



# Deep Learning Overview

*Images and Video*



# Agenda

- I. Deep learning in engineering and science
- II. Developing a deep learning solution in MATLAB
- III. MathWorks deep learning support



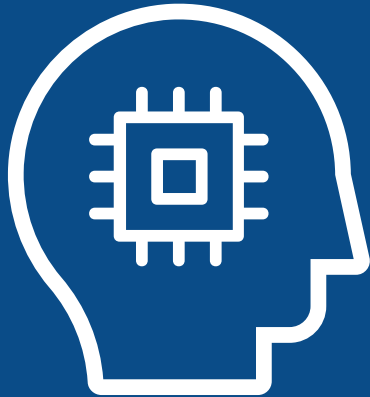
# Agenda

- I. Deep learning in engineering and science
- II. Developing a deep learning solution in MATLAB
- III. MathWorks deep learning support

# Deep learning is a key technology driving the AI megatrend

## ARTIFICIAL INTELLIGENCE

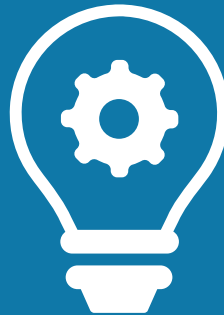
Any technique that enables machines to mimic human intelligence



1950s

## MACHINE LEARNING

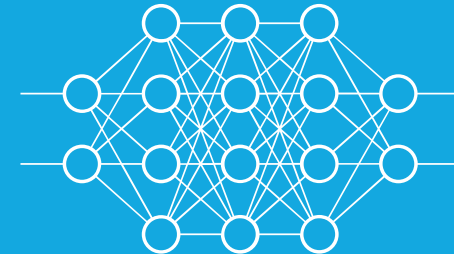
Statistical methods that enable machines to “learn” tasks from data without explicitly programming



1980s

## DEEP LEARNING

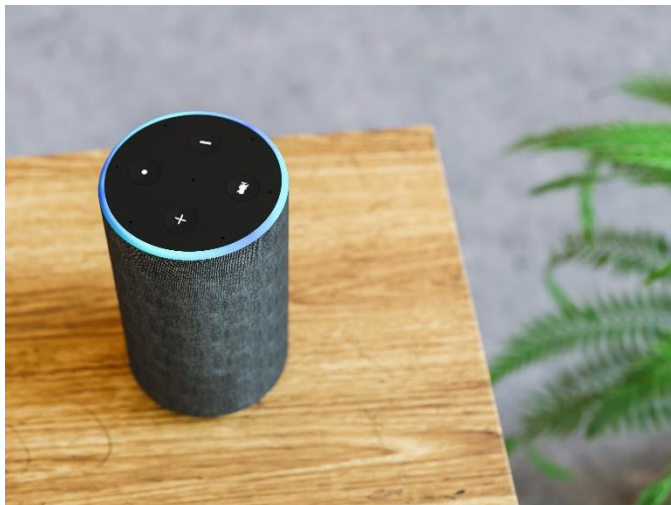
Neural networks with many layers that learn representations and tasks “directly” from data



2010s



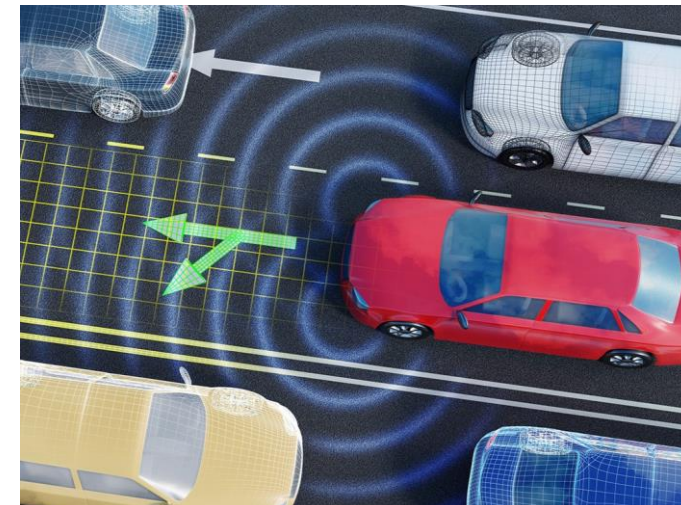
# Deep learning is part of our everyday lives



