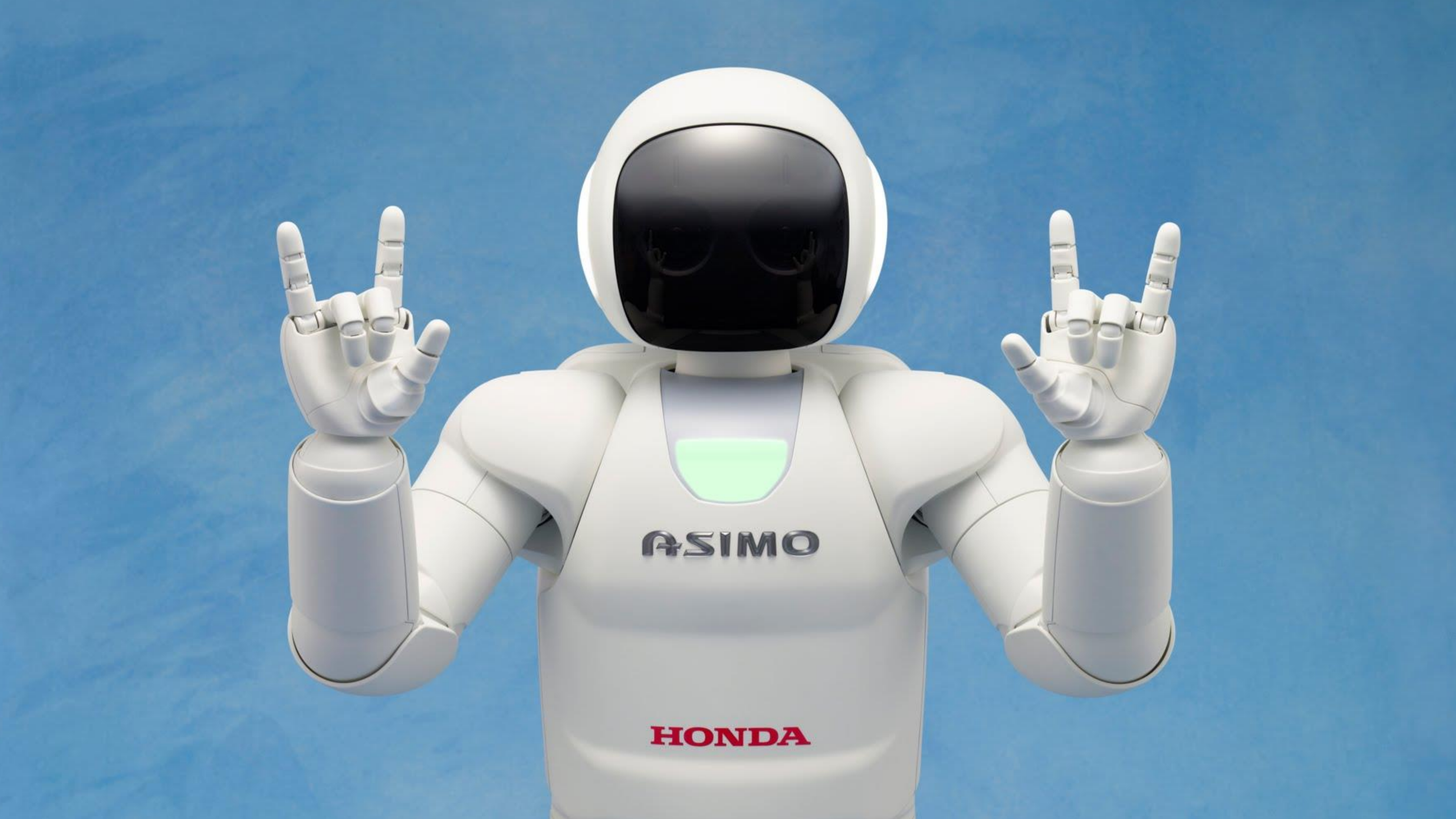
Deep Learning: Development to Deployment

Spandhana Gonuguntla, PhD
Education Technical Evangelist
sgonugun@mathworks.com

Indian Institute of Science
27th Feb 2020

Administrivia

- If you have laptops running MATLAB 2019b, please ensure you collect the files necessary for this workshop
- Download support package for AlexNet if you haven't already done so
  - >> net = alexnet (shouldn't throw an error)
- Fill up this survey : https://tinyurl.com/w5xcfht


ASIMO
HONDA


amazon

Artificial Intelligence

The capability of a machine to imitate intelligent human behavior

Artificial Intelligence

The capability of a machine to ***match or exceed*** *intelligent human behavior*

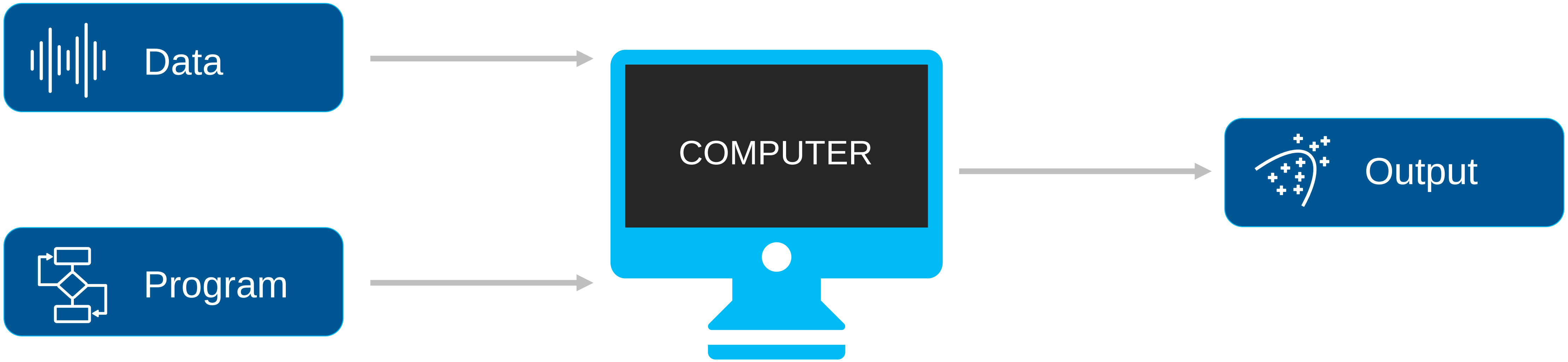
Artificial Intelligence Today

The capability of a machine to ***match or exceed*** *intelligent human behavior*

by training a machine to learn the desired behavior

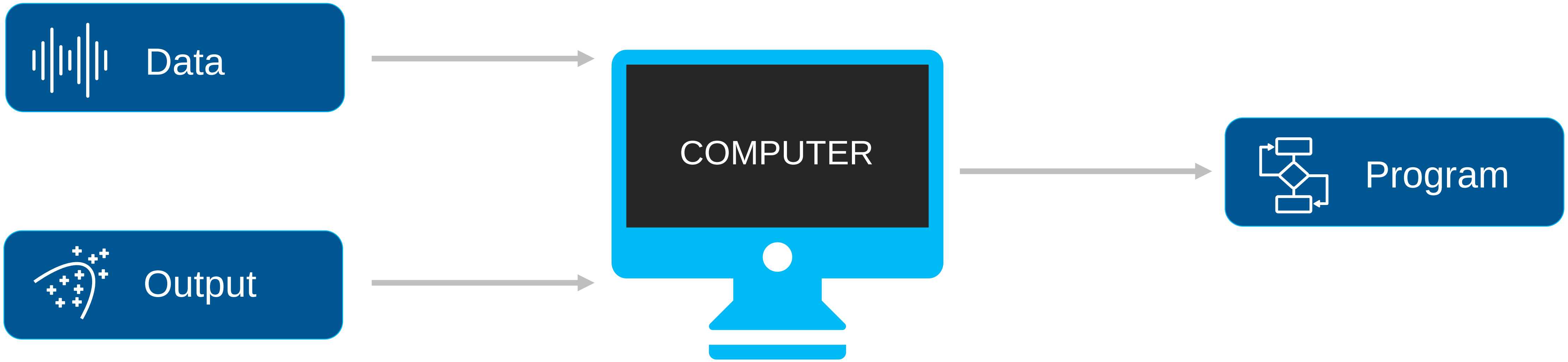
There are two ways to get a computer to do what you want

Traditional Programming



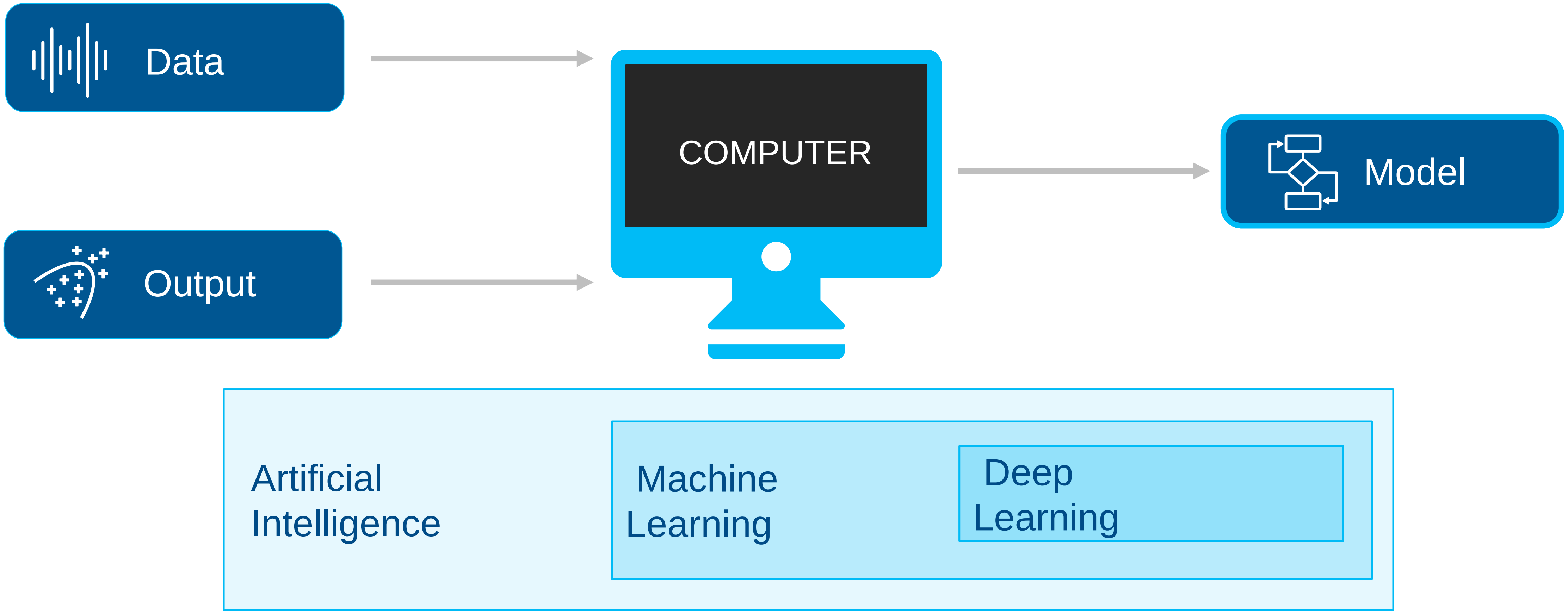
There are two ways to get a computer to do what you want

Machine Learning



There are two ways to get a computer to do what you want

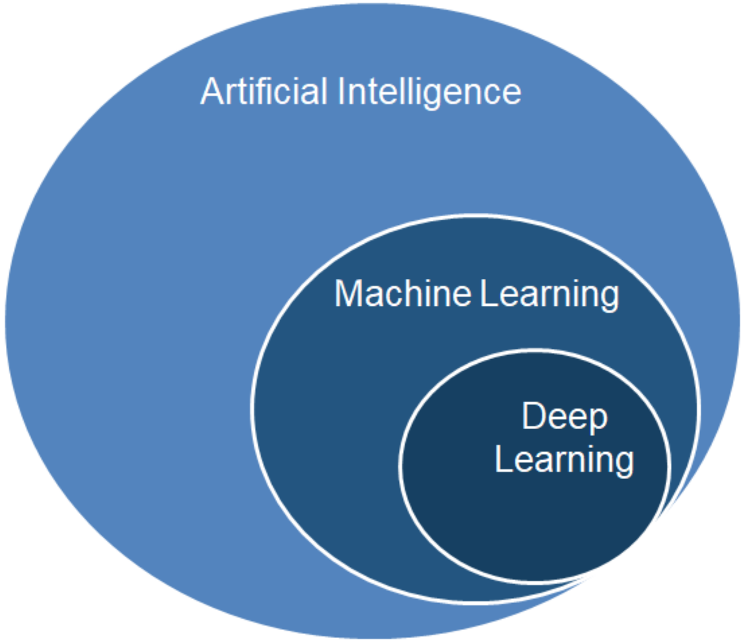
Machine Learning

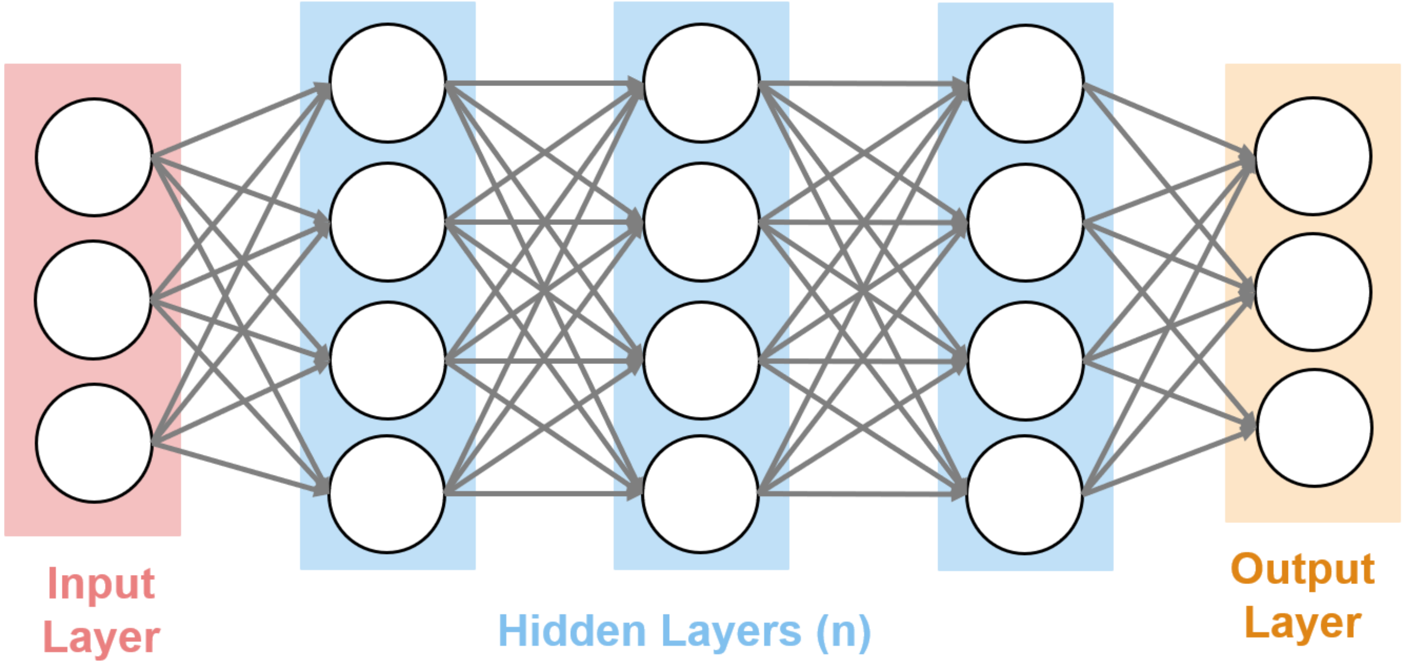


What is Deep Learning really?

Image from: xkcd.com

Deep Learning is a subset of machine learning that uses neural networks to extract features from data



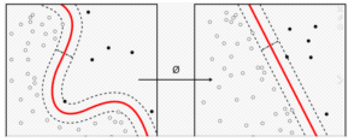


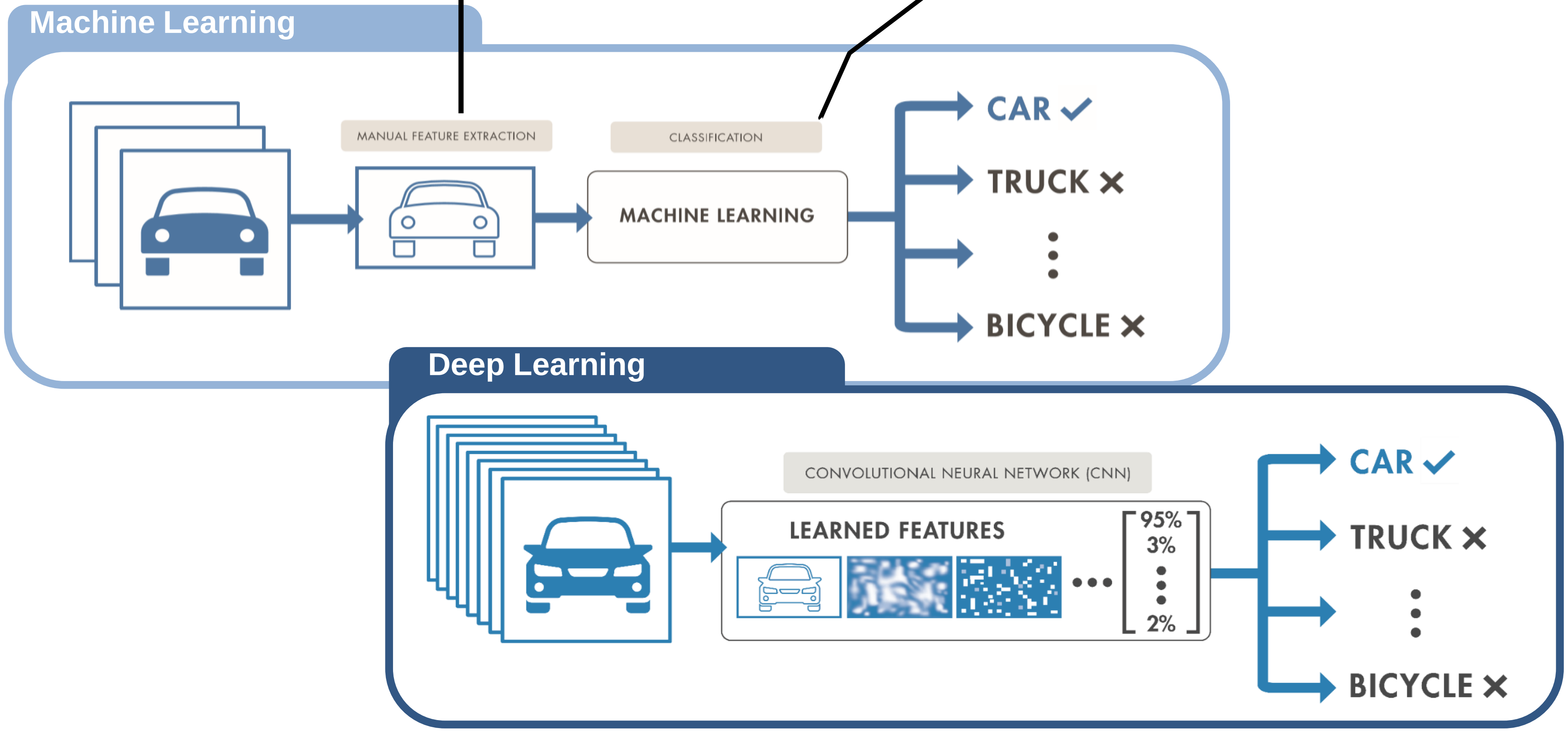
Machine Learning vs Deep Learning

Deep learning performs <u>end-to-end learning</u> by <u>learning features, representations and tasks</u> directly from **images, time-series, and text data**

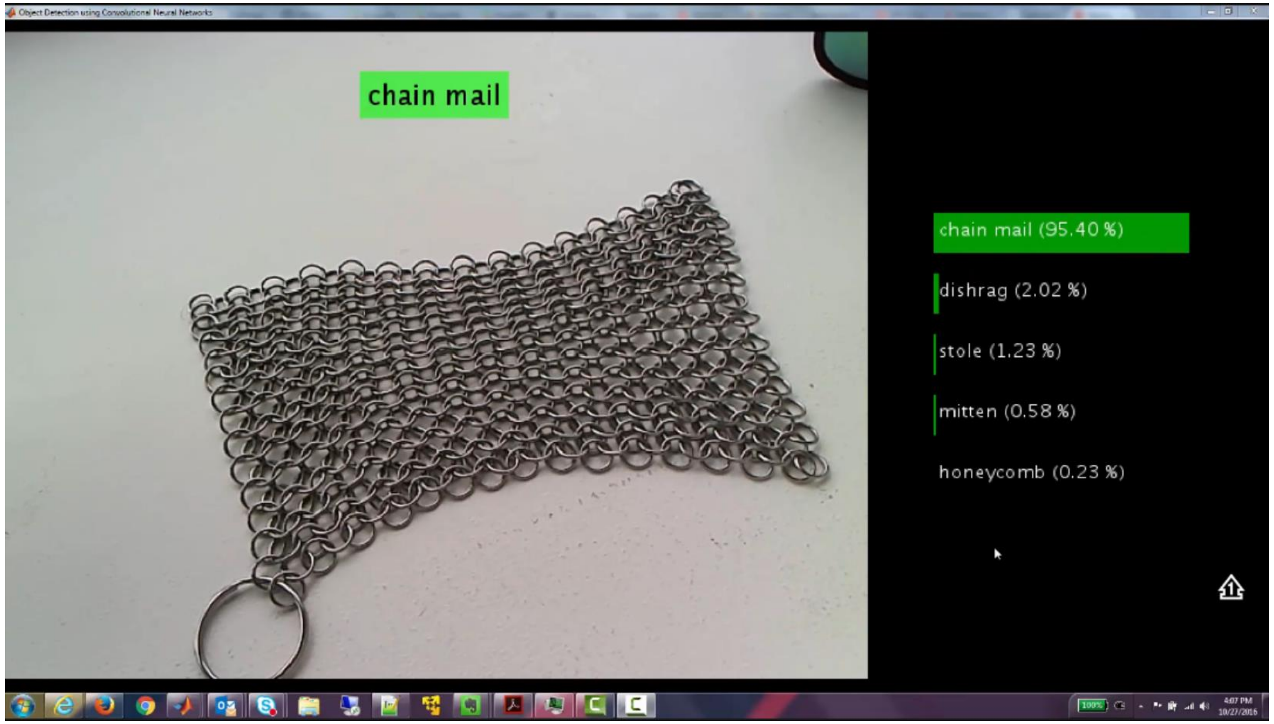
We specify the nature of the features we want to extract…

…and the type of model we want to build.



