

The background features a dark blue field on the left and a lighter blue field on the right, separated by a diagonal line. In the upper right, there are white, stylized wave-like lines. In the lower right, there is a 3D wireframe mesh with a color gradient from yellow at the top to blue at the bottom. On the far right, there are faint, light blue circuit-like patterns.

MATLAB EXPO 2017

Simplifying Image Processing and Computer Vision Application Development

Elza John

Agenda

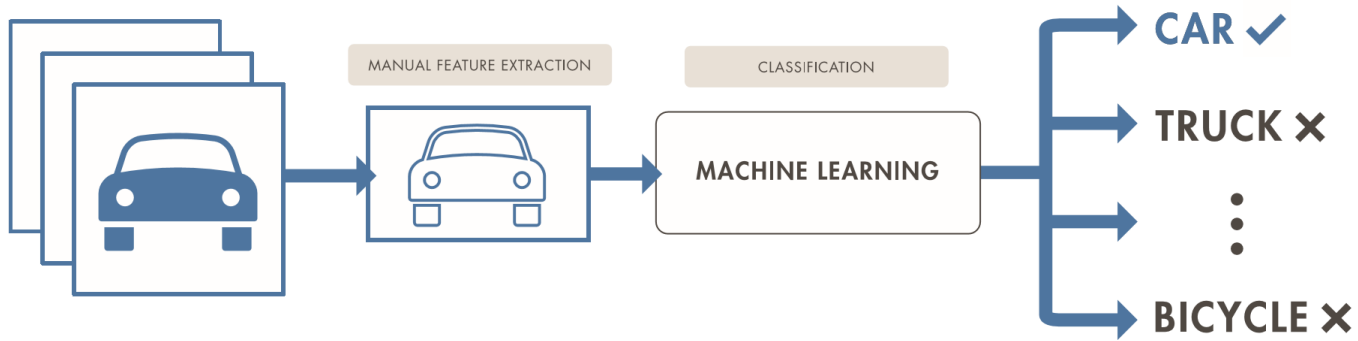
- Deep learning for Computer Vision
- Image processing on 3D data sets

Deep Learning for Computer Vision

***New MATLAB framework makes deep learning easy
and accessible***

Deep Learning is a Subset of Machine Learning

Machine Learning

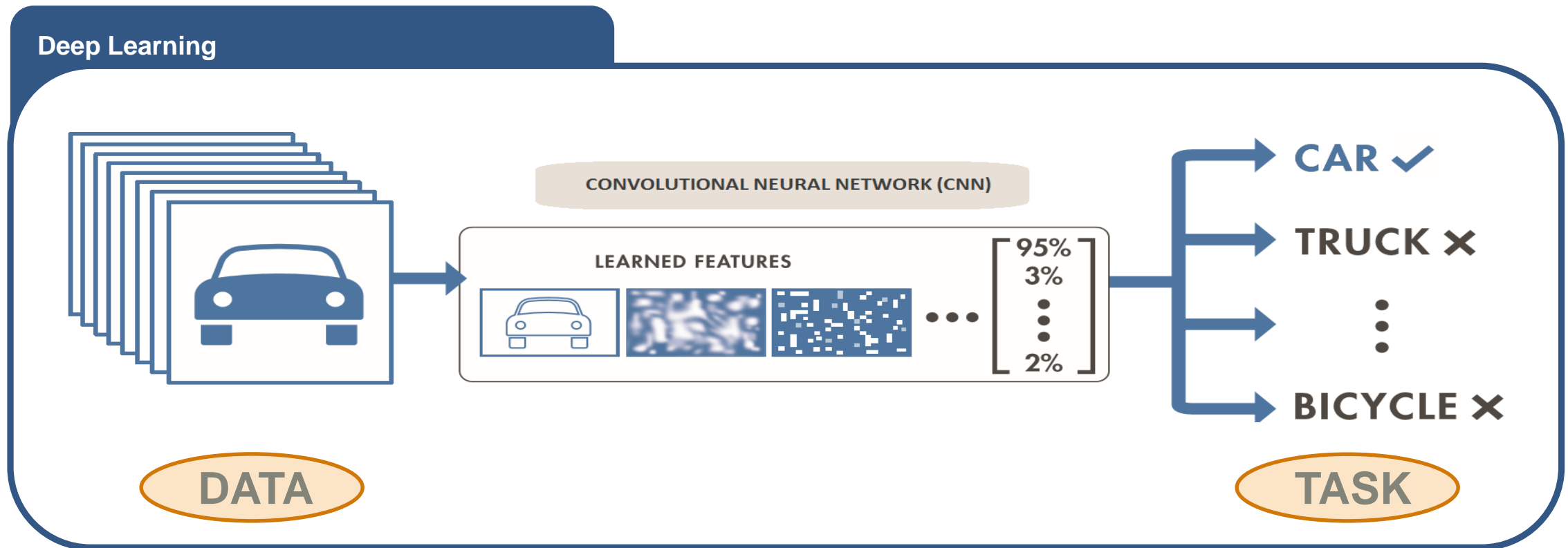


Deep Learning

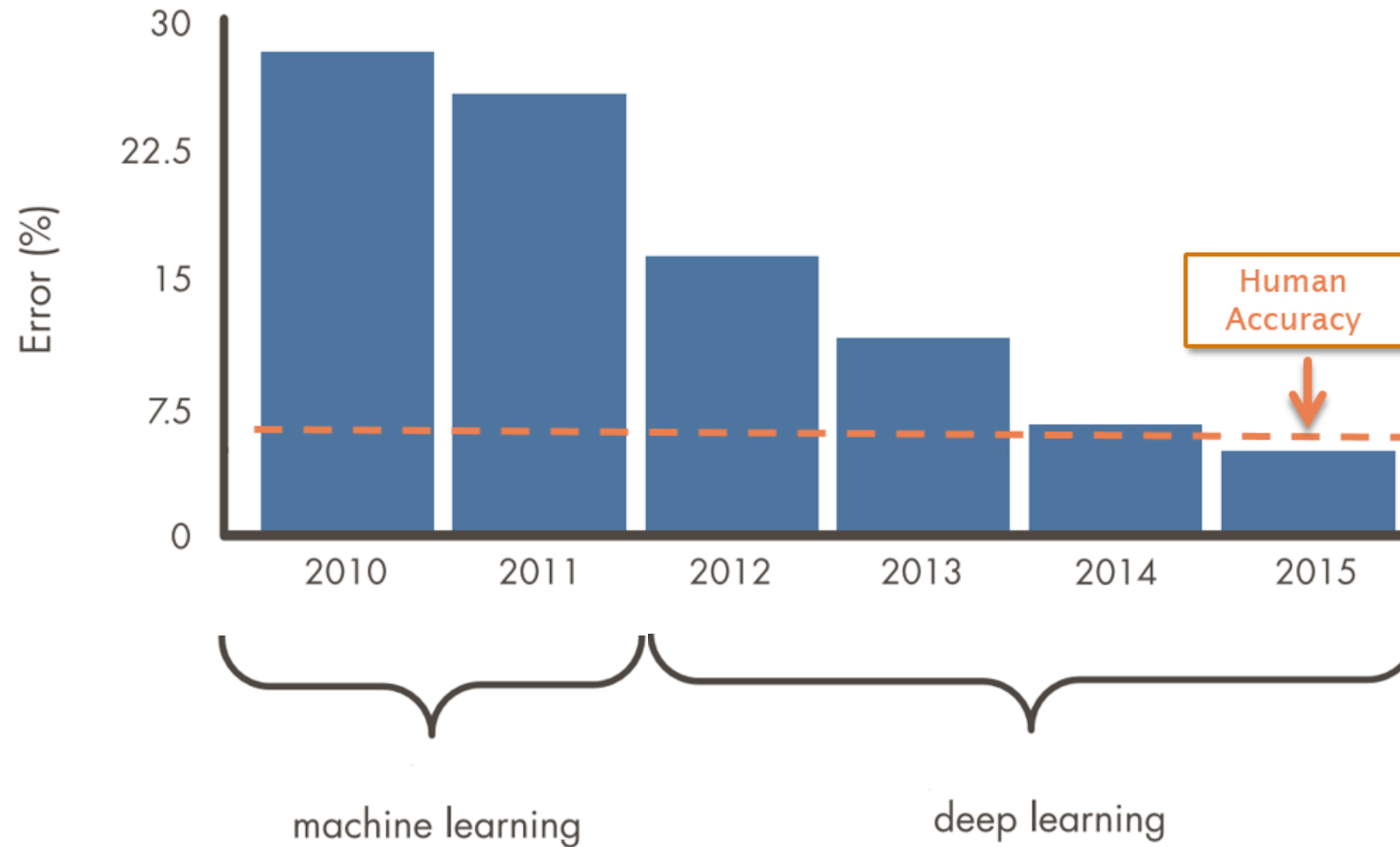


What is Deep Learning ?

Deep learning is a **type of machine learning** that performs **end-to-end learning** by learning **tasks** directly from **images, text, and sound**.



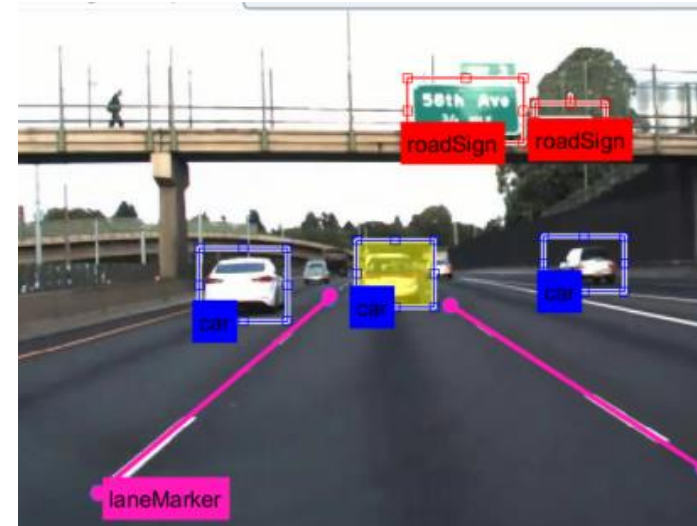
Why is Deep Learning So Popular Now?