Speech Recognition



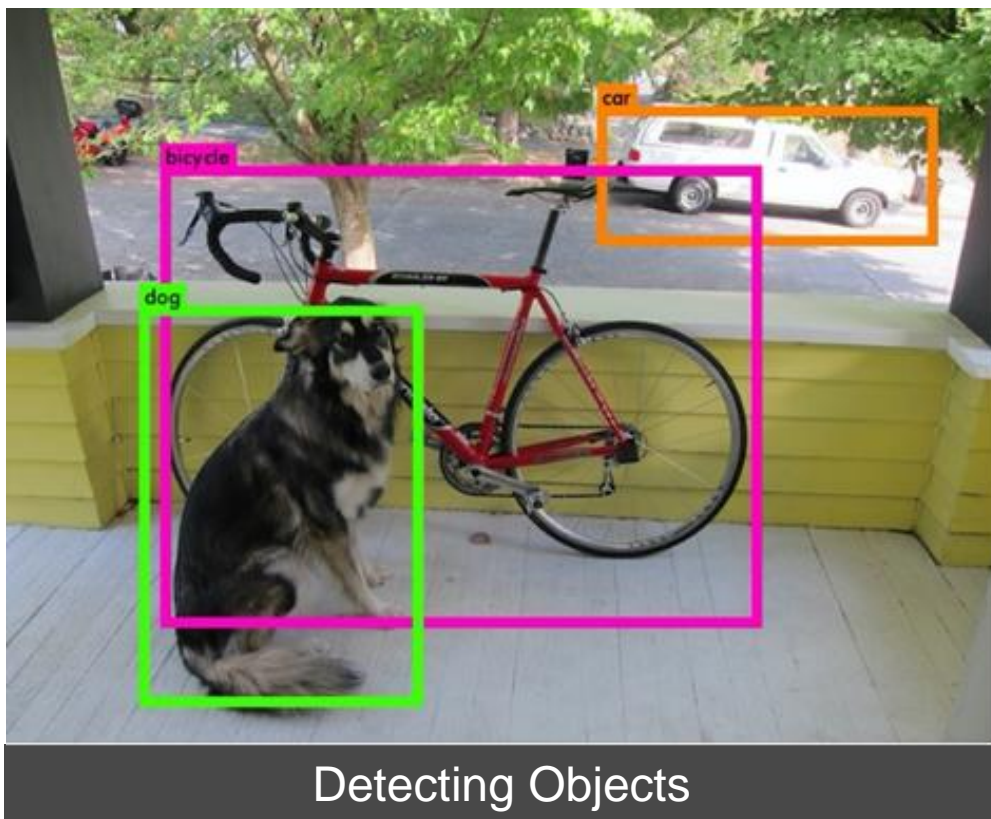
Face Detection



Automated Driving

# Deep learning applications: mainstream vs. engineering

## Mainstream



## Engineering and Science

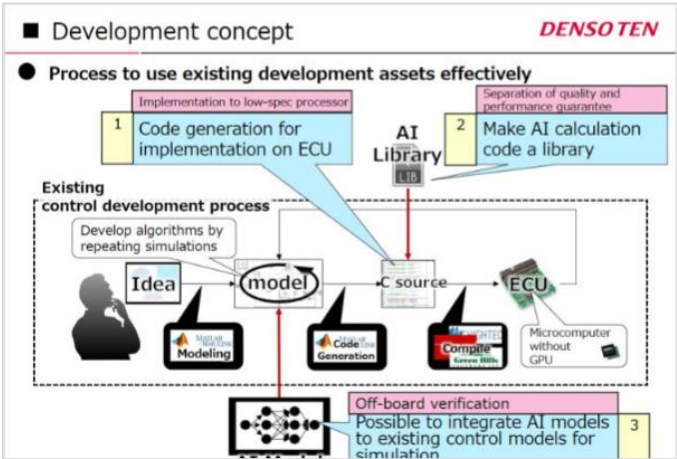


Deep Learning Detection

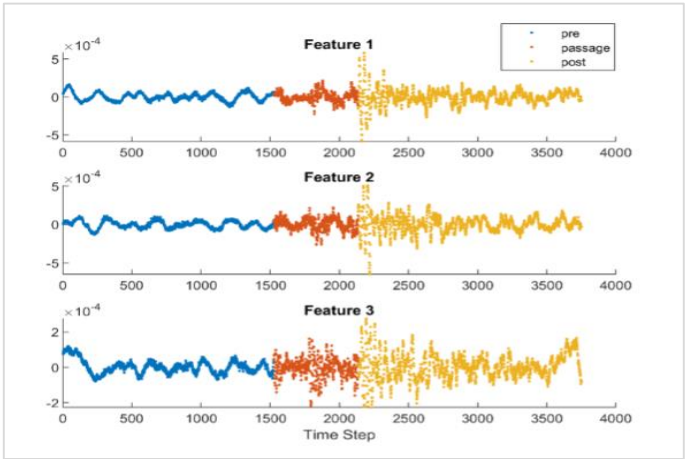
# MATLAB Deep Learning used in Industry



**Automatic Defect Detection**  
Airbus



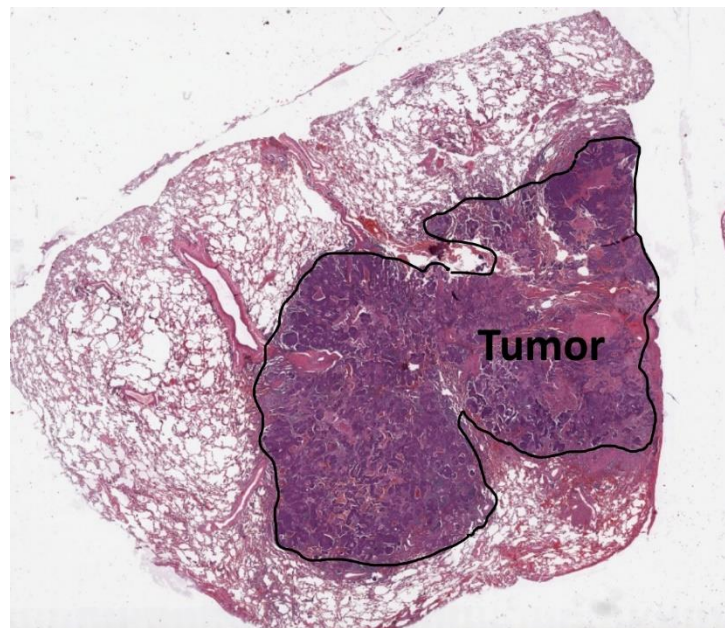
**ECU Vehicle Control**  
Denso



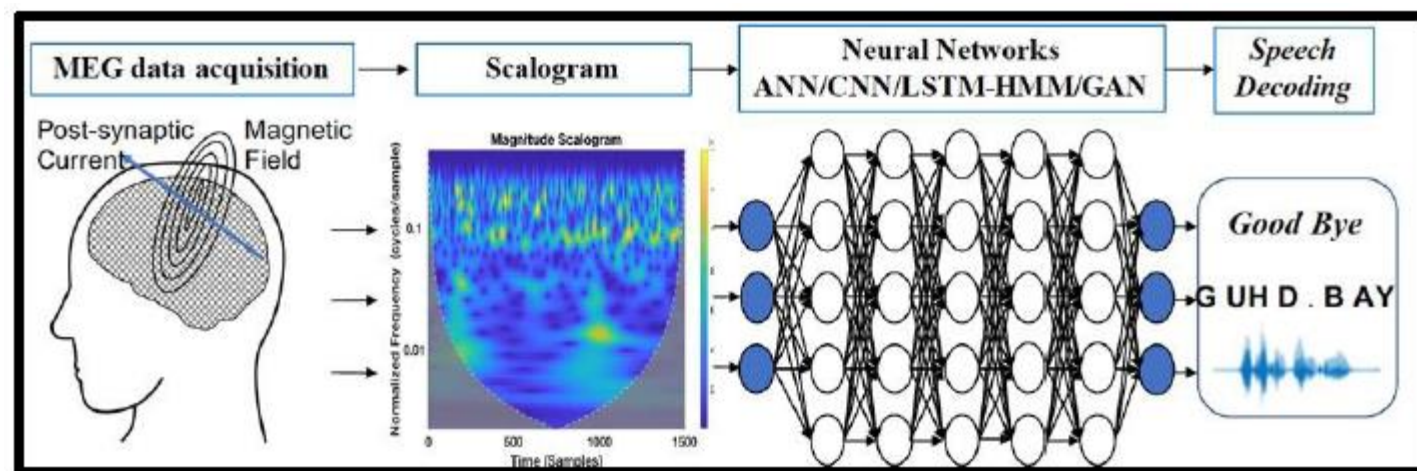
**Seismic Event Detection**  
Shell



# MATLAB Deep Learning used in Research



Predicting gastrointestinal cancer (July 2019)



Converting brain waves to speech to help ALS patients communicate (Nov 2019)



# Evolution of Deep Learning in MATLAB

2016

CNN's  
Pretrained Models  
Caffe Importer

2017

## **Name Change**

- Neural Network Toolbox to Deep Learning Toolbox

## **Algorithms**

- LSTM's
- Directed Acyclic Graphs
- Multi-GPU Training

## **Code Generation**

- GPU Coder

## **Apps**

- Image Labeler

## **Interoperability**

- TensorFlow-Keras Importer

2018

## **Examples**

- Signal Processing
- Audio
- Text Analytics

## **Algorithms**

- Wavelet Scattering

## **Code Generation**

- MATLAB Coder C++

## **Apps**

- Deep Network Designer
- Video Labeler
- Audio Labeler

## **Interoperability**

- ONNX Support

2019

- Reinforcement Learning

## **Algorithms**

- Automatic Differentiation
- Custom Training Loops
- Weight Sharing
- Big Image

## **Examples**

- GANs
- Siamese Network
- Autoencoders
- 3-D support
- **Explainable AI**
  - Occlusion
  - Grad-CAM

## **Code Generation**

- MATLAB Coder (ARM)

## **Apps**

- Signal Labeler

2020

- Deep Learning Data Sets

## **Apps**

- Experiment Manager

## **Examples**

- 5G Communications
- **Over 200+ examples!**

## **Algorithms**

- Point Cloud

## **Code Generation**

- Quantization

# Applications of deep learning for images and video



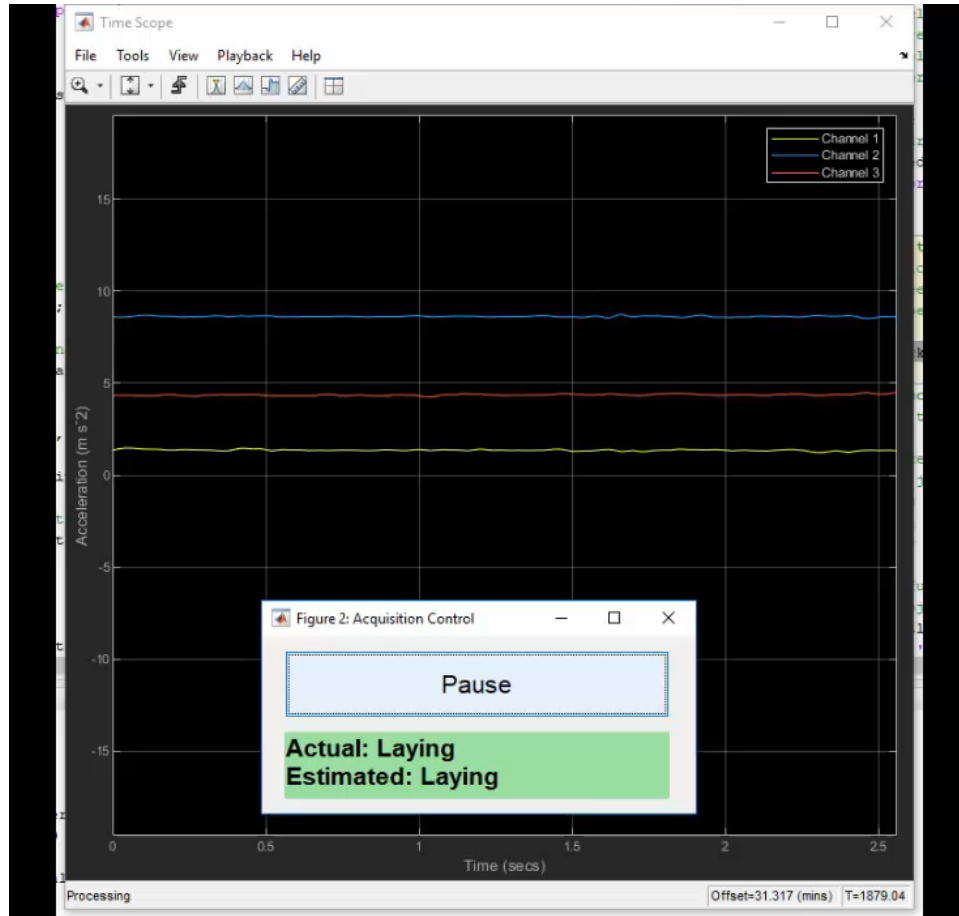
YOLO v2 (You Only Look Once)



