Are you at the right place ?

20 July, 1pm - 4pm AEST: Deep Learning with Images and MATLAB

Overview

Please join MathWorks and learn how to get started with MATLAB for Deep Learning with Images. In this hands-on workshop, we will introduce you to fundamentals of Deep Learning with Images. You'll have the opportunity to try out specific examples using MATLAB tools. The hands-on component of the workshop will be run via MATLAB Online – so attendees do NOT need to have MATLAB locally installed on their computers.

Highlights

- Learn the Deep Learning image classification workflow in MATLAB
- Image Data management
- Network assembly, training
- Experiment management
- Create a Convolution Neural Network (CNN) from scratch
- Programmatically and using APPs
- Explore how to access and adjust pretrained models (transfer learning)
- Explore how to evaluate the network and improve its accuracy

Prework:

- Attendees will need to bring their own laptops **(MATLAB does NOT need to be installed)**
- This workshop is for NCI users who have a MATLAB license. For more information about MATLAB license, please check our webpage about MATLAB license supported groups. The workshop will be held at ANU campus. **See the following FAQ for instructions on how to create a MathWorks account: https://www.mathworks.com/videos/create-a-mathworks-account-using-a-matlab-portal-1600159919958.html**
- Attendees should have a basic level of understanding of MATLAB syntax. The FREE online course MATLAB OnRamp would be a recommended prerequisite for any new users of MATLAB.

Any issues doing this ?

https://www.eventbrite.com.au/e/nci-presents-mathworks-workshop-series-2021-tickets-156992022365

Workshop Setup

Should take 5 minutes !

MathWorks®

Setup overview

- Background:
  - Today we'll be using MATLAB Online
    - A version of MATLAB that runs in your web Browser
  - But please use the special MATLAB Online link that I will share with you shortly.
    - This gives access to Cloud GPUs

- STEPS
  1. Confirm you are using a supported Browser
  2. Make sure you have a MathWorks Account
  3. Use the special MATLAB Online link
  4. Copy the Workshop files
  5. Confirm you have access to a cloud GPU

Should take 5-10 minutes

A Word on Browser support

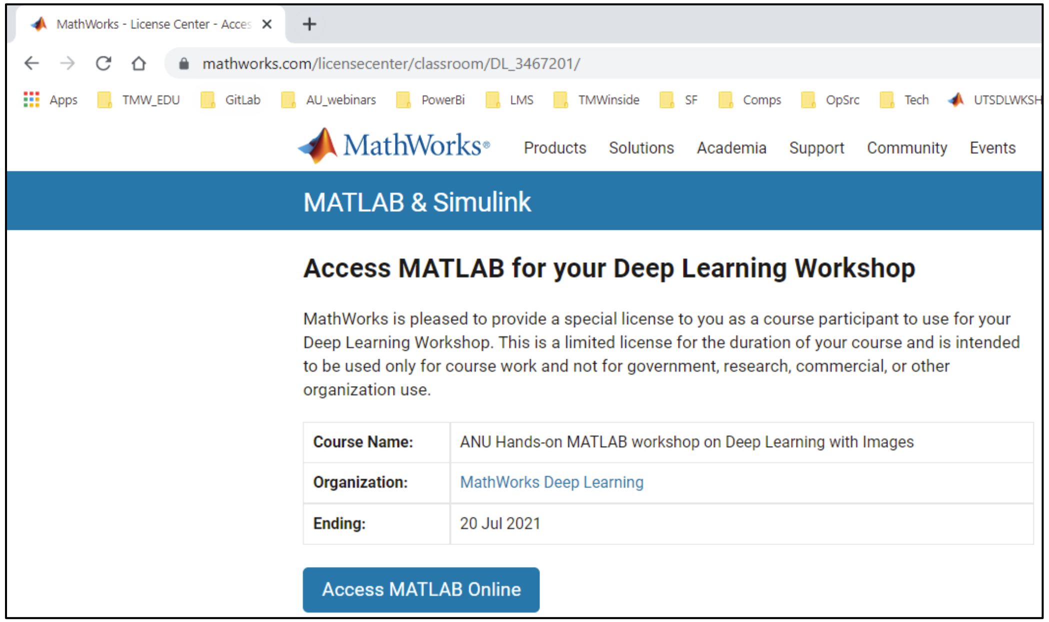
Highly recommended to use Google chrome

Google Chrome

Deep Network Designer

Eg: our `deepNetworkDesigner` APP is only supported in Chrome

Stop …. Stare …. bookmark



https://www.mathworks.com/licensecenter/classroom/DL_3467201/

Download the Setup PDF

https://drive.matlab.com/sharing/56c9e7cb-400c-4fb0-93c8-72f438d7fe46

Set-Up Instructions – part 1 of 3

Step 1: MATLAB Drive - Login with your MathWorks Account used to register for the event

Login to your **MATLAB Drive** at https://drive.matlab.com/login

- **Use the email address that you submitted to register for the event.**

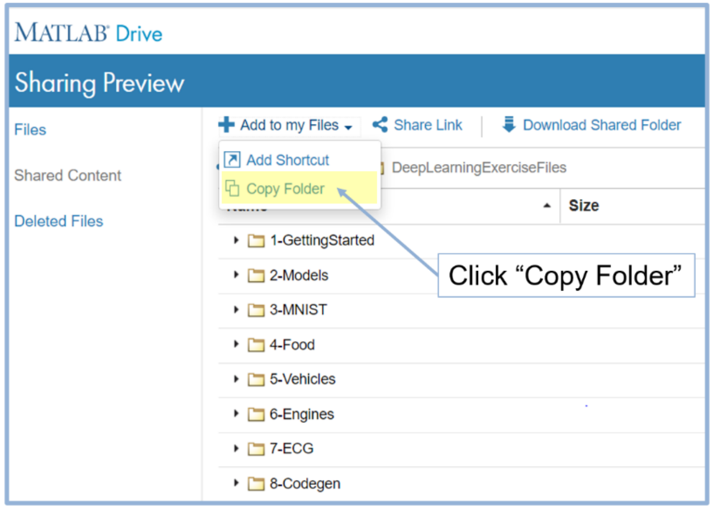
If creating a new account, visit https://www.mathworks.com/mwaccount/register

Step 2: Copy Workshop Files

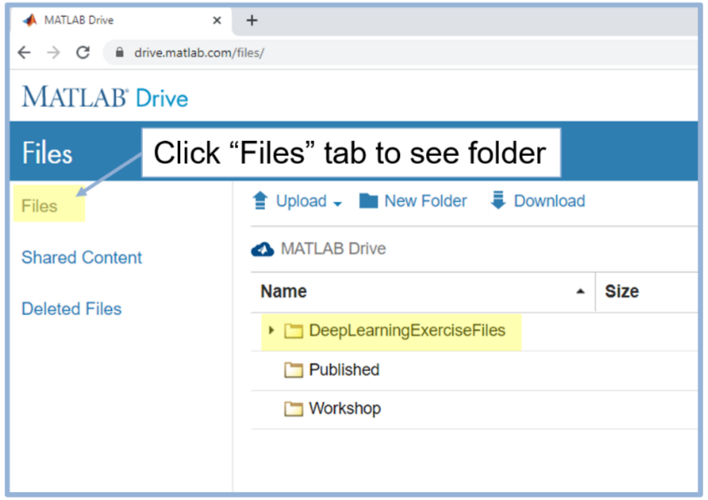
- Click **HERE** and accept the shared files from Pitambar Dayal.
  - FYI: the full address is:
    - https://drive.matlab.com/sharing/e1d60207-94f1-4af3-aee4-8174370eb421

Note: If you are unable to access the above link, wait 30 minutes and try again.

Step 2a

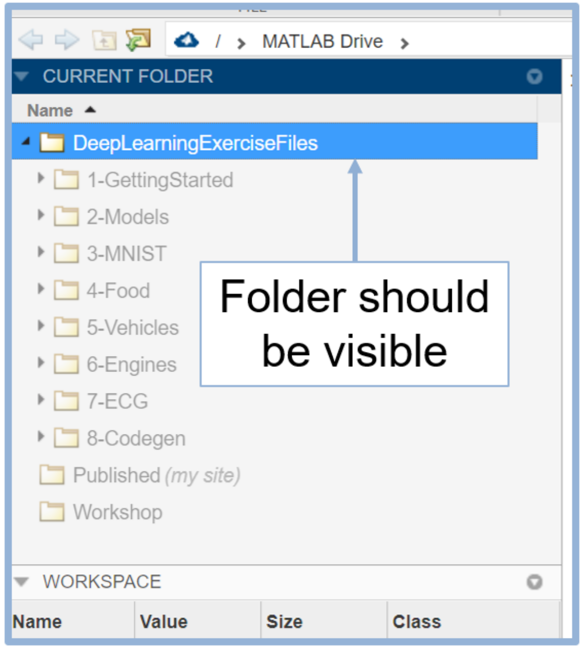


Step 2b



Step 3: Log into the Workshop MATLAB Online and Confirm Web Browser

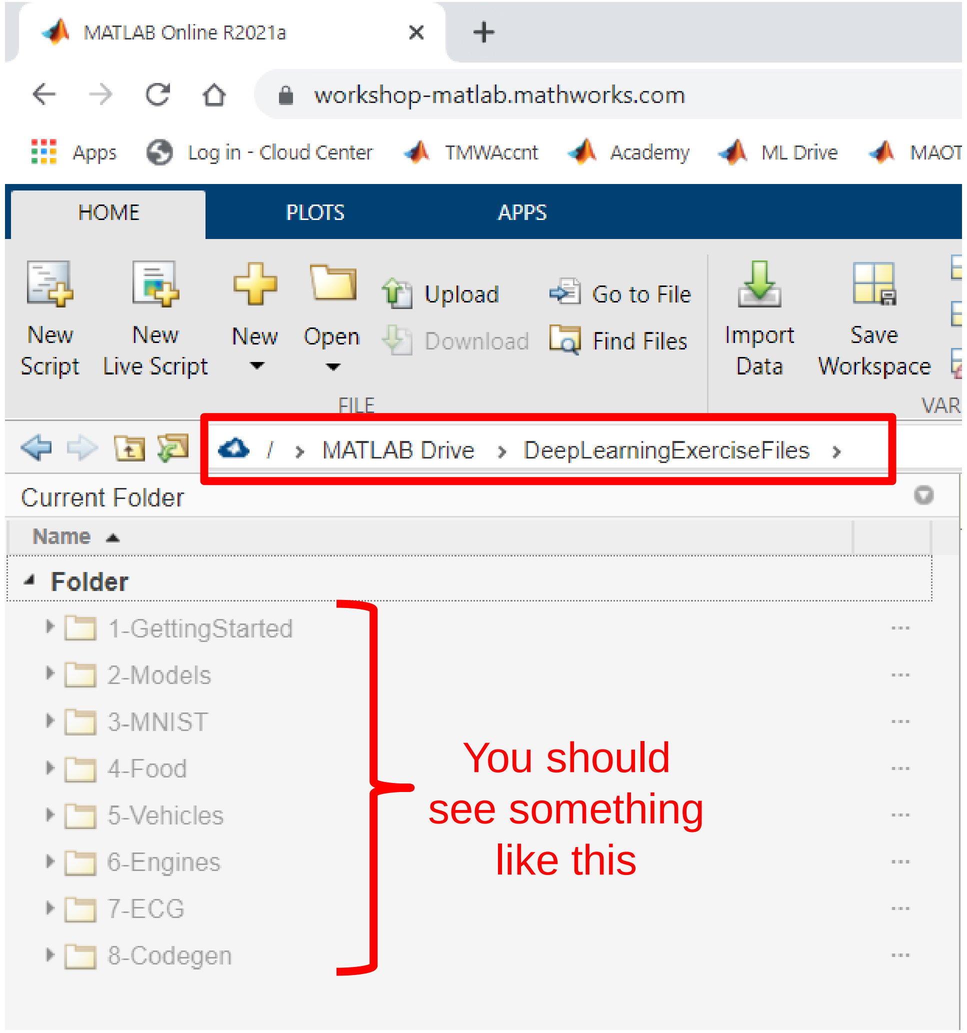
- Visit the following URL and login to access MATLAB Online
  - https://www.mathworks.com/licensecenter/classroom/DL_3467201/
- ***Note:*** If you are unable to login or access the above link, wait a few minutes and try again. If having issues with your browser, Chrome has been tested and usually works well.



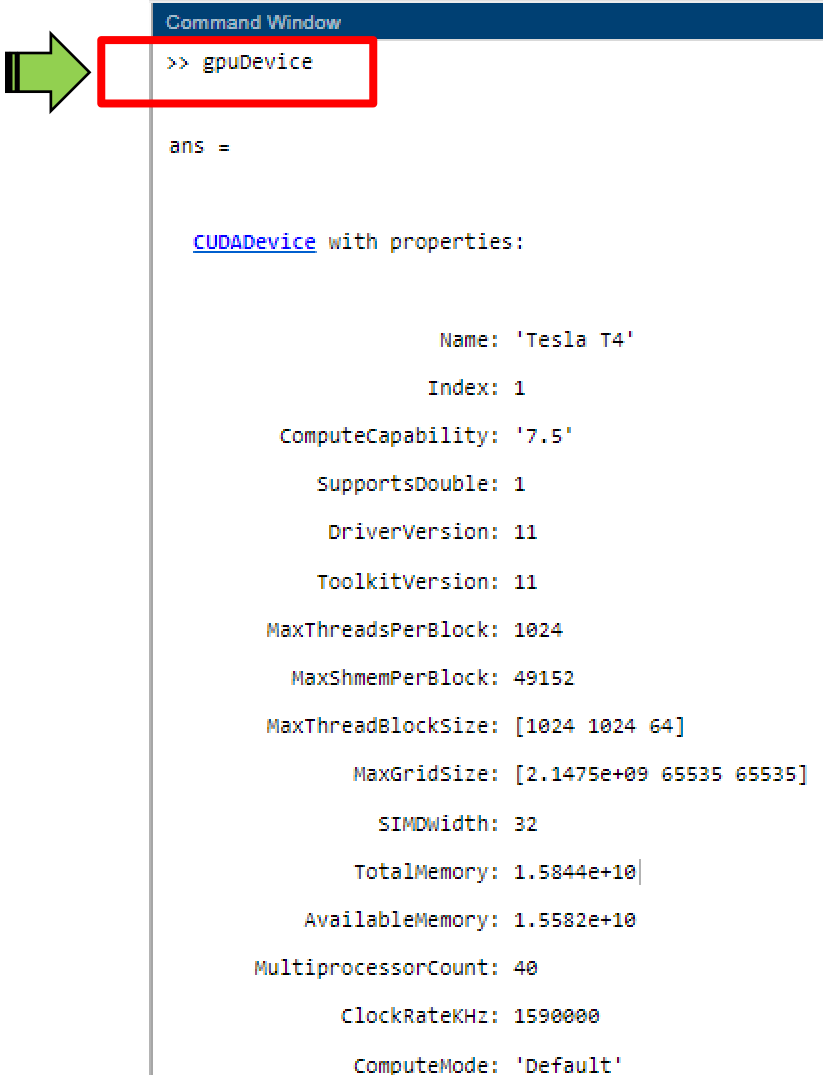
- Your current folder browser should have the folder you copied over.

Set-Up Instructions – part 2 of 3

3 – Navigate into the `DeepLearningExerciseFiles` folder

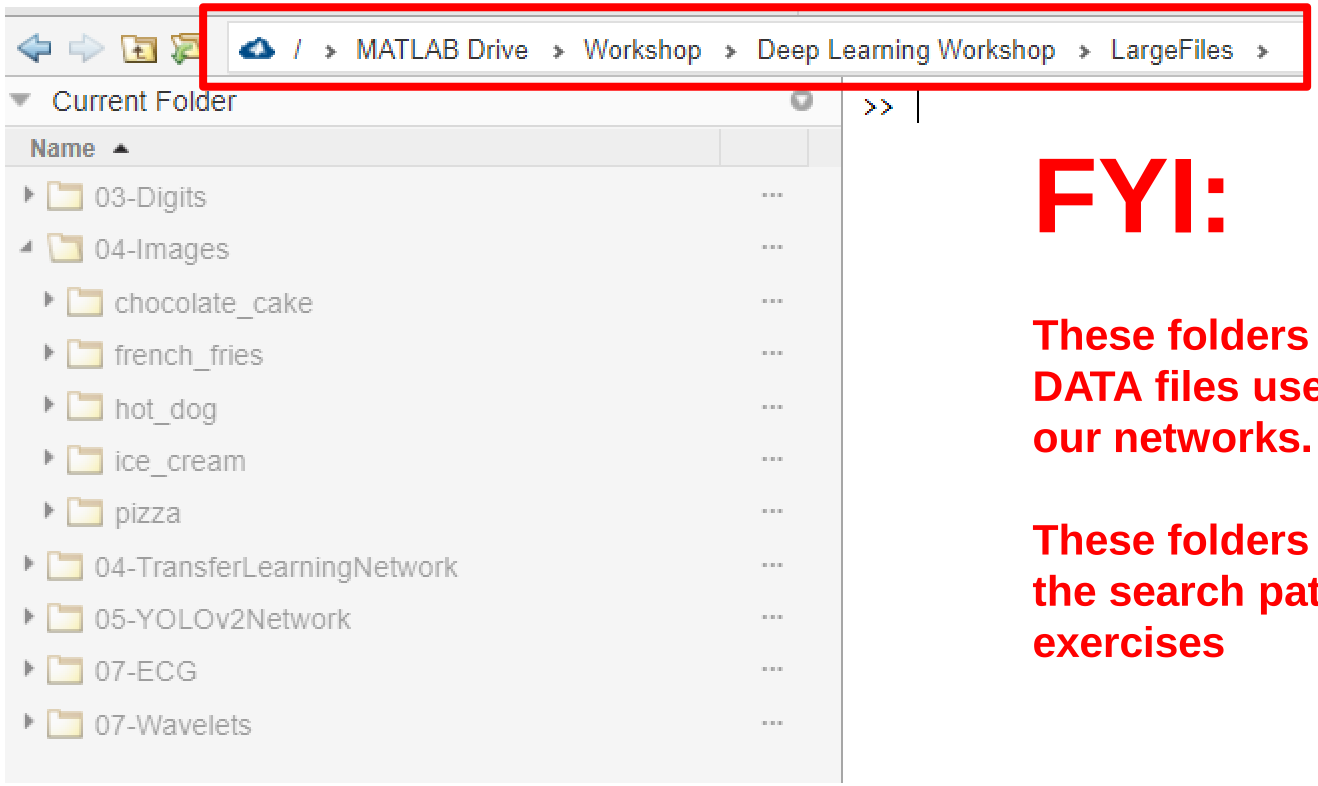


4 – Confirm you have a reserved GPU



You should see something like this

Set-Up Instructions – part 3 of 3



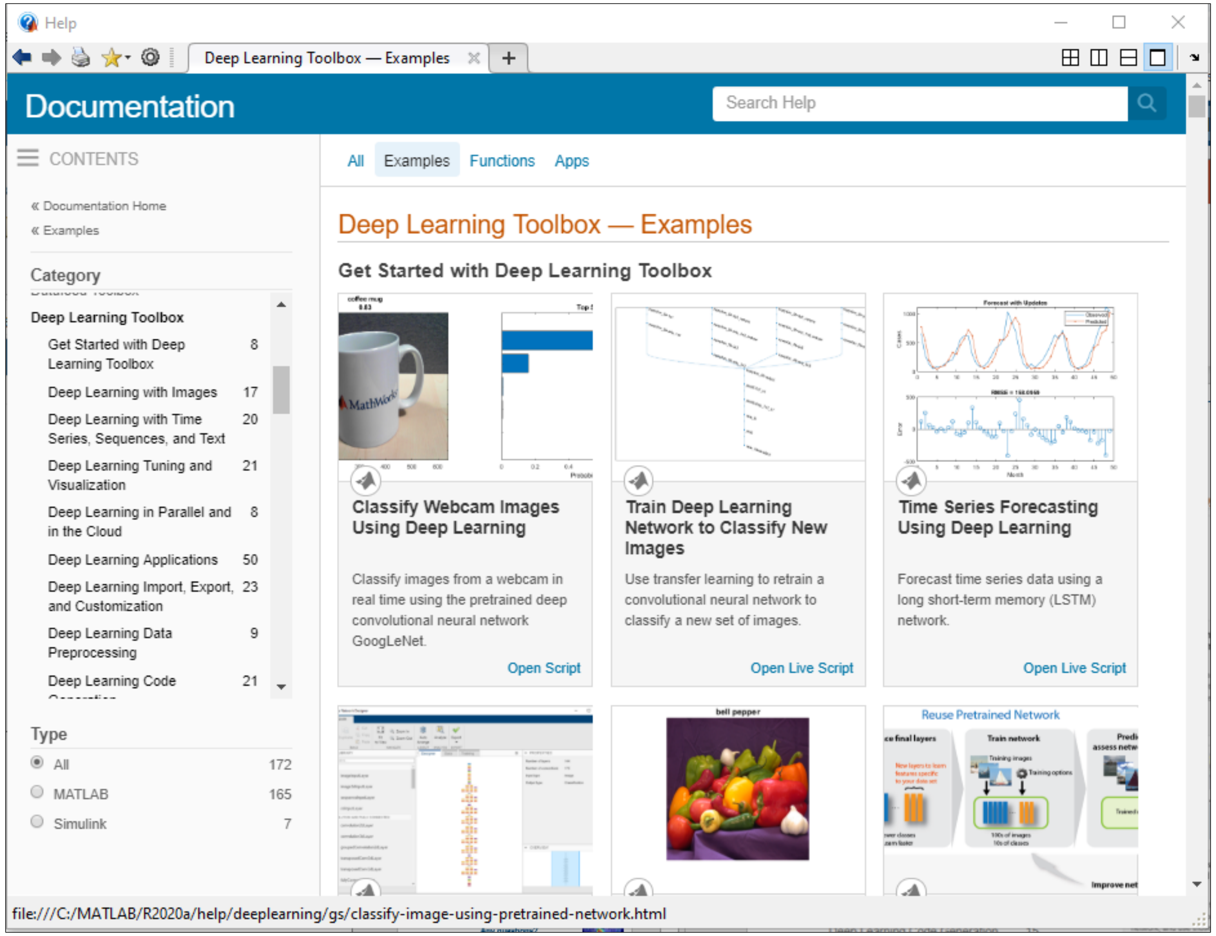
FYI:

These folders contain the DATA files used for training our networks.

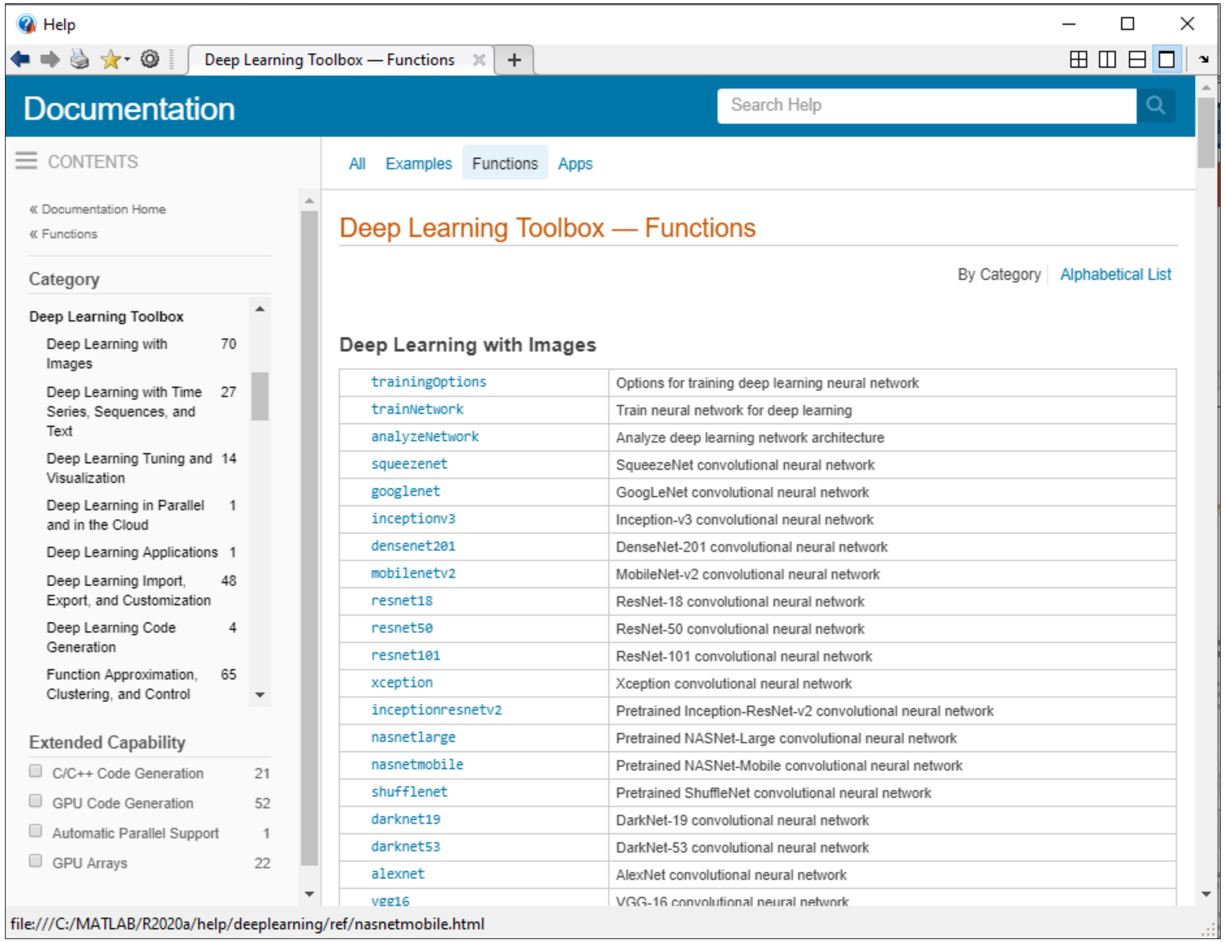
These folders will be added to the search path during the exercises

Deep Learning Toolbox

172 shipping examples to explore



230 functions/classes



Fun fact:

The **MATLAB Campus License** is almost at every university in AU/NZ !

