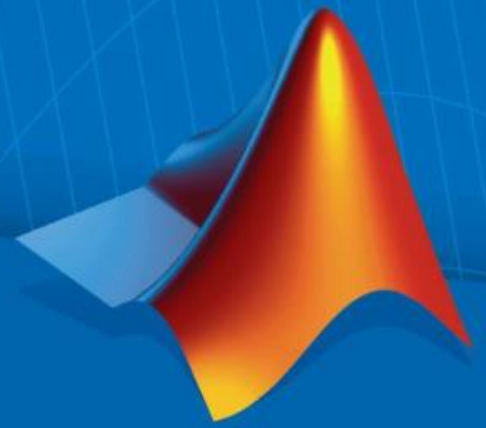
TeraSoft
www.terasoft.com.tw



AI Computer Vision with MATLAB

Fred Liu

Application Engineer

MathWorks®
Accelerating the pace of engineering and science

Agenda

- Introduction AI
  - History
  - Application
- Deep Learning with MATLAB
  - What is Deep Learning?
  - Layers in Convolution Neural Network
  - Image classification using pre-trained network
  - Train a new model
  - Object detection & GPU Coder

© Terasoft, Inc.

Deep Learning in Action



Speech recognition

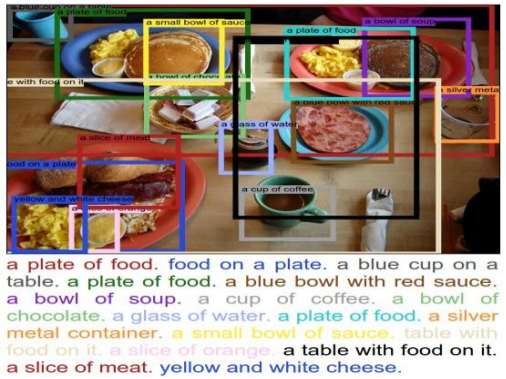
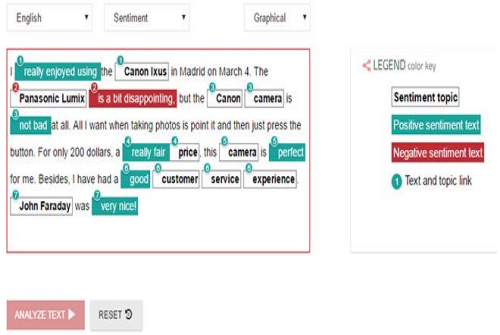


Image captioning



Algorithmic Trading



Sentiment Analysis



Control Application

Where do we start?

- Application
- Inputs/Outputs
- Network Architecture

© Terasoft, Inc.

Sequence Applications ↔ Inputs/Outputs

one to one

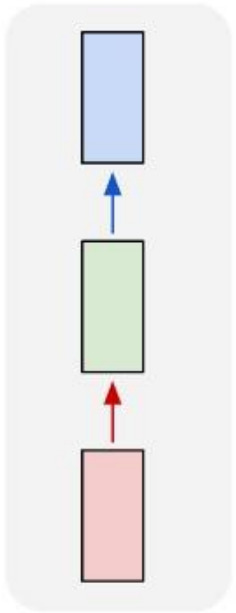


Image classification

one to sequence

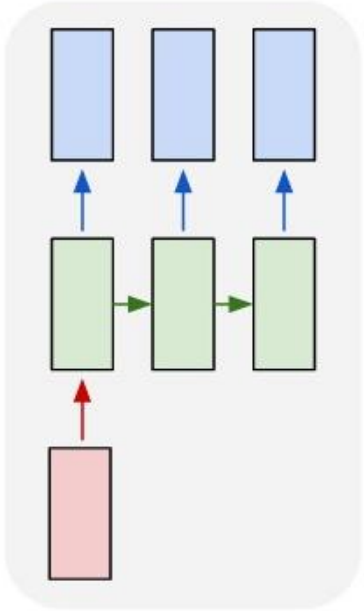
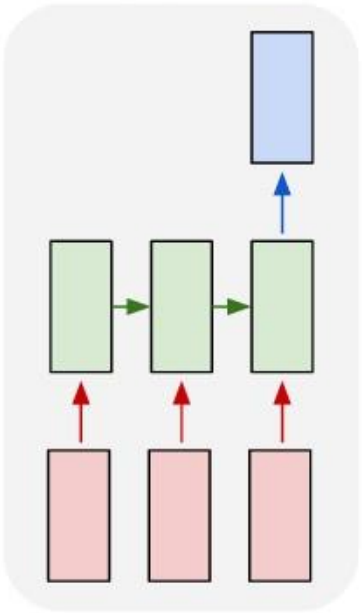


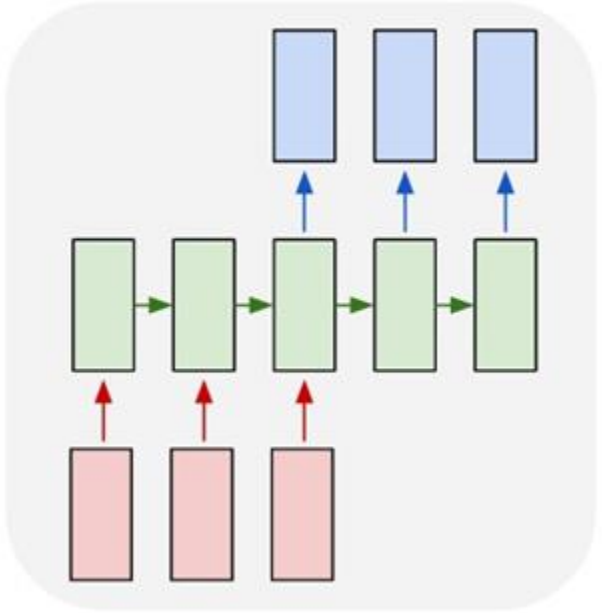
Image captioning

sequence to one



- Time series classification
- Sentiment analysis
- Activity recognition

sequence to sequence



- Language translation
- Video description
- Speech-to-text

© Terasoft, Inc.

Examples



Speech recognition

Type: sequence to sequence
Input: Signal features
Output: Phrases

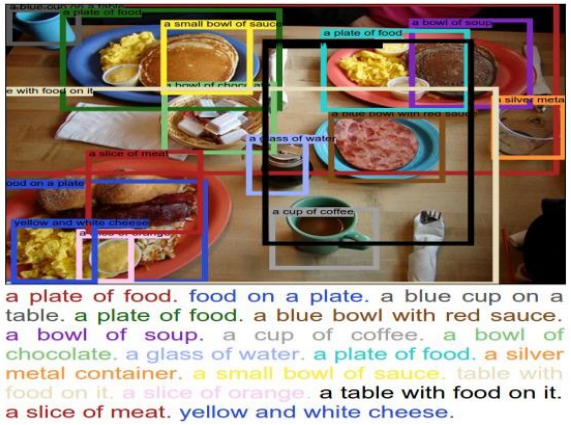


Image captioning

Type: one to sequence
Input: Images
Output: Captions

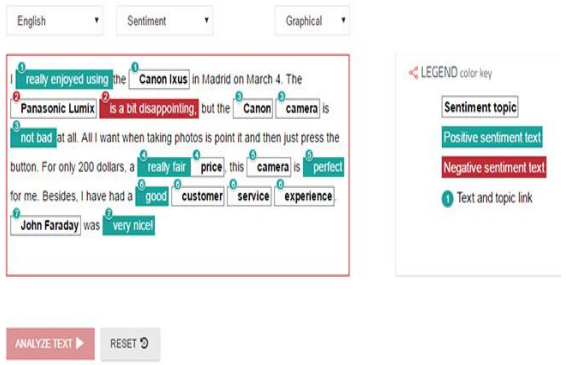


Financial Trading

Type: sequence to one
Input: Time series data
Output: Buy/Sell

© Terasoft, Inc.

Examples



Sentiment Analysis



Control Application

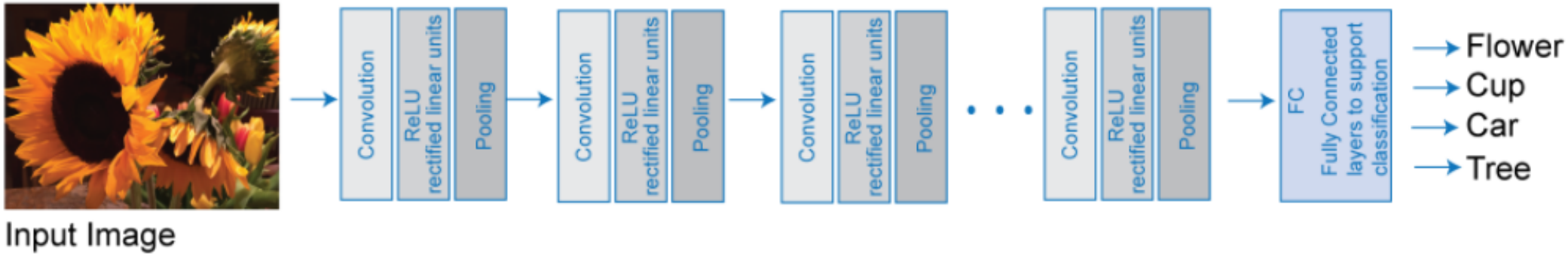
Type: sequence to one
Input: Text
Output: Phrases

Type: one to sequence
Input: Agent+Rules
Output: Policy

© Terasoft, Inc.

Architecture: Convolutional Neural Networks (CNN)

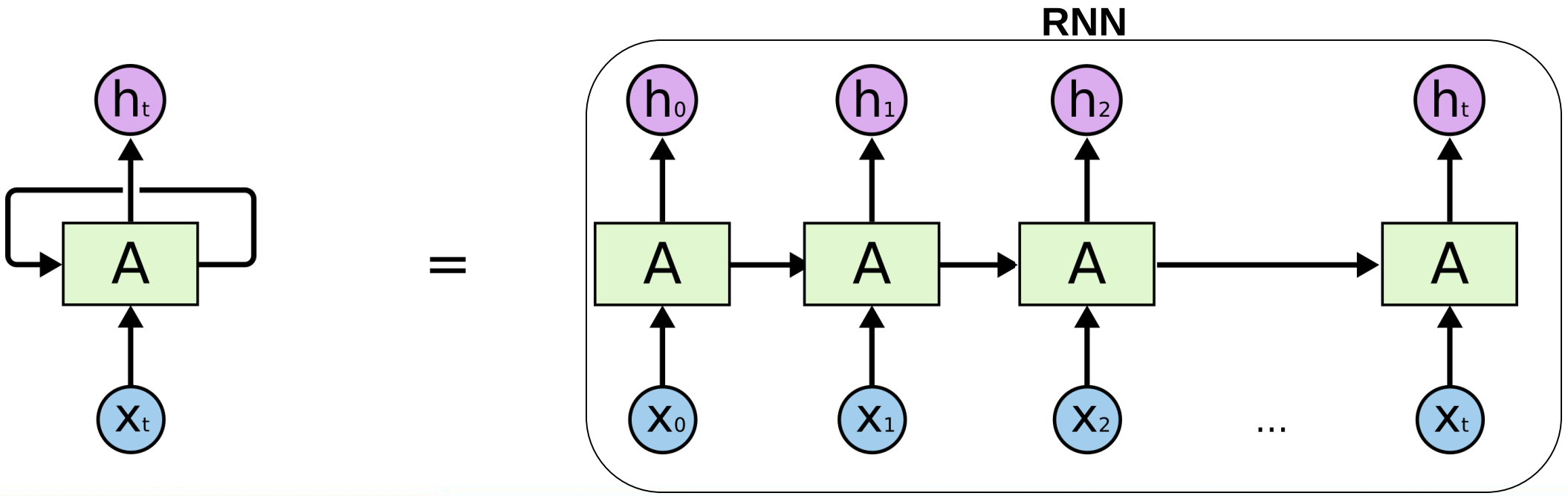
- CNNs are ideal for images and videos processing
- CNN take a fixed size input and generate fixed-size outputs.
- Convolution puts the input images through a set of convolutional filters, each of which activates certain features from the images.



© Terasoft, Inc.

Architecture: Recurrent Neural Networks (RNN)

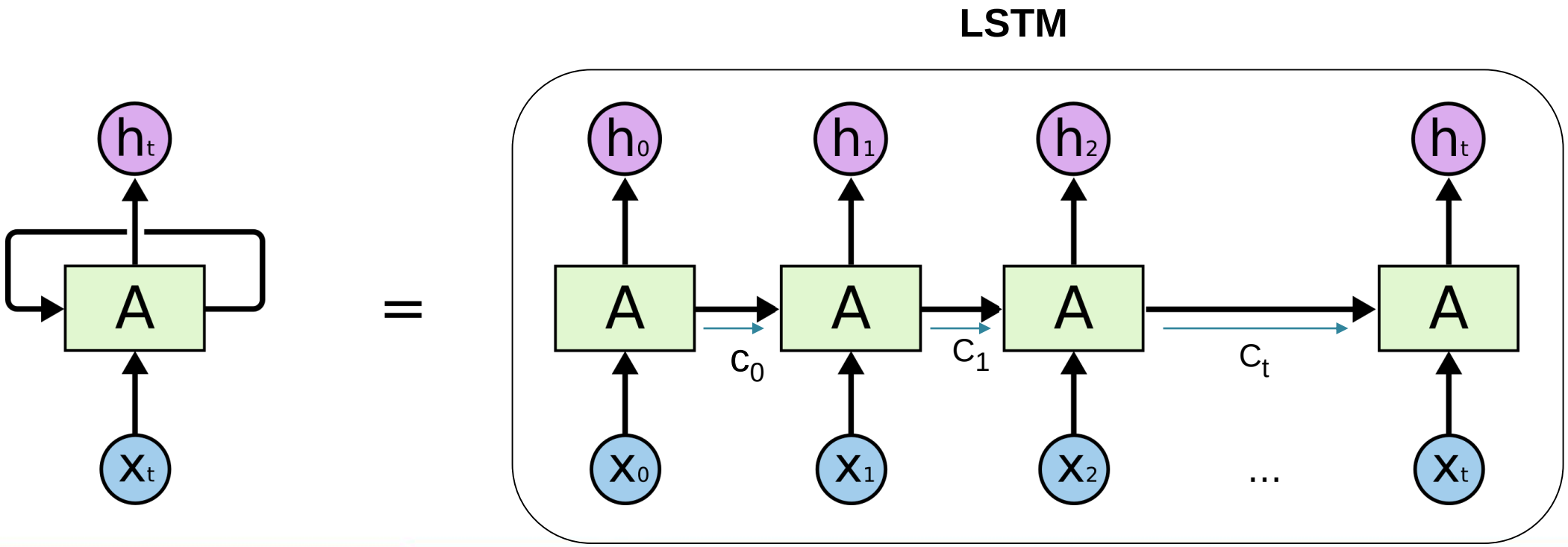
- RNNs are ideal for text and speech analysis.
- RNNs are neural networks which **connect through time.**
- RNN can handle arbitrary input/output lengths.
- They have the capability to use the dependencies among inputs.



© Terasoft, Inc.

Architecture: Long Short-Term Memory (LSTM)

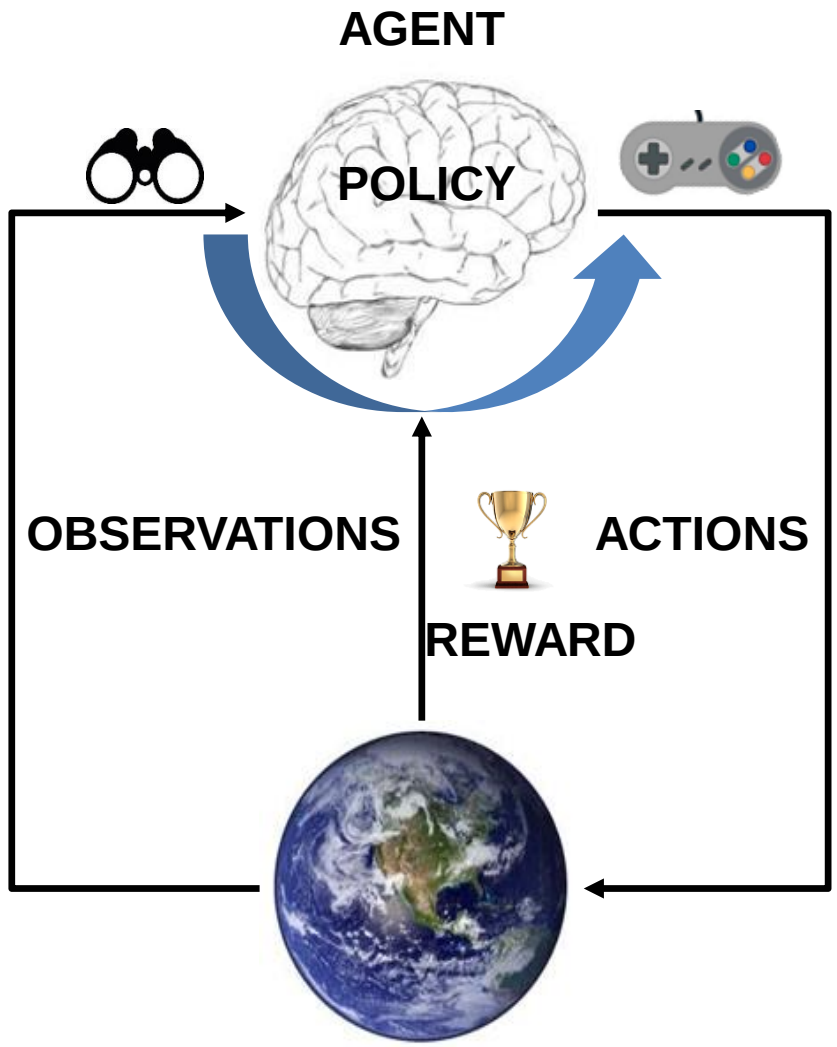
- LSTM are an extension of Recurrent Neural Networks
- LSTMs also **connect through time.** They preserve the long-term and short-term dependencies



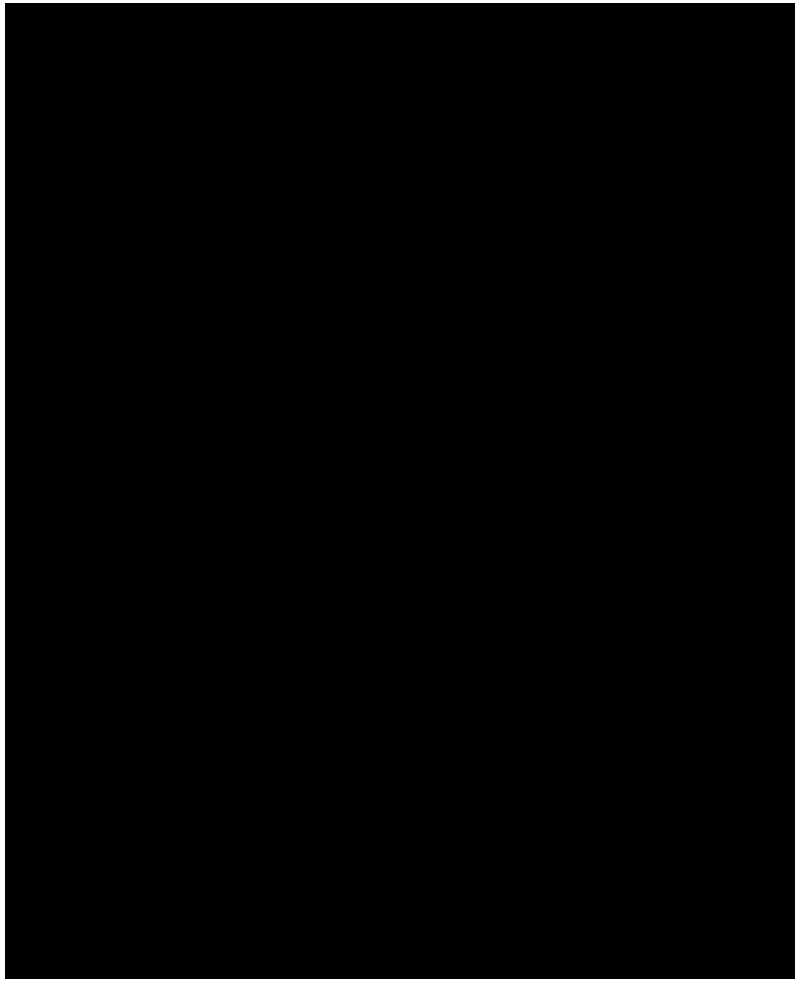
© Terasoft, Inc.