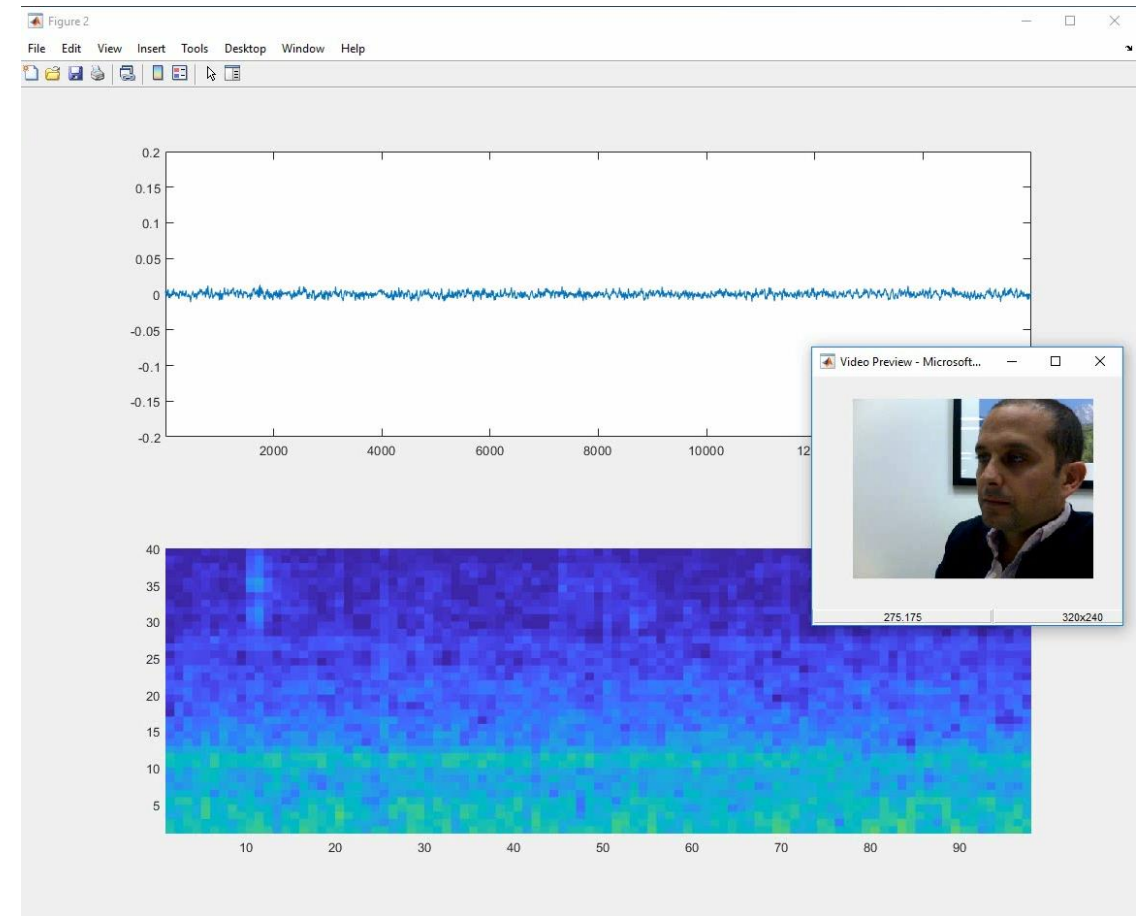
Semantic Segmentation using SegNet



# Applications of deep learning for signal processing



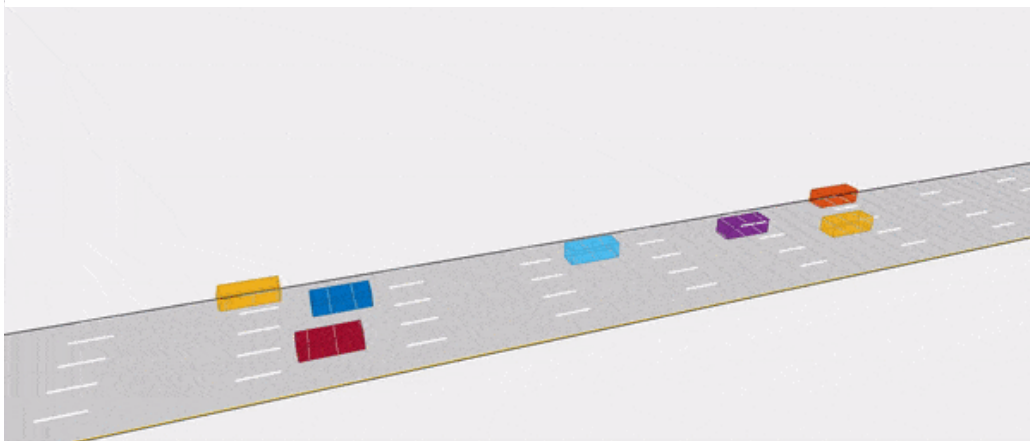
Signal Classification using LSTMs



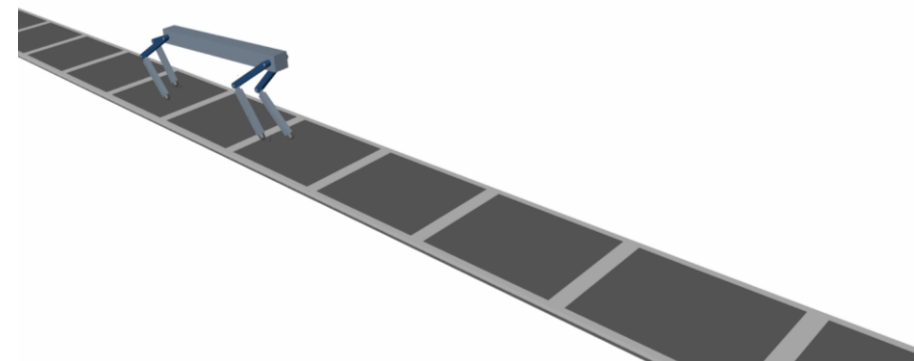
Speech Recognition using CNNs



# Applications of reinforcement learning



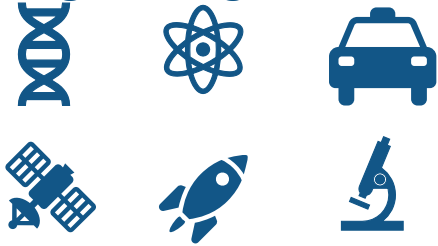
**Teach a car to navigate traffic**



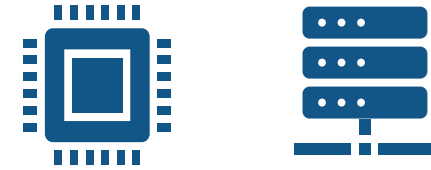
**Train a Quadruped robot to walk**

# Why MATLAB & MathWorks for Deep Learning?

Domain-specialized workflows  
for **engineering and science**



Multi-platform **deployment** of  
full applications and systems



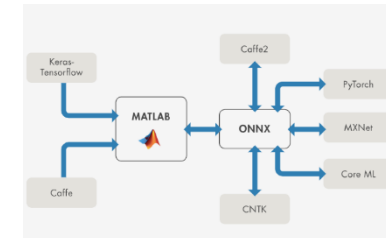
People



Platform productivity



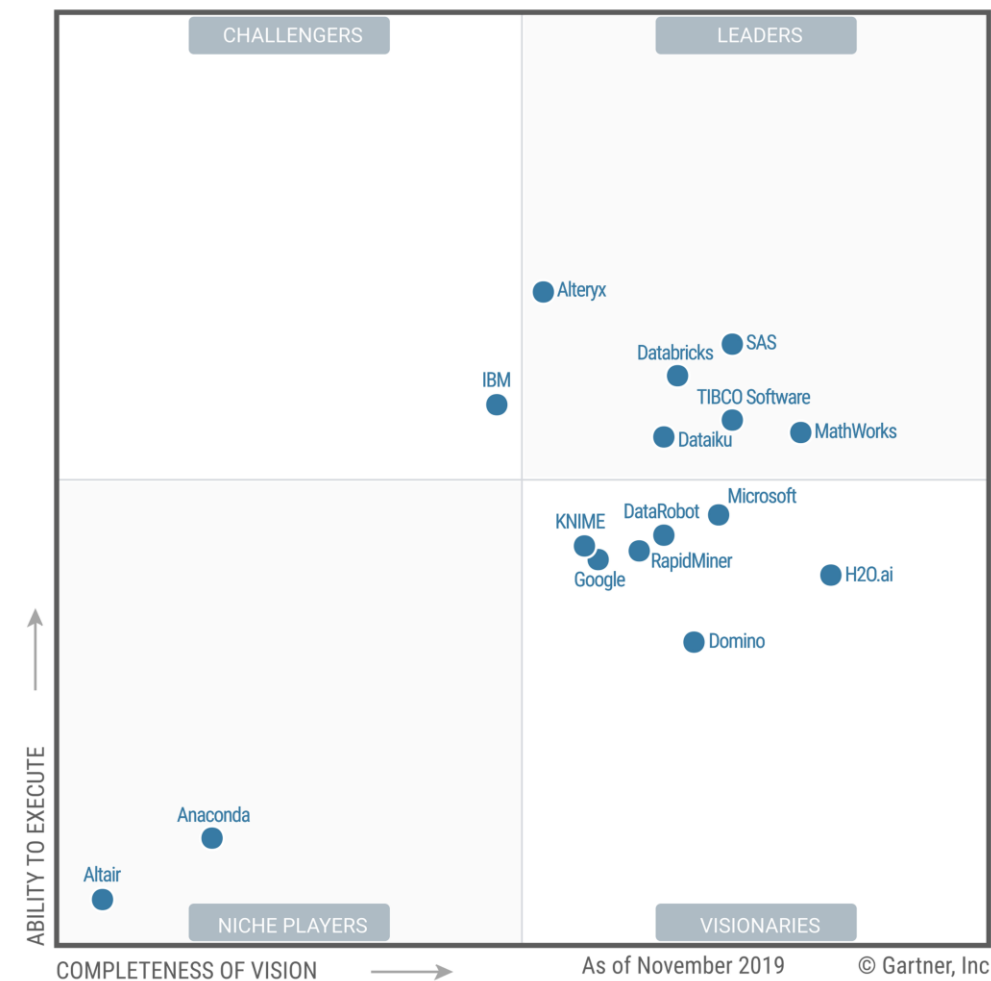
**Interoperability** with  
TensorFlow and PyTorch





is a **Leader** in the Gartner Magic Quadrant for 2020 Data Science and Machine Learning Platforms