Deep Learning Enablers

Acceleration with GPU's

Massive sets of labeled data



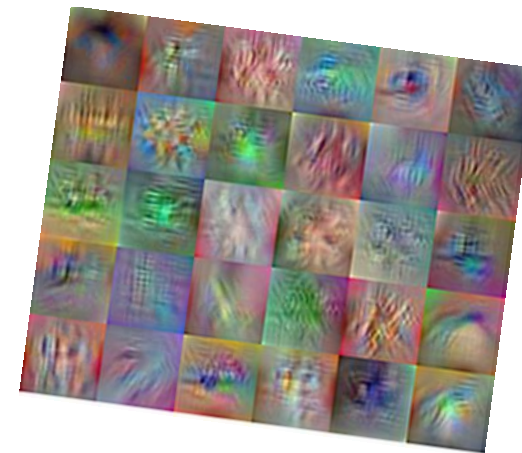
Availability of state of the art models from experts

AlexNet
PRETRAINED MODEL
Caffe
MODELS
VGG-16
PRETRAINED MODEL

MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts



AlexNet
PRETRAINED MODEL
Caffe MODELS
VGG-16
PRETRAINED MODEL

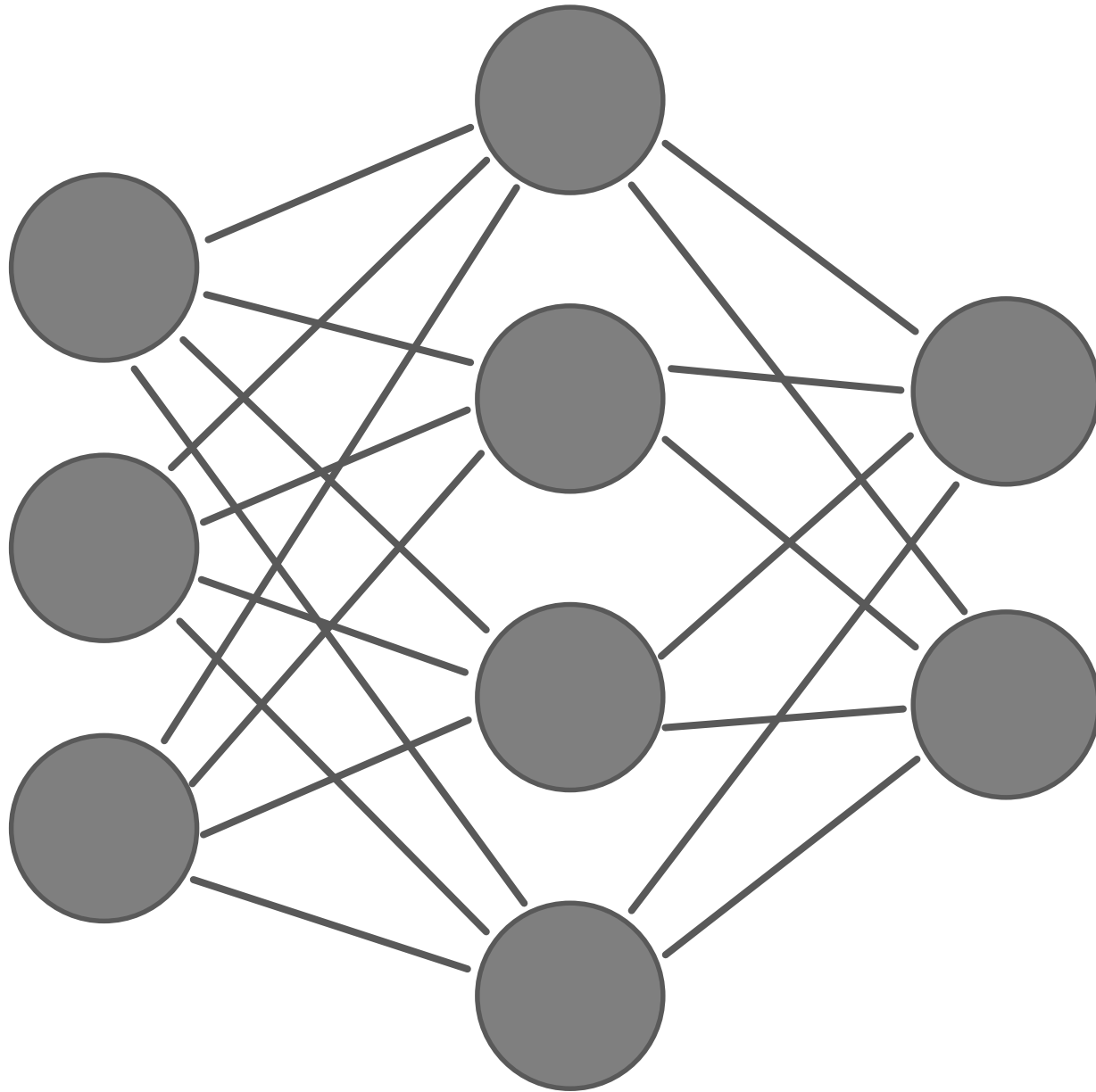


Image classification using pre-trained network

Transfer learning to classify new objects

Locate & classify objects in images and video

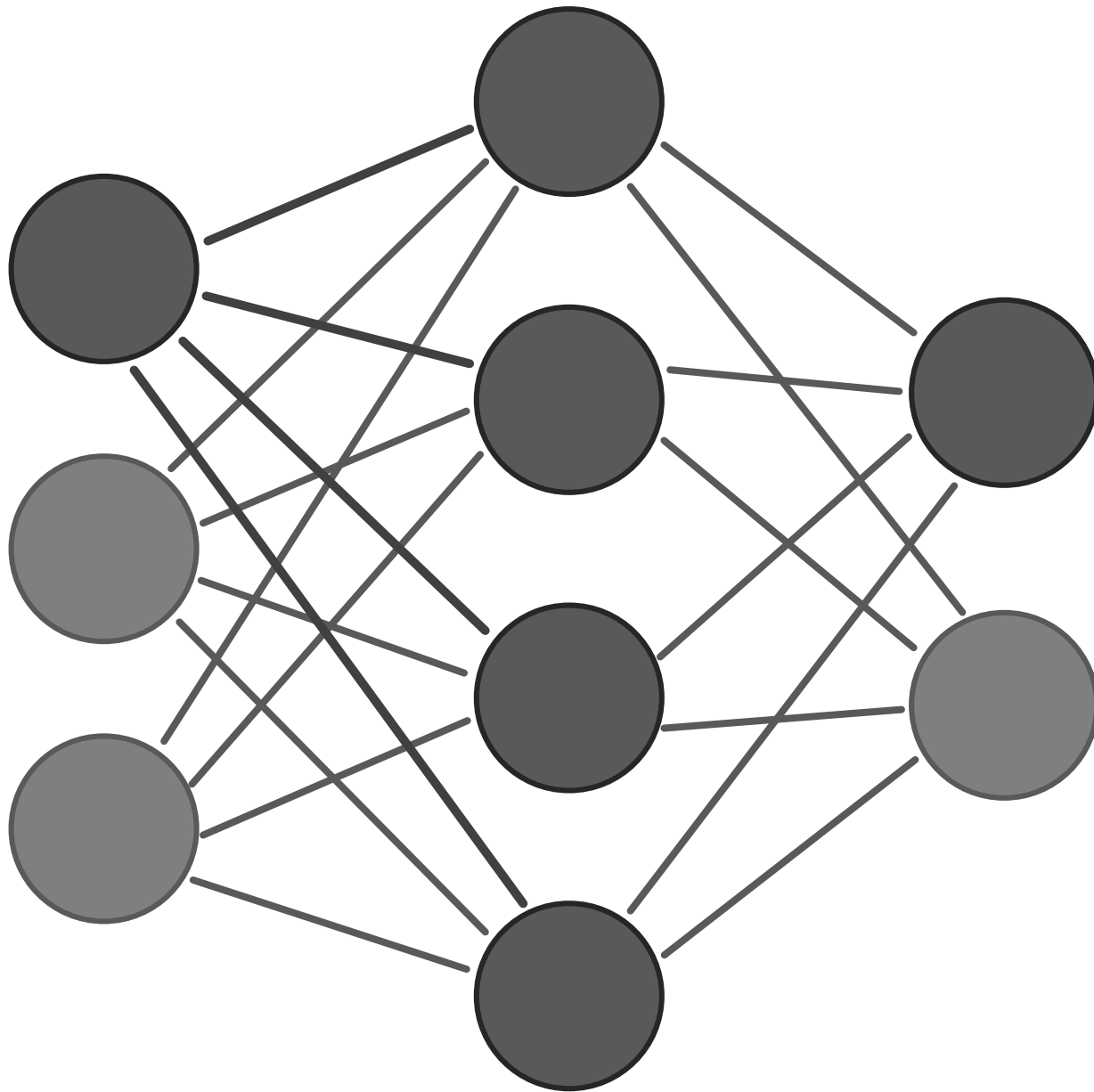


Image classification using pre-trained network

Transfer learning to classify new objects

Locate & classify objects in images and video

Convolutional Neural Networks

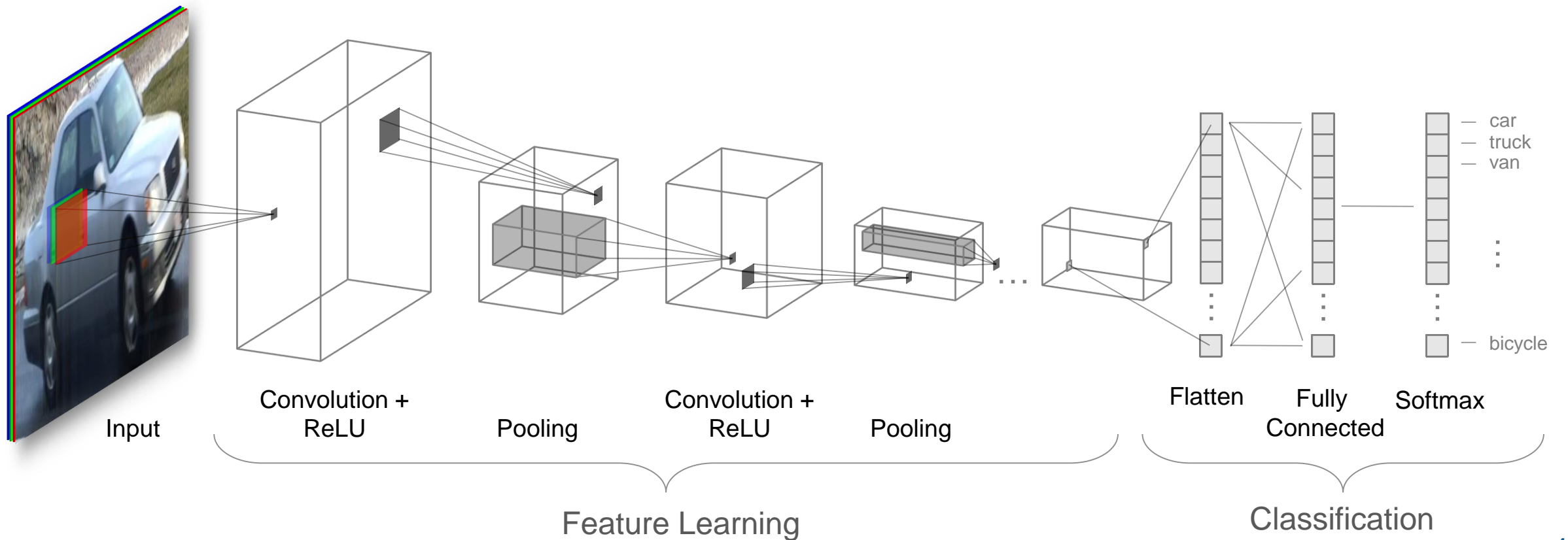
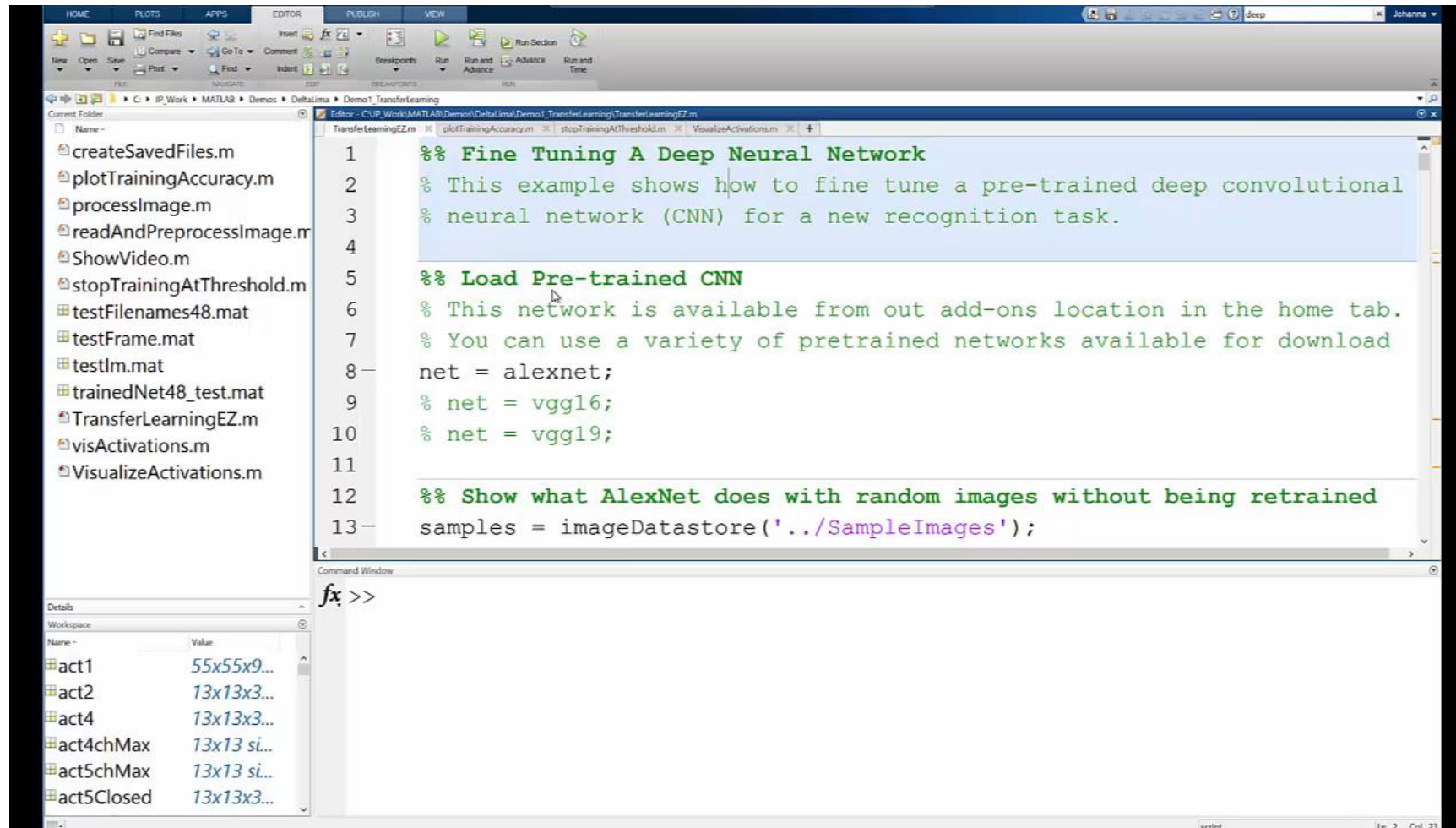


Image Classification Using Pre-trained Network (Video)



The screenshot displays the MATLAB IDE interface. The main editor window shows a script titled "TransferLearningEZ.m" with the following content:

```
1  %% Fine Tuning A Deep Neural Network
2  % This example shows how to fine tune a pre-trained deep convolutional
3  % neural network (CNN) for a new recognition task.
4
5  %% Load Pre-trained CNN
6  % This network is available from our add-ons location in the home tab.
7  % You can use a variety of pretrained networks available for download
8  net = alexnet;
9  % net = vgg16;
10 % net = vgg19;
11
12 %% Show what AlexNet does with random images without being retrained
13 samples = imageDatastore('.../SampleImages');
```

The Command Window shows the MATLAB prompt `fx >>`. The Workspace window displays the following variables:

Name	Value
act1	55x55x9...
act2	13x13x3...
act4	13x13x3...
act4chMax	13x13 si...
act5chMax	13x13 si...
act5Closed	13x13x3...

