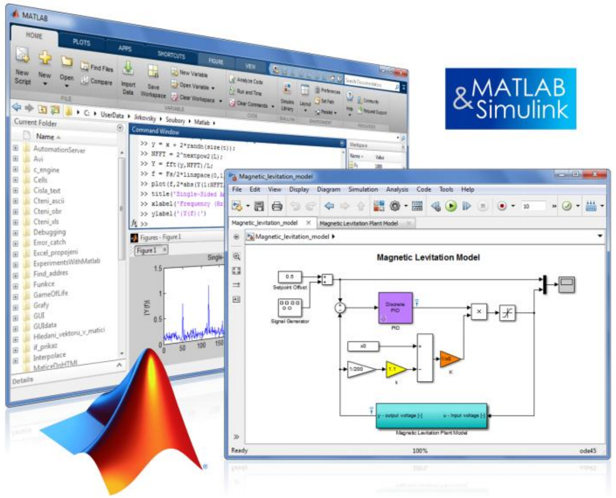
7.9.2017 Brno

HUMUSOFT®

TCC 2017

Deep Learning (a Computer Vision)



```
>> y = x + 2*randn(size(t));
>> NFFT = 2^nextpow2(L);
>> Y = fft(y,NFFT)/L;
>> f = Fs/2*linspace(0,1
>> plot(f,2*abs(Y(1:NFFT
>> title('Single-Sided A
>> xlabel('Frequency (Hz
>> ylabel('|Y(f)|')
>>
```

MATLAB & Simulink

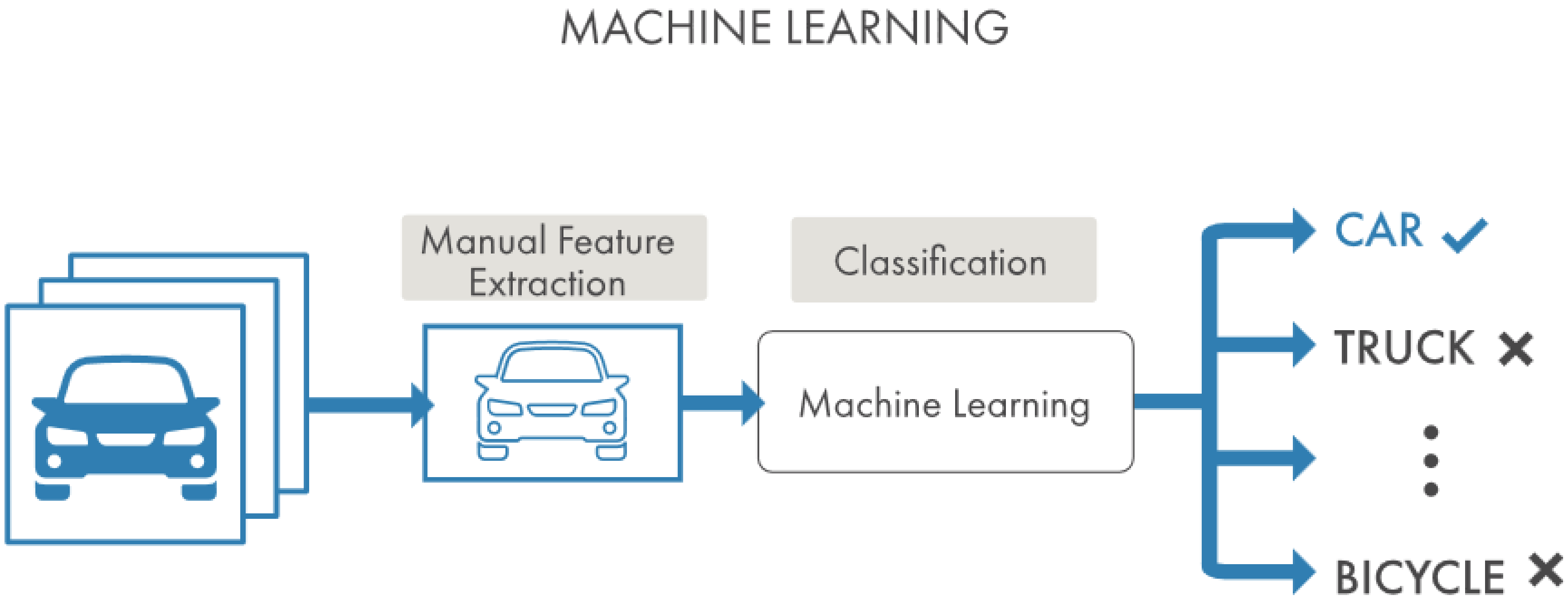
Jaroslav Jirkovský
jirkovsky@humusoft.cz

www.humusoft.cz
info@humusoft.cz

www.mathworks.com

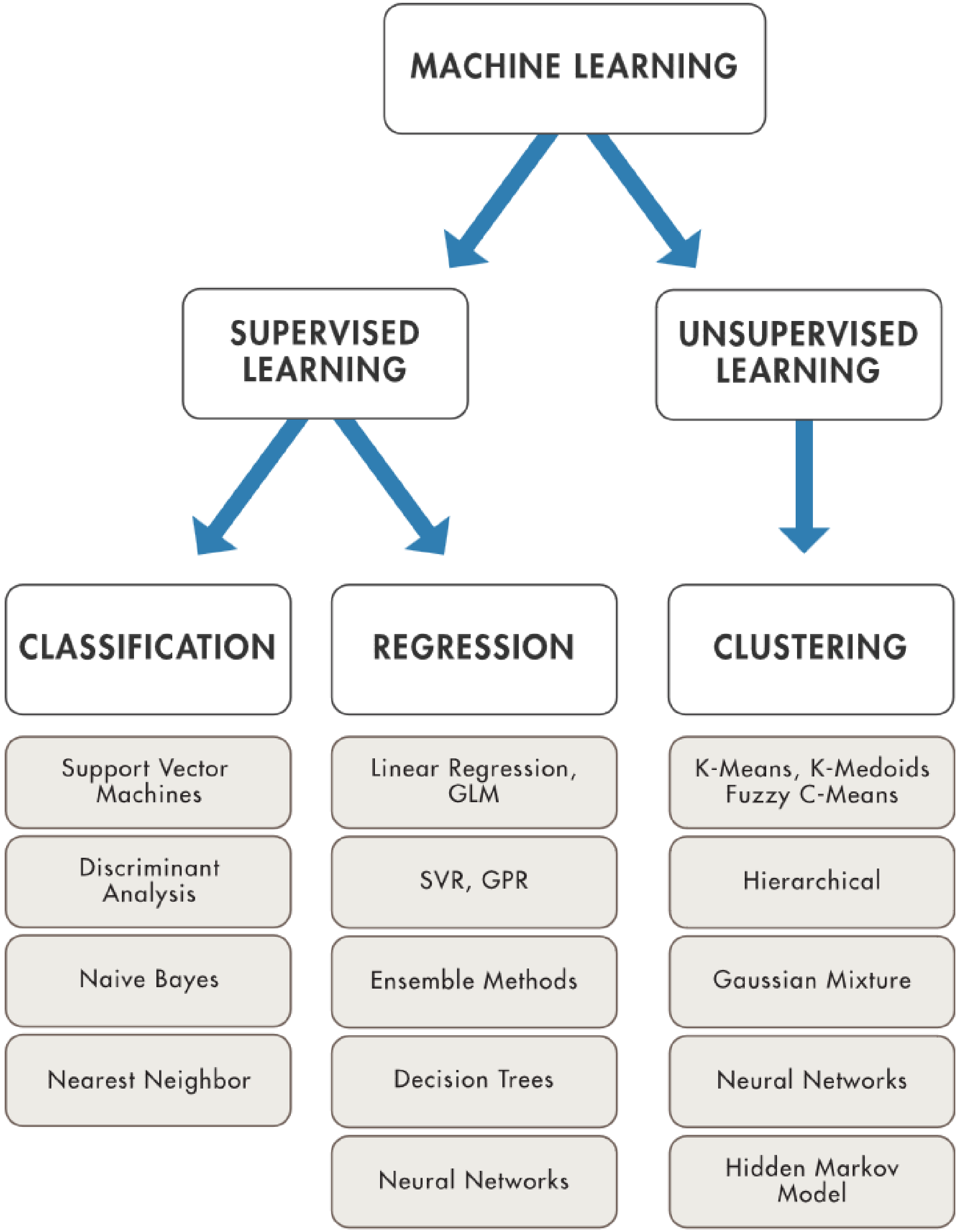
What is Machine Learning ?