Deep Reinforcement Learning (E.g. Deep Q Network)

- Policy is a sequence of actions to observations to get maximum reward
- Reinforcement Learning finds the optimal policy maximizing the reward
- Reinforcement Learning adapts to changes in environment by improving the policy
- No need for explicit model (model-free)



© Terasoft, Inc.



Google Deepmind’s Deep Q Learning playing Atari Breakout

© Terasoft, Inc.

Examples



Speech recognition

Type: sequence to sequence
Input: Signal features
Output: Phrases
Network: RNN/LSTM

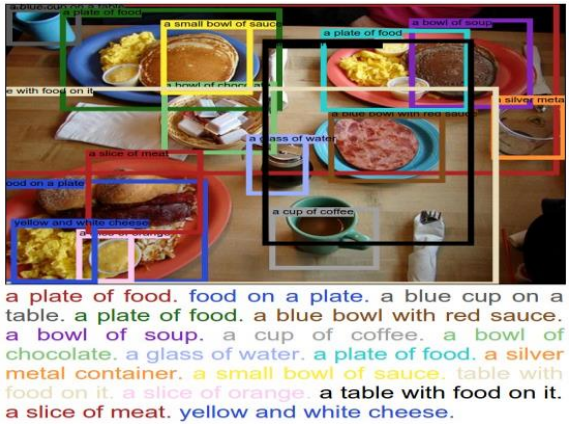


Image captioning

Type: one to sequence
Input: Images
Output: Captions
Network: CNN/RNN

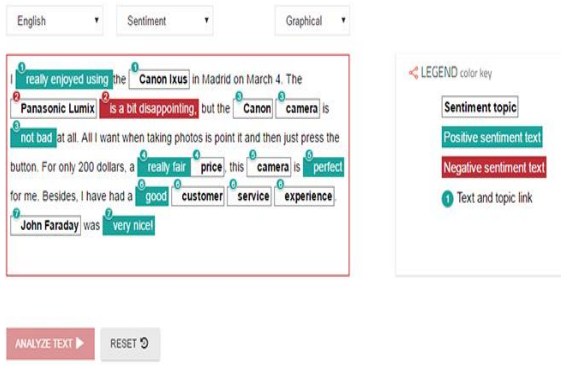


Financial Trading

Type: sequence to one
Input: Time series data
Output: Buy/Sell
Network: RNN/LSTM

© Terasoft, Inc.

Examples



Sentiment Analysis



Control Application

Type: sequence to one
Input: Text
Output: Phrases
Network: RNN/LSTM

Type: one to sequence
Input: Agent+Rules
Output: Policy
Network: CNN/RNN/DQN

© Terasoft, Inc.

Deep Learning Landscape

Deep Learning Framework: Software used to train a deep neural network.

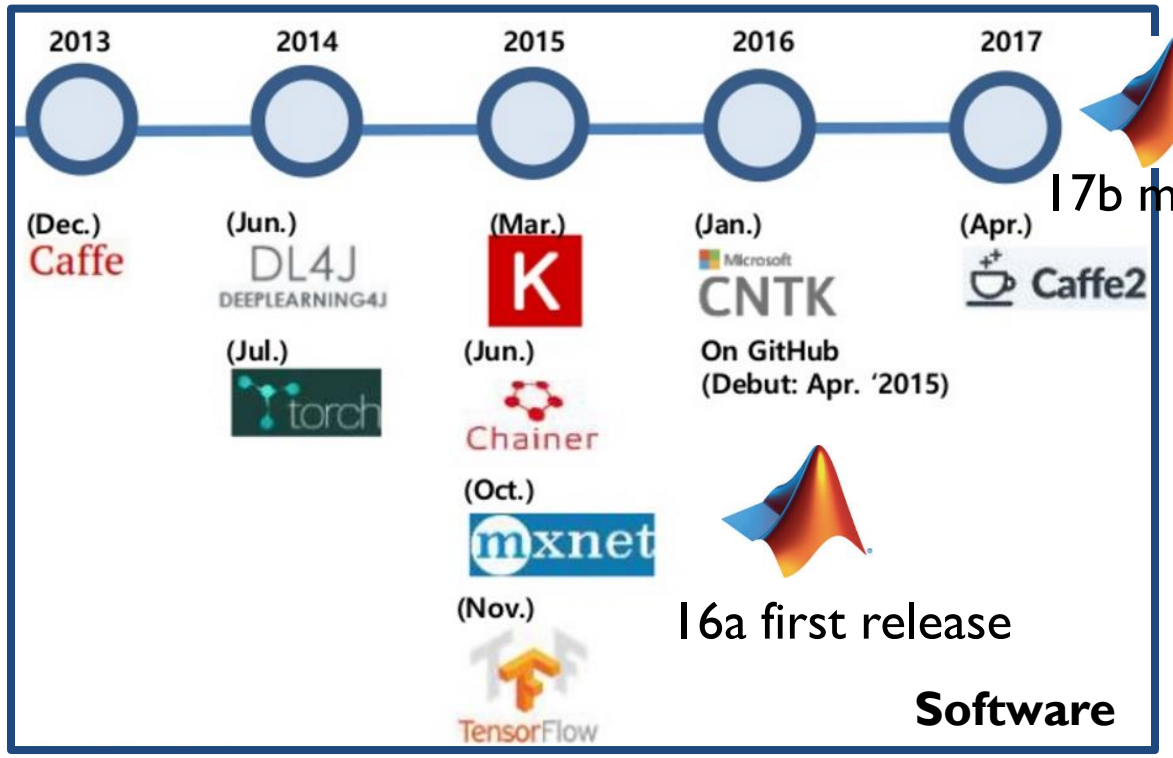
- E.g. MATLAB, TensorFlow

GPU Acceleration

NVIDIA

Hardware

+



© Terasoft, Inc.

Deep Learning Landscape

Deep Learning Framework: Software used to train a deep neural network.

- E.g. MATLAB, TensorFlow

Top 5 Frameworks by Popularity

1. TensorFlow (big margin)
2. Caffe
3. Torch
4. CNTK
5. MxNet

2016 2017

17b mega-release

(Jan.) (Ap

Microsoft CNTK

Caffe2

GPU Acceleration

NVIDIA

Hardware

ase

Software

Call to Action

"MATLAB vs Tensorflow" Tech Forum tomorrow

Handout available at back of room.

MATLAB / Tensorflow-Keras Comparison

Access Data

Capabilities	MATLAB	Tensorflow-Keras	Winner
Accessing large datasets	+ imageDatastore + datastore + miniBatchDatastore *new 18a*	+ datagenerator used in Keras	Keras MATLAB

Preprocessing

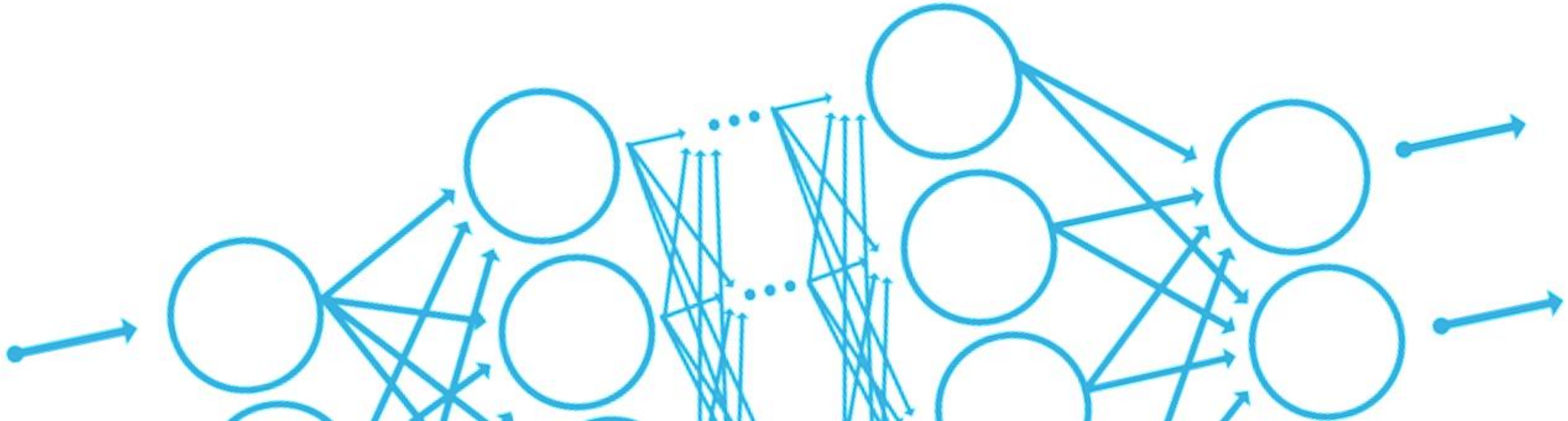
Capabilities	MATLAB	Tensorflow-Keras	Winner
Ground Truth Labeling	+ ImageLabeler – app for semantic segmentation and image classification	Δ None	Keras MATLAB
Access Pretrained Models	Δ Less Models Supported + Most popular models supported (Inception-v3, ResNet-101, VGG-16, etc.) + One line of code to access models	+ Supports more models + Open Source repositories for additional models	Keras MATLAB

© Terasoft, Inc.

Deep learning is a type of machine learning in which a model learns to perform classification tasks directly from images, text, or sound.

Deep learning is usually implemented using a **neural network architecture.**

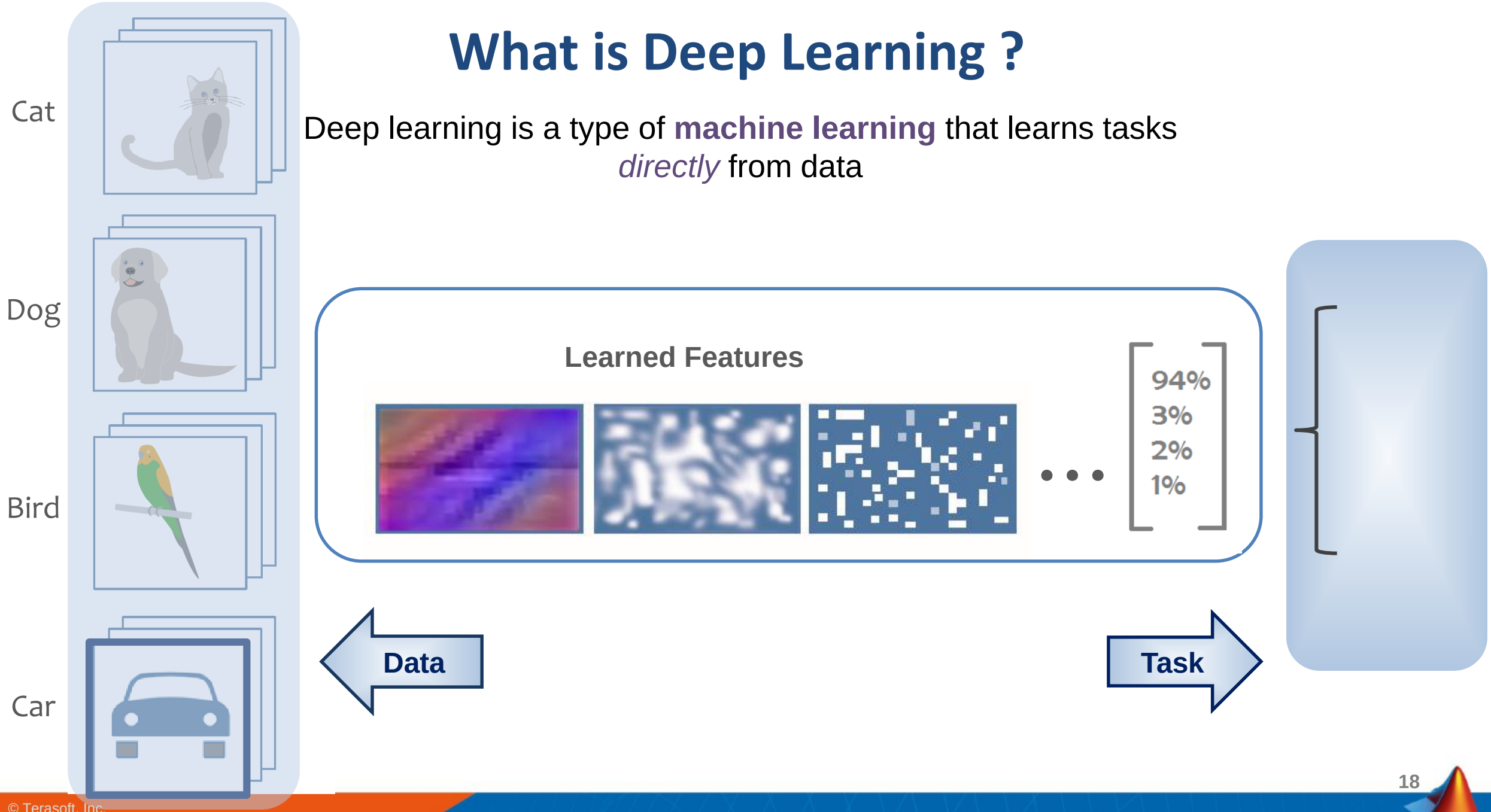
The term “deep” refers to the **number of layers** in the network—the more layers, the deeper the network.



© Terasoft, Inc.

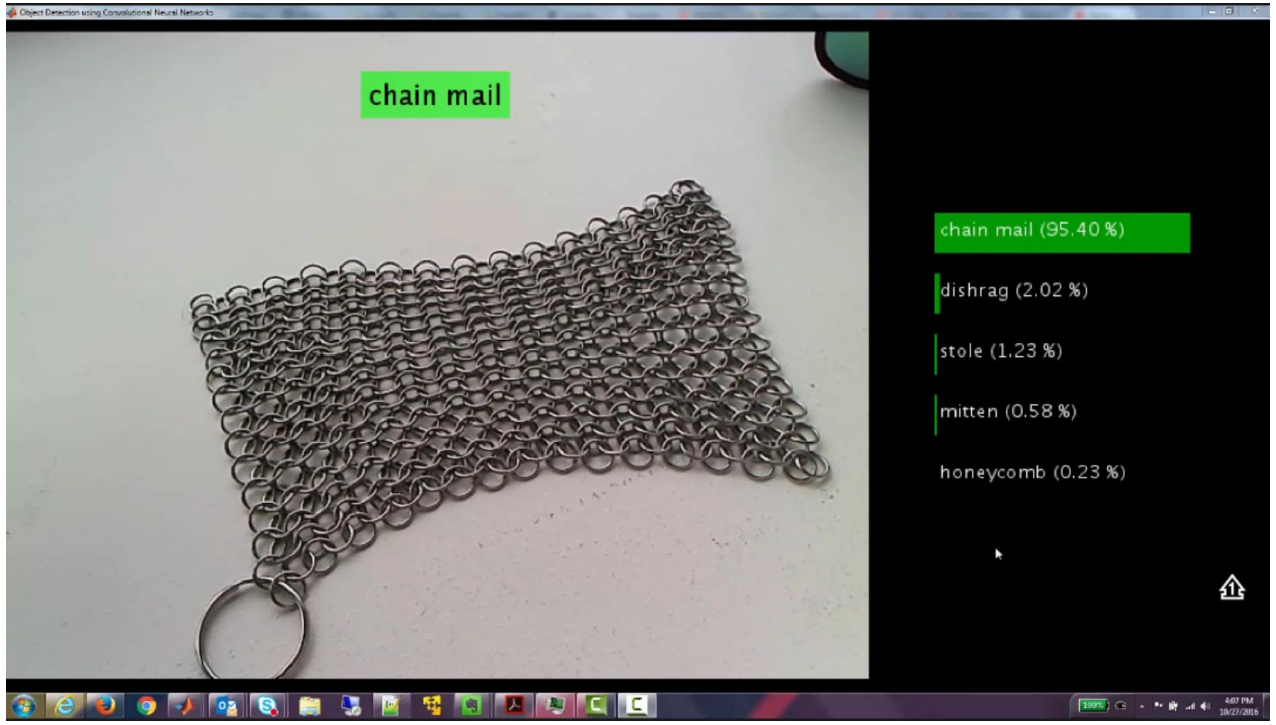
What is Deep Learning ?

Deep learning is a type of **machine learning** that learns tasks *directly* from data



© Terasoft, Inc.

