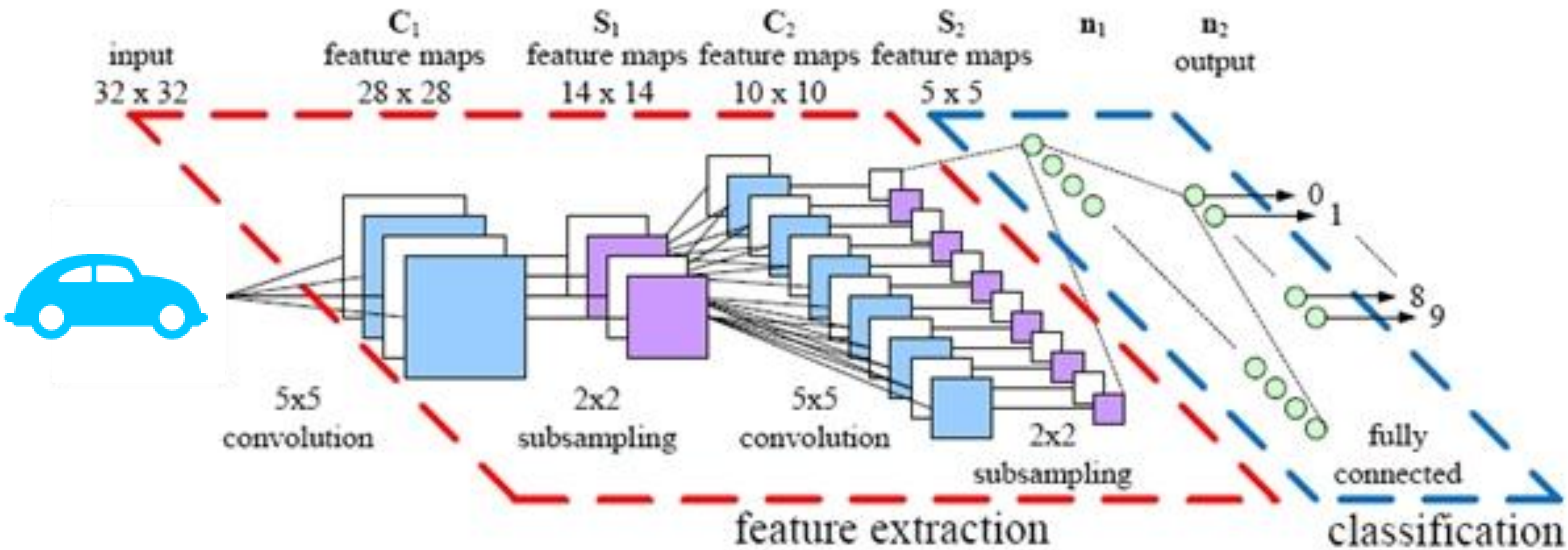


Image Source: https://www.kdnuggets.com/wp-content/uploads/cnn-architecture.jpg

So, 1. **what exactly is deep learning** ?

And, 2. **why is it generally better** than other methods on image, speech and certain other types of data?

The short answers

1. **'Deep Learning' means** using a **neural network** with **several layers of nodes** between input and output

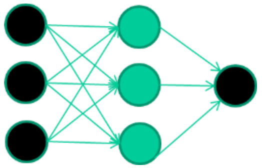
2. **the series of layers between input & output do feature identification and processing in a series of stages, just as our brains seem to.**

hmmm… OK, but:

3. **multilayer neural networks have been around for 25 years. What's actually new?**

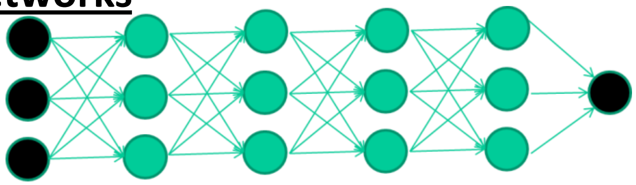
- More data
- Better Learning Algorithms
- More Computing Power

we have always had good algorithms for learning the weights in networks with 1 hidden layer



but these algorithms are not good at learning the weights for networks with more hidden layers

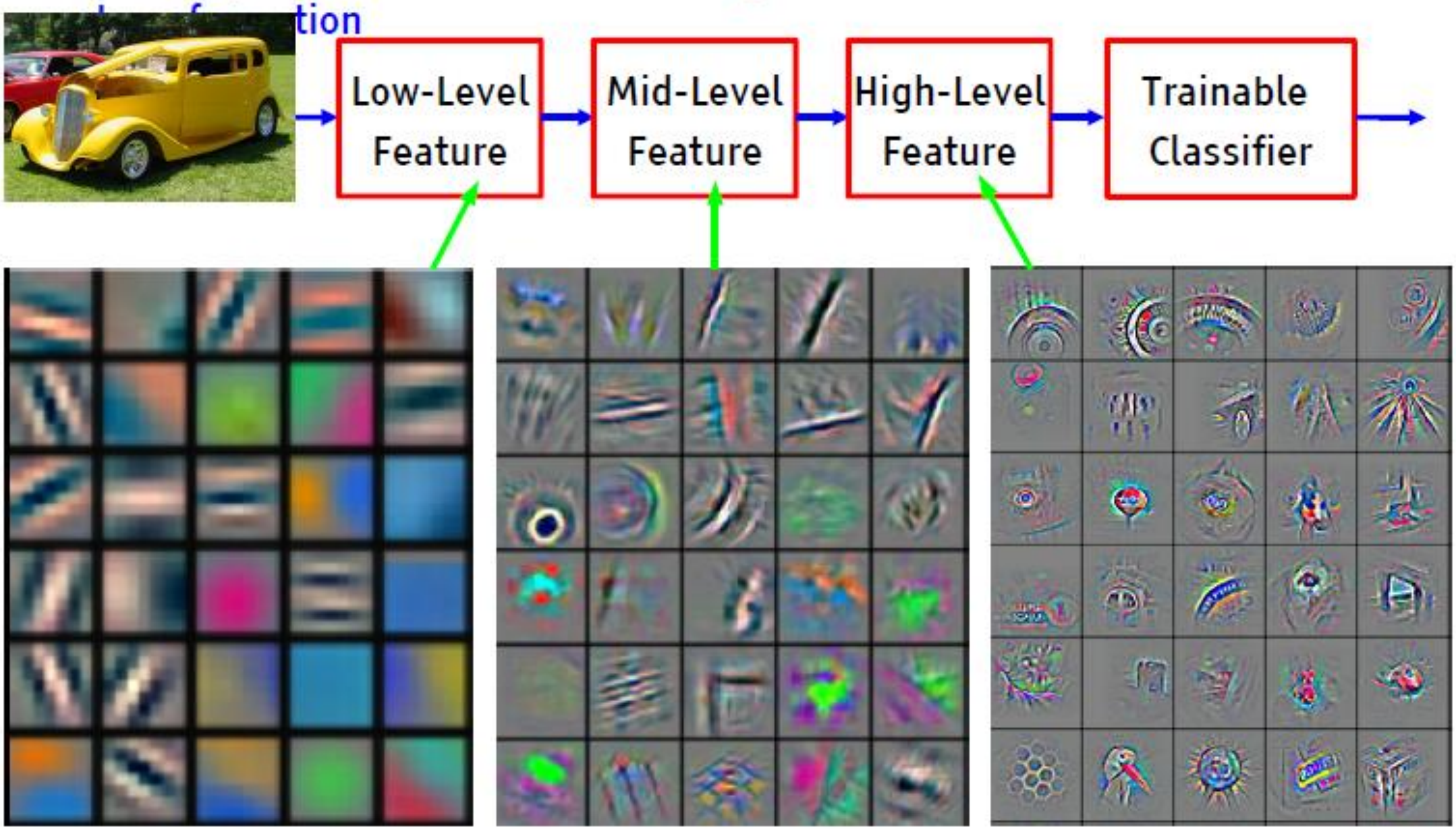
what's new is: **<u>algorithms for training many-later networks</u>**



Deep Learning = Learning Hierarchical Representations

Y LeCun
MA Ranzato

- It's deep if it has more than one stage of non-linear feature transformation

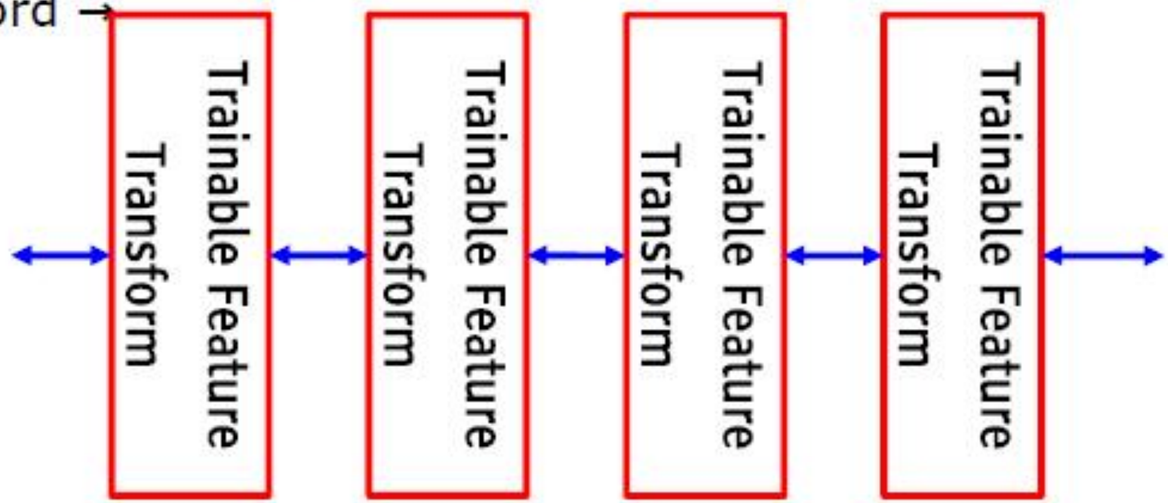


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Trainable Feature Hierarchy

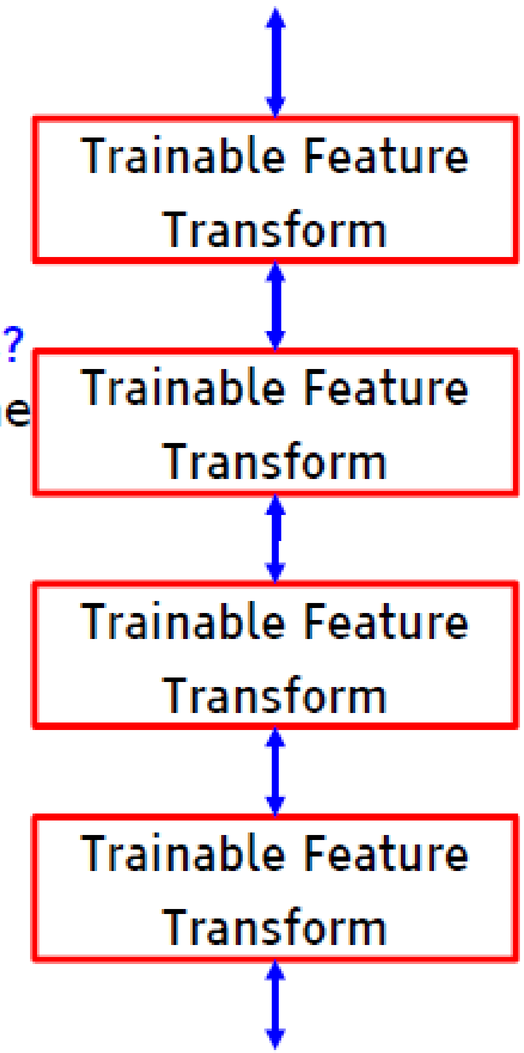
Y LeCun
MA Ranzato

- Hierarchy of representations with increasing level of abstraction
- Each stage is a kind of trainable feature transform
- Image recognition
  - Pixel → edge → texton → motif → part → object
- Text
  - Character → word → word group → clause → sentence → story
- Speech
  - Sample → spectral band → sound → … → phone → phoneme → word →



Learning Representations: a challenge for ML, CV, AI, Neuroscience, Cognitive Science...

- How do we learn representations of the perceptual world?
  - How can a perceptual system build itself by looking at the world?
  - How much prior structure is necessary
- ML/AI: how do we learn features or feature hierarchies?
  - What is the fundamental principle? What is the learning algorithm? What is the architecture?
- Neuroscience: how does the cortex learn perception?
  - Does the cortex "run" a single, general learning algorithm? (or a small number of them)
- CogSci: how does the mind learn abstract concepts on top of less abstract ones?