Figure 1. Magic Quadrant for Data Science and Machine Learning Platforms



Source: Gartner (February 2020)

\*Gartner Magic Quadrant for Data Science and Machine Learning Platforms, Peter Krensky, Erick Brethenoux, Jim Hare, Carlie Idoine, Alexander Linden, Svetlana Sicular, 11 February 2020 .

This graphic was published by Gartner, Inc. as part of a larger research document and should be evaluated in the context of the entire document. The Gartner document is available upon request from MathWorks. Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, express or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.





# AI-driven system design

## Data Preparation



Data cleansing and preparation



Human insight



Simulation-generated data

## AI Modeling



Model design and tuning



Hardware accelerated training



Interoperability

## Simulation & Test



Integration with complex systems



System simulation



System verification and validation

## Deployment



Embedded devices



Enterprise systems



Edge, cloud, desktop



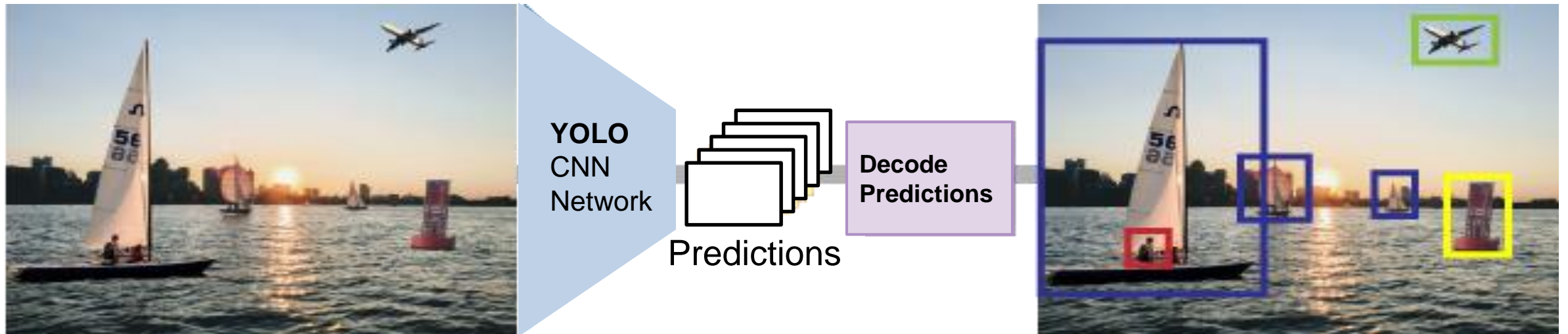
# Agenda

- I. Deep learning in engineering and science
- II. Developing a deep learning solution in MATLAB
- III. MathWorks deep learning support

# Featured Example: Detecting Objects with YOLO v2

Build, test, and deploy a deep learning solution that can detect objects in images and video.

- [You Only Look Once](#)
- Real-time object detector
- Autonomous driving, traffic monitoring
- 1000x faster than R-CNN







# Data preparation represents most of your AI effort...

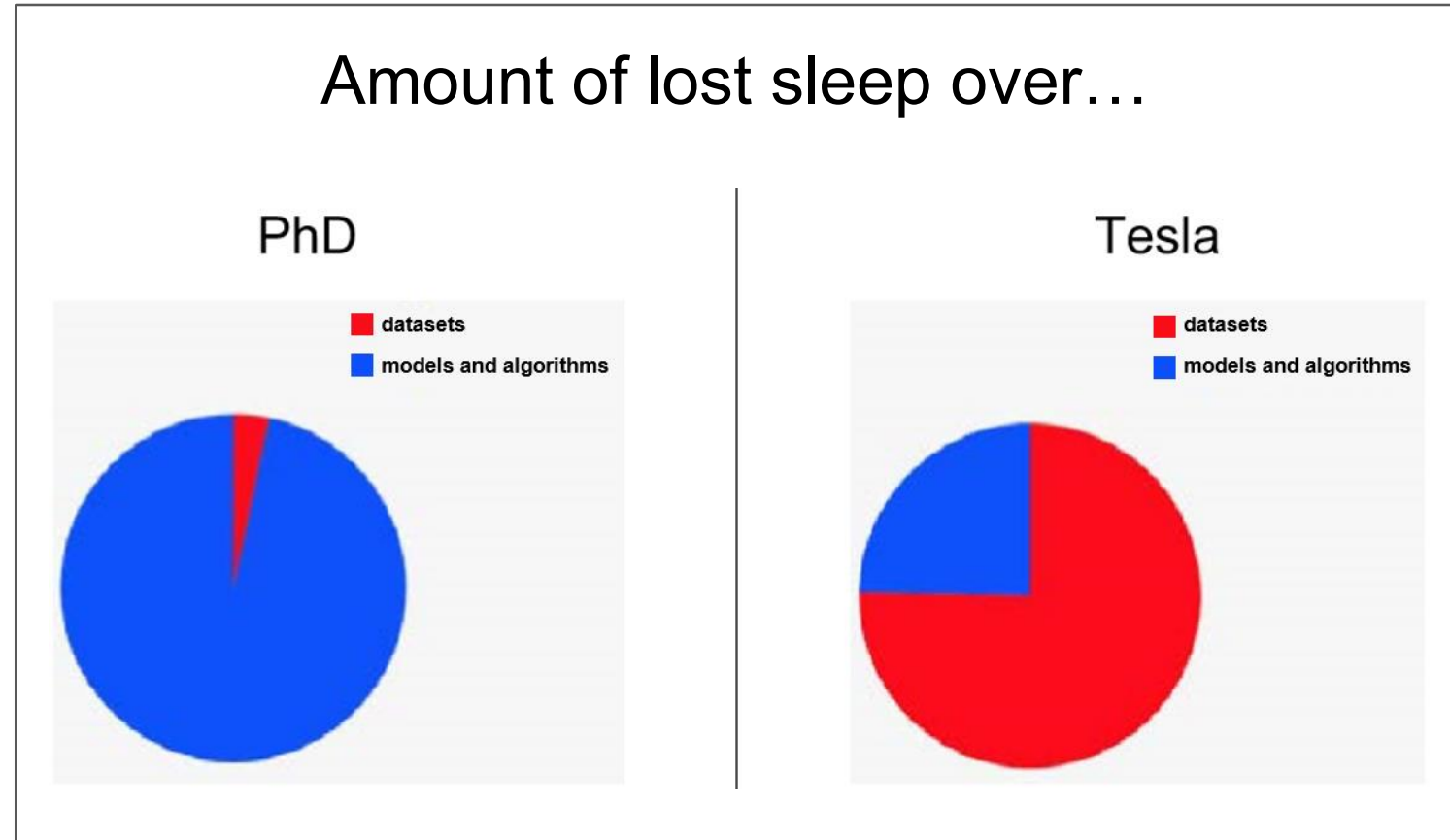
Transforming raw data for useful modeling and analysis is a critical step.

**Data Preparation**

 Data cleansing and preparation

 Human insight

 Simulation-generated data



Source: Andrej Karpathy slide from TrainAI 2018

# Spend less time preprocessing and labeling data

Synchronize disparate time series, filter noisy signals, automate labeling of video, and more.