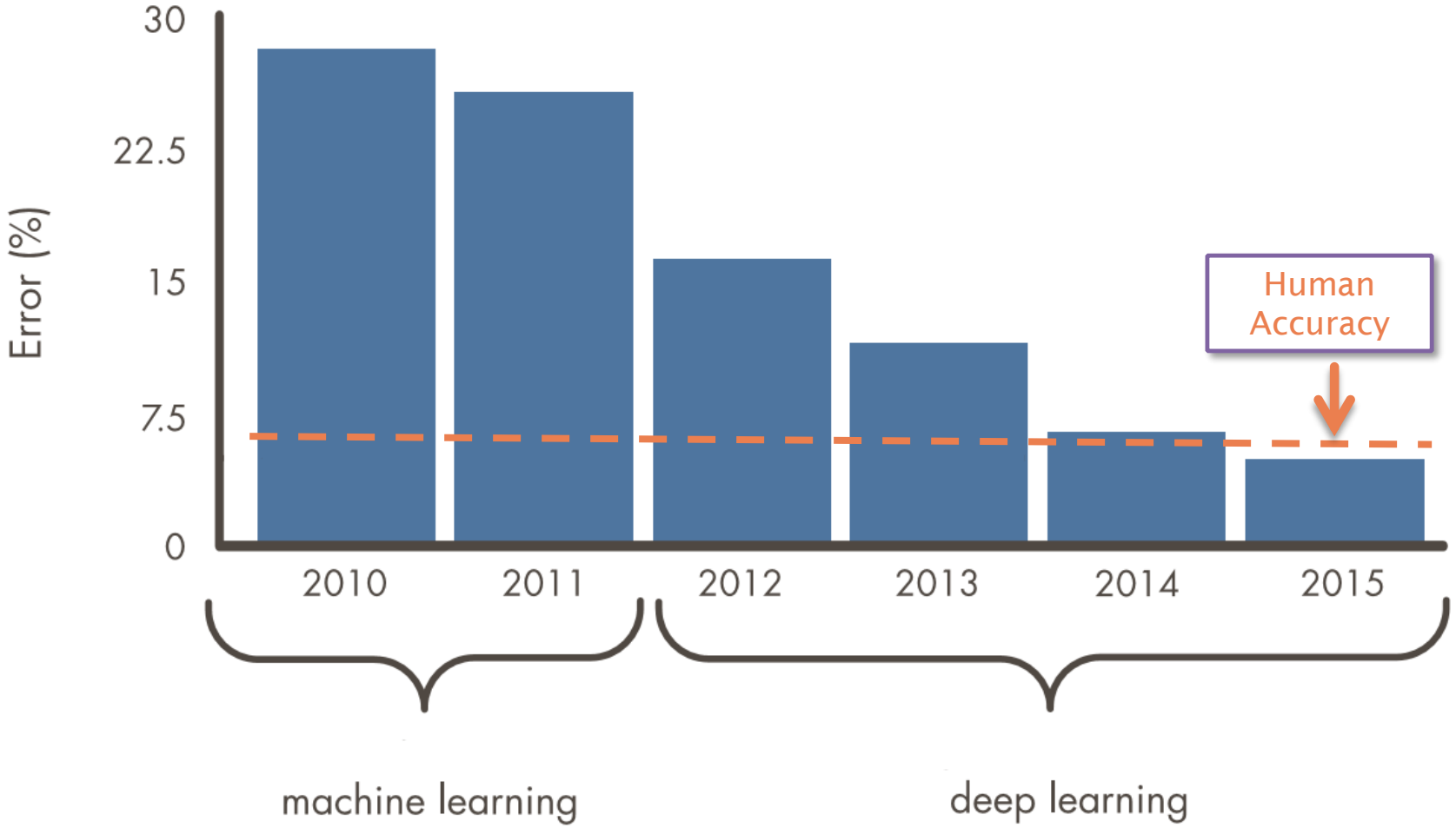
Object Recognition Using Deep Learning



Training (GPU)	Millions of images from 1000 different categories
Prediction	Real-time object recognition using a webcam connected to a laptop

© Terasoft, Inc.

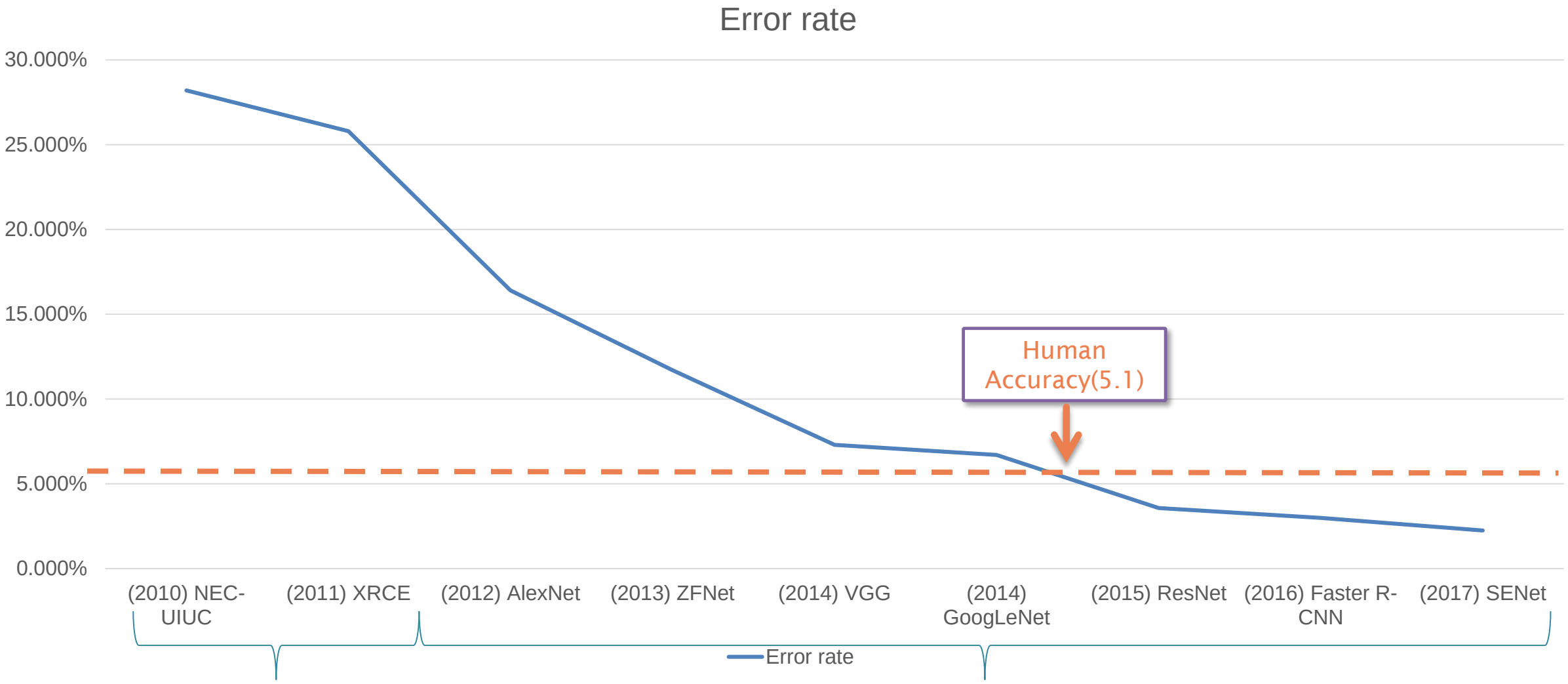
Why is Deep Learning So Popular Now?



Source: ILSVRC Top-5 Error on ImageNet

© Terasoft, Inc.

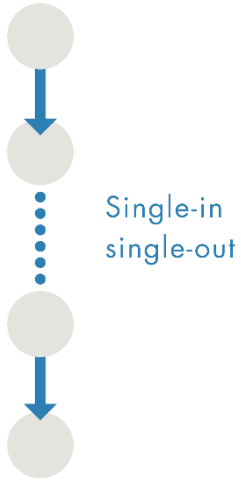
Why is Deep Learning So Popular Now?



© Terasoft, Inc.

Other deep neural networks

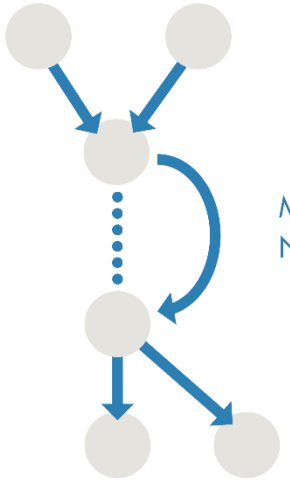
SeriesNetwork



Single-in
single-out

Networks:
- MNIST
- Alexnet
- VGG
- Lane detection
- Pedestrian detection

DAGNetwork



R2017b

Multi-in, multi-out
No feedback loops

Networks:
- R-CNN (fast, faster) — *Object detection*
- GoogLeNet
- ResNet
- SegNet — *Semantic segmentation*
- FCN
- DeconvNet

Recurrent Network



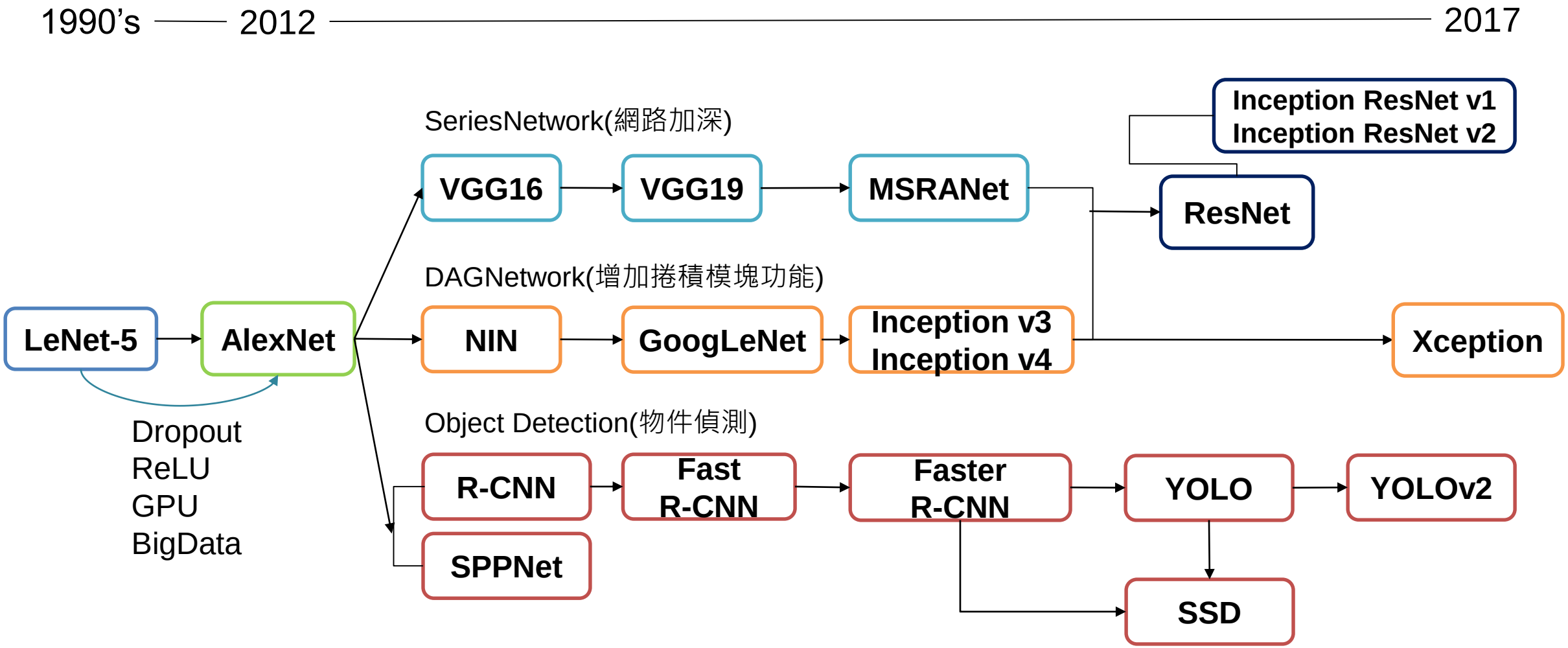
R2017b

Memory

Networks:
- LSTM

© Terasoft, Inc.

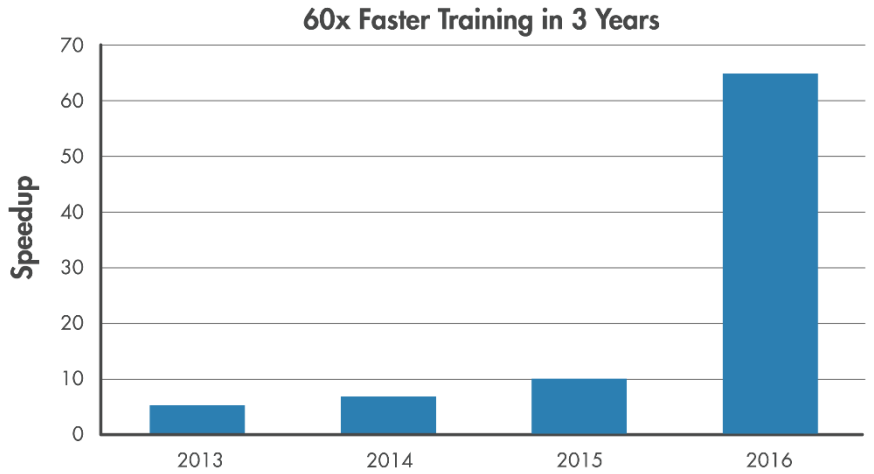
Deep Learning - CNN Architecture diagram



© Terasoft, Inc.

Deep Learning Enablers

- Increased GPU acceleration



- World-class models

AlexNet
PRETRAINED MODEL

VGG-16
PRETRAINED MODEL

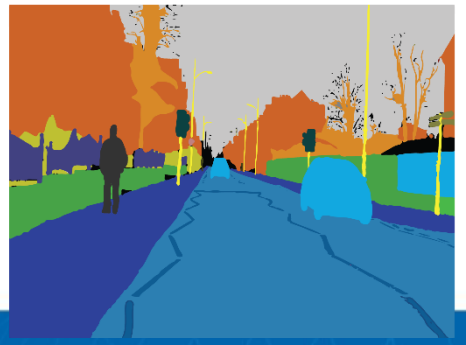
ResNet
PRETRAINED MODEL

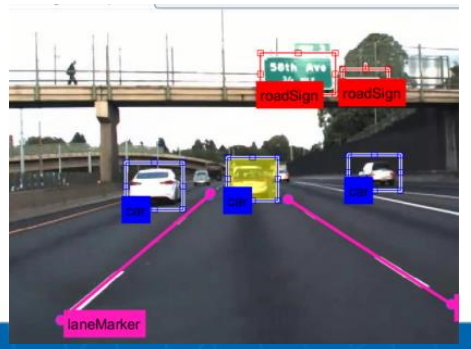
Caffe
MODELS

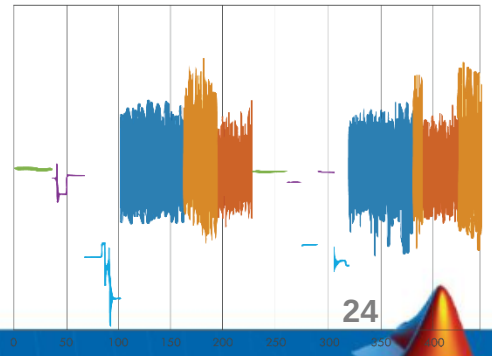
GoogLeNet
PRETRAINED MODEL

TensorFlow/Keras
MODELS

- Labeled public datasets



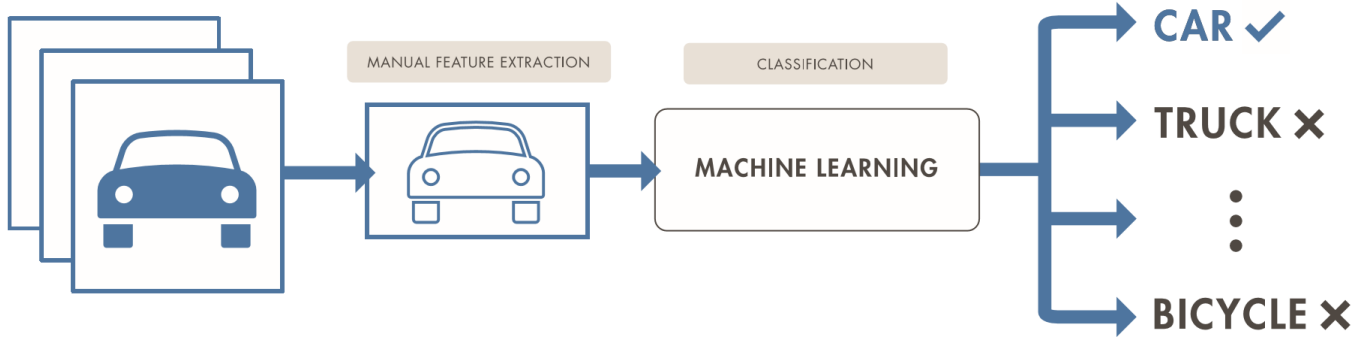




© Terasoft, Inc.

Deep Learning is a Subset of Machine Learning

Machine Learning



優點:
資料量需求較少
客製化特徵提取
可以選擇不同的機器學習

缺點:
單一模型泛用性低
需要較多影像處理與電腦視覺的知識

Deep Learning



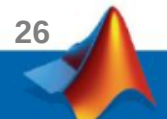
優點:
單一模型泛用性高
需要較少影像處理知識
多領域內辨識率較好

缺點:
資料量需求大
訓練時間較長
較難調參數

© Terasoft, Inc.

Agenda

- Deep Learning with MATLAB
  - What is Deep Learning?
  - Layers in Convolution Neural Network
  - Image classification using pre-trained network
  - Train a new model
  - Object detection



© Terasoft, Inc.

What are Neural Networks?