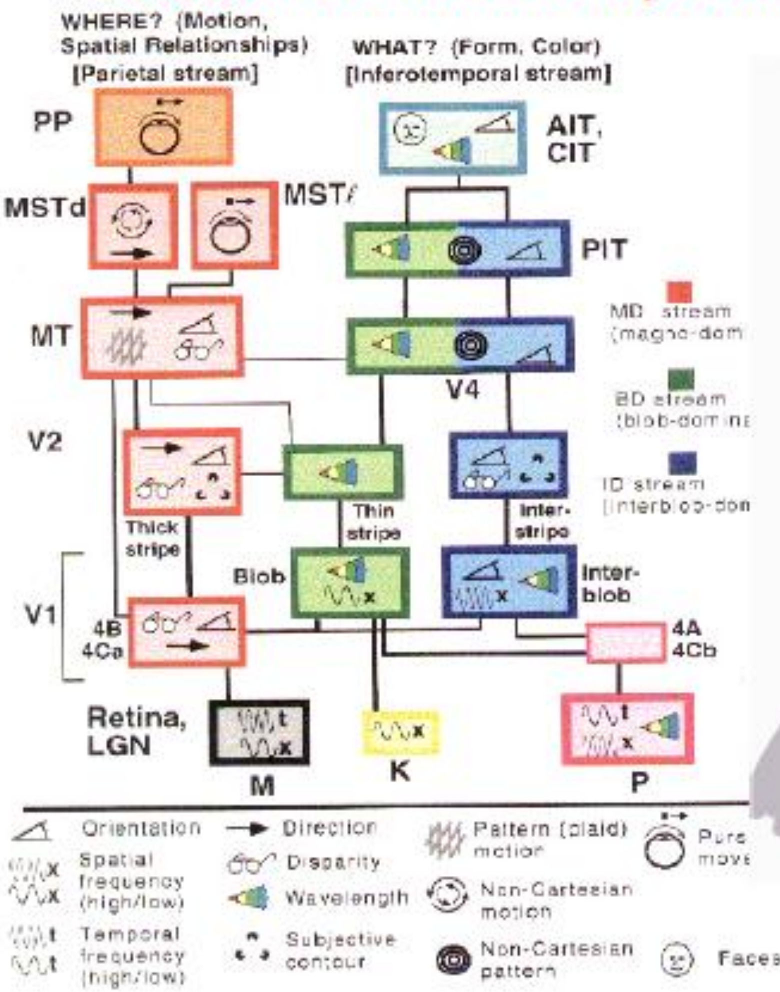


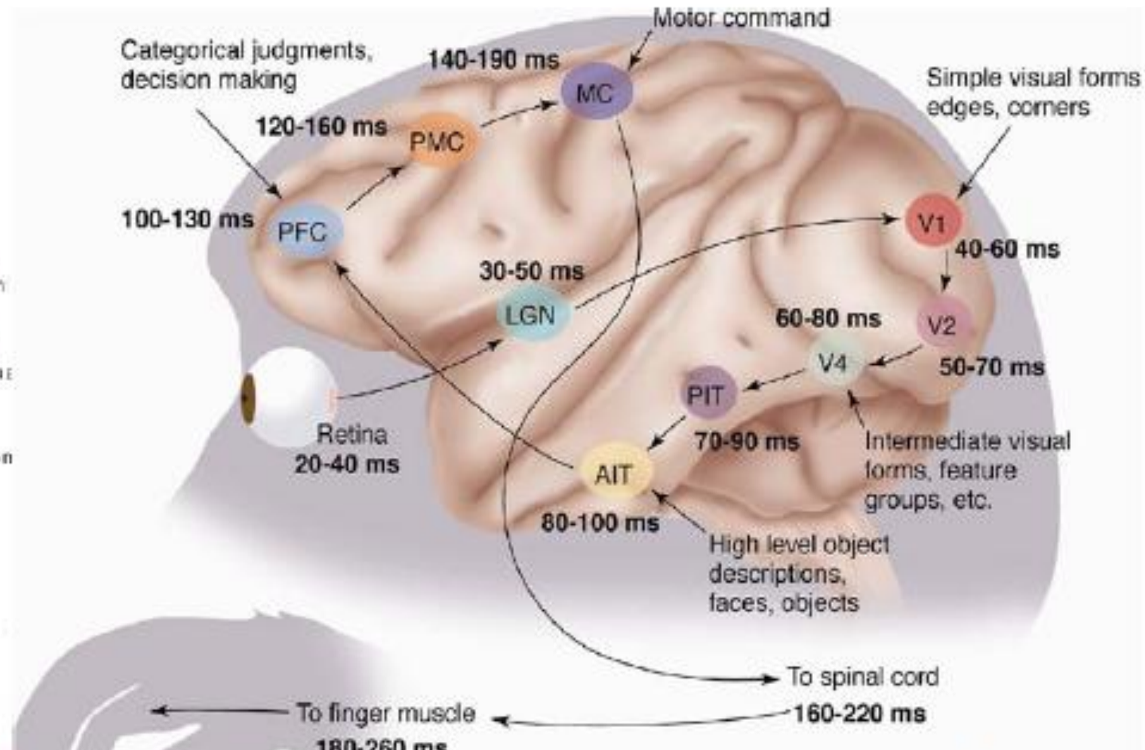
- Deep Learning addresses the problem of learning hierarchical representations with a single algorithm

The Mammalian Visual Cortex is Hierarchical

Y LeCun
MA Ranzato

- The ventral (recognition) pathway in the visual cortex has multiple stages
- Retina - LGN - V1 - V2 - V4 - PIT - AIT ....
- Lots of intermediate representations





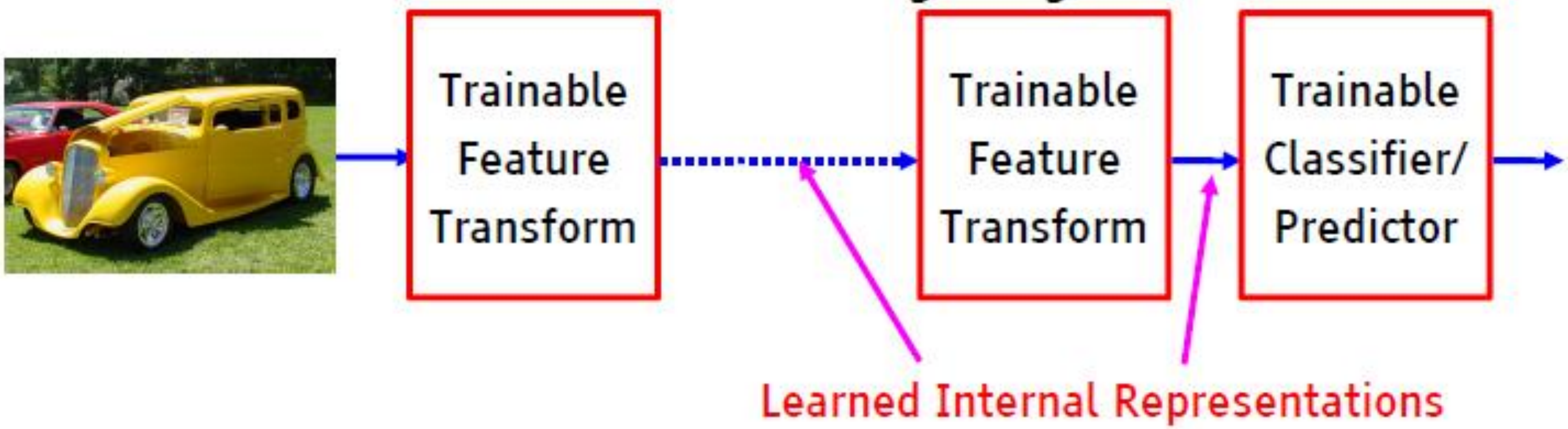
[picture from Simon Thorpe]

[Gallant & Van Essen]

Trainable Feature Hierarchies: End-to-end learning

Y LeCun
MA Ranzato

- **A hierarchy of trainable feature transforms**
  - Each module transforms its input representation into a higher-level one.
  - High-level features are more global and more invariant
  - Low-level features are shared among categories



- **How can we make all the modules trainable and get them to learn appropriate representations?**

Do we really need deep architectures?

Y LeCun
MA Ranzato

- Theoretician's dilemma: "We can approximate any function as close as we want with shallow architecture. Why would we need deep ones?"

$$y = \sum_{i=1}^{P} \alpha_i K(X, X^i) \qquad y = F(W^1.F(W^0.X))$$

  - kernel machines (and 2-layer neural nets) are "universal".
- Deep learning machines

$$y = F(W^K.F(W^{K-1}.F(....F(W^0.X)...)))$$

- Deep machines are more efficient for representing certain classes of functions, particularly those involved in visual recognition
  - they can represent more complex functions with less "hardware"
- We need an efficient parameterization of the class of functions that are useful for "AI" tasks (vision, audition, NLP...)

Deep Learning: A Theoretician's Paradise?

- **Deep Learning is about representing high-dimensional data**
  - There has to be interesting theoretical questions there
  - What is the geometry of natural signals?
  - Is there an equivalent of statistical learning theory for unsupervised learning?
  - What are good criteria on which to base unsupervised learning?
- **Deep Learning Systems are a form of latent variable factor graph**
  - Internal representations can be viewed as latent variables to be inferred, and deep belief networks are a particular type of latent variable models.
  - The most interesting deep belief nets have intractable loss functions: how do we get around that problem?
- **Lots of theory at the 2012 IPAM summer school on deep learning**
  - Wright's parallel SGD methods, Mallat's "scattering transform", Osher's "split Bregman" methods for sparse modeling, Morton's "algebraic geometry of DBN",....

Why Deep Learning

1. Its surprising performance on range of different problems.

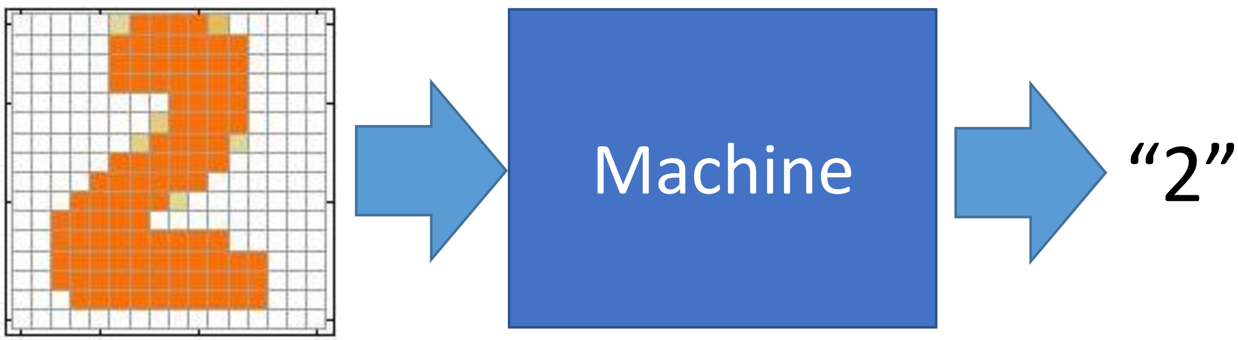
2. Ability to self-learn high level feature representations from raw input data.

3. Modularity (an extremely important property). You only need to understand few building lego blocks and you are ready to go.

4. Ability to build model using Transfer Learning

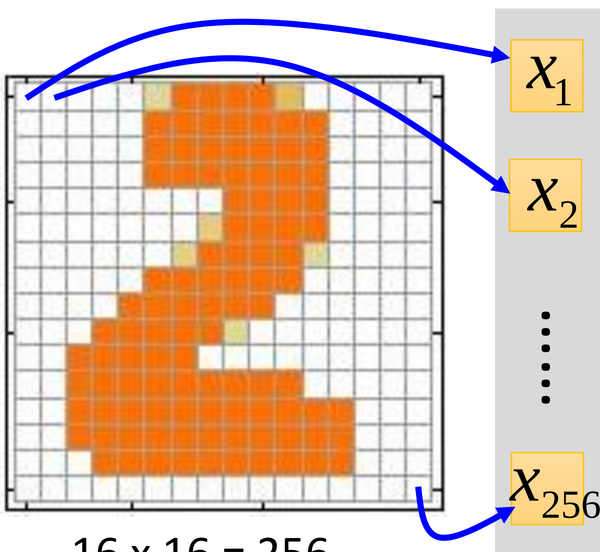
Example Application

- Handwriting Digit Recognition



Handwriting Digit Recognition

Input

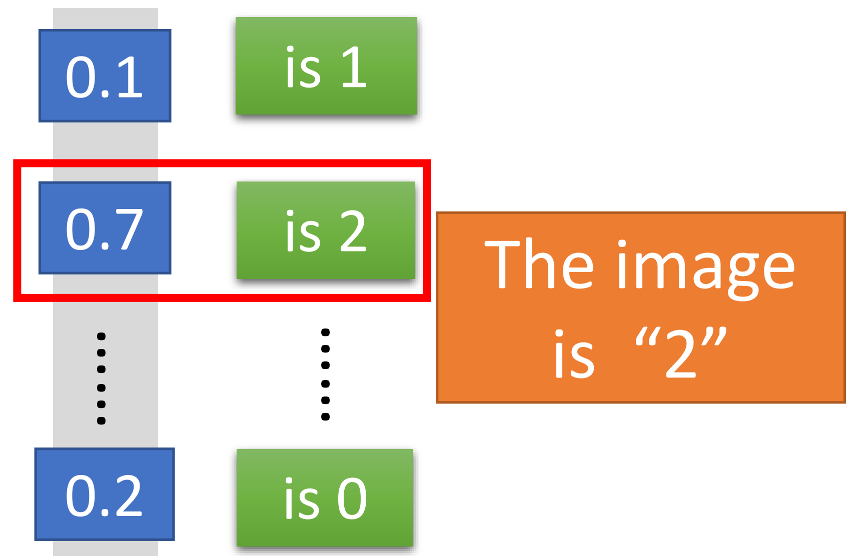


16 x 16 = 256

Ink → 1

No ink → 0

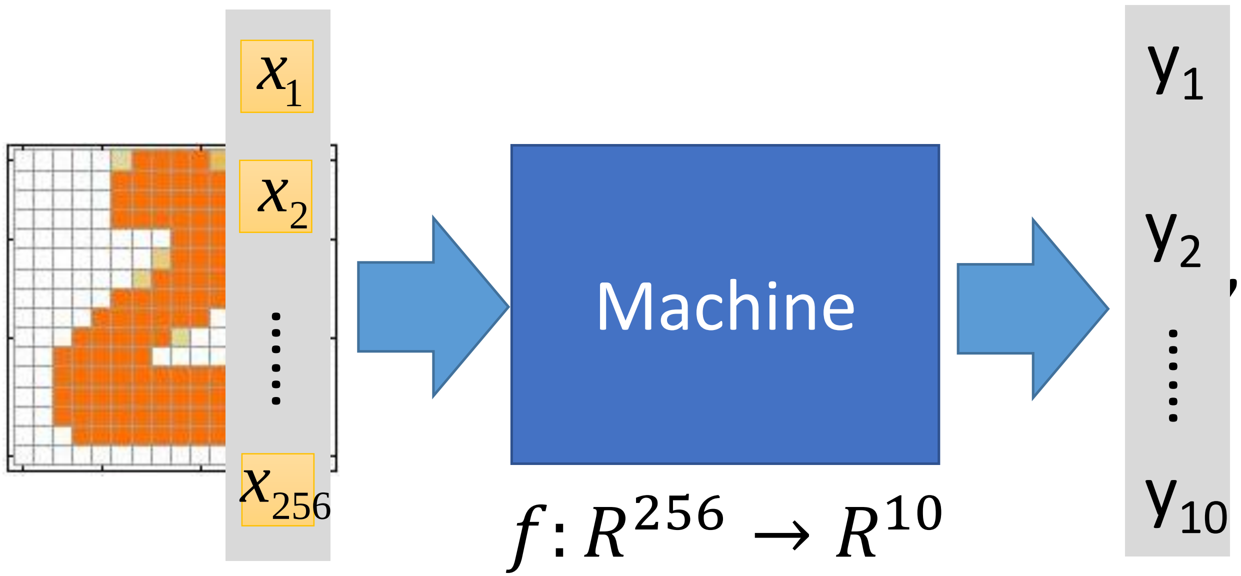
Output



Each dimension represents the confidence of a digit.