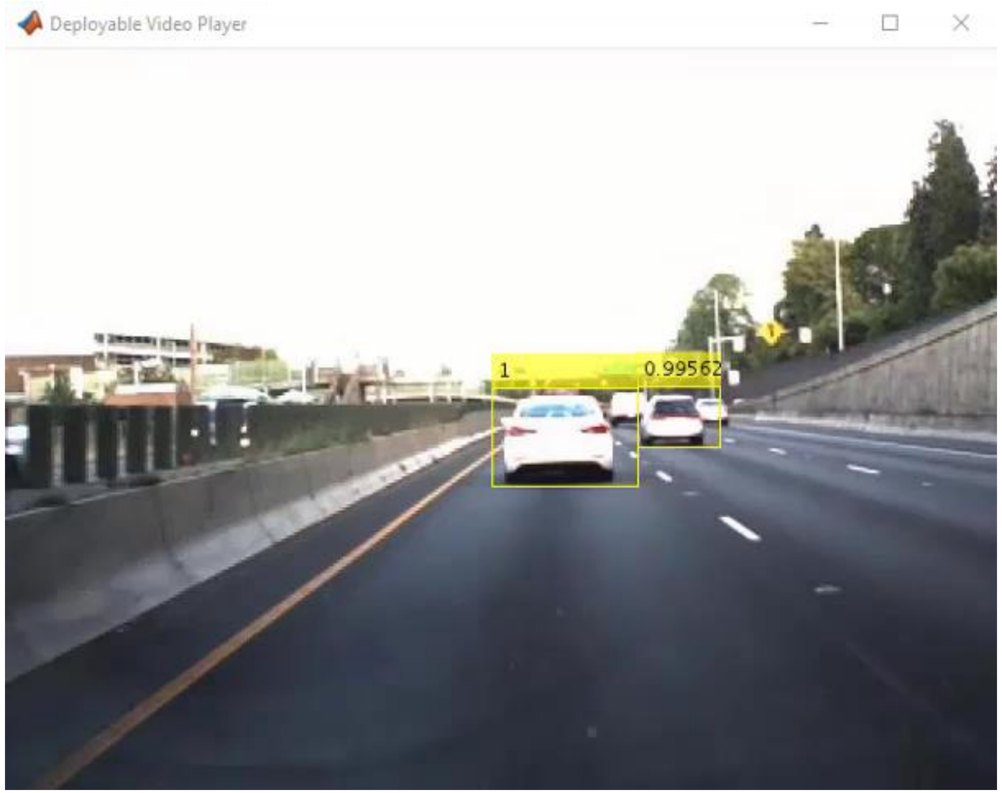


Object recognition using deep learning



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

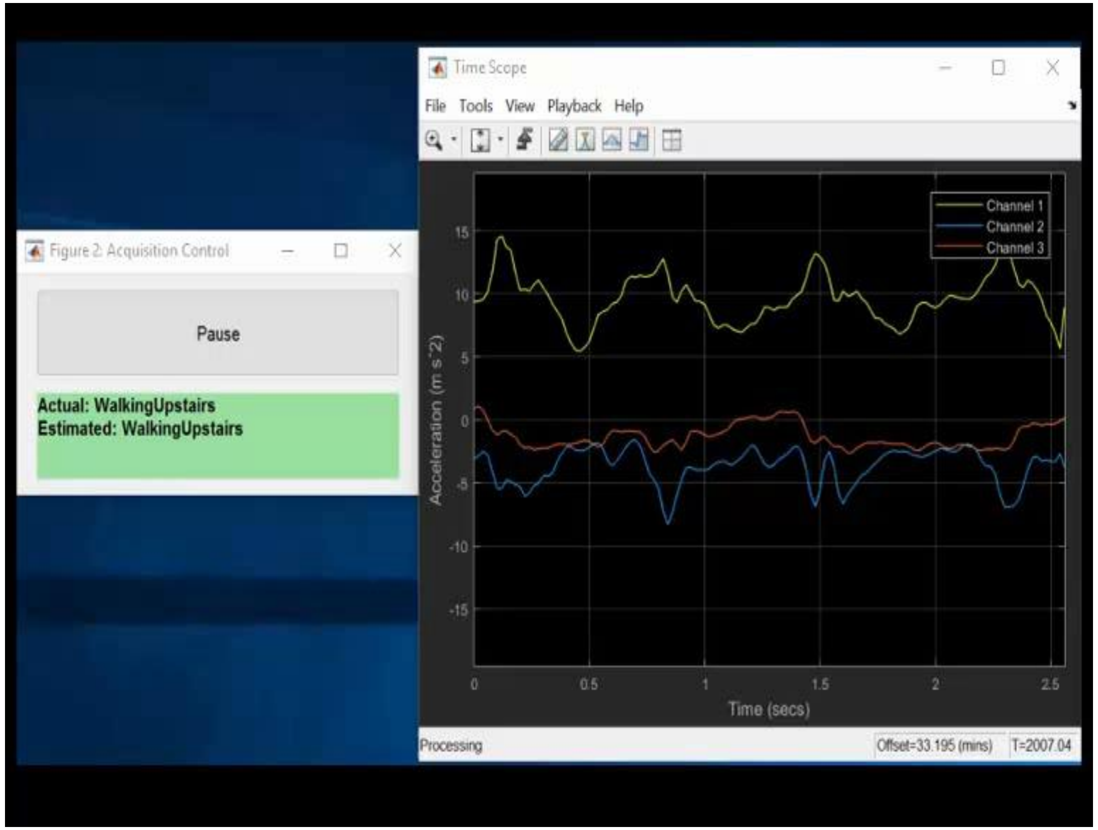
Detection and localization using deep learning

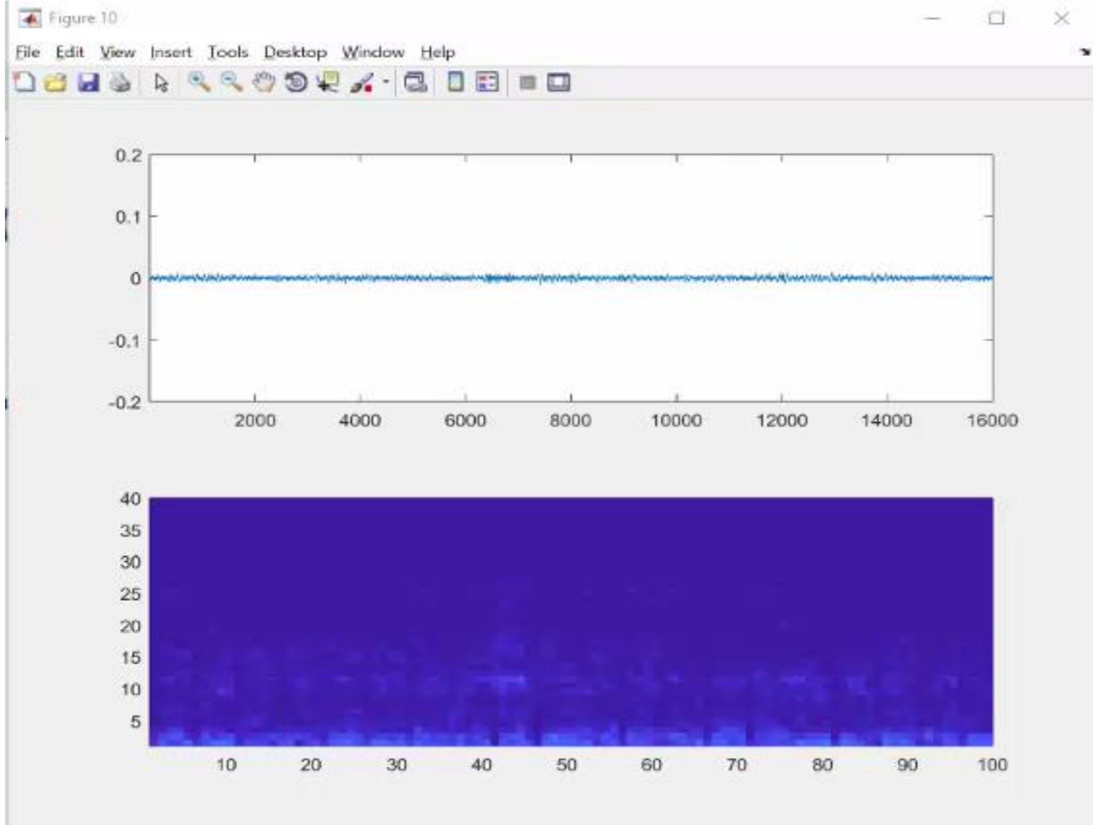


Regions with Convolutional Neural Network Features (R-CNN)

Semantic Segmentation using SegNet

Analyzing signal data using deep learning

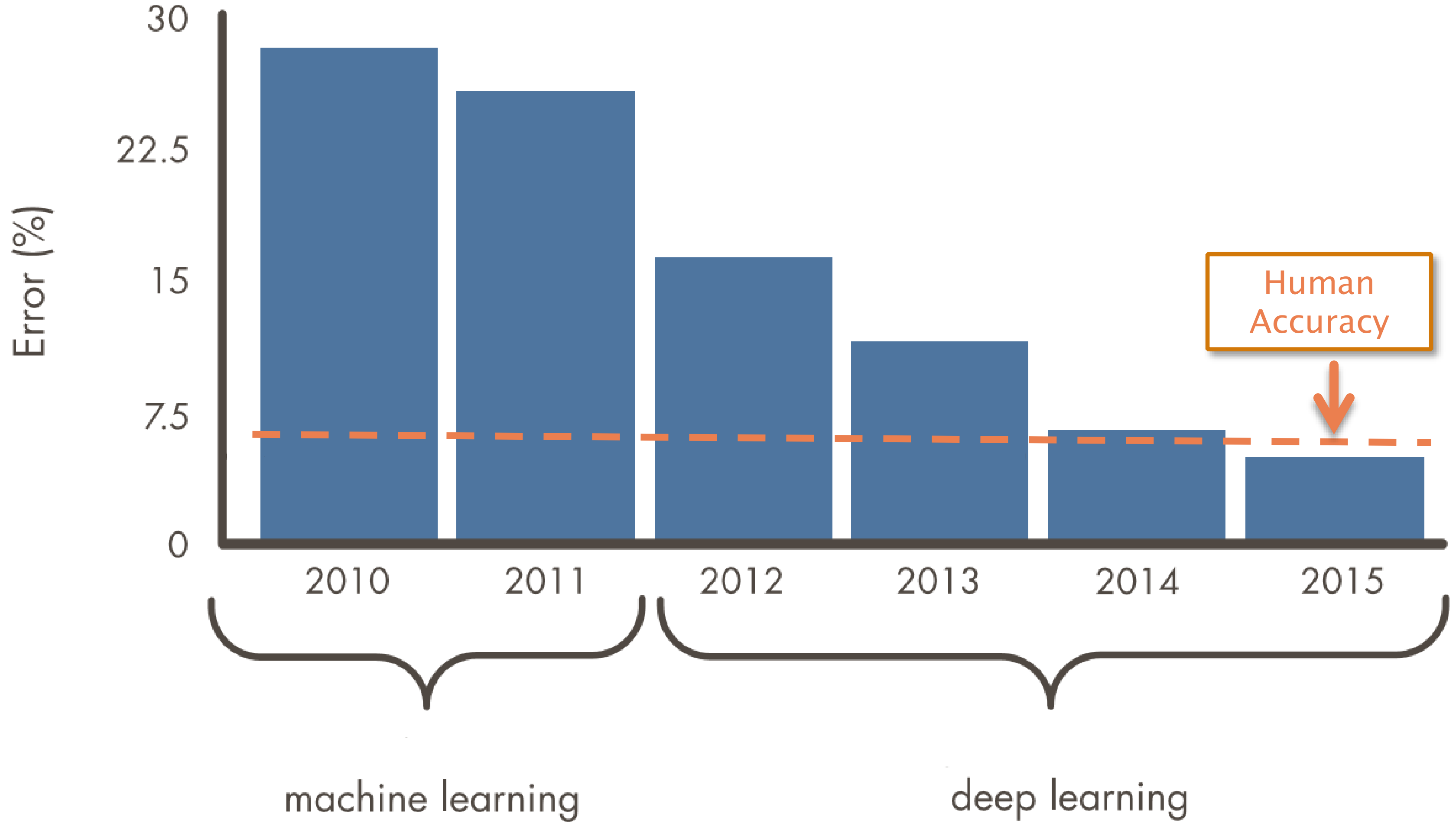




Signal Classification using LSTMs

Speech Recognition using CNNs

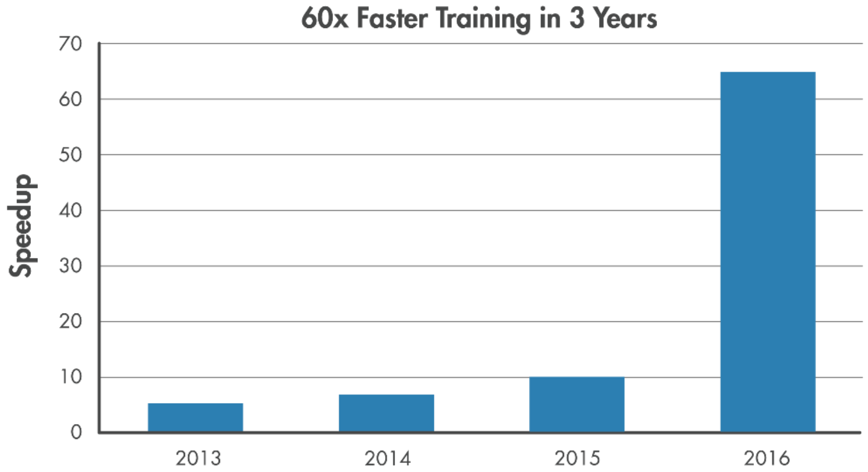
Why is Deep Learning So Popular Now?



Source: ILSVRC Top-5 Error on ImageNet

Deep Learning Enablers

- Increased GPU acceleration



- World-class models

AlexNet
PRETRAINED MODEL

VGG-16
PRETRAINED MODEL

ResNet-50
PRETRAINED MODEL

ONNX Converter
MODEL CONVERTER

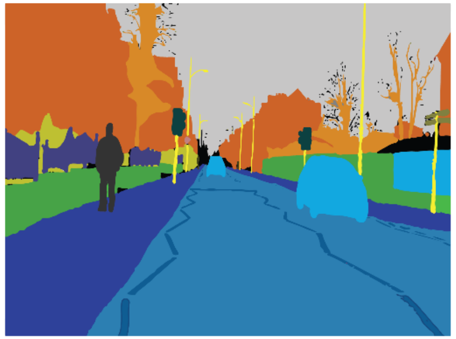
Caffe
IMPORTER

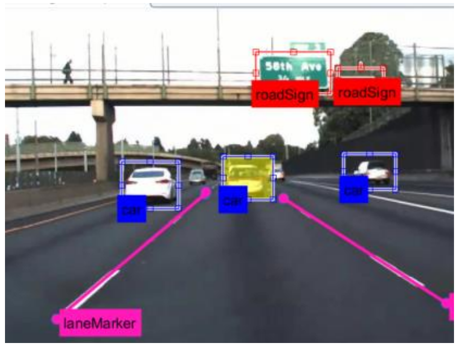
GoogLeNet
PRETRAINED MODEL

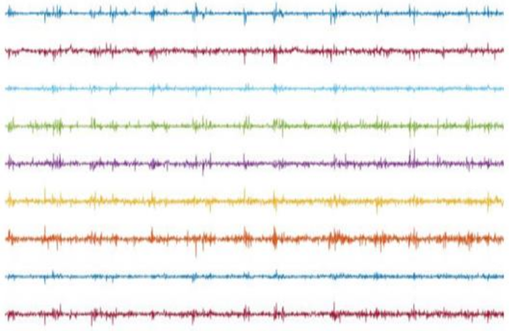
TensorFlow-Keras
IMPORTER

Inception-v3
MODELS

- Labeled public datasets







Deep Learning Workflow

Access and Explore Data → **Label and Preprocess Data** → **Develop Predictive Models** → **Integrate Models with Systems**

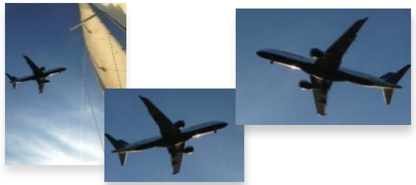
Access and Explore Data

Files

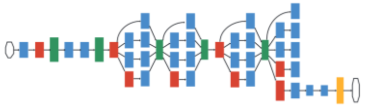

Databases


Sensors

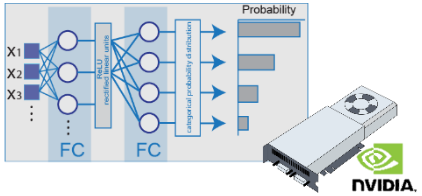

Label and Preprocess Data

Data Augmentation/ Transformation


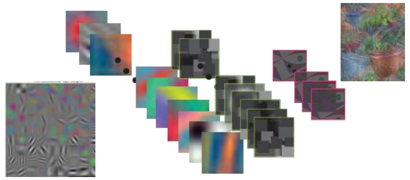
Labeling Automation


Import Reference Models


Develop Predictive Models

Hardware-Accelerated Training


Hyperparameter Tuning


Network Visualization


Integrate Models with Systems

Desktop Apps


Enterprise Scale Systems


Embedded Devices and Hardware


Introducing Deep Learning Inference in 5 lines

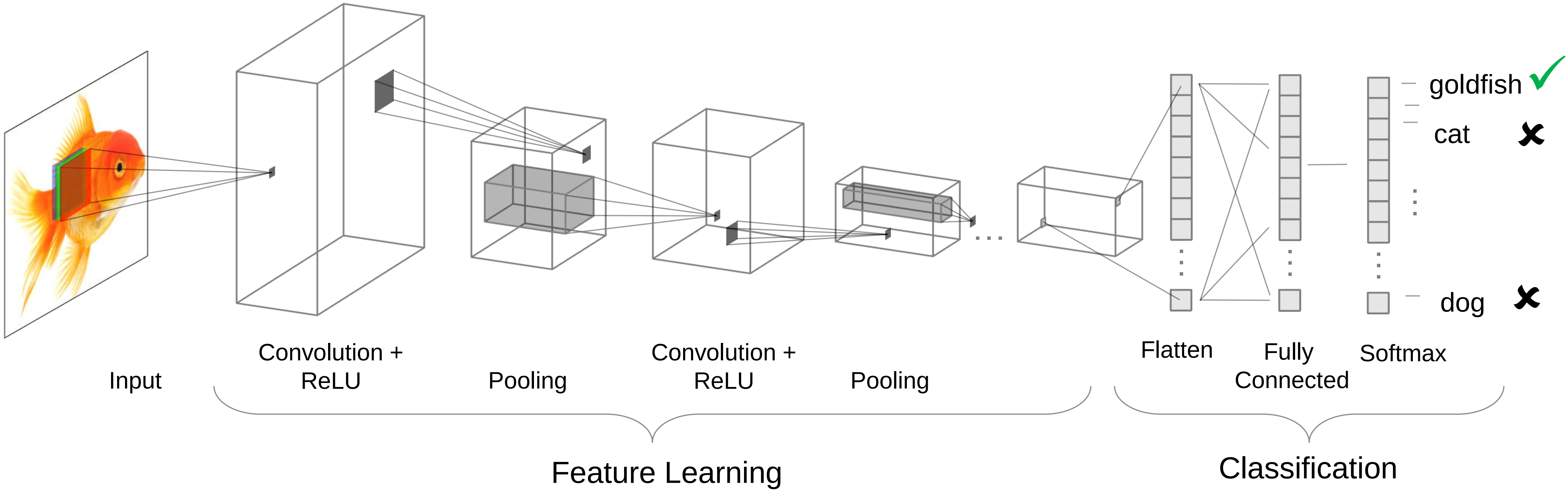
```
>> net = alexnet;
>> I = imread('peacock.jpg')
>> I1 = imresize(I,[227 227]);
>> classify(net,I1)
ans =
    categorical
       peacock
```