AU: 33 out of 40
NZ: 6 out of 8

So ?

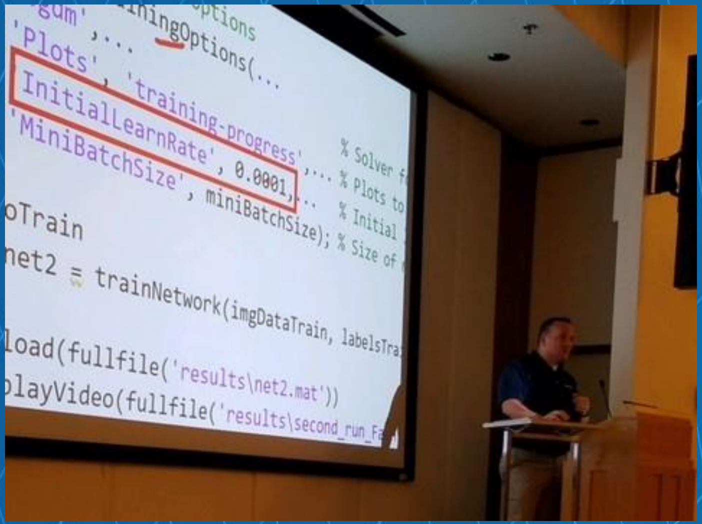
- Every student
- **Every Product**
- Every Computer(campus)
- Every Computer(personal)

MathWorks

MATLAB
Simulink
5G Toolbox
Aerospace Blockset
Aerospace Toolbox
Antenna Toolbox
Audio Toolbox
Automated Driving Toolbox
AUTOSAR Blockset
Bioinformatics Toolbox
Communications Toolbox
Computer Vision Toolbox
Control System Toolbox
Curve Fitting Toolbox
Data Acquisition Toolbox
Database Toolbox
Datafeed Toolbox
DDS Blockset
Deep Learning HDL Toolbox
Deep Learning Toolbox
DSP System Toolbox
Econometrics Toolbox
Embedded Coder
Filter Design HDL Coder
Financial Instruments Toolbox
Financial Toolbox
Fixed-Point Designer
Fuzzy Logic Toolbox
Global Optimization Toolbox
GPU Coder
HDL Coder
HDL Verifier
Image Acquisition Toolbox
Image Processing Toolbox
Instrument Control Toolbox
Lidar Toolbox
LTE Toolbox
Mapping Toolbox
MATLAB Coder
MATLAB Compiler
MATLAB Compiler SDK
MATLAB Parallel Server
MATLAB Production Server
MATLAB Report Generator
MATLAB Web App Server
Mixed-Signal Blockset
Model Predictive Control Toolbox
Model-Based Calibration Toolbox
Motor Control Blockset
Navigation Toolbox
OPC Toolbox
Optimization Toolbox
Parallel Computing Toolbox
Partial Differential Equation Toolbox
Phased Array System Toolbox
Polyspace Bug Finder
Polyspace Code Prover
Powertrain Blockset
Predictive Maintenance Toolbox
Radar Toolbox
Reinforcement Learning Toolbox
RF Blockset
RF Toolbox
Risk Management Toolbox
RoadRunner
RoadRunner Asset Library
Robotics System Toolbox
Robust Control Toolbox
ROS Toolbox
Satellite Communications Toolbox
Sensor Fusion and Tracking Toolbox
SerDes Toolbox
Signal Processing Toolbox
SimBiology
SimEvents
Simscape
Simscape Driveline
Simscape Electrical
Simscape Fluids
Simscape Multibody
Simulink 3D Animation
Simulink Check
Simulink Code Inspector
Simulink Coder
Simulink Compiler
Simulink Control Design
Simulink Coverage
Simulink Design Optimization
Simulink Design Verifier
Simulink Desktop Real-Time
Simulink PLC Coder
Simulink Real-Time
Simulink Report Generator
Simulink Requirements
Simulink Test
SoC Blockset
Spreadsheet Link
Stateflow
Statistics and Machine Learning Toolbox
Symbolic Math Toolbox
System Composer
System Identification Toolbox
Text Analytics Toolbox
UAV Toolbox
Vehicle Dynamics Blockset
Vehicle Network Toolbox
Vision HDL Toolbox
Wavelet Toolbox
Wireless HDL Toolbox
WLAN Toolbox

MathWorks®

Hands-on Deep Learning Workshop



Agenda

DEMOs

Gentle "Warm up" stretches

Exercise 01

Deep Learning in 6 lines

Driving a Live Script

Exercise 02

Managing Data files

What is a datastore ?

interesting bits

Exercise 03

Digit classification

Exercise 04

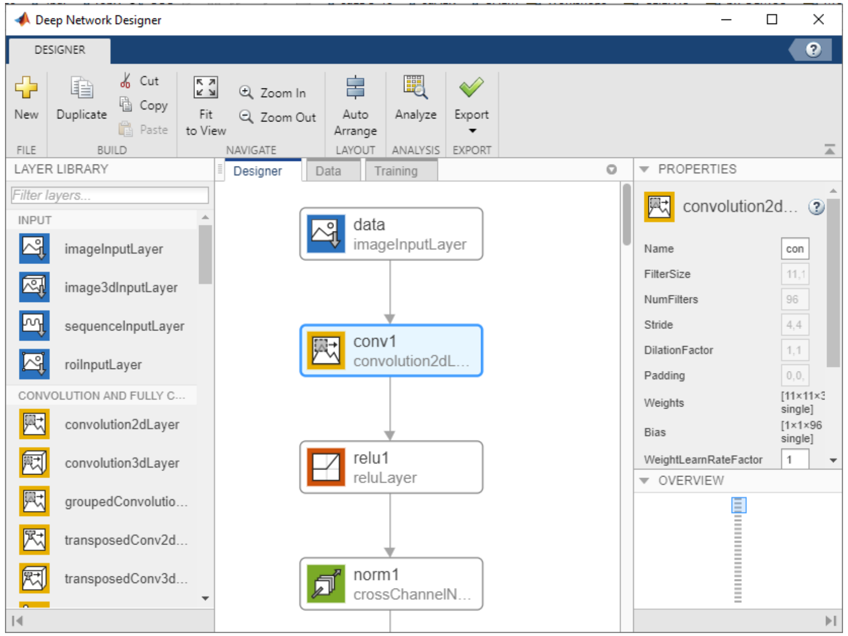
Transfer Learning

Exercise 05

YOLO Object detector

WORKFLOW#1: Deep Network Designer APP

WORKFLOW#2: Manual Coding



Exercise 07

ECG Heart Condition classifier

The Experiment Manager APP

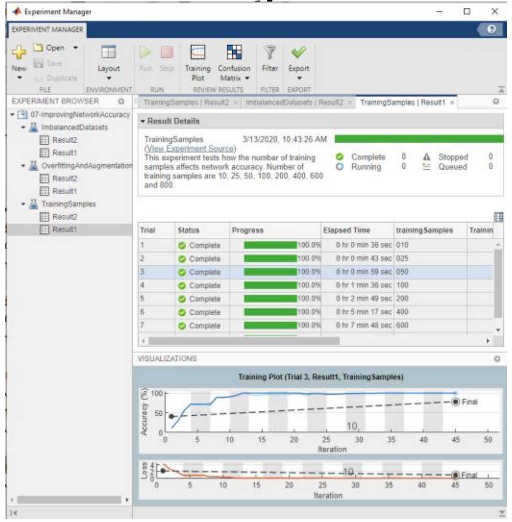
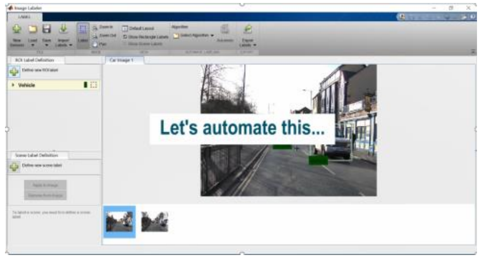
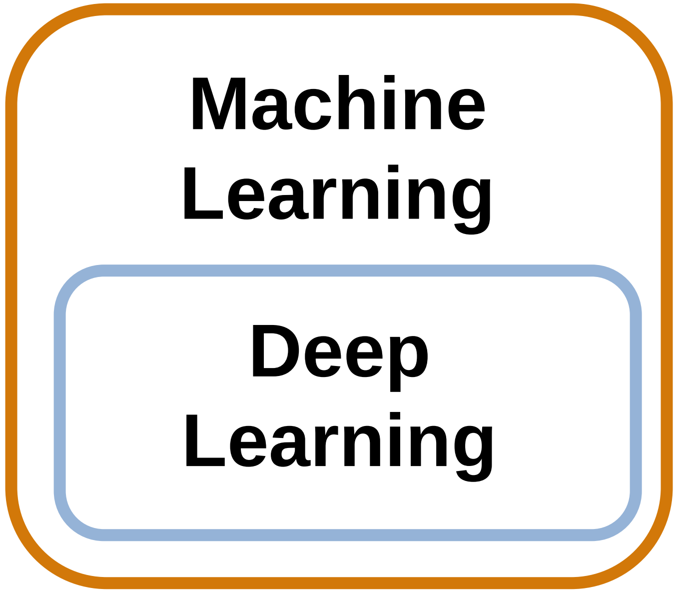


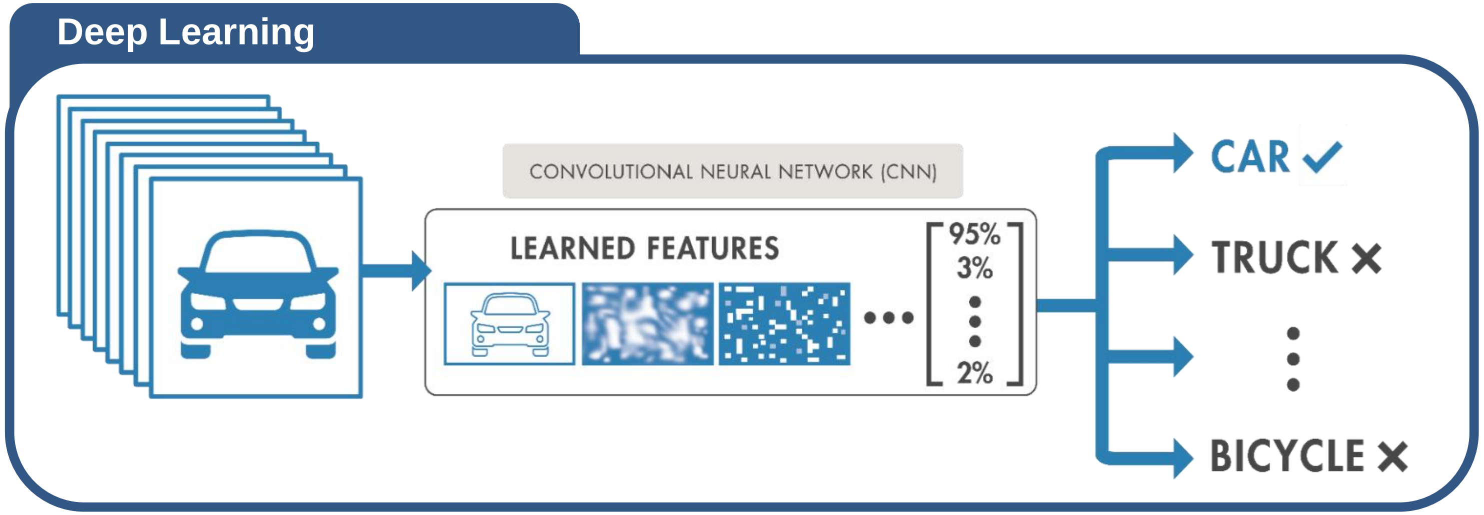
Image Labeller APP



What is Deep Learning?

- Subset of machine learning with automatic feature extraction
  - Learns features and tasks directly from data
- Accuracy can surpass traditional ML Algorithms





Deep Learning Workflow

Data Preparation	AI Modeling	Simulation & Test	Deployment
Data cleansing and preparation	Model design and tuning	Integration with complex systems	Embedded devices
Human insight	Hardware accelerated training	System simulation	Enterprise systems
Simulation-generated data	Interoperability	System verification and validation	Edge, cloud, desktop

Today we'll focus on this part

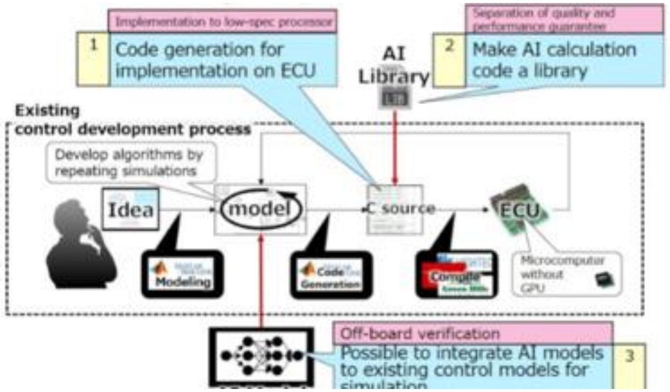
Deep Learning and AI in Industry

Oversteering Detection

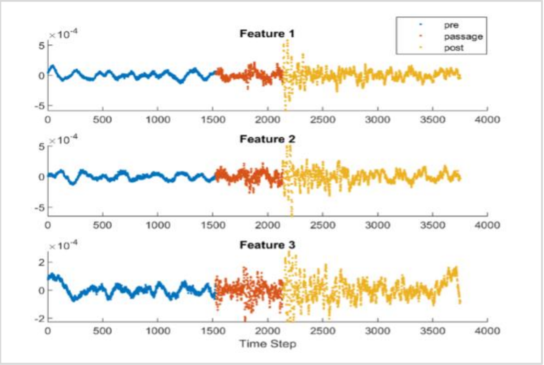
Digital Twins of Compressors

Energy use optimization

Automatic Defect Detection



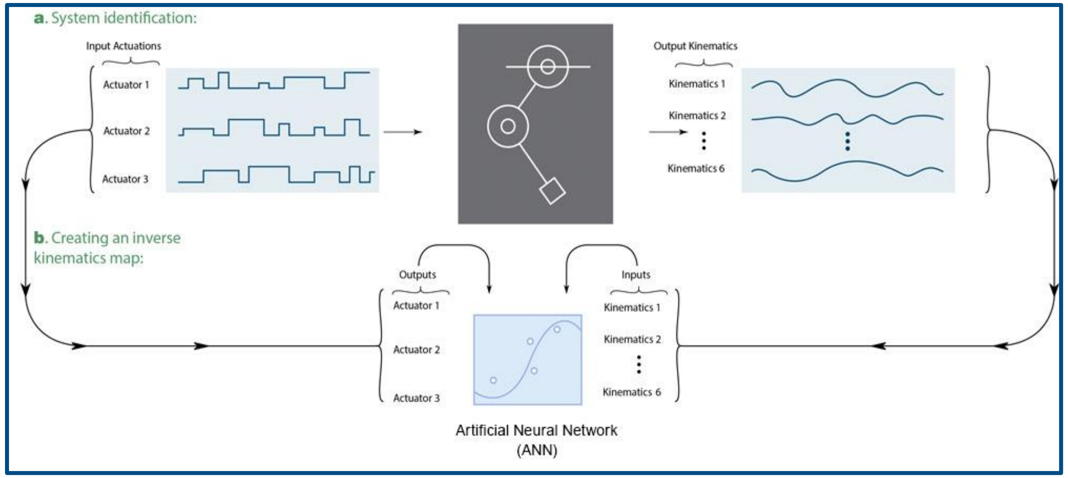
ECU Vehicle Control



Seismic Event Detection

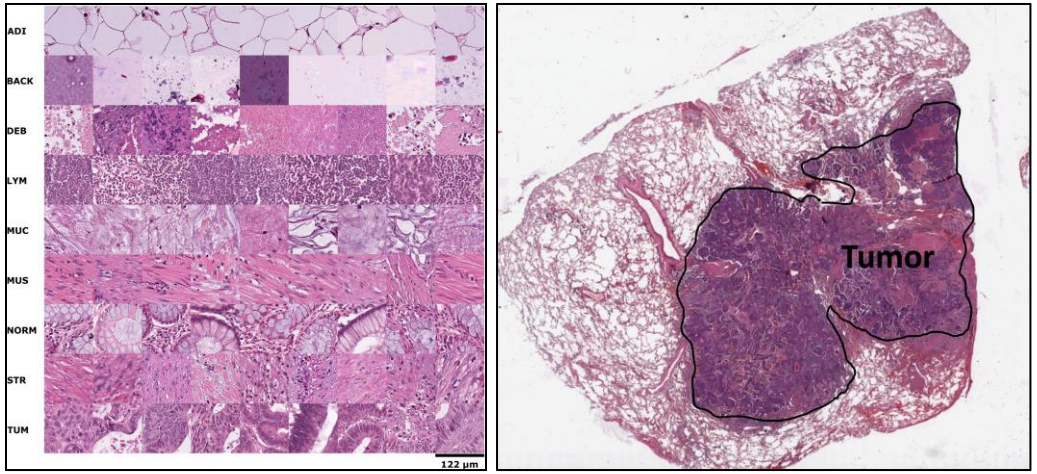
Deep Learning and AI in Research

University of Southern California



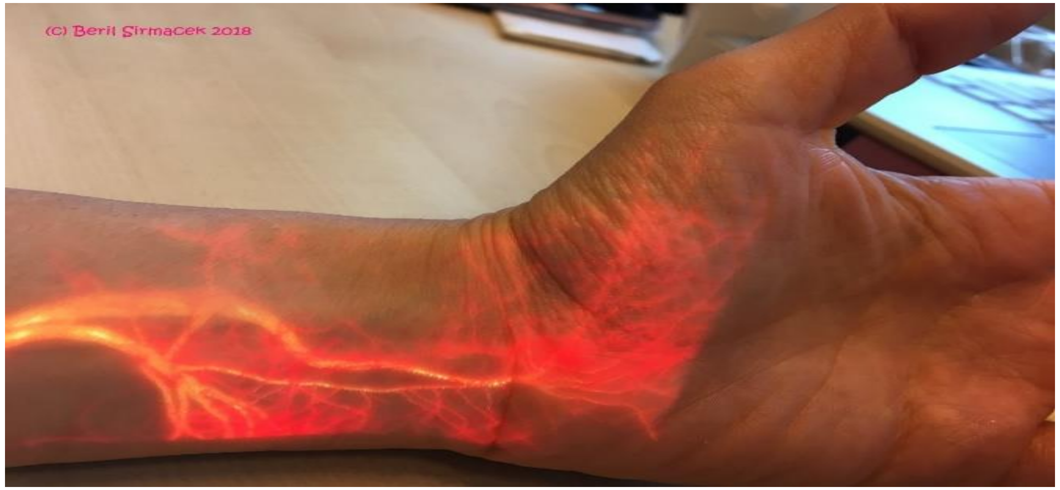
[Reinforcement Learning for Robotic Arm]

DKFZ Heidelberg



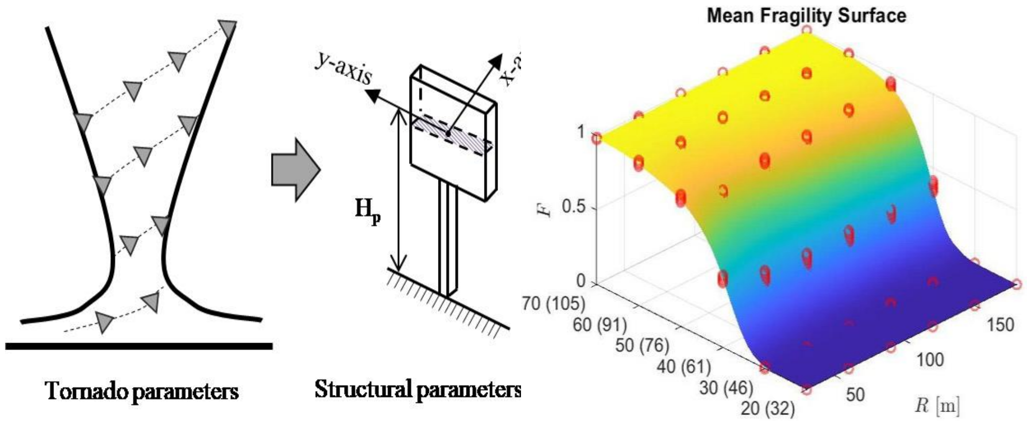
[Deep Learning for Tumor Detection]

University of Twente



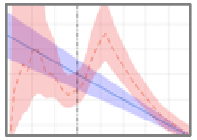
[Augmented Reality of blood flow]

Northeastern University



[Neural Networks simulate tornadic wind load]

MathWorks Focus on Deep Learning and AI for Engineering and Science

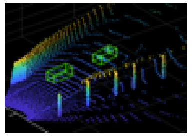


Predictive Maintenance
- Bearing Prognosis
- Pump Fault Diagnosis

Predictive Maintenance Toolbox™

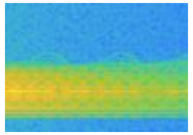
Land-Use Classification
- Semantic Segmentation for Multispectral Images

Image Processing Toolbox™



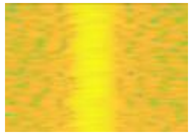
Lidar
- Lidar Point Cloud Semantic Segmentation
- 3-D Object Detection Using PointPillars

Lidar Toolbox™



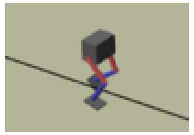
Radar
- Radar Waveform Classification
- Pedestrian and Bicyclist Classification

Phased Array System Toolbox™



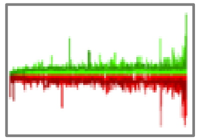
Wireless Communications
- Modulation Classification
- Detect WLAN Router Impersonation

Communications Toolbox™



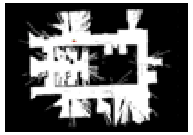
Reinforcement Learning
- Train Biped Robot to Walk
- PMSM Motor Control

Reinforcement Learning Toolbox™



Computational Finance
- Machine Learning for Statistical Arbitrage

Financial Toolbox™

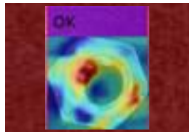


Robotics
- Avoid Obstacles using Reinforcement Learning

Robotics System Toolbox™

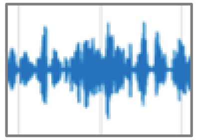
Automated Driving
- Deep Learning Vehicle Detector
- Occupancy Grid with Semantic Segmentation

Automated Driving Toolbox™



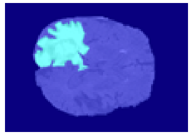
Visual Inspection
- Manufacturing Defect Detection
- Anomaly Detection for Cloth Manufacturing

Image Processing Toolbox™



Audio
- Speech Command Recognition
- Cocktail Party Source Separation

Audio Toolbox™

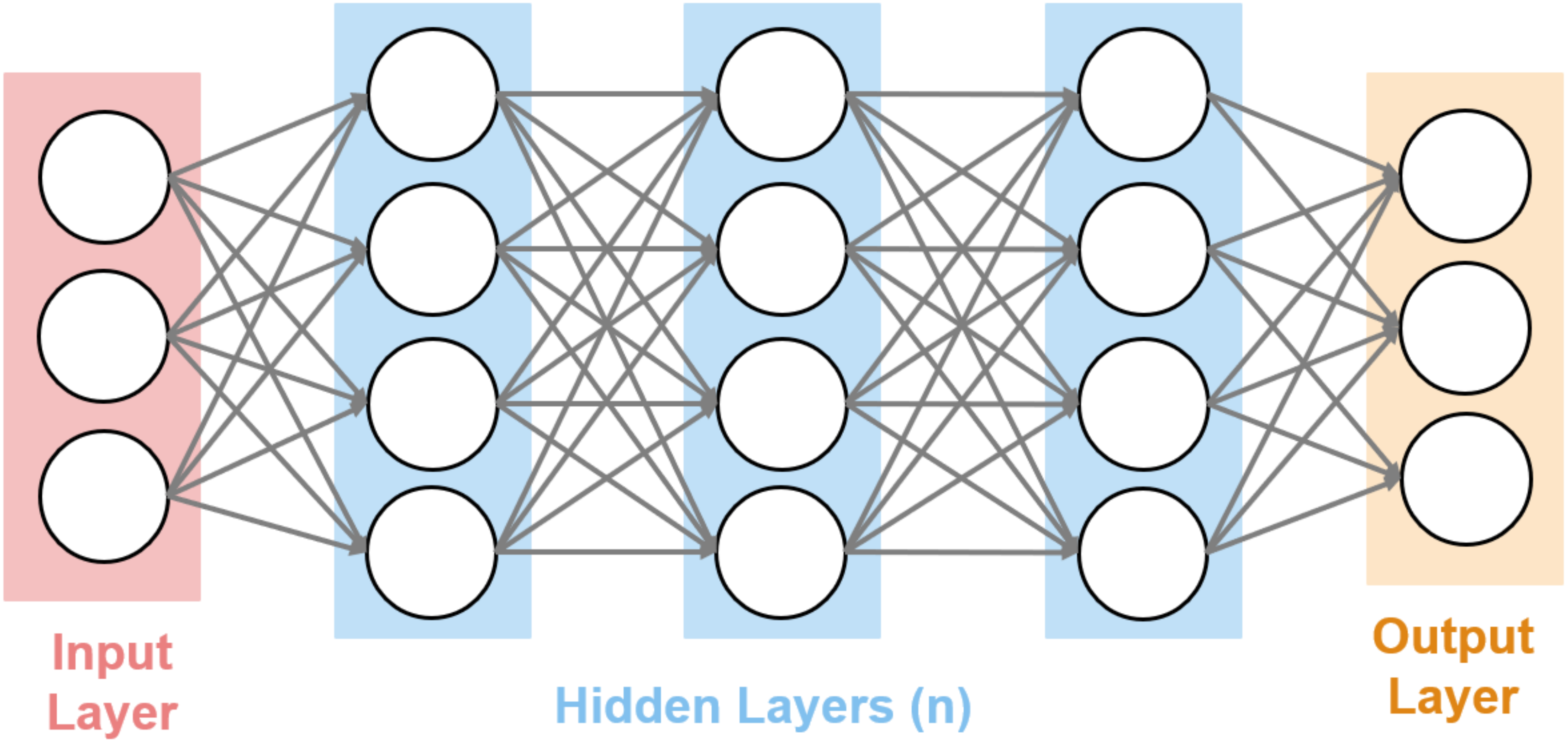


Medical Imaging
- 3-D Brain Tumor Segmentation
- Breast Cancer Tumor Classification