Example Application

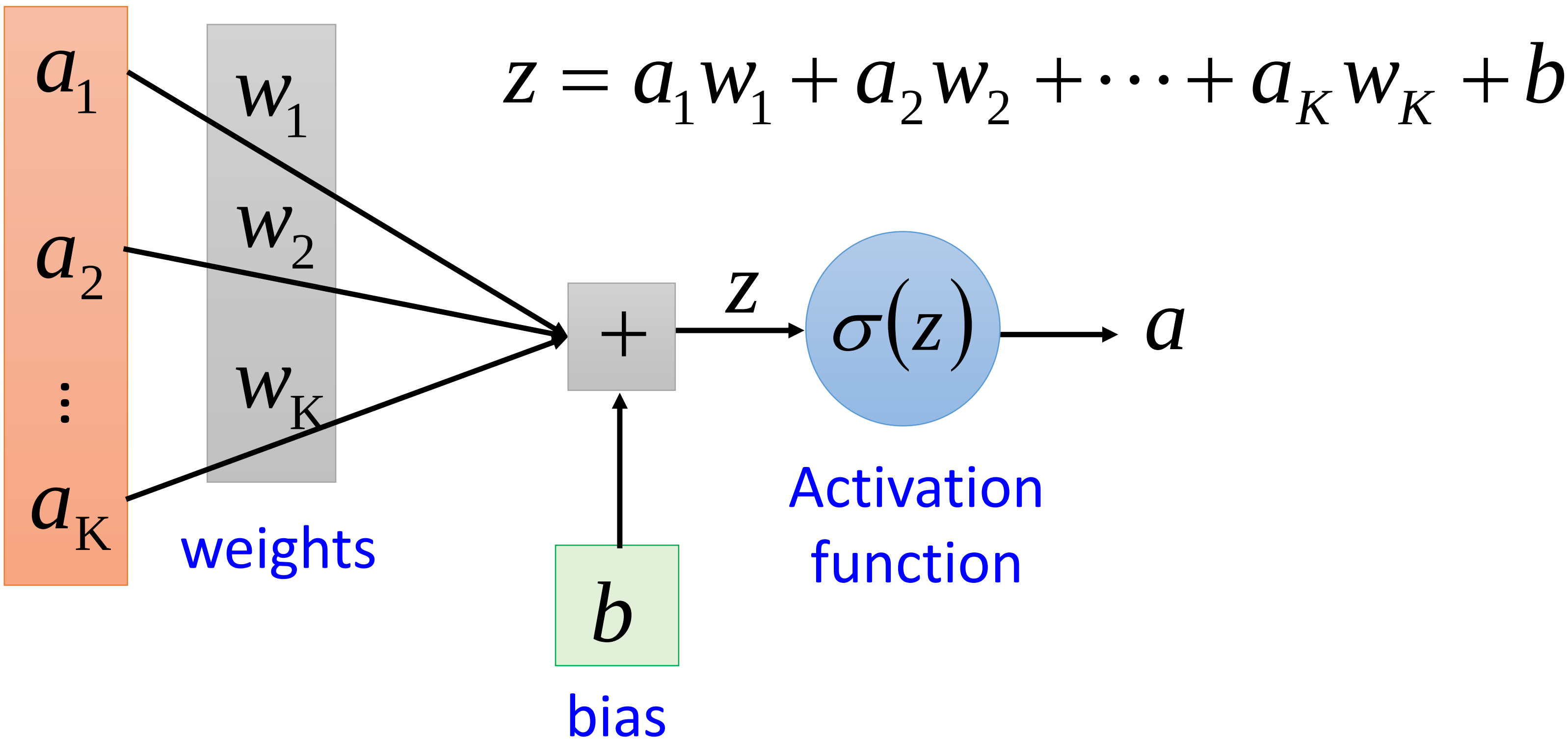
- Handwriting Digit Recognition



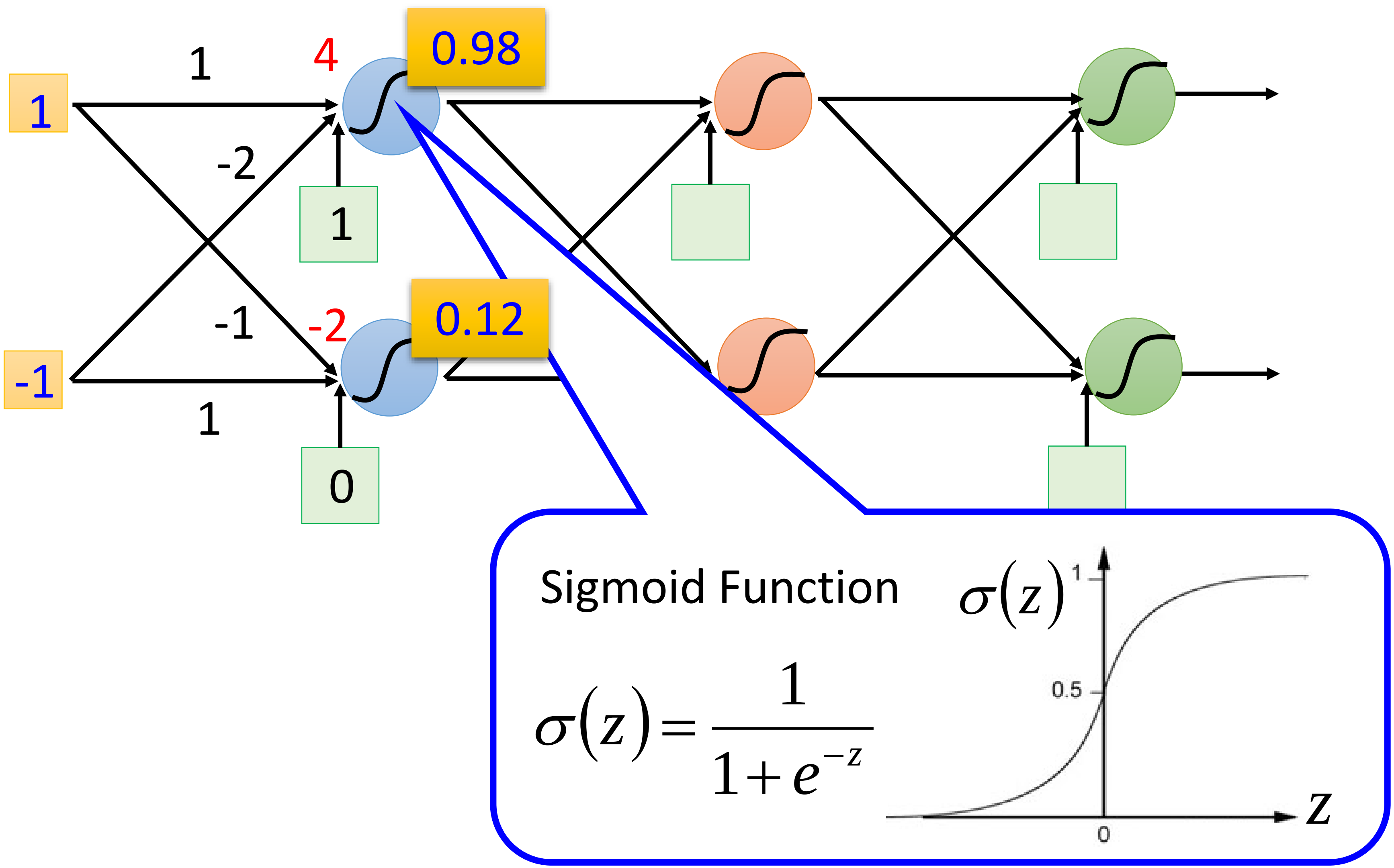
In deep learning, the function f is represented by neural network

Element of Neural Network

<u>***Neuron***</u> $f : R^K \rightarrow R$

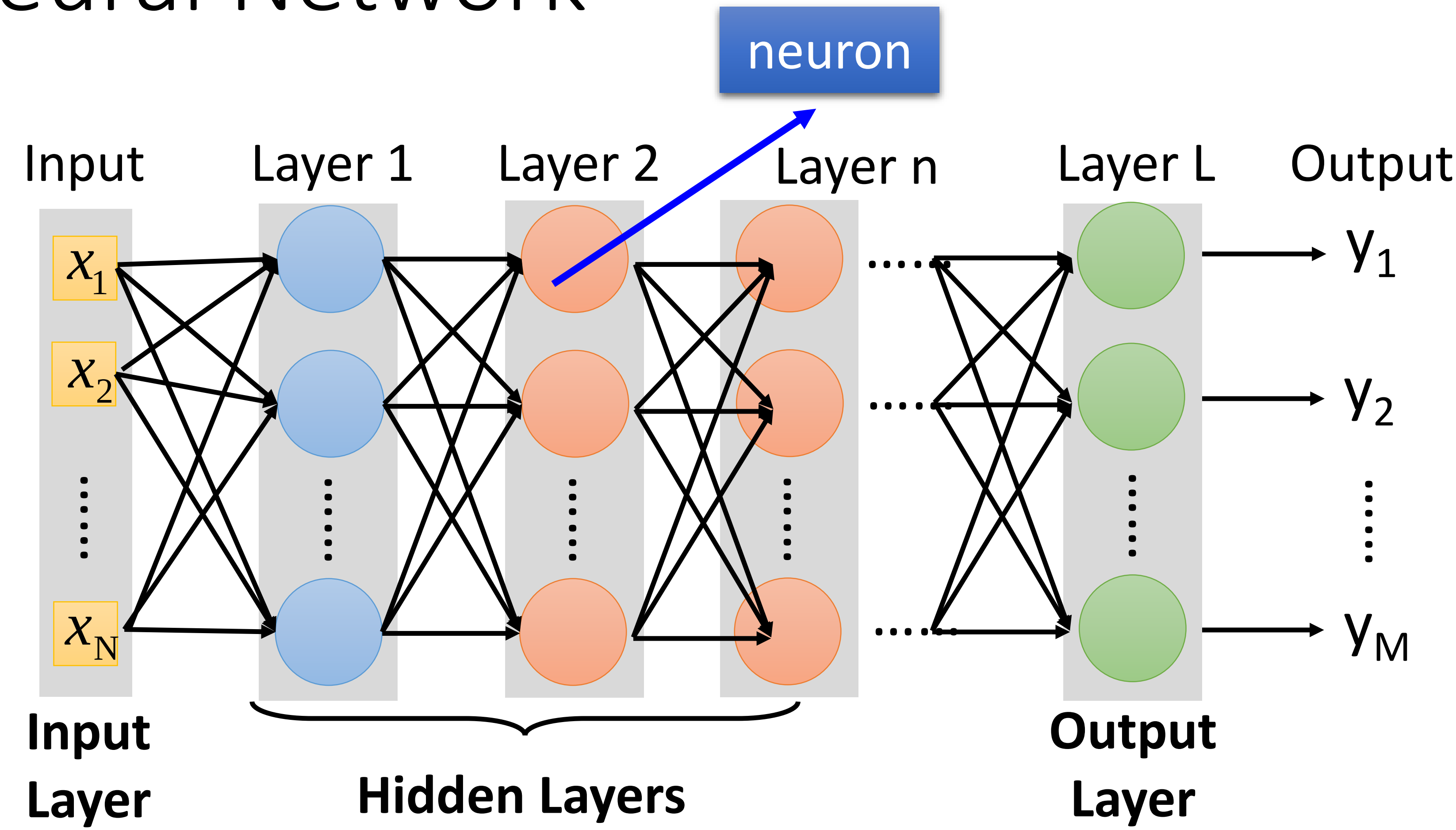


Example of Neural Network



$$\sigma(z) = \frac{1}{1+e^{-z}}$$

Neural Network



Deep means many hidden layers

what is this unit doing?