Classifies into classes that it has been trained on

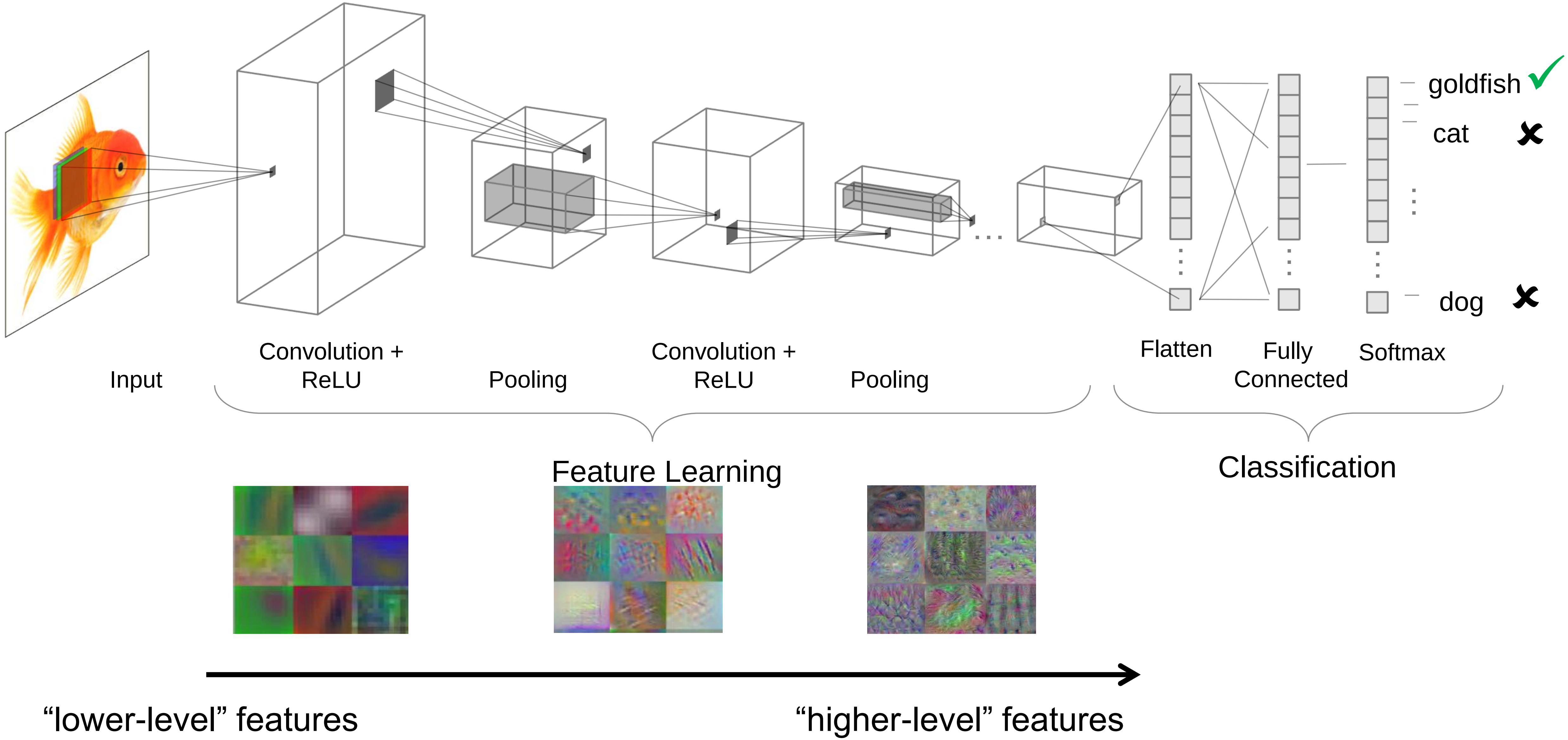
```
>> wordcloud(net.Layers(end).Classes)
```

imagenet has 1000 categories, 14 million+ images

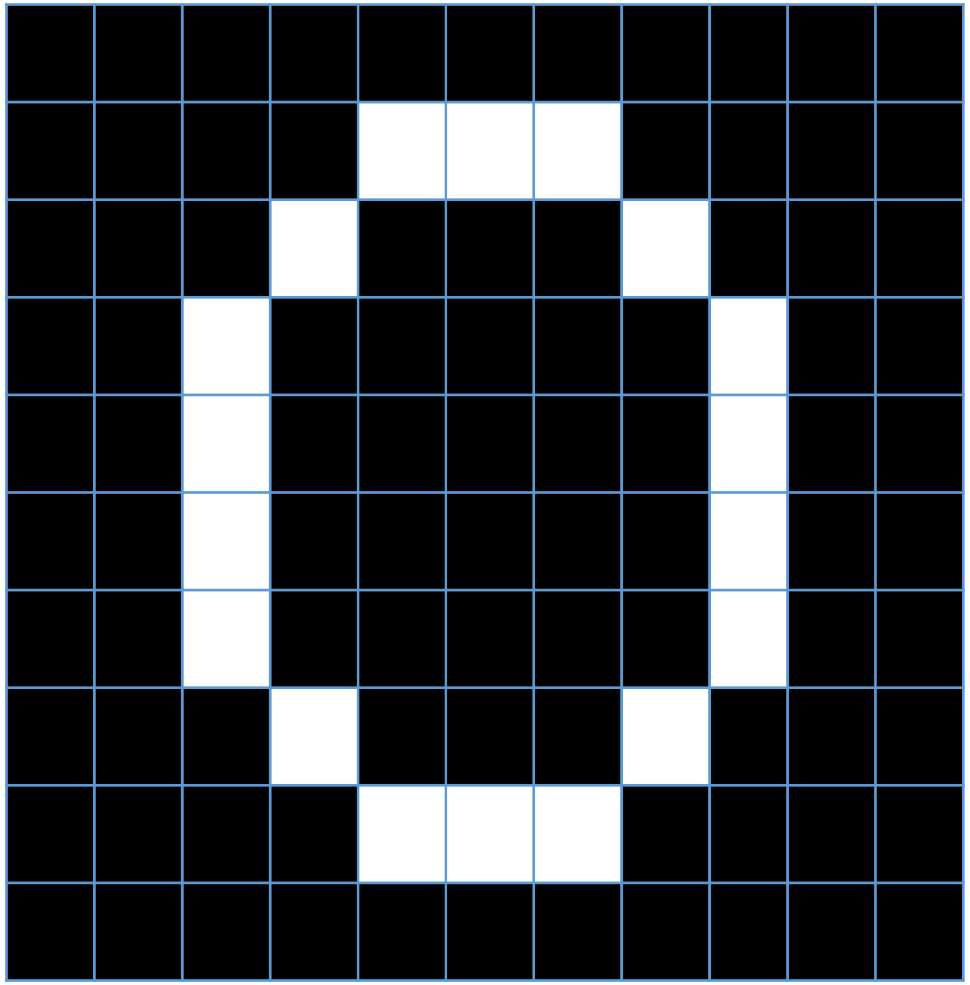
Convolutional Neural Network (CNN)/ Deep Neural Network (DNN)

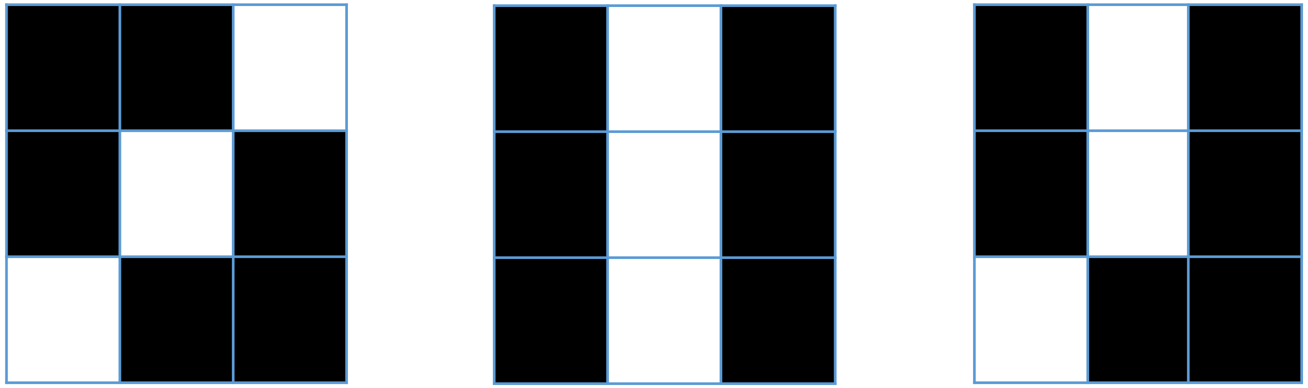


What do these layers learn?

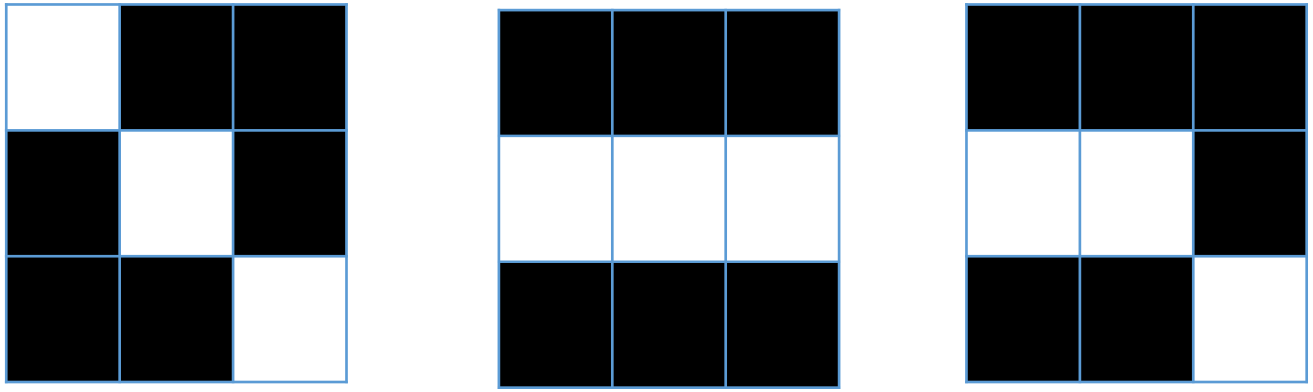


Convolution Layers Search for Patterns



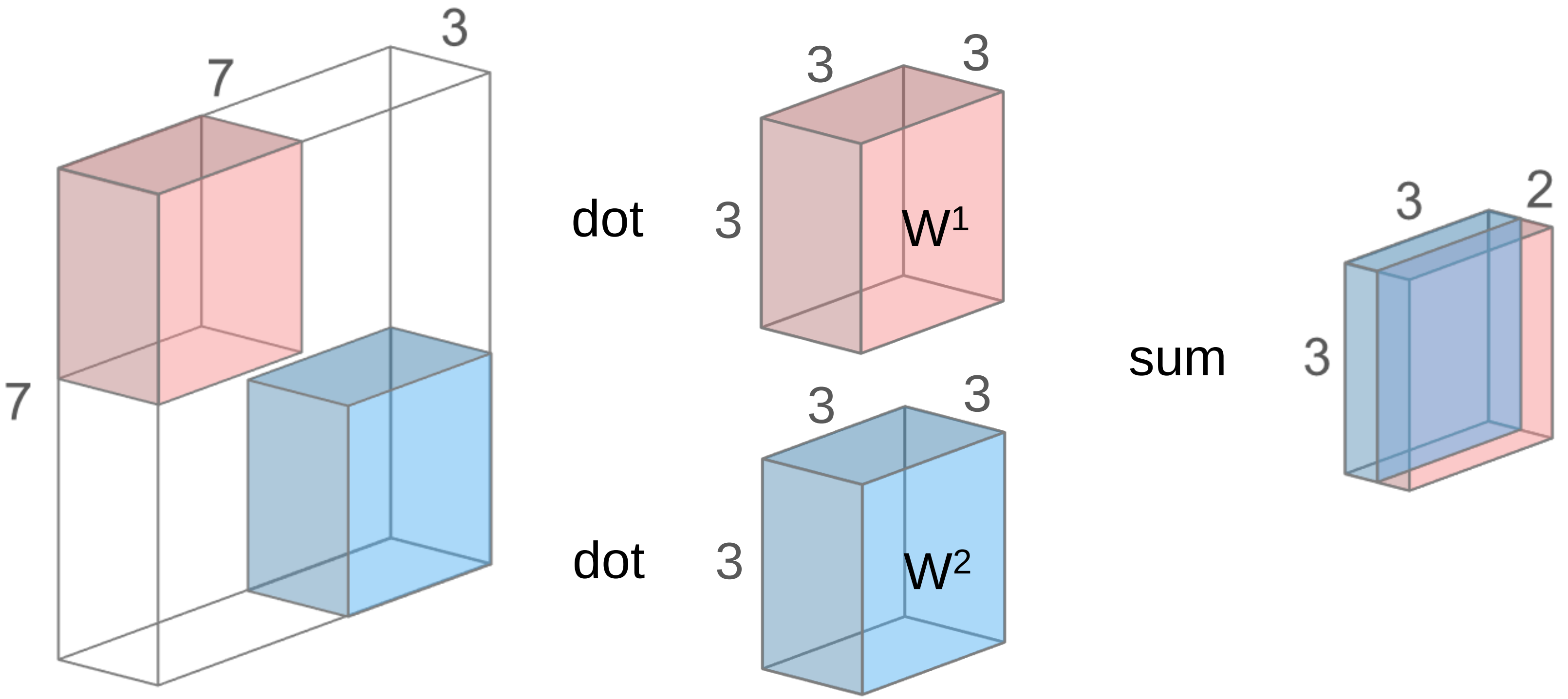


These patterns would be common in the number 0



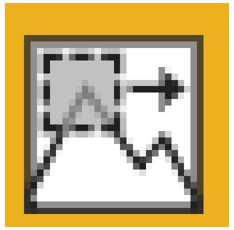
Convolution Layer – What makes a Deep Neural Network a Convolutional Neural Network

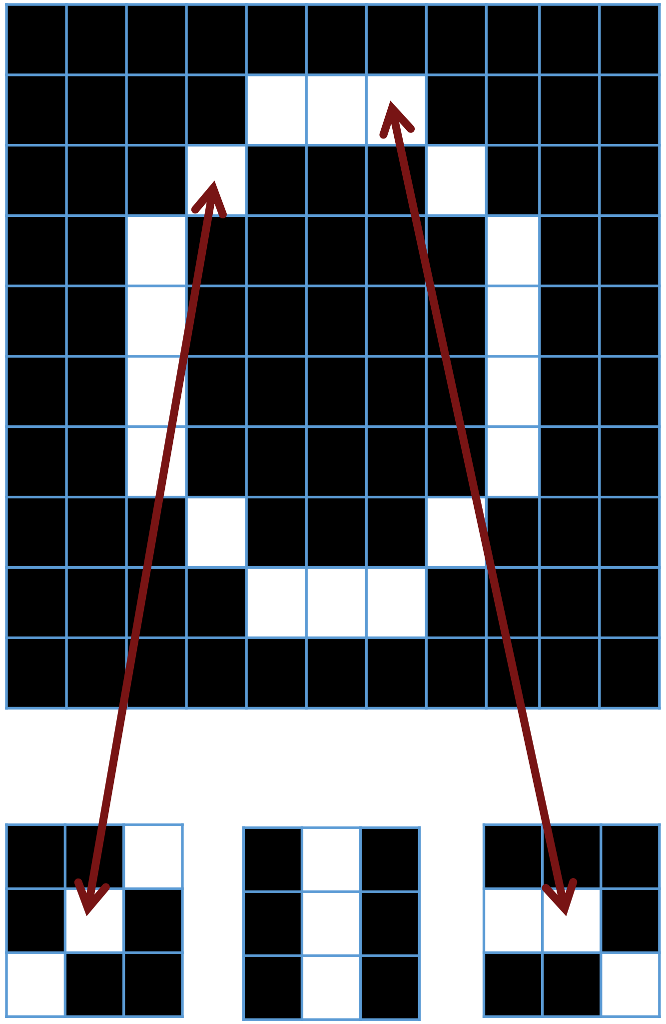
- Core building block of a CNN
- Convolve the filters sliding them across the input, computing the dot product

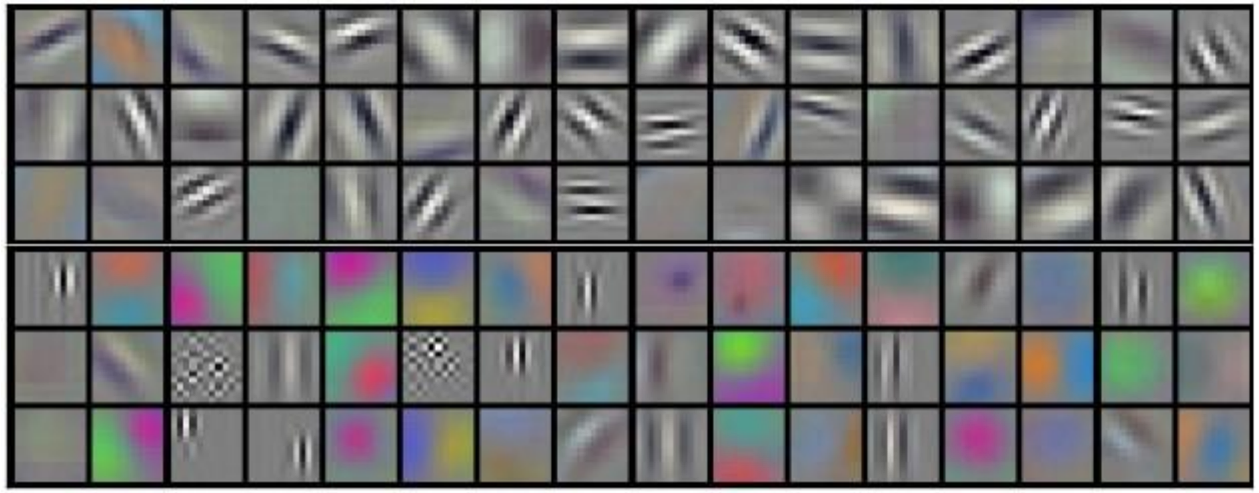


- Intuition: learn filters that activate when they "see" some specific feature

Convolution Layers Learn Patterns



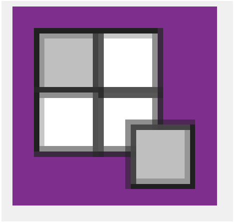




alexnet learnt these ‘patterns’ from imagenet – from 96 filters of size11-by-11 in ‘conv1’ (Krizhevsky et al, 2012)

```
% In MATLAB:
>> layer = convolution2dLayer(filterSize, numFilters)
```

See here for more: https://cs231n.github.io/convolutional-networks/



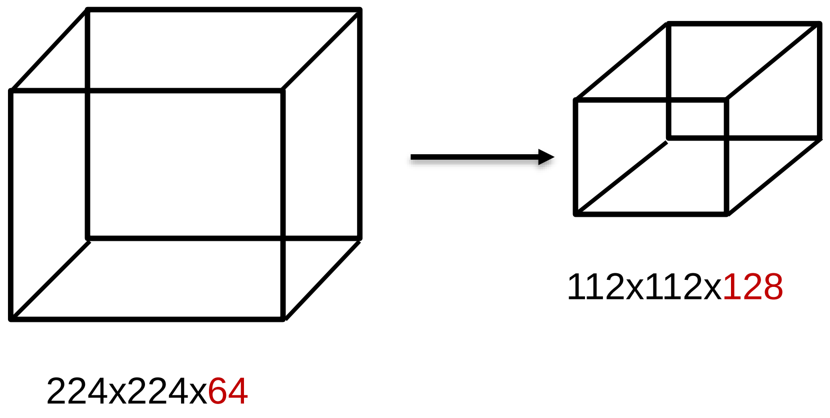
Max Pooling sub-samples activations

1	3	5	1
3	7	4	3
0	4	6	5
2	3	4	1

2x2 filters →

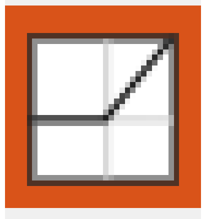
Stride Length = 2

7	5
4	6




```
% In MATLAB:
>> layer = maxPooling2dLayer(poolSize)
```

Also look for minPooling and averagePooling.