Machine learning uses data and produces a program to perform a task



Machine Learning

- **Different Types of Learning:**



MACHINE LEARNING

SUPERVISED LEARNING

UNSUPERVISED LEARNING

CLASSIFICATION

REGRESSION

CLUSTERING

Support Vector Machines

Discriminant Analysis

Naive Bayes

Nearest Neighbor

Linear Regression, GLM

SVR, GPR

Ensemble Methods

Decision Trees

Neural Networks

K-Means, K-Medoids Fuzzy C-Means

Hierarchical

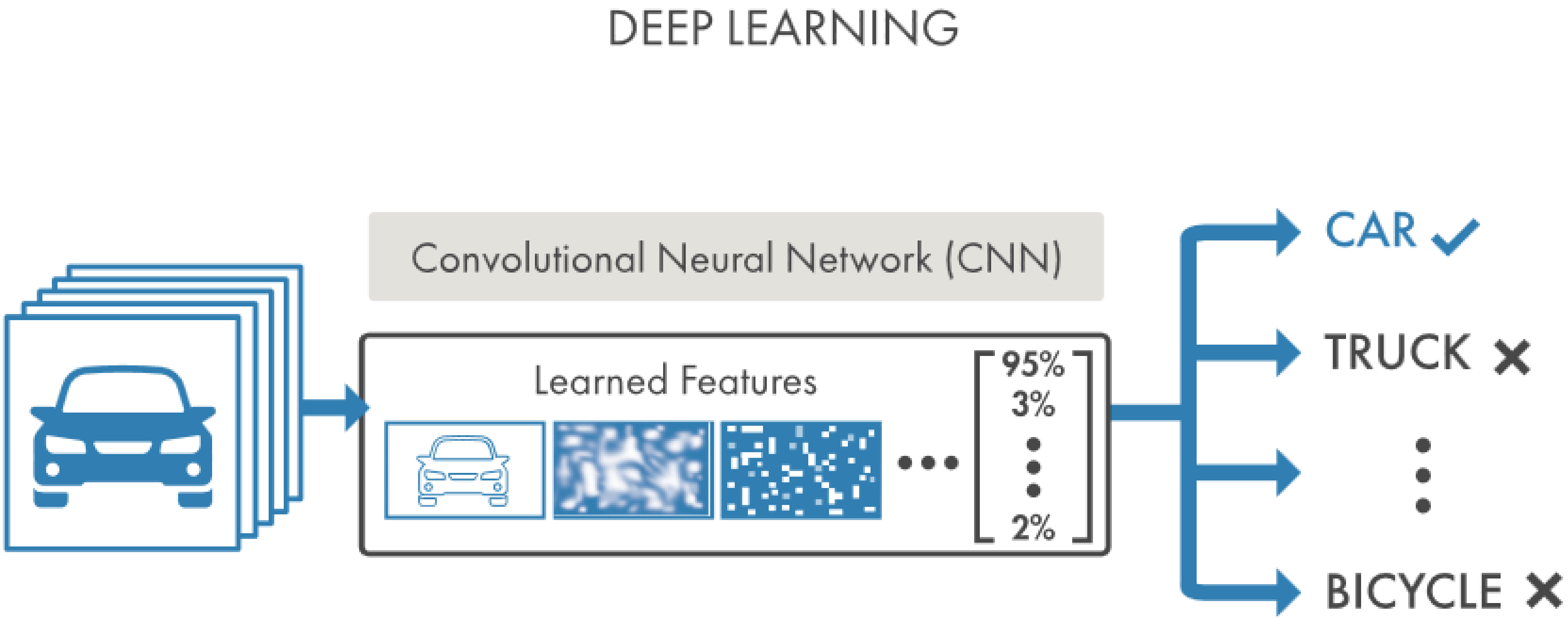
Gaussian Mixture

Neural Networks

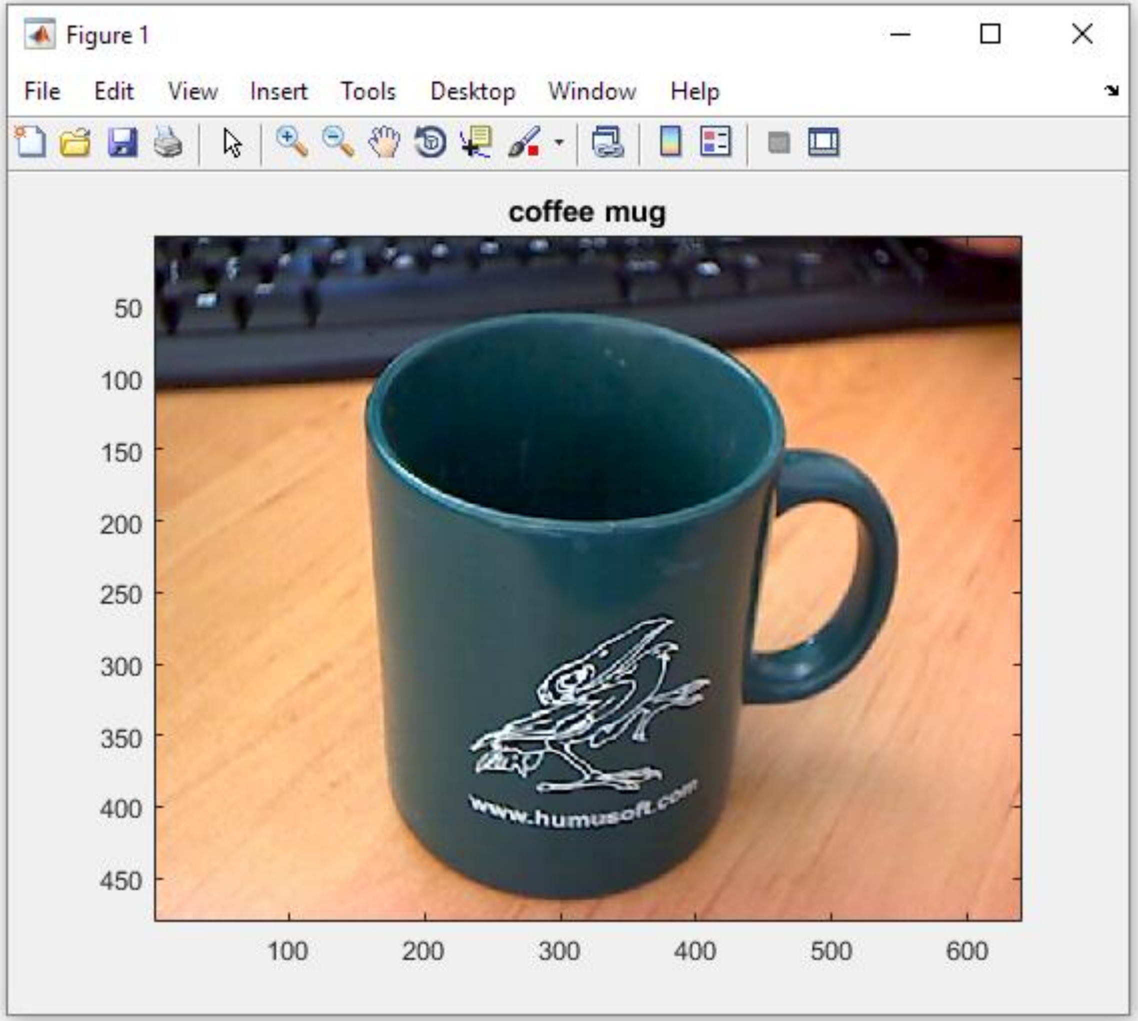
Hidden Markov Model

What is Deep Learning ?

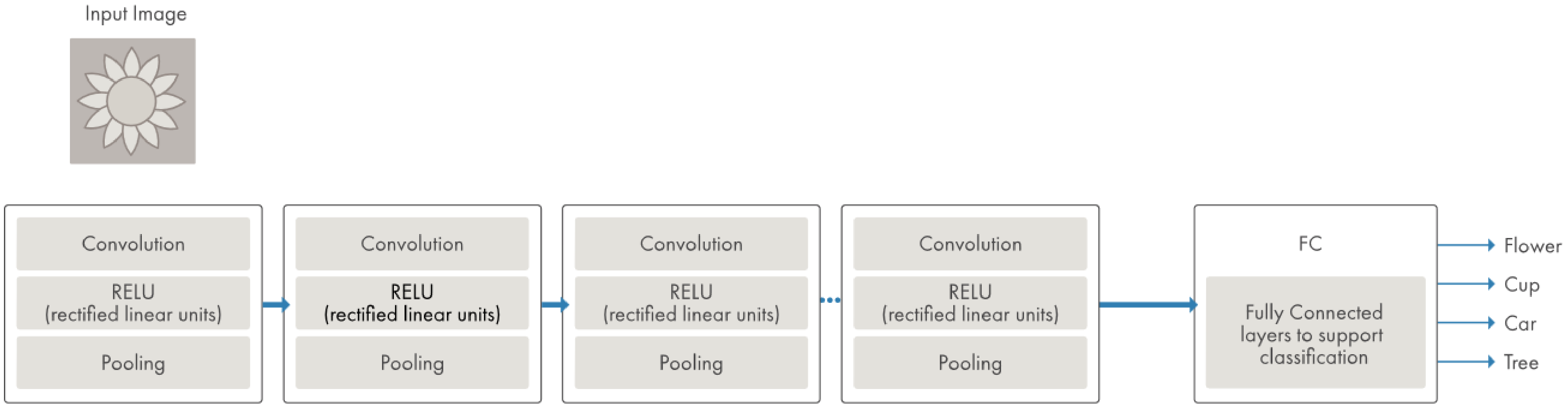
Deep learning performs end-end learning by learning features, representations and tasks directly from images, text and sound



Demo : Live Object Recognition with Webcam



Convolutional Neural Networks



Input Image

Convolution
RELU
(rectified linear units)
Pooling

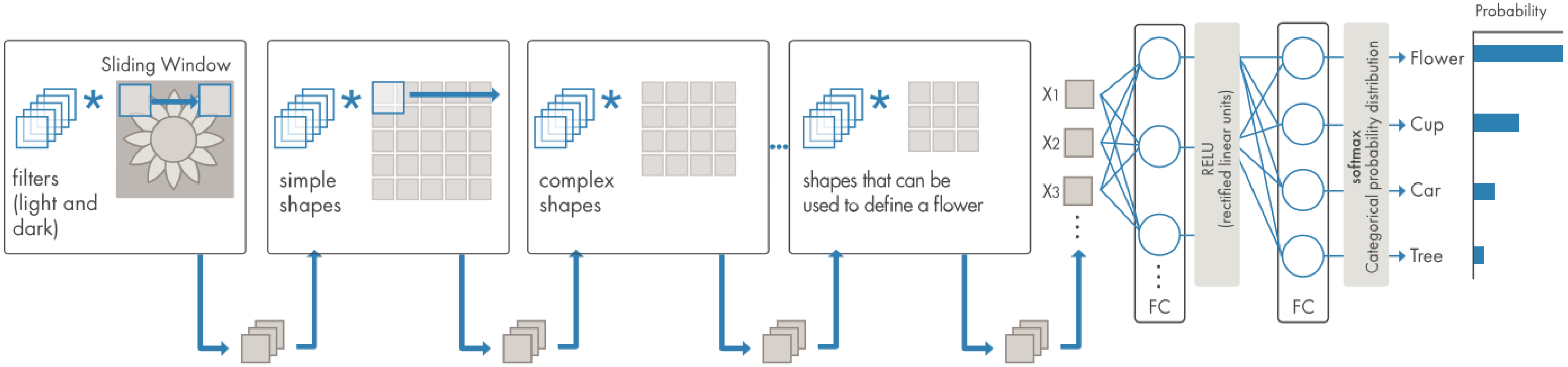
Convolution
RELU
(rectified linear units)
Pooling