## Data Preparation



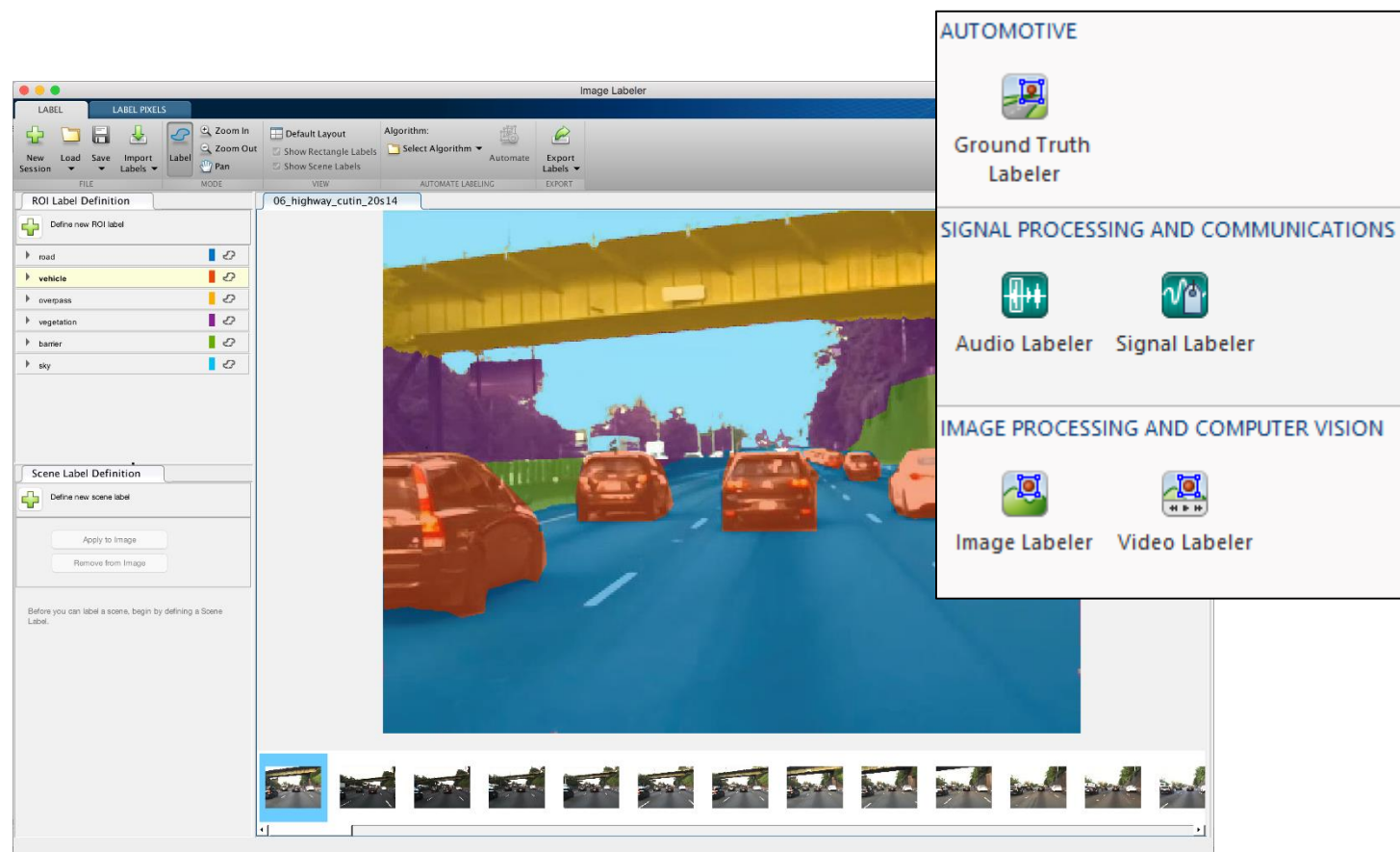
Data cleansing and preparation



Human insight

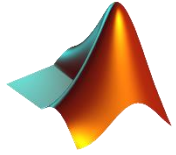


Simulation-generated data



Use labeling apps for deep learning workflows like semantic segmentation

# Data Preparation Demo



## Open Script Part 1

### Data Preparation



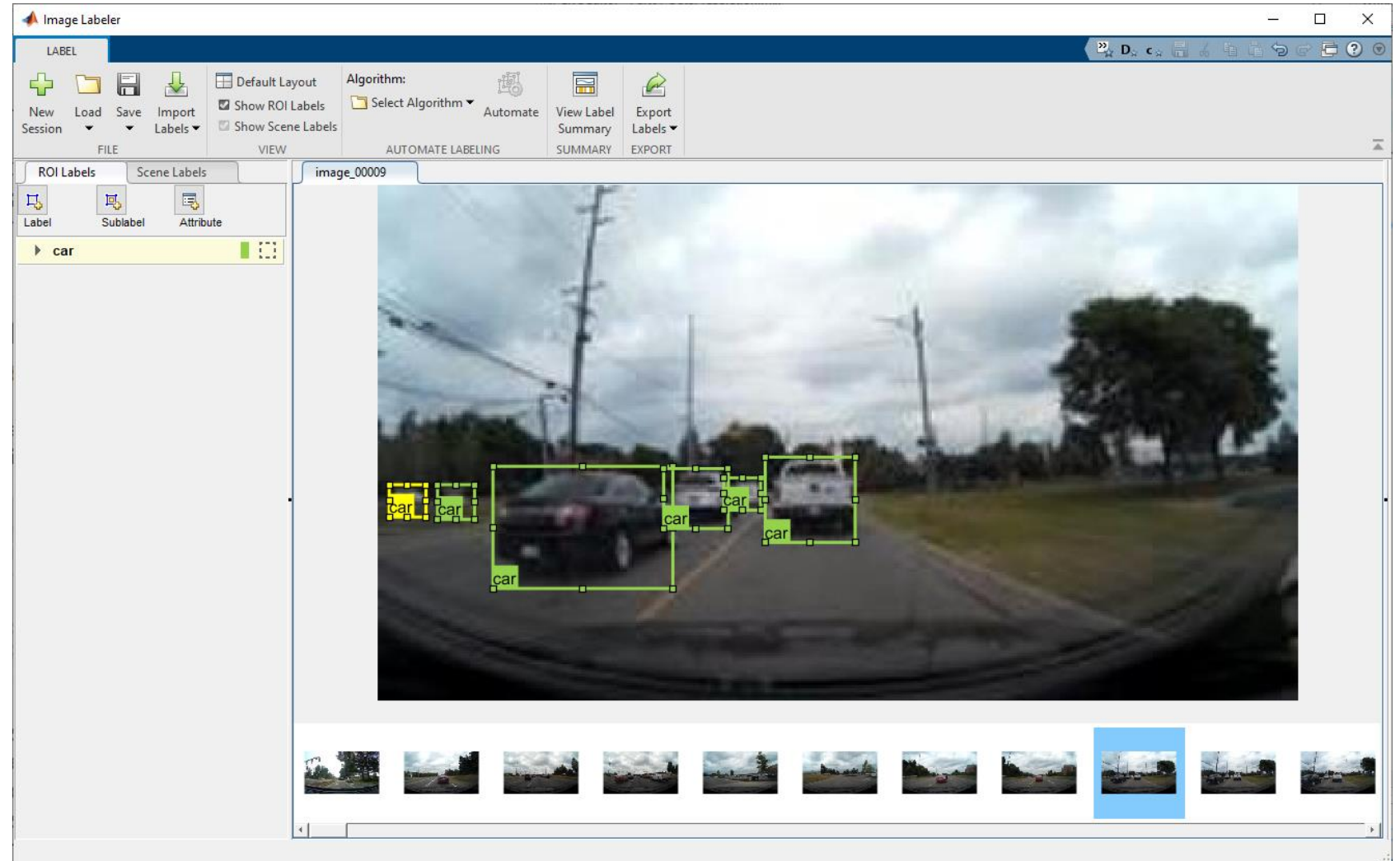
Data cleansing and  
preparation



Human insight




Simulation-  
generated data







# Start with a complete set of algorithms and pre-built models

**AI Modeling**

 Model design and tuning

 Hardware accelerated training

 Interoperability

## Algorithms

### Machine learning

Trees, Naïve Bayes, SVM...

### Deep learning

CNNs, GANs, LSTM, MIMO...

### Reinforcement learning

DQN, A2C, DDPG...

### Regression

Linear, nonlinear, trees...

### Unsupervised learning

K-means, PCA, GMM...

### Predictive maintenance

RUL models, condition indicators...

### Bayesian optimization

## Pre-built models

### Image classification models

AlexNet, GoogLeNet, VGG,  
SqueezeNet, ShuffleNet, ResNet,  
DenseNet, Inception...

## Reference examples

### Object detection

Vehicles, pedestrians, faces...

### Semantic segmentation

Roadway detection, land cover  
classification, tumor detection...

### Signal and speech processing

Denoising, music genre recognition,  
keyword spotting, radar waveform  
classification...

...and more...

# Increase productivity using Apps for design and analysis

Use MATLAB Apps to design deep learning networks, explore a wide range of classifiers, train regression models, train an optical character recognition model, and more.