Agenda

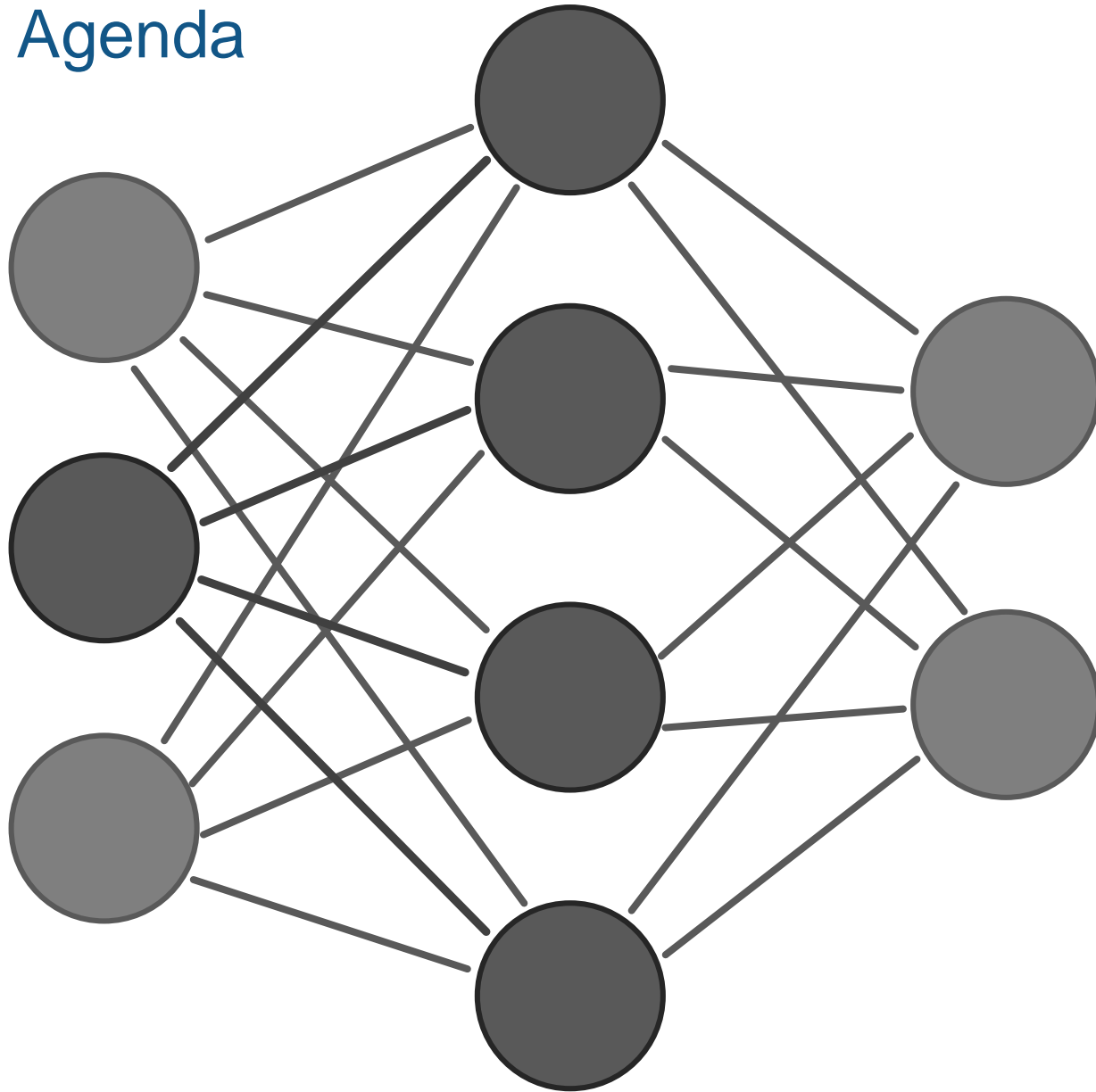


Image classification using pre-trained network

Transfer learning to classify new objects

Locate & classify objects in images and video

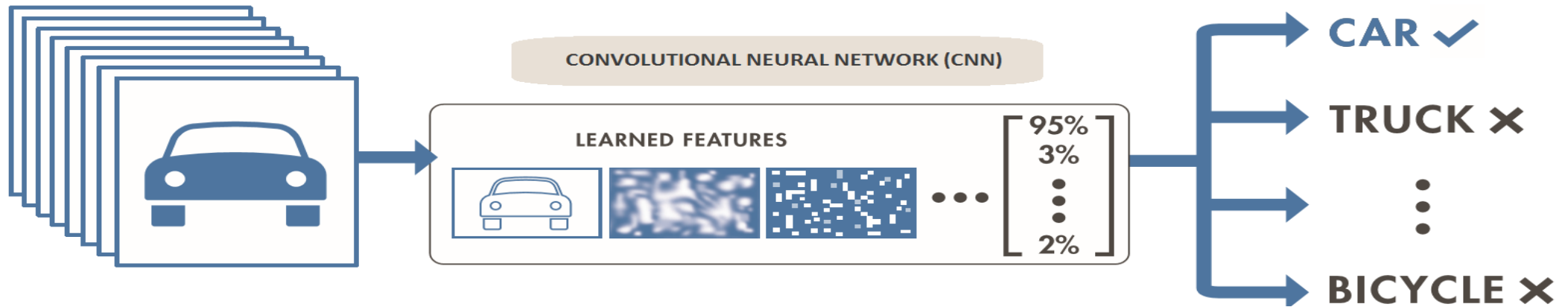
Why should I train my own network?

- 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

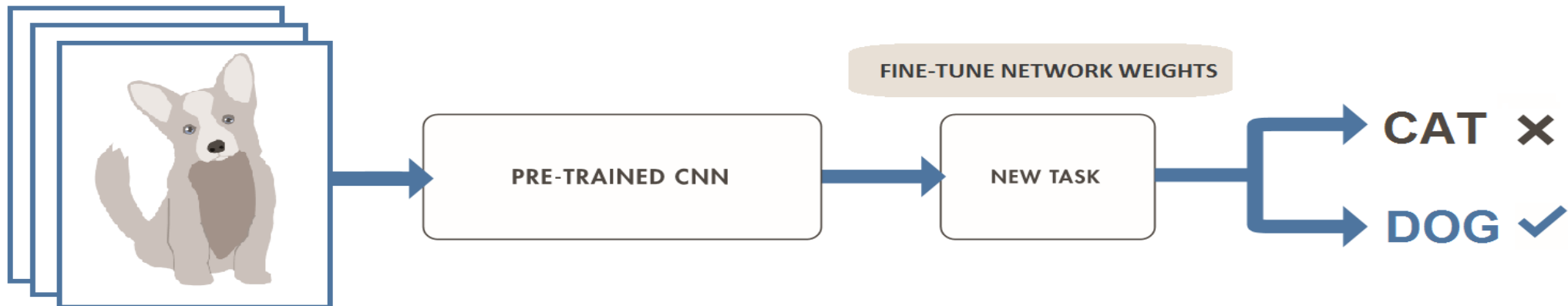


Two Approaches for Deep Learning

1. Train a Deep Neural Network from Scratch



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



Example: Classify Vehicles With Transfer Learning

**AlexNet
Pretrained Model**

1000 classes
Trained on millions of images

Transfer learning – use AlexNet as
starting point

**Vehicle
Classifier
(5 Class)**

Car
SUV
Van
Truck
Large Truck

New Data

Why Perform Transfer Learning

- Requires less data and training time
- Reference models (like AlexNet, VGG-16, VGG-19) have learned rich feature representations for a wide range of images.
- Leverage best network types from top researchers

Transfer Learning to Classify New Objects

The screenshot displays the MATLAB environment with the following components:

- Current Folder:** A list of files including `testim.mat`, `trainedNet48_test.mat`, and `visActivations.m`.
- Editor:** A script named `TransferLearningDemo.m` with the following key sections:
 - Comments: `% The network is training, how can we see inside?`, `% CNNs are historically fairly black box solutions, but we want insight into & the training`, `% Let's look at some options`, `% Plot the accuracy as we are training`, `% Stop training based on certain criteria`.
 - Code: `load('trainedNet48_test.mat');`, `load('testFileNames48.mat');`, `testDS = imageDatastore(testFileNames, 'LabelSource', 'folderNames', ... 'IncludeSubfolders', 1);`, `testDS.ReadFcn = @readAndPreprocessImage;`
 - Comments: `% Test new classifier on validation set`, `% Now run the network on the test data set to see how well it does`, `% If you get an out of memory error for the GPU, please lower the 'MiniBatchSize' to a lower value. (certain GPUs need this to be 1)`.
- Workspace:** A table showing variables:

Name	Value
ans	1x1000 cell
functions	1x2 cell
ii	5
imds	1x1 ImageDatastore
layers	25x1 Layer
net	1x1 SeriesNetwork
num_objects	5
tbl	5x2 table
testDS	1x1 ImageDatastore
trainDS	1x1 ImageDatastore
visImds	1x1 ImageDatastore
- Command Window:** A message: "New to MATLAB? See resources for Getting Started." Below it, a table showing network layers:

Layer Name	Layer Type	Description
'norm2'	Cross Channel Normalization	cross channel normalization with 3 channels per element
'pool2'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
'conv3'	Convolution	384 3x3x256 convolutions with stride [1 1] and padding [1 1]
'relu3'	ReLU	ReLU
'conv4'	Convolution	384 3x3x192 convolutions with stride [1 1] and padding [1 1]
'relu4'	ReLU	ReLU
'conv5'	Convolution	256 3x3x192 convolutions with stride [1 1] and padding [1 1]
'relu5'	ReLU	ReLU
'pool5'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
'fo6'	Fully Connected	4096 fully connected layer
'relu6'	ReLU	ReLU
'drop6'	Dropout	50% dropout
'fo7'	Fully Connected	4096 fully connected layer
'relu7'	ReLU	ReLU
'drop7'	Dropout	50% dropout
'fo8'	Fully Connected	5 fully connected layer
'prob'	Softmax	softmax
'muNeurClassifier'	Classification Output	crossentropyex

Transfer Learning to Classify New Objects

The image shows the MATLAB R2017a environment with a script titled 'TransferLearningDemo.m' open in the Editor. The script is designed to run on a video stream and classify objects using deep learning. The code includes comments and functions for reading frames, processing images, and classifying them.

```

201 % Run on a video stream
202 % please note that a more powerful GPU will cause this to run quickly
203 % and a less powerful GPU may classify objects slowly.
204 while hasFrame(videoReader)
205     %
206     frame = readFrame(videoReader);
207     frame = imresize(frame,[720 1280]);
208     fgMask = step(detector, frame);
209     bbox = step(blcb, fgMask);
210     RGB = frame;
211     for iiii = 1:size(bbox,1)
212         if bbox(iiii,1) > 400 &&...
213             bbox(iiii,2) > 270 &&...
214             bbox(iiii,1) < 800 &&...
215             bbox(iiii,2) < 600 &&...
216             bbox(iiii,4) > 20
217
218             newbox = bbox(iiii,:);
219             temp = imcrop(frame,[newbox(1) newbox(2) newbox(3) newbox(4)]);
220
221             im = processImage(temp);
222
223             [label,score] = classify(dec,im); % classify with deep learning
224
225             RGB = insertObjectAnnotation(RGB,'rectangle',newbox,char(label),...
226                 'TextBoxOpacity',0.9,'FontSize',18);
227
228     end
  
```

The Workspace window on the right shows various variables and their values, including:

- act1: 55x55x96 single
- act2: 13x13x364 single
- ans: 4-D single
- blcb: 1x1 BlobAnalysis
- conf: [0.9518,1.5896e-05,4...
- confMat: 5x5 double
- deep_layer: 1x1x4096 single
- detector: 1x1 ForegroundDetect...
- err_test: 2891x5 single
- frame: 720x1280x3 uint8
- functionum: 1
- functions: 443x443x3 single
- ii: 5
- im: 227x227x3 uint8
- im_display: 56x166x3 uint8
- inds: 1x1 ImageDatastore
- label: 1x1 categorical
- labels: 2891x1 categorical
- layers: 25x1 Layer
- maxEpochs: 48
- minBatchSize: 16
- net: 1x1 SeriesNetwork
- newim: 227x227x3 uint8
- num_objects: 5
- opts: 1x1 TrainingOptionsS...
- position: [10,600]
- position2: [35,550]
- position3: [10,40]
- randNum: 1147
- shapelsorter: 1x1 Shapesorter
- tbl: 5x2 table
- testDS: 1x1 ImageDatastore
- testfilenames: 2891x1 cell
- testim: 227x227x3 uint8
- trainDS: 1x1 ImageDatastore
- tt: 2891x2 table
- videoPlayer: 1x1 VideoPlayer
- videoReader: 1x1 VideoReader
- vidms: 1x1 ImageDatastore

MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts

```
imageDS = imageDatastore(dir)  
Easily manage large sets of images
```

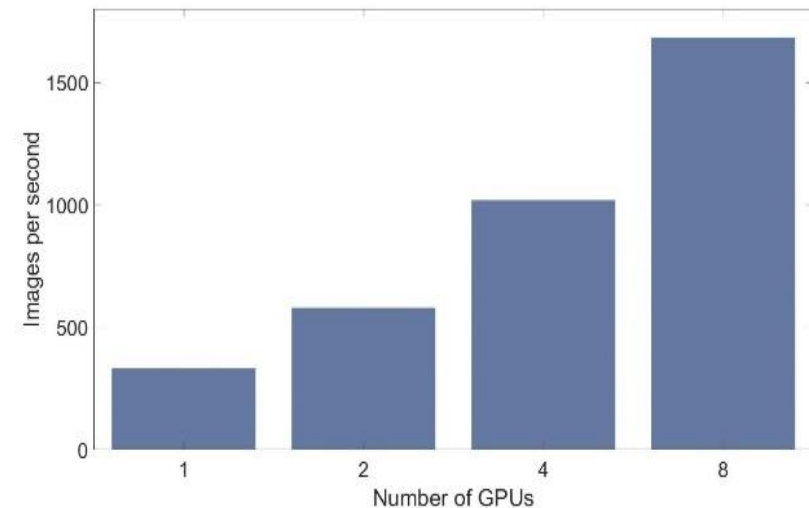
MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts

Training modes supported:

Auto Select (CPU)
GPU
Multi GPU (local)
Multi GPU (cluster)

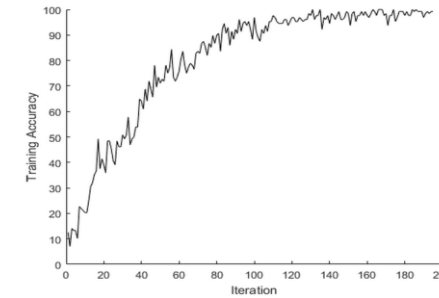


Acceleration with Multiple GPUs

MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts



Training Accuracy Plot



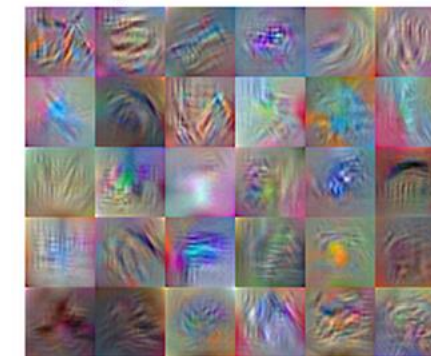
Deep Dream



Network Activations



Layer conv3 Features

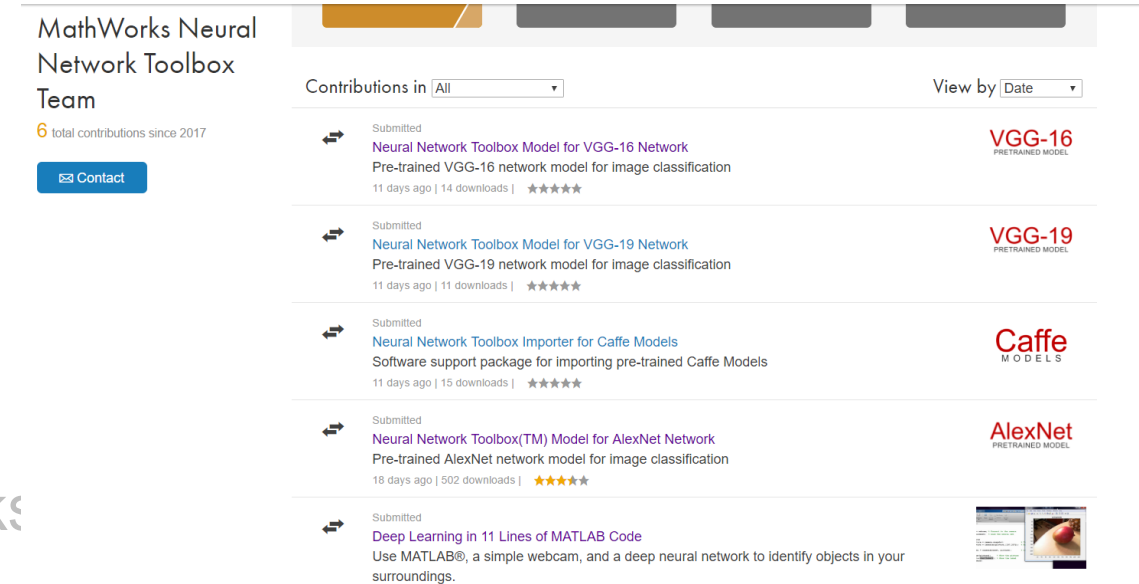


Feature Visualization

MATLAB makes Deep Learning Easy and Accessible

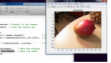
Learn about new MATLAB capabilities to

- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts



MathWorks Neural Network Toolbox Team
6 total contributions since 2017
[Contact](#)

Contributions in View by

Submitted	Neural Network Toolbox Model for VGG-16 Network Pre-trained VGG-16 network model for image classification	VGG-16 PRETRAINED MODEL
Submitted	Neural Network Toolbox Model for VGG-19 Network Pre-trained VGG-19 network model for image classification	VGG-19 PRETRAINED MODEL
Submitted	Neural Network Toolbox Importer for Caffe Models Software support package for importing pre-trained Caffe Models	Caffe MODELS
Submitted	Neural Network Toolbox(TM) Model for AlexNet Network Pre-trained AlexNet network model for image classification	AlexNet PRETRAINED MODEL
Submitted	Deep Learning in 11 Lines of MATLAB Code Use MATLAB®, a simple webcam, and a deep neural network to identify objects in your surroundings.	

Curated Set of Pretrained Models

Access Models with 1-line of MATLAB Code

```
Net1 = alexnet
Net2 = vgg16
Net3 = vgg19
```


Regression Support for Deep Learning

Classification vs. Regression

- Classification – outputs categories/labels
- Regression – outputs numbers

Supported by new regression layer:

```
rouputlayer = regressionLayer('Name', 'rouput')
```

Example predict facial key-points:



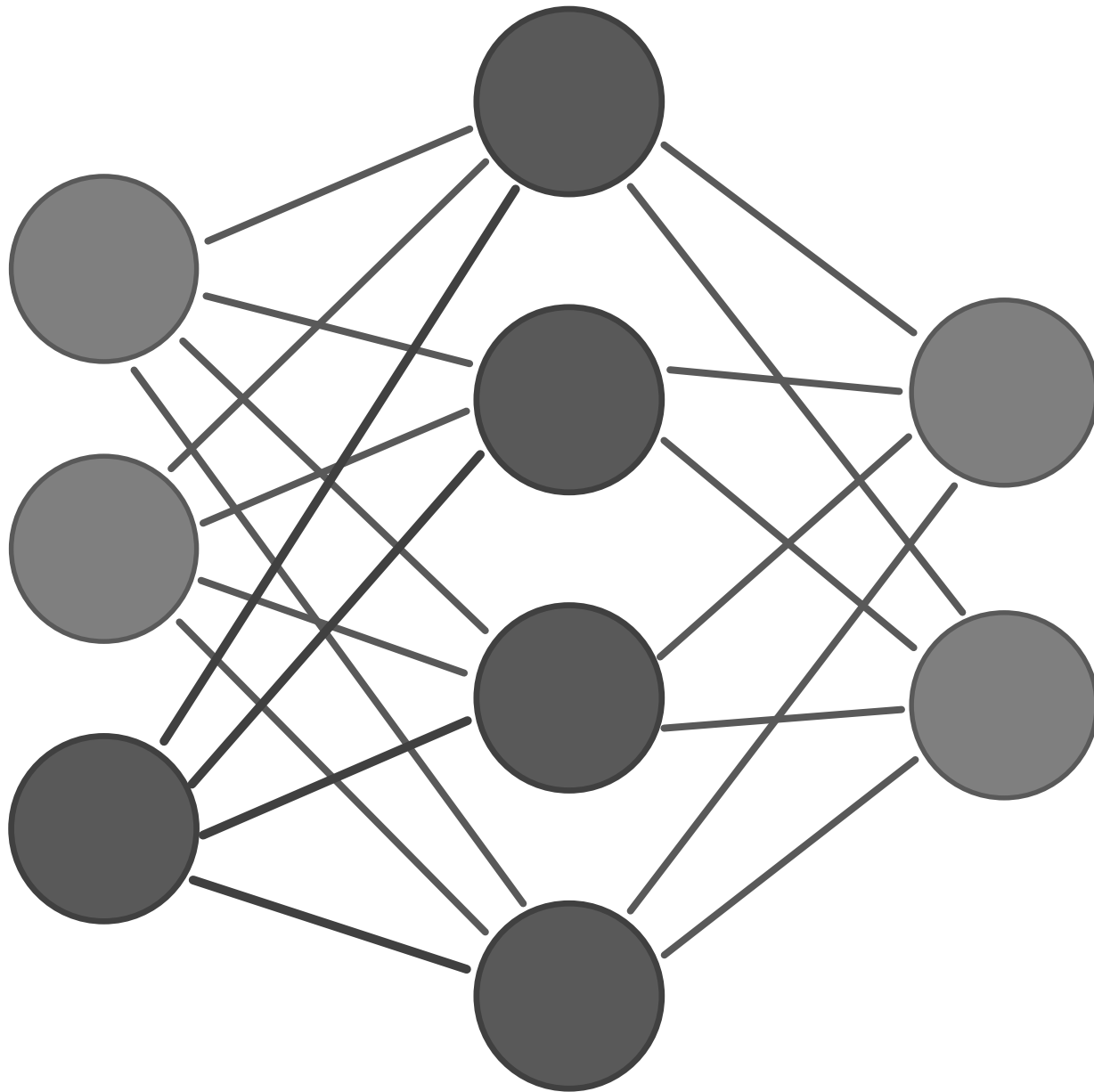


Image classification using pre-trained network

Transfer learning to classify new objects

Locate & classify objects in images and video

Is Object Recognition/Classification Enough ?

Car

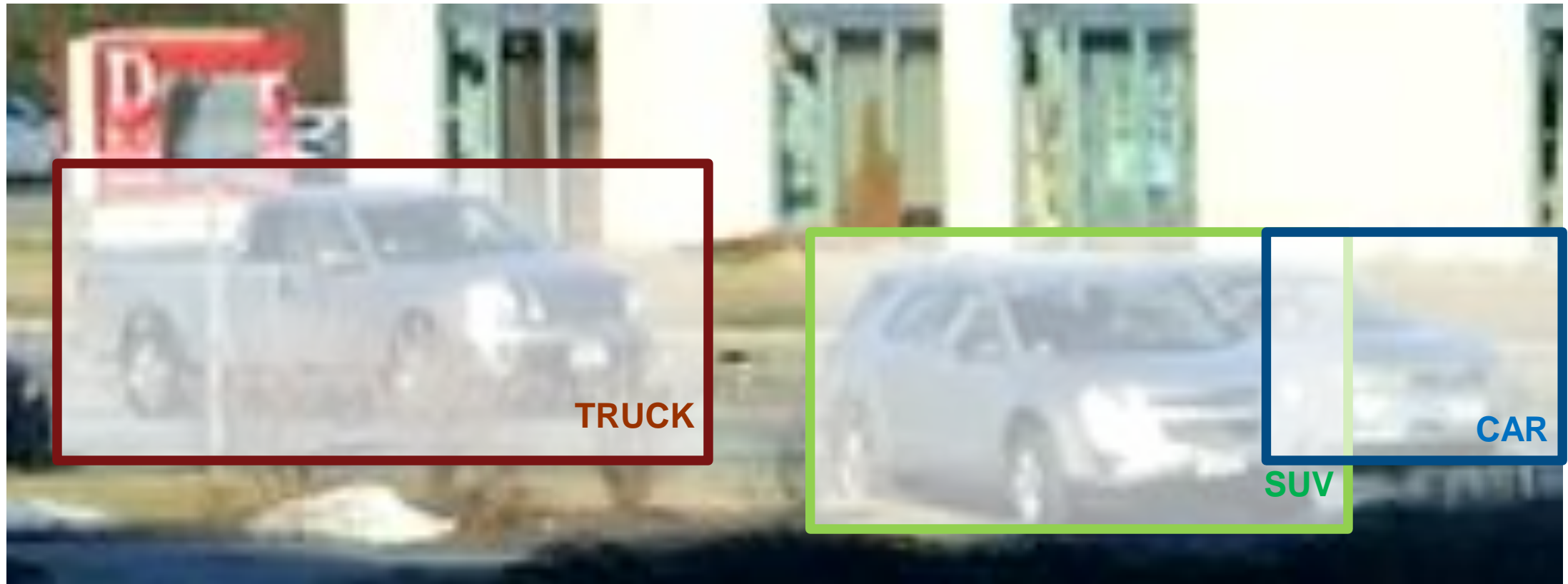


Label for entire image



Car ? SUV? Truck?

Object Detection – Locate and Classify Object



Goal: Create Object Detector to Locate Vehicles

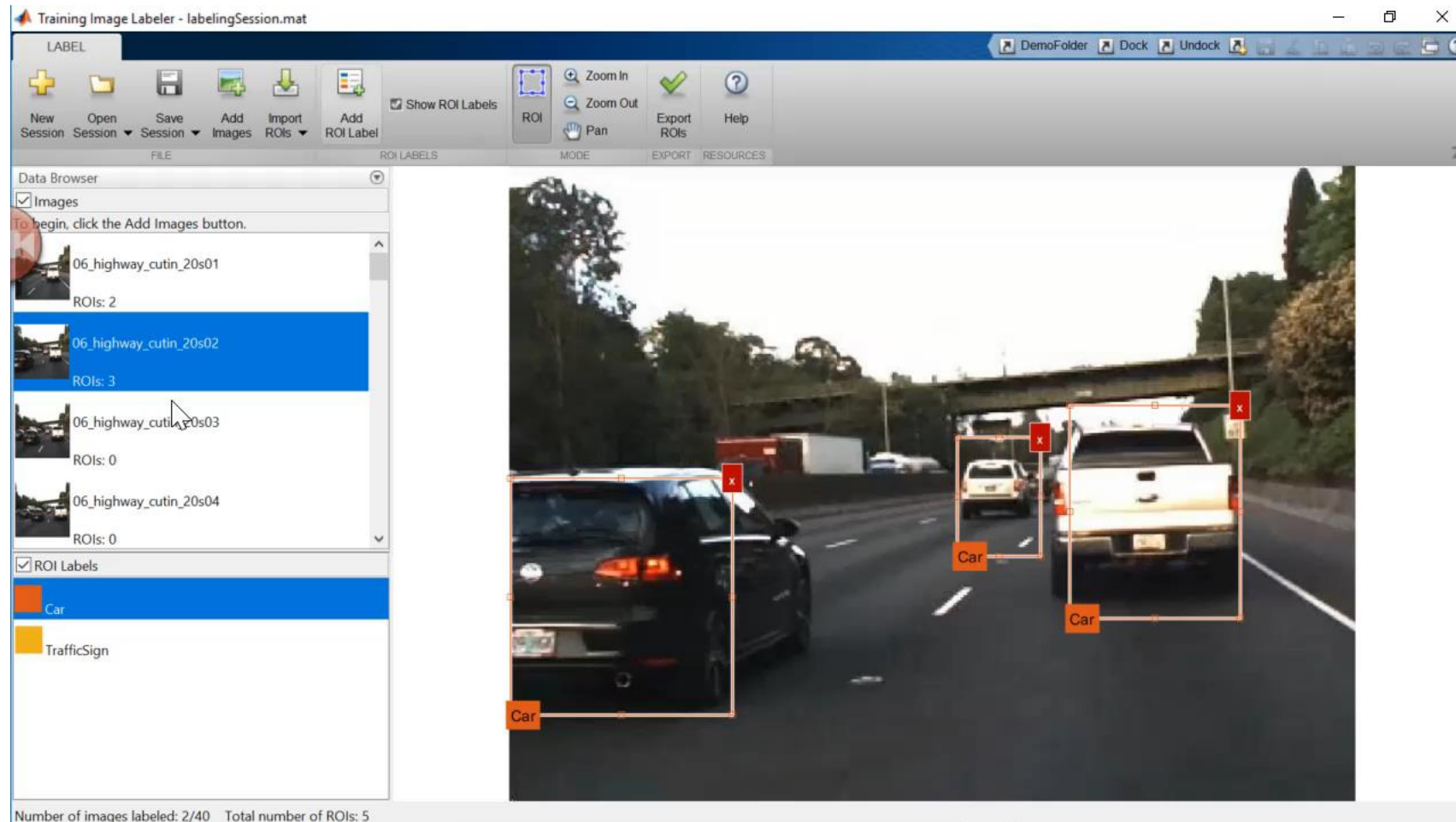


Step 1: Label / Crop data

Step 2: Train detector

Step 3: Use detector

Label Images with MATLAB



Labeling Videos with MATLAB

The screenshot displays the 'Ground Truth Labeler' application window. At the top, a blue arrow points to the 'Automate Labeling' button in the toolbar. The interface is divided into several sections:

- Toolbar:** Includes 'Zoom In', 'Zoom Out', 'Pan', 'Settings', 'Run', 'Stop', 'Undo Run', 'Accept', and 'Cancel' buttons.
- ROI Label Definition:** A list on the left showing 'car' (selected) and 'laneMarker'.
- Scene Label Definition:** A section on the left with 'Sunny' and 'Shadows' labels.
- Video View:** A central window showing a street scene with a red bounding box around a car labeled 'car'.
- Scene Labels:** A panel on the right with checkboxes for 'Sunny' and 'Shadows'.
- Point Tracker:** A panel on the right containing instructions for ROI selection, running the automation, reviewing and modifying labels, changing settings, and accepting/canceling results.
- Timeline:** A bottom control bar with 'Start Time', 'Current', 'End Time', and 'Max Time' fields, along with playback controls and a 'Zoom Out Time Interval' button.

MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts

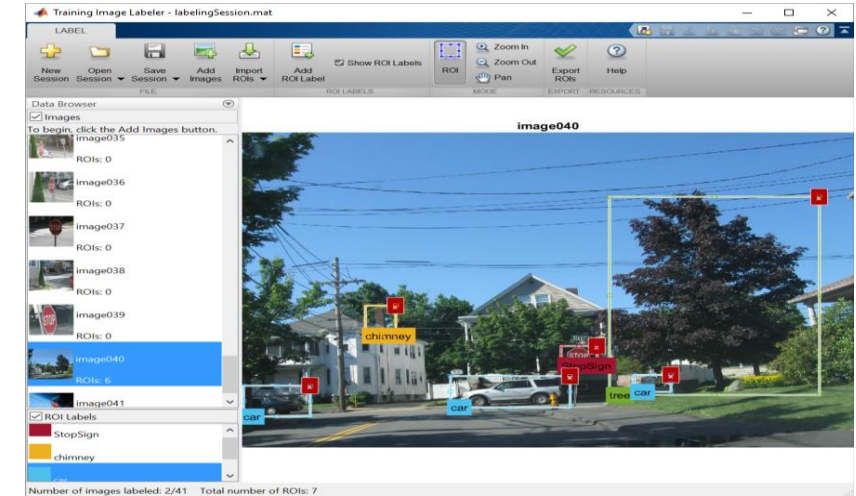
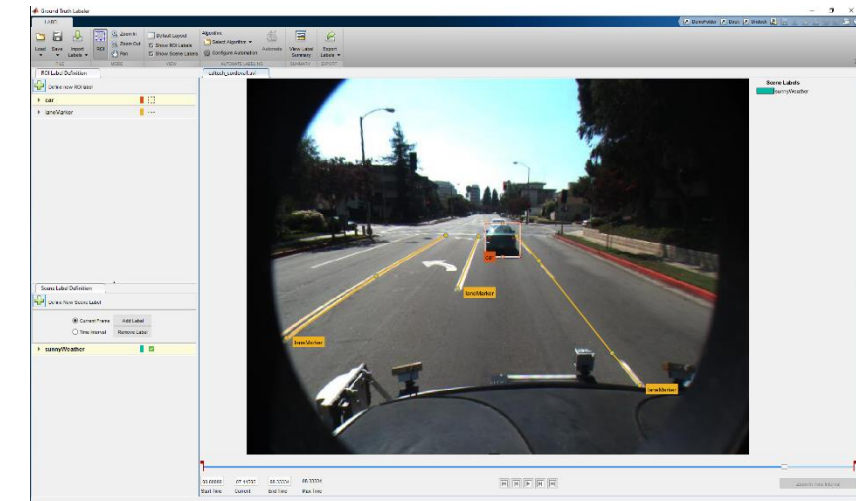


Image Labeler



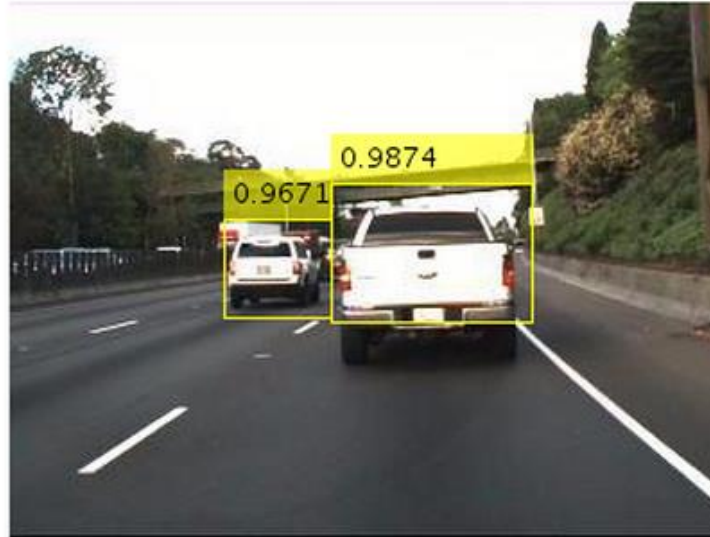
Video Labeler

**New MATLAB framework makes deep learning easy
and accessible**

and
**MATLAB can be used by experts for real deep
learning problems**

Deep Learning Object Detection Frameworks in MATLAB

Detected Vehicles and Detection Scores



Deep Learning

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

Single Line of Code to Train Each Detector

E.g.

```
trainFasterRCNNObjectDetector
```

Object Detection Frameworks in MATLAB

Machine Learning

1. Cascade Object Detector
2. Aggregate Channel Features (ACF)

Deep Learning

1. R-CNN
2. Fast R-CNN
3. Faster R-CNN

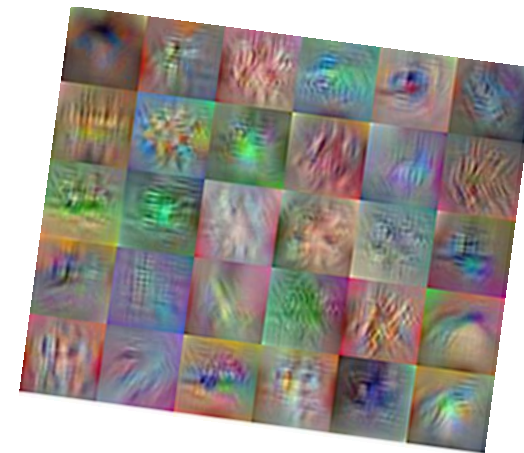


Same labels , train any detector.

MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts



AlexNet
PRETRAINED MODEL
Caffe MODELS
VGG-16
PRETRAINED MODEL

3D Image Processing

3-D Image Processing

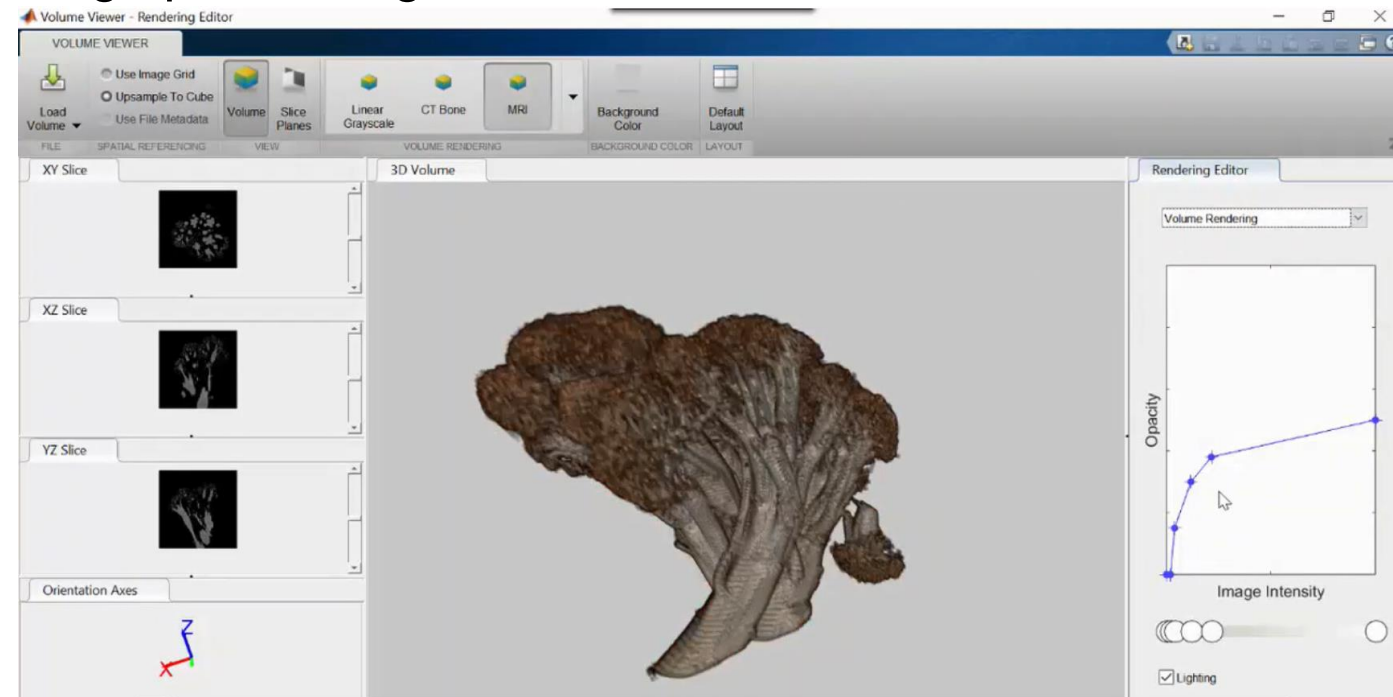
R2016b R2017a

Over 40 functions support 3-D volumetric image processing

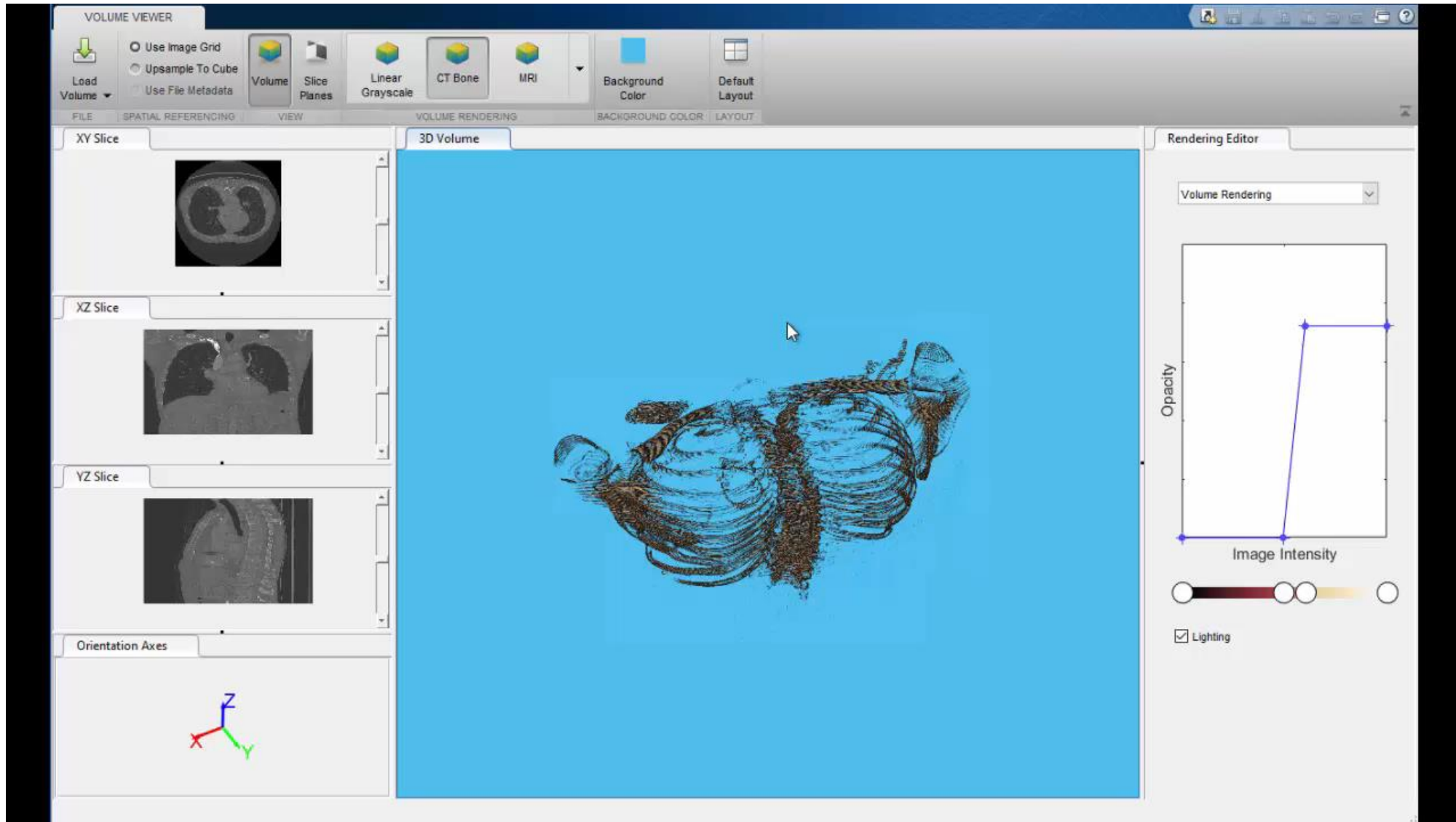
Capabilities Includes:

- Image arithmetic
- Morphology
- Segmentation
- Geometric transforms
- Enhancement

Volume Viewer App for exploration



3-D Image Processing



MathWorks® | *Training Services*

Flexible delivery options:

- Public training available worldwide
- Onsite training with standard or customized courses
- Web-based training with live, interactive instructor-led courses
- Self-paced interactive online training



ENHANCE YOUR SKILLS

ADVANCE YOUR CAREER

More than 30 course offerings:

- Introductory and intermediate training on MATLAB, Simulink, Stateflow, code generation, and Polyspace products
- Specialized courses in control design, signal processing, parallel computing, code generation, communications, financial analysis, and other areas

Email: training@mathworks.in



Image Processing with MATLAB

This two-day course provides hands-on experience with performing image analysis. Examples and exercises demonstrate the use of appropriate MATLAB® and Image Processing Toolbox™ functionality throughout the analysis process.

Topics include:

- Importing and exporting images
- Analyzing images interactively
- Removing noise
- Aligning images and creating a panoramic scene
- Detecting edges, lines, and circles in an image
- Segmenting objects based on their color and texture
- Modifying objects' shape using morphological operations
- Measuring shape properties
- Performing batch analysis over sets of images

 **MathWorks®** | *Training Services*

Computer Vision with MATLAB

This one-day course provides hands-on experience with performing computer vision tasks. Examples and exercises demonstrate the use of appropriate MATLAB® and Computer Vision System Toolbox™ functionality

Topics include:

- Importing, displaying and annotating images and videos
- Detecting, extracting and matching object features
- Automatically aligning images using geometric transformations
- Detecting objects in images and videos
- Tracking objects and estimating their motion in a video
- Removing lens distortion from images
- Measuring planar objects



Accelerating and Parallelizing MATLAB Code



This two-day course covers a variety of techniques for making your MATLAB® code run faster. If you are working with long-running simulations, you will benefit from the hands-on demonstrations and exercises in the course

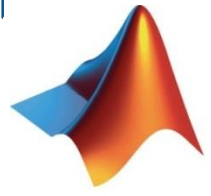
Topics include:

- Improving performance within core MATLAB
- Generating MEX-files
- Parallelizing computations
- Offloading execution
- Working with clusters
- GPU computing

MathWorks Training

 **Guaranteed to run**

Upcoming Public Trainings	Dates	Location
Image Processing with MATLAB	May 24 – 25 	Bangalore
Computer Vision with MATLAB	May 26 	Bangalore
Machine Learning with MATLAB	July 10 - 11	Hyderabad
Machine Learning with MATLAB	Sept 26 - 27	Pune



MathWorks®

Accelerating the pace of engineering and science

Speaker Details

Email: Elza.John@mathworks.in

Contact MathWorks India

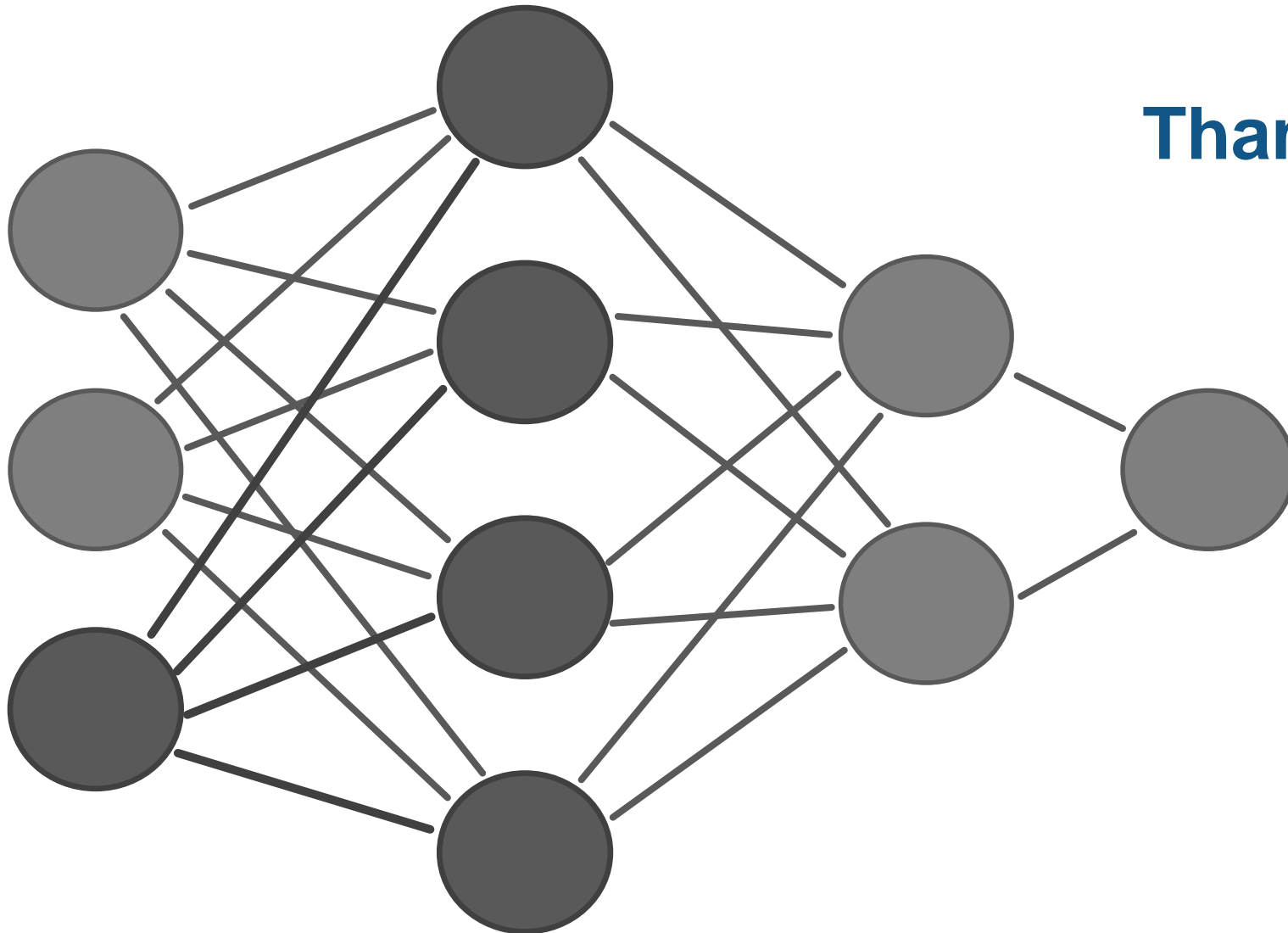
Products/Training Enquiry Booth

Call: 080-6632-6000

Email: info@mathworks.in

Your feedback is valued.

Please complete the feedback form provided to you.



Thank You