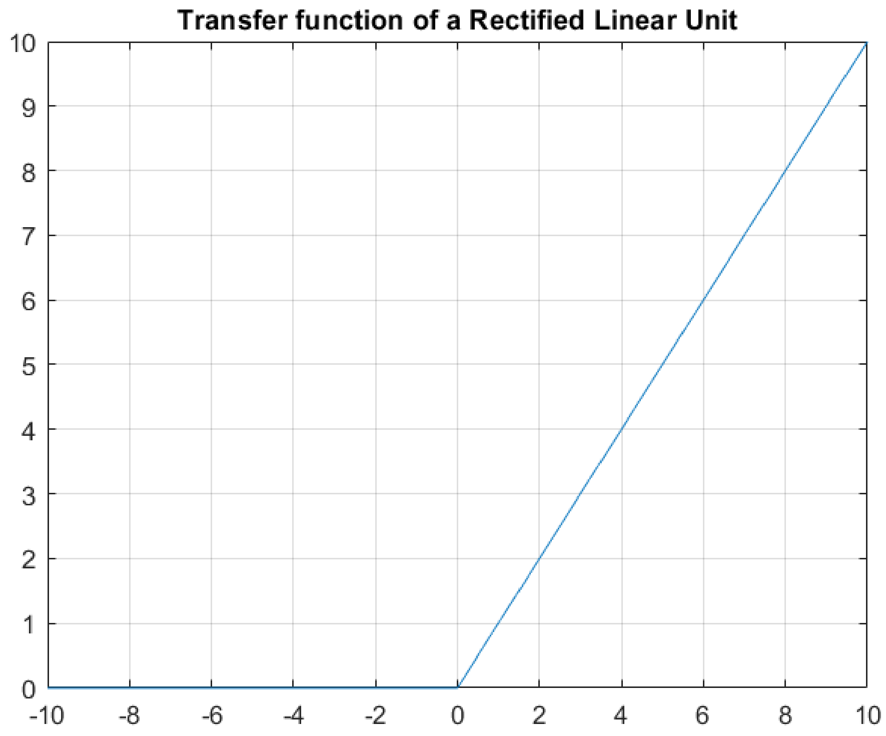
Rectified Linear Units Layer (ReLU)

Typically converts negative numbers to zero

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

→

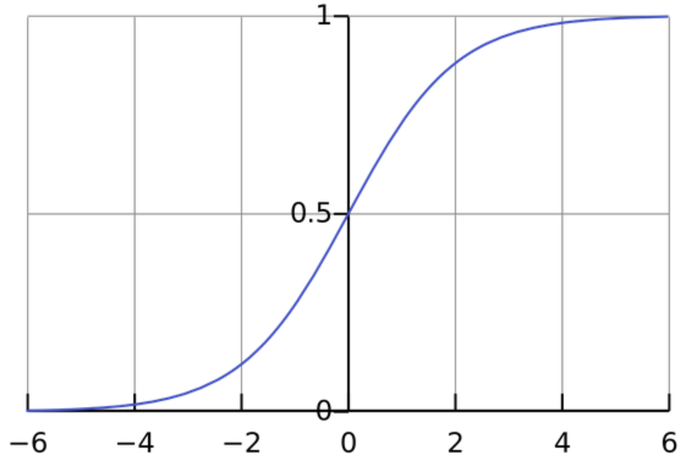
0	0	5	4
3	0	0	3
1	4	6	0
0	0	4	1




```
% In MATLAB:
>> layer = reluLayer()
```

Classification Problems End with 3 Layers

- Fully Connected Layer
  - Looks at which high-level features correspond to a specific category
  - Calculates scores for each category (highest score wins)



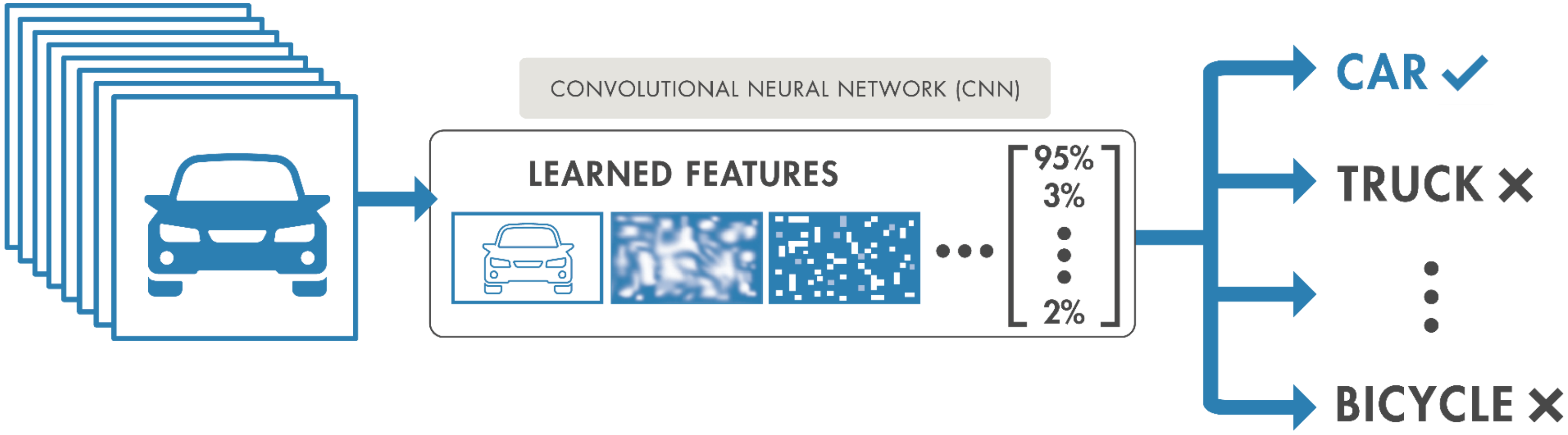
- Softmax Layer
  - Turns scores into probabilities.

- Classification Layer
  - Categorizes image into one of the classes that the network is trained on

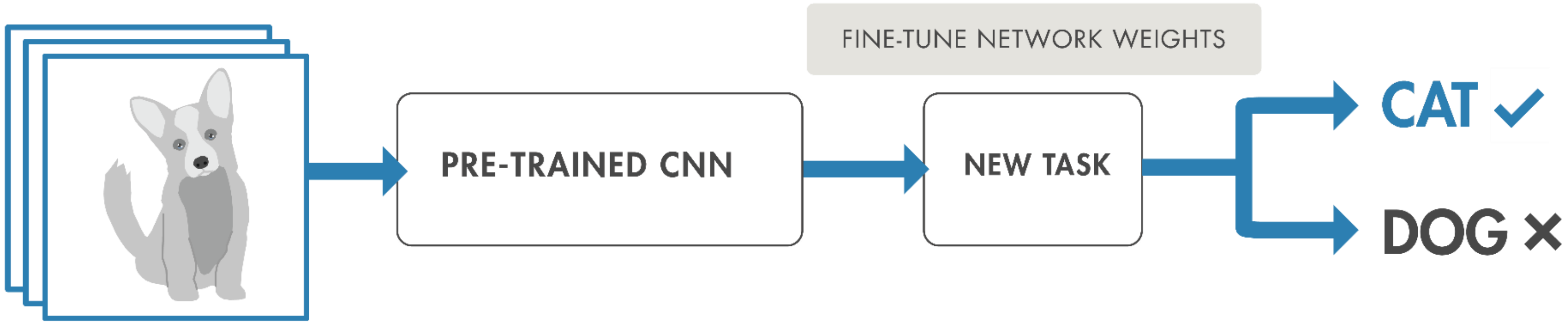
Note: Regression problems end with a fully connected layer and regression layer

Two Approaches for Deep Learning

1. Train a Deep Neural Network from Scratch



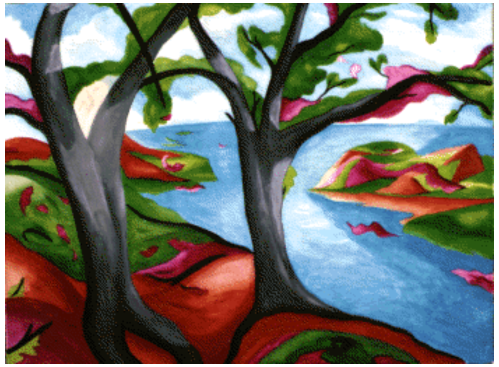
2. Fine-tune a pre-trained model (transfer learning)

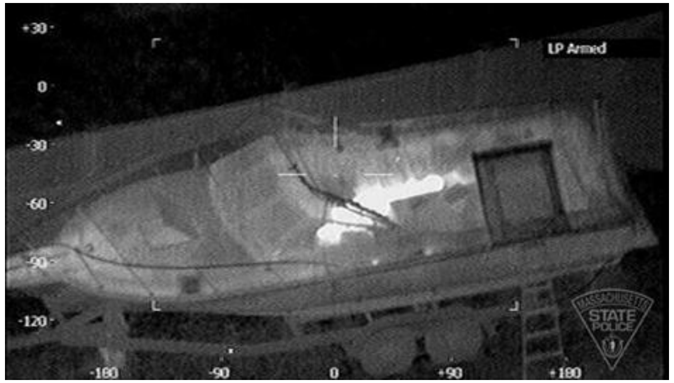


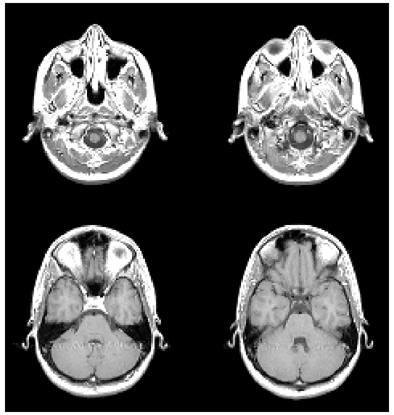
When Do I Need to Train My Own Model ?

- Available pre-trained models do not work
  - Low accuracy on your data-set
  - Different category definitions
  - Different task (classification v/s regression)

- Pre-trained model not available for your data type
  - Most available networks trained on natural images







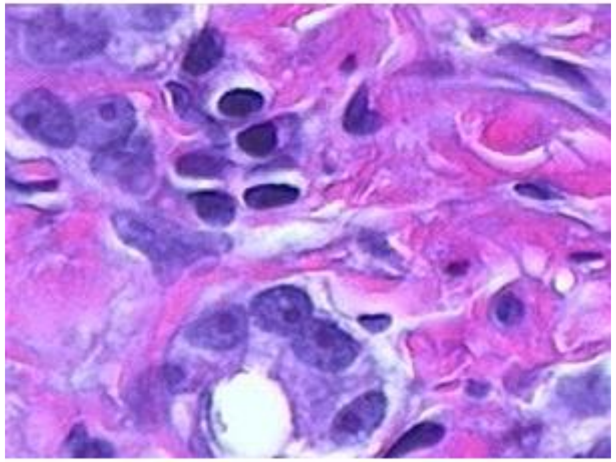


Image courtesy of Alan Partin, Johns Hopkins University

3 Components to Train a Network

Data

How much data?

It depends…but

A LOT

Network Architecture

Define Inputs and layers for deep learning

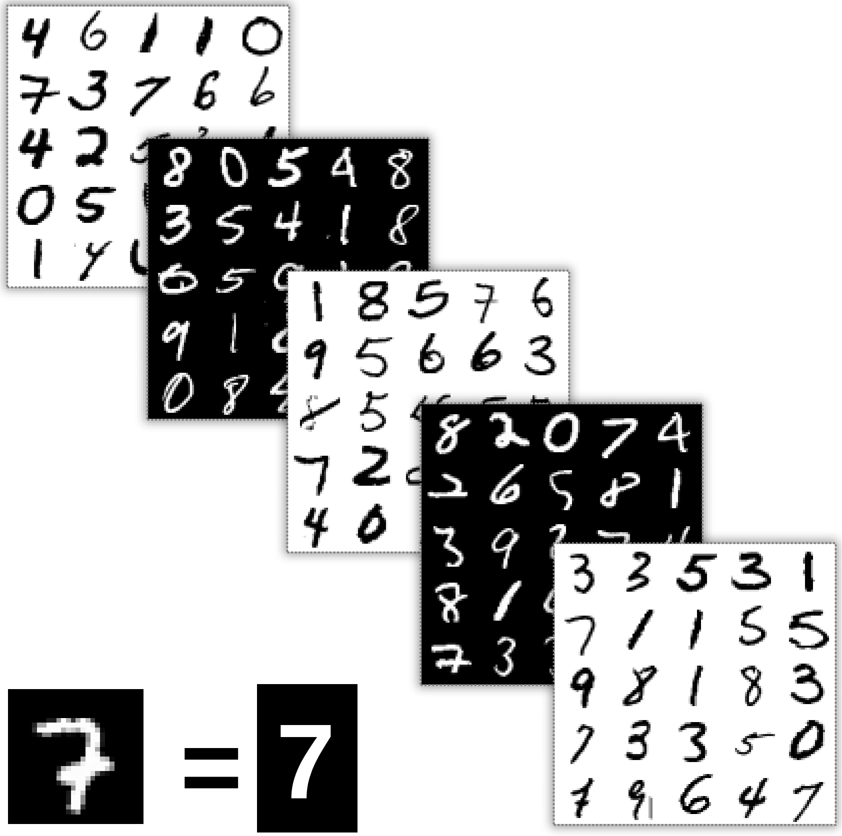
Training Options

Influence training time and accuracy

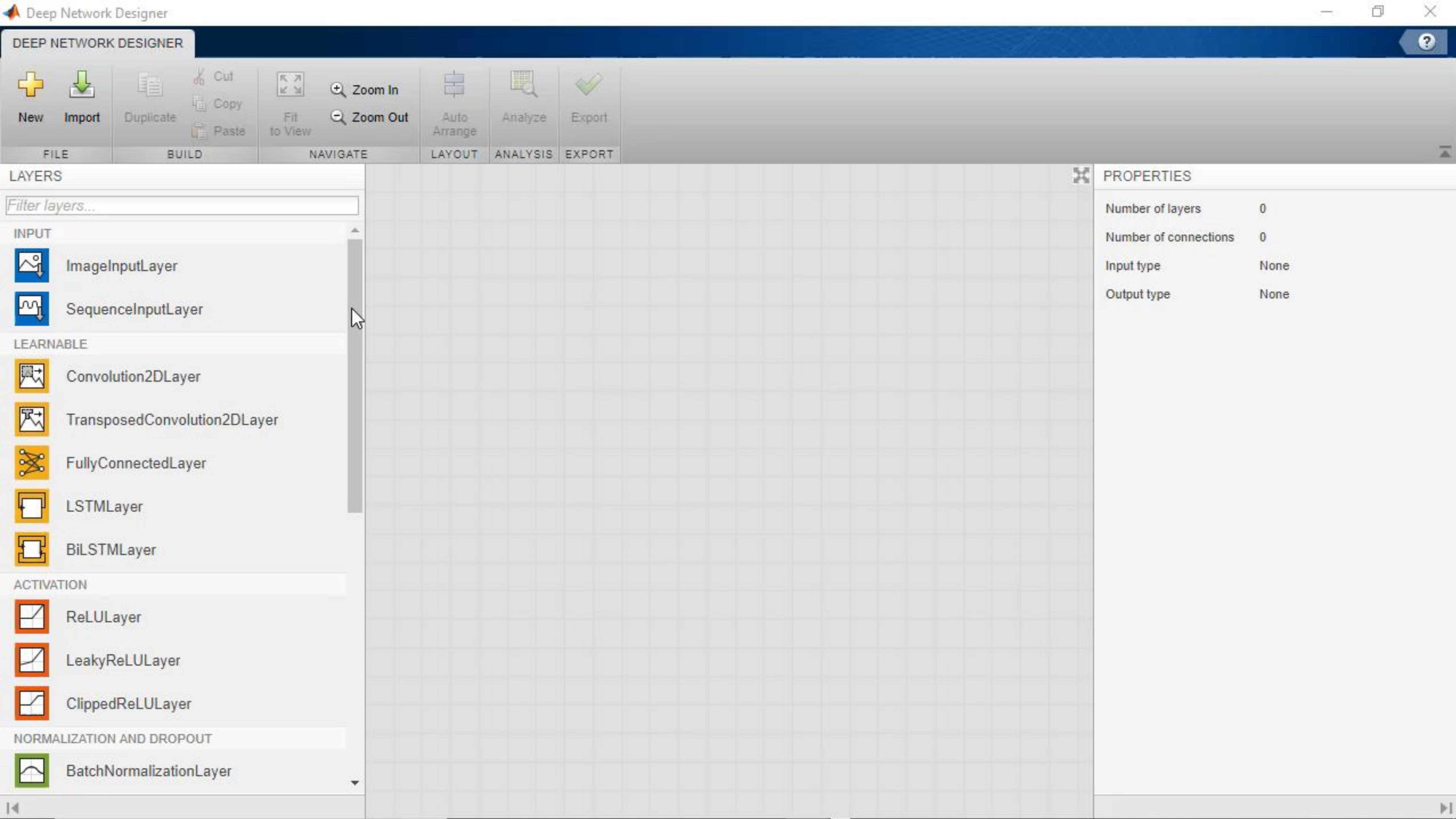
- Solver type
- Initial Learn Rate
- Minibatch Size
- Max Epochs
- …

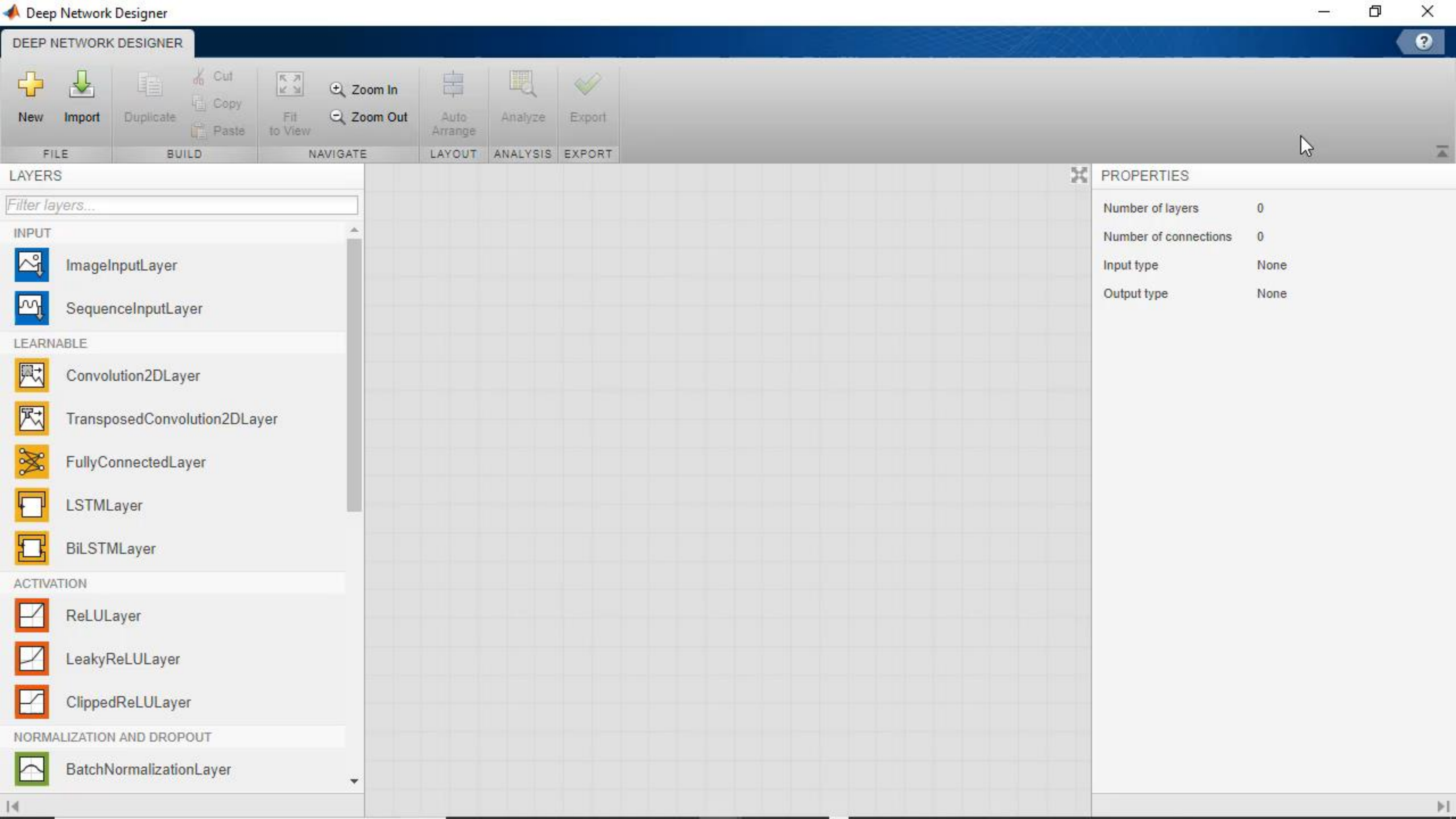
MNIST: The "Hello, World!" of computer vision

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



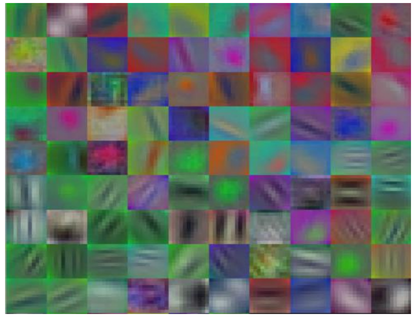
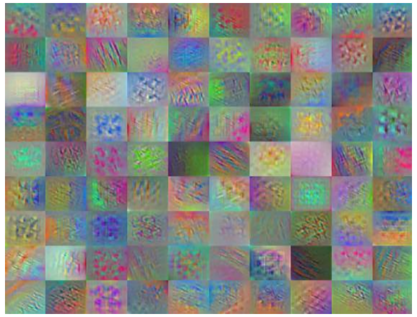
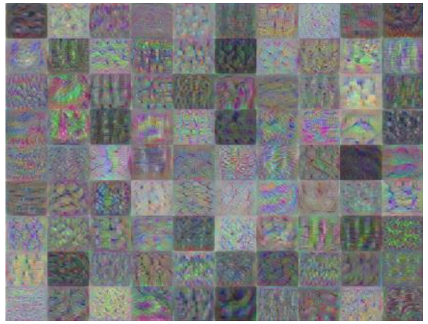
Sources: http://yann.lecun.com/exdb/mnist/
https://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results


Deep Network Designer
DEEP NETWORK DESIGNER
New
Import
Duplicate
Cut
Copy
Paste
Fit to View
Zoom In
Zoom Out
Auto Arrange
Analyze
Export
FILE
BUILD
NAVIGATE
LAYOUT
ANALYSIS
EXPORT
LAYERS
Filter layers...
INPUT
ImageInputLayer
SequenceInputLayer
LEARNABLE
Convolution2DLayer
TransposedConvolution2DLayer
FullyConnectedLayer
LSTMLayer
BiLSTMLayer
ACTIVATION
ReLULayer
LeakyReLULayer
ClippedReLULayer
NORMALIZATION AND DROPOUT
BatchNormalizationLayer
PROPERTIES
Number of layers
0
Number of connections
0
Input type
None
Output type
None