Image Processing Toolbox™

Deep Learning Models are Neural Networks

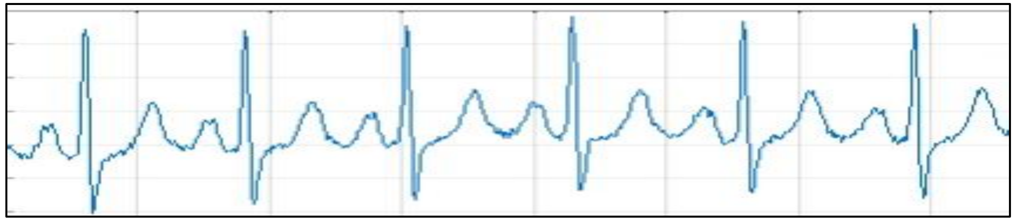
- Deep neural networks have many layers
- Data is passed through the network, and the layer parameters are updated (training)



Deep Learning Networks Take in Numeric Data

199	206	208	201	188	178	165	164	180
202	205	202	188	176	169	178	186	183
203	206	189	178	181	183	182	154	87
203	192	184	186	177	167	153	181	192
191	182	176	166	153	141	136	180	227
166	165	154	154	138	137	169	170	211
158	150	145	183	144	156	158	154	179
143	51	98	144	129	130	143	178	123
107	50	33	95	152	173	192	159	87
104	100	84	120	132	172	131	64	94
119	101	97	81	90	109	87	106	111
127	122	110	97	108	120	133	131	134
111	117	108	119	131	143	146	141	156
126	122	113	119	139	142	155	161	151
129	126	130	111	103	130	149	149	156
138	128	136	144	136	129	134	122	145
154	133	134	141	168	150	126	127	151

Images are a numeric matrix



131	158	185	212	239	266	293	320	347	374	376

Signals are numeric vectors

The Bird Flies = [0 13 5 6]

The Leaf Is Brown = [13 3 11 2]

Text is processed as numeric vectors

[Deep Learning Documentation](#)

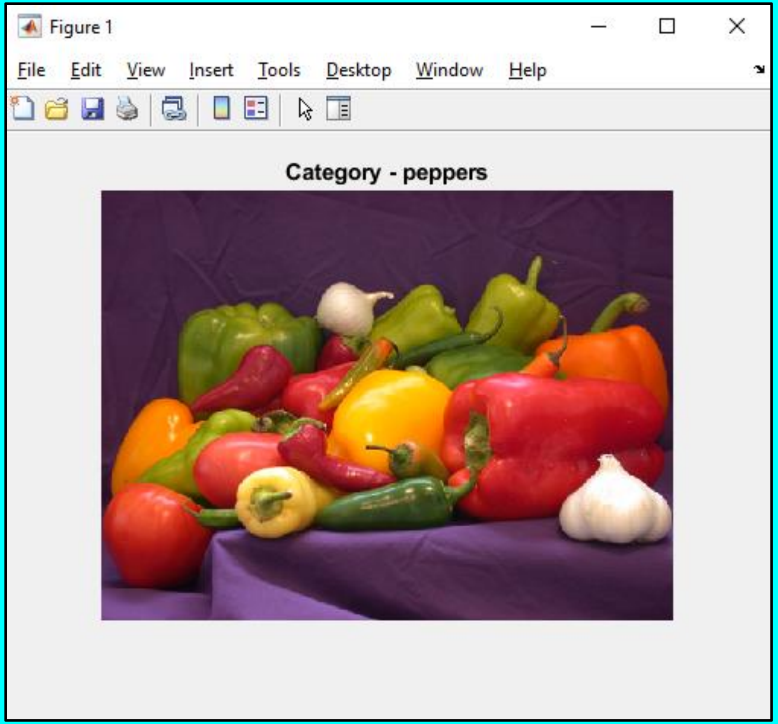
Exercise 1 – Deep Learning in 6 Lines of Code

Purpose:

- Use a **PRETRAINED** neural network to classify an image
- Introduction to some basic functions

To Do:

1. Open `Work_GettingStarted.mlx`
2. Follow along with instructor



What we just did

`alexnet( )`

`analyzeNetwork( )`

`classify( )`

Help

`imresize( )`

384 x 512 x 3

Resize image

227 x 227 x 3

Pre-TRAINED CNN

```
this_is_a_what = categorical
     bell pepper
```

Let's discuss the following

1. What is a convolutional Neural Network (CNN) ?
2. How do you assemble them ?
3. How do you train them ?
4. How do I assess their performance ?
    - Is it good at what it does ?

We Can Build Networks from Scratch or Use Pretrained Models

- Pretrained models have predefined layer orders and parameter values
- Can be used directly for inference (AlexNet Example)

AlexNet
VGG-16
VGG-19
GoogLeNet

Get started with these Models

ResNet-18
ResNet-101
ResNet-50
Inception-v3
DenseNet-201
Xception

Effective for object detection and semantic segmentation workflows

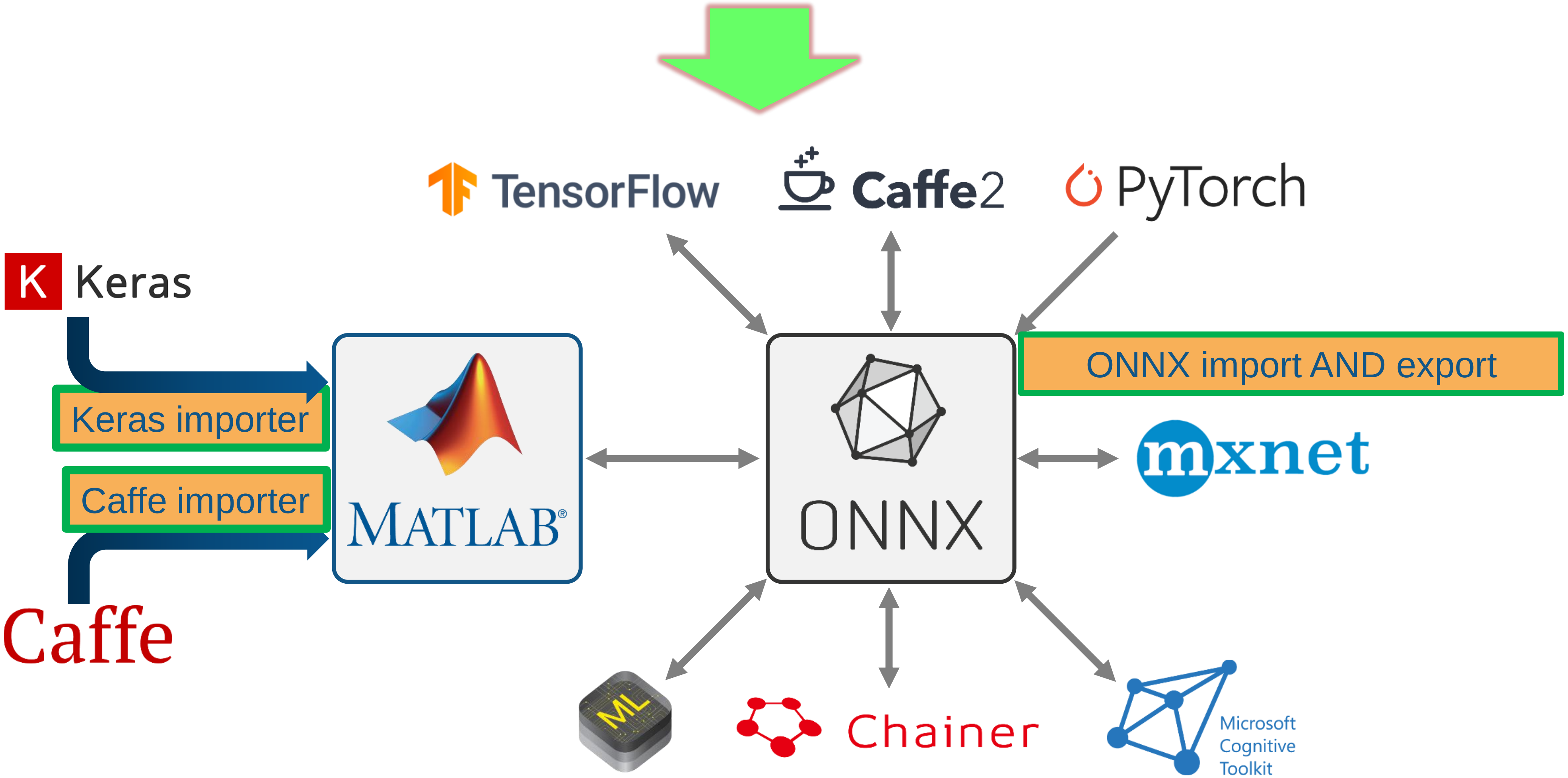
SqueezeNet
MobileNet-v2
ShuffLeNet

Lightweight and computationally efficient

Full list of models available HERE

Access Pretrained Models from Within MATLAB or Import from the Web

https://www.mathworks.com/help/releases/R2021a/deeplearning/deep-learning-import-export-and-customization.html



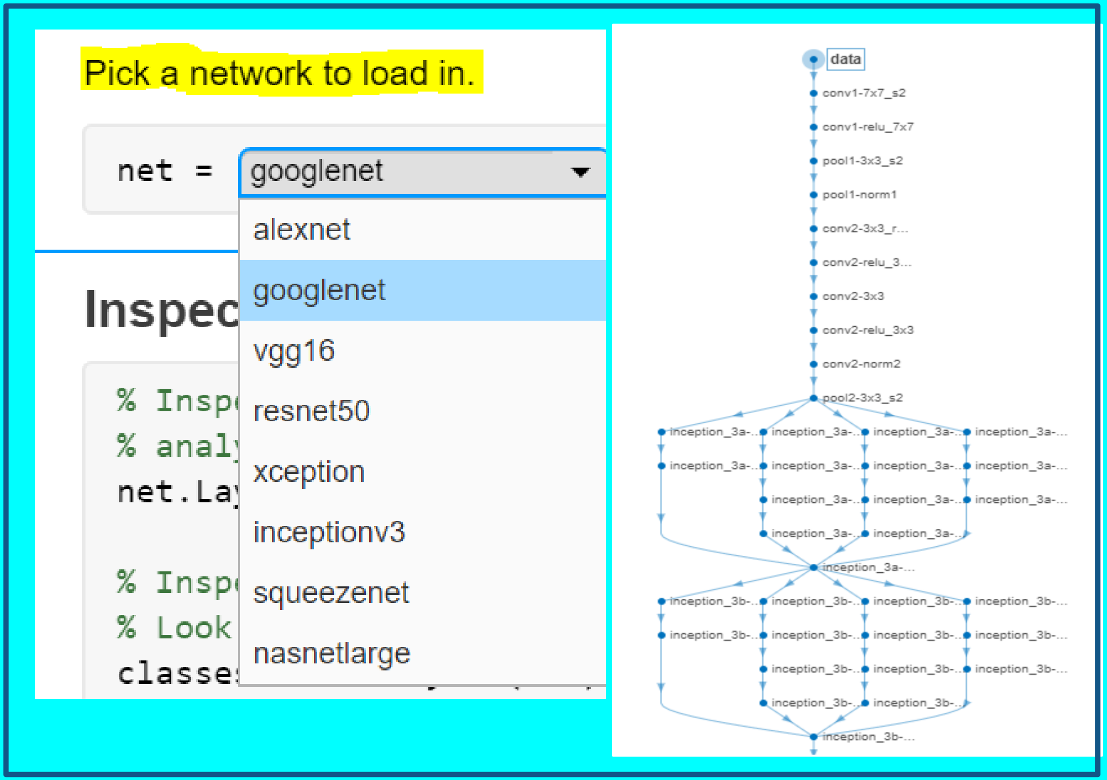
Exercise 2 – Models

Purpose:

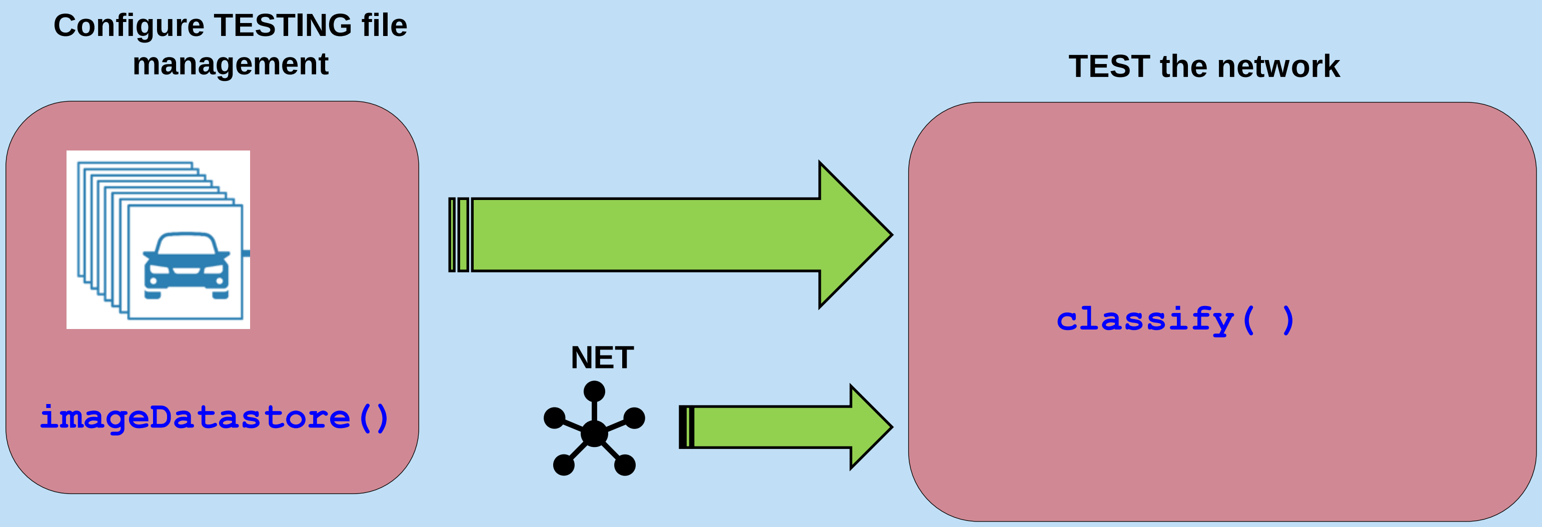
- Classify Images using pretrained models.
- See how different network architectures affect results.
- Use **datastores** to access data efficiently

To Do:

1. Open `Work_Models.mlx`



What we just did



Pretrained models aren't always enough. We may have to build and train networks from scratch

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

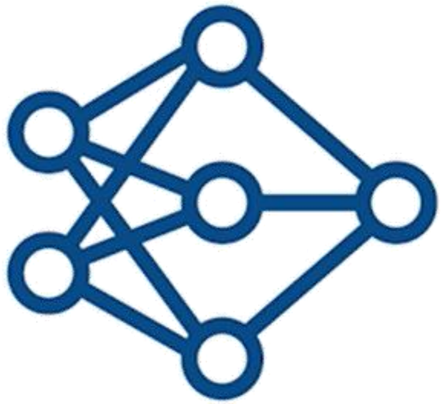
- Embedded devices
- Enterprise systems
- Edge, cloud, desktop

…. So let's talk about building and training models from scratch

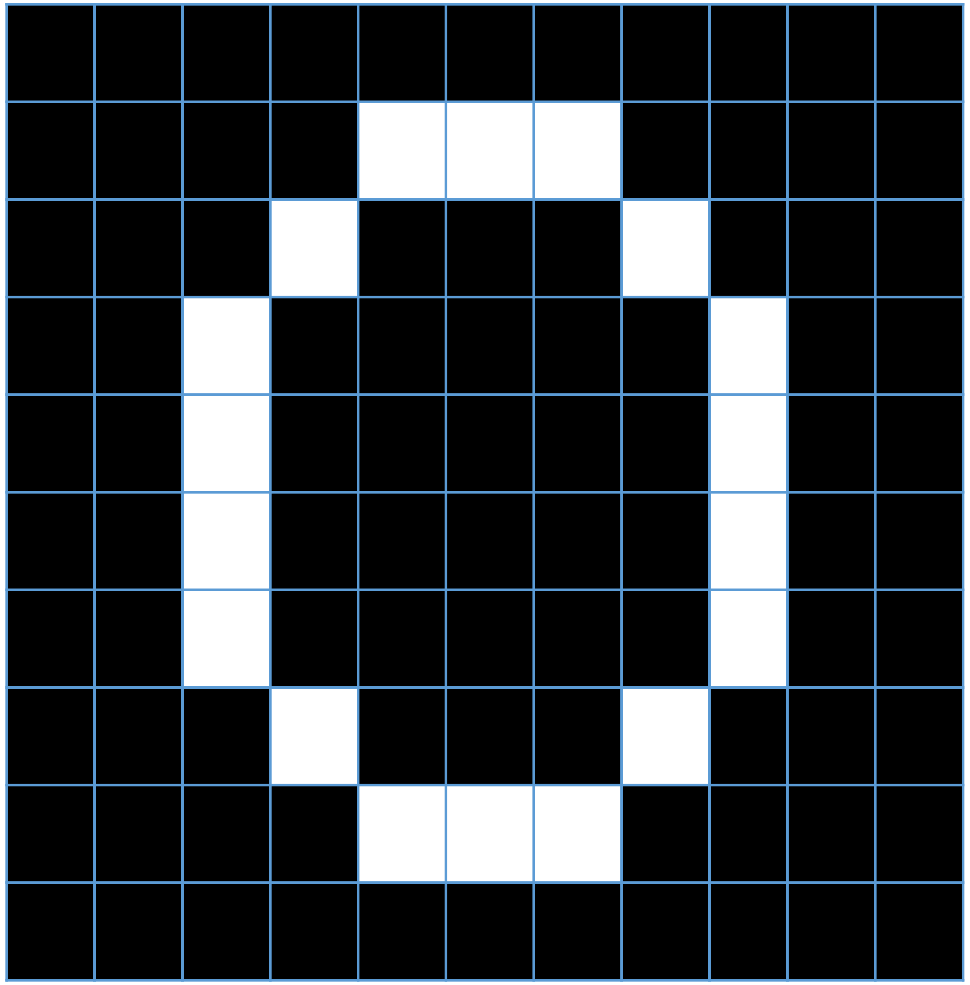
Creating Layer Architectures

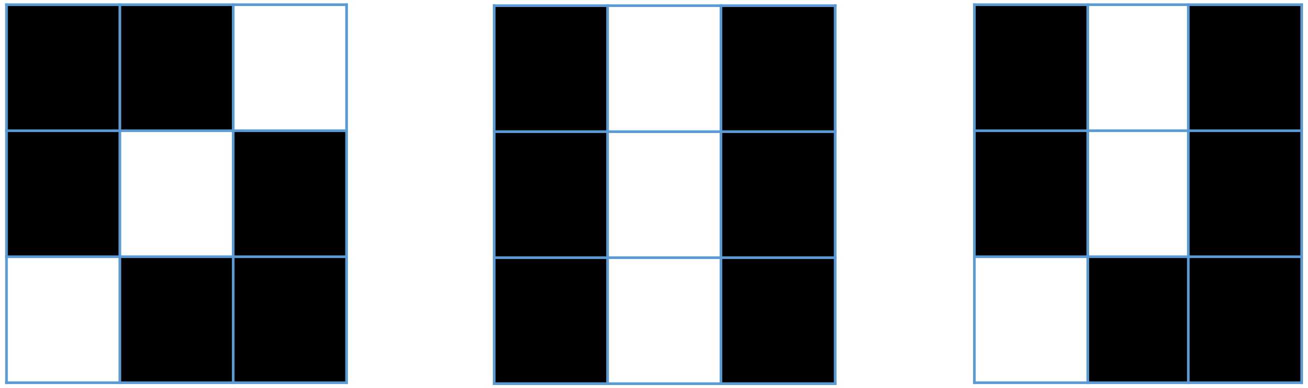
- Convolution Neural Networks – CNN
- Special layer combinations that make them adept at classifying images

- Convolution Layer
- ReLU Layer
- Max Pooling Layer

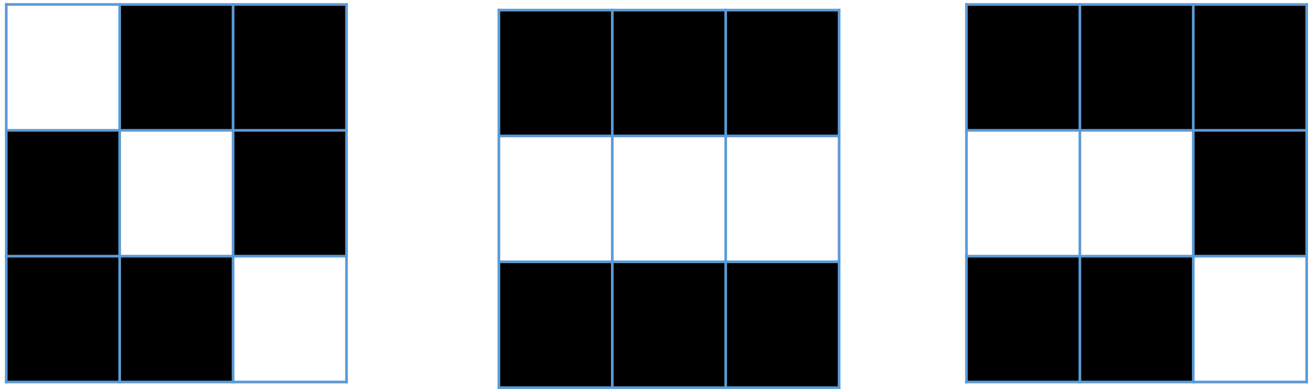


Convolution Layers Search for Patterns

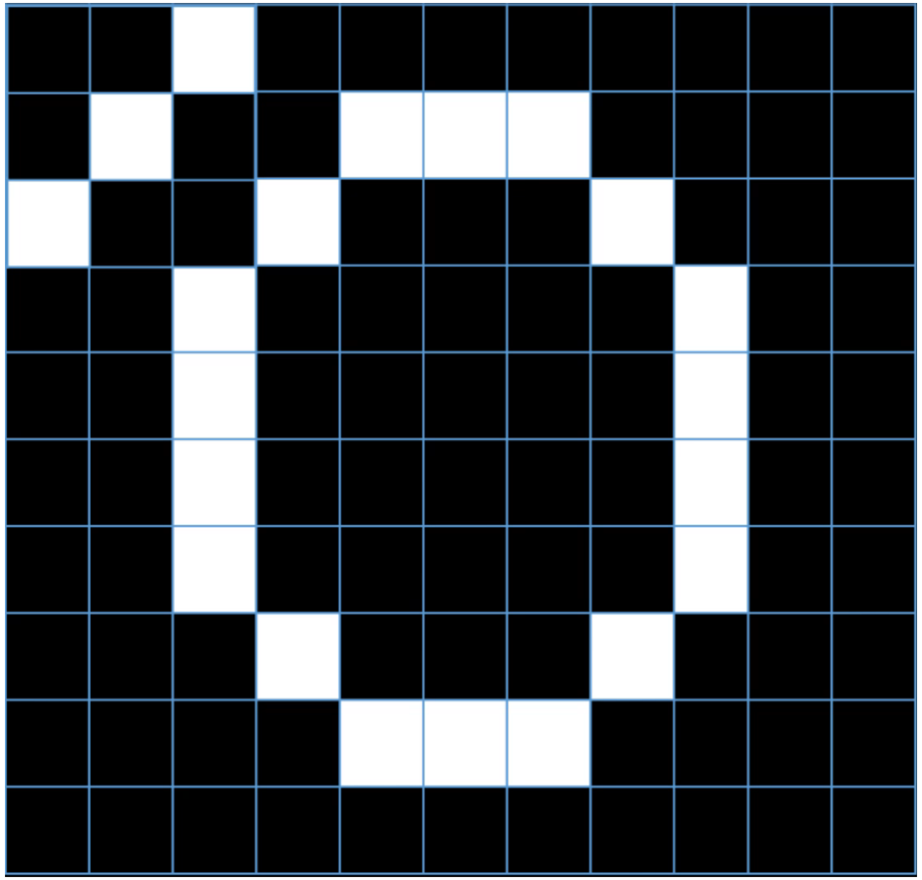


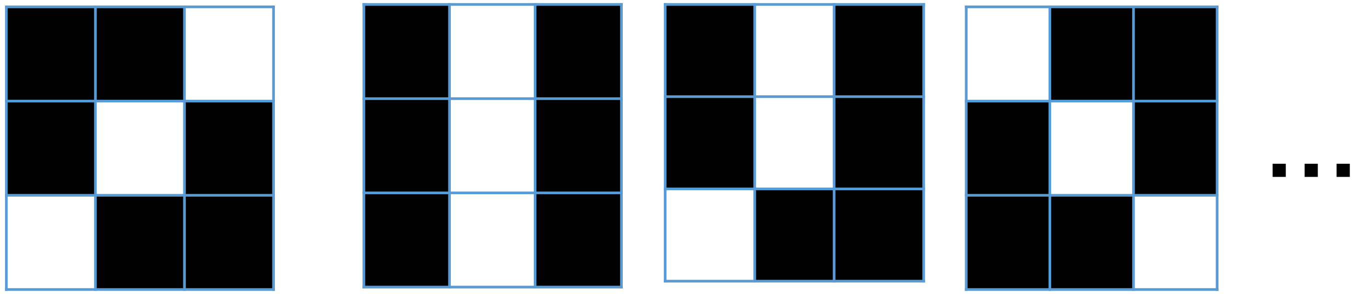


These patterns would be common in the number 0



All patterns are compared to the patterns on a new image.