Convolution
RELU
(rectified linear units)
Pooling

Convolution
RELU
(rectified linear units)
Pooling

FC
Fully Connected layers to support classification

Flower
Cup
Car
Tree



Sliding Window
filters (light and dark)

simple shapes

complex shapes

shapes that can be used to define a flower

X1
X2
X3

FC

RELU
(rectified linear units)

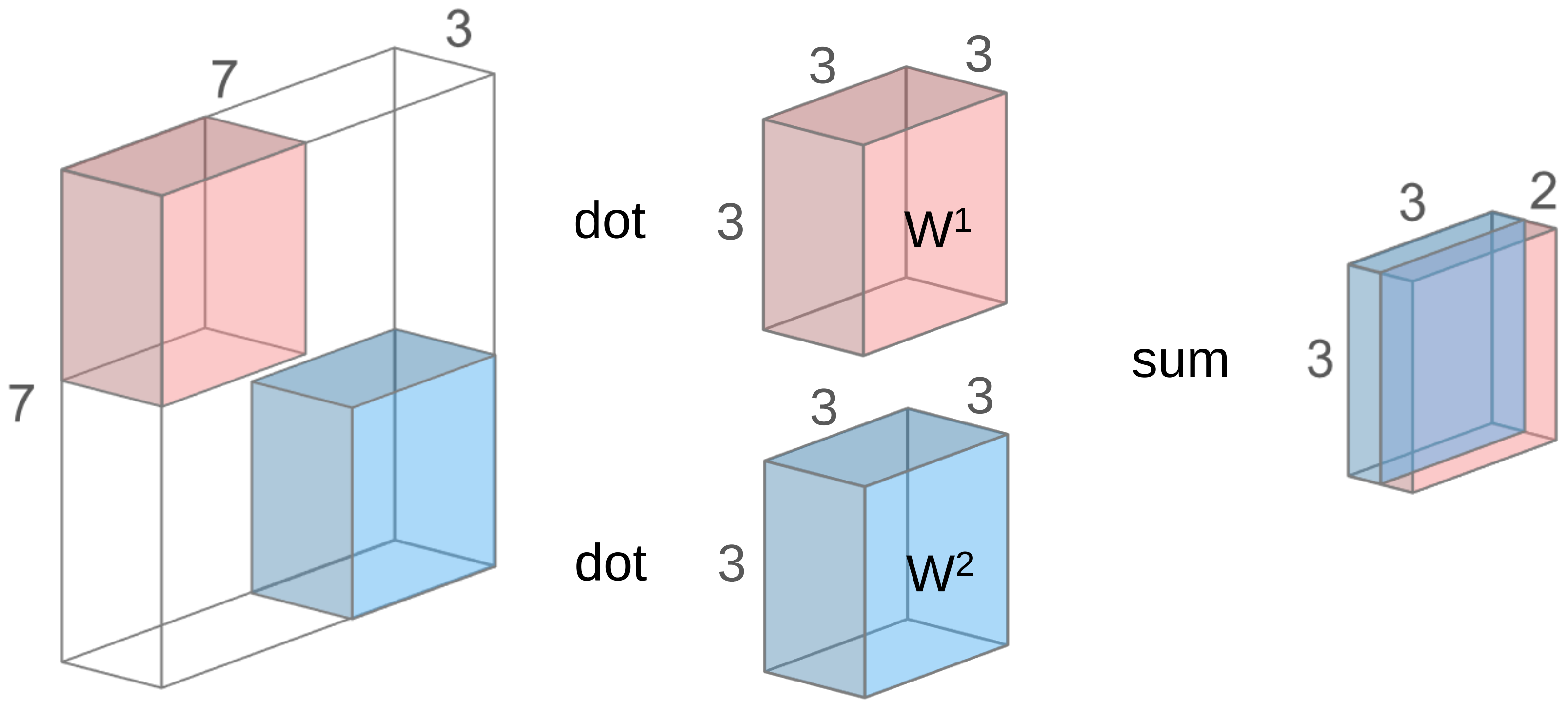
FC

softmax
Categorical probability distribution

Probability
Flower
Cup
Car
Tree

Convolution Layer

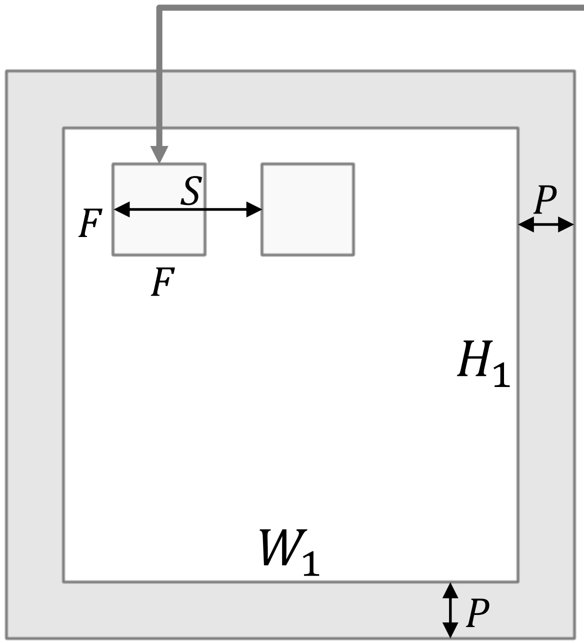
- **Core building block of a CNN**
- **Convolve the filters sliding them across the input, computing the dot product**

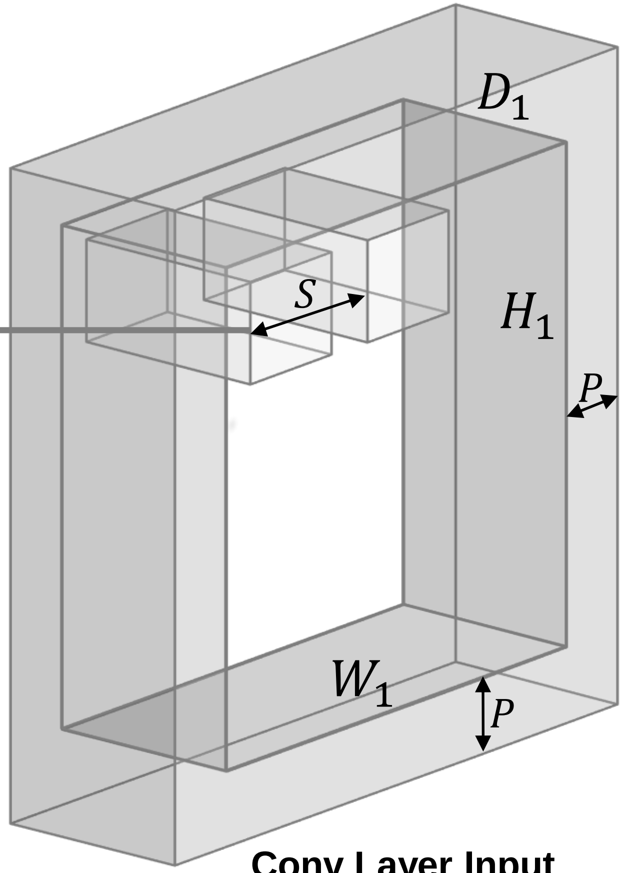


- **Intuition: learn filters that activate when they “see” some specific feature**

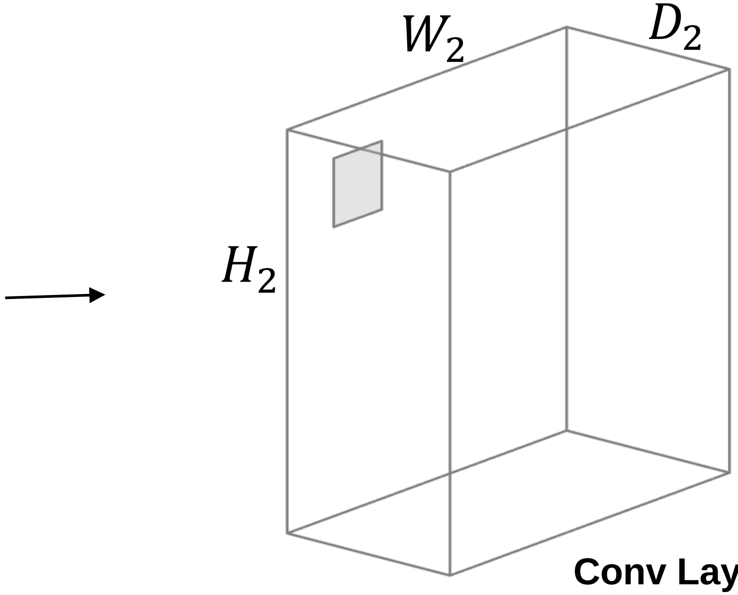
Convolution Layer – Choosing Hyperparameters

- **Number of filters,** K
- **Filter size,** F
- **Stride,** S
- **Zero padding,** P