A model modelled on the human brain and nervous system

Neurons

- Form network via axons
- Relaying information

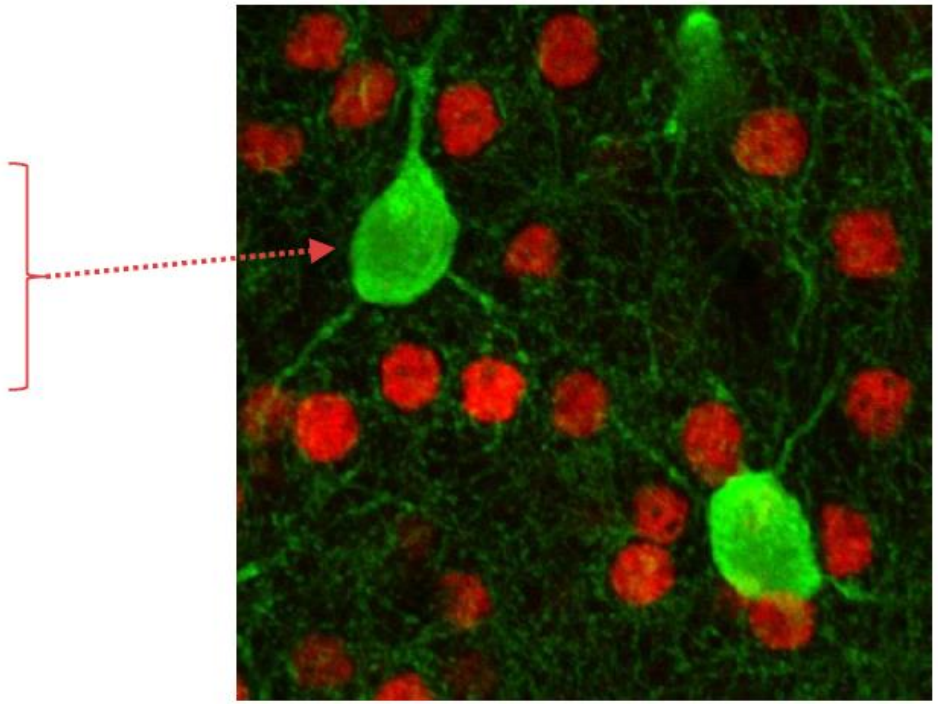
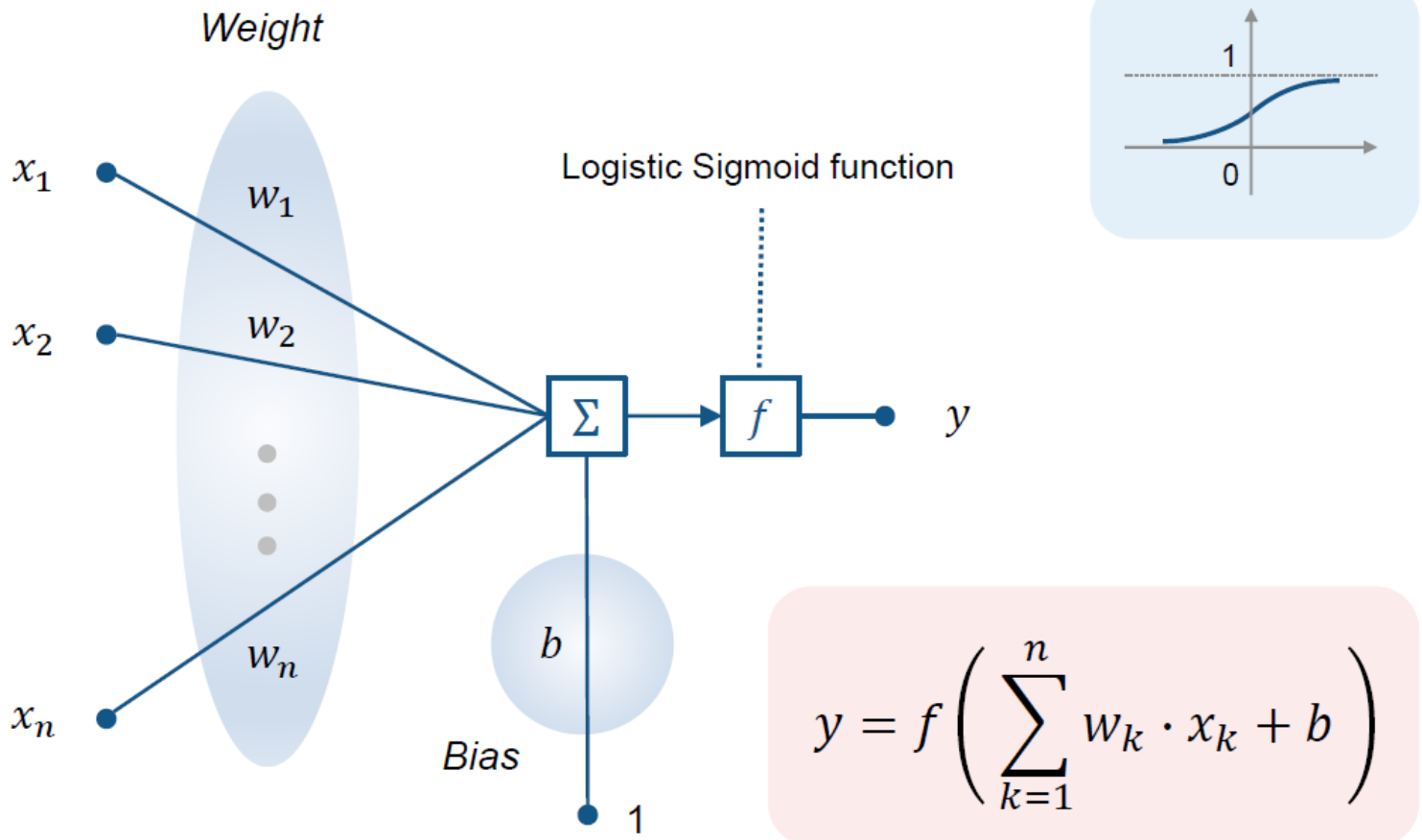


Image from Riken

© Terasoft, Inc.

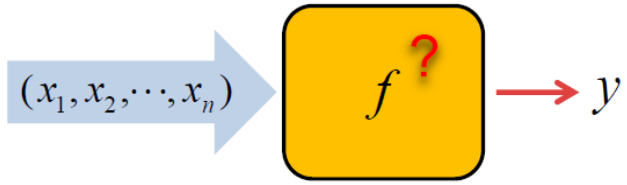
What are Neural Networks?



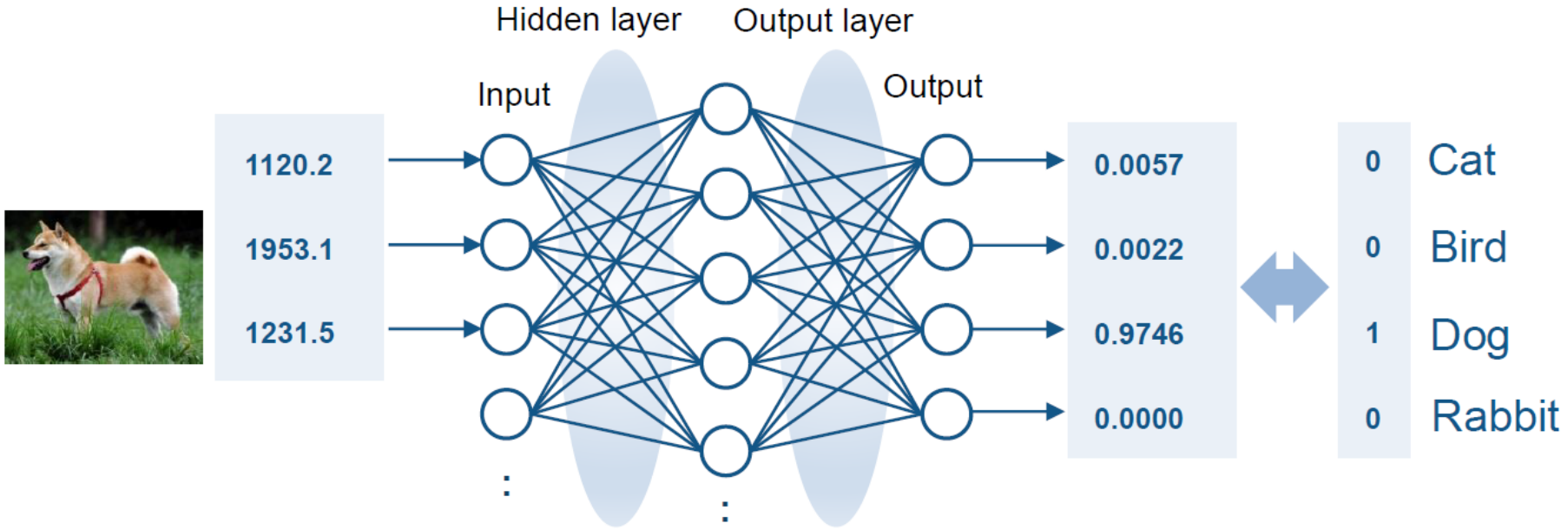
$$y = f\left(\sum_{k=1}^{n} w_k \cdot x_k + b\right)$$

© Terasoft, Inc.

How Neural Networks Work



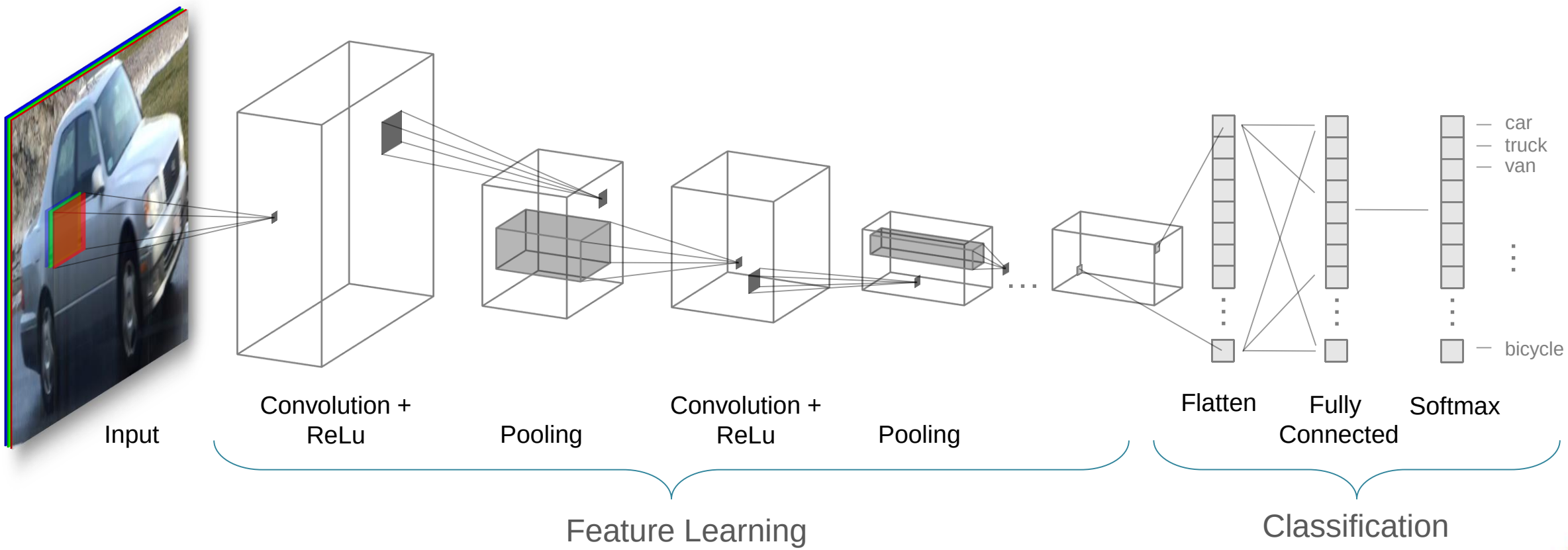
- Example: Networks with N inputs and 4 outputs



© Terasoft, Inc.

Deep Learning:
Convolutional Neural Networks (CNN/ConvNet)

- Multi-layered neural network



© Terasoft, Inc.

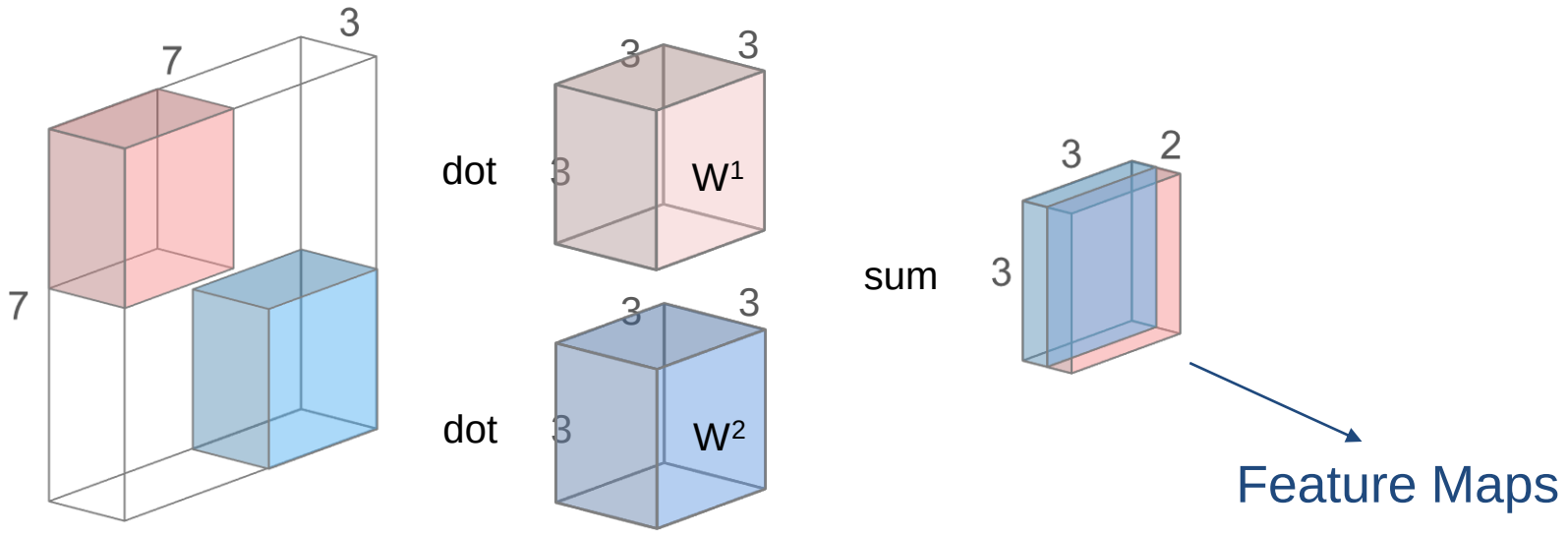
Common Convolutional Neural Networks Layers

- Convolution Layer
- Rectified Linear Unit (ReLU) Layer
- Pooling Layer
- Fully Connected Layer
- Dropout Layer
- Softmax Layer
- Classification Layer

© Terasoft, Inc.

Convolution Layer

- Core building block of a CNN
- Convolve the filters sliding them across the input, computing the dot product

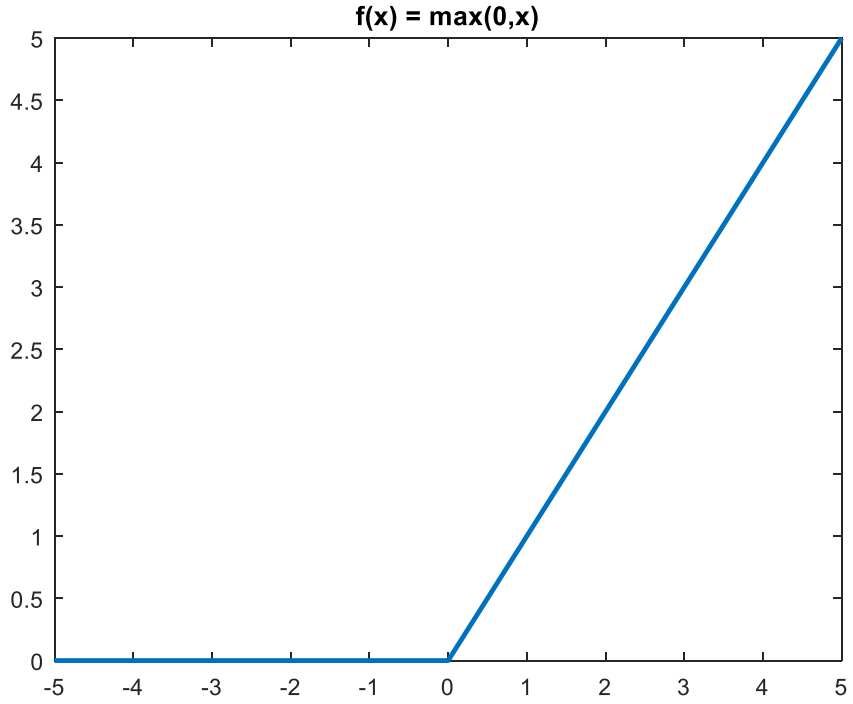


- Intuition: learn filters that activate when they “see” some specific feature

© Terasoft, Inc.

Rectified Linear Unit (ReLU) Layer

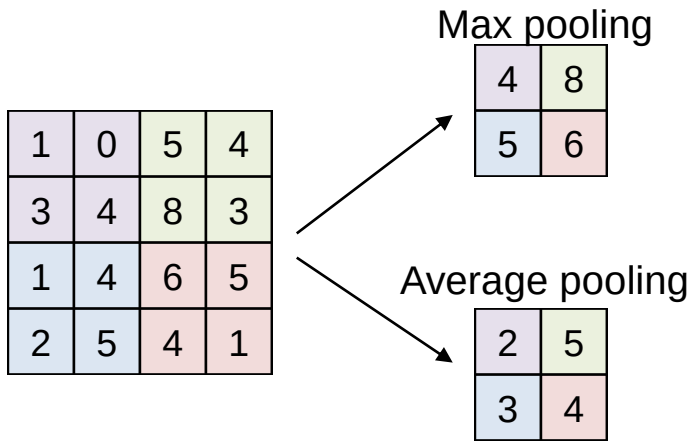
- Frequently used in combination with Convolution layers
- Allows for faster and more effective training
- $\boldsymbol{f(x) = max(0, x)}$, activation is thresholded at 0



© Terasoft, Inc.

Pooling Layer

- Perform a downsampling operation across the spatial dimensions
- Goal: reducing the number of parameters
- Max pooling and average pooling methods



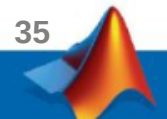
© Terasoft, Inc.

Convolution

0	0	0	0	0	0	0	0	0	0
0	-8	-7	-9	4	4	-8	8	2	0
0	6	-1	3	-5	-7	0	9	-2	0
0	5	0	-9	-6	2	-3	3	1	0
0	8	3	-2	4	3	-8	1	2	0
0	0	-7	6	1	3	2	5	-3	0
0	3	9	4	0	9	7	2	6	0
0	-1	-5	-4	2	5	6	-4	-9	0
0	7	-2	8	9	1	2	-7	5	0
0	0	0	0	0	0	0	0	0	0

1	0	0
0	1	0
0	0	1

© Terasoft, Inc.

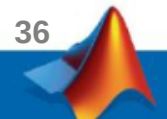


Convolution

0	0	0	0	0	0	0	0	0	0
0	-8	-7	-9	4	4	-8	8	2	0
0	6	-1	3	-5	-7	0	9	-2	0
0	5	0	-9	-6	2	-3	3	1	0
0	8	3	-2	4	3	-8	1	2	0
0	0	-7	6	1	3	2	5	-3	0
0	3	9	4	0	9	7	2	6	0
0	-1	-5	-4	2	5	6	-4	-9	0
0	7	-2	8	9	1	2	-7	5	0
0	0	0	0	0	0	0	0	0	0

1	0	0
0	1	0
0	0	1

© Terasoft, Inc.



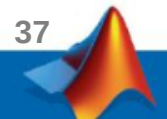
Convolution

0	0	0	0	0	0	0	0	0	0
0	-8	-7	-9	4	4	-8	8	2	0
0	6	-1	3	-5	-7	0	9	-2	0
0	5	0	-9	-6	2	-3	3	1	0
0	8	3	-2	4	3	-8	1	2	0
0	0	-7	6	1	3	2	5	-3	0
0	3	9	4	0	9	7	2	6	0
0	-1	-5	-4	2	5	6	-4	-9	0
0	7	-2	8	9	1	2	-7	5	0
0	0	0	0	0	0	0	0	0	0

1	0	0
0	1	0
0	0	1

-9							

© Terasoft, Inc.