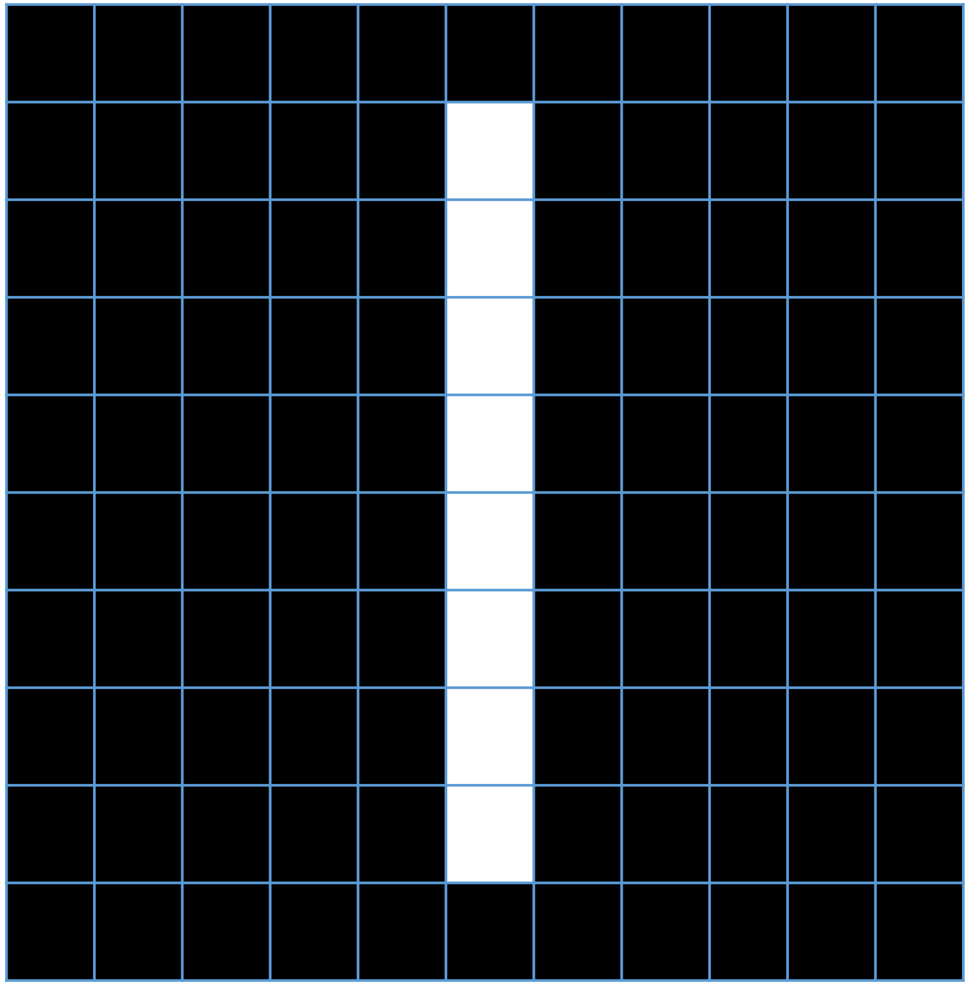




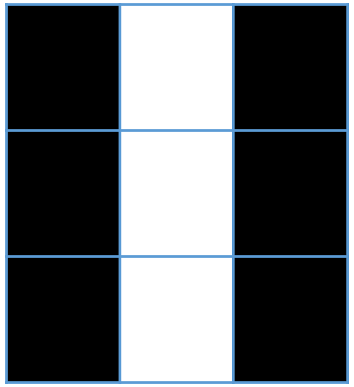
- Pattern starts at left corner
  - Perform comparison
  - Slide over one pixel
- Reach end of image
- Repeat for next pattern

Play the video !

Good pattern matching in convolution improves chances that object will classify properly

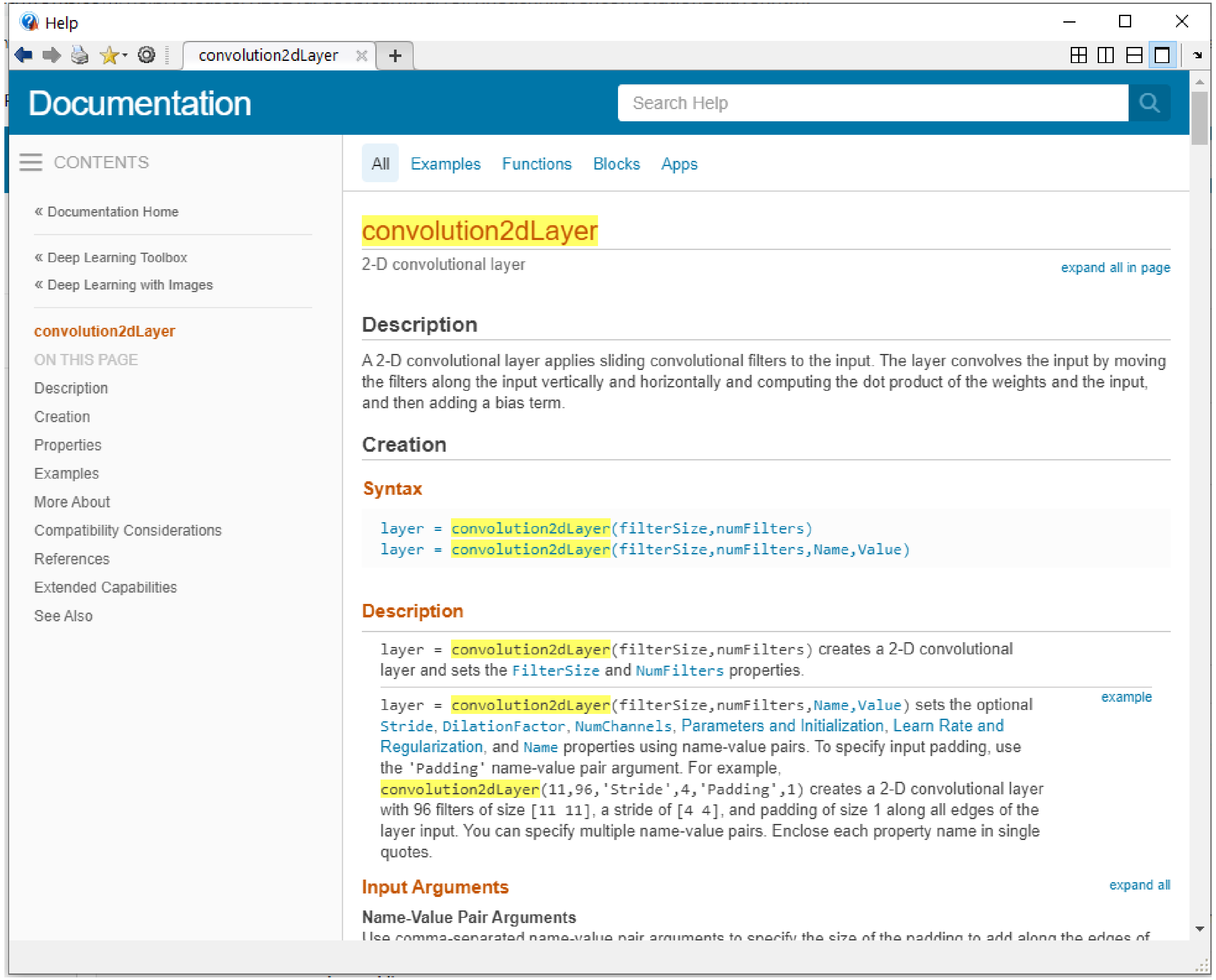


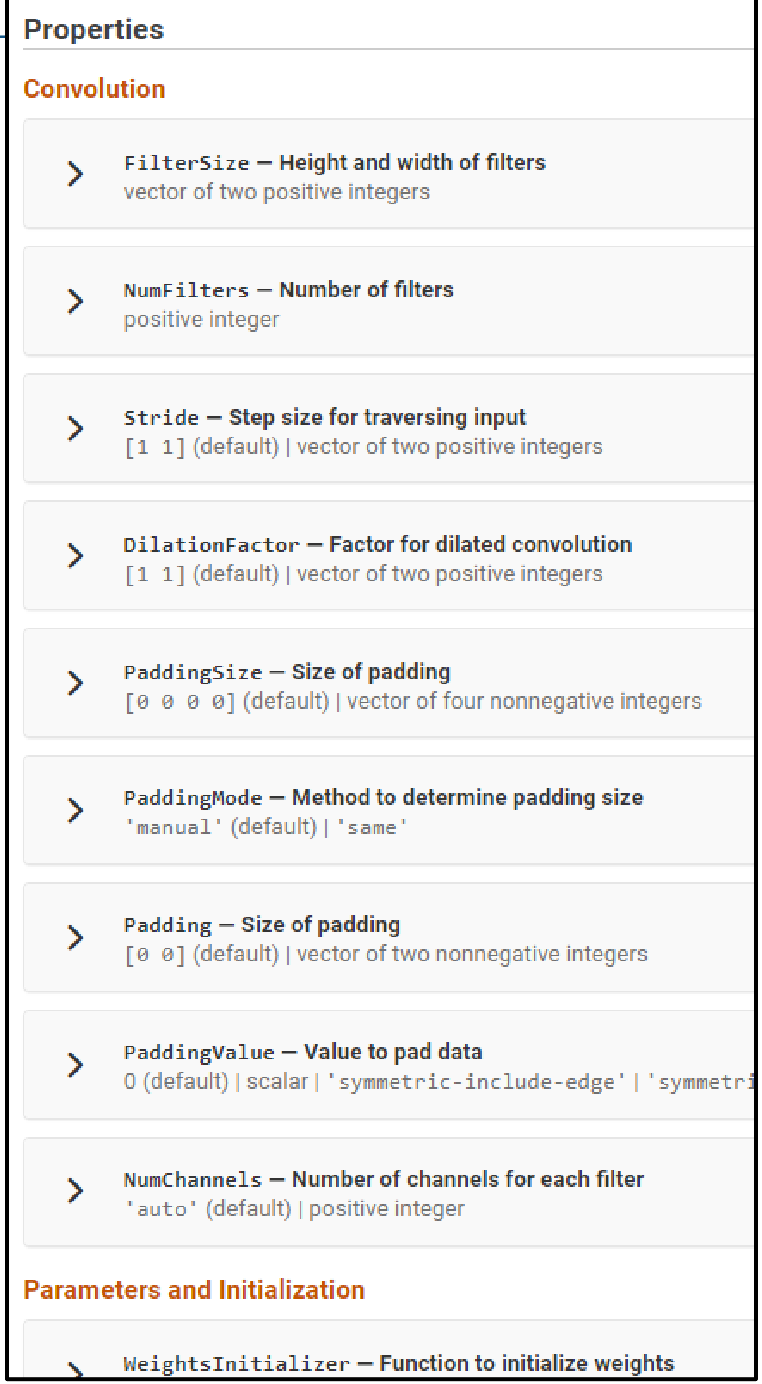
- This image would not match well against the patterns for the number zero
- It would only do very well against this pattern



Convolution Layers attributes

`convolution2dLayer()`





Rectified Linear Units Layer (ReLU)

Converts negative numbers to zero

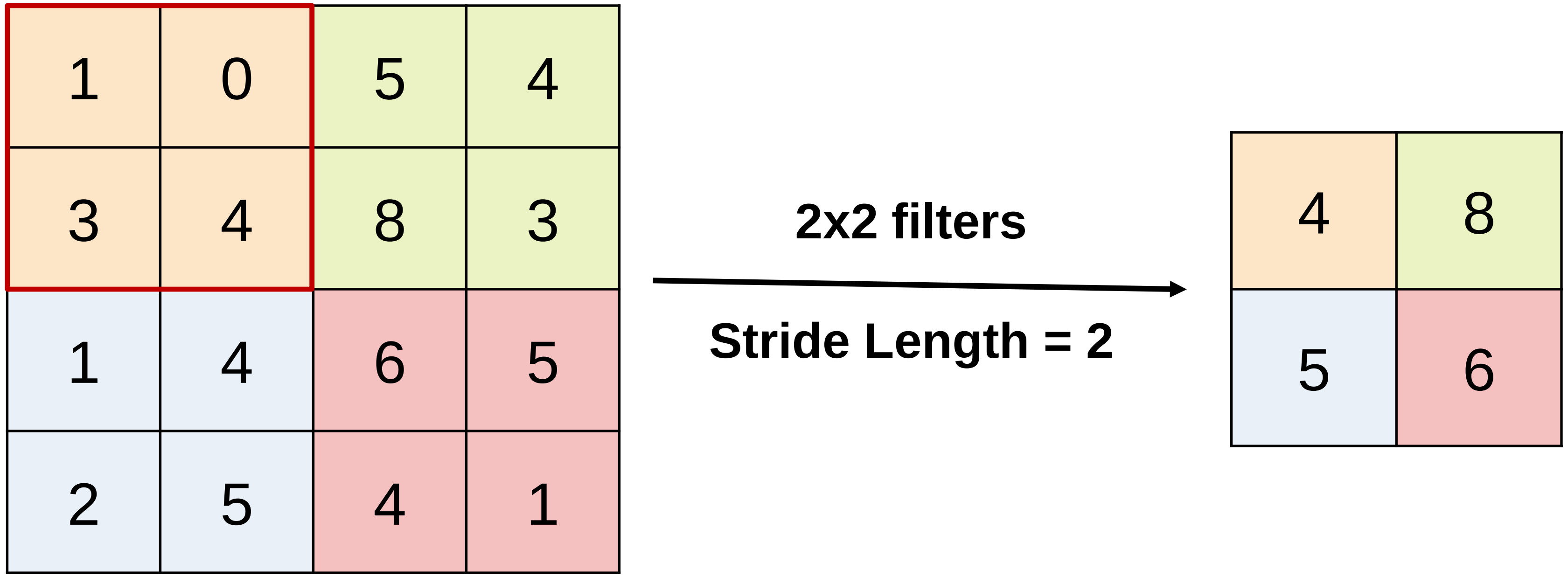
-1	0	5	4
3	-4	-8	3
1	4	6	-5
-2	-5	4	1

0	0	5	4
3	**0**	**0**	3
1	4	6	**0**
0	**0**	4	1

MathWorks

Max Pooling is a down-sampling operation

Shrink large images while preserving important information

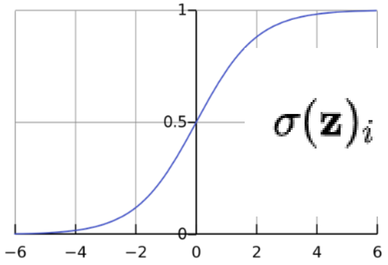


Last layers

- **Classification problems end with 3 Layers**

A **Fully Connected Layer**
  - looks at which high-level features correspond to a specific category
  - calculates scores for each category (highest score wins)
  - “flattens” the matrix into a column vector

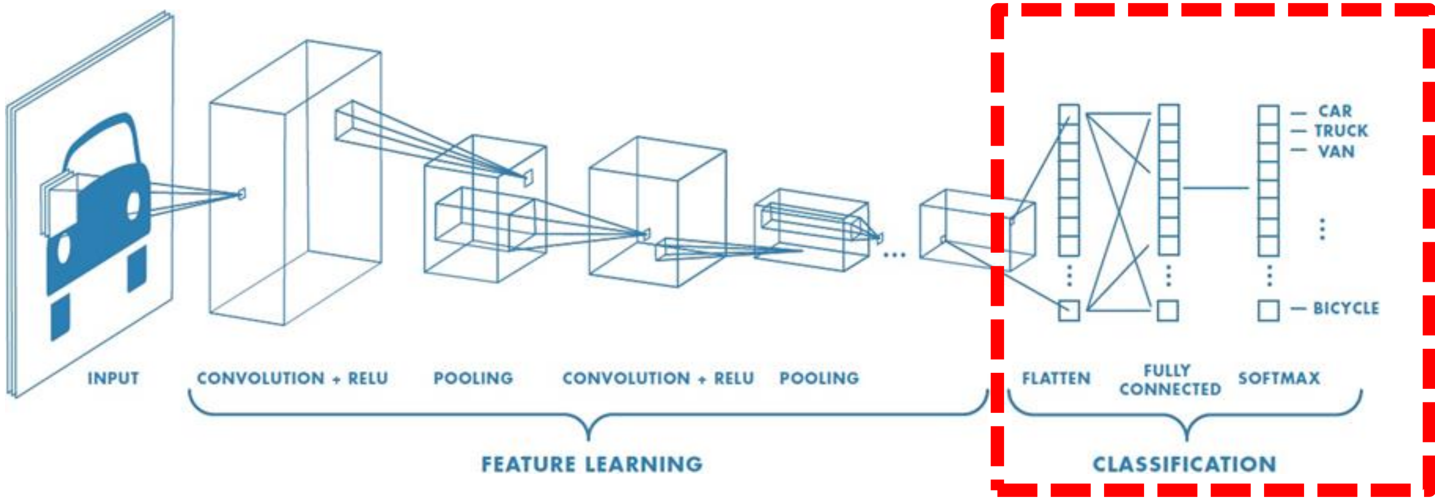
B **Softmax Layer**
  - turns scores into probabilities

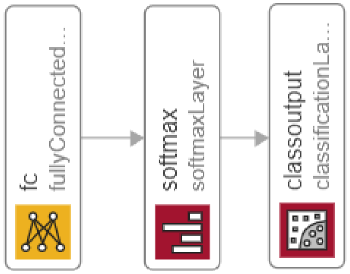


$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

C **Classification Layer**
  - categorizes image into one of the classes that the network is trained on

- Note: regression problems end with 2 layers
  – Fully Connected Layer
  – Regression Layer





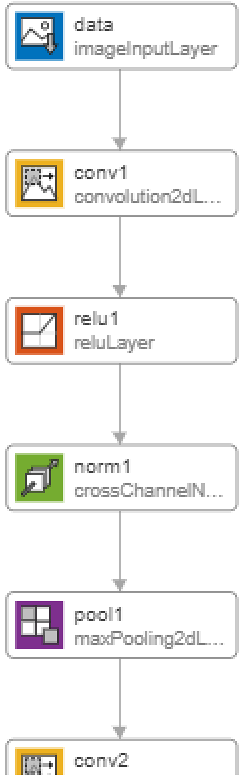
A B C

How does Deep Learning work?

- **Hyperparameters** are set manually
  - are not updated during training
  - number of hidden layers, learning rate, minibatch size, epochs, activation function, etc.

- **Learnable Parameters** are chosen by the network
  - calculated within the neurons
  - adjusted during training by comparing the predicted (final) output with the actual output

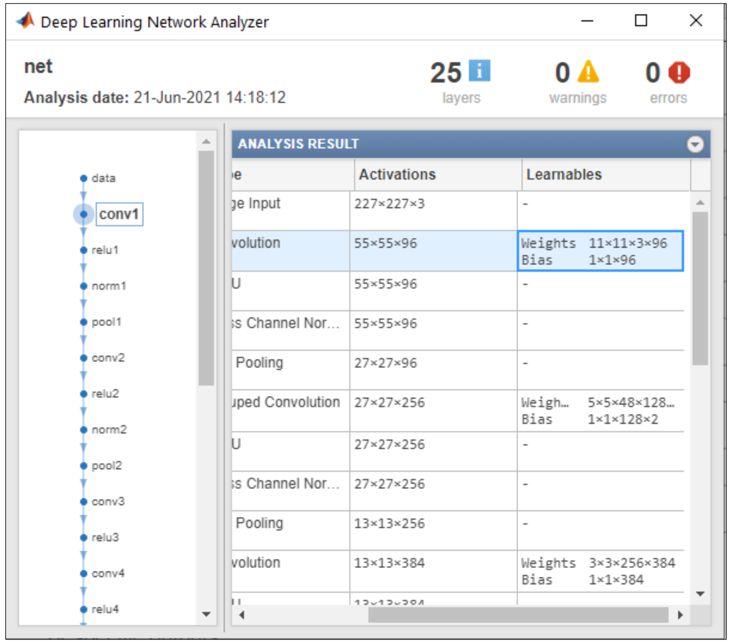
- The deeper the network, the more information is processed
  - multiple types of neural networks with different structure (e. g. CNN, RNN, GAN, and many more)
  - only considering specific regions (e. g. receptive fields in CNN) or specific outputs




```
net.Layers(2)
  Convolution2DLayer with properties:
  Name: 'conv1'

  Hyperparameters
      • FilterSize: [11 11]
      • NumChannels: 3
      • NumFilters: 96
      • Stride: [4 4]
      • DilationFactor: [1 1]
      • PaddingMode: 'manual'
      • PaddingSize: [0 0 0 0]

  Learnable Parameters
      • Weights: [11×11×3×96
        single]
      • Bias: [1×1×96 single]
```



How Do I know Which Layers to Use?

Feature Extraction - Images

- 2D and 3D convolution
- Transposed convolution (…)

Activation Functions

- ReLU
- Tanh (…)

Sequence Data

Signal, Text, Numeric

- LSTM
- BiLSTM
- Word Embedding (…)

Normalization

- Dropout
- Batch normalization
- (…)

Research papers and doc examples can provide guidelines for creating architecture.

Documentation on Network Layers

3 Components to Train any Network

Data Preparation → **Network Architecture** → **Training Options**

"How much data do I need?"

It depends…but
A LOT

Define inputs and layers for deep learning model

Influence training time and accuracy

- Solver Type
- Initial Learn Rate
- Minibatch Size
- Max Epochs
- …

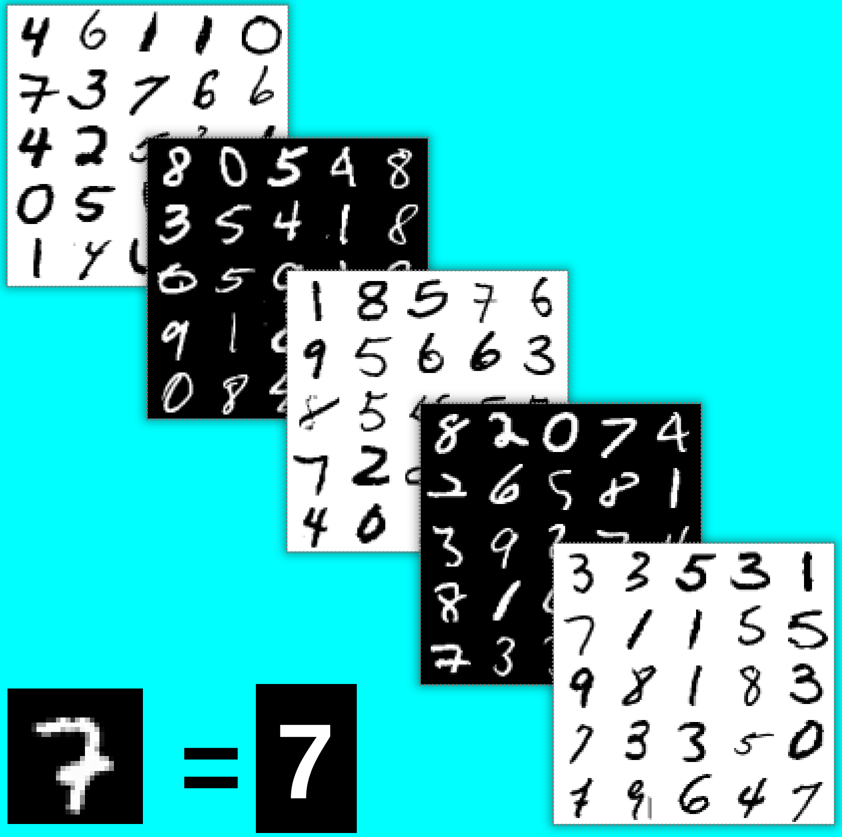
Exercise 3 - MNIST

Purpose:

- Learn how to create and train deep neural network
- Use MATLAB's **Deep Network Designer**
- Explore hyperparameters

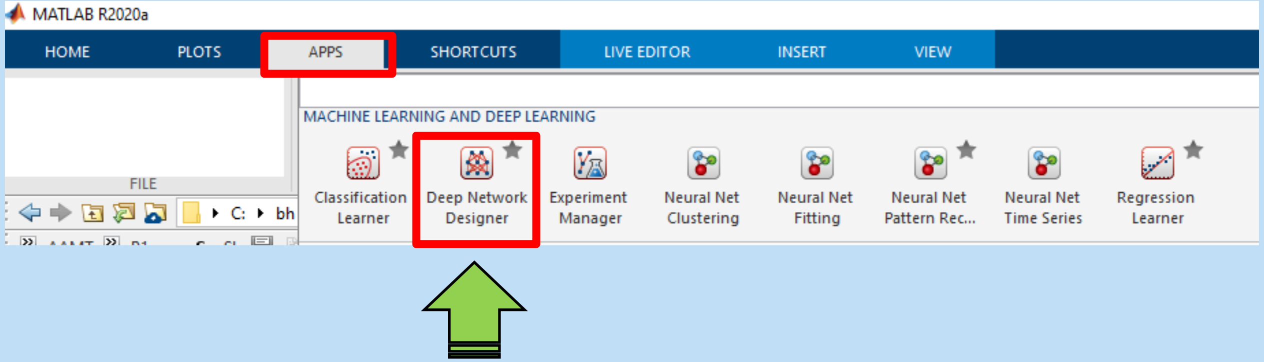
Details

- Dataset consists of handwritten digits 0-9
- 60,000 training images
- 10,000 test images