Conv Layer Input

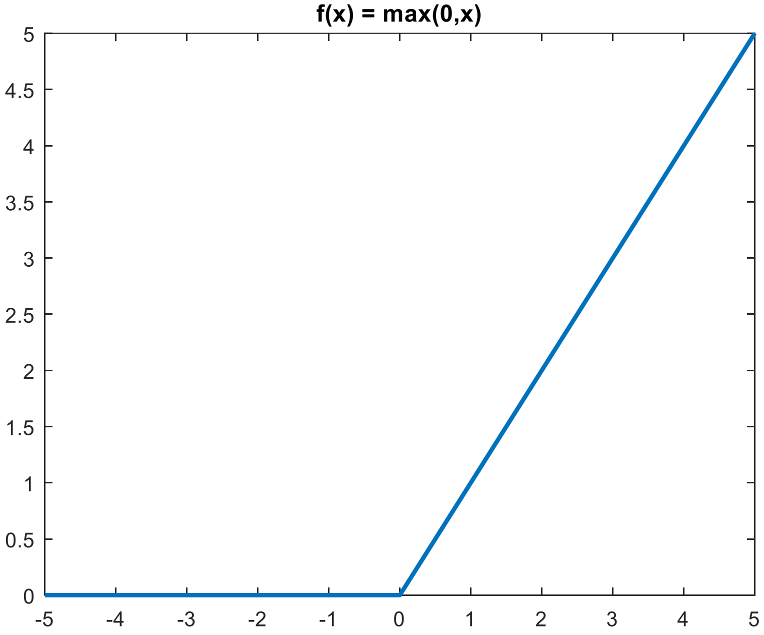


Conv Layer Output

$$W_2 = (W_1 - F + 2P)/S + 1$$
$$H_2 = (H_1 - F + 2P)/S + 1$$
$$D_2 = K$$

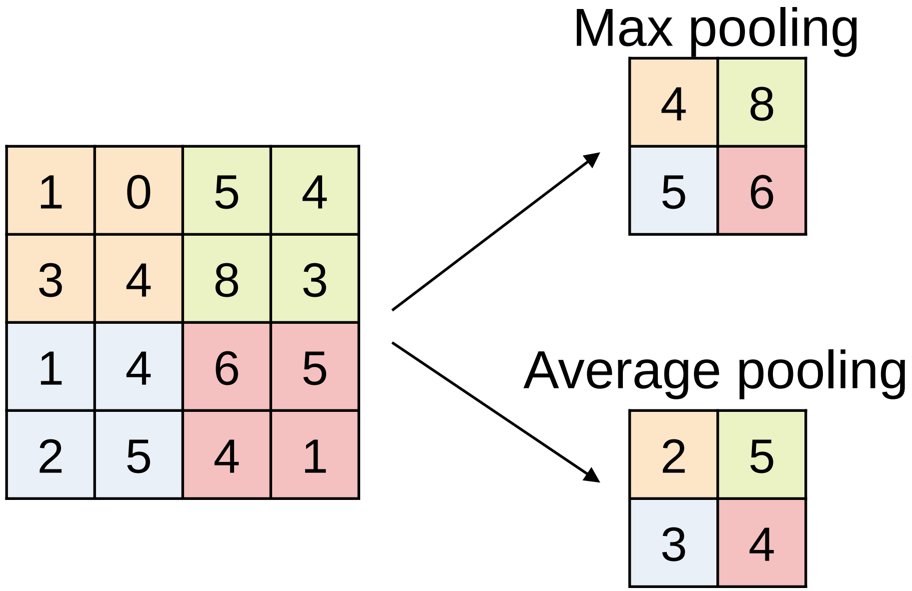
Rectified Linear Unit (ReLU) Layer

- **Frequently used in combination with Convolution layers**
- **Do not add complexity to the network**
- **Most popular choice: $\boldsymbol{f(x) = max(0, x)}$, activation is thresholded at 0**



Pooling Layer

- **Perform a downsampling operation across the spatial dimensions**
- **Goal: progressively decrease the size of the layers**
- **Max pooling and average pooling methods**
- **Popular choice: Max pooling with 2x2 filters, Stride = 2**



1	0	5	4
3	4	8	3
1	4	6	5
2	5	4	1

Max pooling

4	8
5	6

Average pooling

2	5
3	4

Other Layers

- **Fully Connected**
  - Full connections to all activation in previous layer, as in regular Neural Networks
  - Each entry is treated as a feature that the network has learned
- **Softmax**
  - Computes the probability of a sample belonging to a specific class
- **Classification**
  - Performs the classification (output layer)
- **Local Response Normalization, Dropout, etc.**

Deep Learning is Ubiquitous

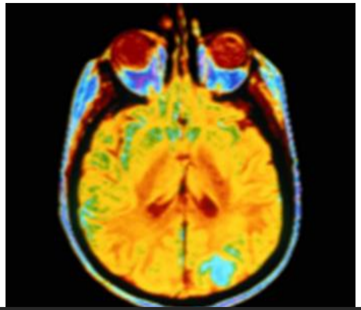
Computer Vision

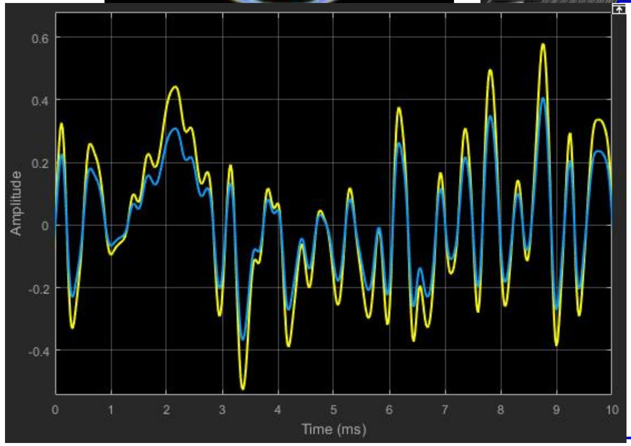
- **Pedestrian and traffic sign detection**
- **Landmark identification**
- **Scene recognition**
- **Medical diagnosis and drug discovery**

Text and Signal Processing

- **Speech Recognition**
- **Speech & Text Translation**

Robotics & Controls





and many more…

Why is Deep Learning so Popular ?

- **Results: Achieved substantially better results on ImageNet large scale recognition challenge**
  - **95% + accuracy on ImageNet 1000 class challenge**

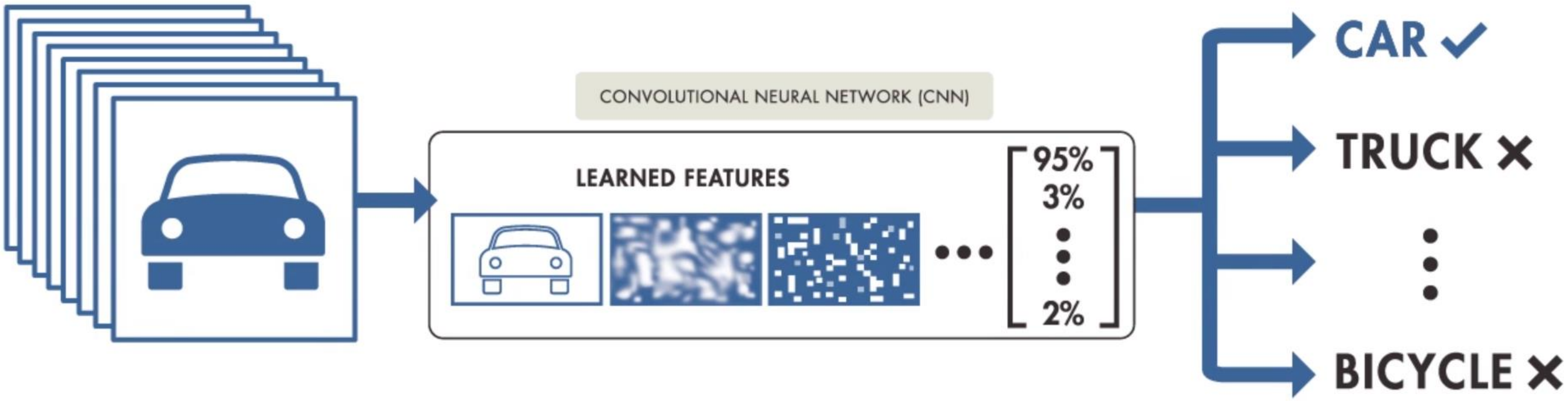
Year	Error Rate
Pre-2012 (traditional computer vision and machine learning techniques)	> 25%
2012 (Deep Learning)	~ 15%
2015 (Deep Learning)	<5 %

- **Computing Power: GPU’s and advances to processor technologies have enabled us to train networks on massive sets of data.**

- **Data: Availability of storage and access to large sets of labeled data**
  - **E.g. ImageNet , PASCAL VoC , Kaggle**