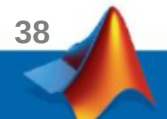
Convolution

0	0	0	0	0	0	0	0	0	0
0	-8	-7	-9	4	4	-8	8	2	0
0	6	-1	3	-5	-7	0	9	-2	0
0	5	0	-9	-6	2	-3	3	1	0
0	8	3	-2	4	3	-8	1	2	0
0	0	-7	6	1	3	2	5	-3	0
0	3	9	4	0	9	7	2	6	0
0	-1	-5	-4	2	5	6	-4	-9	0
0	7	-2	8	9	1	2	-7	5	0
0	0	0	0	0	0	0	0	0	0

1	0	0
0	1	0
0	0	1

-9	-4						

© Terasoft, Inc.



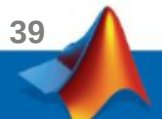
Convolution

0	0	0	0	0	0	0	0	0	0
0	-8	-7	-9	4	4	-8	8	2	0
0	6	-1	3	-5	-7	0	9	-2	0
0	5	0	-9	-6	2	-3	3	1	0
0	8	3	-2	4	3	-8	1	2	0
0	0	-7	6	1	3	2	5	-3	0
0	3	9	4	0	9	7	2	6	0
0	-1	-5	-4	2	5	6	-4	-9	0
0	7	-2	8	9	1	2	-7	5	0
0	0	0	0	0	0	0	0	0	0

1	0	0
0	1	0
0	0	1

-9	-4	-14	3	4	1	6	2
6	-18	-10	-12	-6	7	2	6
8	4	-6	-6	-11	-9	5	10
1	14	-1	-2	-7	-1	-5	5
9	5	9	8	14	1	3	-2
-2	-5	-1	11	16	6	-5	11
-3	6	14	7	7	8	8	-7
7	-3	3	5	3	7	-1	1

© Terasoft, Inc.



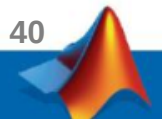
Convolution

0	0	0	0	0	0	0	0	0	0
0	-8	-7	-9	4	4	-8	8	2	0
0	6	-1	3	-5	-7	0	9	-2	0
0	5	0	-9	-6	2	-3	3	1	0
0	8	3	-2	4	3	-8	1	2	0
0	0	-7	6	1	3	2	5	-3	0
0	3	9	4	0	9	7	2	6	0
0	-1	-5	-4	2	5	6	-4	-9	0
0	7	-2	8	9	1	2	-7	5	0
0	0	0	0	0	0	0	0	0	0

1	0	0
0	1	0
0	0	1

-9	-4	-14	3	4	1	6	2
6	-18	-10	-12	-6	7	2	6
8	4	-6	-6	-11	-9	5	10
1	14	-1	-2	-7	-1	-5	5
9	5	9	8	14	1	3	-2
-2	-5	-1	11	16	6	-5	11
-3	6	14	7	7	8	8	-7
7	-3	3	5	3	7	-1	1

© Terasoft, Inc.



Activation

-9	-4	-14	3	4	1	6	2
6	-18	-10	-12	-6	7	2	6
8	4	-6	-6	-11	-9	5	10
1	14	-1	-2	-7	-1	-5	5
9	5	9	8	14	1	3	-2
-2	-5	-1	11	16	6	-5	11
-3	6	14	7	7	8	8	-7
7	-3	3	5	3	7	-1	1



-9	-4	-14	3	4	1	6	2
6	-18	-10	-12	-6	7	2	6
8	4	-6	-6	-11	-9	5	10
1	14	-1	-2	-7	-1	-5	5
9	5	9	8	14	1	3	-2
-2	-5	-1	11	16	6	-5	11
-3	6	14	7	7	8	8	-7
7	-3	3	5	3	7	-1	1

© Terasoft, Inc.

Activation

-9	-4	-14	3	4	1	6	2
6	-18	-10	-12	-6	7	2	6
8	4	-6	-6	-11	-9	5	10
1	14	-1	-2	-7	-1	-5	5
9	5	9	8	14	1	3	-2
-2	-5	-1	11	16	6	-5	11
-3	6	14	7	7	8	8	-7
7	-3	3	5	3	7	-1	1



0	0	0	3	4	1	6	2
6	0	0	0	0	7	2	6
8	4	0	0	0	0	5	10
1	14	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	11	16	6	0	11
0	6	14	7	7	8	8	0
7	0	3	5	3	7	0	1

© Terasoft, Inc.

Activation

-9	-4	-14	3	4	1	6	2
6	-18	-10	-12	-6	7	2	6
8	4	-6	-6	-11	-9	5	10
1	14	-1	-2	-7	-1	-5	5
9	5	9	8	14	1	3	-2
-2	-5	-1	11	16	6	-5	11
-3	6	14	7	7	8	8	-7
7	-3	3	5	3	7	-1	1



0	0	0	3	4	1	6	2
6	0	0	0	0	7	2	6
8	4	0	0	0	0	5	10
1	14	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	11	16	6	0	11
0	6	14	7	7	8	8	0
7	0	3	5	3	7	0	1

© Terasoft, Inc.

Activation

-9	-4	-14	3	4	1	6	2
6	-18	-10	-12	-6	7	2	6
8	4	-6	-6	-11	-9	5	10
1	14	-1	-2	-7	-1	-5	5
9	5	9	8	14	1	3	-2
-2	-5	-1	11	16	6	-5	11
-3	6	14	7	7	8	8	-7
7	-3	3	5	3	7	-1	1



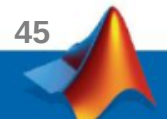
0	0	0	3	4	1	6	2
6	0	0	0	0	7	2	6
8	4	0	0	0	0	5	10
1	14	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	11	16	6	0	11
0	6	14	7	7	8	8	0
7	0	3	5	3	7	0	1

© Terasoft, Inc.

Pooling

0	0	0	3	4	1	6	2
6	0	0	0	0	7	2	6
8	4	0	0	0	0	5	10
1	14	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	11	16	6	0	11
0	6	14	7	7	8	8	0
7	0	3	5	3	7	0	1

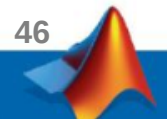
© Terasoft, Inc.



Pooling

0	0	0	3	4	1	6	2
6	0	0	0	0	7	2	6
8	4	0	0	0	0	5	10
1	14	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	11	16	6	0	11
0	6	14	7	7	8	8	0
7	0	3	5	3	7	0	1

© Terasoft, Inc.

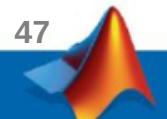


Pooling

0	0	0	3	4	1	6	2
6	0	0	0	0	7	2	6
8	4	0	0	0	0	5	10
1	14	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	11	16	6	0	11
0	6	14	7	7	8	8	0
7	0	3	5	3	7	0	1

6			

© Terasoft, Inc.

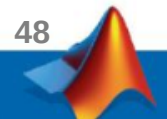


Pooling

0	0	0	**3**	4	1	6	2
6	0	0	0	0	7	2	6
8	4	0	0	0	0	5	10
1	14	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	11	16	6	0	11
0	6	14	7	7	8	8	0
7	0	3	5	3	7	0	1

6	**3**		

© Terasoft, Inc.

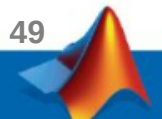


Pooling

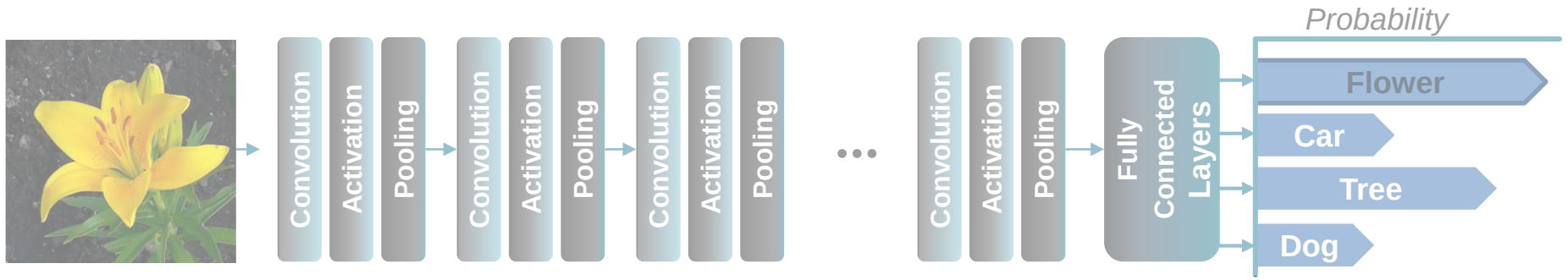
0	0	0	**3**	4	1	**6**	2
6	0	0	0	0	**7**	2	6
8	4	0	**0**	**0**	0	5	**10**
1	**14**	0	0	0	0	0	5
9	5	9	8	14	1	3	0
0	0	0	**11**	**16**	6	0	**11**
0	6	**14**	7	7	**8**	**8**	0
7	0	3	5	3	7	0	1

6	**3**	**7**	**6**
14	**0**	**0**	**10**
9	**11**	**16**	**11**
7	**14**	**8**	**8**

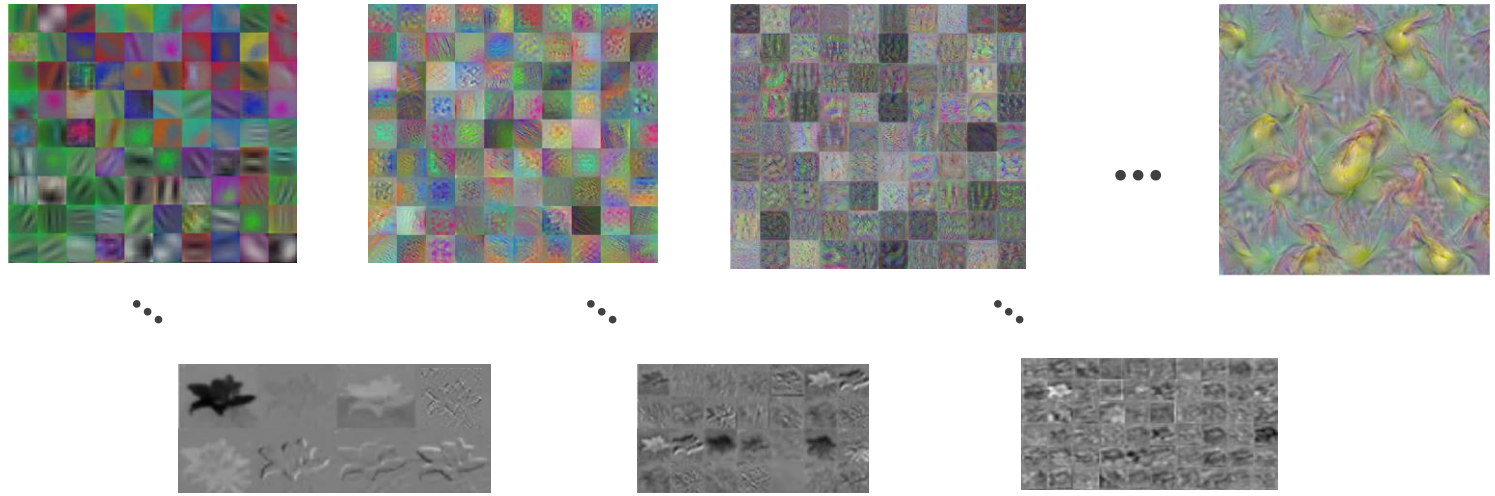
© Terasoft, Inc.



CNN – Feature Learning



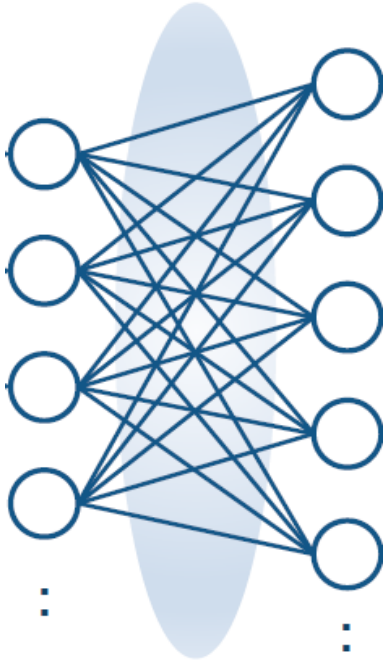
Filters



© Terasoft, Inc.

Fully Connected Layer

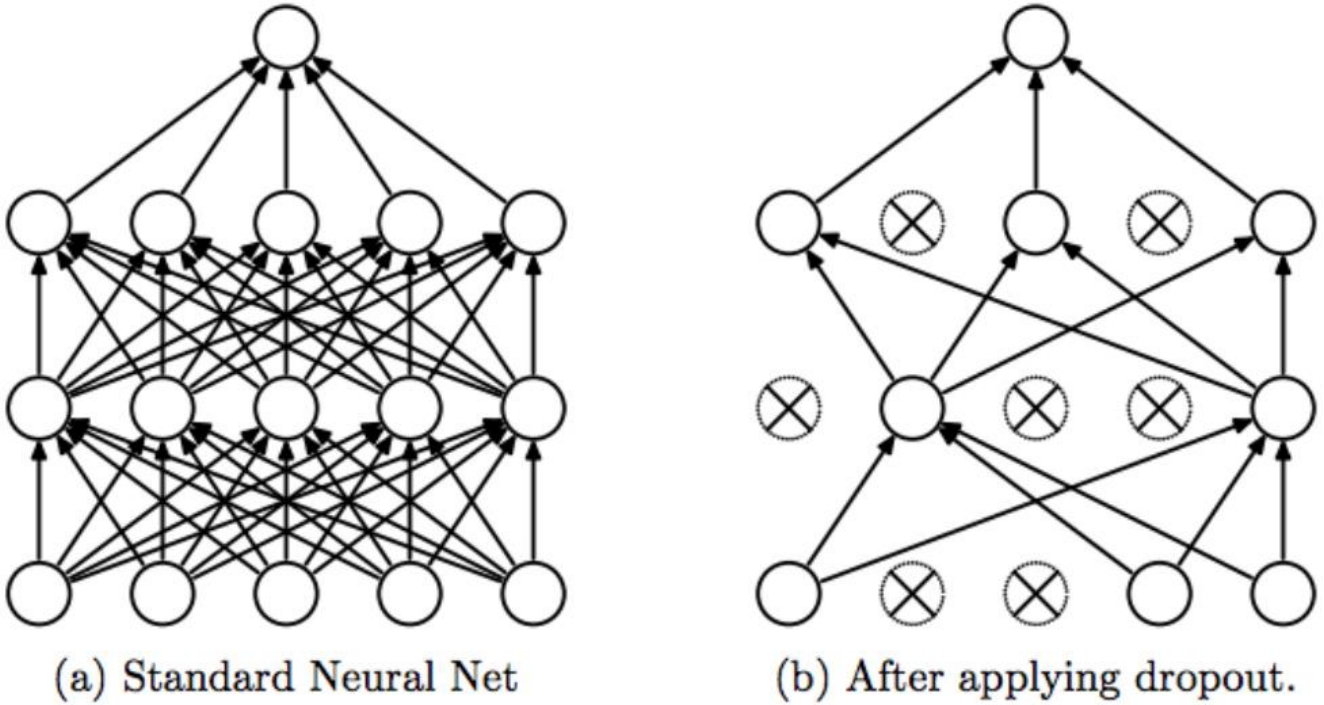
- This layer combines all of the features (local information) learned by the previous layers across the image to identify the larger patterns.



© Terasoft, Inc.

Dropout Layer

- Dropout consists in randomly setting a fraction rate of input units to 0 at each update during training time, which helps prevent overfitting



(a) Standard Neural Net

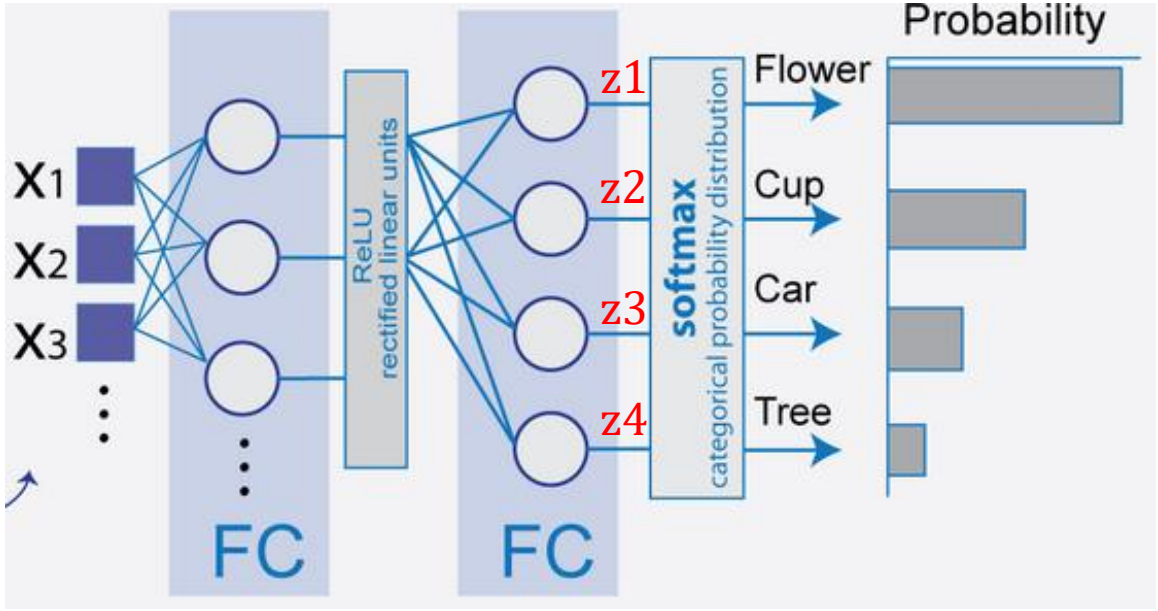
(b) After applying dropout.

Src: http://www.jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf

© Terasoft, Inc.