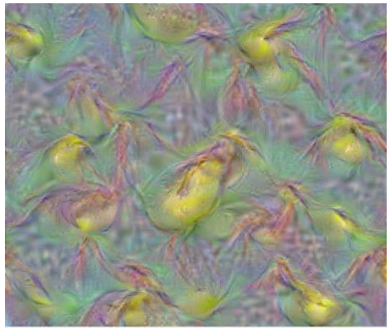

Deep Network Designer
DEEP NETWORK DESIGNER
New
Import
Duplicate
Cut
Copy
Paste
Fit to View
Zoom In
Zoom Out
Auto Arrange
Analyze
Export
FILE
BUILD
NAVIGATE
LAYOUT
ANALYSIS
EXPORT
LAYERS
Filter layers...
INPUT
ImageInputLayer
SequenceInputLayer
LEARNABLE
Convolution2DLayer
TransposedConvolution2DLayer
FullyConnectedLayer
LSTMLayer
BiLSTMLayer
ACTIVATION
ReLULayer
LeakyReLULayer
ClippedReLULayer
NORMALIZATION AND DROPOUT
BatchNormalizationLayer
PROPERTIES
Number of layers 0
Number of connections 0
Input type None
Output type None

Visualizations for Understanding Network Behavior

Filters

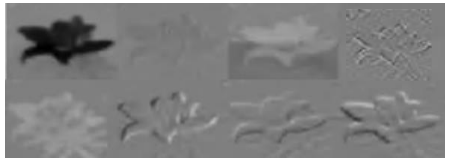
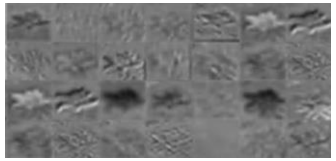
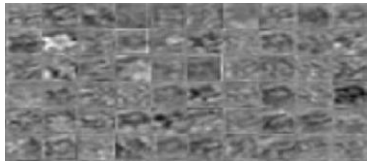




• • •



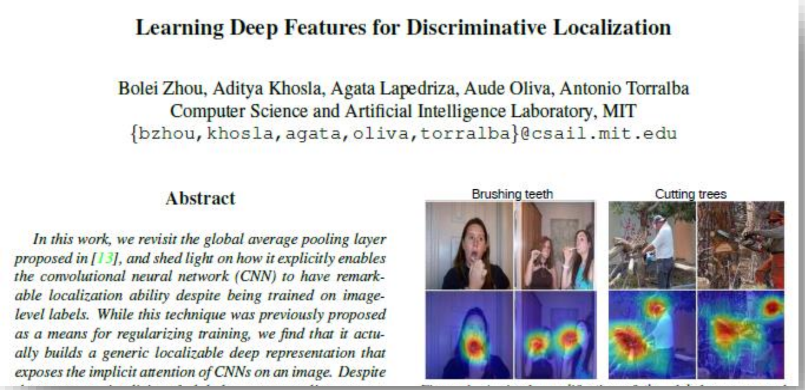
Deep Dream

Activations

- generate images that strongly activate a particular channel of the network layers.
- Custom visualizations
  - Example: Class Activation Maps
- Use activations on layers

http://cnnlocalization.csail.mit.edu/Zhou_Learning_Deep_Features_CVPR_2016_paper.pdf



Learning Deep Features for Discriminative Localization

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, Antonio Torralba
Computer Science and Artificial Intelligence Laboratory, MIT
{bzhou,khosla,agata,oliva,torralba}@csail.mit.edu

Abstract

In this work, we revisit the global average pooling layer proposed in [13], and shed light on how it explicitly enables the convolutional neural network (CNN) to have remarkable localization ability despite being trained on image-level labels. While this technique was previously proposed as a means for regularizing training, we find that it actually builds a generic localizable deep representation that exposes the implicit attention of CNNs on an image. Despite

Visualization Technique – Deep Dream

```
deepDreamImage(...
      net, 'fc5', channel,
     'NumIterations', 50, ...
     'PyramidLevels', 4,...
     'PyramidScale', 1.25);
```

Synthesizes images that strongly activate a channel in a particular layer

Example Available Here

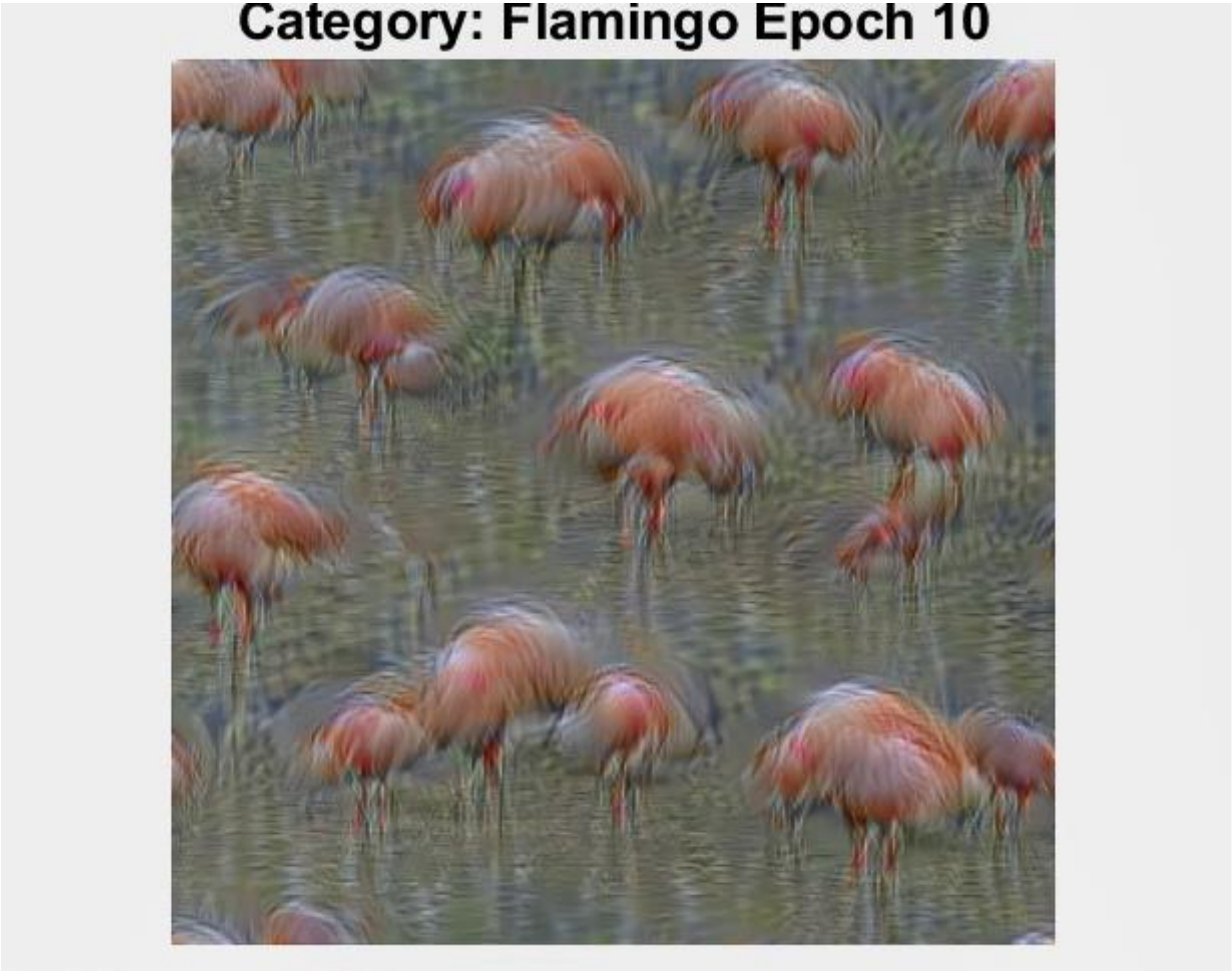
Visualize Features Learned During Training
AlexNet Example

Sample Training Data

Features Learned by Network

Visualize Features Learned During Training
AlexNet Example

Sample Training Data

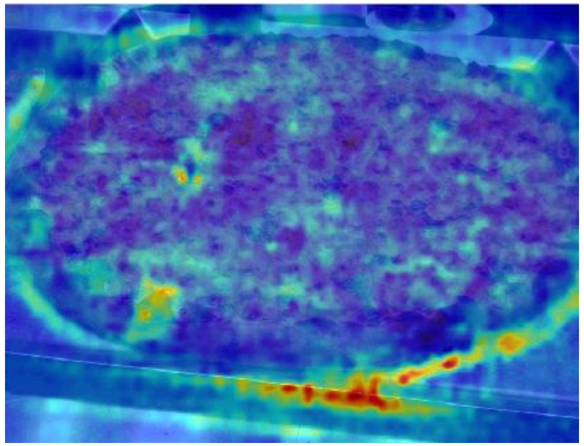


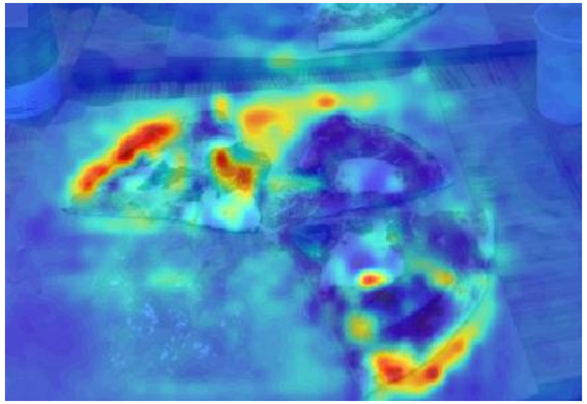
Features Learned by Network

occlusionSensitivity for visualizing what layers mostly “look” at

- Easy-to-understand visualization to help explain *why* a network makes the decision it does
- Most commonly used for image classification e.g. to show which part of the image causes the ‘pizza’ classification

```
pizzaImg = imds.read();
predLabel = classify(net, pizzaImg);
% “pizza” – but why?

map = occlusionSensitivity(net, pizzaImg, ...
  predLabel);

imshow(pizzaImg); hold on;
imagesc(map, ‘Alpha’, 0.5);
```

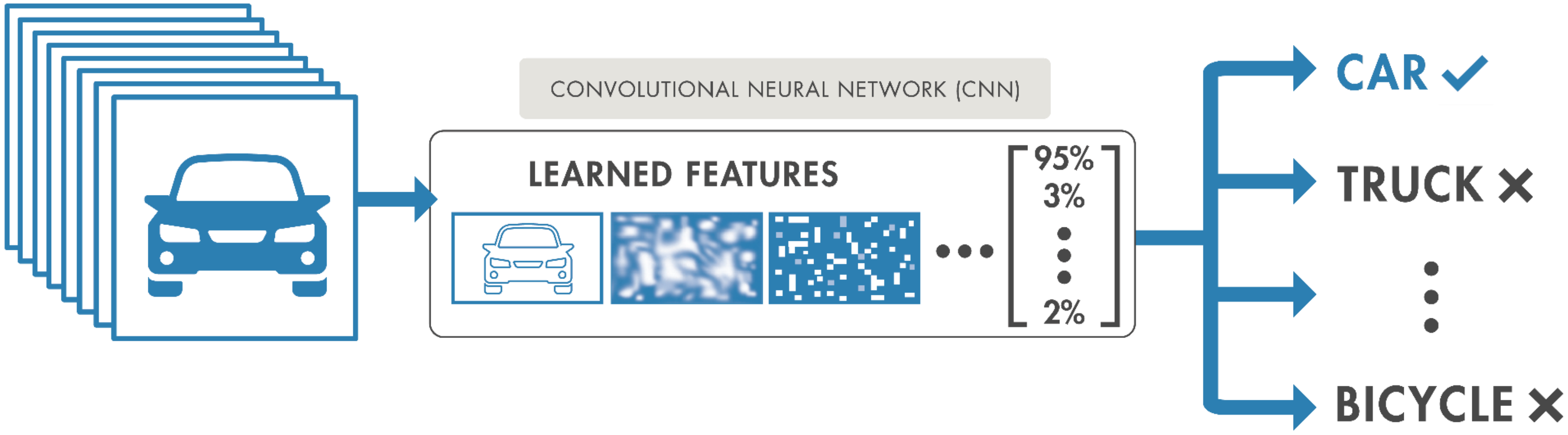




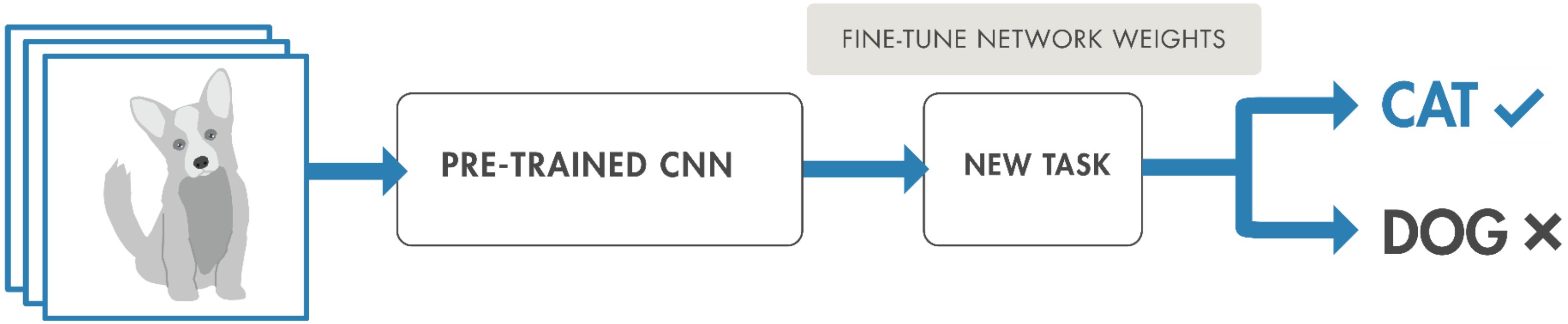
MathWorks

Two Approaches for Deep Learning

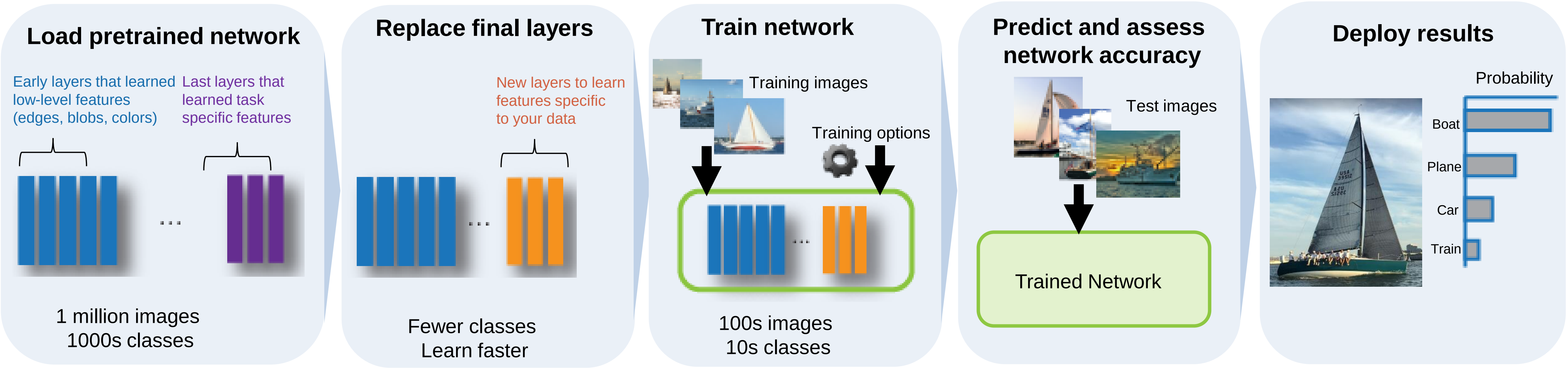
✓ **Train a Deep Neural Network from Scratch**



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



Transfer Learning Workflow



Example: Food classifier using deep transfer learning





Hot dog →

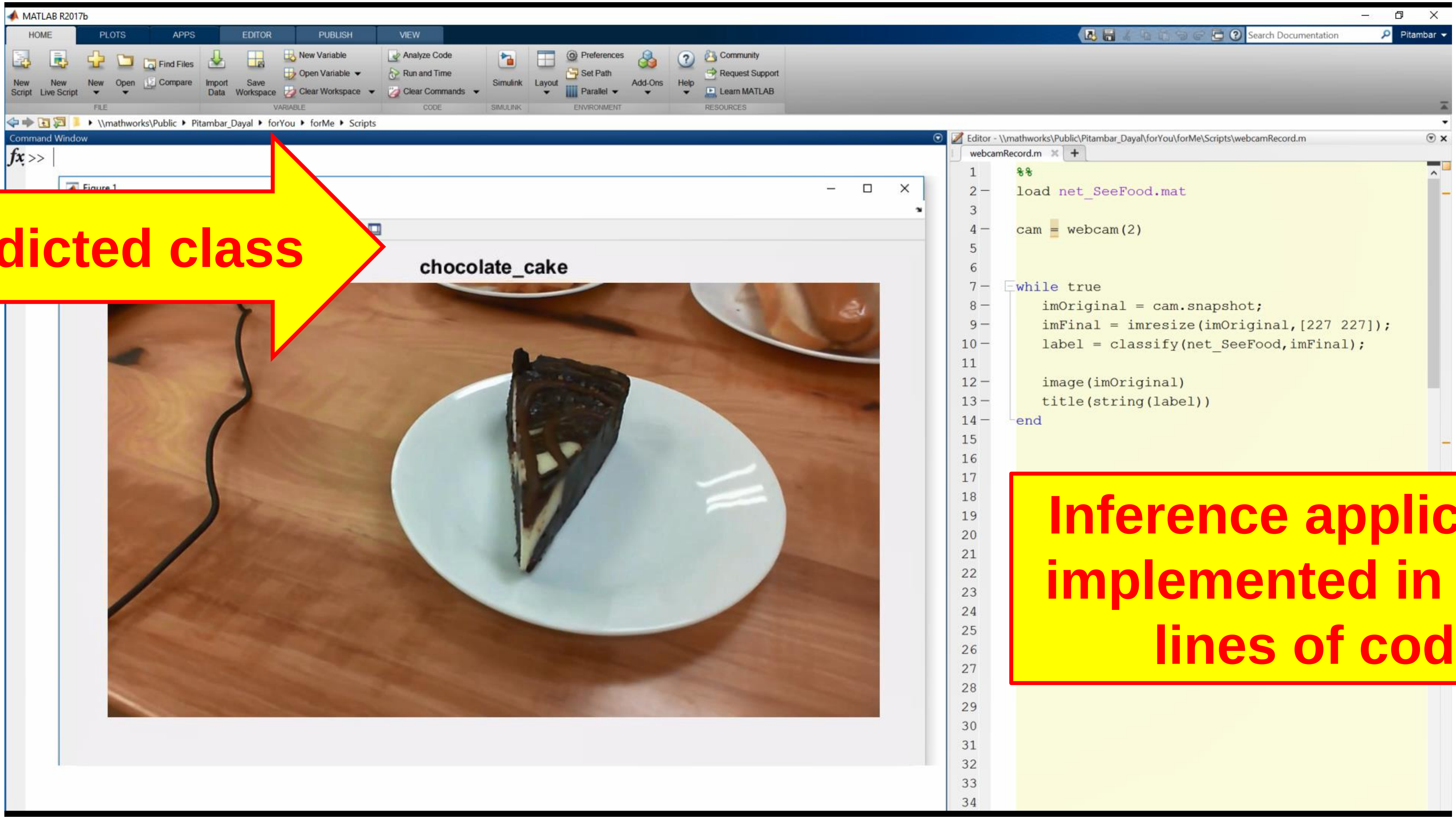
French fries →

Chocolate cake →

Pizza →

Ice cream →

5 Category Classifier

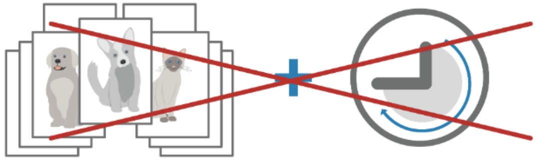


Predicted class

Inference application implemented in a few lines of code

Why Perform Transfer Learning ?

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



Pretrained Models in MATLAB

- AlexNet `net = alexnet;`
- VGG-16 `net = vgg16;`
- VGG-19 `net = vgg19;`
- GoogLeNet `net = googlenet;`
- Inceptionv3 `net = inceptionv3;`
- Resnet50 `net = resnet50;`
- Resnet101 `net = resnet101;`
- InceptionResnetv2 `net = inceptionresnetv2;`
- Squeezenet `net = squeezenet;`

Download from within MATLAB

Two Approaches for Deep Learning

Train a deep neural network from scratch

Recommended when:

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

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

Recommended when:

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

Hurdles at every turn…

ACCESS AND EXPLORE DATA

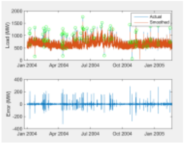
LABEL AND PREPROCESS DATA

Files

Databases

Sensors

Working with Messy Data



Labeling Automation

Data Augmentation/ Reduction/ Transformation

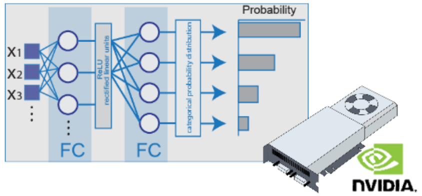
- Big Data = Big Memory/Disk.
- Visualizing and Analyzing Labeled Data?
- Actually labeling large datasets?
- Handling insufficient or poor quality data?

Hurdles at every turn…

- Training a model is more art than science! Can I train on my laptop/desktop? How to scale up to a GPU?
- How do I tune hyperparameters? Learning Rate, Weights Initialization?
- Model assessment – how good is good?
- How to deploy a trained model? Desktop, Cloud, Embedded Devices?