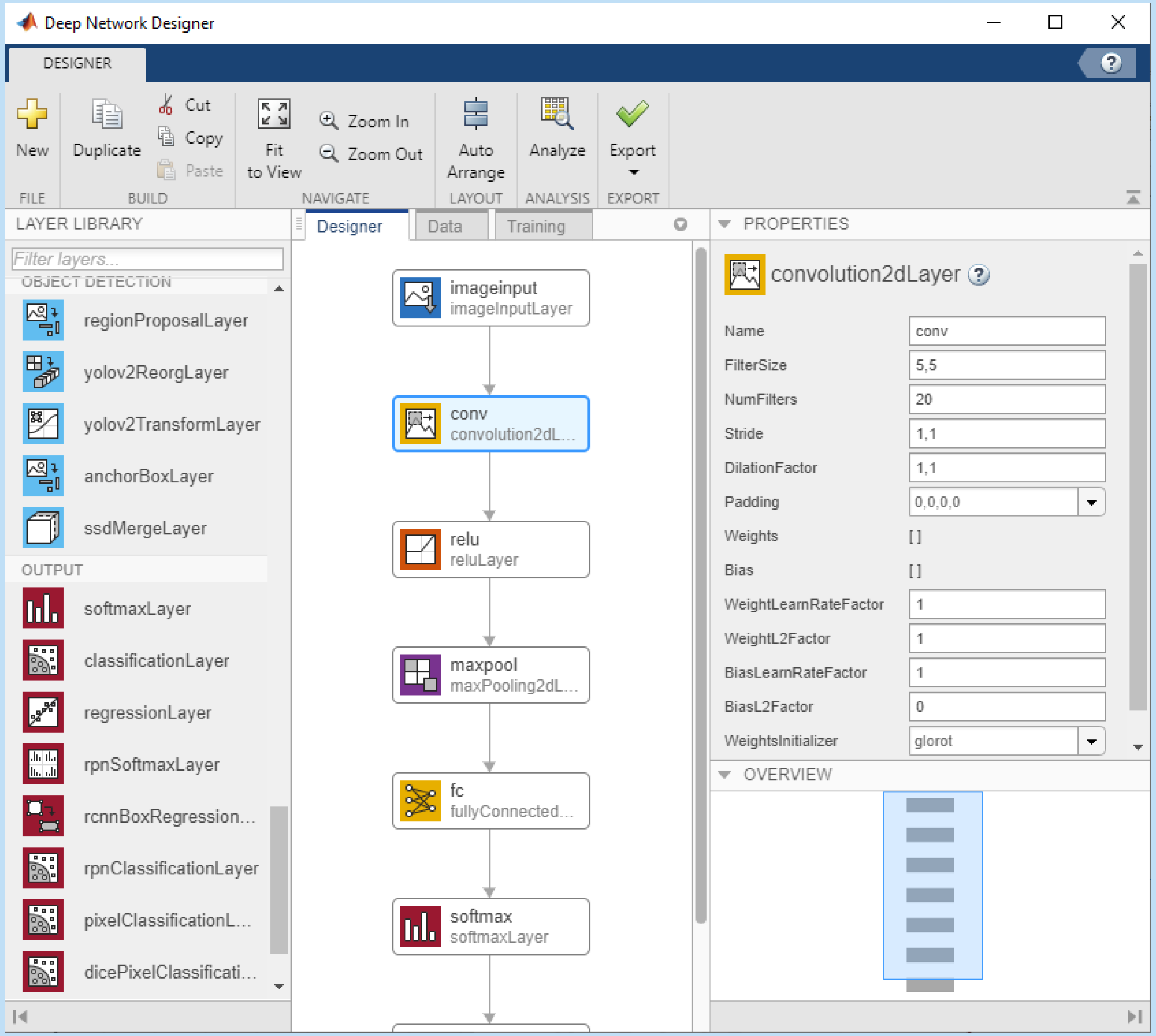


Sources: [MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges]

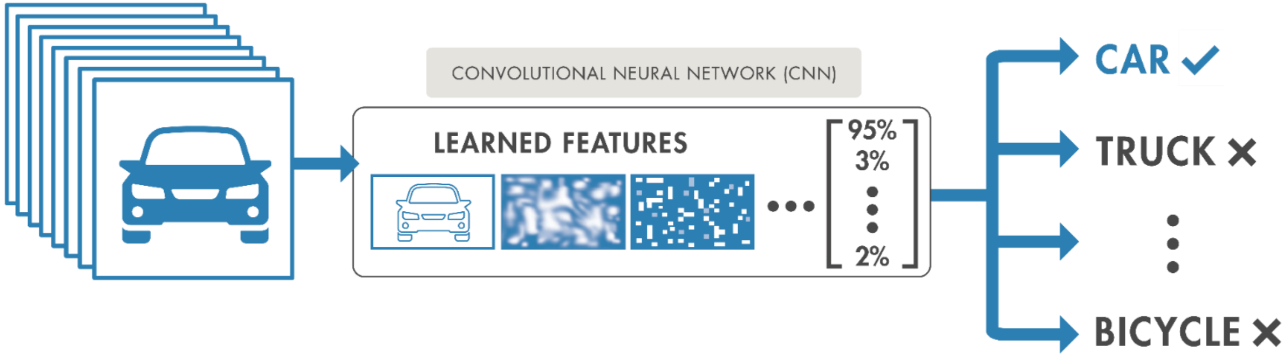
What we just did



The [Deep Network Designer](#)



- Train a deep neural network **from scratch**



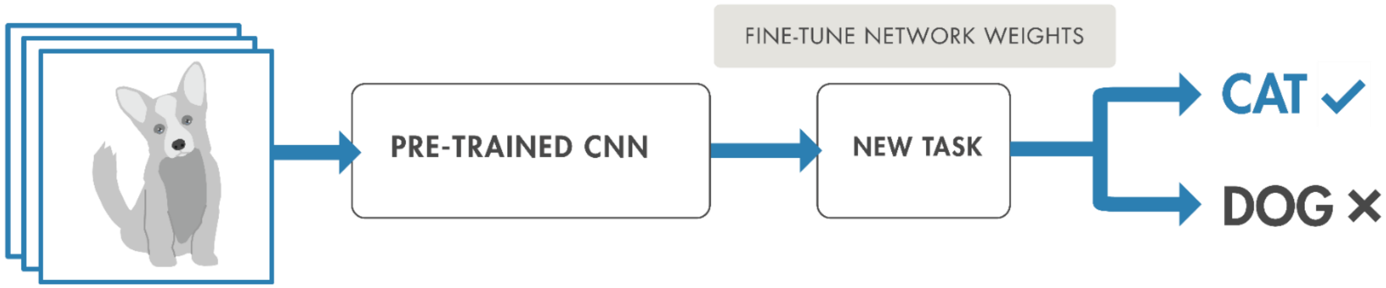
We did this in the MNIST exercise

But WHY start from scratch ?

Can we start with an already "partially" trained model

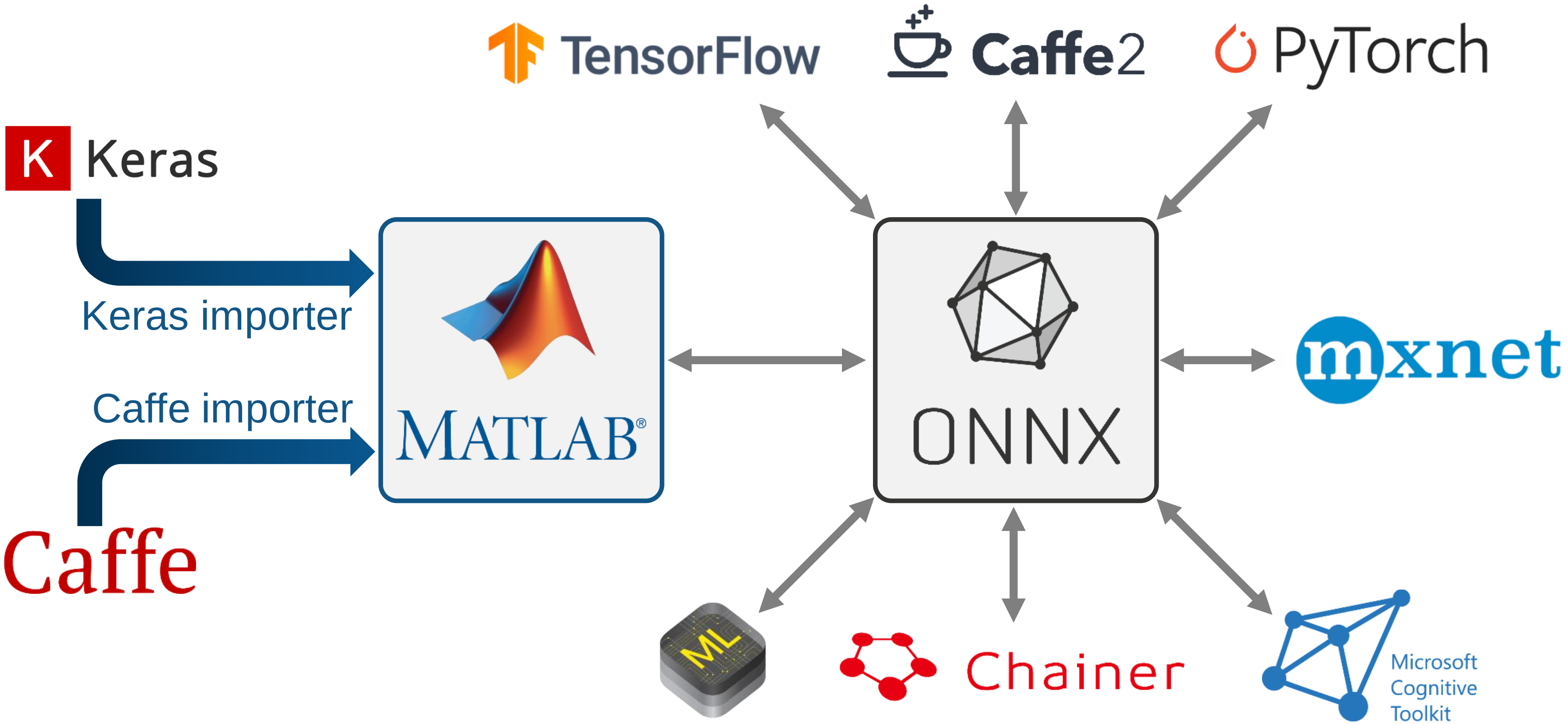
Transfer Learning

- Use a pretrained model – Transfer Learning



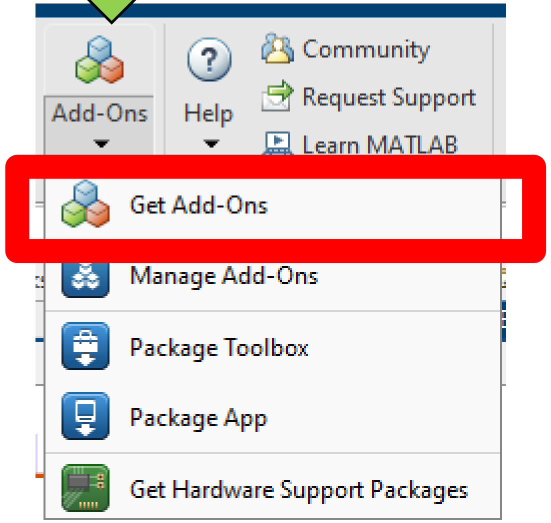
More details in the next session

Access Pretrained Models from Within MATLAB or Import from the Web



How

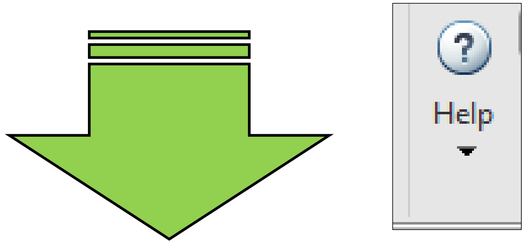
MATLAB desktop



Search:
Onnx, Keras, etc

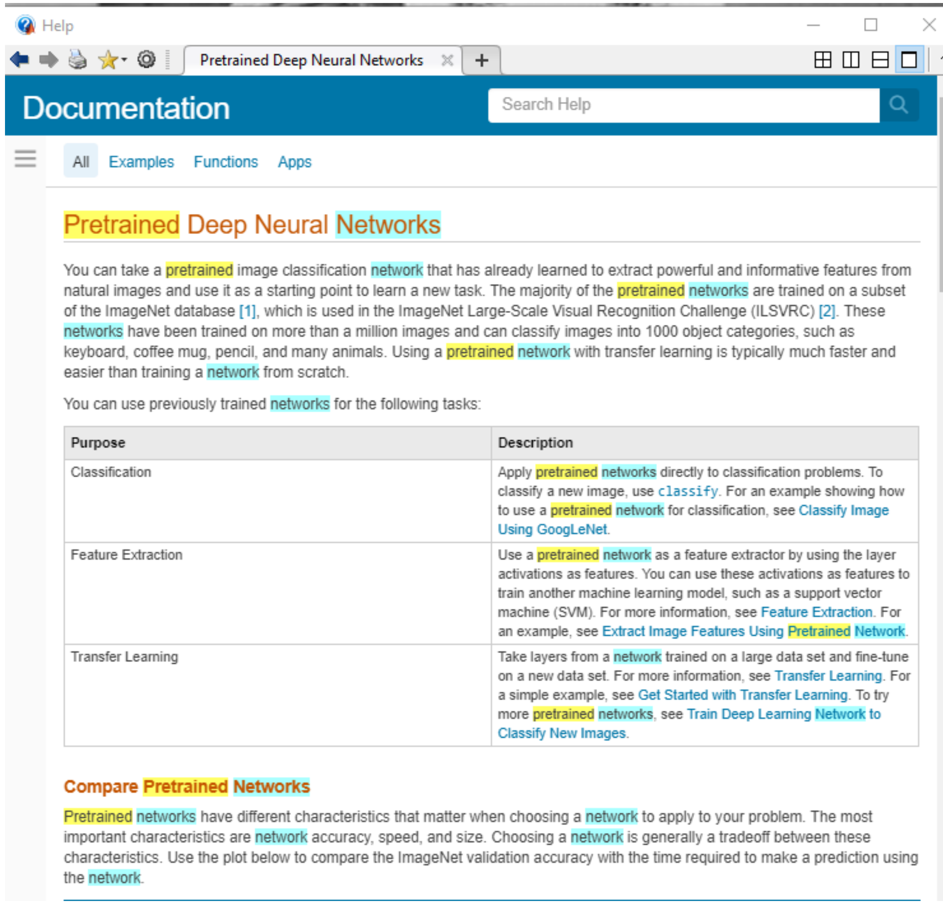
Pretrained Models

Full list of models available [HERE](#)



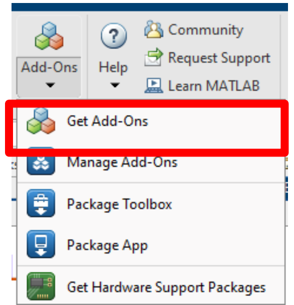
- Pretrained models have predefined layer orders and parameter values
- Can be used for inference without training

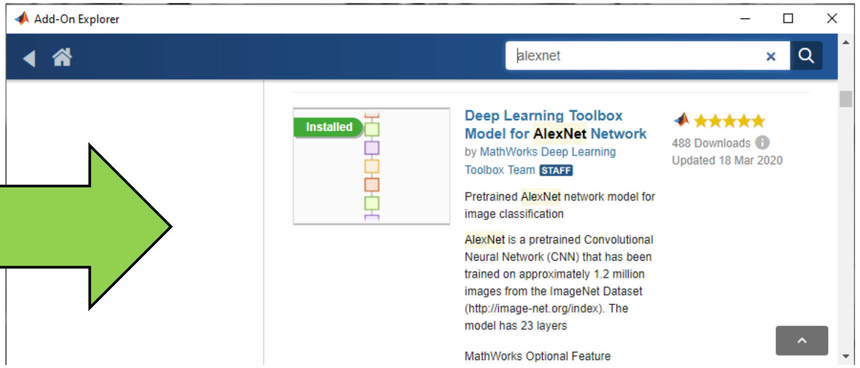
AlexNet VGG-16 VGG-19 GoogLeNet	ResNet-18 ResNet-101 ResNet-50	Inception-v3 DenseNet-201 Xception	SqueezeNet MobileNet-v2 ShuffLeNet
Get started with these Models	*Effective for object detection and semantic segmentation workflows*		*Lightweight and computationally efficient*



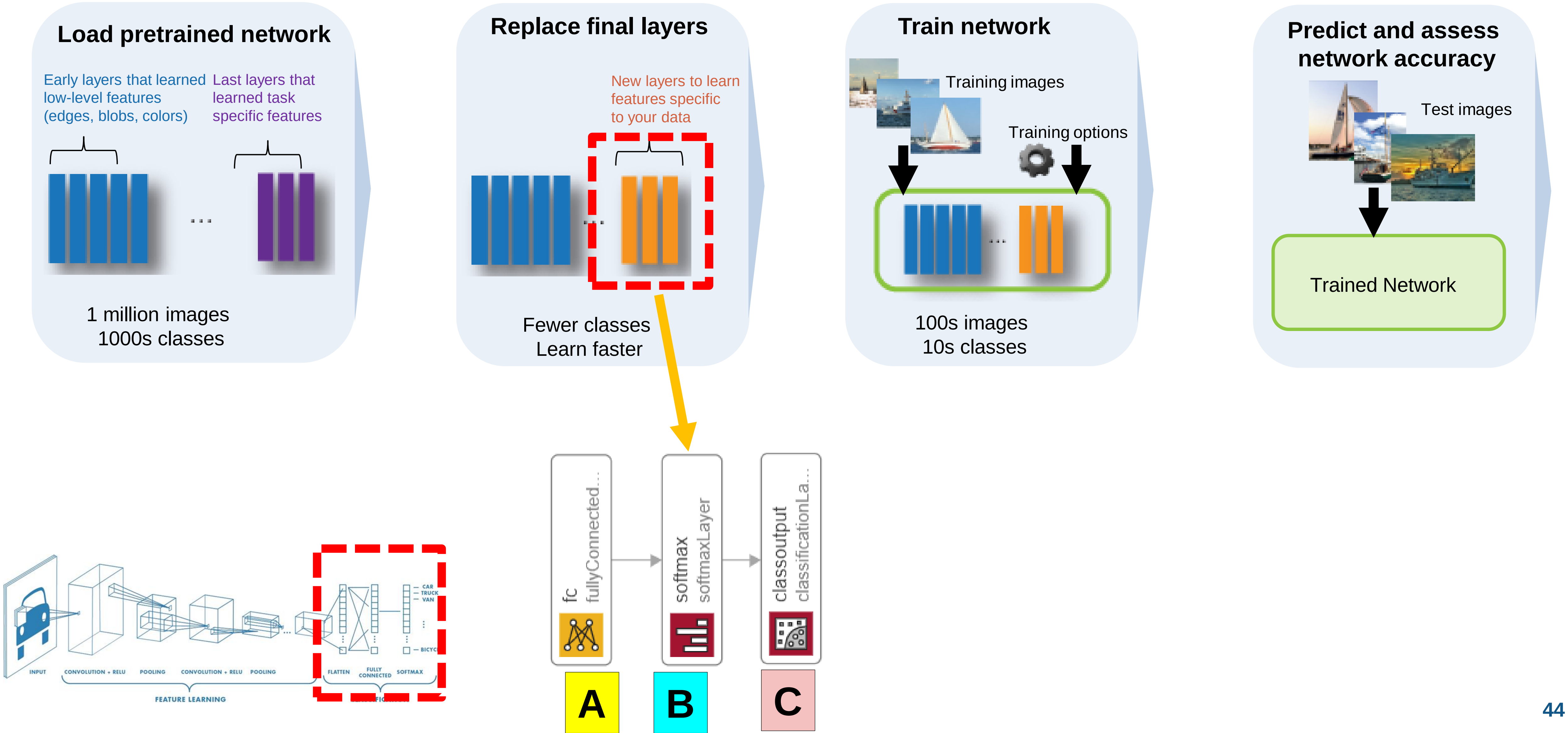
MATLAB desktop

How





Transfer Learning Workflow – model assembly



Exercise 4 – Food

Purpose:

- Use transfer learning to leverage a pretrained model to classify 5 types of food
- Visualize activations within a network

To Do:

1. Open `Work_Food.mlx`

What we just did

Configure TRAINING file management

`imageDatastore( )`

`splitEachLabel( )`

`augmentedImageDatastore( )`

Configure TRAINING options

`trainingOptions( )`

Define Network Layers

TRAIN the network

`trainNetwork( )`

NET

Configure TESTING file management

`imageDatastore()`

`augmentedImageDatastore()`

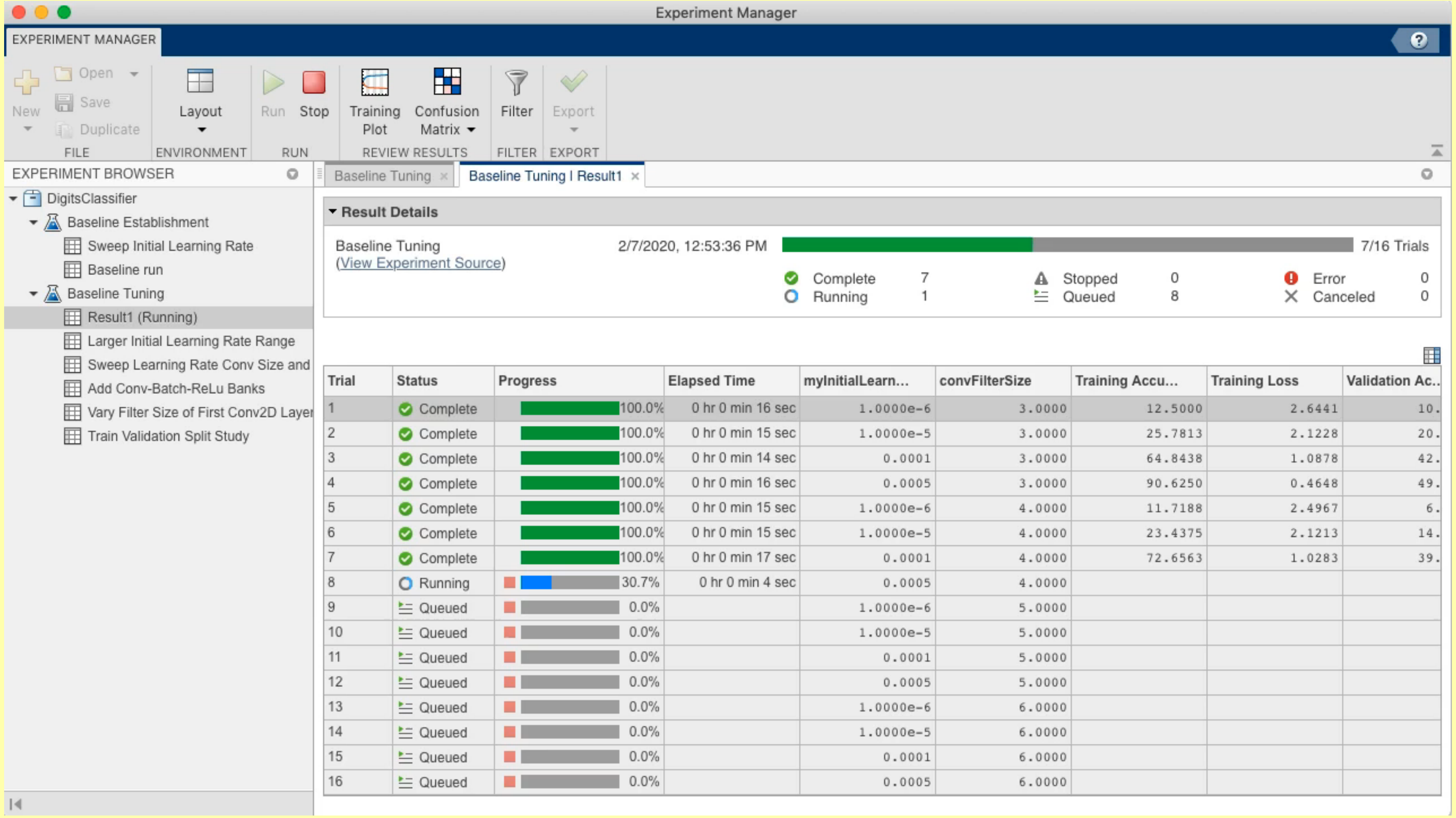
NET

TEST the trained network

`classify( )`

DEMO – Experiment Manager

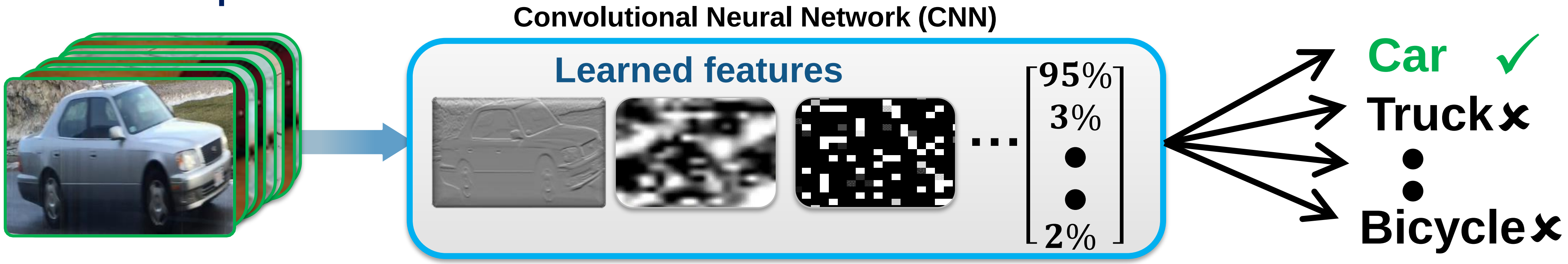
- Run, Track, and Analyze Multiple Deep Learning Experiments



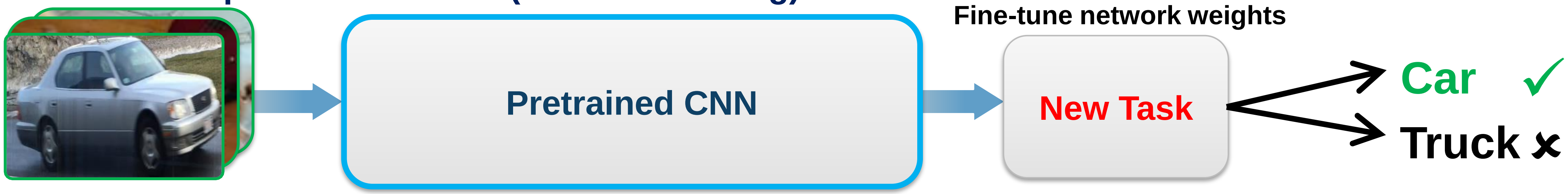
Play the video !