## AI Modeling



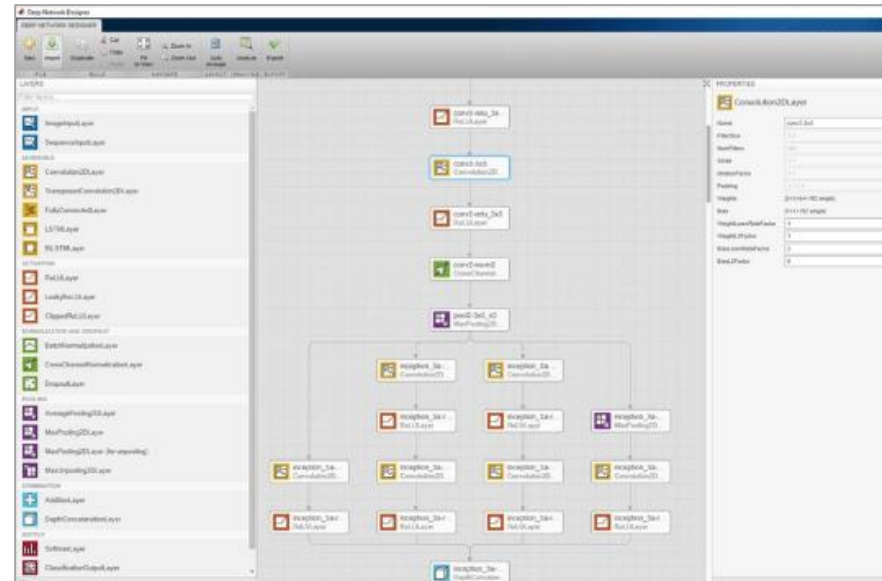
Model design and tuning



Hardware accelerated training



Interoperability



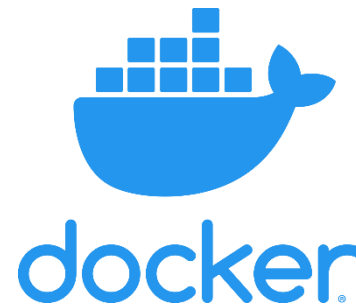
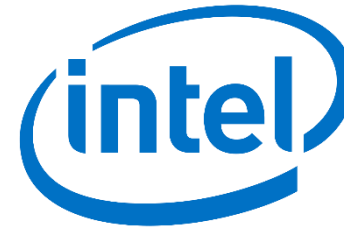
Deep Network Designer app to build, visualize, and edit deep learning networks

| Trial | Status   | Progress | Elapsed Time      | myInitialLearn... | convFilterSize | Training Accu... | Training Loss | Validation Ac... |
|-------|----------|----------|-------------------|-------------------|----------------|------------------|---------------|------------------|
| 1     | Complete | 100.0%   | 0 hr 0 min 16 sec | 1.0000e-6         | 3.0000         | 12.5000          | 2.4441        | 10.              |
| 2     | Complete | 100.0%   | 0 hr 0 min 15 sec | 1.0000e-5         | 3.0000         | 25.7813          | 2.1228        | 20.              |
| 3     | Complete | 100.0%   | 0 hr 0 min 14 sec | 0.0001            | 3.0000         | 64.8438          | 1.0878        | 42.              |
| 4     | Complete | 100.0%   | 0 hr 0 min 16 sec | 0.0005            | 3.0000         | 90.6250          | 0.4648        | 49.              |
| 5     | Complete | 100.0%   | 0 hr 0 min 15 sec | 1.0000e-6         | 4.0000         | 11.7188          | 2.4967        | 6.               |
| 6     | Complete | 100.0%   | 0 hr 0 min 15 sec | 1.0000e-5         | 4.0000         | 23.4375          | 2.1233        | 14.              |
| 7     | Complete | 100.0%   | 0 hr 0 min 17 sec | 0.0001            | 4.0000         | 72.6563          | 1.0283        | 39.              |
| 8     | Running  | 30.7%    | 0 hr 0 min 4 sec  | 0.0005            | 4.0000         |                  |               |                  |
| 9     | Queued   | 0.0%     |                   | 1.0000e-6         | 5.0000         |                  |               |                  |
| 10    | Queued   | 0.0%     |                   | 1.0000e-5         | 5.0000         |                  |               |                  |
| 11    | Queued   | 0.0%     |                   | 0.0001            | 5.0000         |                  |               |                  |
| 12    | Queued   | 0.0%     |                   | 0.0005            | 5.0000         |                  |               |                  |
| 13    | Queued   | 0.0%     |                   | 1.0000e-6         | 6.0000         |                  |               |                  |
| 14    | Queued   | 0.0%     |                   | 1.0000e-5         | 6.0000         |                  |               |                  |
| 15    | Queued   | 0.0%     |                   | 0.0001            | 6.0000         |                  |               |                  |
| 16    | Queued   | 0.0%     |                   | 0.0005            | 6.0000         |                  |               |                  |


Experiment Manager app to manage multiple deep learning experiments, analyze and compare results and code


# Hardware acceleration and scaling are critical for training


MATLAB accelerates AI training on GPUs, cloud, and datacenter resources without specialized programming.



**AI Modeling**

 Model design and tuning

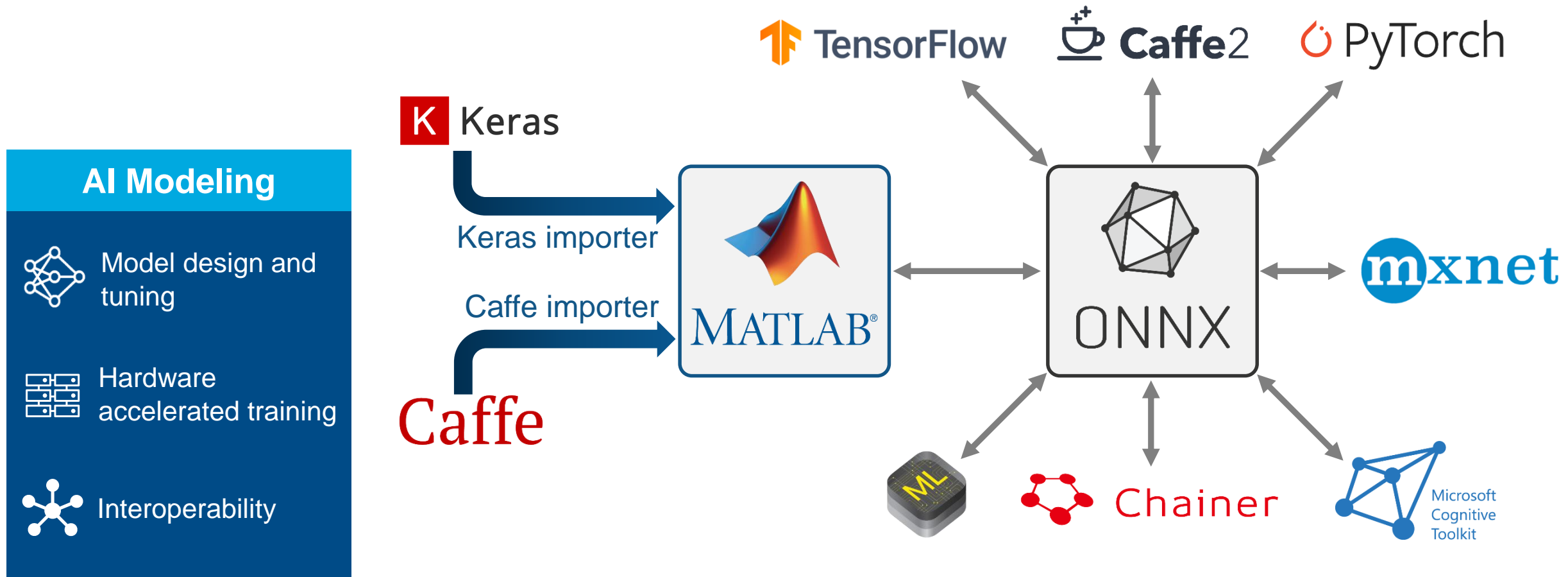
 Hardware accelerated training

 Interoperability

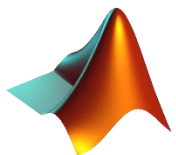


# MATLAB interoperates with other frameworks

Supports ONNX and can exchange models with PyTorch, TensorFlow, and other frameworks.



# Modeling Demo



**Open Script  
Part 2**

## AI Modeling



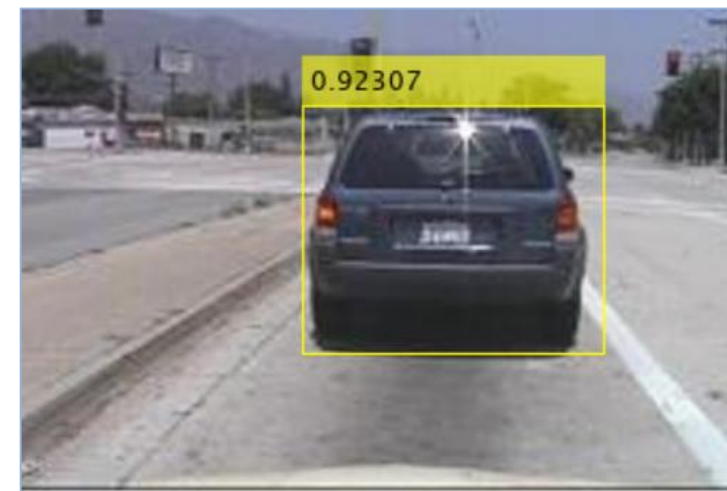
Model design and  
tuning



Hardware  
accelerated training



Interoperability



# Models need to exist within a complete system

In automated driving systems, AI for perception must integrate with algorithms for path planning, braking, acceleration, and other controls.