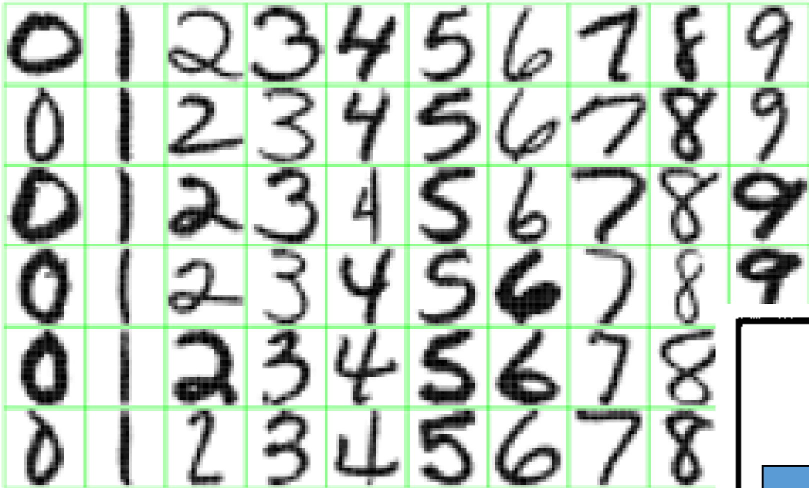
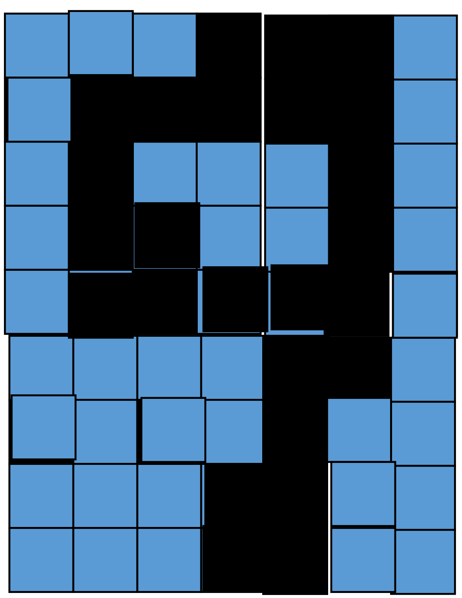
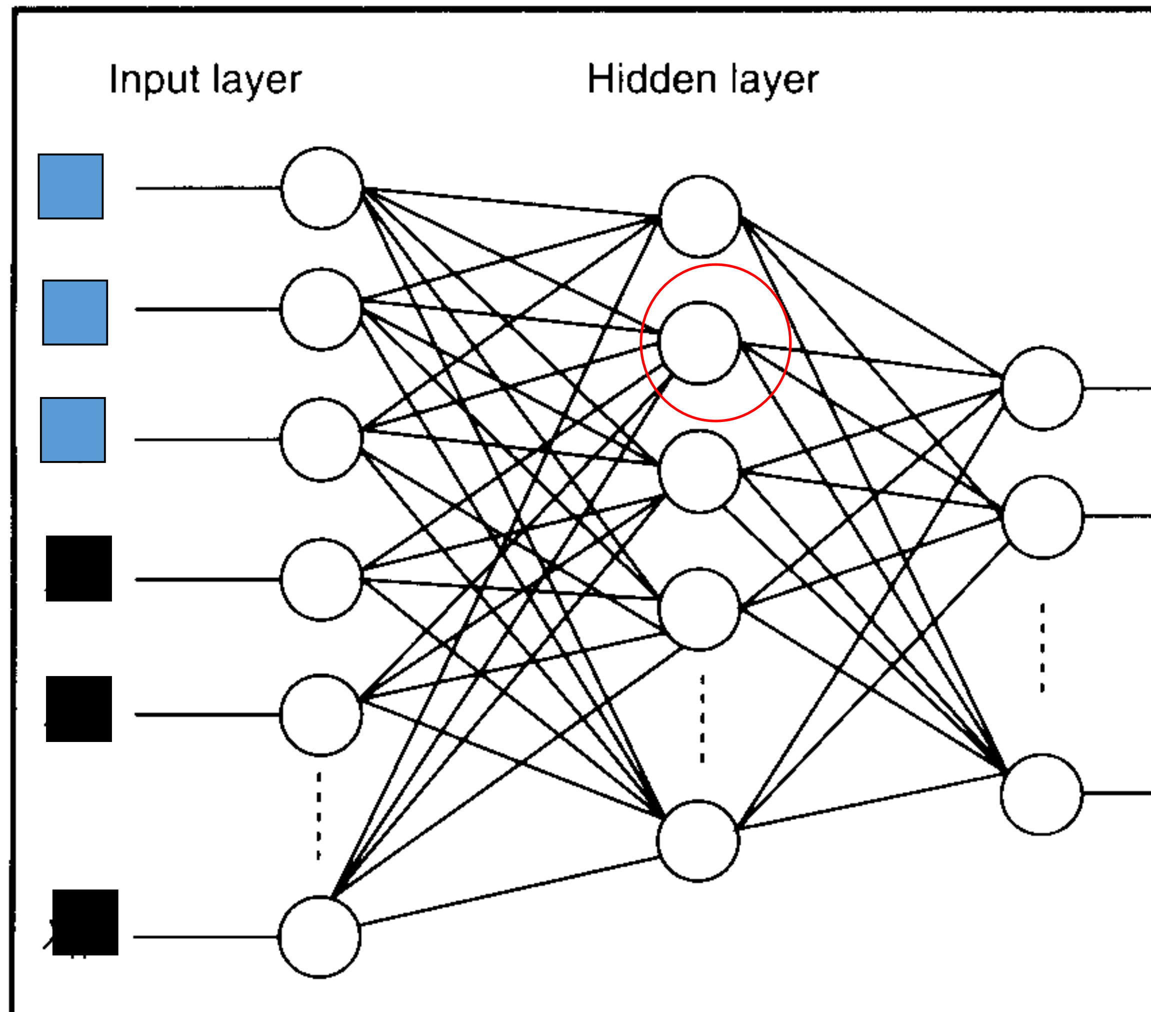


Figure 1.2: *Examples of handwritten digits fro*
postal envelopes.





Feature detectors

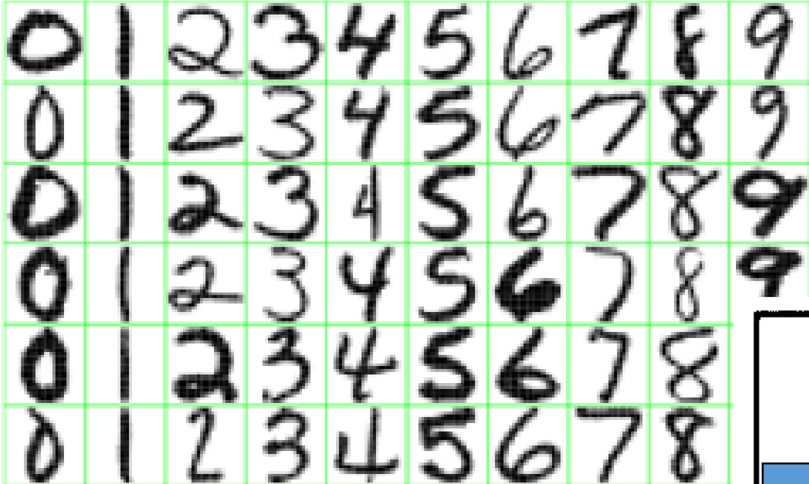
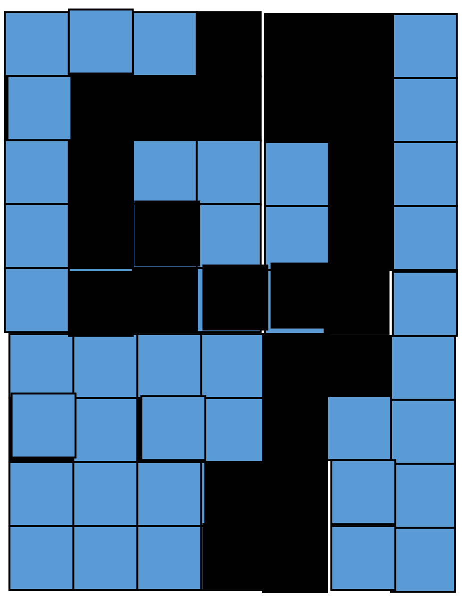
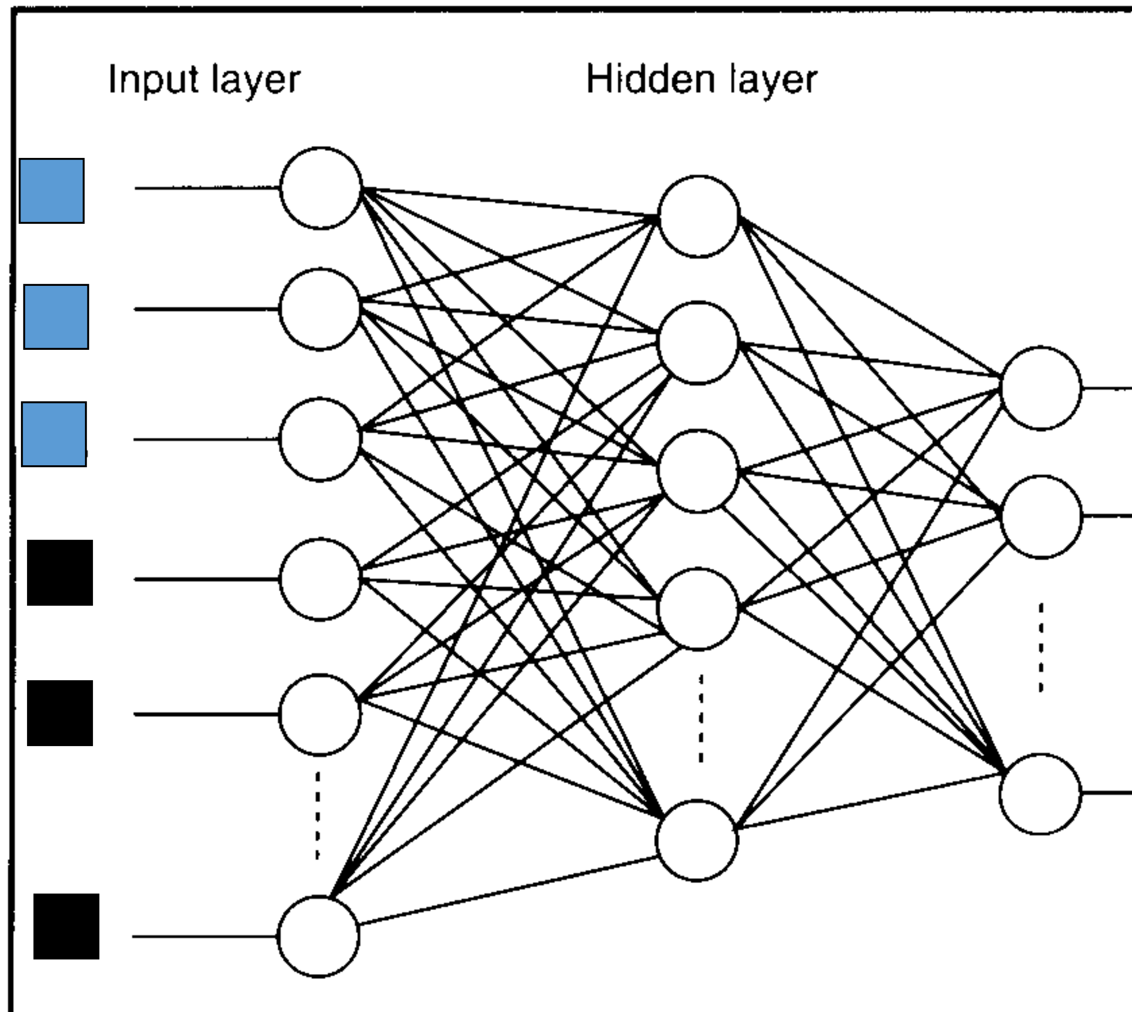