Techniques Covered so Far

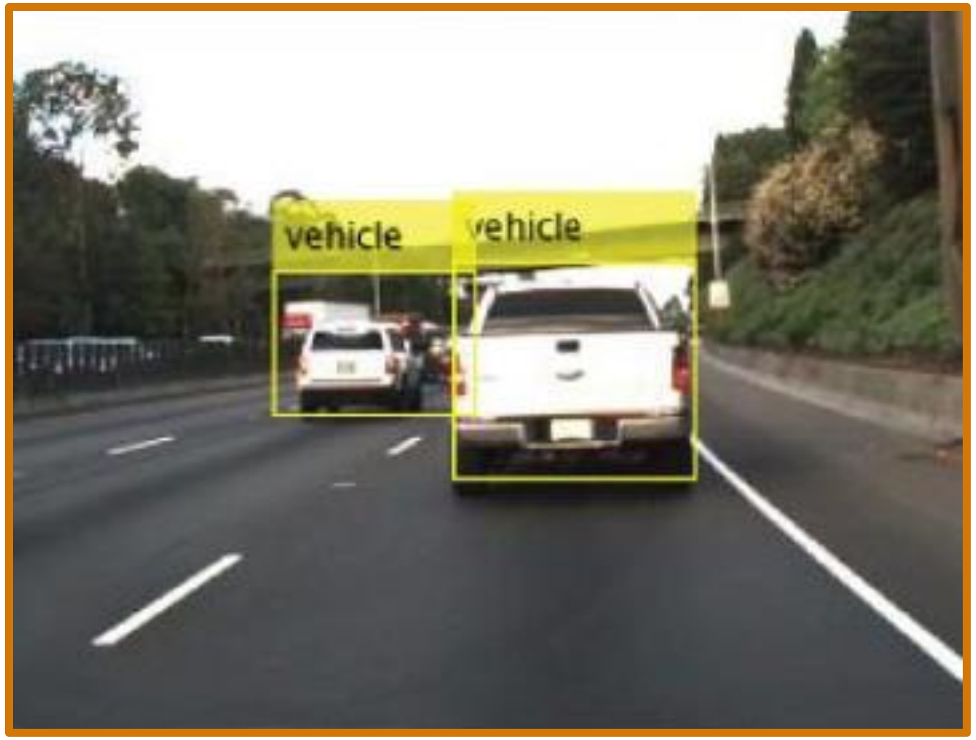
1. Train a Deep Neural Network from Scratch



2. Fine-tune a pretrained model (transfer learning)



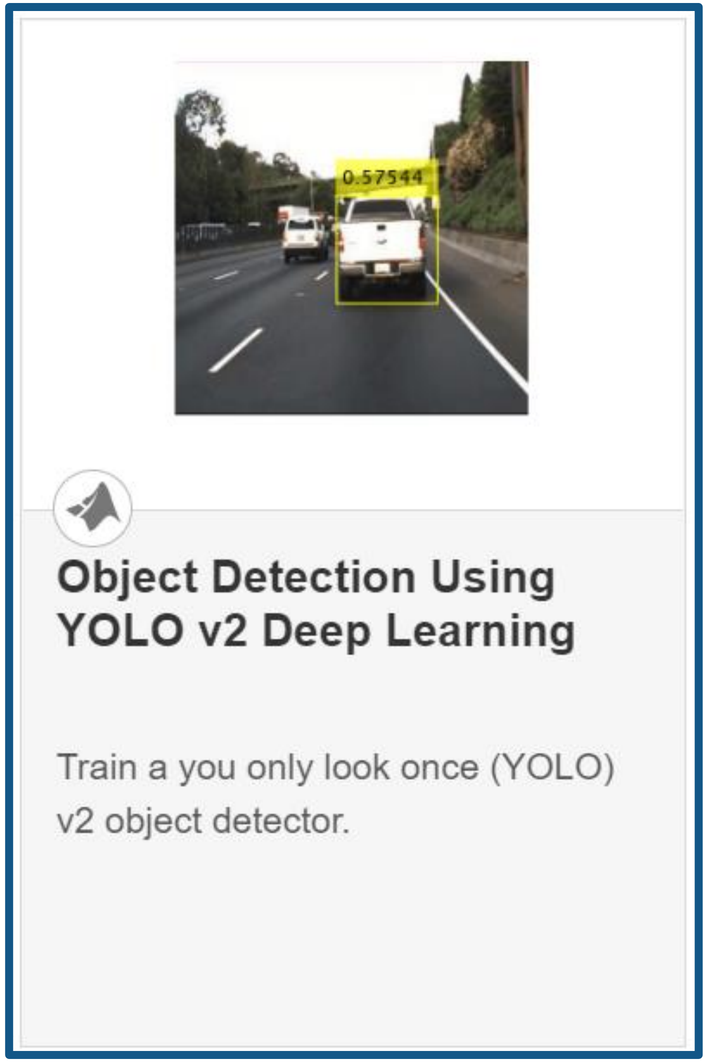
Classification vs. Object Detection



Classification predicts a label for an entire image.

Object detection predicts the location and label for objects in an image

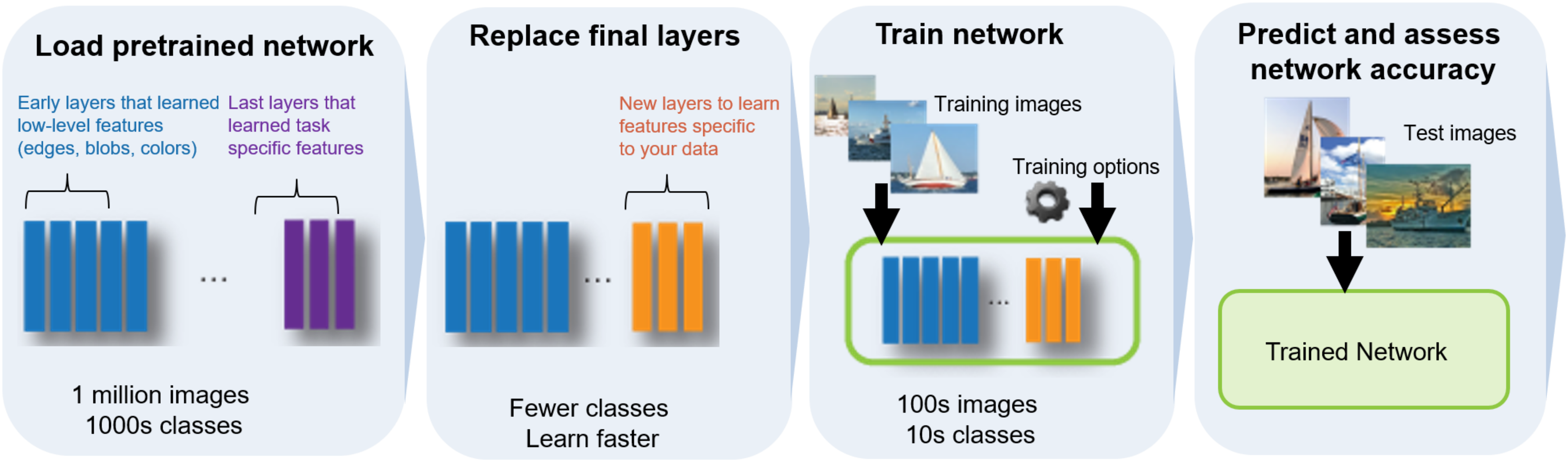
Deep Learning Object Detection Examples in MATLAB



Documentation examples:

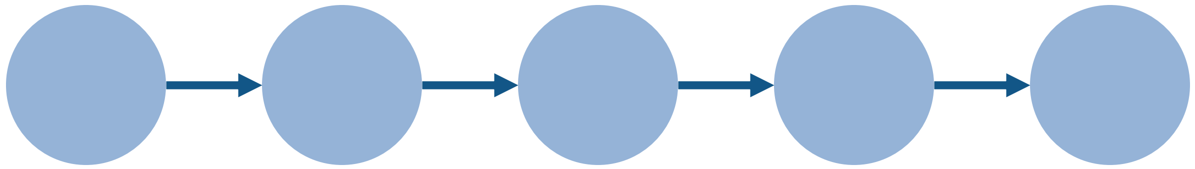
- [Faster R-CNN]
- [YOLO v2]
- [YOLO v3]
- [Single Shot Detector]

Transfer Learning is Commonly Used for Object Detection

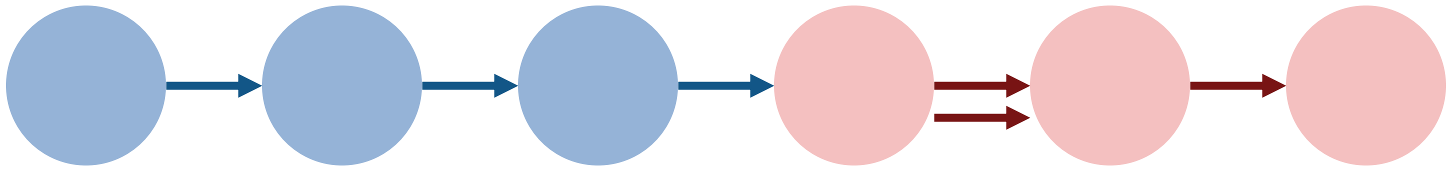


Transfer Learning Applied to YOLO v2 Object Detection

1. Import Pretrained model (ResNet-50)



2. Replace last layers with YOLO detection layers (yolov2Layers)



3. Train network with training data

Exercise 5 – Vehicles

Purpose:

- Use transfer learning to create YOLO v2 network
- Train network to detect vehicles in image

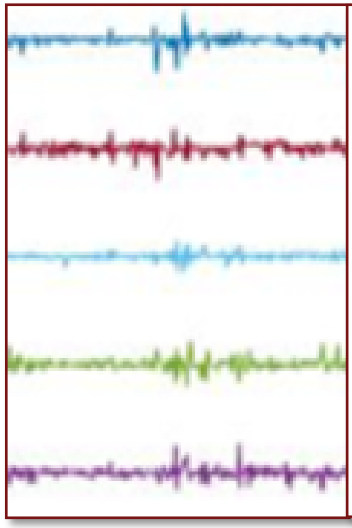
To Do:

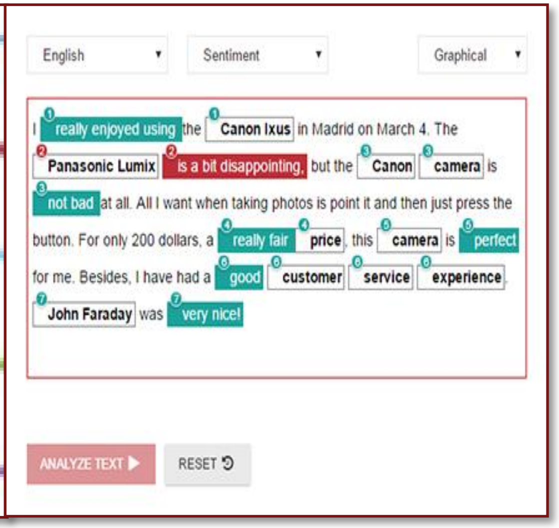
1. Open `Work_Vehicles.mlx`.

Selecting a Network Architecture

Image Data

Signal or Text Data





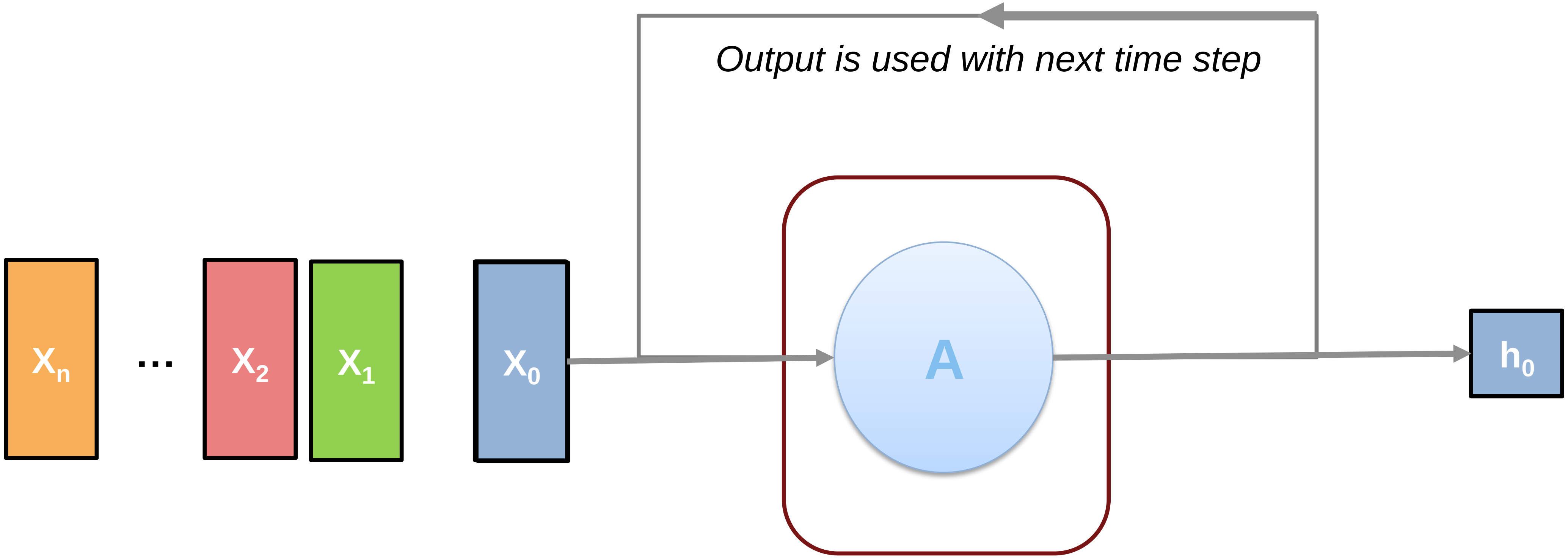
CNN

LSTM or CNN

LSTM = Long Short Term Series Network (more detail in later slides)

Recurrent Neural Networks

Take into account previous data when making new predictions



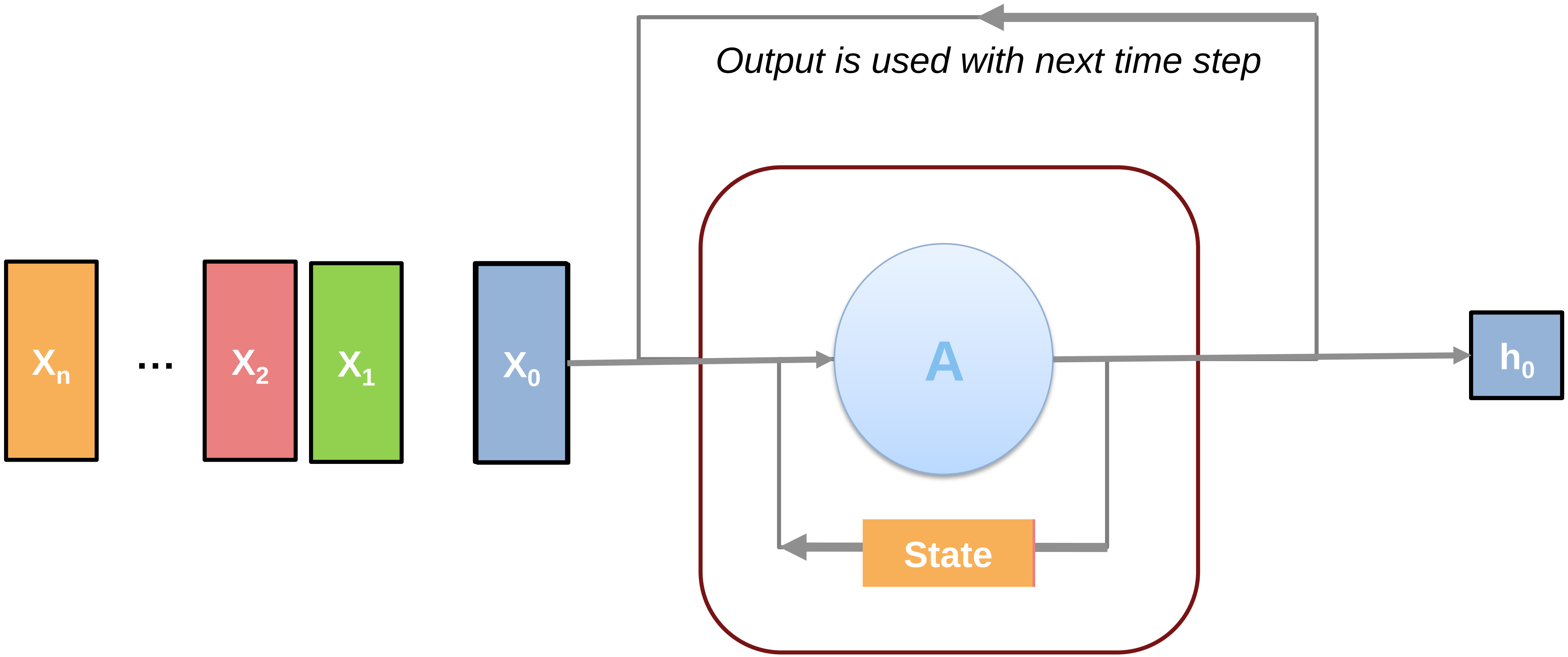
I was born in France…

[2000 words]

… I speak ___________ ?

Long Short-Term Memory Network

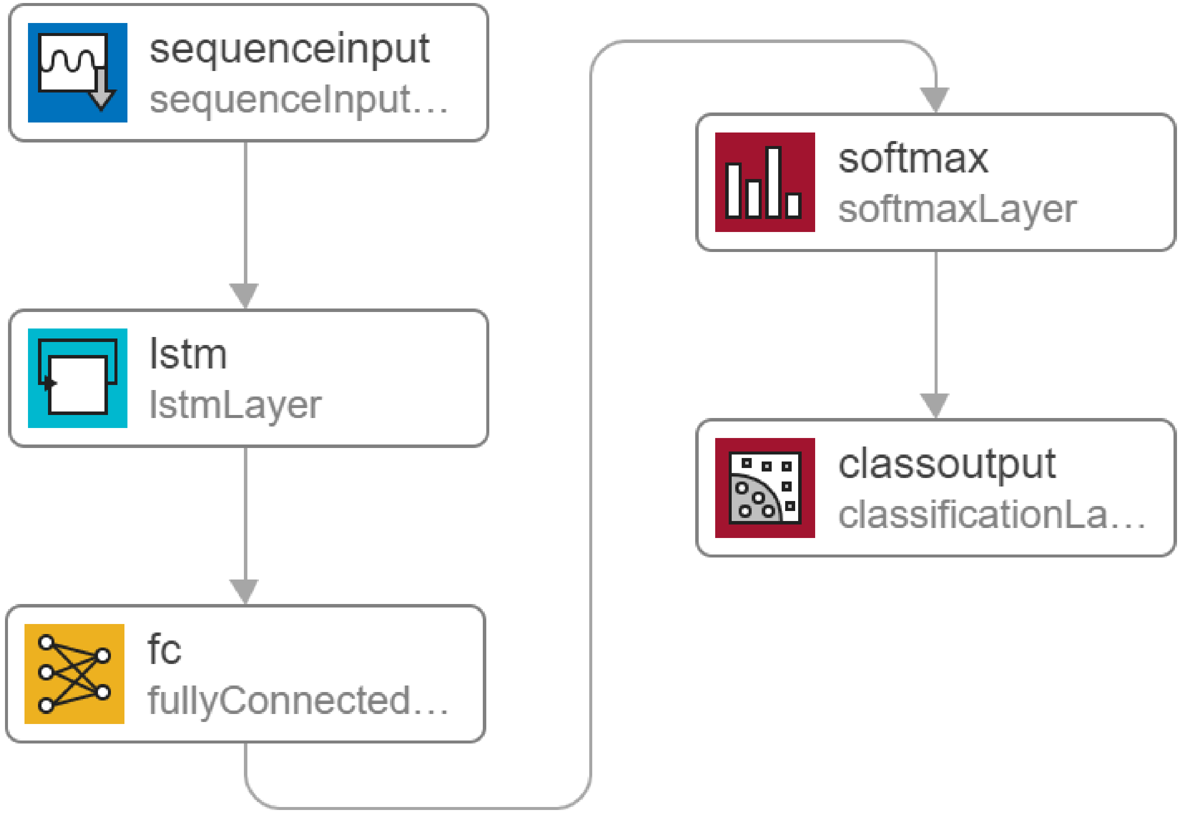
Recurrent Neural Network that carries a memory cell (state) throughout the process



Simple LSTM Network Architecture

- Layers:
  - Input
  - LSTM
  - Fully Connected
  - Softmax
  - Classification

- LSTMs can be used for classification or regression



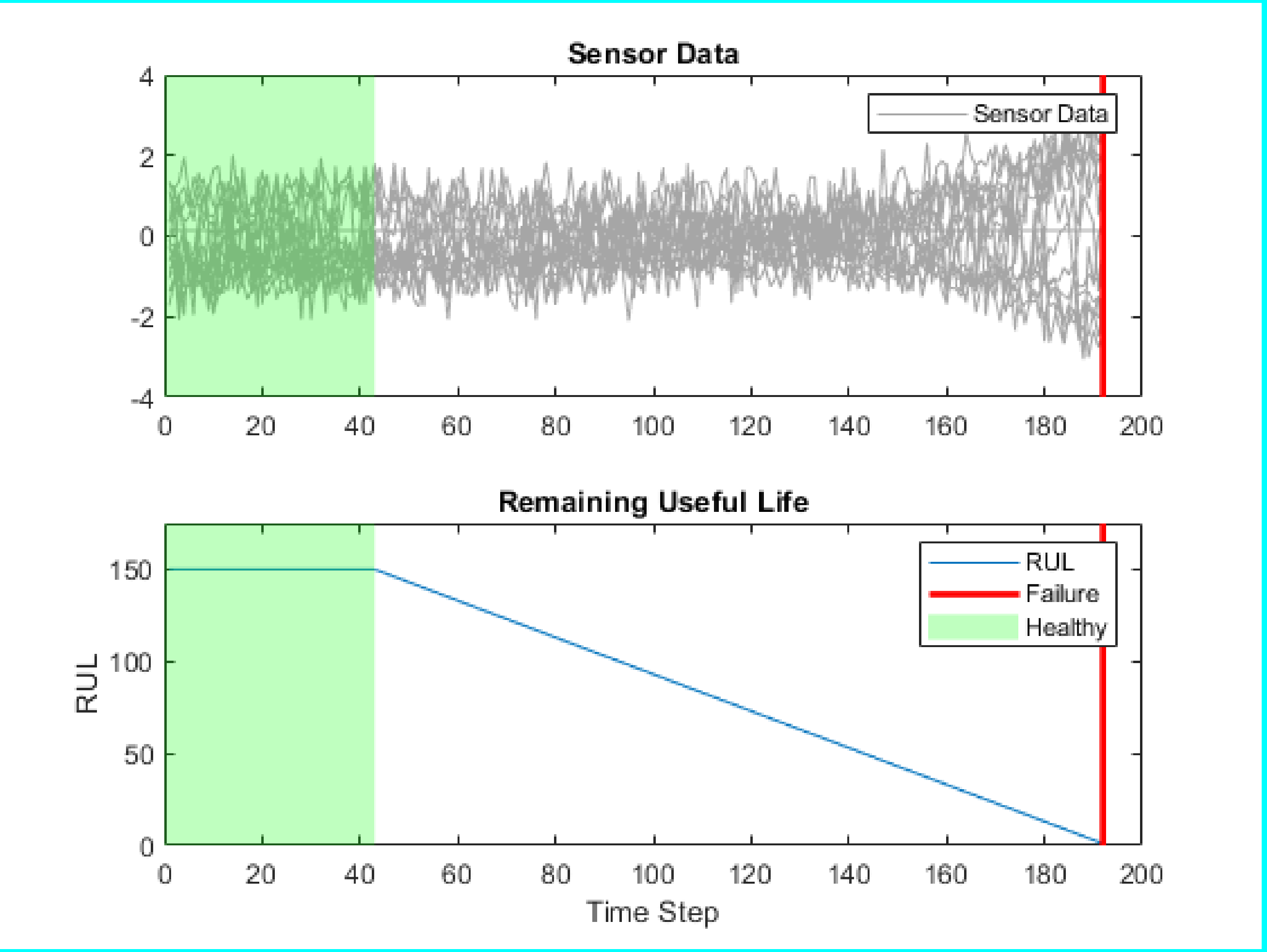
Exercise 6 – Engines

Purpose:

- Use an LSTM network to predict the remaining useful life of engines based on sensor data (regression)

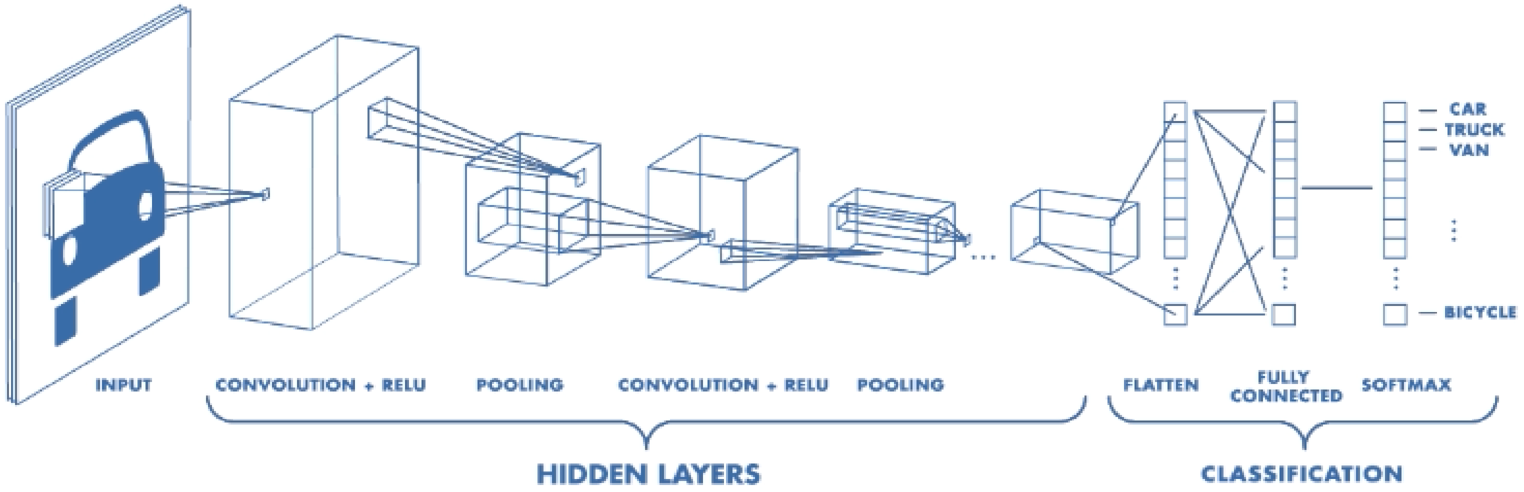
To Do:

1. Open Work_Engines.mlx.

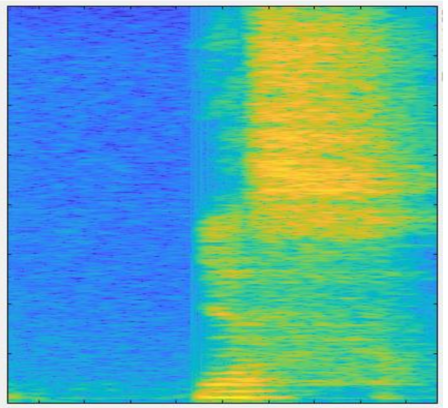


Using CNNs on Signal Data (Time-Freq Transforms)

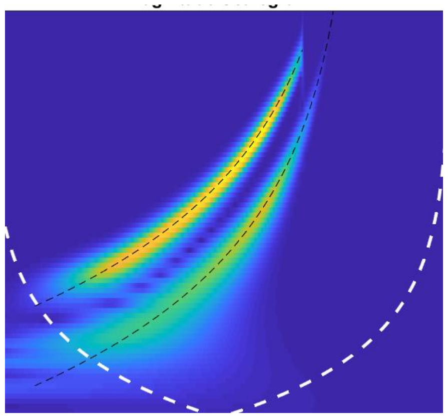
- CNNs are typically used to classify images
- Time-Frequency representations of signals can be used as images
- This approach can serve as a good starting point for signal classifications



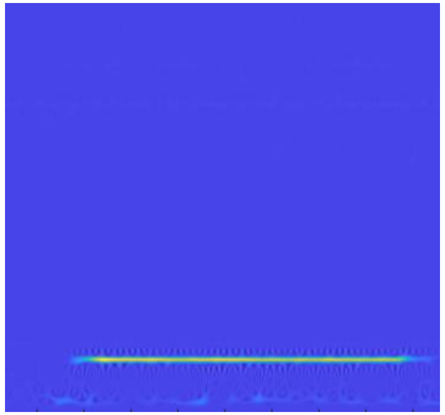
Different Types of Time-Frequency Transforms



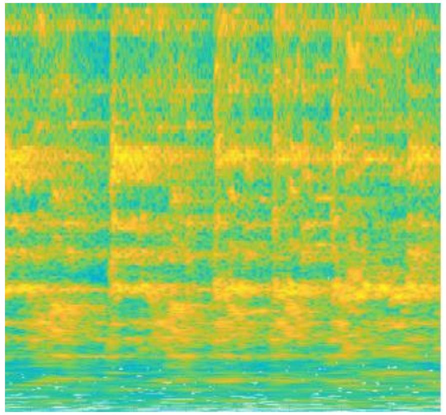
Basic spectrogram



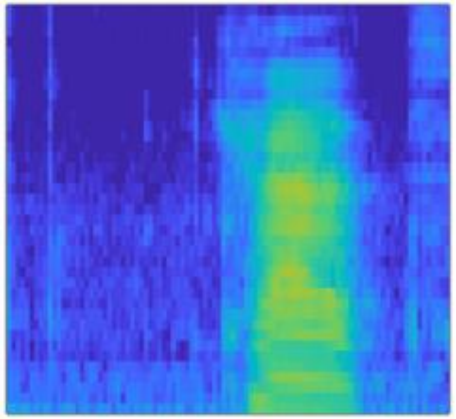
Wavelet scalogram



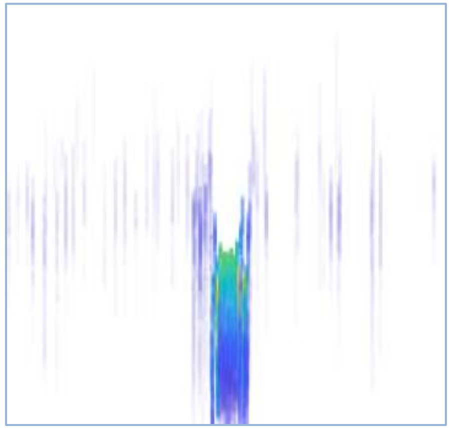
Wigner-Ville Transform



Hilbert-Huang Transform



Constant Q transform

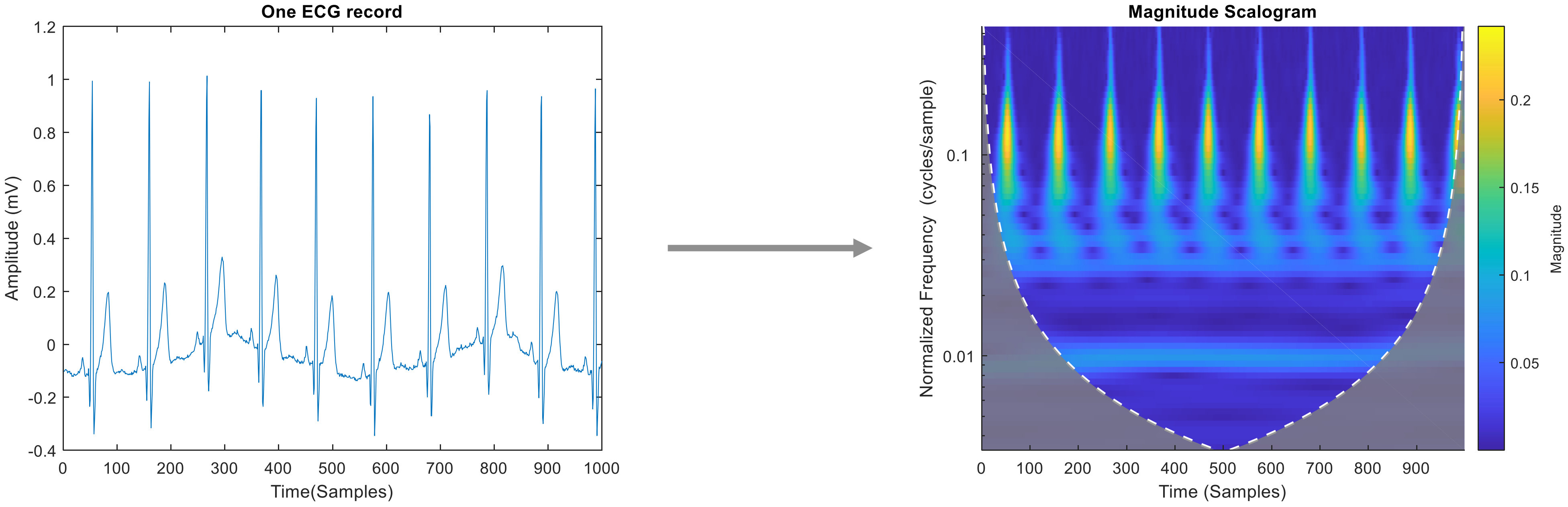


Perceptually-spaced Spectrogram

[MATLAB Time –Frequency Gallery]

Continuous Wavelet Transform

- We will use this time-frequency transform in our exercise (Work_ECG_1).
- Differentiates signals from different classes well compared to basic spectrogram.

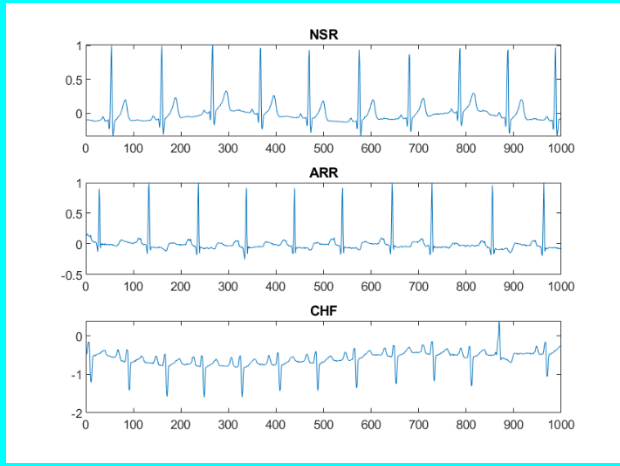


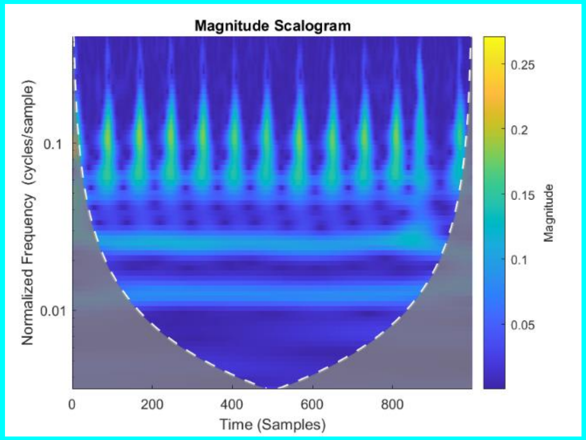
Exercise 7 – ECG

Purpose:

- Part 1: Classify different types of ECG signals using time-frequency transform + CNNs

- Part 2: Classify these same signals using feature extraction + LSTM





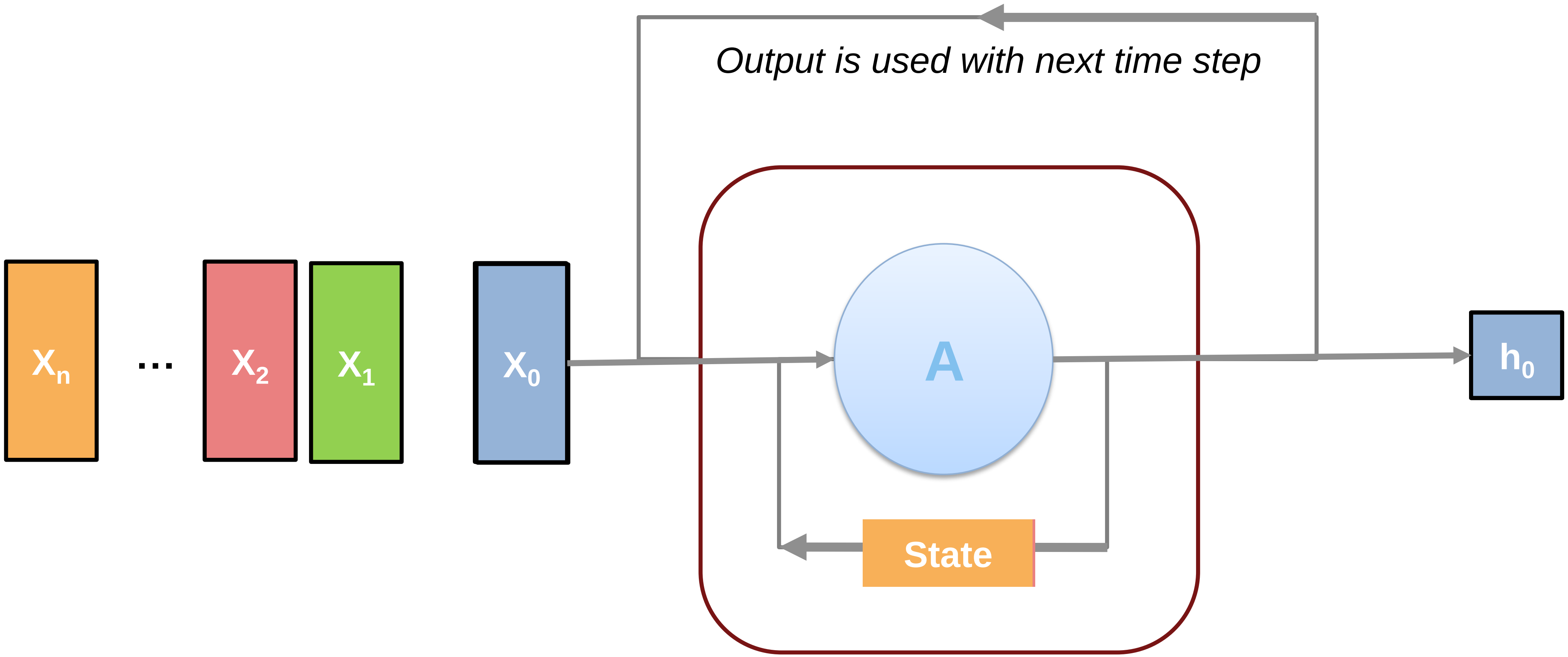
To Do:

1. Open Work_ECG_1.mlx.
2. Open Work_ECG_2.mlx

Part 1 is required for part 2 to run

Long Short-Term Memory Network

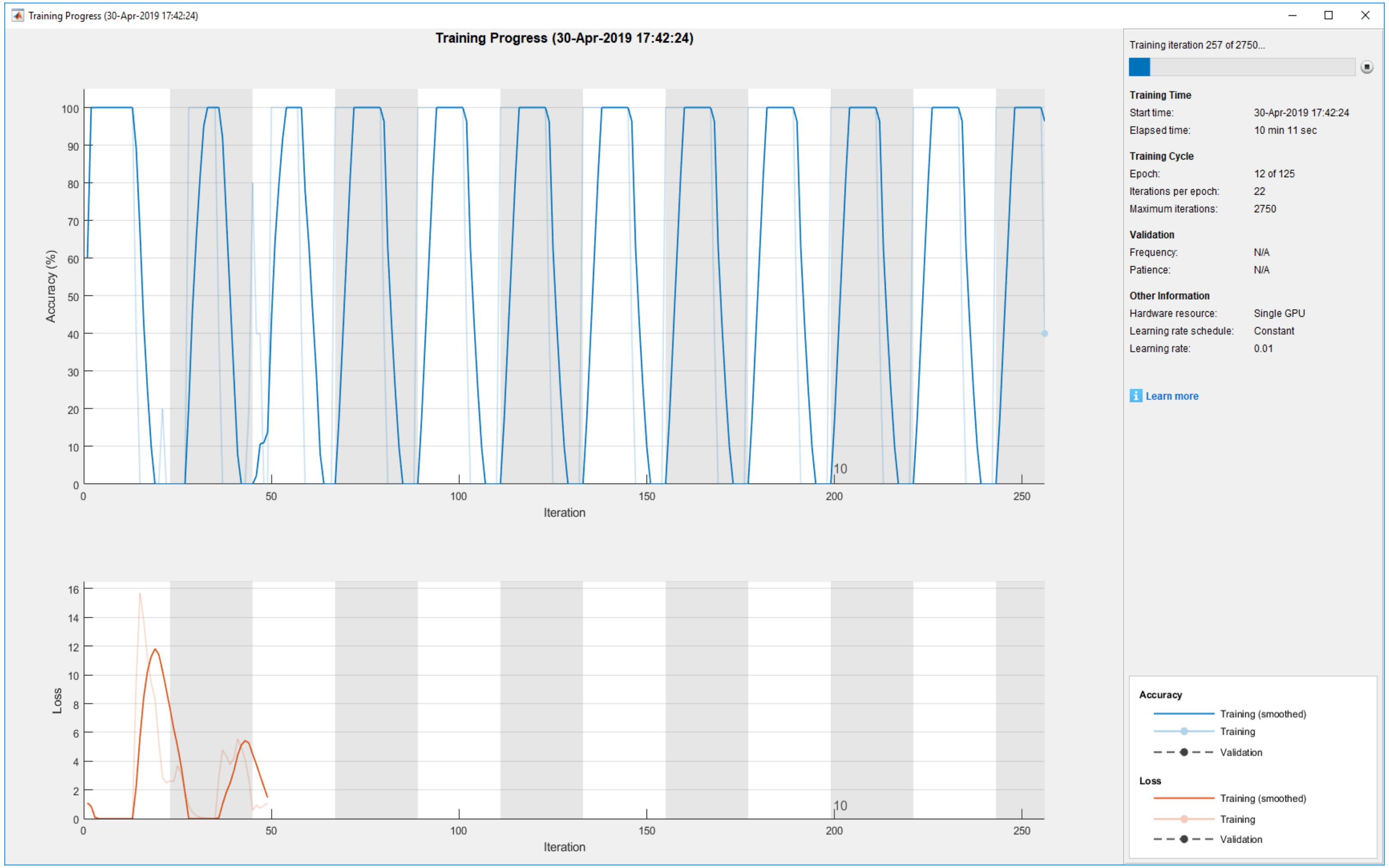
Recurrent Neural Network that carries a memory cell (state) throughout the process



Training LSTMs Directly on Signals?

<u>Common Problem</u>

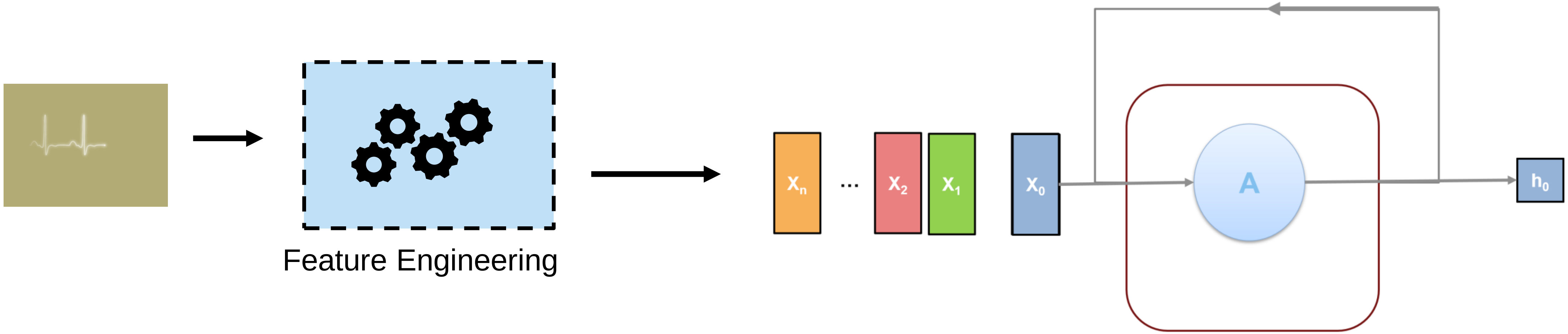
- Long signals subject to signal drift
- Training directly leads to poor network accuracy



Solution: Feature Extraction

- Extracting features from signals will:
  - Preserve important information from signal
  - Have smaller length compared to original signal

- Use extracted feature vectors as input to LSTM



Feature Engineering

Automatic Feature Extraction with Wavelet Scattering

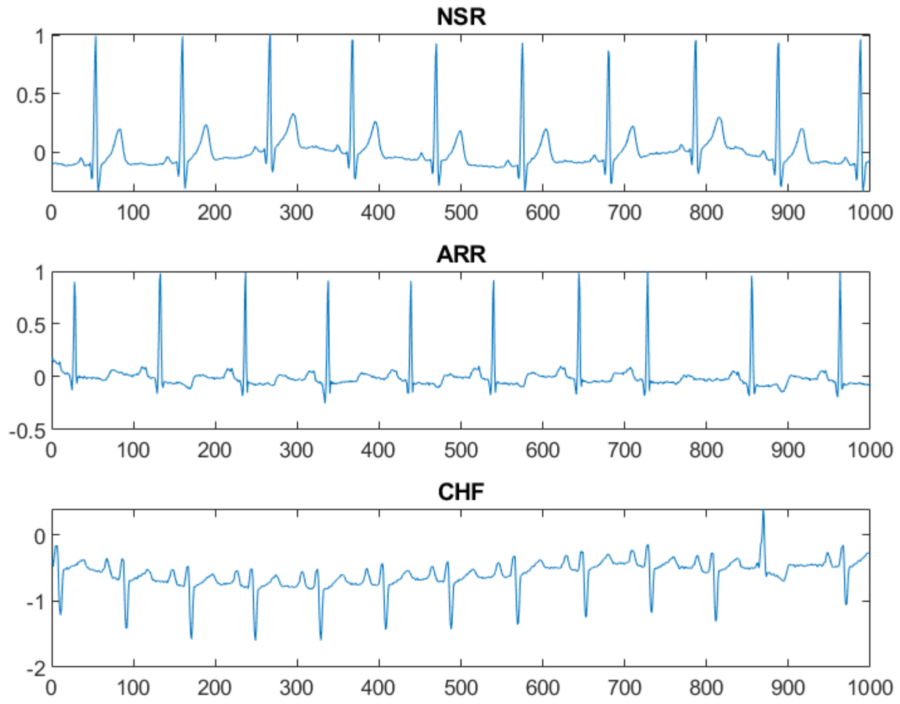
- We will use a technique called wavelet scattering to extract features from our signal

Original Signals

- 65,000+ samples long

Feature extraction

- Dimensions: 499x8
- 499 features with
  8 different channels



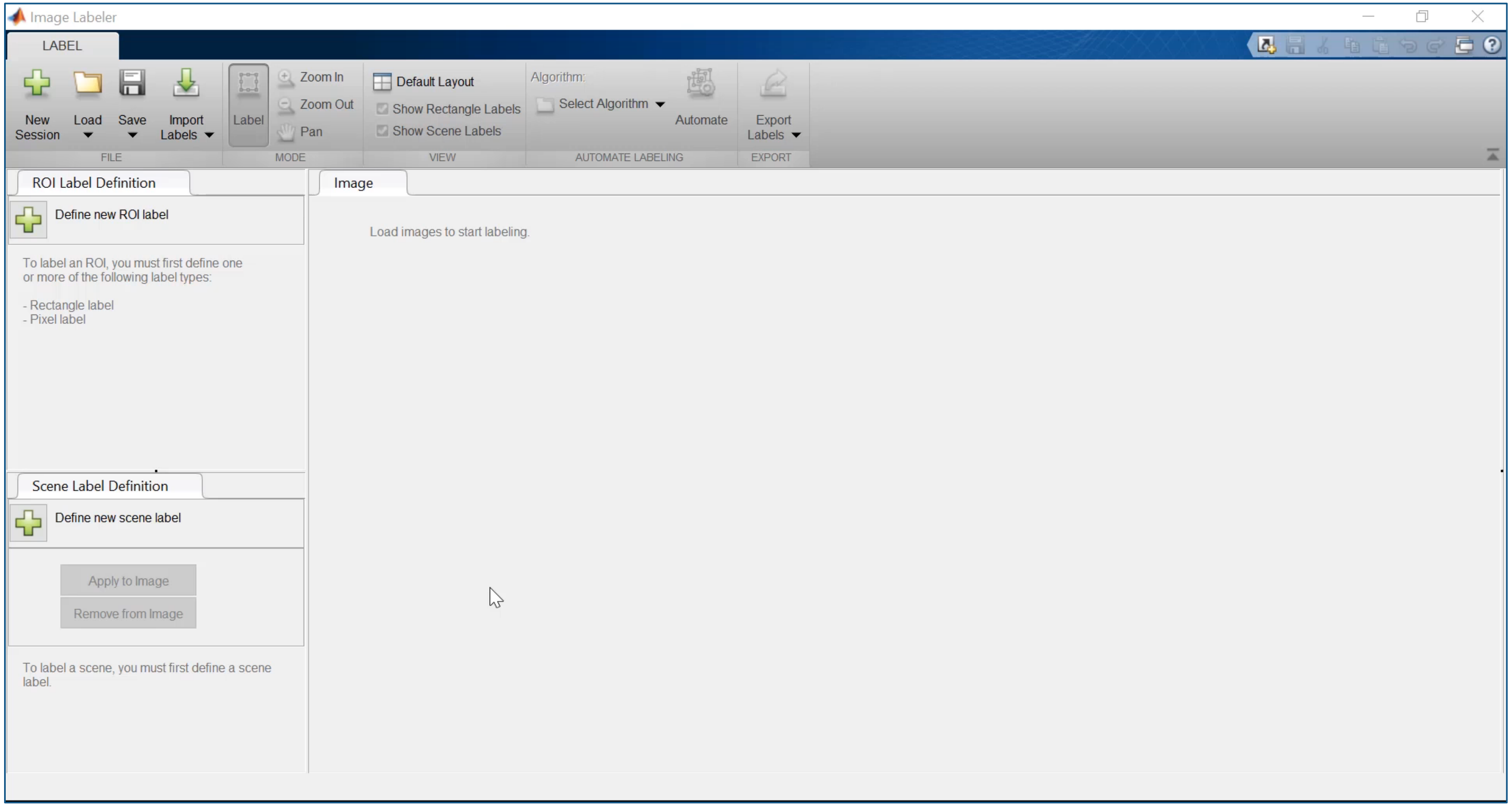
Deep Learning Workflow – Prepare Data

Data Preparation	AI Modeling	Simulation & Test	Deployment
Data cleansing and preparation	Model design and tuning	Integration with complex systems	Embedded devices
Human insight	Hardware accelerated training	System simulation	Enterprise systems
Simulation-generated data	Interoperability	System verification and validation	Edge, cloud, desktop

How do I label my data?