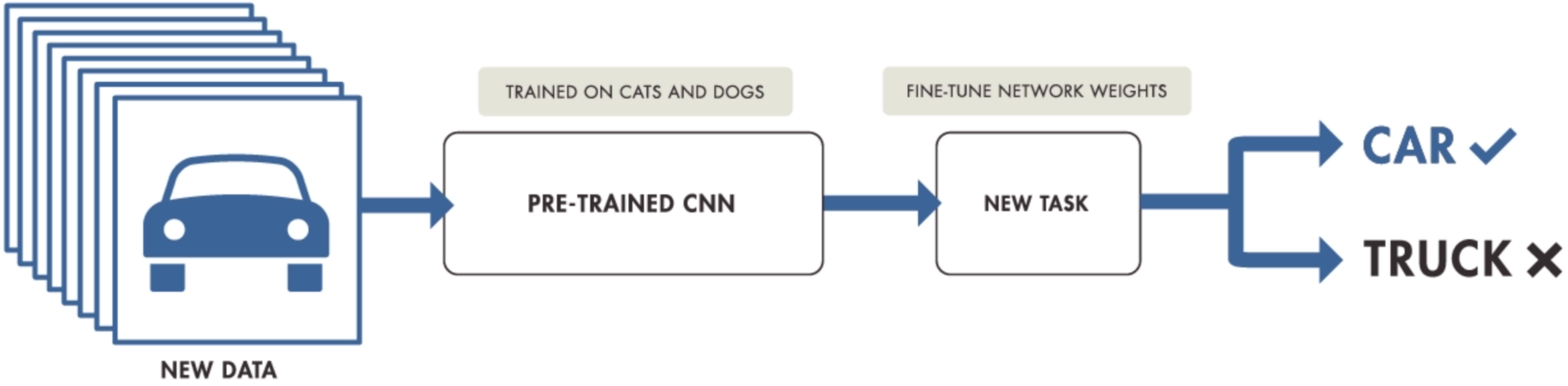
3 Approaches for Deep Learning

- **Approach 1: Train a Deep Neural Network from Scratch**



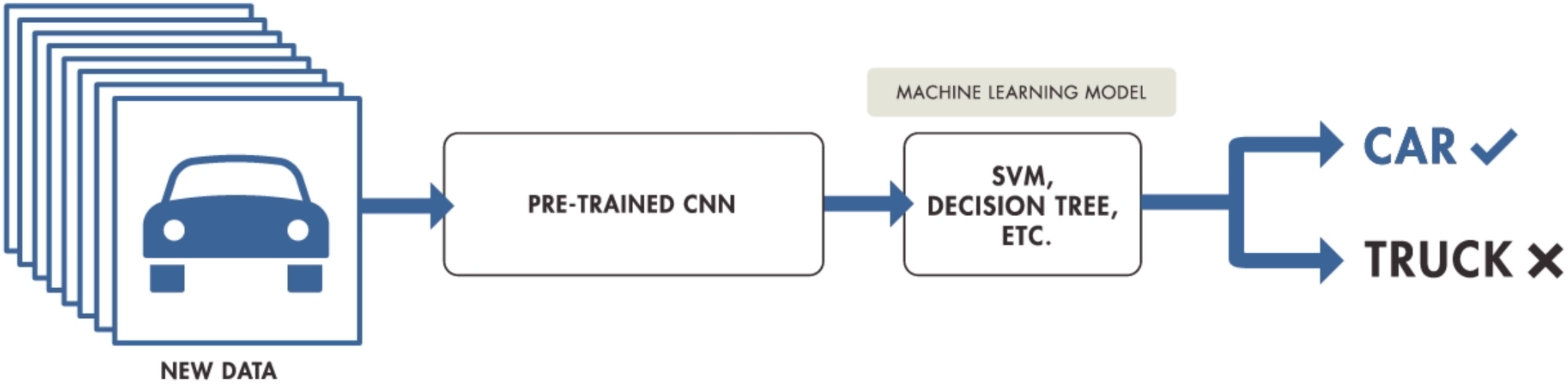
3 Approaches for Deep Learning

- **Approach 2: Fine-tune a pre-trained model (transfer learning)**



3 Approaches for Deep Learning

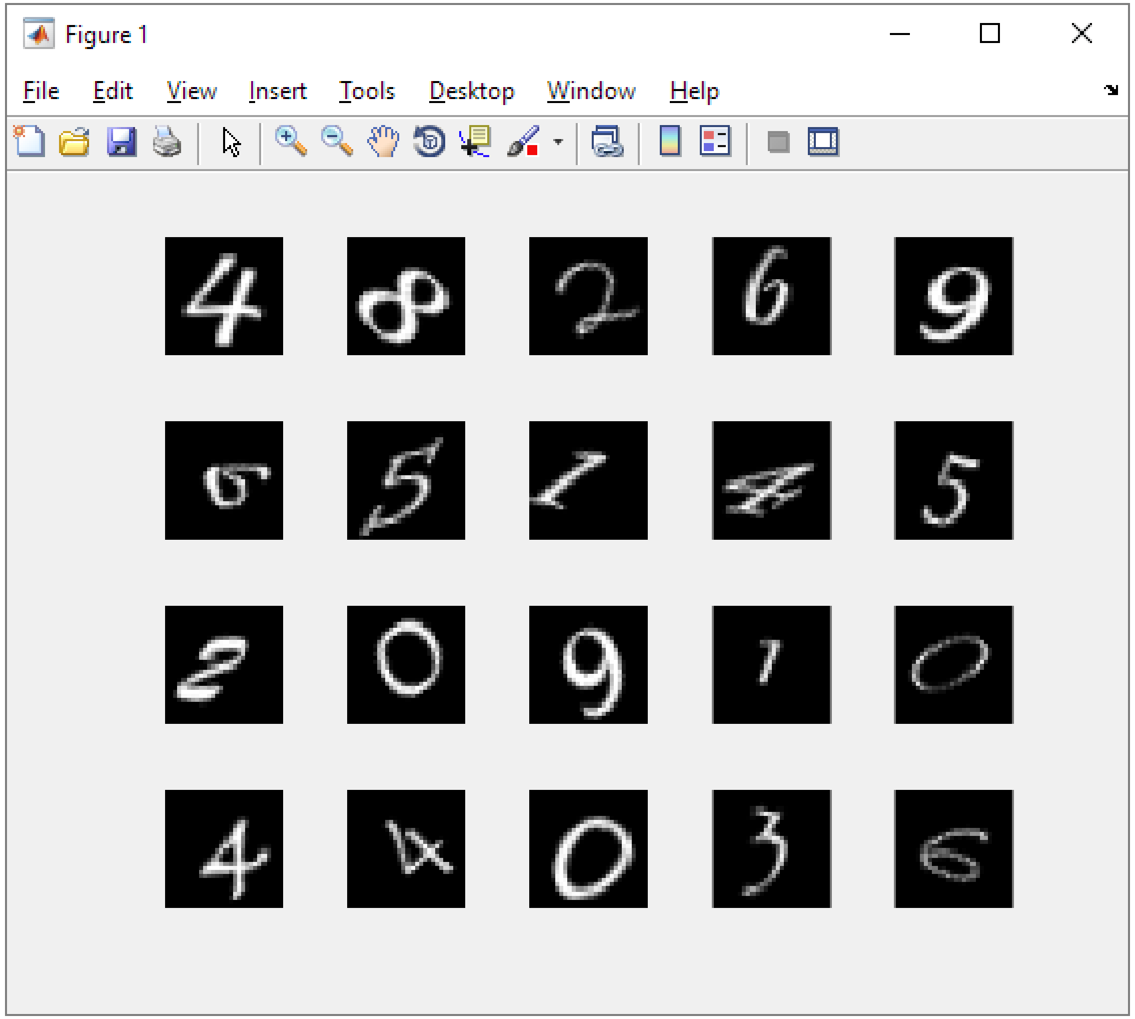
- **Approach 3: Feature Extraction with traditional Machine Learning**



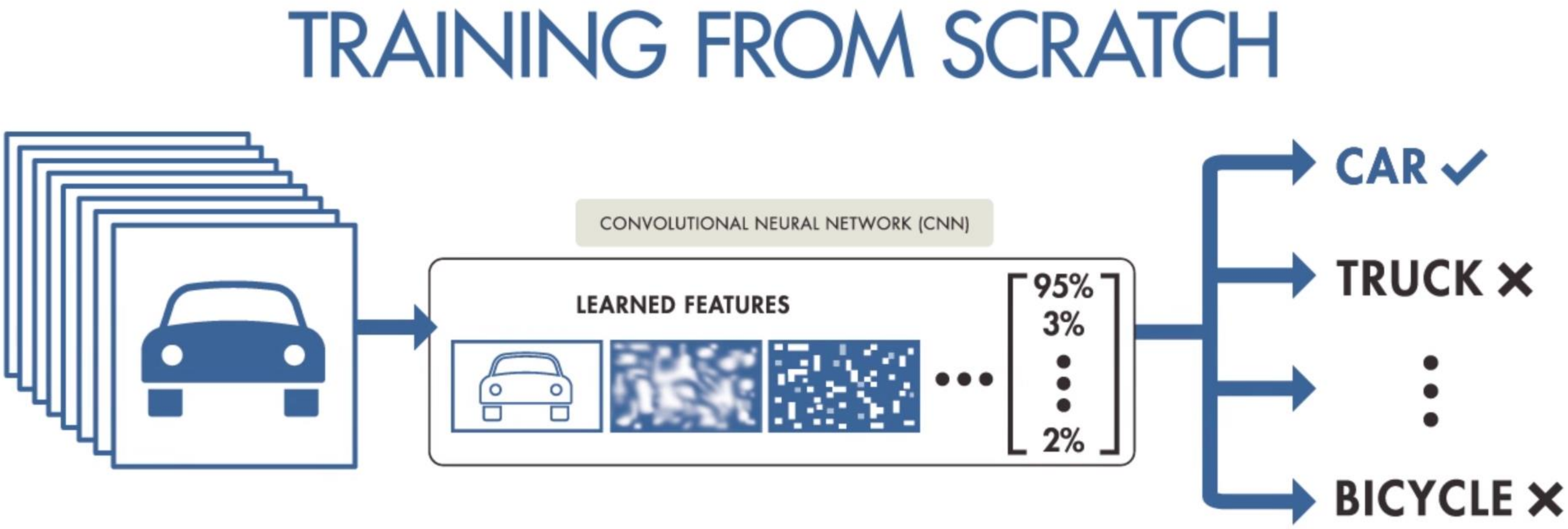
CNN in MATLAB

```
layers = [imageInputLayer([28 28 1])
          convolution2dLayer(5,20)
          reluLayer()
          maxPooling2dLayer(2,'Stride',2)
          fullyConnectedLayer(10)
          softmaxLayer()
          classificationLayer()];
options = trainingOptions('sgdm');
convnet = trainNetwork(trainingData,layers,options);
results = classify(convnet,newData);
```