Figure 1.2: *Examples of handwritten digits from postal envelopes.*





1

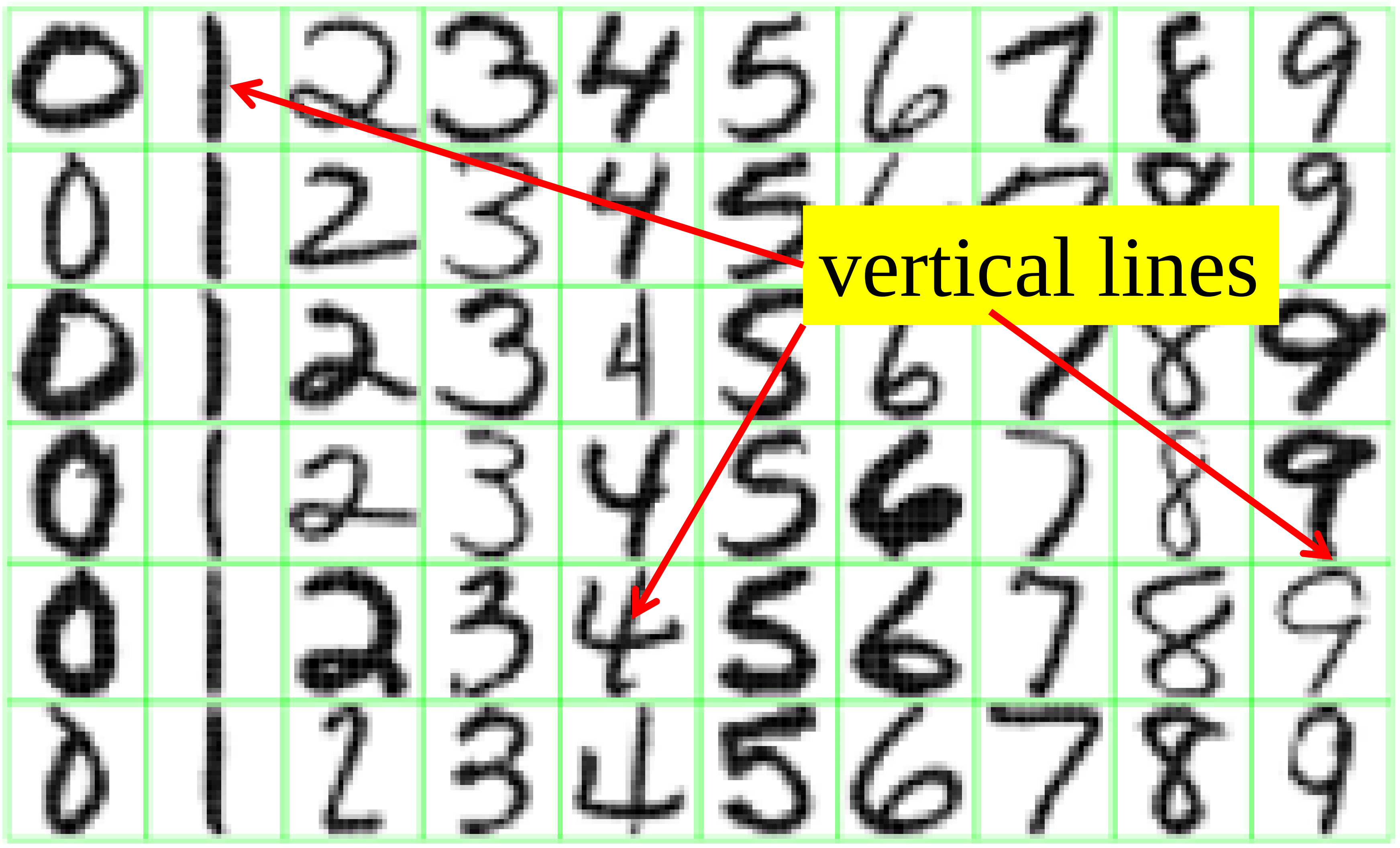


Figure 1.2: *Examples of handwritten digits from U.S. postal envelopes.*

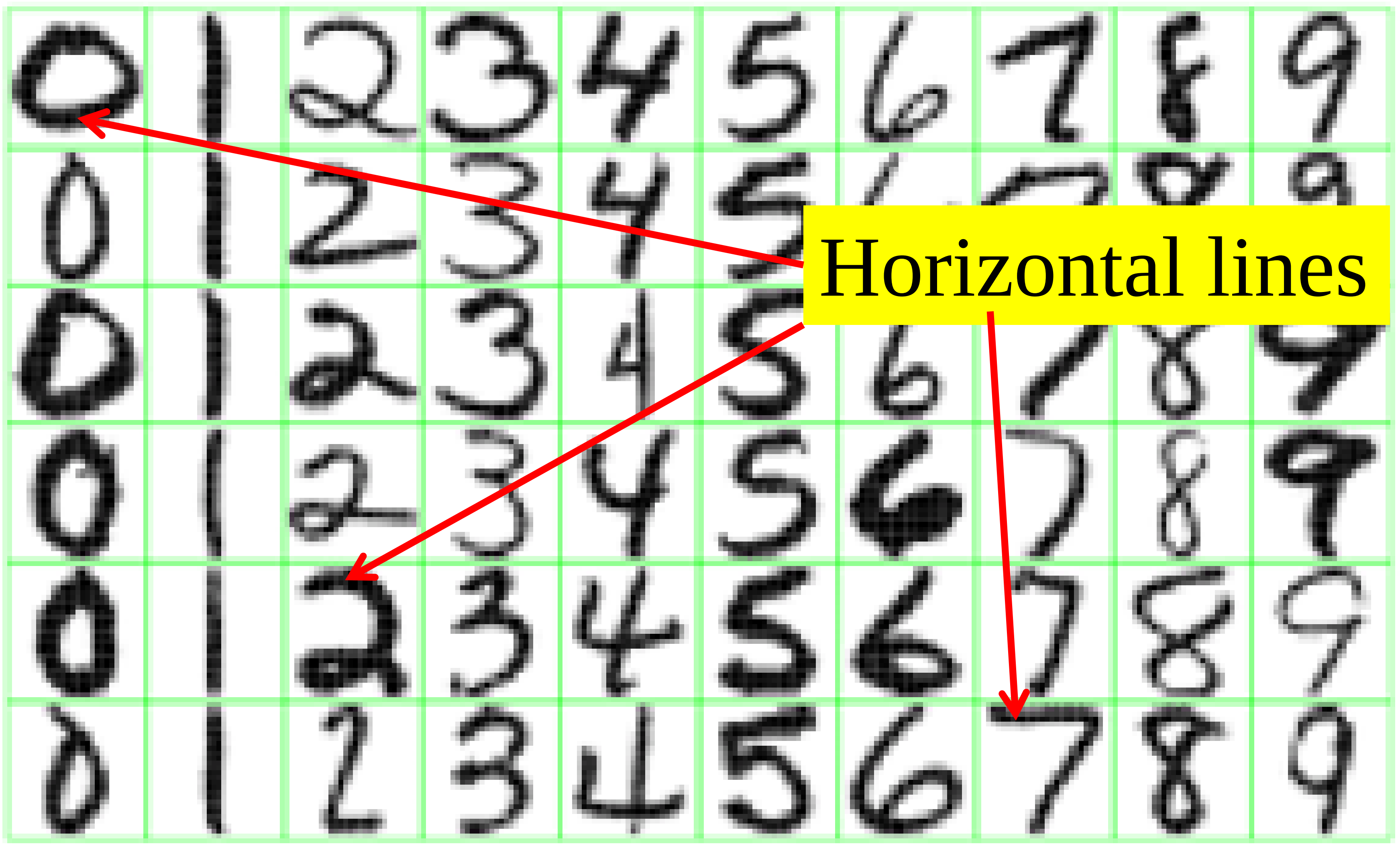
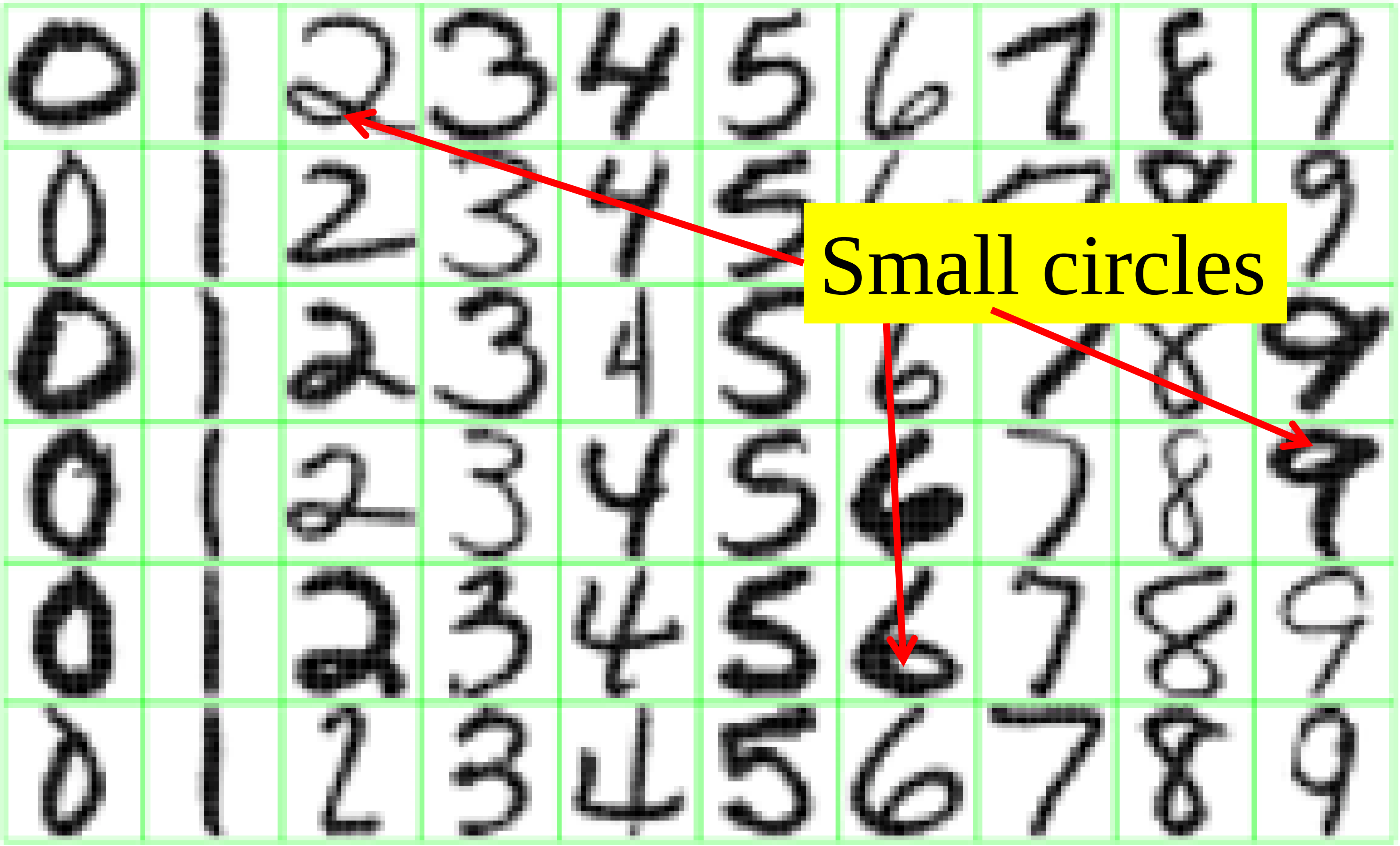


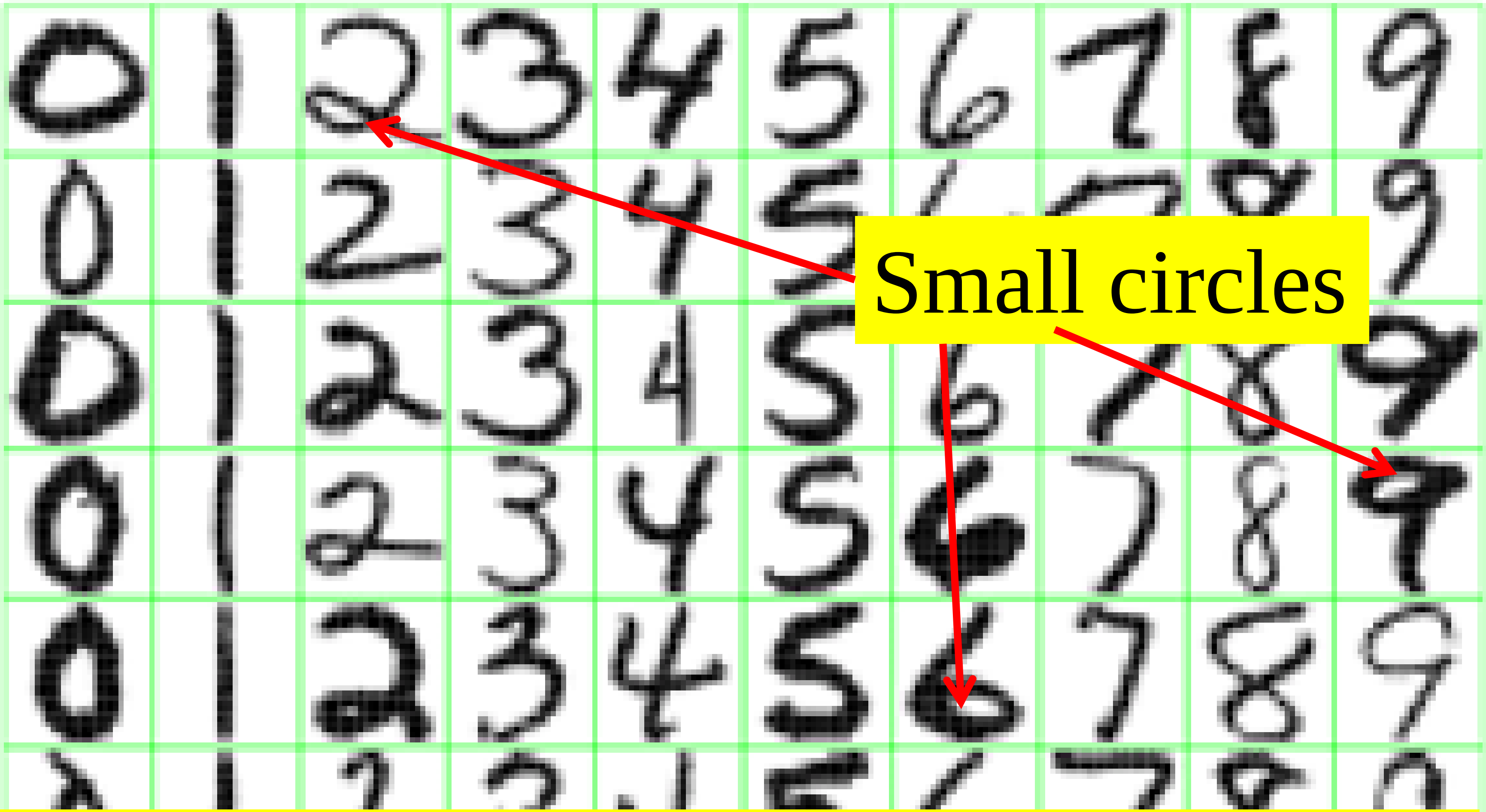
Figure 1.2: *Examples of handwritten digits from U.S. postal envelopes.*



1

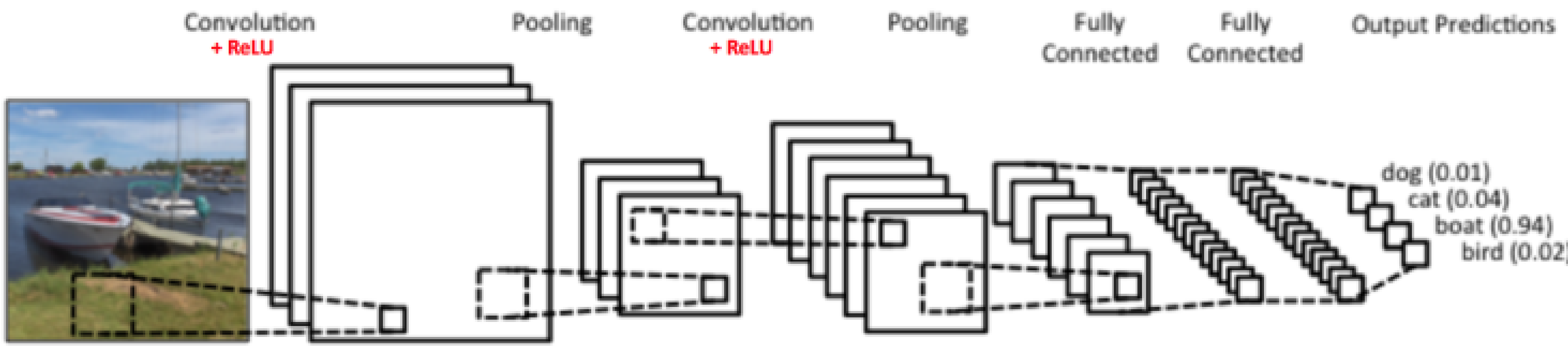
Figure 1.2: *Examples of handwritten digits from U.S. postal envelopes.*

1



But what about position invariance ???
our example unit detectors were tied to
specific parts of the image