[Image Labeler
+ Video labeler]()

Signal Labeler
+ Audio Labeler

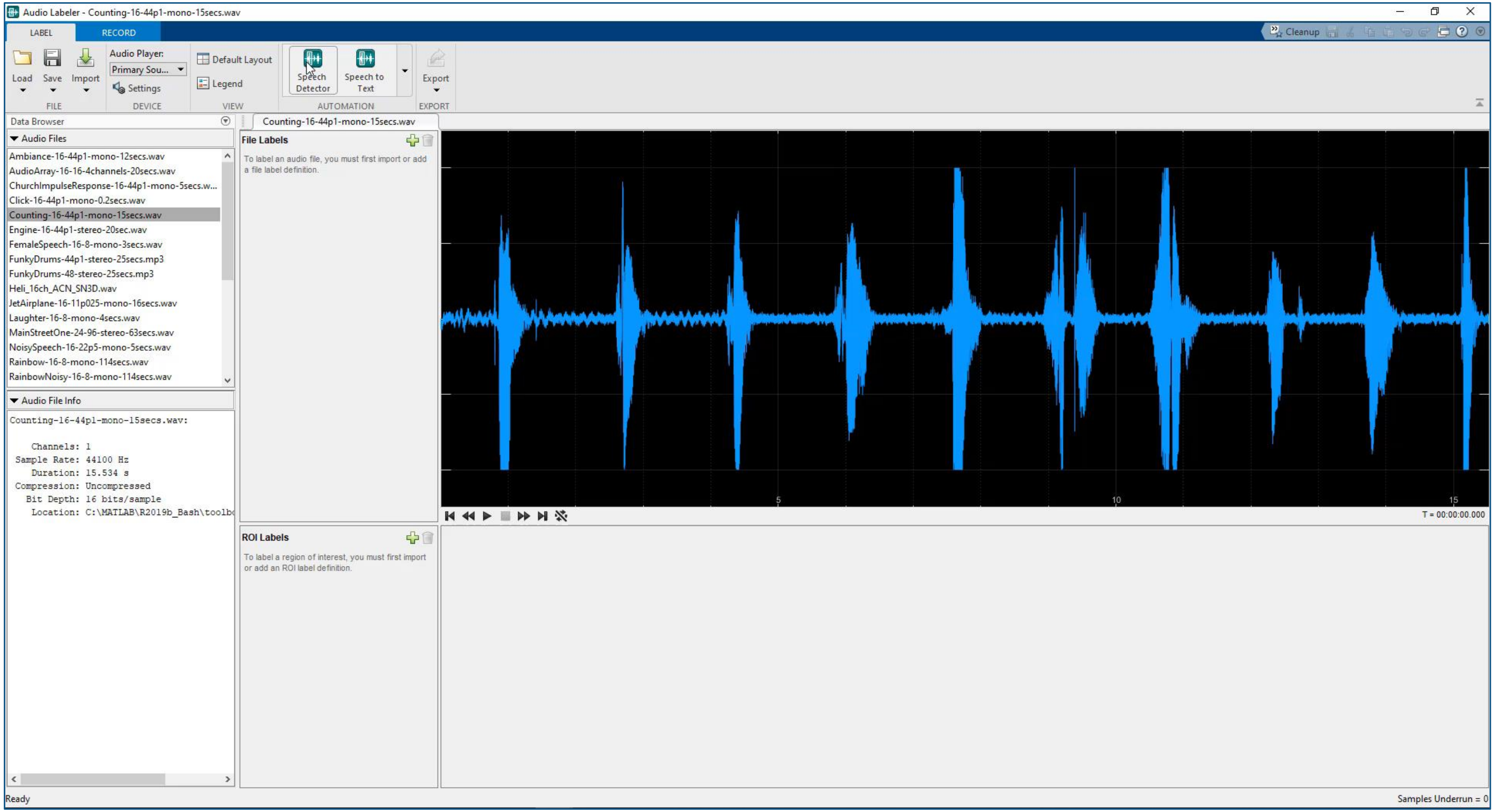


Play the video !

How do I label my data?

Image Labeler
+ Video labeler

[Signal Labeler
+ Audio Labeler]()



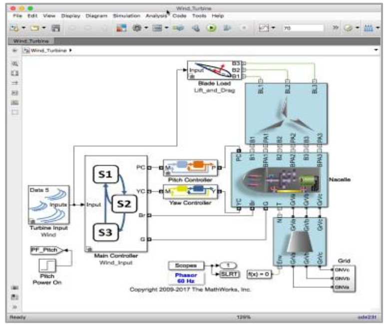
Play the video !

Deep Learning Workflow – Deploy System

Data Preparation	AI Modeling	Simulation & Test	Deployment
Data cleansing and preparation	Model design and tuning	Integration with complex systems	Embedded devices
Human insight	Hardware accelerated training	System simulation	Enterprise systems
Simulation-generated data	Interoperability	System verification and validation	Edge, cloud, desktop

Deployment and Scaling for A.I.

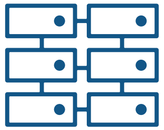
MATLAB



Embedded Systems

CPU GPU FPGA

Enterprise Systems



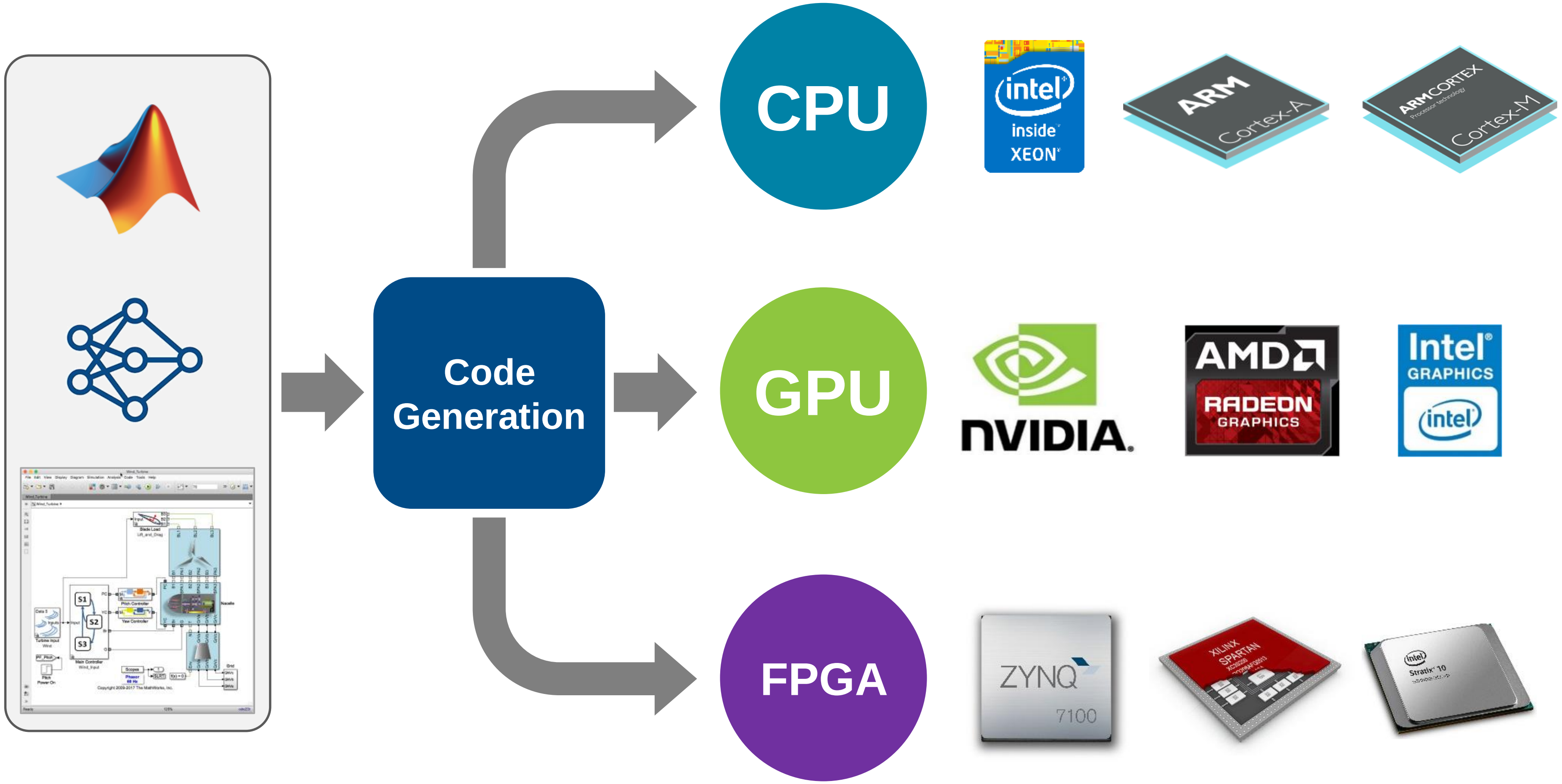
Embedded Deployment – Automatic Code Generation

MATLAB Code

Auto-generated Code (C/C++/CUDA)

Deployment Target

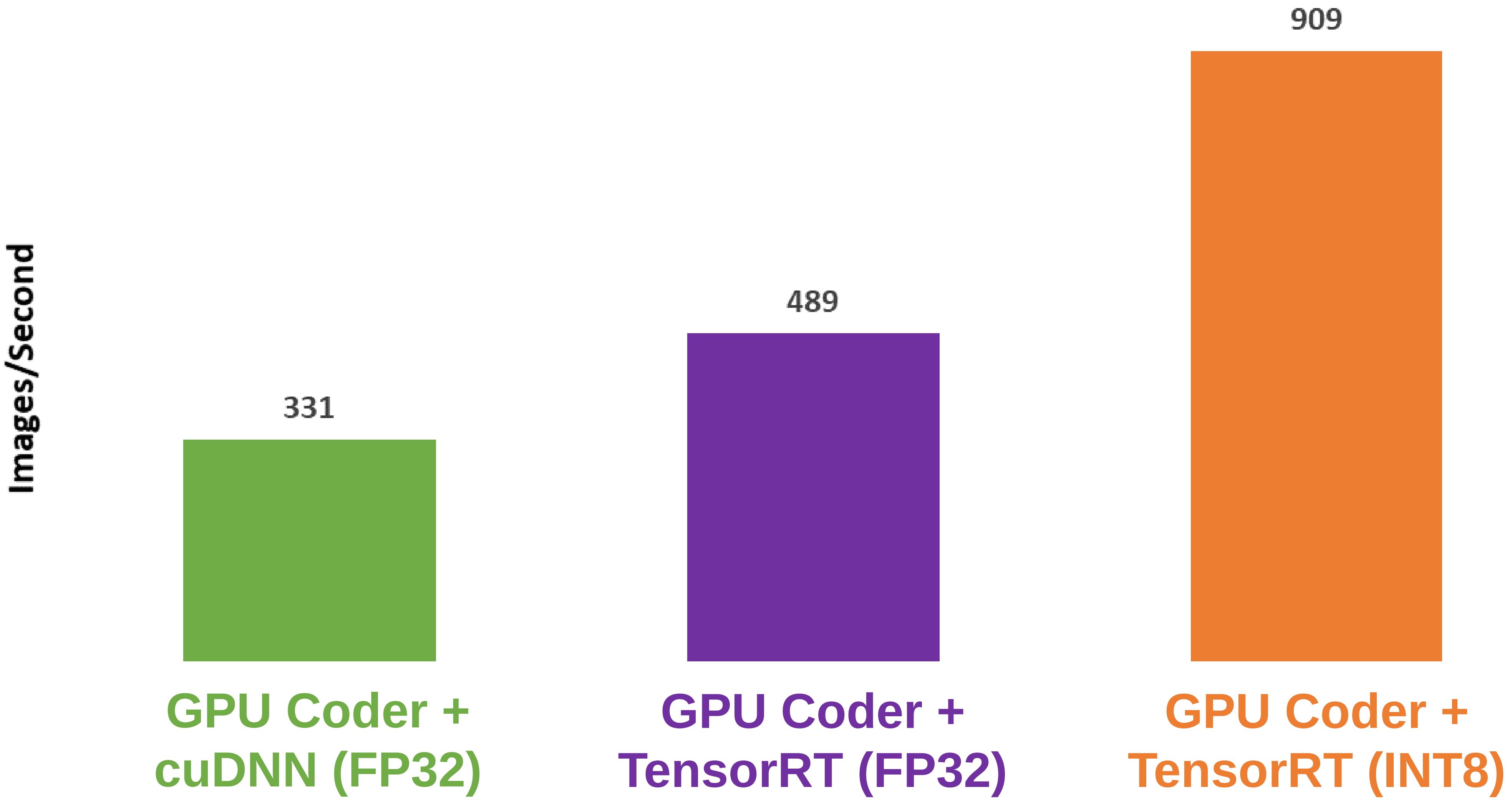
Deploying Models for Inference



NVIDIA, the NVIDIA logo, and TensorRT are registered trademarks of NVIDIA Corporation
Intel logo is a registrered trademark of Intel Coroporation

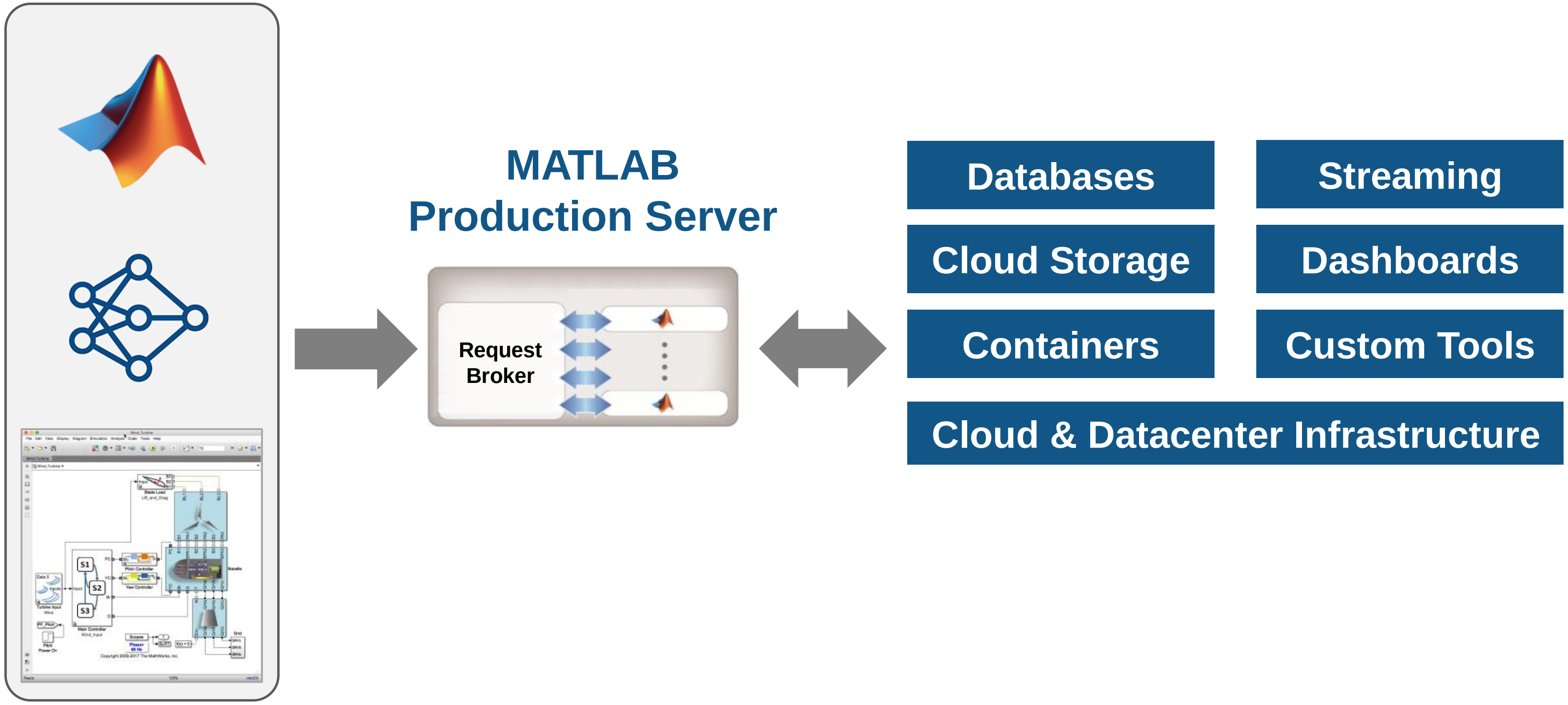
GPU Coder Inference Performance with ResNet-50 on Titan V

Batch 1

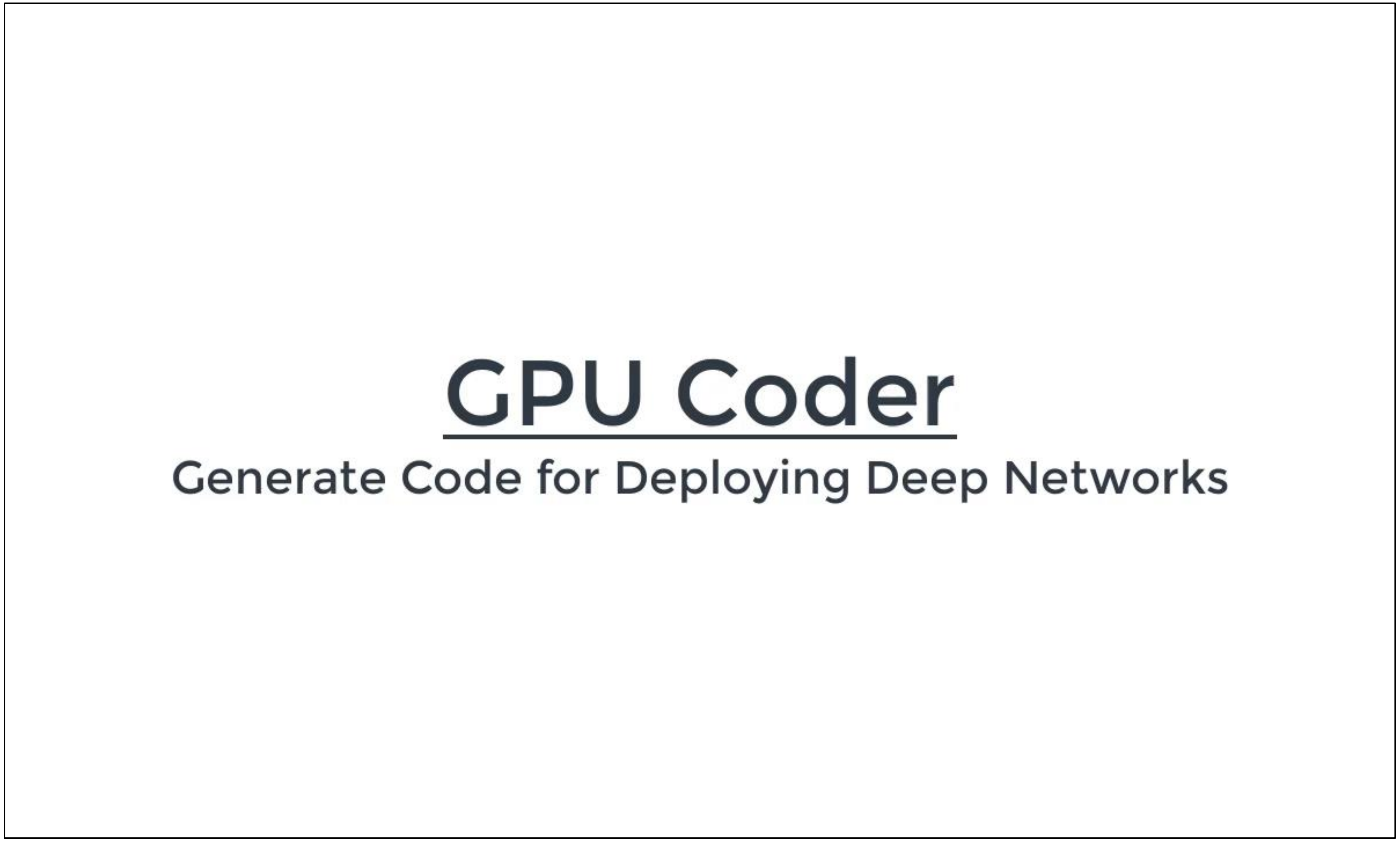


Intel® Xeon® CPU 3.6 GHz - Titan V - NVIDIA libraries: CUDA10.0/1 - cuDNN 7.5.0

Deploy to Enterprise IT Infrastructure

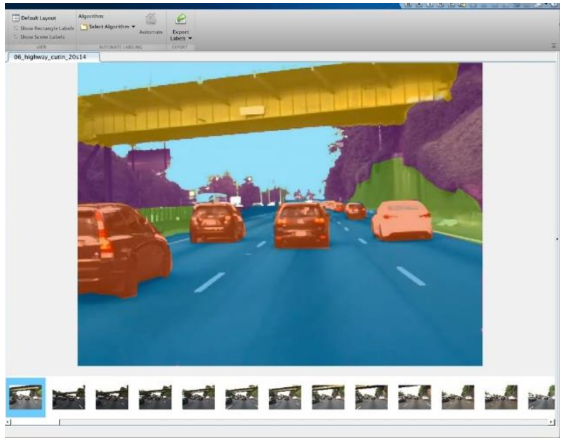


Generate GPU Code for Deep Networks



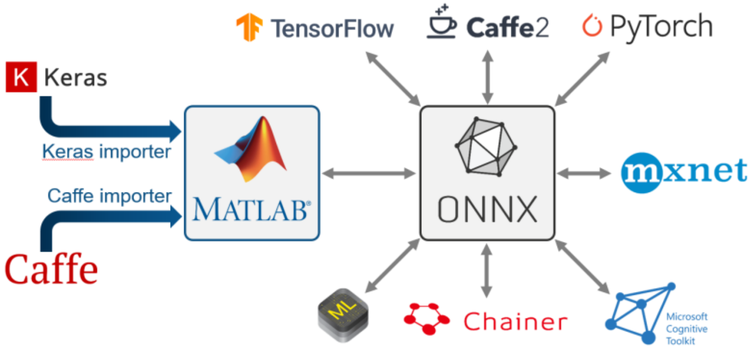
Play the video !

Why Use MATLAB?



MATLAB supports the **data preparation, training, and deployment** workflow

MATLAB has specialized DL tools designed for **scientists and engineers**



MATLAB **interoperates and enhances** Open Source frameworks

MathWorks Engineering Support

Training

Consulting

Onsite Workshops and Seminars

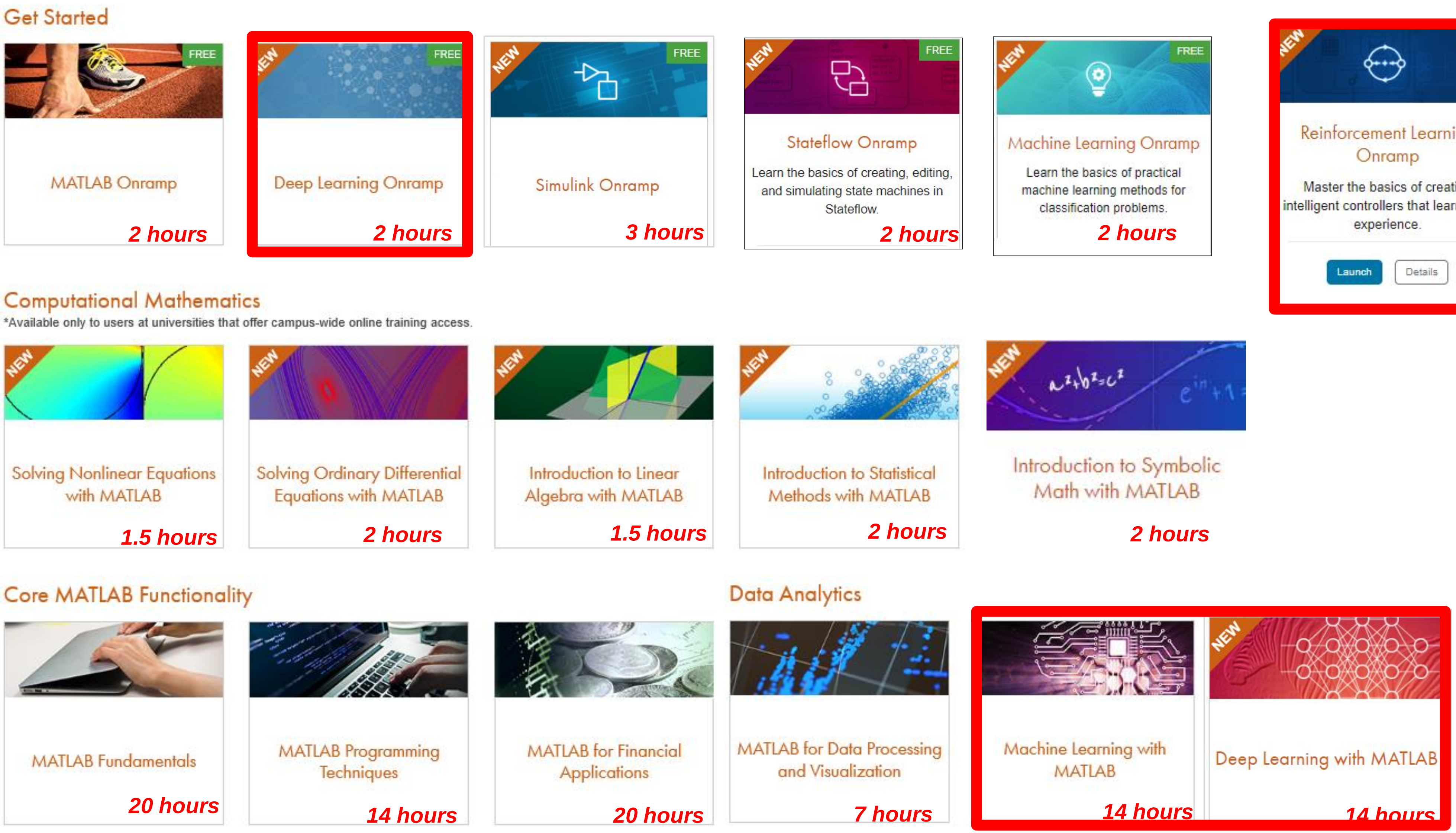
Guided Evaluations

Technical Support

Self-Paced Online Courses

https://matlabacademy.mathworks.com/

Every MATLAB Campus License in AU/NZ has access to these courses, eg: **ANU**, UNSW, Usyd, UTS, UQ, QUT, etc



6 in-depth courses for enhancing MATLAB skills