Demo : Train a Deep Neural Network from Scratch



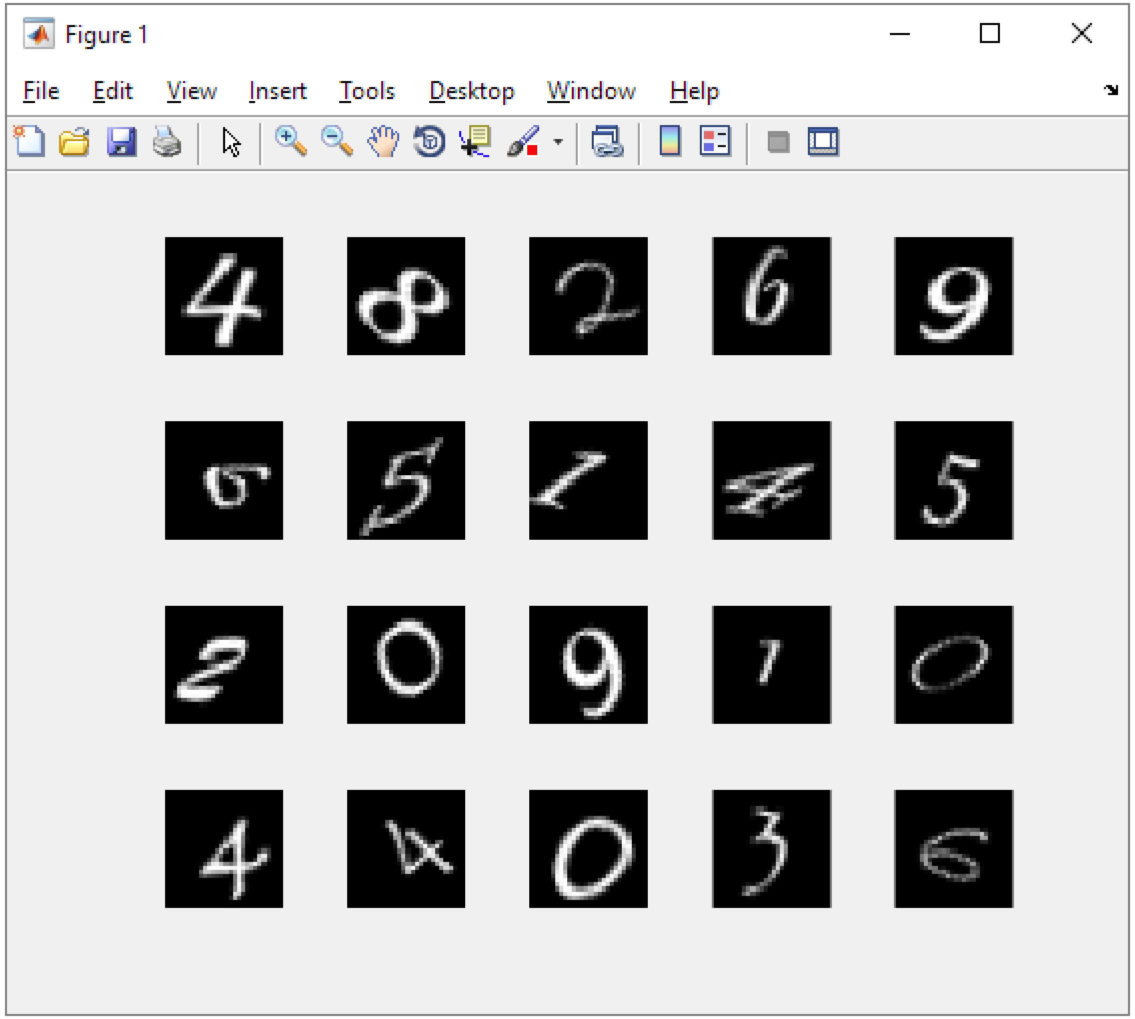
Compare Approaches



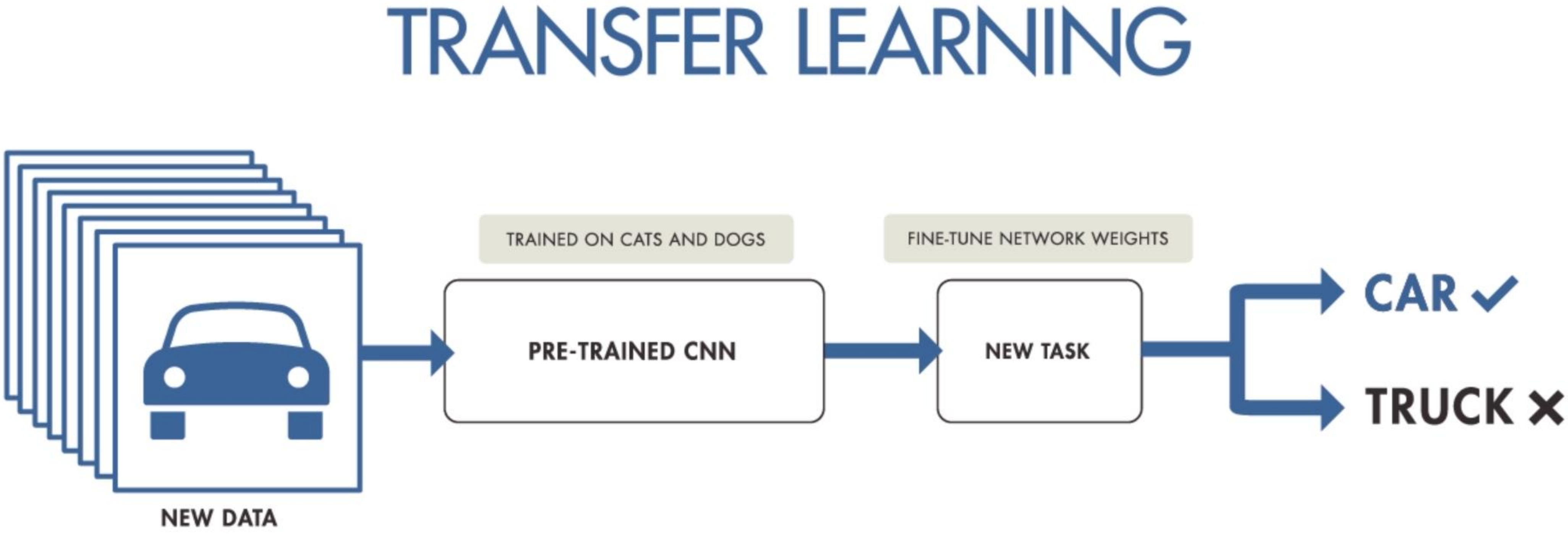
Recommended only when:

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 overfit to small datasets)

Demo : Fine-tune a pre-trained model (transfer learning)



Compare Approaches



Recommended when:

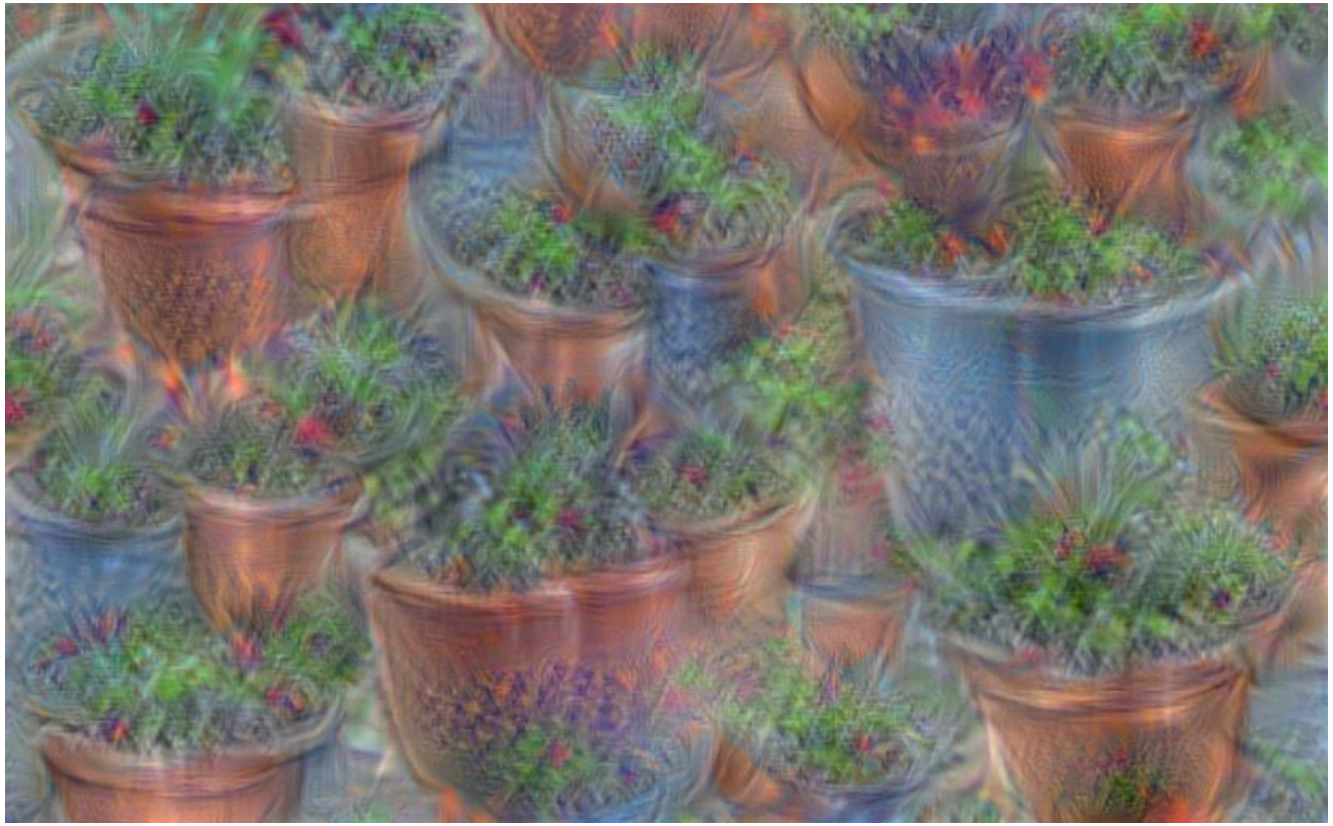
Training data	100s to 1000s of labeled images (small)
Computation	Moderate computation (GPU optional)
Training Time	Seconds to minutes
Model accuracy	Good, depends on the pre-trained CNN model