A Simple CNN Architecture



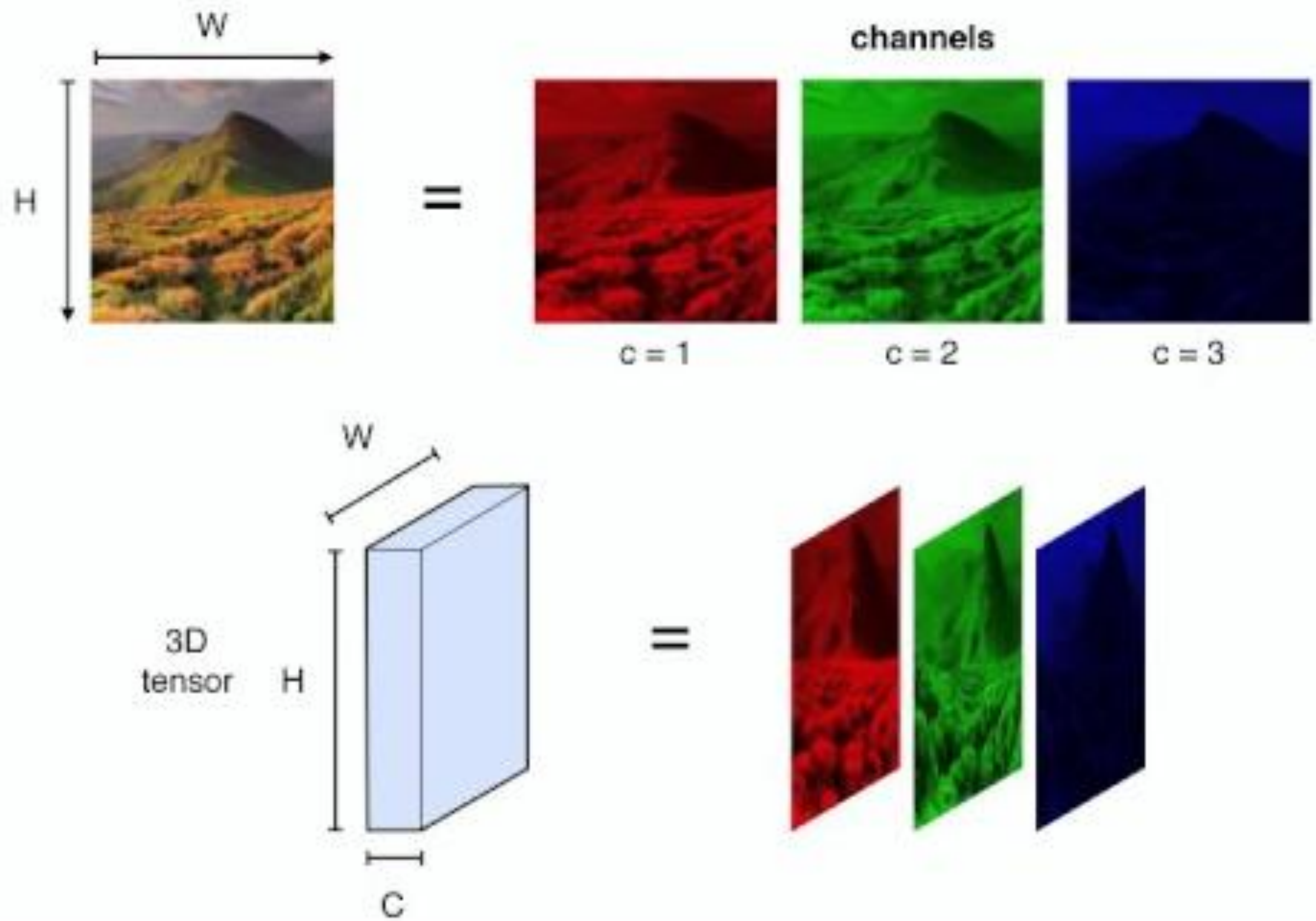
- There are four main operations in ConvNet shown above:

1. **Convolution**
2. **Non Linearity (ReLU)**
3. **Pooling or Sub Sampling**
4. **Classification (Fully Connected Layer)**

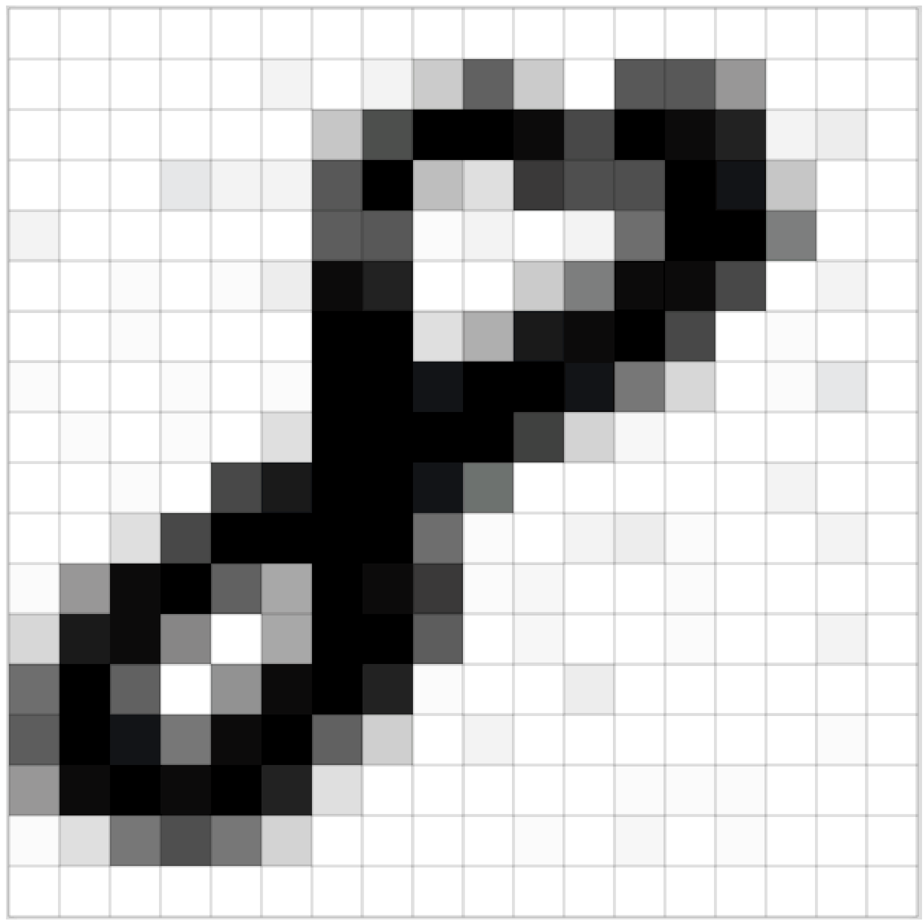
https://www.clarifai.com/technology

Data = 3D tensors

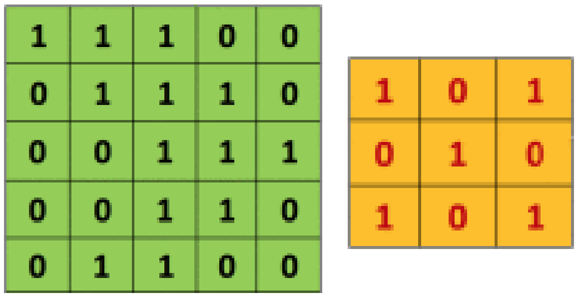
There is a vector of feature channels (e.g. RGB) at each spatial location (pixel).

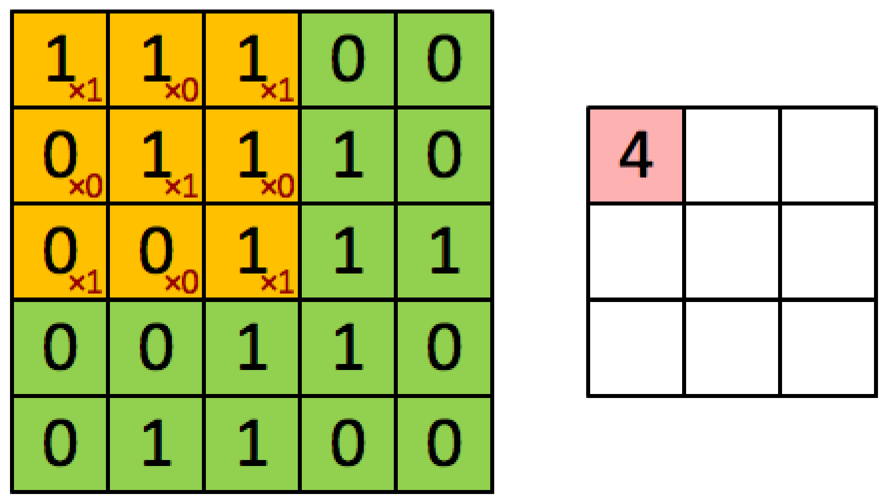


An Image is a matrix of pixel values



1. Convolution Step

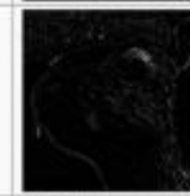




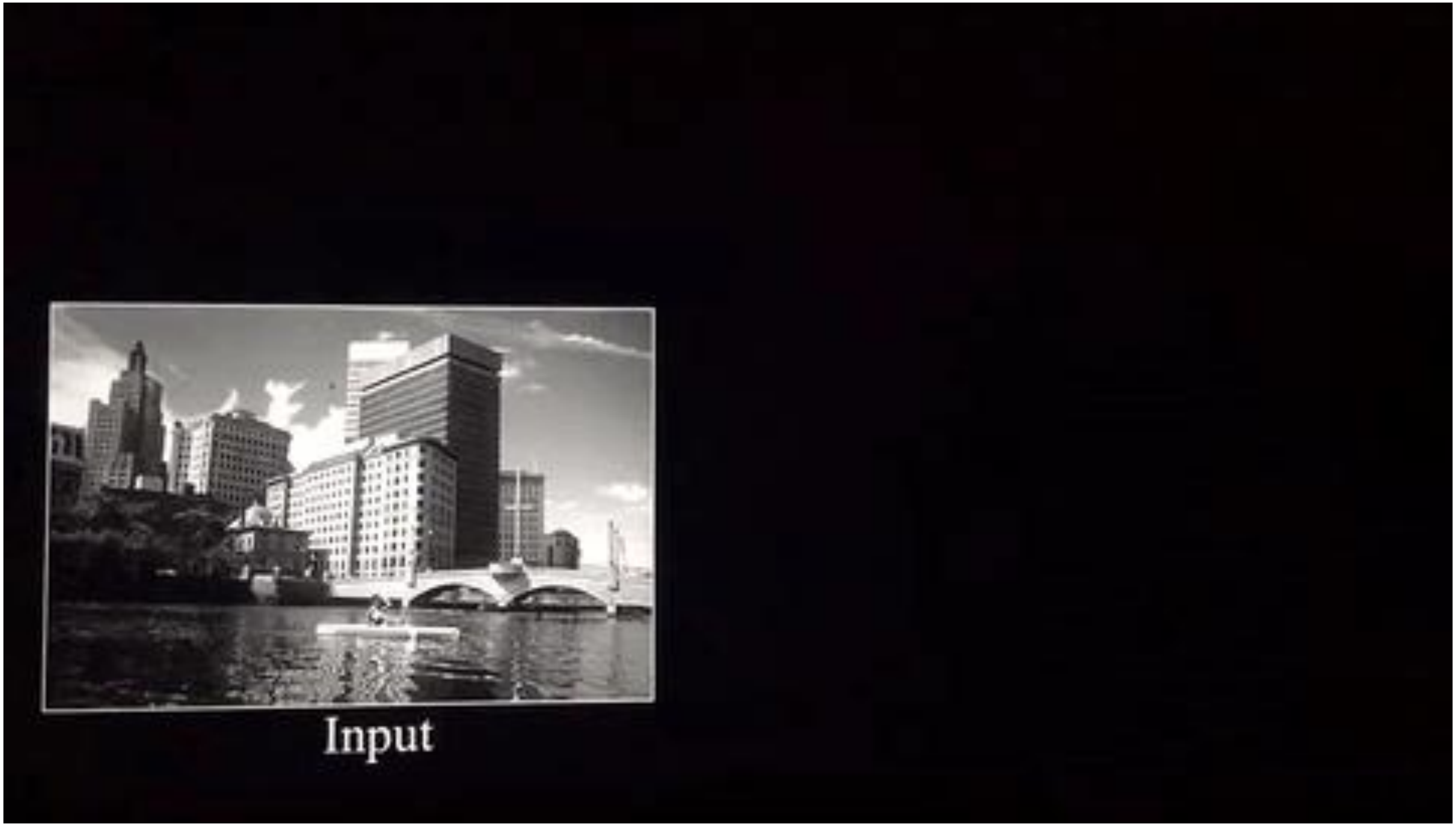
Image

Convolved Feature

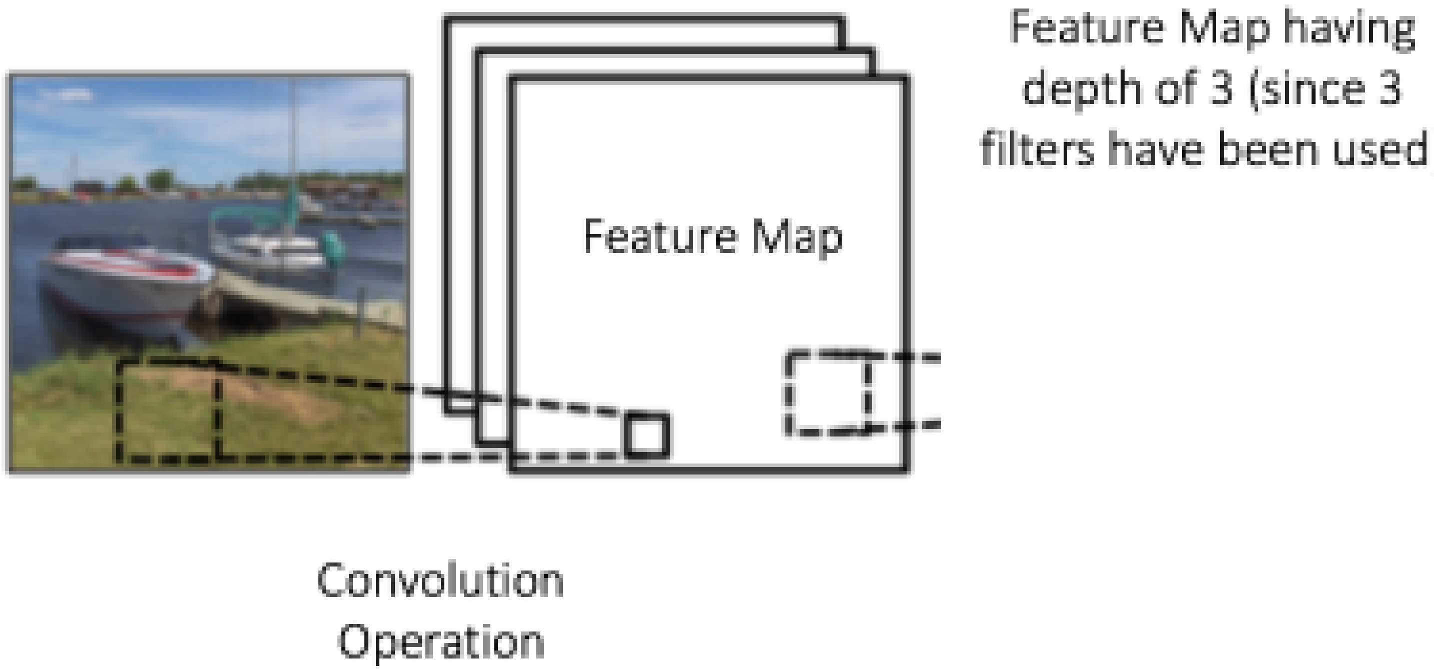
Different Filters

Operation	Filter	Convolved Image
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
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)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