Softmax Layer

- Categorical probability distribution.
- Normalized exponential function.



$$y_1 = \frac{e^{z_1}}{\sum_{k=1}^{4} e^{z_k}}$$

$$y_2 = \frac{e^{z_2}}{\sum_{k=1}^{4} e^{z_k}}$$

$$y_3 = \frac{e^{z_3}}{\sum_{k=1}^{4} e^{z_k}}$$

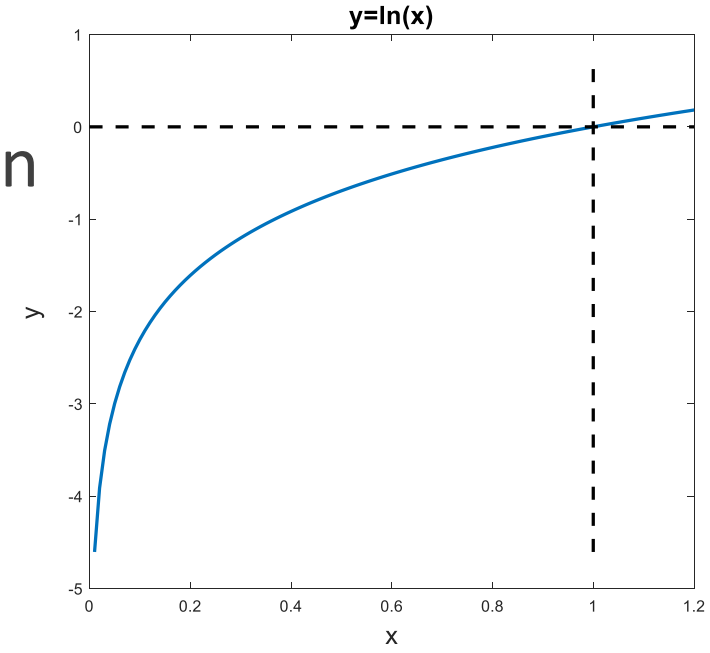
$$y_4 = \frac{e^{z_4}}{\sum_{k=1}^{4} e^{z_k}}$$

© Terasoft, Inc.

Classification Layer

- The loss (error) : the cross entropy function

$$E(\theta) = -\sum_{i=1}^{n}\sum_{j=1}^{k} t_{ij} \ln y_j(x_i, \theta),$$



Label index
(answer)

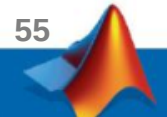
$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \qquad \begin{bmatrix} 0.2 \\ 0.5 \\ 0.1 \\ 0.1 \\ \vdots \end{bmatrix}$$

Value from the softmax layer
(predicted)

© Terasoft, Inc.

Agenda

- Deep Learning with MATLAB
  - What is Deep Learning?
  - Layers in Convolution Neural Network
  - Image classification using pre-trained network
  - Train a new model
  - Object detection



© Terasoft, Inc.

Image Classification Using Pre-trained Network

```
%% Load Pre-trained CNN
% This network is available from out add-ons location in the home tab.
% You can use a variety of pretrained networks available for download
net = alexnet;
% Show the architecture of AlexNet
net.Layers

%% Classify 'peppers' in 4 lines of code
% import a testing image.
img = imread('peppers.png');
imshow(img);

% There is a size requirement of 227 x 227 for AlexNet.
img = imresize(img, [227 227]);

% Recognize the testing image
classify(net, img)

```



```
25x1 Layer array with layers:

     1   'data'     Image Input                   227x227x3 images with 'zerocenter' normalization
     2   'conv1'    Convolution                   96 11x11x3 convolutions with stride [4  4] and padding [0  0]
     3   'relu1'    ReLU                          ReLU
     4   'norm1'    Cross Channel Normalization   cross channel normalization with 5 channels per element
     5   'pool1'    Max Pooling                   3x3 max pooling with stride [2  2] and padding [0  0]
     6   'conv2'    Convolution                   256 5x5x48 convolutions with stride [1  1] and padding [2  2]
     7   'relu2'    ReLU                          ReLU
     8   'norm2'    Cross Channel Normalization   cross channel normalization with 5 channels per element
     9   'pool2'    Max Pooling                   3x3 max pooling with stride [2  2] and padding [0  0]
    10   'conv3'    Convolution                   384 3x3x256 convolutions with stride [1  1] and padding [1  1]
    11   'relu3'    ReLU                          ReLU
    12   'conv4'    Convolution                   384 3x3x192 convolutions with stride [1  1] and padding [1  1]
    13   'relu4'    ReLU                          ReLU
```


© Terasoft, Inc.

Import the Latest Models for Transfer Learning

Pretrained Models*

- AlexNet
- VGG-16
- VGG-19
- GoogLeNet
- Resnet50
- InceptionV3 (coming soon)

Import Models from Frameworks

- Caffe Model Importer
- TensorFlow/Keras Model Importer

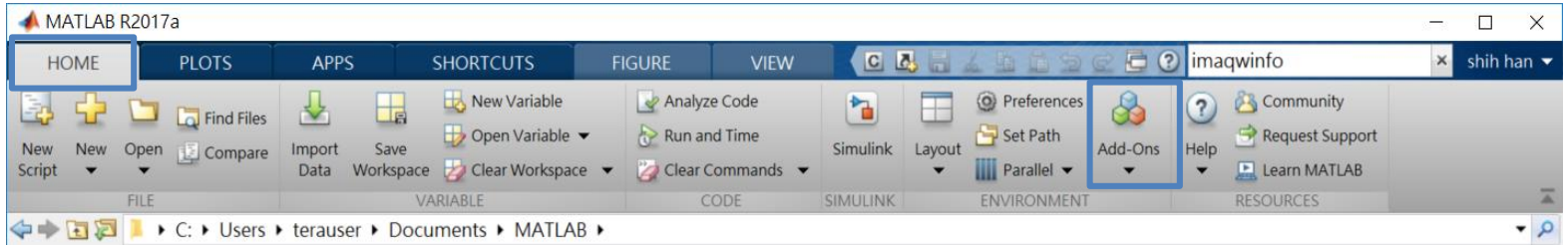


* single line of code to access model

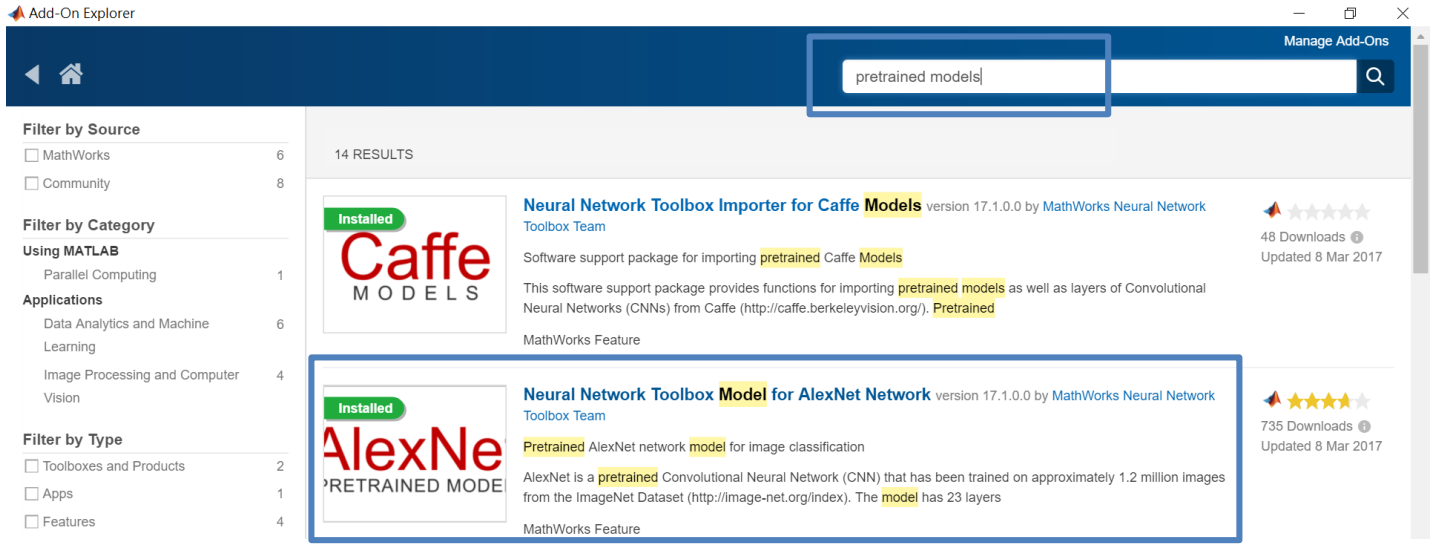
© Terasoft, Inc.

Download and Install Pre-trained Networks

- Go to the **Home** tab and click the **Add-Ons** icon.



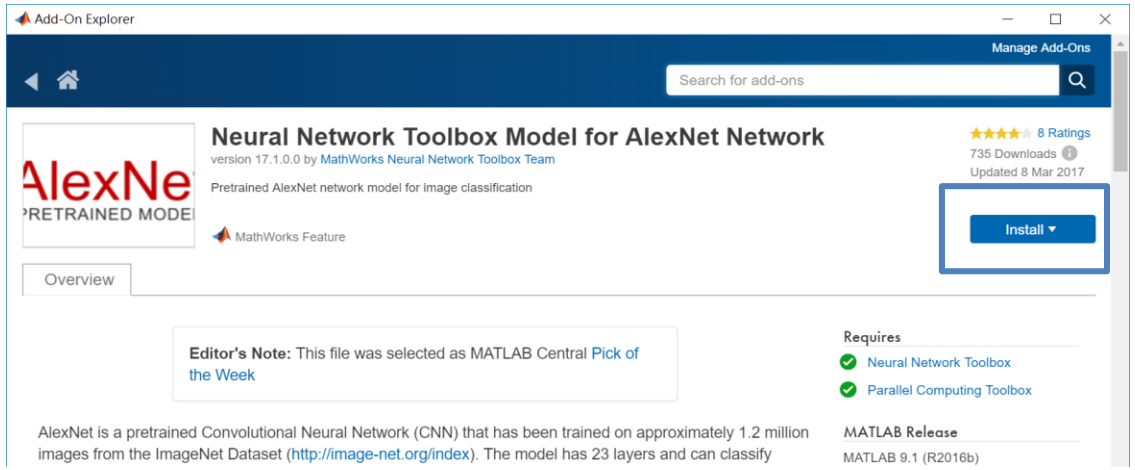
- Search **'pretrained models'.** Click AlexNet add-on to open its detailed information page.



© Terasoft, Inc.

Download and Install Pre-trained Networks

3. Install the add-on.



4. Import the model by typing **net = alexnet** in command window!

```
Command Window
>> net = alexnet

net =

  SeriesNetwork with properties:

    Layers: [25×1 nnet.cnn.layer.Layer]
```

AlexNet PRETRAINED MODEL

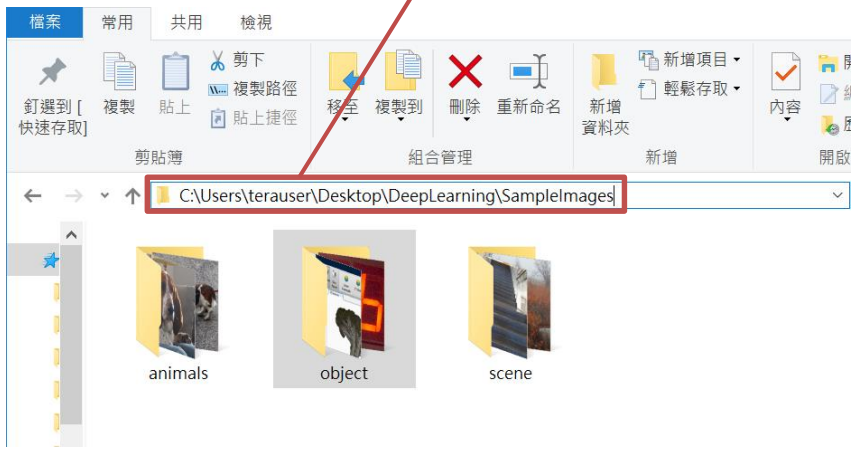
VGG-16 PRETRAINED MODEL

Caffe MODELS

© Terasoft, Inc.

Manage Large Sets of Images

- imds = imageDatastore(ImagesPath, 'IncludeSubfolders', true,…
  'LabelSource','foldernames', 'ReadFcn', @customreader);




```
function Iout = customreader(filename)

    I = imread(filename);

    % Some images may be grayscale. Replicate the image 3 times to
    % create an RGB image.
    if ismatrix(I)
       I = cat(3,I,I,I);
    end

    % Resize the image as required for the CNN.
    Iout = imresize(I, [227 227]);
```

- Methods for ImageDatastore:

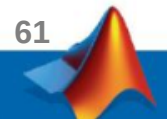
Method	Description
splitEachLabel	Split ImageDatastore labels by proportions
countEachLabel	Count files in ImageDatastore labels
readimage	Read specified image from ImageDatastore

© Terasoft, Inc.

Agenda

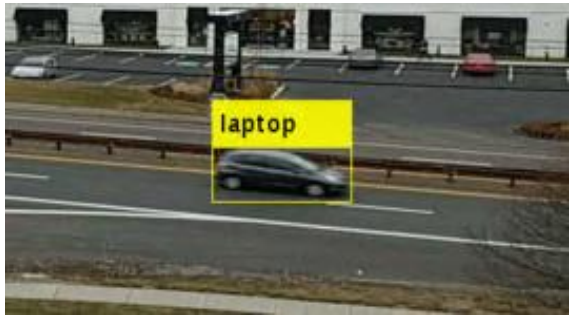
- Deep Learning with MATLAB
  - What is Deep Learning?
  - Layers in Convolution Neural Network
  - Image classification using pre-trained network
  - Train a new model
  - Object detection

© Terasoft, Inc.



Why Train a New Model ?

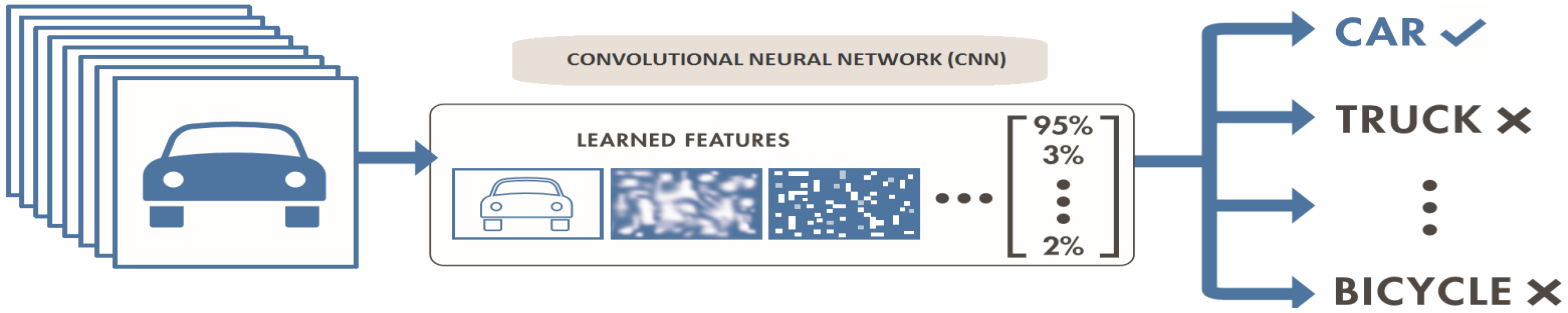
- Models from research do not work on your data
- Pre-trained model not available for your data type
- Improve results by creating a model specific to your problem



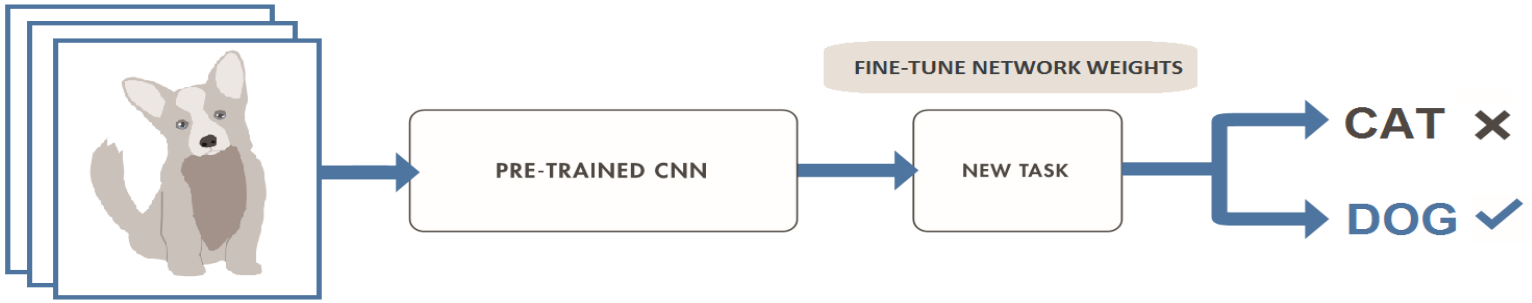
© Terasoft, Inc.

Two Approaches for Training New Model

1. Train a Deep Neural Network from Scratch

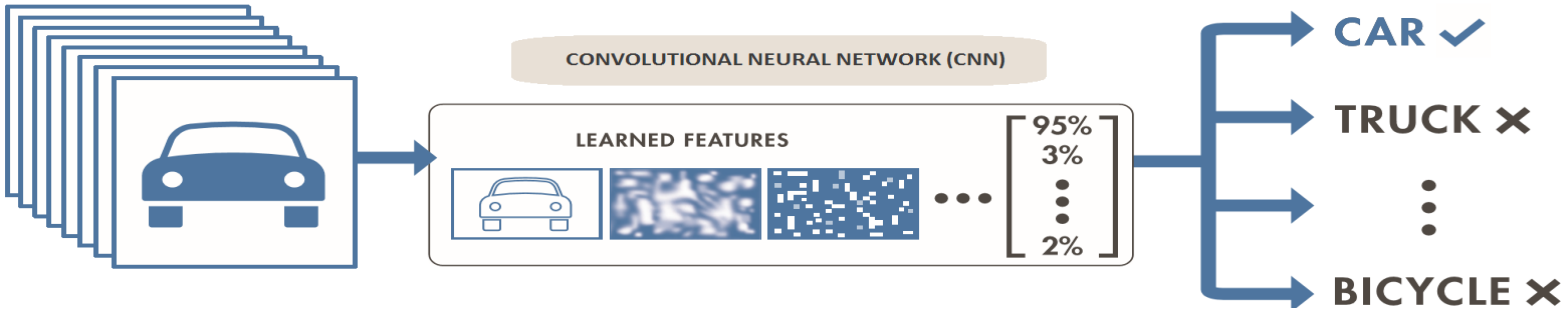


2. Fine-tune a pre-trained model (transfer learning)



© Terasoft, Inc.