Available pre-trained CNNs

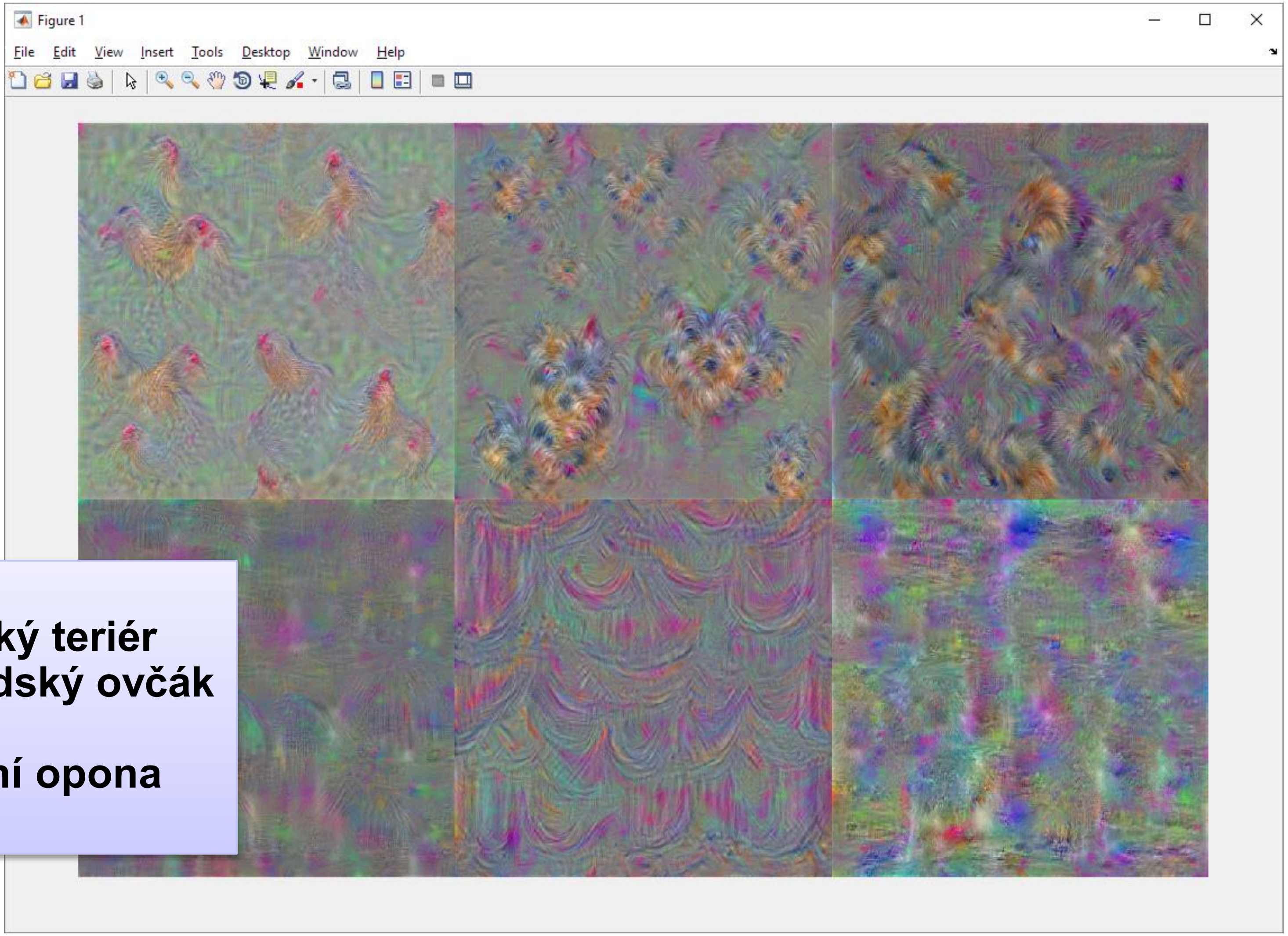
- **AlexNet**
  - The AlexNet model is trained on more than a million images and can classify images into 1000 object categories
- **VGG-16 and VGG-19**
  - VGG-16 and VGG-19 are both trained using the same data set as AlexNet
  - VGG-16 has 41 layers, 16 layers with learnable weights
  - VGG-19 has 47 layers, 19 layers with learnable weights
- `importCaffeNetwork`
  - many pretrained networks available in Caffe Model Zoo
- `importCaffeLayers`
  - import the network architectures of Caffe networks, without importing the pretrained network weights

Verification using Deep Dream Images

- **Visualize what the learned features look like**
- **Generate images that strongly activate a particular channel of the network layers**
- **function `deepDreamImage`**

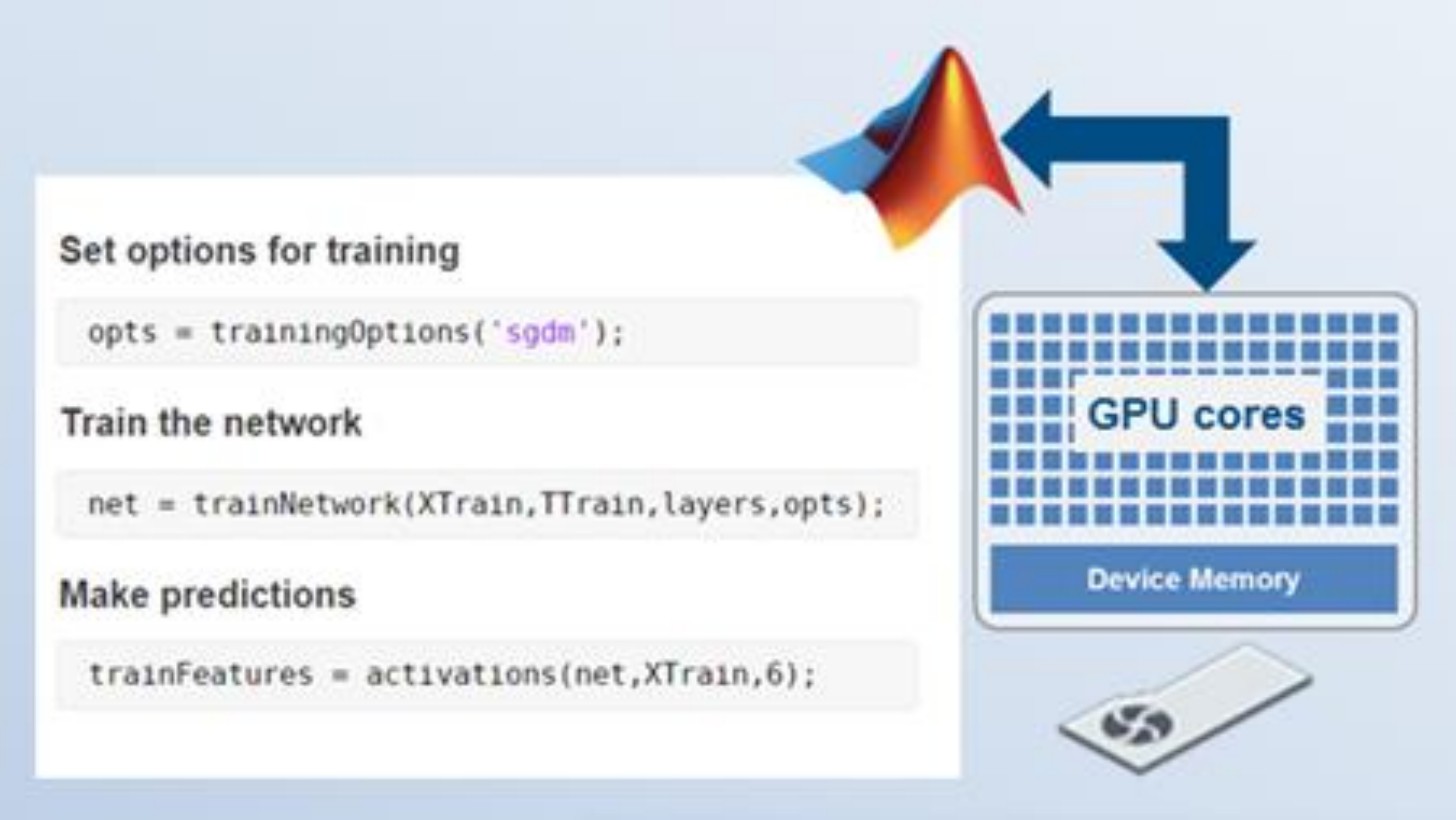


Demo : Deep Dream Images Using AlexNet



- slepice
- jorkšírský teriér
- shetlandský ovčák
- kašna
- divadelní opona
- gejzír