## Simulation & Test

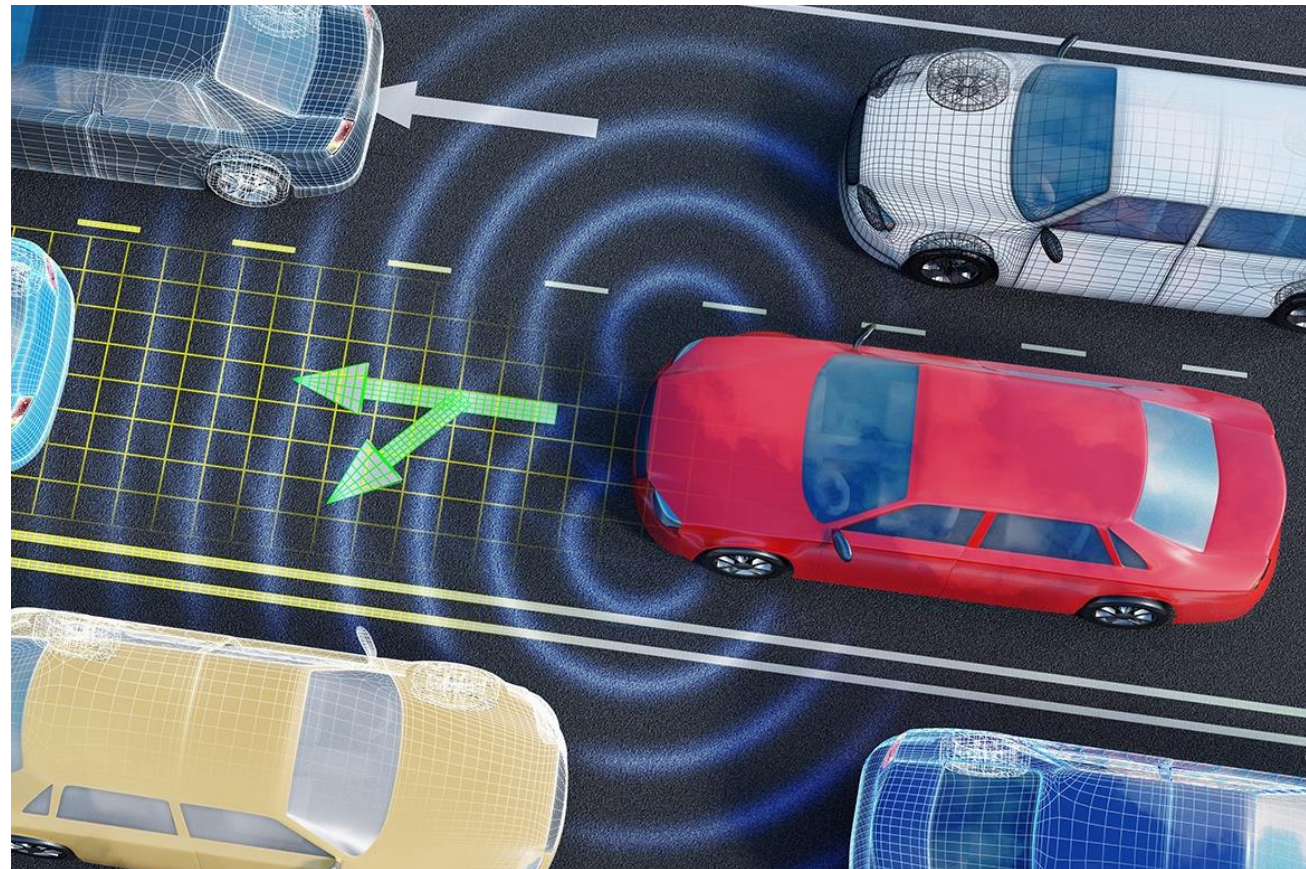


Integration with complex systems



System simulation

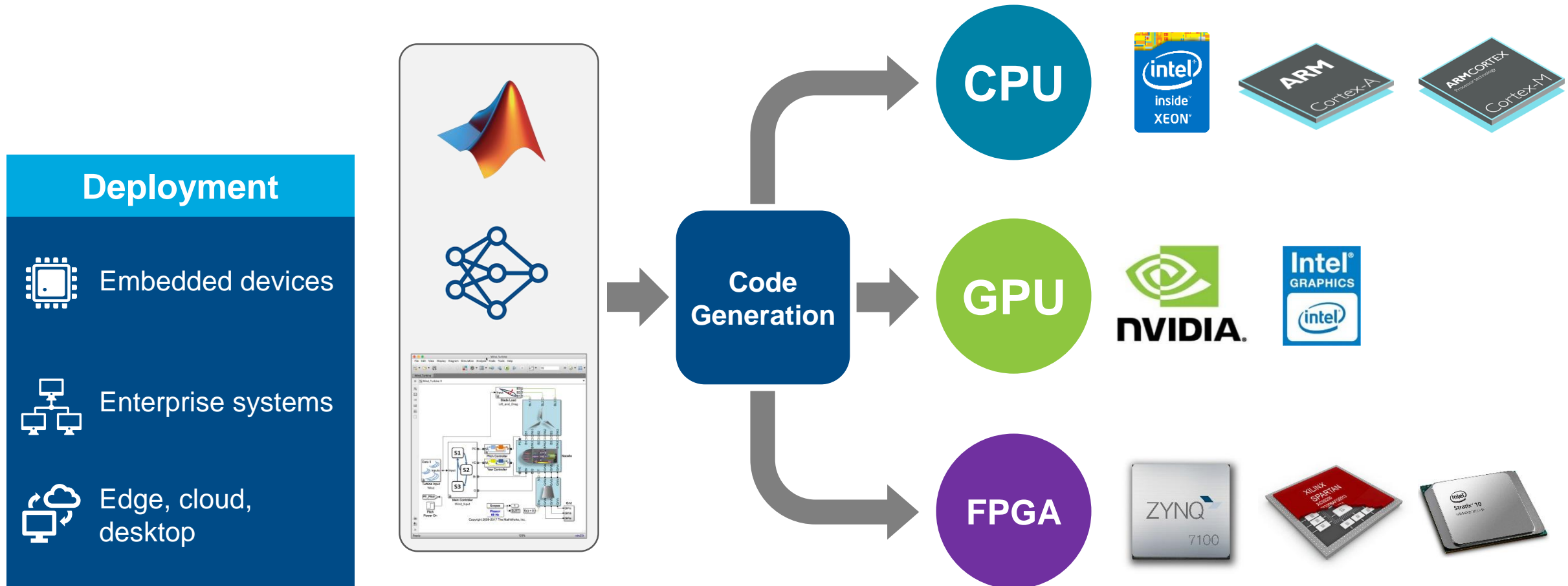
— x System verification  
— ✓ and validation





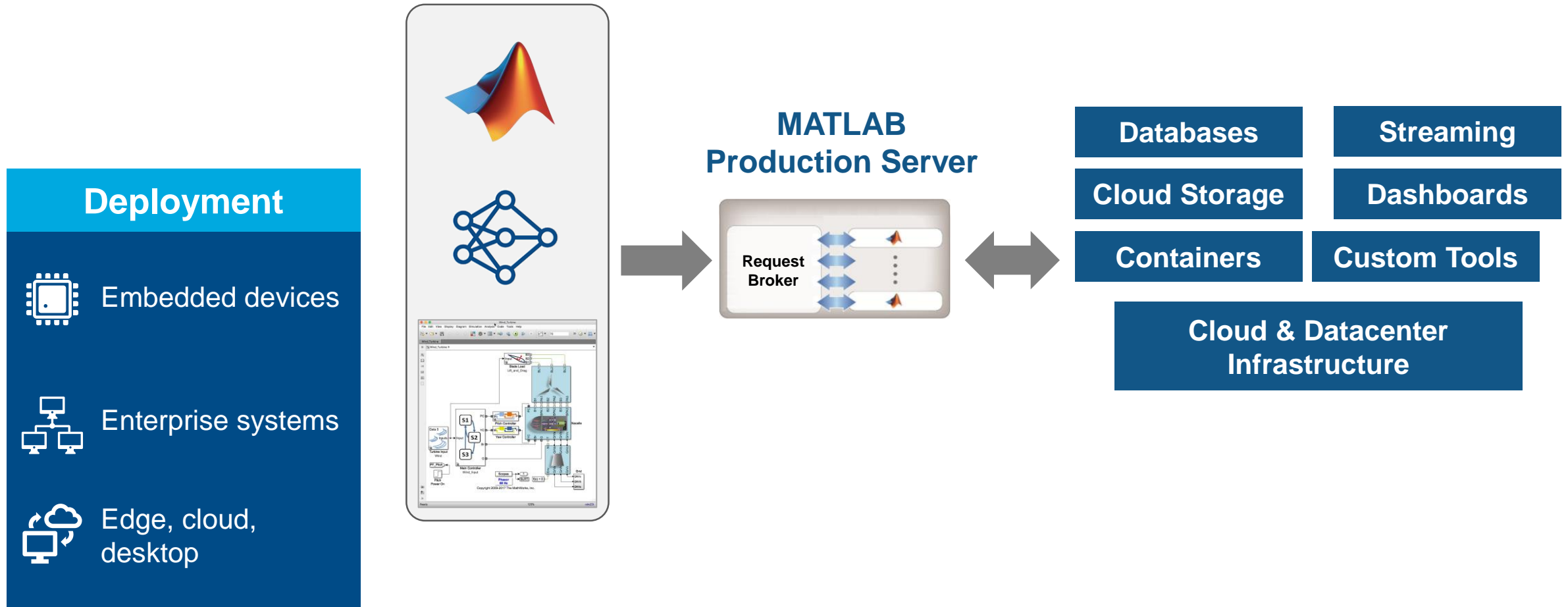
# Deploy to any processor with best-in-class performance