Train a Deep Neural Network from Scratch



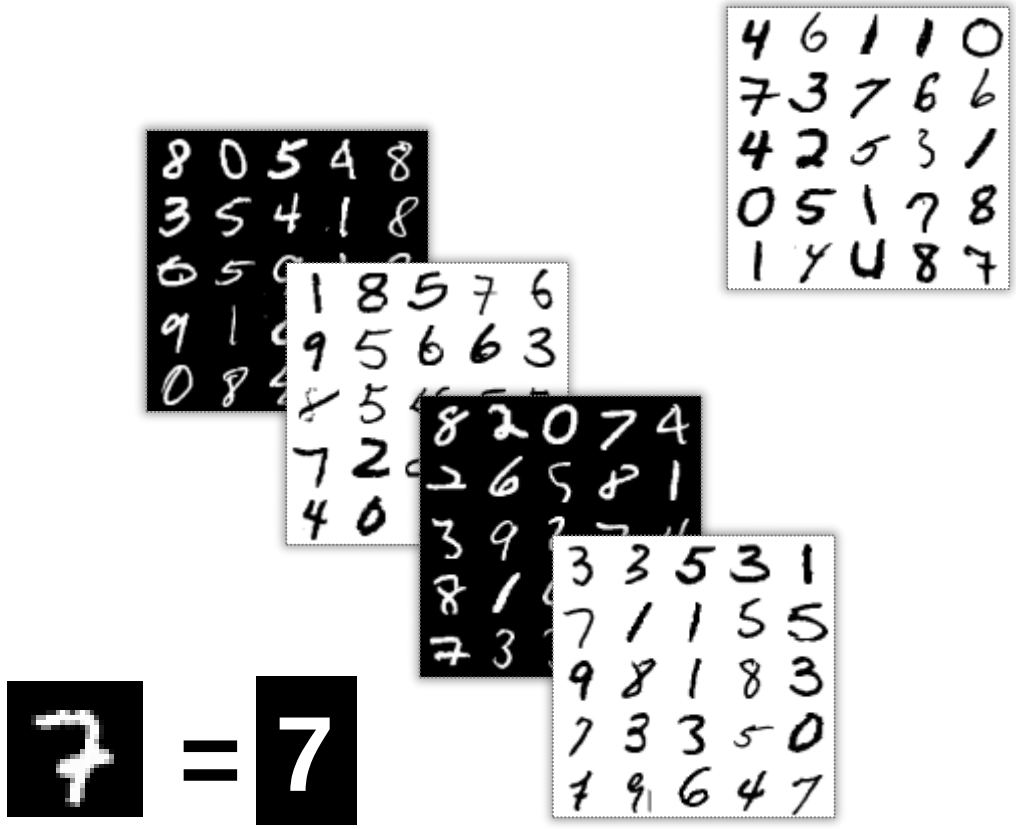
Recommended only when:

Training data	1000s to millions of labeled images
Computation	Compute intensive (requires GPU)
Training Time	Days to Weeks for real problems
Model accuracy	High (can over fit to small datasets)

© Terasoft, Inc.

MNIST: The “Hello, World!” of Computer Vision – (1)

What?	A set of handwritten digits from 0-9
Why?	An easy task for machine learning beginners
How many?	60,000 training images 10,000 test images
Best results?	99.79% accuracy



Sources: http://yann.lecun.com/exdb/mnist/
https://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results

© Terasoft, Inc.

MNIST: The “Hello, World!” of Computer Vision – (2)

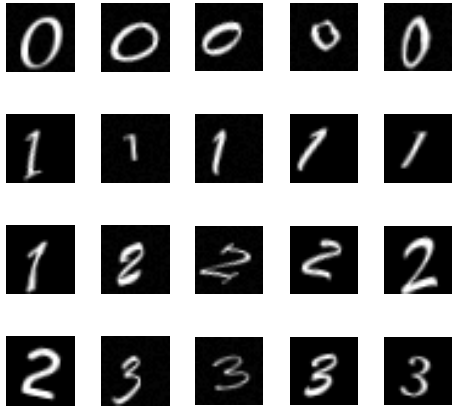
```
imds= imageDatastore(location);
```

```
layers = [ ...
    imageInputLayer([28 28 1], 'Normalization', 'none');
    convolution2dLayer(5, 20);
    reluLayer();
    maxPooling2dLayer(2, 'Stride', 2);
    fullyConnectedLayer(10);
    softmaxLayer();
    classificationLayer()];
```

```
opts = trainingOptions('sgdm',…
                       'MaxEpochs', 5,…
                       'InitialLearnRate',0.01,...
                       'MiniBatchSize',128,...
                       'ExecutionEnvironment', ‘multi-gpu’,…
                       'Plots','training-progress');

net = trainNetwork(imds, layers, opts);
```

```
label = classify(net, img)
```



Handle training images.

Create the layers for CNN.

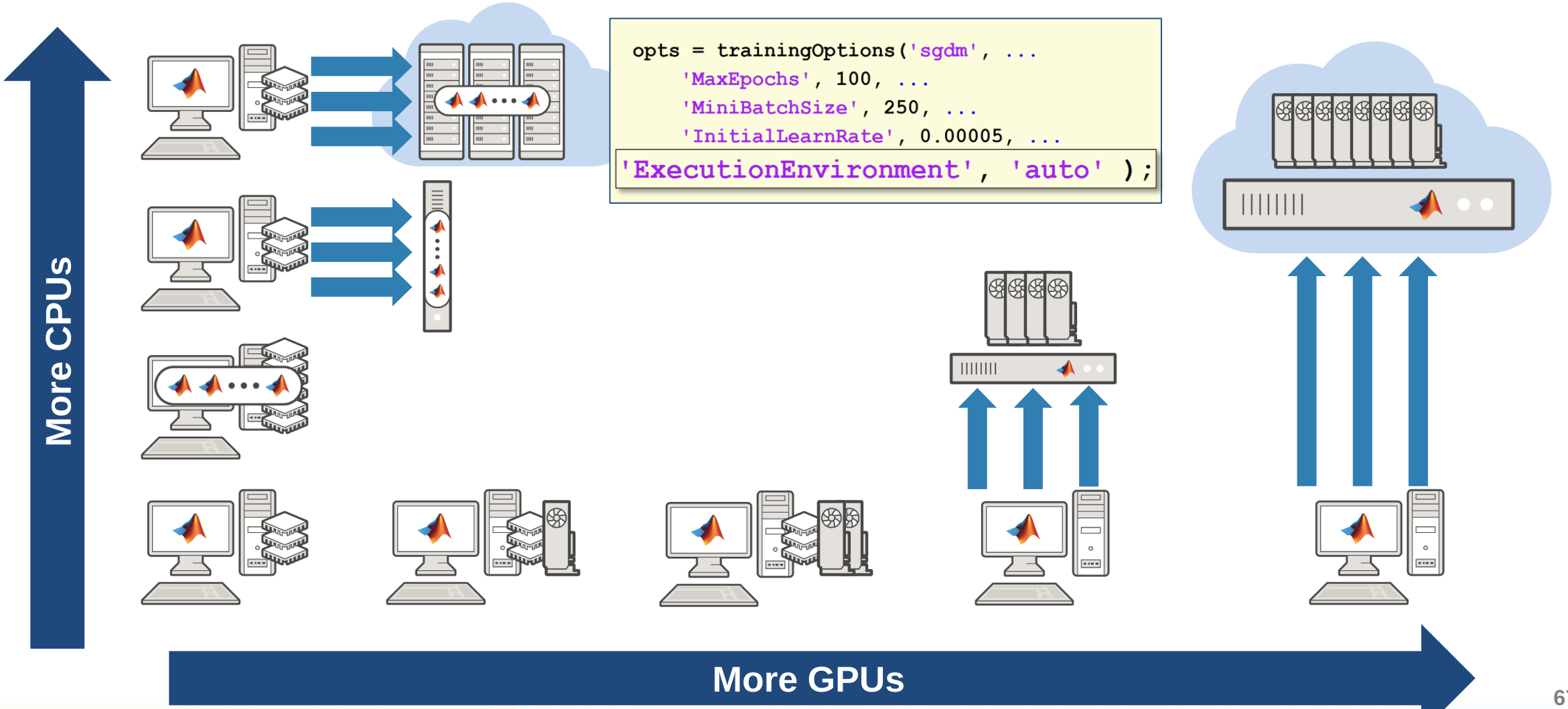
Set training options and train.

Classify image.

© Terasoft, Inc.

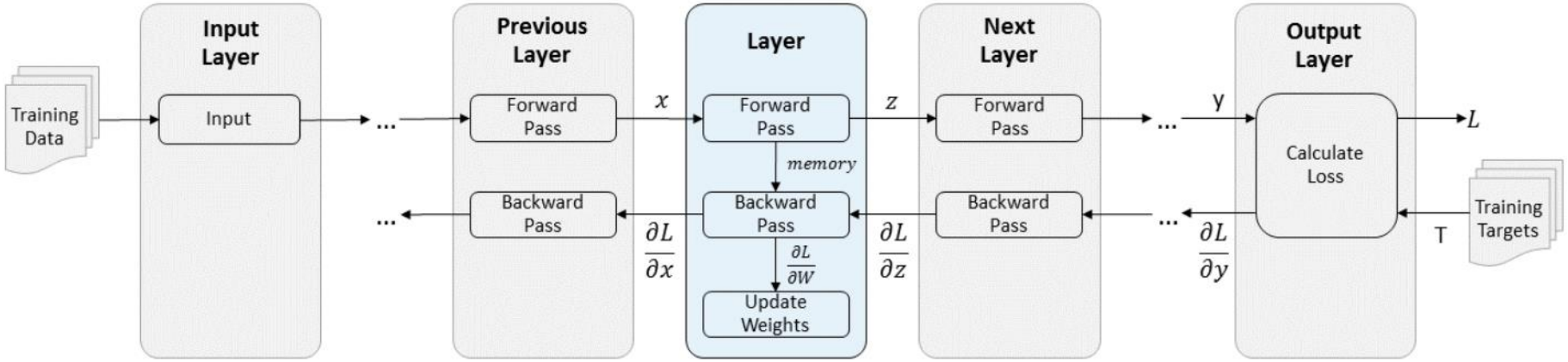
Deep Learning on CPU, GPU, multi-GPU and clusters

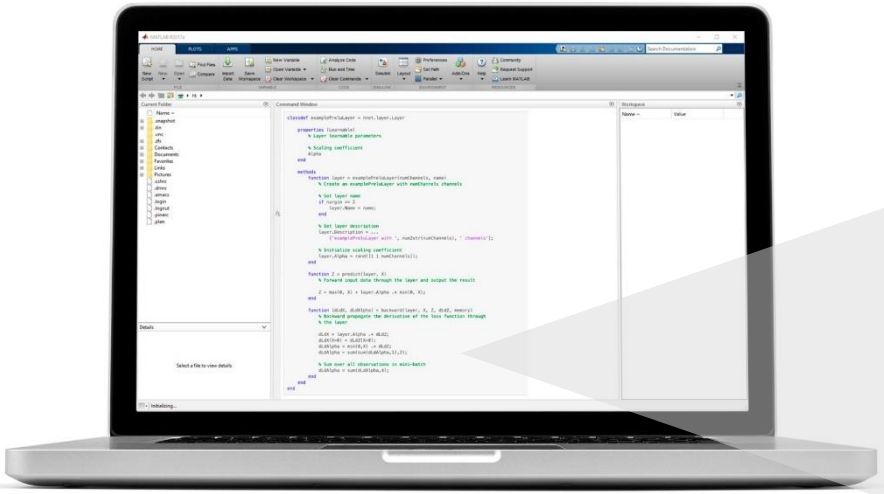
```
opts = trainingOptions('sgdm', ...
    'MaxEpochs', 100, ...
    'MiniBatchSize', 250, ...
    'InitialLearnRate', 0.00005, ...
'ExecutionEnvironment', 'auto' );
```



© Terasoft, Inc.

Define New Deep Neural Network Layers





```
function [dLdX, dLdAlpha] = backward(layer, X, Z, dLdZ, memory)
    % Backward propagate the derivative of the loss function through
    % the layer

    dLdX = layer.Alpha .* dLdZ;
    dLdX(X>0) = dLdZ(X>0);
    dLdAlpha = min(0,X) .* dLdZ;
    dLdAlpha = sum(sum(dLdAlpha,1),2);
```


© Terasoft, Inc.

Transfer Learning Workflow

Load pretrained network

Early layers that learned low-level features (edges, blobs, colors)

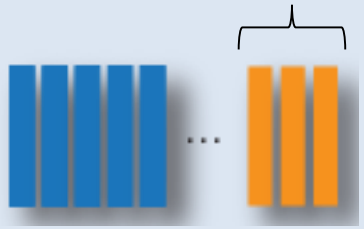
Last layers that learned task specific features



1 million images
1000s classes

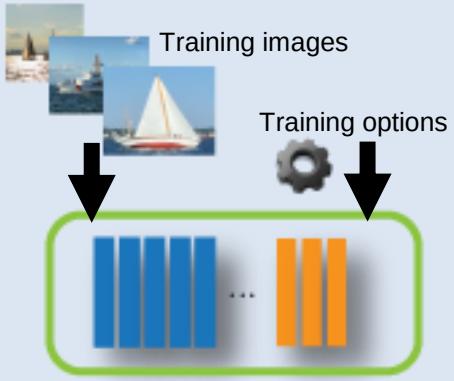
Replace final layers

New layers to learn features specific to your data



Fewer classes
Learn faster

Train network



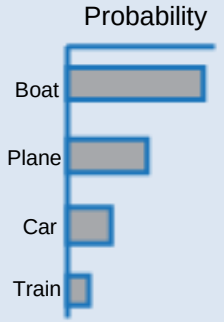
100s images
10s classes

Predict and assess network accuracy



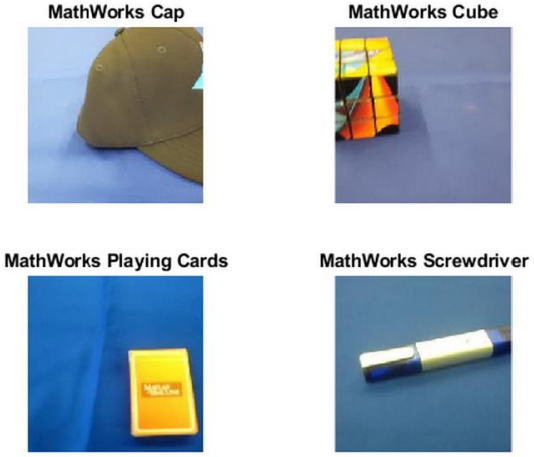
Deploy results





© Terasoft, Inc.

Transfer Learning Using AlexNet



Perform net surgery

Modify the existing network by deleting later layers and adding new ones.

```
% Here we only need to keep everything except the last 3 layers.
layers = net.Layers(1:end-3);

% Add new fully connected layer for 2 categories.
layers(end+1) = fullyConnectedLayer(64, 'Name', 'special 2');
layers(end+1) = reluLayer;

% the new layer adding non-linearity and improves the network's ability to handle data
layers(end+1) = fullyConnectedLayer(height(tbl), 'Name', 'fc8_2');

% Add the softmax layer and the classification layer
layers(end+1) = softmaxLayer;
layers(end+1) = classificationLayer()
```

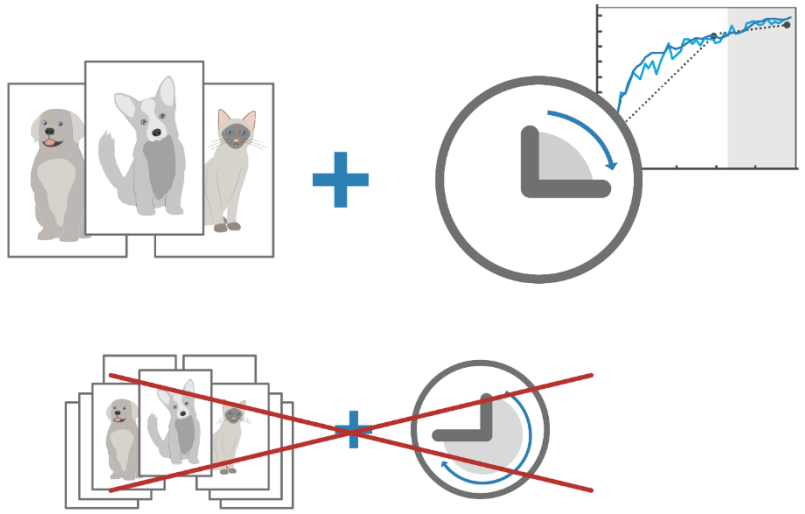
DELETE LAYERS

ADD LAYERS

© Terasoft, Inc.

Why Perform Transfer Learning

- Requires less data and training time
- Reference models (like AlexNet, VGG-16, VGG-19) are great feature extractors
- Leverage best network types from top researchers



AlexNet
PRETRAINED MODEL

VGG-16
PRETRAINED MODEL

ResNet
PRETRAINED MODEL

Caffe
MODELS

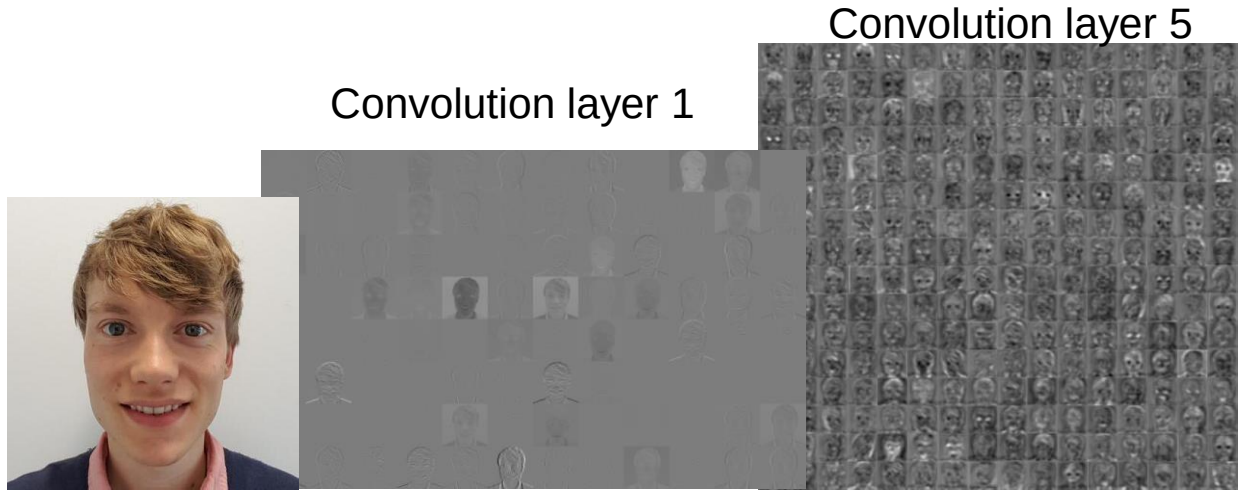
GoogLeNet
PRETRAINED MODEL

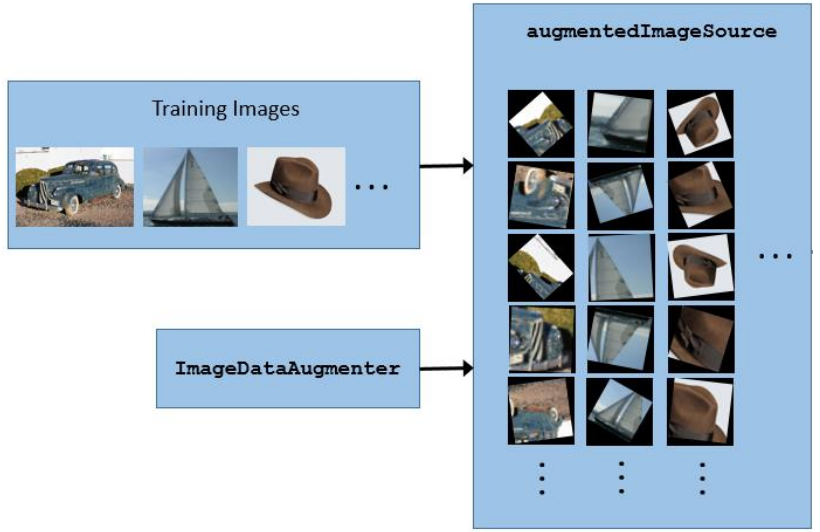
TensorFlow/Keras
MODELS

© Terasoft, Inc.