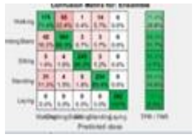
DEVELOP PREDICTIVE MODELS

Hardware-Accelerated Training



Hyperparameter Tuning

Model Validation



INTEGRATE MODELS WITH SYSTEMS

Desktop Apps

Enterprise Scale Systems

Embedded Devices and Hardware

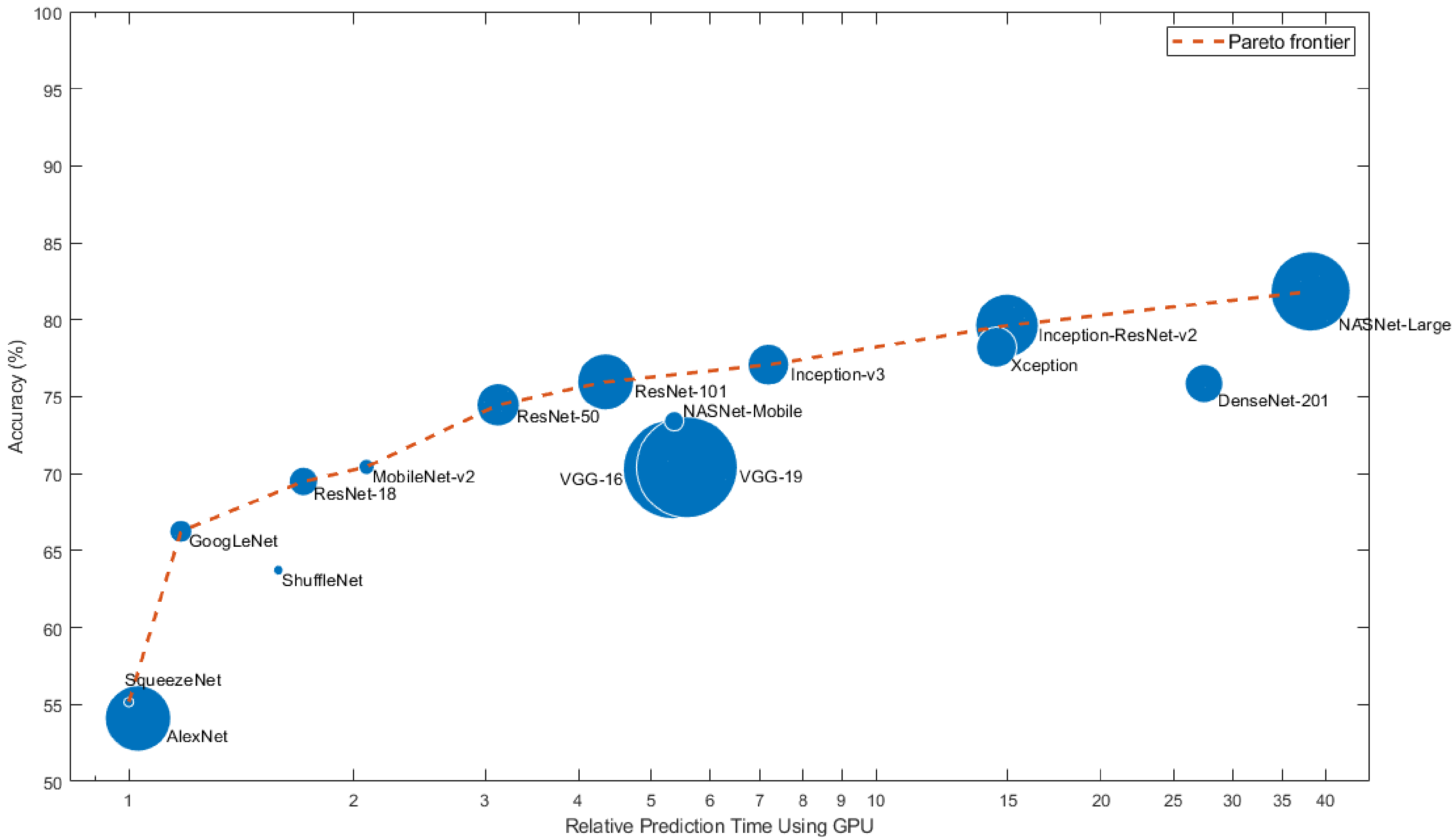
Top five issues we hear about while working with Deep Neural Networks

1. How do I choose a network architecture?
2. How much data do I need?
3. How do I improve accuracy of my network?
4. How do I speed up training?
5. I have a trained model – what do I do next?

#1: How do I choose a network architecture?

- Image Classification?
  - alexnet
  - vgg16
  - vgg19
  - squeezenet
  - nasnetmobile
  - resnet50
  - resnet101
  - googlenet
  - inceptionv3
  - …
  - Roll your own architecture!
  - more in the near future.
- Object Detection?
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - YOLO v2
  - more in the near future.
- Semantic Segmentation?
  - Segnet
  - U-net
  - DeepLabv3plus
  - more in the near future.
- Instance Segmentation?
- More exotic networks?
  - Talk to us! We're interested to learn more about your research!

#1: Accuracy vs run-time performance of classification



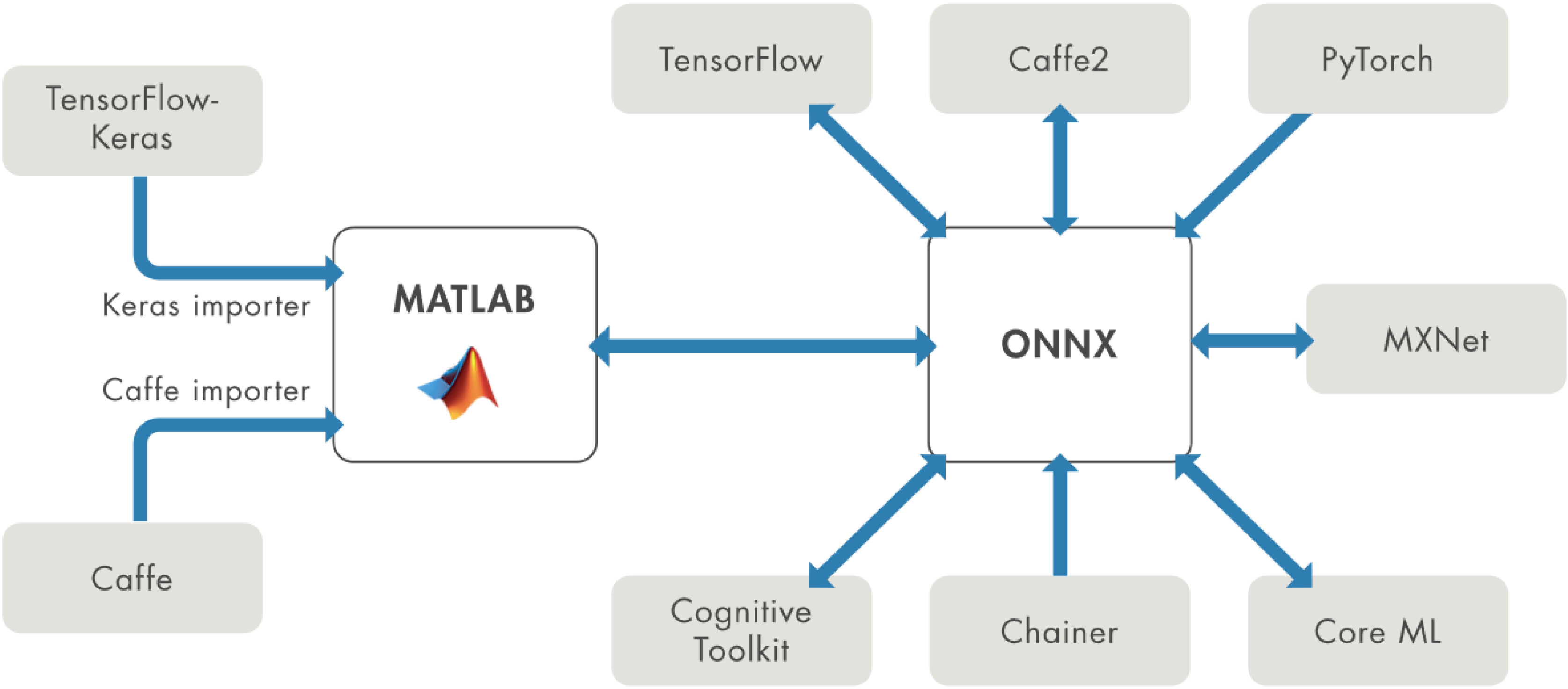
A network is *Pareto efficient* if there is no other network that is better on the accuracy and prediction time boundary. The set of all Pareto efficient networks is called the ***Pareto frontier.***

#1: Memory/Depth constraints for Classification Networks

Network	Depth	Size	Parameters (Millions)	Image Input Size
alexnet	8	227 MB	61.0	227-by-227
vgg16	16	515 MB	138	224-by-224
vgg19	19	535 MB	144	224-by-224
squeezenet	18	4.6 MB	1.24	227-by-227
googlenet	22	27 MB	7.0	224-by-224
inceptionv3	48	89 MB	23.9	299-by-299
densenet201	201	77 MB	20.0	224-by-224
mobilenetv2	53	13 MB	3.5	224-by-224
resnet18	18	44 MB	11.7	224-by-224
resnet50	50	96 MB	25.6	224-by-224
resnet101	101	167 MB	44.6	224-by-224
xception	71	85 MB	22.9	299-by-299
inceptionresnetv2	164	209 MB	55.9	299-by-299
shufflenet	50	6.3 MB	1.4	224-by-224
nasnetmobile	*	20 MB	5.3	224-by-224
nasnetlarge	*	360 MB	88.9	331-by-331

#1. Where Can I Access Pretrained Models?

- Many are built into MATLAB
- Others can found on the web and imported into MATLAB



Open Neural Network Exchange

Interoperability with other Deep Learning Frameworks

KERAS IMPORTER

Importer for TensorFlow-Keras Models

Caffe

MODELS

TensorFlow-Keras Model Importer

- `importKerasLayers`
- `importKerasNetwork`

ONNX - Importer/ Exporter

- `importONNXNetwork`
- `importONNXLayers`
- `exportONNXNetwork`

Import Models from Frameworks

- Caffe Model Importer (including Caffe Model Zoo)
  - `importCaffeLayers`
  - `importCaffeNetwork`

Download from within MATLAB

Add-Ons

Top five issues we hear about while working with Deep Neural Networks

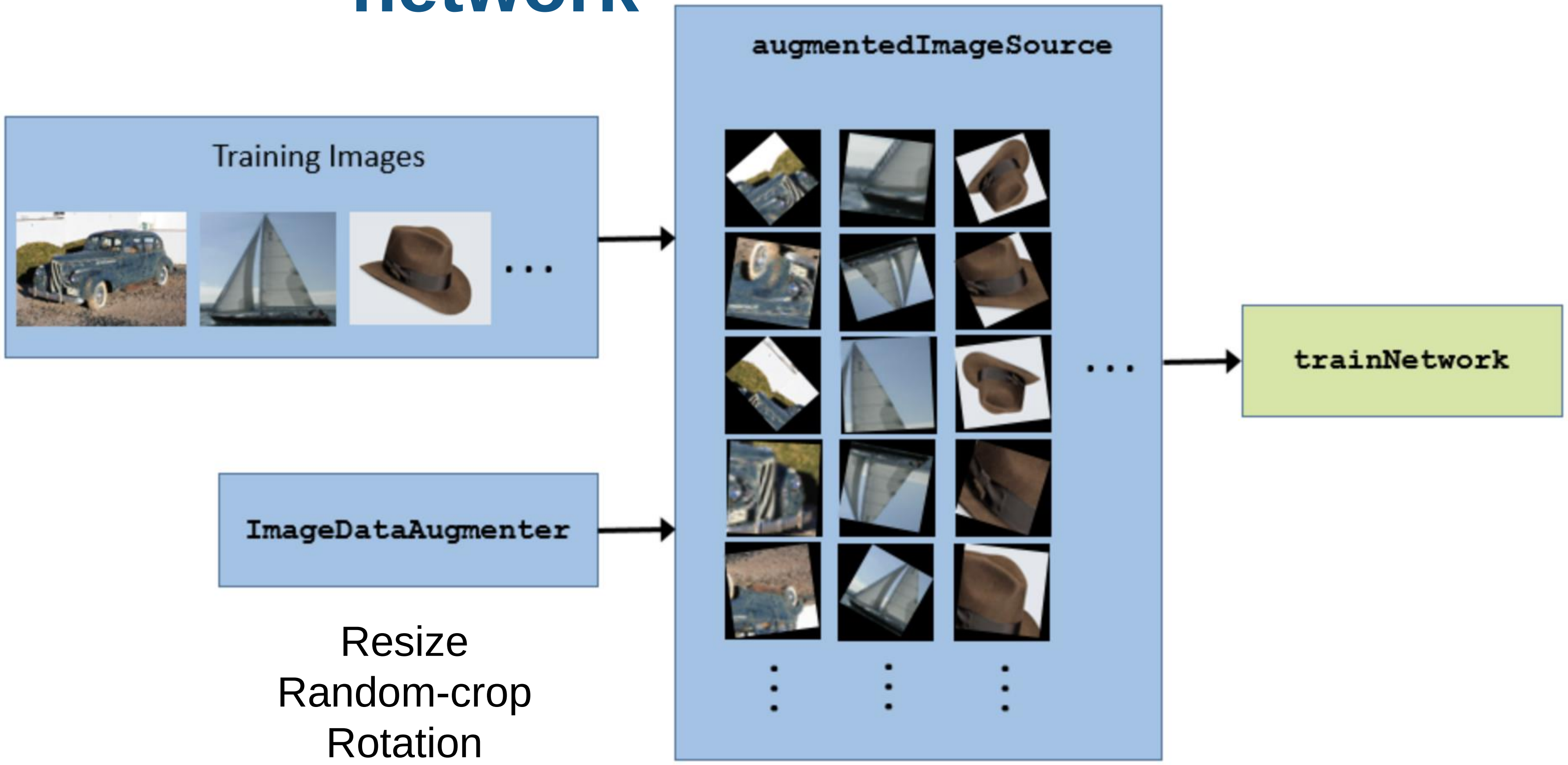
1. How do I choose a network architecture?
2. How much data do I need?
3. How do I improve accuracy of my network?
4. How do I speed up training?
5. I have a trained model – what do I do next?

#2: Need LOTS of data! + Augmentation to generalize network

How much data?

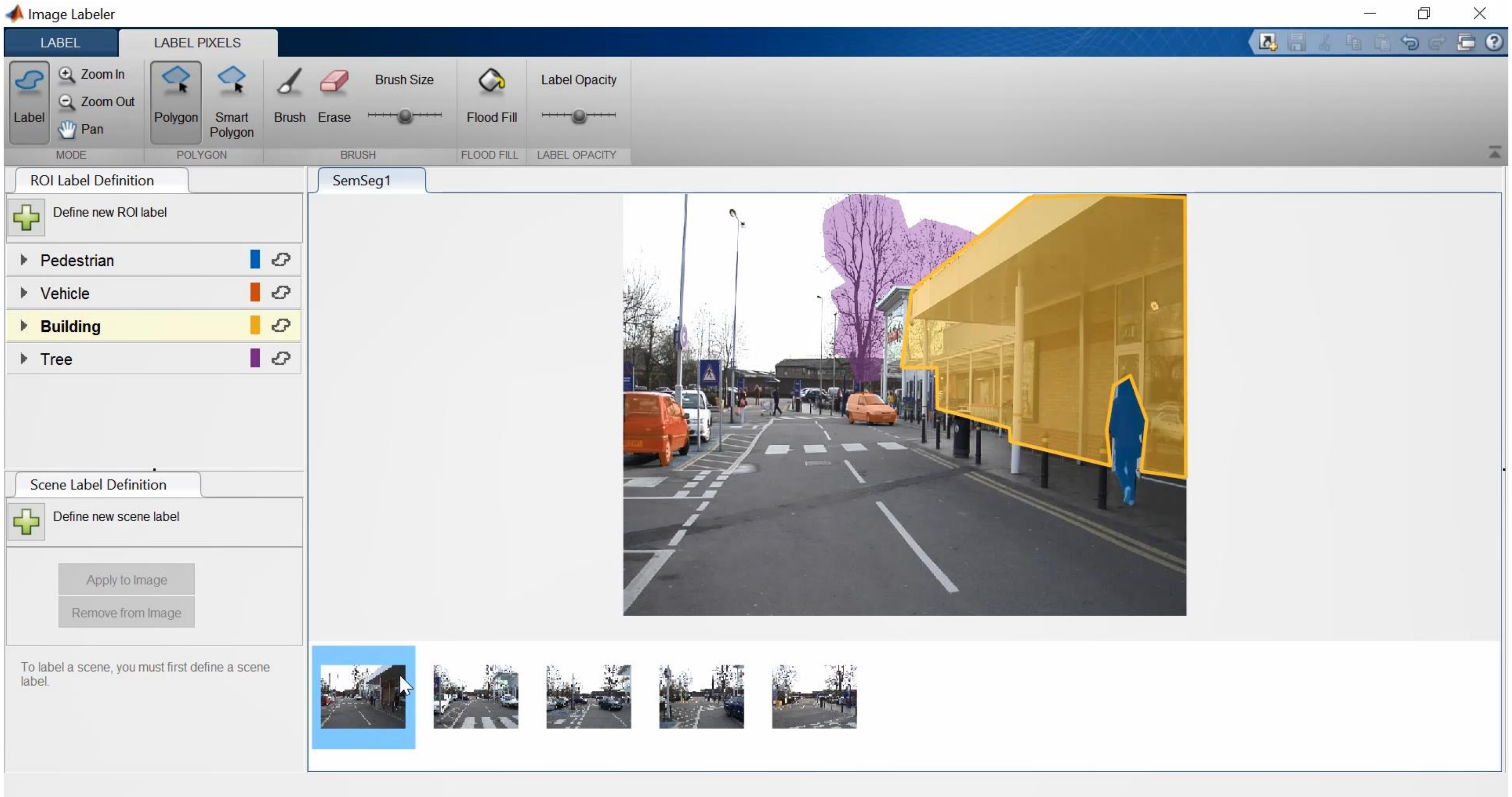
It depends (from scratch, or, Transfer Learning?)

typically **A LOT!**

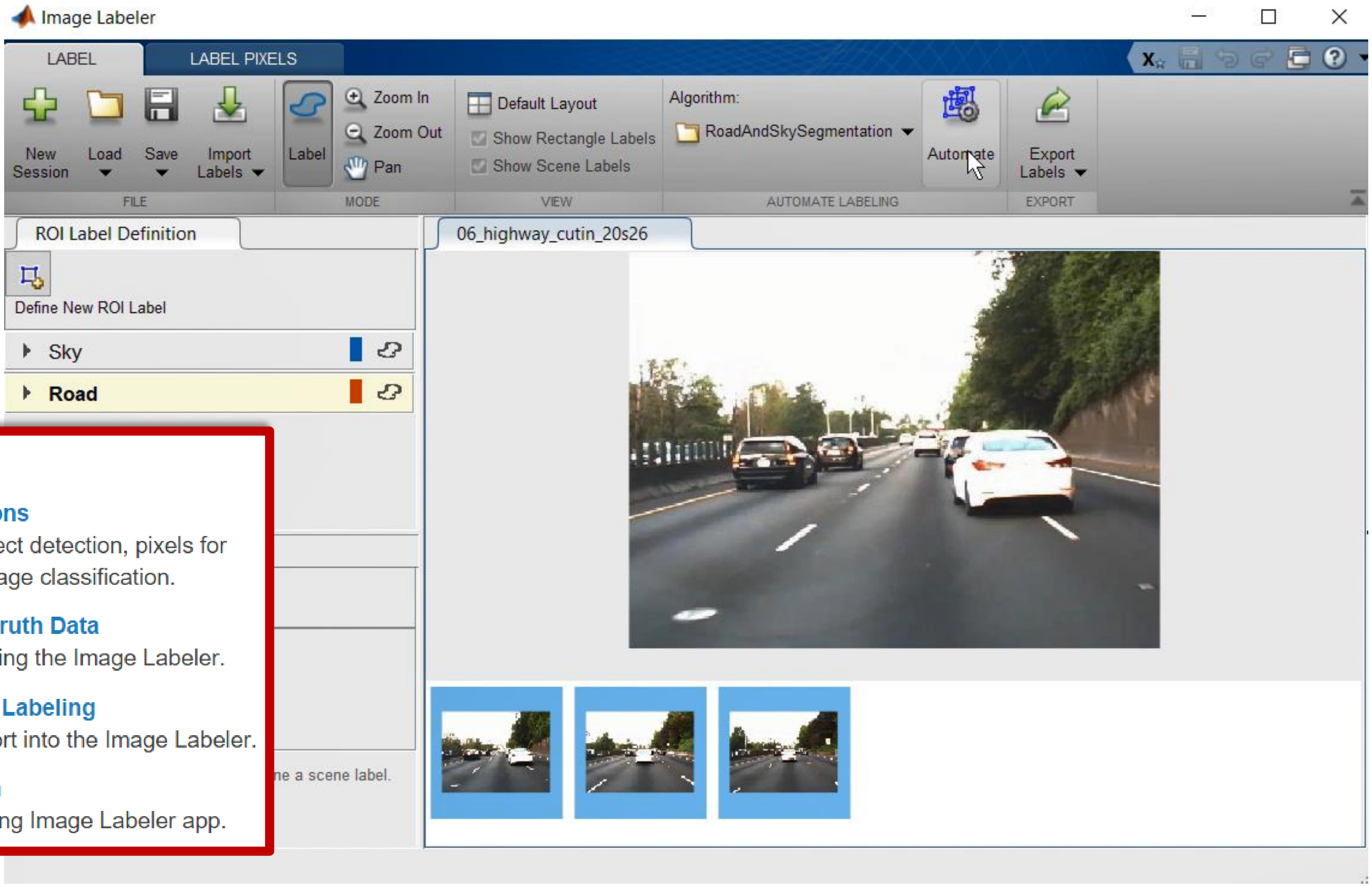


Resize
Random-crop
Rotation
Reflection
Scaling
Shearing
Translation

#2: Not just data – high quality labeled data!