Applying Filters



Feature Map

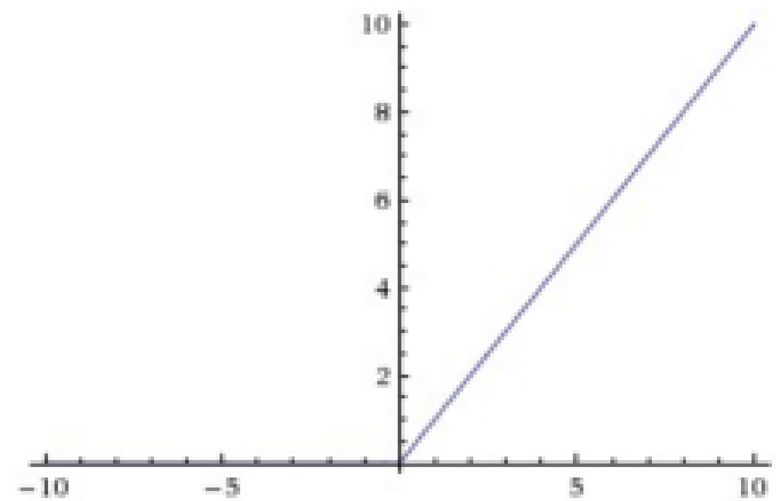


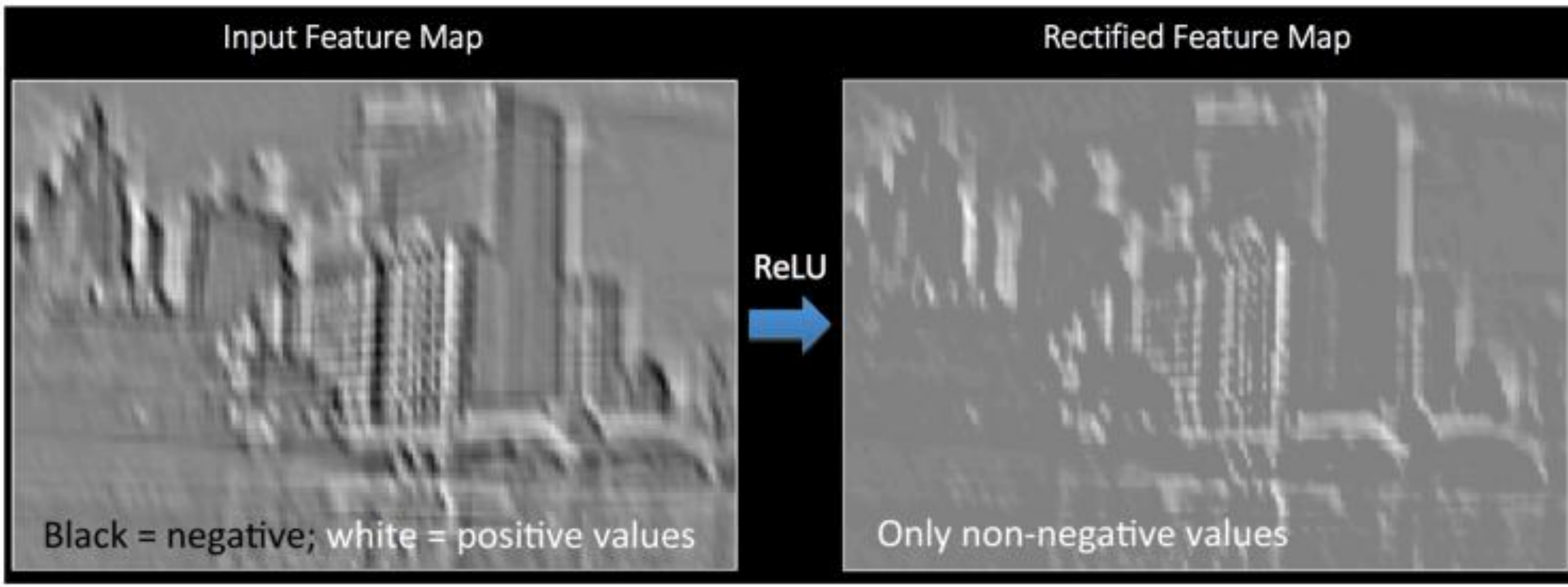
Feature Map having depth of 3 (since 3 filters have been used)
Feature Map
Convolution Operation

ReLU

- ReLU stands for Rectified Linear Unit and is a non-linear operation.
- It is applied every convolution step

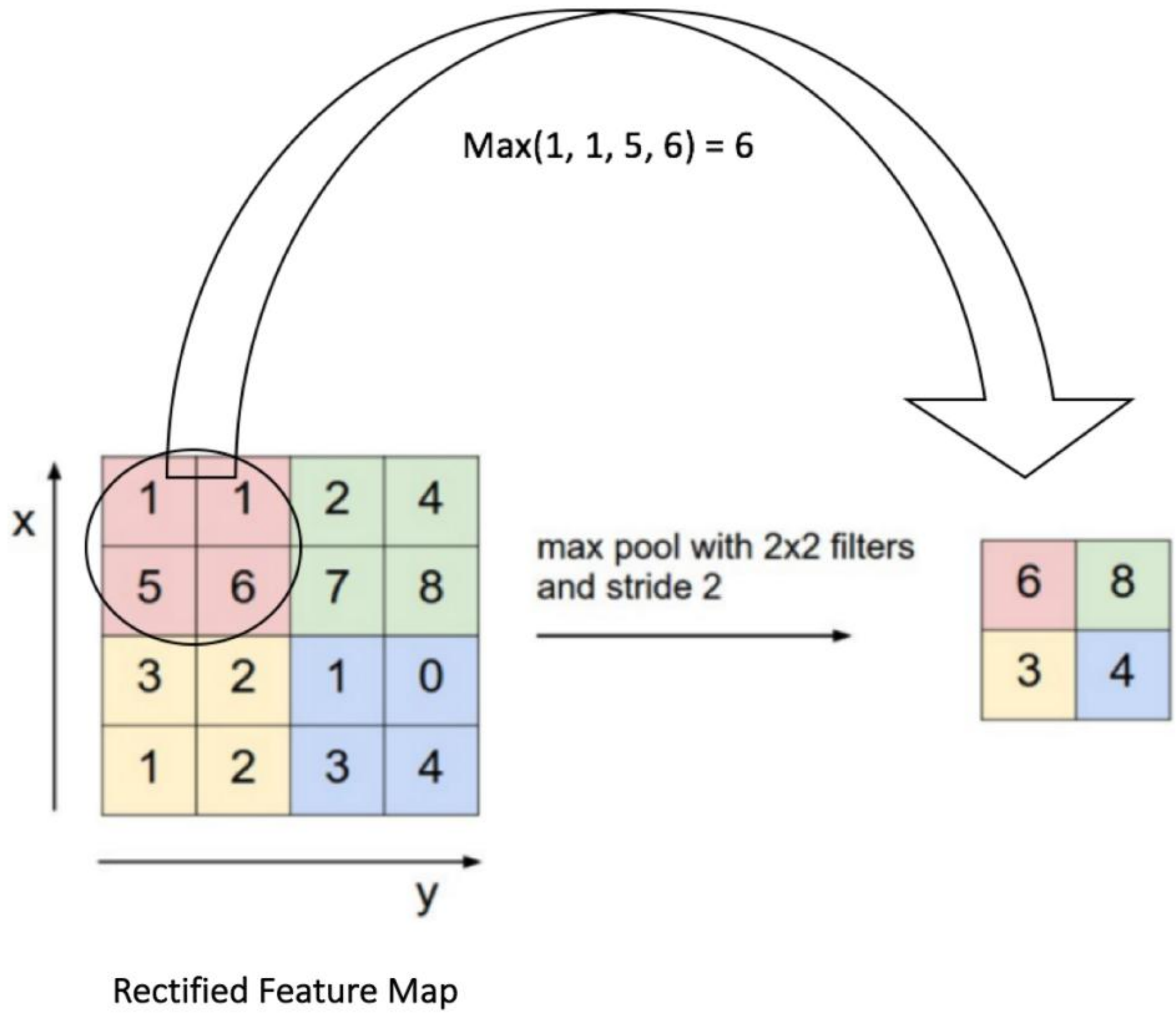
Output = Max(zero, Input)

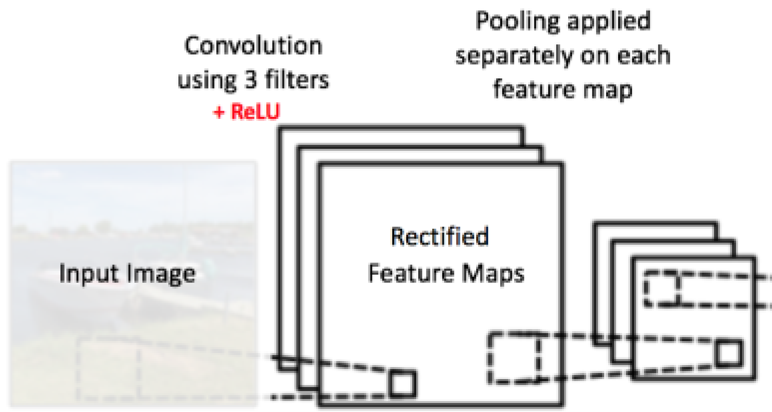


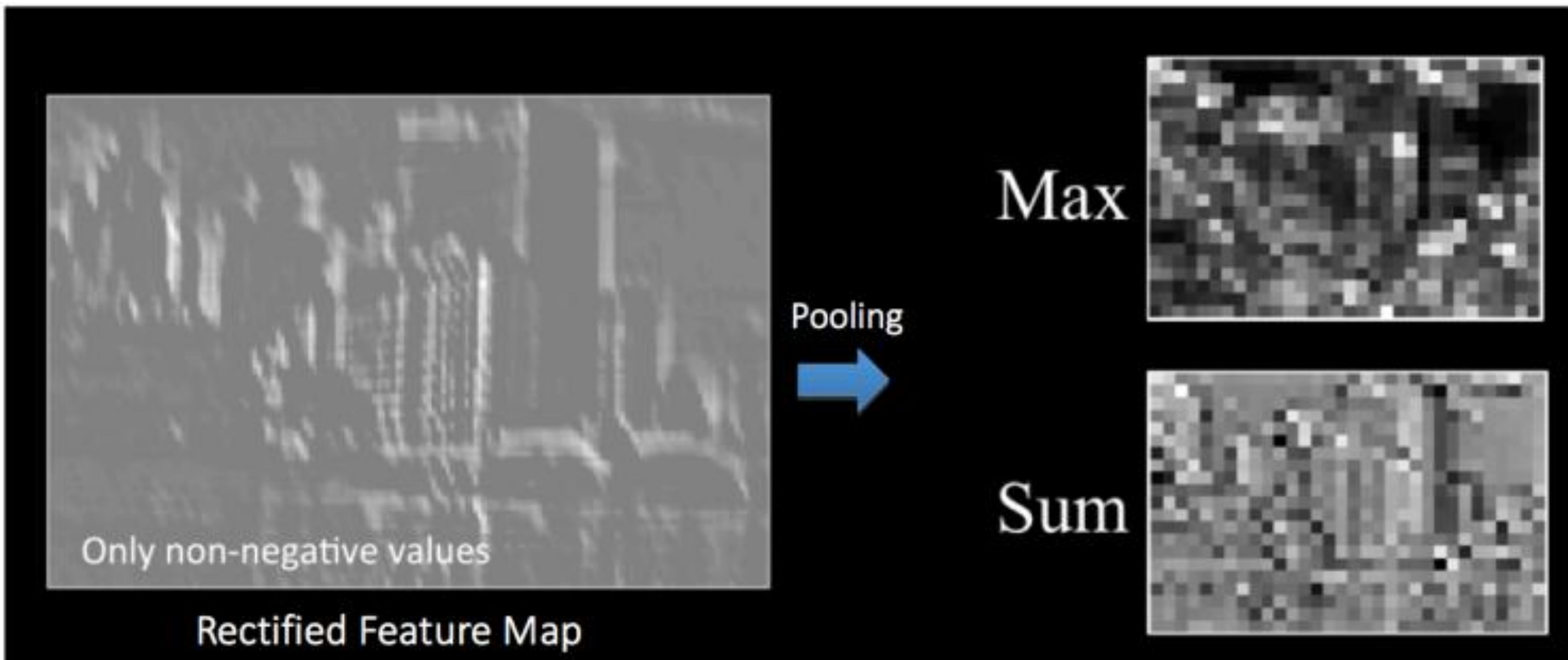


Input Feature Map
Rectified Feature Map
ReLU
Black = negative; white = positive values
Only non-negative values