Accelerating Deep Learning Models with GPUs



Set options for training

```
opts = trainingOptions('sgdm');
```

Train the network

```
net = trainNetwork(XTrain,TTrain,layers,opts);
```

Make predictions

```
trainFeatures = activations(net,XTrain,6);
```

Deep Learning Models for Regression

- To predict continuous data such as angles and distances in images
- Include a regression layer at the end of the network

```
layers = [imageInputLayer([28 28 1])
          convolution2dLayer(12,25)
          reluLayer()
          fullyConnectedLayer(1)
          regressionLayer()];
options = trainingOptions('sgdm');
convnet = trainNetwork(trainImages,trainAngles,layers,options);
results = predict(convnet,newImages);
```

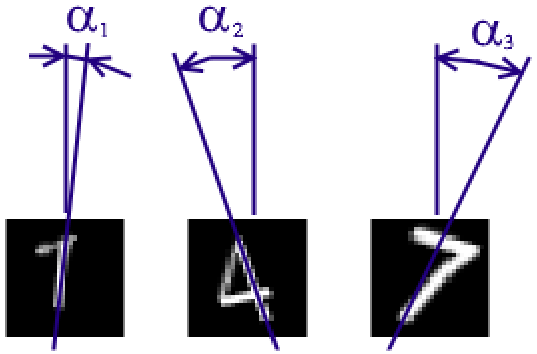
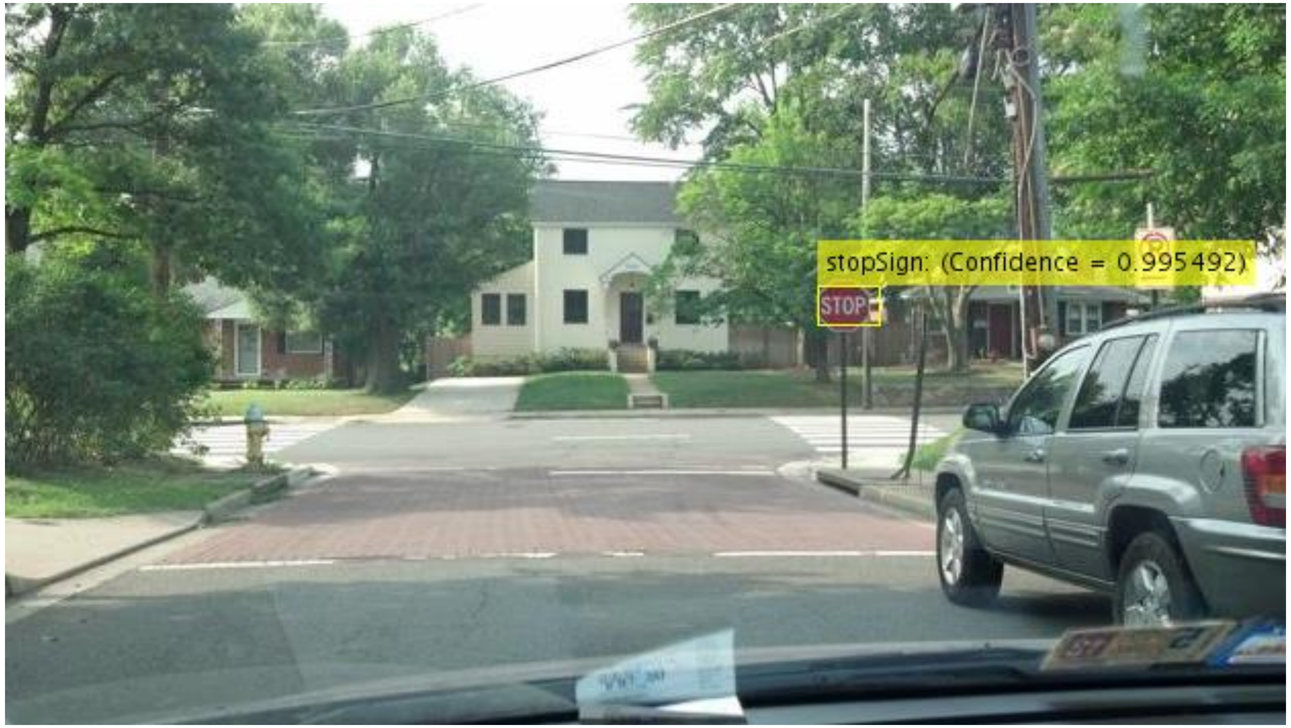


Image Classification vs. Object Detection

- Image Classification
  - classify whole image using set of distinct categories
  - object recognition
  - scene recognition
- Object Detection
  - determine the location of an (small) object in an image
  - multiple objects in one image



Standard Object Detection Algorithms in MATLAB

- **Object detection using extracted features**
  - **edges, corners, SURF, MSER, HOG, LBP, …**
- **Bag of features**
- **Template matching**
- **Image segmentation and blob analysis**
- **Viola-Jones algorithm**

Object Detection using Deep Learning

- **Family of R-CNN object detectors**
  - **Regions with Convolutional Neural Networks**
- **Uses region proposal to detect objects within images**

Detector	Function
R-CNN deep learning detector	`trainRCNNObjectDetector`
Fast R-CNN deep learning detector	`trainFastRCNNObjectDetector`
Faster R-CNN deep learning detector	`trainFasterRCNNObjectDetector`

Choose Among R-CNN, Fast R-CNN, or Faster R-CNN

- **Number of proposed regions ⇨ time it takes to detect objects**
- **Fast R-CNN and Faster R-CNN**
  - **improve detection performance with a large number of regions**

Detector	**Description**
R-CNN deep learning detector	• Less time to train an object detector • Detection time is slow • Allows custom region proposal
Fast R-CNN deep learning detector	• Allows custom region proposal
Faster R-CNN deep learning detector	• Optimal runtime performance • Does not require a custom region proposal

Demo : Object Detection using Deep Learning

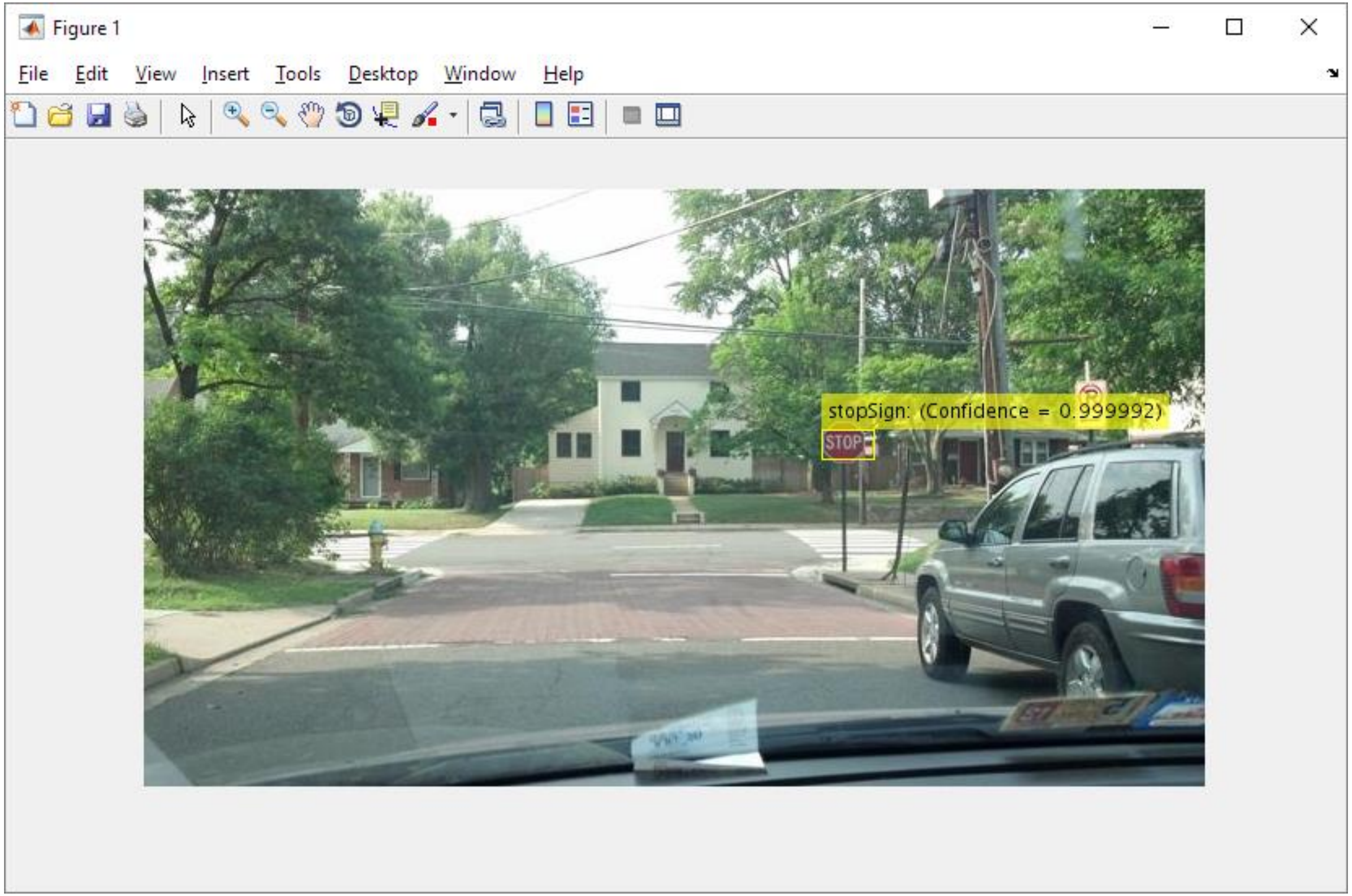


Figure 1
File Edit View Insert Tools Desktop Window Help
stopSign: (Confidence = 0.999992)
STOP

Děkuji za pozornost