#2: Accelerate Labeling With Automation Algorithms



[Learn More]

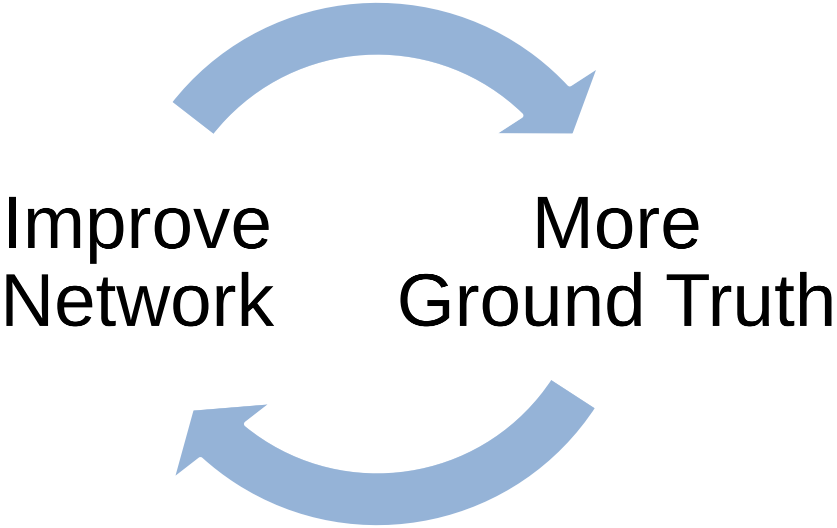
Define Ground Truth for Image Collections
Interactively label rectangular ROIs for object detection, pixels for semantic segmentation, and scenes for image classification.

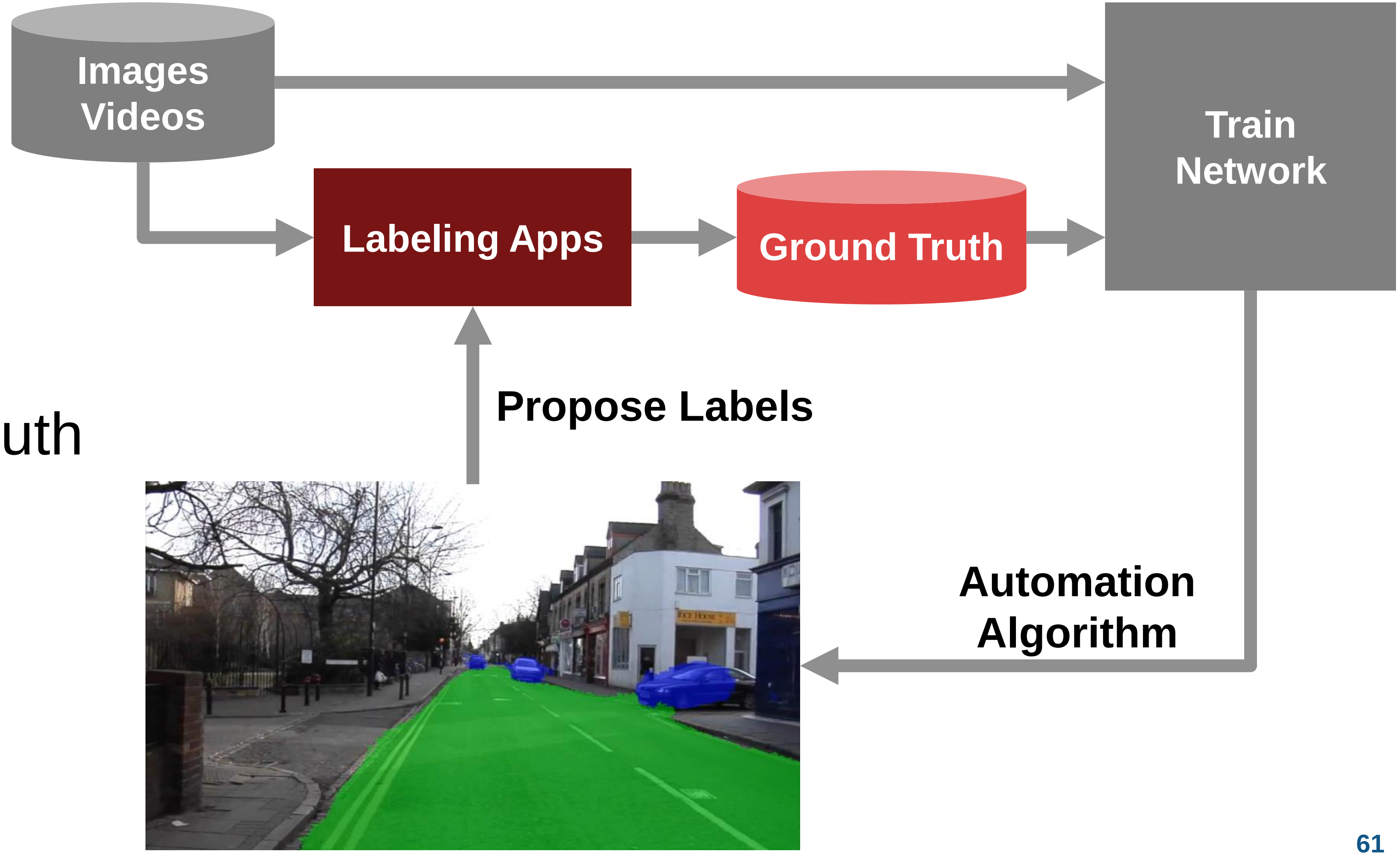
Train an Object Detector from Ground Truth Data
Create training data for object detection using the Image Labeler.

Create Automation Algorithm for Image Labeling
Create a custom tracking algorithm to import into the Image Labeler.

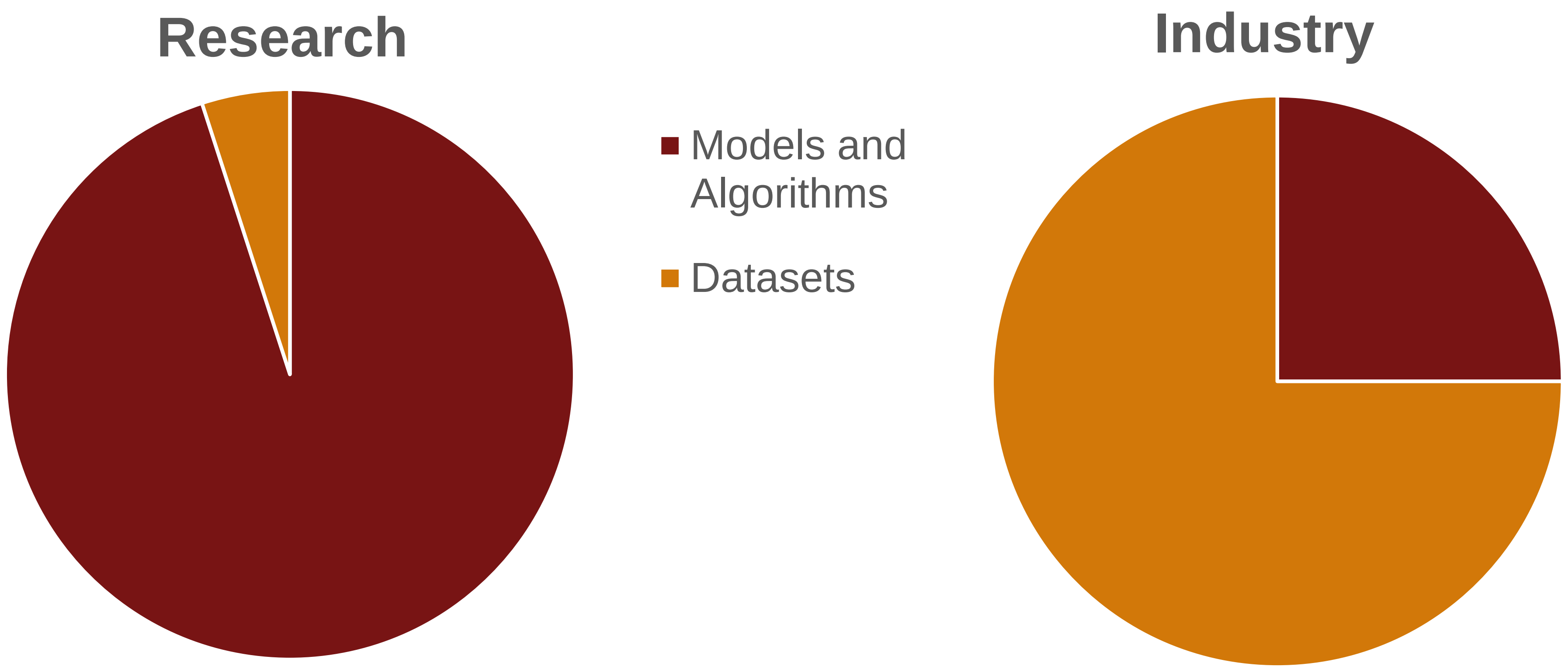
Label Pixels for Semantic Segmentation
Label pixels for semantic segmentation using Image Labeler app.

#2: Perform Bootstrapping to Label Large Datasets





Current Time/Effort Investments – Models vs. Data



From "Troubleshooting deep neural networks" (Josh Tobin et al., Jan 2019)

Top five issues we hear about while working with Deep Neural Networks

1. How do I choose a network architecture?
2. How much data do I need?
3. How do I improve accuracy of my network?
4. How do I speed up training?
5. I have a trained model – what do I do next?

#3: Improve accuracy? tune Hyperparameters and experiment!

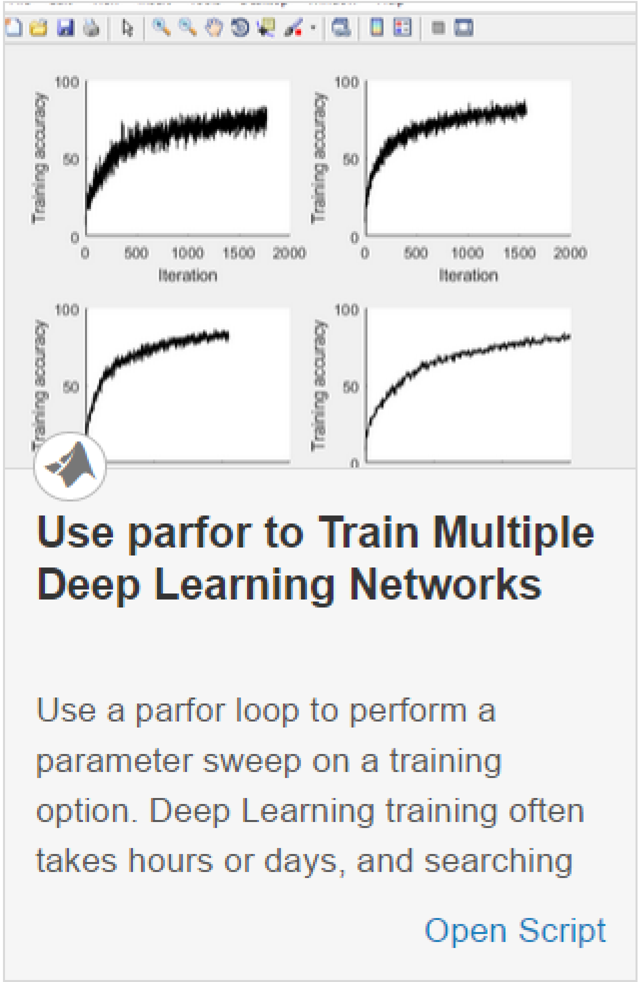
Many hyperparameters

- Network depth/architecture, solver options, learning rates + schedules, regularization, …

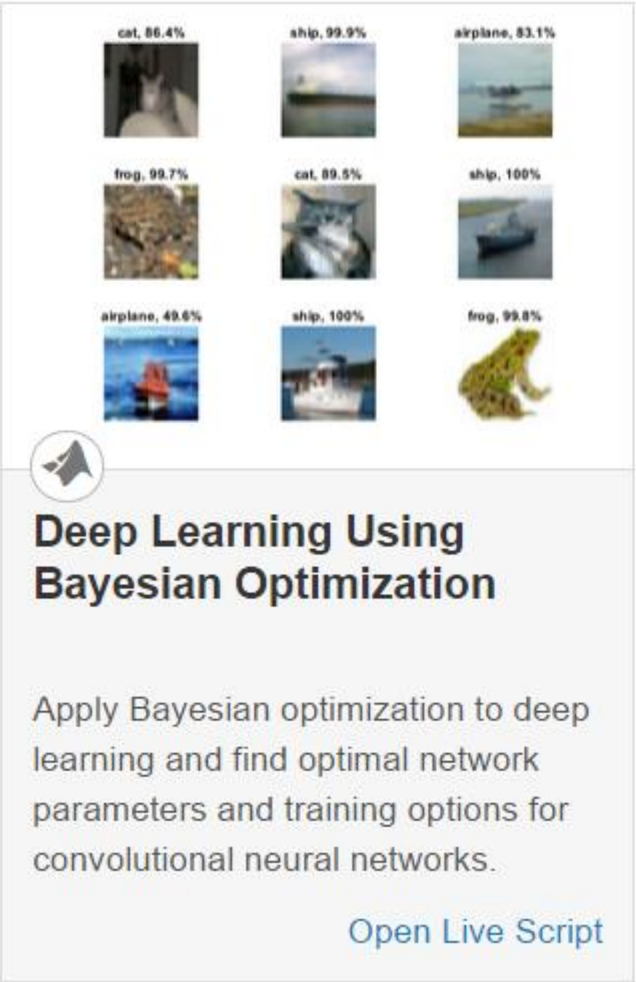
Techniques

- Parameter sweep
- Bayesian optimization

Also, use more DATA!



link



link

Top five issues we hear about while working with Deep Neural Networks

1. How do I choose a network architecture?
2. How much data do I need?
3. How do I improve accuracy of my network?
4. How do I speed up training?
5. I have a trained model – what do I do next?

#4: Speed up training using GPUs, Multi-GPUs & Clusters

How to target?

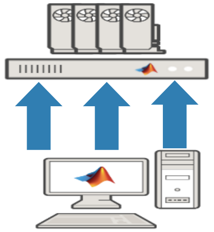
Single CPU

Single CPU Single GPU

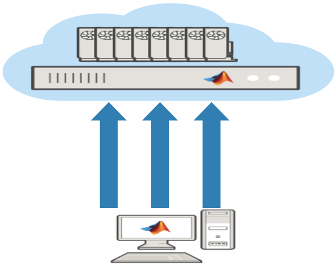
```
opts = trainingOptions('sgdm', ...
    'MaxEpochs', 100, ...
    'MiniBatchSize', 250, ...
    'InitialLearnRate', 0.00005, ...
'ExecutionEnvironment', 'auto' );
```

Single CPU, Multiple GPUs

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



On-prem server with GPUs



Cloud GPUs (AWS)

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

#4: Maybe you fancy cloud GPU instances?

MathWorks® Products Solutions Academia Support Community Events

Documentation

Search R2018b Documentation

CONTENTS

Trial Software Produc

MATLAB Deep Learning Container on NVIDIA GPU Cloud for Amazon Web Services

Speed up your deep learning applications by training neural networks in the MATLA
Learning Container remotely using a web browser or via a VNC connection.

The MATLAB Deep Learning Container contains MATLAB and a range of MATLAB t

This guide helps you run the MATLAB desktop in the cloud on an Amazon EC2® P3
hosted on NVIDIA GPU Cloud, simplifies the process. The container is available at t

Requirements

- Amazon® Web Services account
- NVIDIA GPU Cloud account with valid API key
- Valid MATLAB licenses for the products in the MATLAB Deep Learning Containe

Costs

You are responsible for the cost of the Amazon Web Services used when you create
pages for each AWS service you are using. Prices are subject to change.

Prepare Your AWS Account

If you do not have an Amazon Web Services account, create one at https://aws.ama

MathWorks® Products Solutions Academia Support Community Events

Documentation

Search R2018b Documentation

CONTENTS

Trial Software

MATLAB Deep Learning Container on NVIDIA GPU Cloud for NVIDIA DGX

Speed up your deep learning applications by training neural networks in the MATLAB® Deep Learning Container, designed to take full advantage of high-performance NVIDI
Learning Container remotely using a web browser or via a VNC connection.

The MATLAB Deep Learning Container contains MATLAB and a range of MATLAB toolboxes that are ideal for deep learning (see Additional Information).

This guide helps you run the MATLAB desktop in the cloud on NVIDIA DGX platforms. The MATLAB Deep Learning Container, a Docker container hosted on NVIDIA GPU C
available at the NVIDIA GPU Cloud Container Registry.

Requirements

- Host DGX system with Docker and NVIDIA-Docker installed. For installation instructions, see https://docs.nvidia.com/deeplearning/dgx/preparing-containers/index.html#p
- NVIDIA GPU Cloud account with valid API key.
- Valid MATLAB licenses for the products in the MATLAB Deep Learning Container (see Licensing for details).

Pull the Container

Pulling the container downloads the container image onto the Docker host, the machine that runs the container. You have to pull the container only once.

You can copy the pull command for the container image release from the NVIDIA Container Registry. In the Tags section, locate the container image release that you want t
the `docker pull` command. The command is of the form:

```
docker pull nvcr.io/partners/matlab:r2018b
```

Ensure the last part of the `pull` command matches the MATLAB release you want to use.

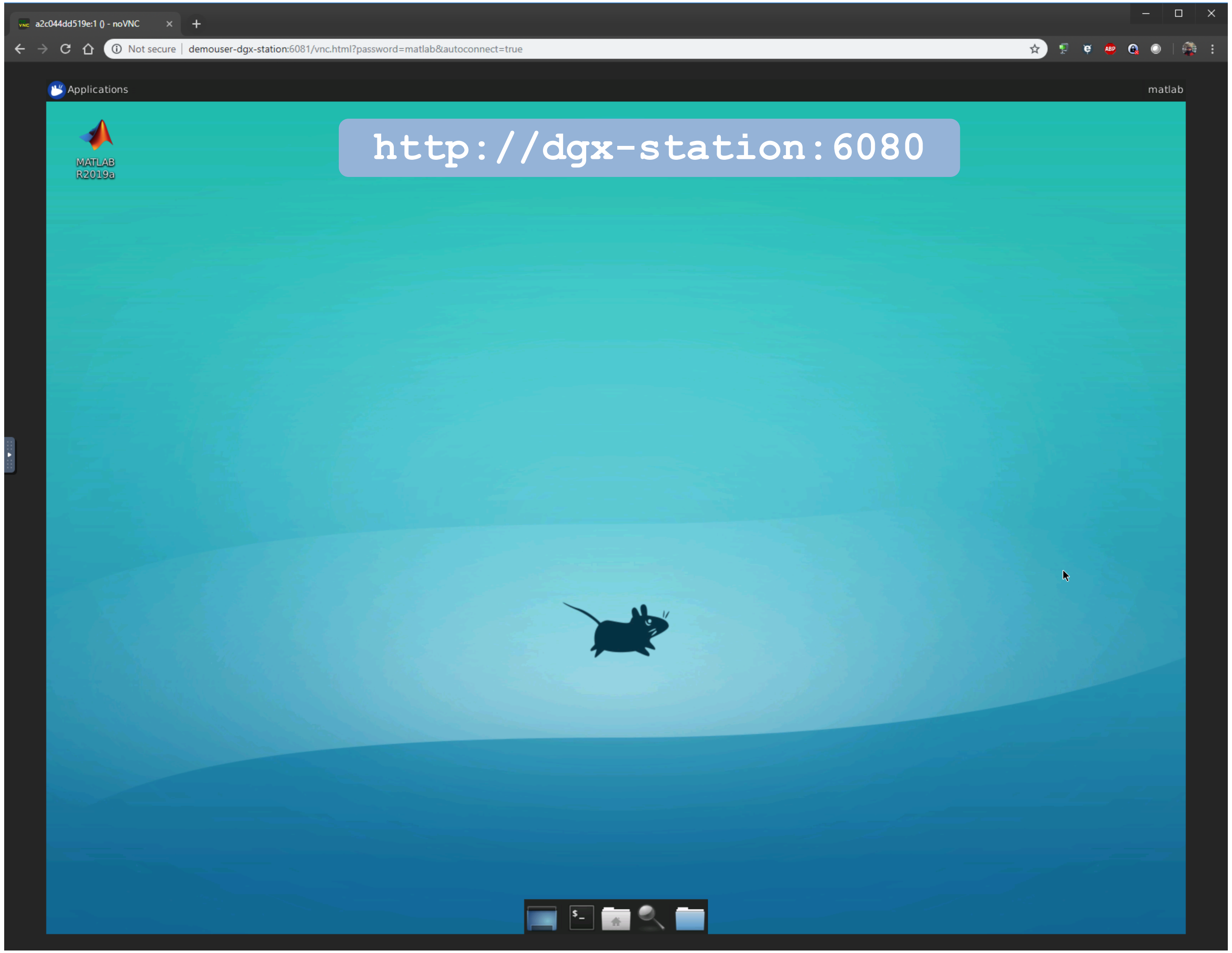
Connect to the Docker host via SSH from your client machine using PuTTY or another SSH client. On the host, log in to the NVIDIA Container Registry using this command:

Scaling up NVIDIA DGX

```
docker login nvcr.io

docker pull nvcr.io/partners/matlab:r2019a

rsync -rave ssh /data/genres/ dgx:/tmp/genres

nvidia-docker run -it --rm -p 6080:6080 \
   --shm-size=512M \
   -v /tmp/genres/:/data \
   nvcr.io/partners/matlab:r2019a
```