Other Deep learning Features

- **Improve training results**
  - Data augmentation with `imageDataAugmenter`
  - Automatic parameter selection with **Bayesian hyperparameter tuning**
- **Visualize and debug the networks**



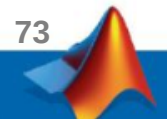


© Terasoft, Inc.

Agenda

- Deep Learning with MATLAB
  - What is Deep Learning?
  - Layers in Convolution Neural Network
  - Image classification using pre-trained network
  - Train a new model
  - Object detection

© Terasoft, Inc.



Is Object Recognition/Classification Enough ?

Car



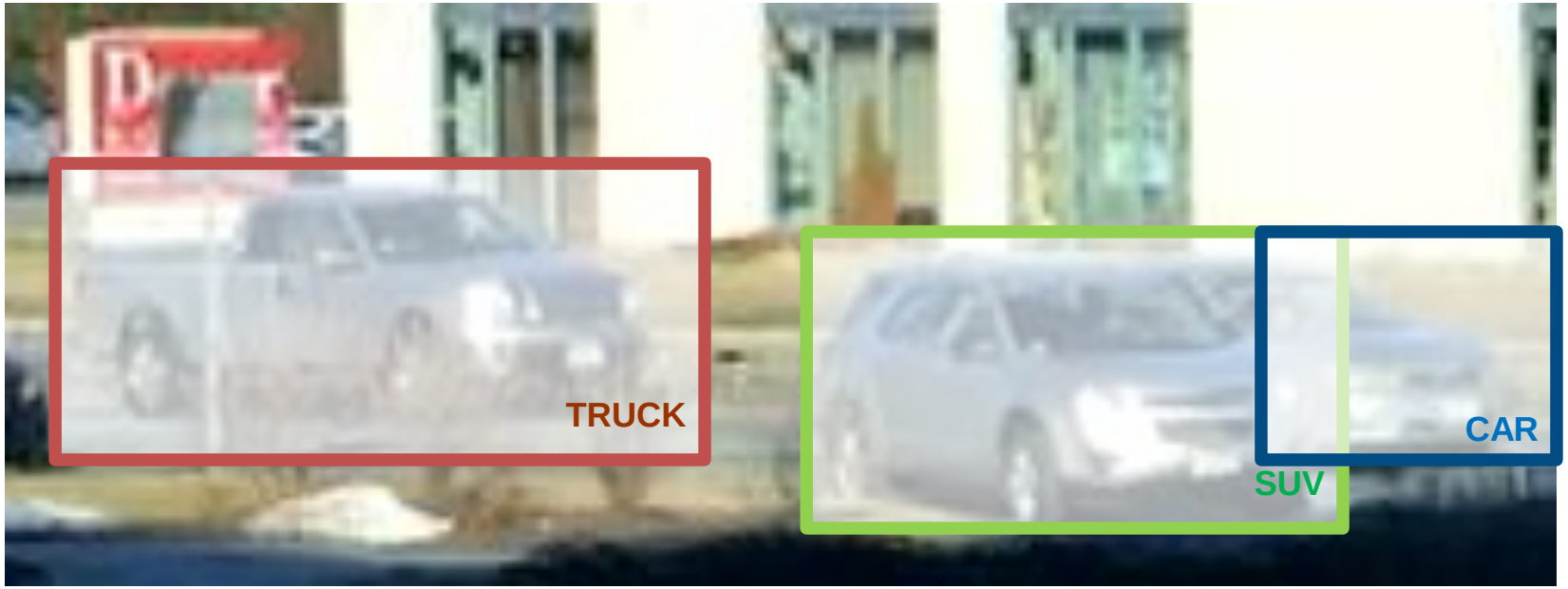
Label for entire image



Car ? SUV? Truck?

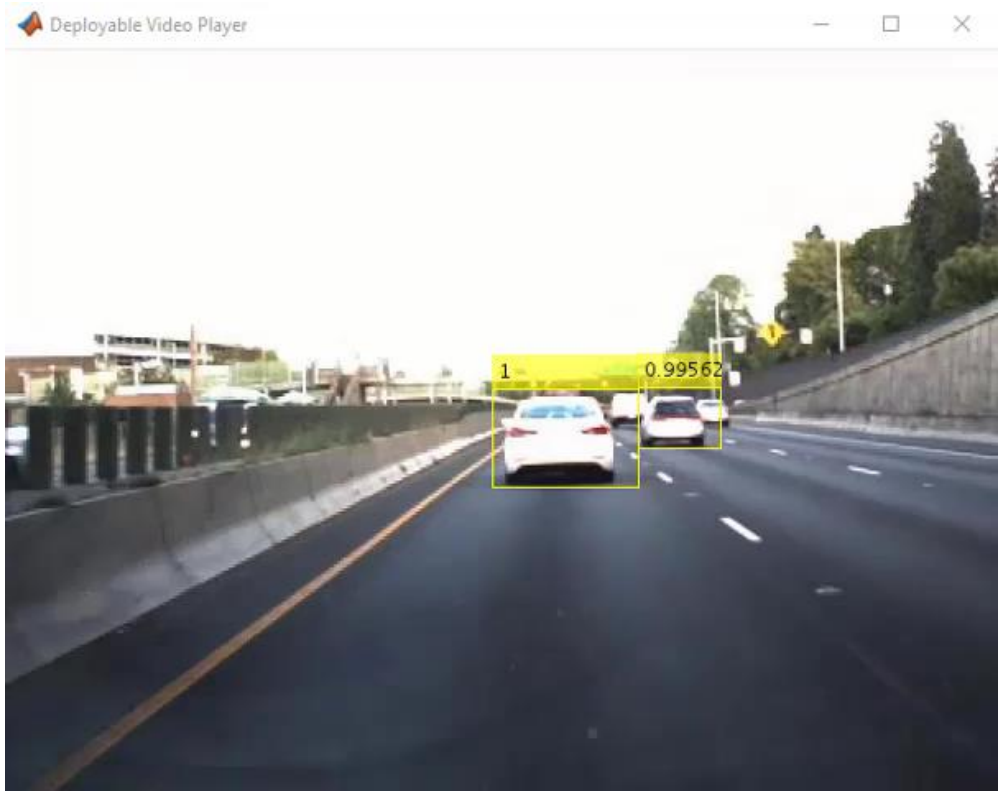
© Terasoft, Inc.

Object Detection – Locate and Classify Object



© Terasoft, Inc.

Detection and Localization Using Deep Learning



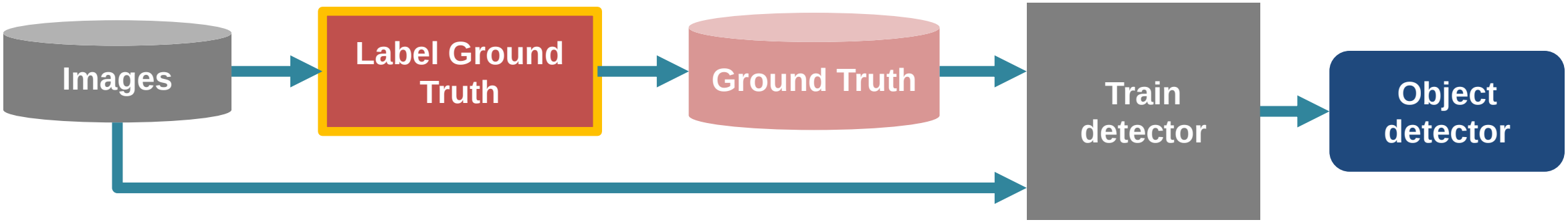


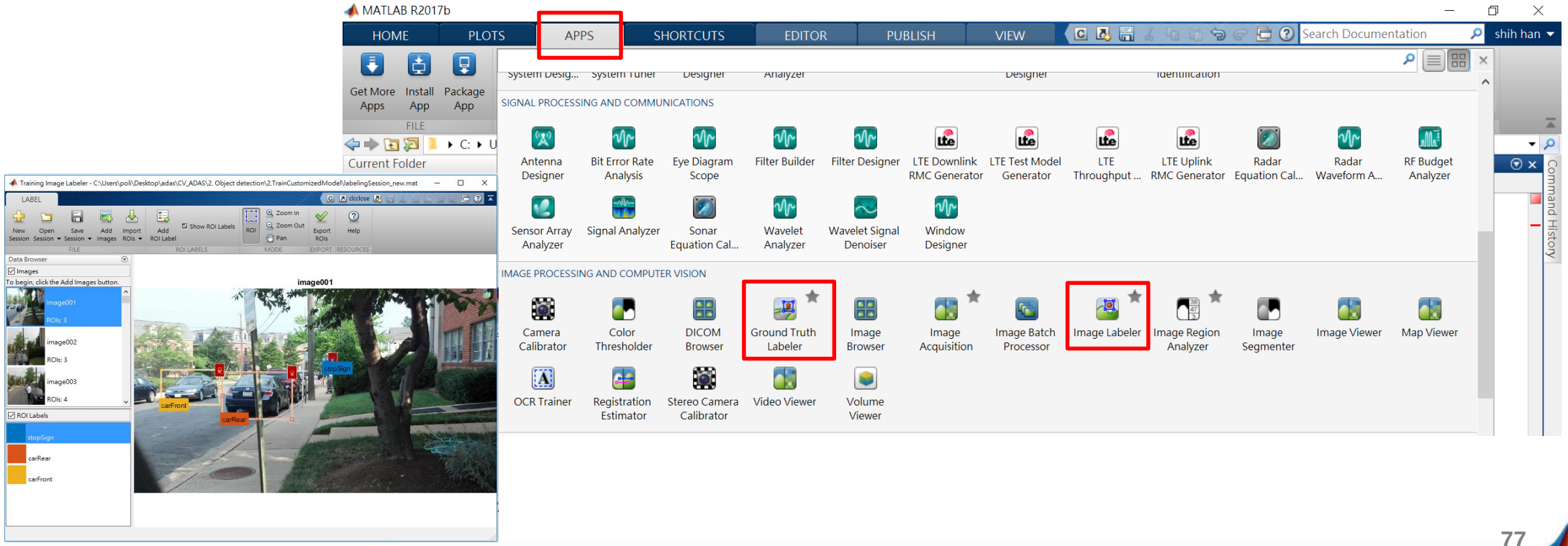
Regions with Convolutional Neural Network Features (R-CNN)

Semantic Segmentation using SegNet

© Terasoft, Inc.

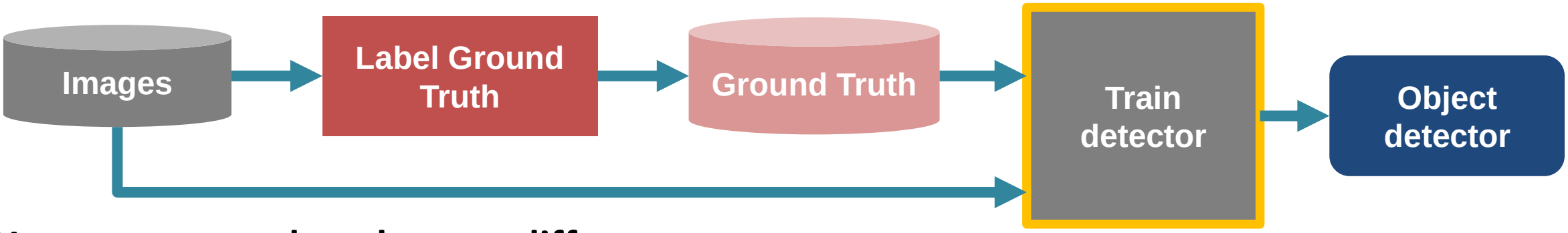
MATLAB Tools to Train Detectors





© Terasoft, Inc.

MATLAB Tools to Train Detectors



Use same ground truth to try different detectors.

Machine Learning

- **Cascade Object Detector**
- **Aggregate Channel Features**

Deep Learning

- **R-CNN**
- **Fast R-CNN**
- **Faster R-CNN**

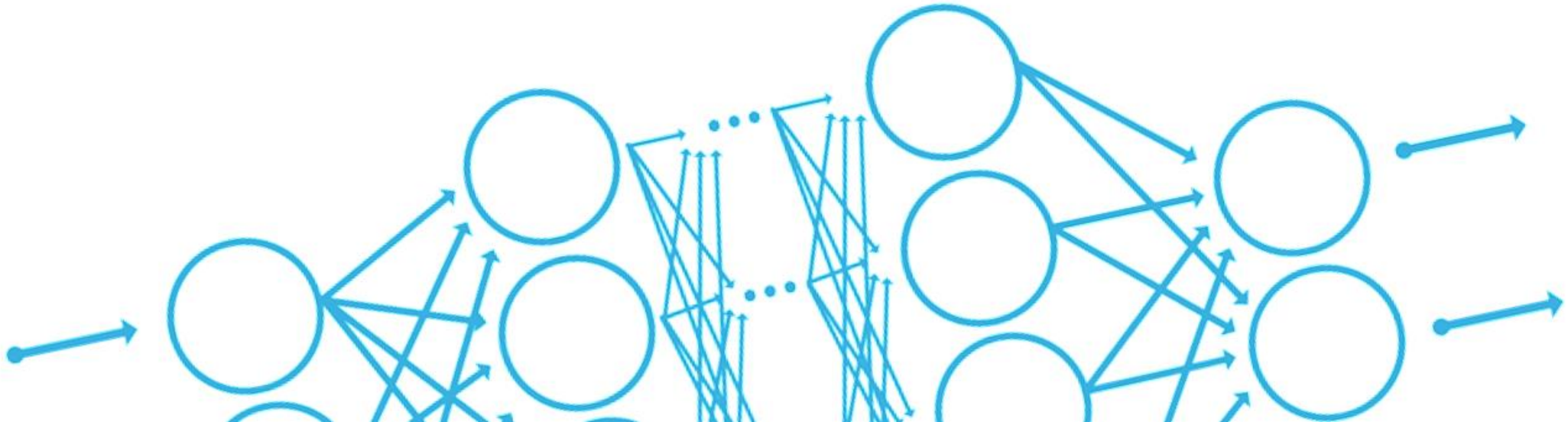
Single Line of Code to Train Each Detector

E.g.

```
trainCascadeObjectDetector
trainFasterRCNNObjectDetector
```

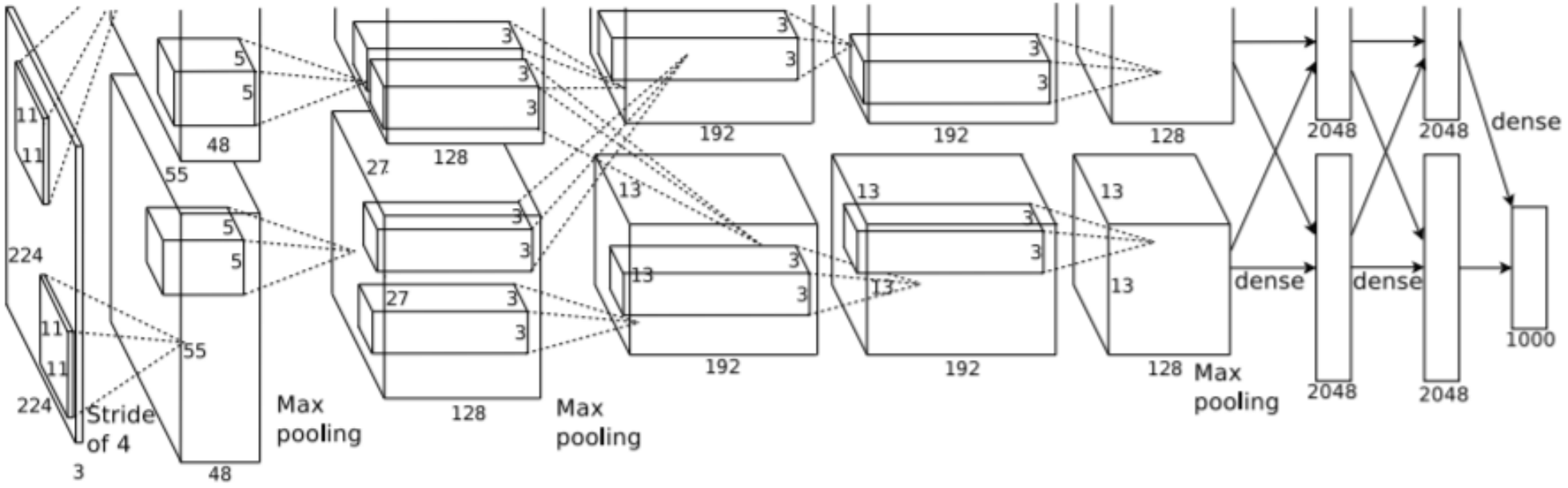

© Terasoft, Inc.

Thank You



© Terasoft, Inc.

Alexnet Model



© Terasoft, Inc.

Alexnet Model in matlab

1	'data'	Image Input	227x227x3 images with 'zerocenter' normalization
2	'conv1'	Convolution	96 11x11x3 convolutions with stride [4 4] and padding [0 0 0 0]
3	'relu1'	ReLU	ReLU
4	'norm1'	Cross Channel Normalization	cross channel normalization with 5 channels per element
5	'pool1'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
6	'conv2'	Convolution	256 5x5x48 convolutions with stride [1 1] and padding [2 2 2 2]
7	'relu2'	ReLU	ReLU
8	'norm2'	Cross Channel Normalization	cross channel normalization with 5 channels per element
9	'pool2'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
10	'conv3'	Convolution	384 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
11	'relu3'	ReLU	ReLU
12	'conv4'	Convolution	384 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
13	'relu4'	ReLU	ReLU
14	'conv5'	Convolution	256 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
15	'relu5'	ReLU	ReLU
16	'pool5'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
17	'fc6'	Fully Connected	4096 fully connected layer
18	'relu6'	ReLU	ReLU
19	'drop6'	Dropout	50% dropout
20	'fc7'	Fully Connected	4096 fully connected layer
21	'relu7'	ReLU	ReLU
22	'drop7'	Dropout	50% dropout
23	'fc8'	Fully Connected	1000 fully connected layer
24	'prob'	Softmax	softmax
25	'output'	Classification Output	crossentropyex with 'tench', 'goldfish', and 998 other classes

5 Convolution layers

3 Fully Connected layers

© Terasoft, Inc.