AI models in MATLAB and Simulink can be deployed on embedded devices, edge devices, enterprise systems, the cloud, or the desktop.

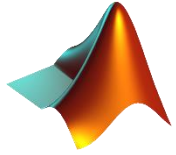




# Deploy to enterprise IT infrastructure



# Deployment Demo



Open Script  
Part 3

## Deployment



Embedded devices



Enterprise systems



Edge, cloud,  
desktop

The screenshot displays the MATLAB Coder interface. The top toolbar includes navigation (Back, Forward, Find), trace (Trace Code), edit (Edit In MATLAB), package (Package Code), and share (Export Report Information) options. The left sidebar shows the 'MATLAB SOURCE' pane with a 'Function List' and 'Call Tree' for 'yolov2\_detect.m', which includes sub-functions 'fx yolov2\_detect' (marked GPU) and 'fx cudnnApi\_register'. Below this is the 'GENERATED CODE' pane, showing 'Files' and 'GPU Kernels' for 'Source Files'. The main editor displays the MATLAB function 'yolov2\_detect' with the following code:

```
1 function outImg = yolov2_detect(in)
2
3 % Copyright 2018-2019 The MathWorks, Inc.
4
5 persistent yolov2obj;
6
7 if isempty(yolov2obj)
8     yolov2obj = coder.loadDeepLearningNetwork('Yolov2UsingResNet50_ONNX.mat');
9 end
10
11 % pass in input
12 [bboxes,~,labels] = yolov2obj.detect(in,'Threshold',0.5);
13
14 % convert categorical labels to cell array of character vectors for MATLAB
15 % execution
16 if coder.target('MATLAB')
17     labels = cellstr(labels);
18 end
19
20 % Annotate detections in the image.
21 outImg = insertObjectAnnotation(in,'rectangle',bboxes,labels);
22
```

The bottom status bar shows a 'SUMMARY' tab with a green checkmark and the message 'Code generation successful'. It also provides details: 'Generated on: 17-Sep-2019 14:21:46', 'Build type: MEX Function', 'Output file: C:\Users\shmitra\Work\Deep\_Learning\Seminar\19b\ResNetImportYolov2\HelperFilesAndFunctions\yolov2\_detect\_mex.mexw64', and 'Processor: Generic->MATLAB Host Computer'.



# Agenda

- I. Deep learning in engineering and science
- II. Developing a deep learning solution in MATLAB
- III. MathWorks deep learning support



# MathWorks is your Deep Learning partner



## The Platform

MATLAB, Simulink, and over 100 add-on products for specialized applications



## Your People

Helping you build an agile workforce today and preparing tomorrow's engineers



## Our Expertise

From onboarding and implementation to solving advanced engineering challenges



# MathWorks Engineering Support



**Training**



**Guided Evaluations**



**Onsite Workshops**



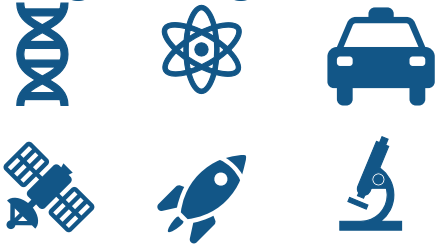
**Consulting**



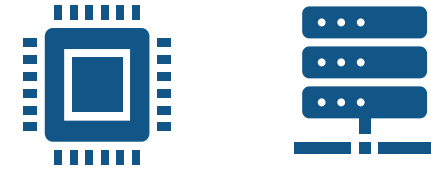
**Technical Support**

# Why MATLAB & MathWorks for Deep Learning?

Domain-specialized workflows  
for **engineering and science**



Multi-platform **deployment** of  
**full applications and systems**



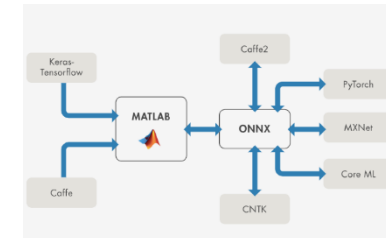
People



Platform productivity

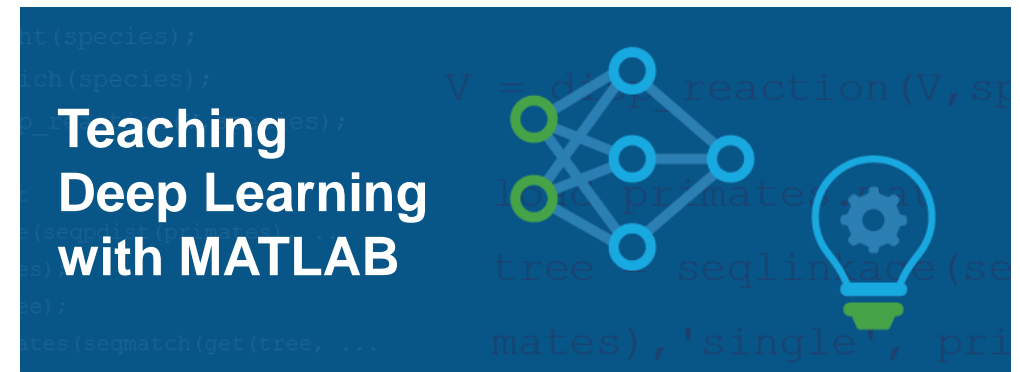


**Interoperability** with  
TensorFlow and PyTorch



# Further Learning & Teaching

- [Deep Learning Onramp](#)
  - 2 hr online tutorial
- Deep Learning Workshop
  - 3 hr hands on session
  - Contact us to schedule
- [Deep Learning Training](#)
  - 16 hr in depth course
  - Online or Instructor Lead
- [Teaching Deep Learning with MATLAB](#)
  - Curriculum support



# Where to find this content on GitHub

- <https://tinyurl.com/deeplearningmatlabimage>



Learn Git and GitHub without any code!  
Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

[Read the guide](#)

matlab-deep-learning / Object-Detection-Using-YOLO-v2-Deep-Learning

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

MATLAB example of deep learning based object detection using Yolo v2.

matlab deep-learning example yolo-v2 object-detection Manage topics

9 commits 1 branch 0 packages 0 releases 1 contributor View license

Branch: master New pull request Create new file Upload files Find file Clone or download

|                                       |                        | Latest commit 147b223 2 hours ago |
|---------------------------------------|------------------------|-----------------------------------|
| CodeGeneration                        | Initial push to GitHub | 19 hours ago                      |
| HelperFiles                           | Initial push to GitHub | 19 hours ago                      |
| Images                                | Updated README.md      | 18 hours ago                      |
| SavedModels                           | Initial push to GitHub | 19 hours ago                      |
| resources/project                     | Initial push to GitHub | 19 hours ago                      |
| Deep Learning Overview For Images.pdf | Initial push to GitHub | 19 hours ago                      |
| LICENSE                               | Initial push to GitHub | 19 hours ago                      |
| Part01_DataPreparation.mlx            | Initial push to GitHub | 19 hours ago                      |
| Part02_Modeling.mlx                   | Updated README.md      | 18 hours ago                      |
| Part01_Deployment.mlx                 | Initial push to GitHub | 19 hours ago                      |
| README.md                             | updated README.md      | 2 hours ago                       |
| YOLOv2ObjectDetection.prj             | Initial push to GitHub | 19 hours ago                      |

README.md

## Object Detection Example using Yolo v2 Deep Learning

This demo shows the full deep learning workflow for an example using image data in MATLAB. In it we use deep learning based object detection using Yolo v2 to identify vehicles of interest in a scene.

A photograph of a dark-colored car parked on a street. A yellow bounding box is drawn around the car, and the text "0.8185" is displayed below it, indicating the confidence score of the object detection model.