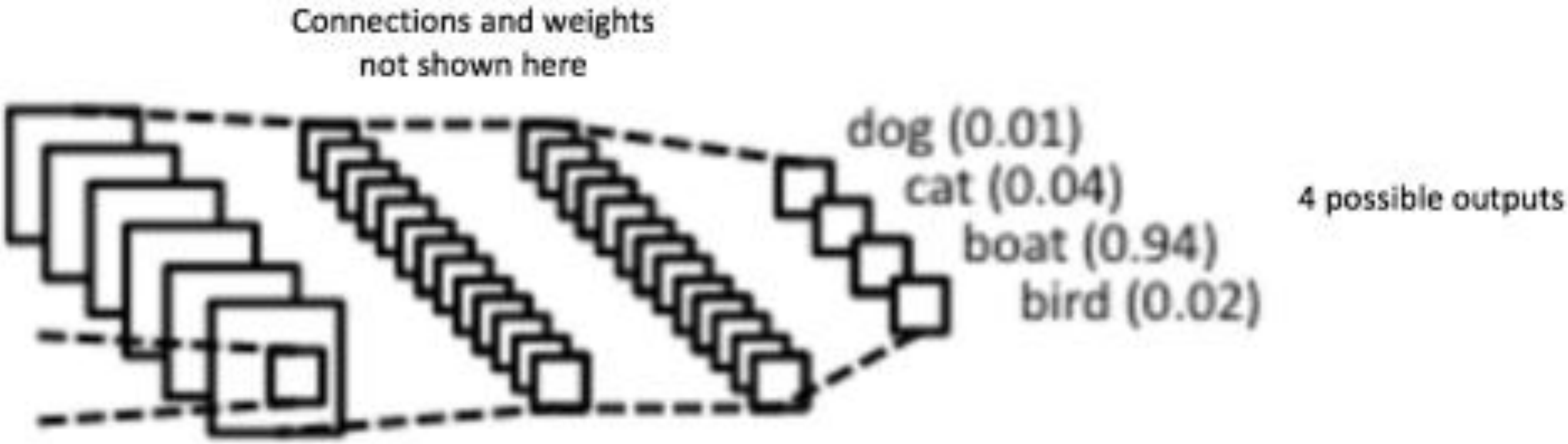
3.The Pooling Step



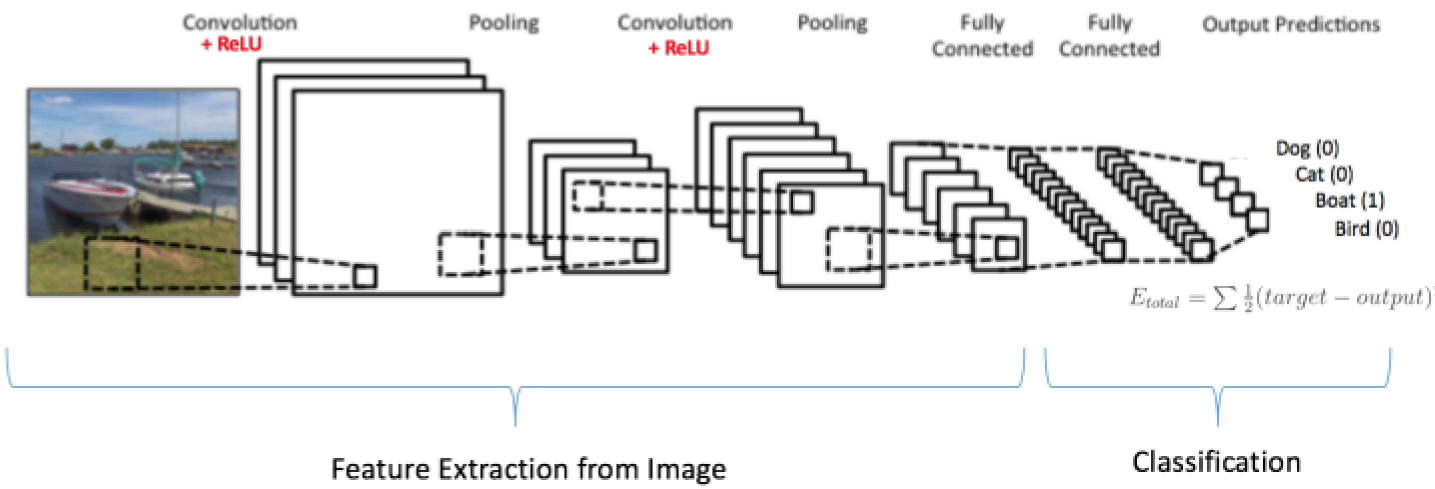

Convolution using 3 filters
+ ReLU
Pooling applied separately on each feature map
Input Image
Rectified Feature Maps

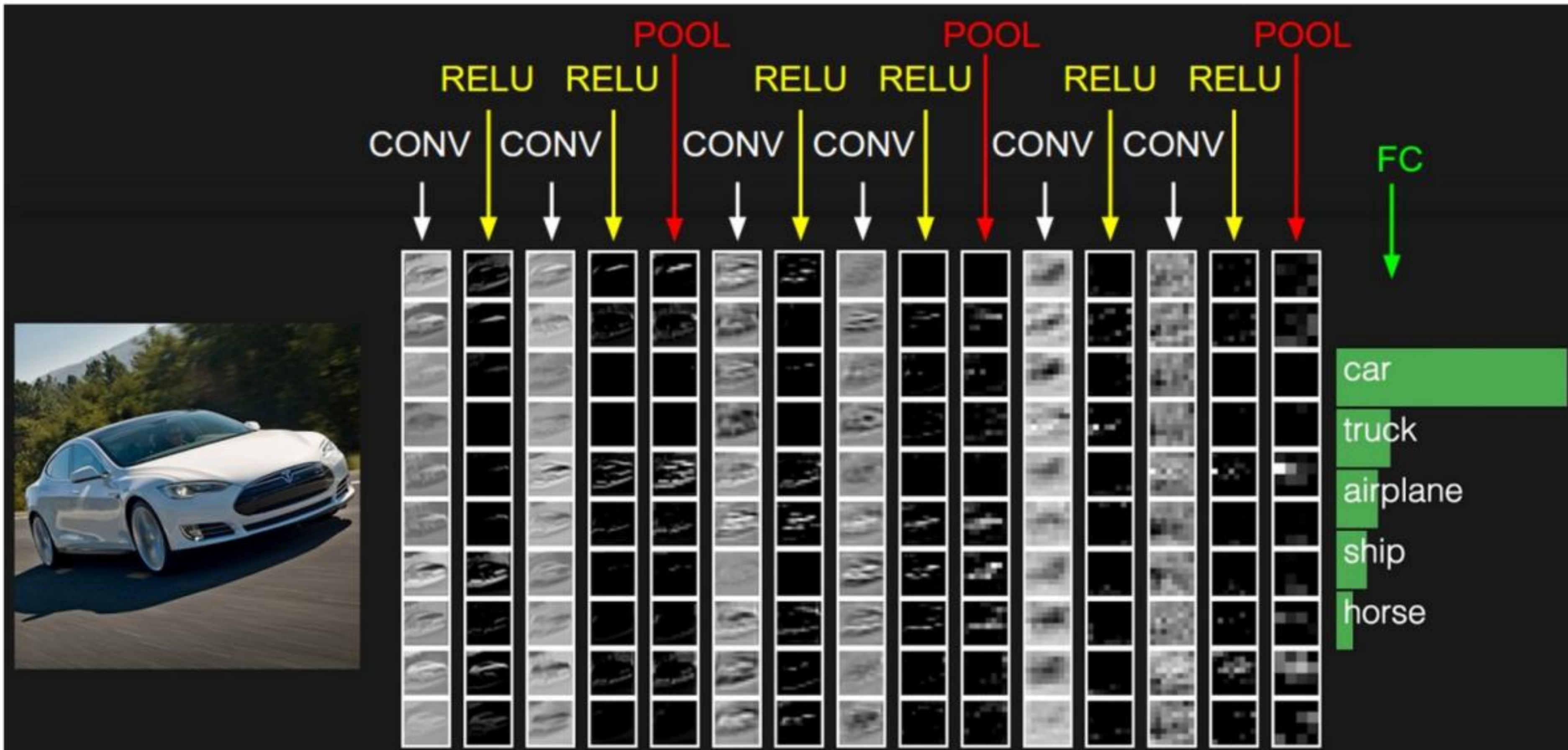

Only non-negative values
Rectified Feature Map
Pooling
Max
Sum

Fully Connected Layer



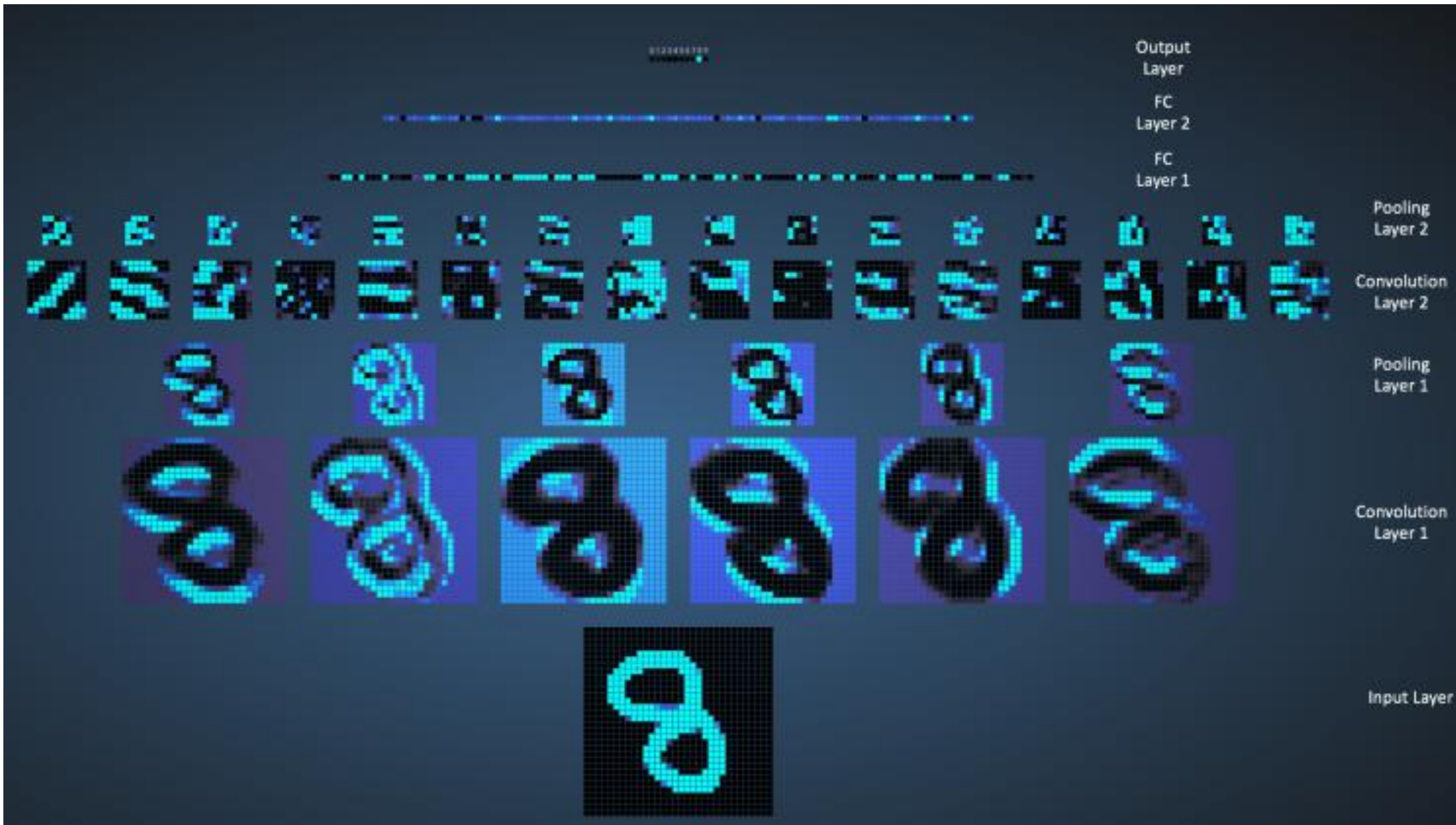
Put it Together





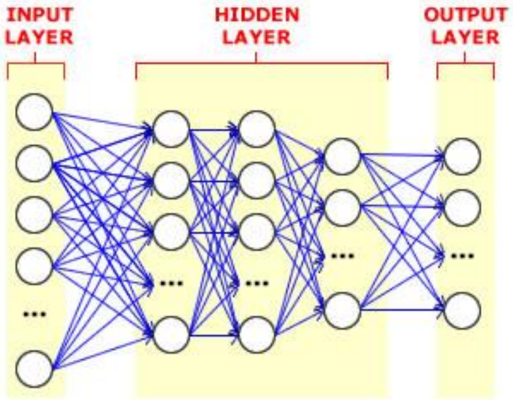
POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse


Pooling
Layer 1
Convolution
Layer 1


Output
Layer
FC
Layer 2
FC
Layer 1
Pooling
Layer 2
Convolution
Layer 2
Pooling
Layer 1
Convolution
Layer 1
Input Layer

And that's that

- That's the basic idea
- There are many many types of deep learning,
- different kinds of autoencoder, variations on architectures and training algorithms, etc…
- Very fast growing area …



Concluding Remarks

- Introduction of deep learning
- Discussing some reasons using deep learning
- New techniques for deep learning
  - ReLU, Maxout
  - Giving all the parameters different learning rates
  - Dropout
- Network with memory
  - Recurrent neural network
  - Long short-term memory (LSTM)

Reading Materials

- “Neural Networks and Deep Learning”
  - written by Michael Nielsen
  - http://neuralnetworksanddeeplearning.com/
- “Deep Learning”
- Written by Yoshua Bengio, Ian J. Goodfellow and Aaron Courville
  - http://www.iro.umontreal.ca/~bengioy/dlbook/

Thank you
for your attention!

shahid.awan@umt.edu.pk

https://sites.google.com/site/shahidmawan/