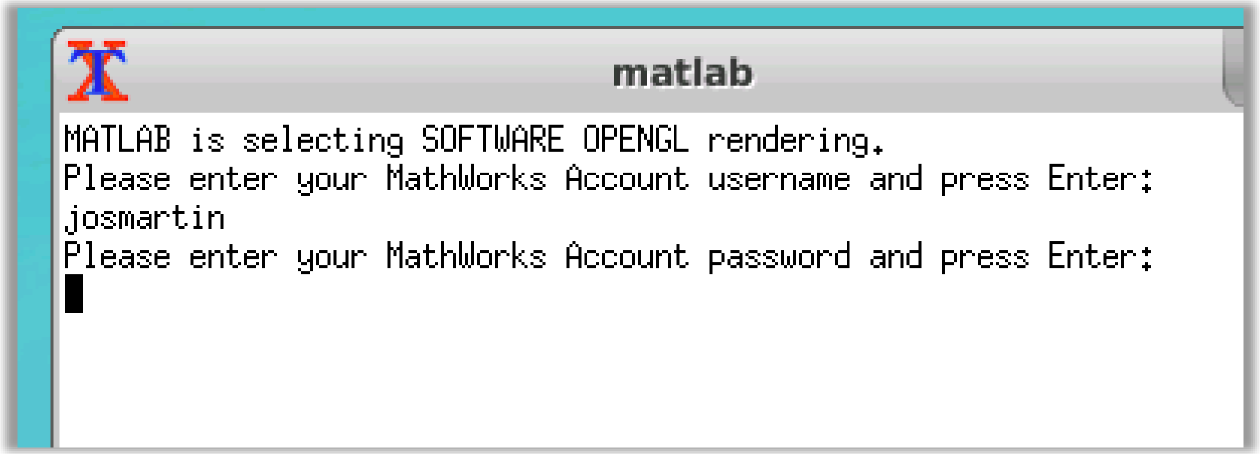
“We’re Gonna Need a Bigger Machine”

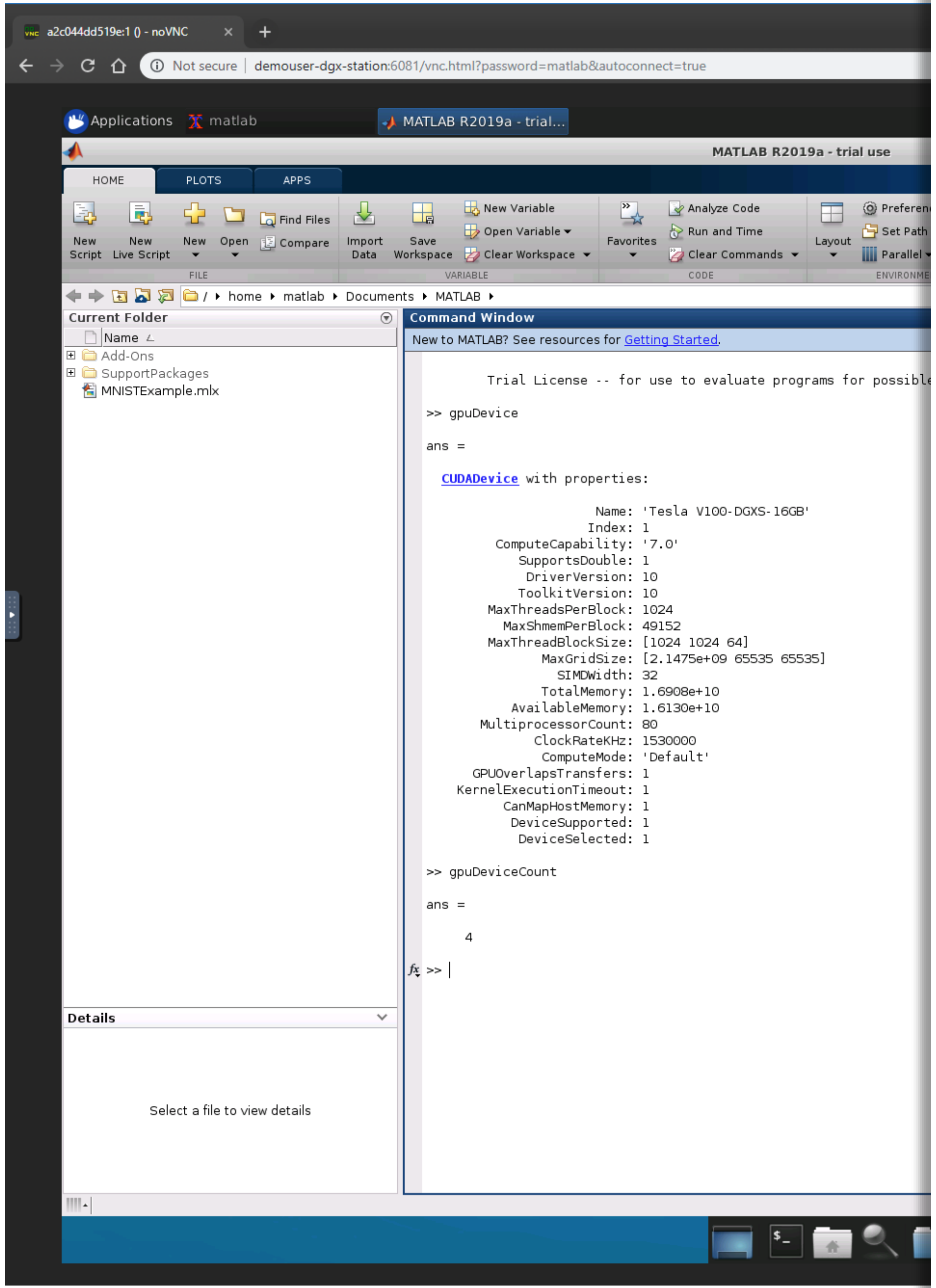
- Desktop access using browser
- Or VNC
- Or at the docker command prompt



Getting Started

- Start MATLAB & login






```
>> gpuDevice

ans =

  CUDADevice with properties:

                      Name: 'Tesla V100-DGXS-16GB'
                     Index: 1
         ComputeCapability: '7.0'
            SupportsDouble: 1
             DriverVersion: 10
            ToolkitVersion: 10
        MaxThreadsPerBlock: 1024
          MaxShmemPerBlock: 49152
        MaxThreadBlockSize: [1024 1024 64]
               MaxGridSize: [2.1475e+09 65535 65535]
                 SIMDWidth: 32
               TotalMemory: 1.6908e+10
           AvailableMemory: 1.6130e+10
       MultiprocessorCount: 80
              ClockRateKHz: 1530000
               ComputeMode: 'Default'
      GPUOverlapsTransfers: 1
    KernelExecutionTimeout: 1
          CanMapHostMemory: 1
           DeviceSupported: 1
            DeviceSelected: 1

>> gpuDeviceCount

ans =

     4
```

Top five issues we hear about while working with Deep Neural Networks

1. How do I choose a network architecture?
2. How much data do I need?
3. How do I improve accuracy of my network?
4. How do I speed up training?
5. I have a trained model – what do I do next?

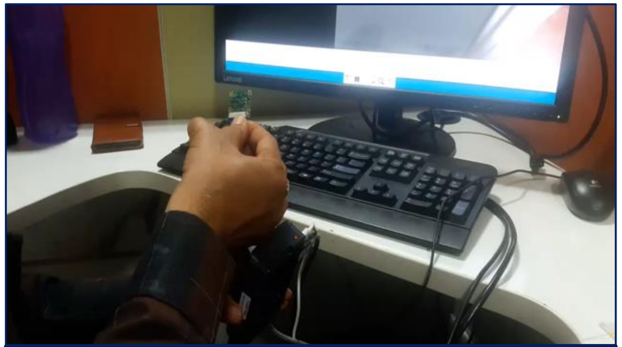
#5: Deploy on platform of choice

- Need code that takes advantage of:
  - NVIDIA® CUDA libraries, including cuDNN and TensorRT
  - Intel® Math Kernel Library for Deep Neural Networks (MKL-DNN) for Intel processors
  - ARM® Compute library for ARM processors

NVIDIA Jetson TX1 board

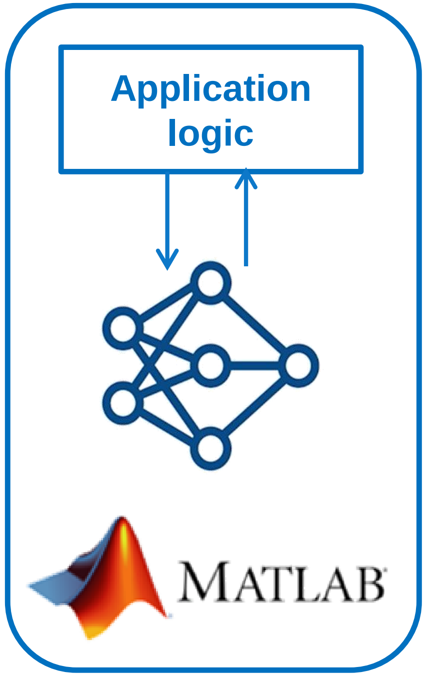
Android Phone

Intel Xeon Desktop PC



Raspberry Pi Board

#5: Easy deployment using a single network representation



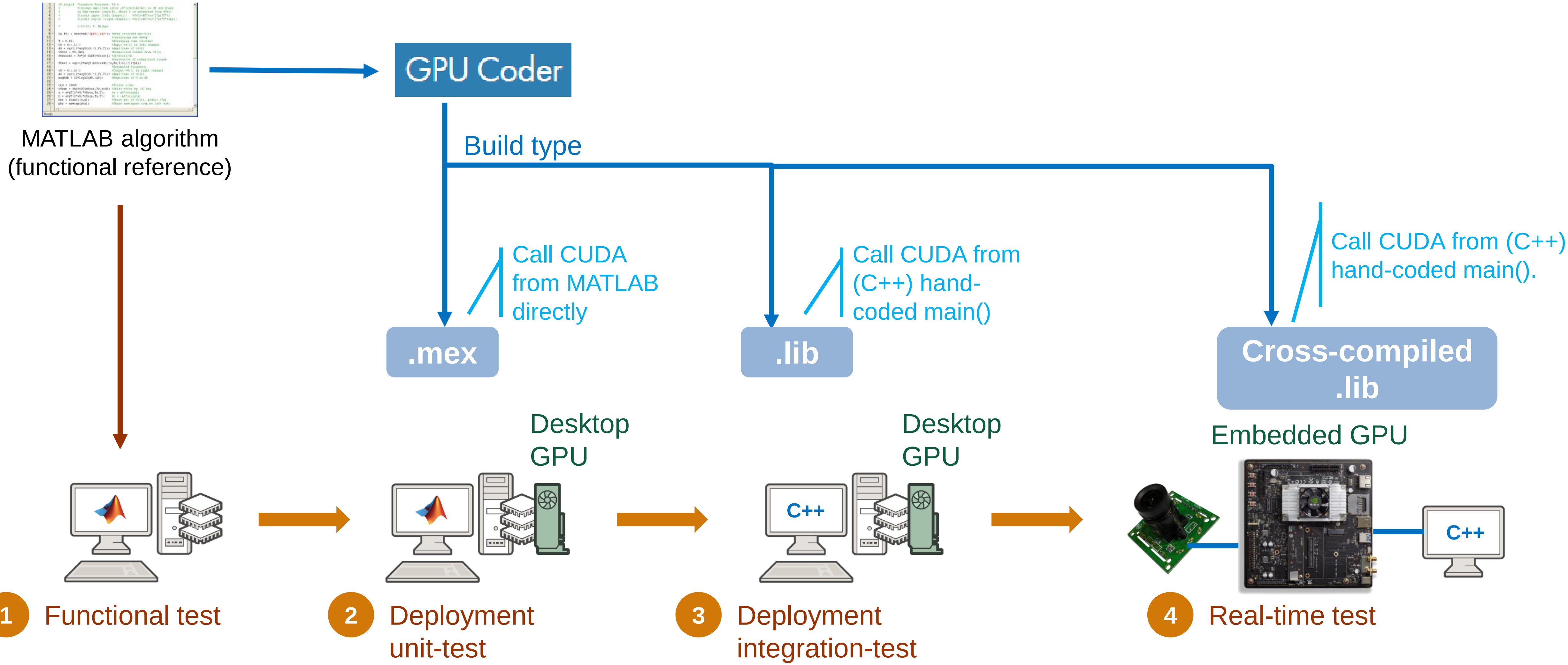
GPU Coder → NVIDIA: **TensorRT & cuDNN Libraries**

MATLAB Coder → intel: **Intel MKL-DNN Library**

MATLAB Coder → ARM NEON™: **ARM Compute Library**

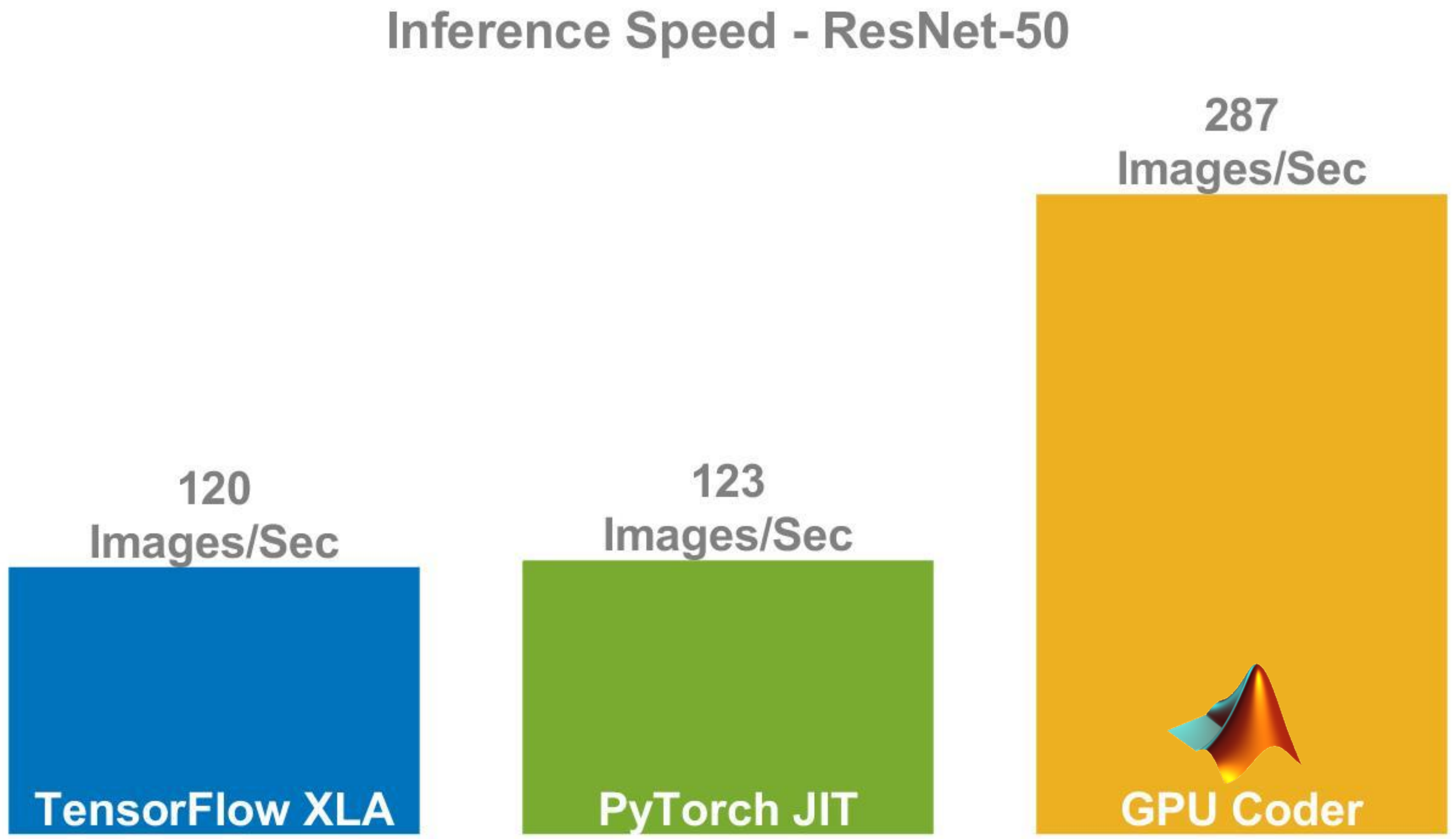
Target Libraries

Summary- *GPU Coder*



GPU Coder more than twice as fast as other frameworks

R2019b

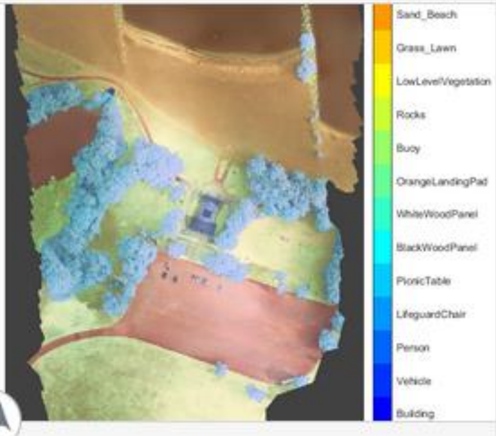


Intel® Xeon® CPU 3.6 GHz – Titan V. NVIDIA libraries: CUDA10.0/1 - cuDNN 7.4/5 - Frameworks: TensorFlow 1.13, MXNet 1.4.1 PyTorch 1.1.0

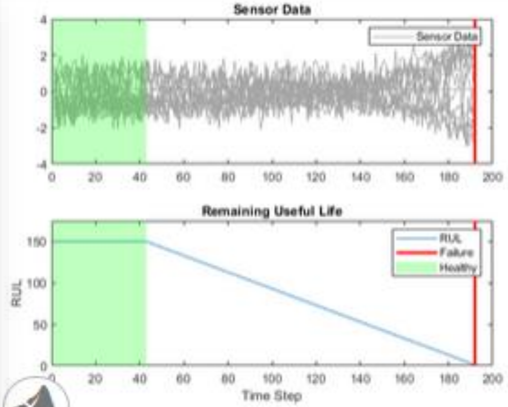
Deep Learning is Versatile

MATLAB Examples Available Here

Object Detection Using Faster R-CNN Deep Learning



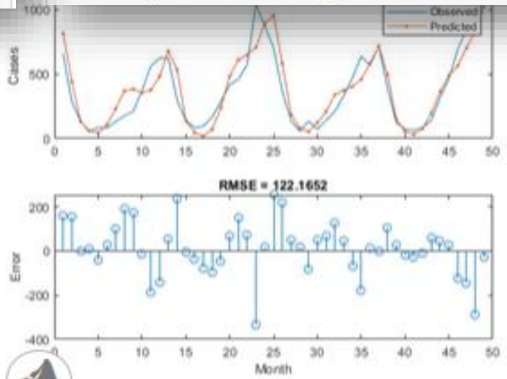
Semantic Segmentation of Multispectral Images Using Deep Learning



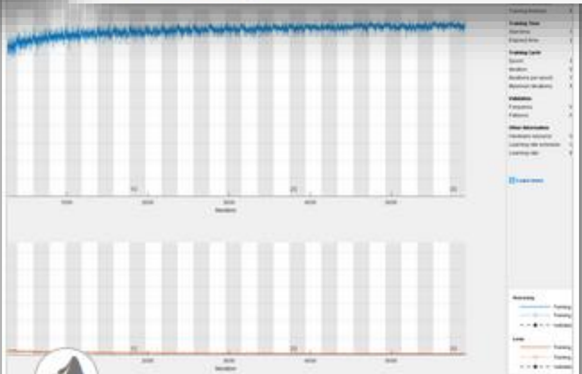
Sequence-to-Sequence Regression Using Deep Learning

Single Image Super-Resolution Using Deep Learning

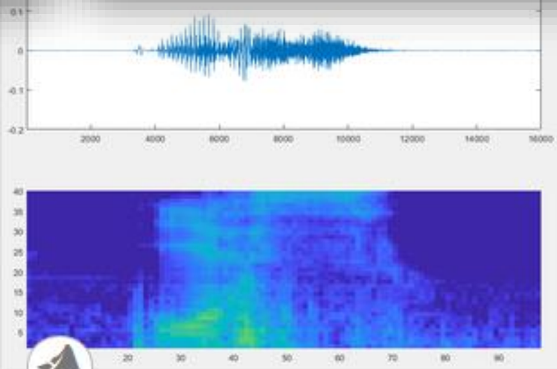
Classify Image Using GoogLeNet



Time Series Forecasting Using Deep Learning



Classify Text Data Using Deep Learning



Deep Learning Speech Recognition

3 Takeaways and Questions?

- Deep Learning poses several challenges regarding data, pre-processing, and training processes.
- Transfer learning needs relatively lesser data, and also possibly lower training time, but has trade-offs.
- MATLAB also supports several hyperparameter tuning methods to optimize the training process.
- **MATLAB <3 Deep Learning**

Next Steps

https://matlabacademy.mathworks.com

Machine Learning Onramp

Learn the basics of practical machine learning methods for classification problems.

Launch | Details

Deep Learning Onramp

Get started quickly using deep learning methods to perform image recognition.

Launch | Details

Machine Learning with MATLAB

Explore data and build predictive models.

Launch | Details

Deep Learning with MATLAB

Learn the theory and practice of building deep neural networks with real-life image and sequence data.

Launch | Details

MathWorks®

Accelerating the pace of engineering and science

Feedback form

Dr. Spandhana Gonuguntla, +91 7483479945

Spandhana Gonuguntla, PhD
Education Technical Evangelist
sgonugun@mathworks.com

https://tinyurl.com/u3tuyop