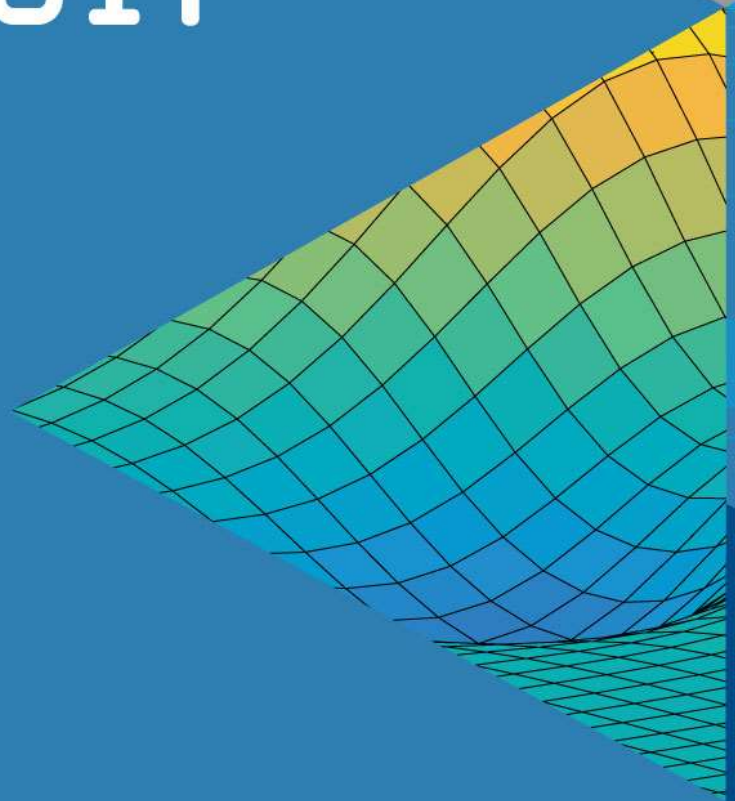
MATLAB EXPO 2017 KOREA

4월 27일, 서울

등록 하기 matlabexpo.co.kr

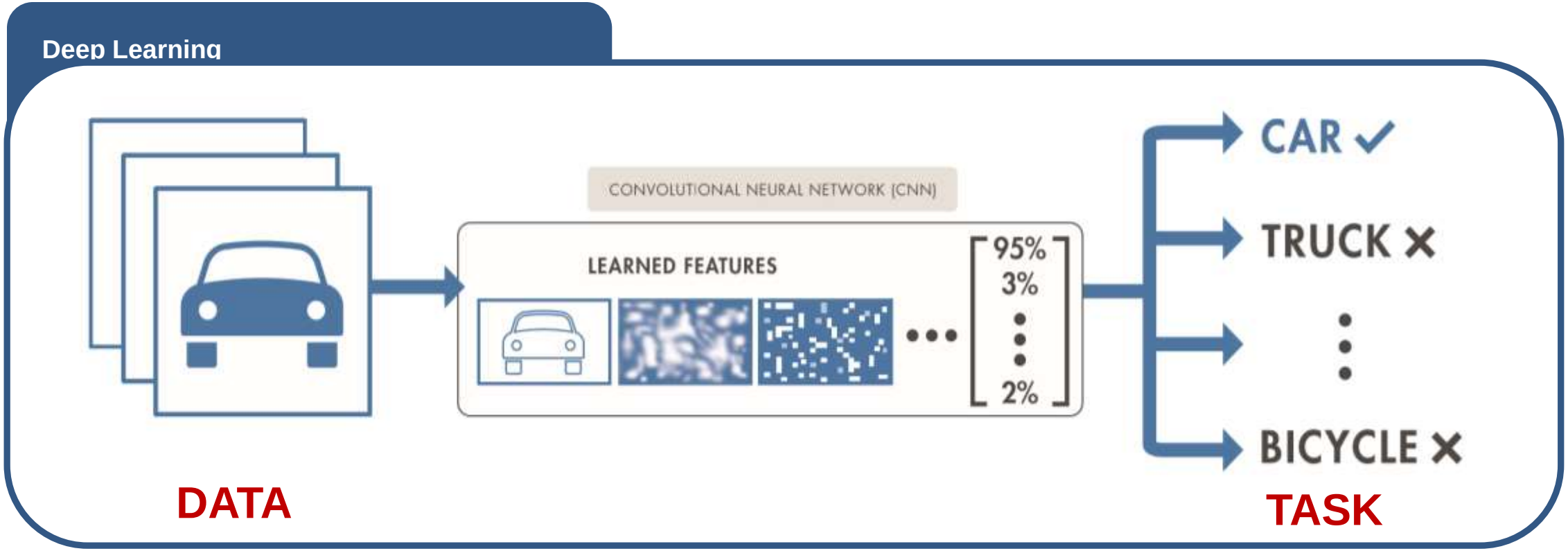


딥러닝 기반 응용 프로그램 작성 기법

Application Engineer
엄준상 과장

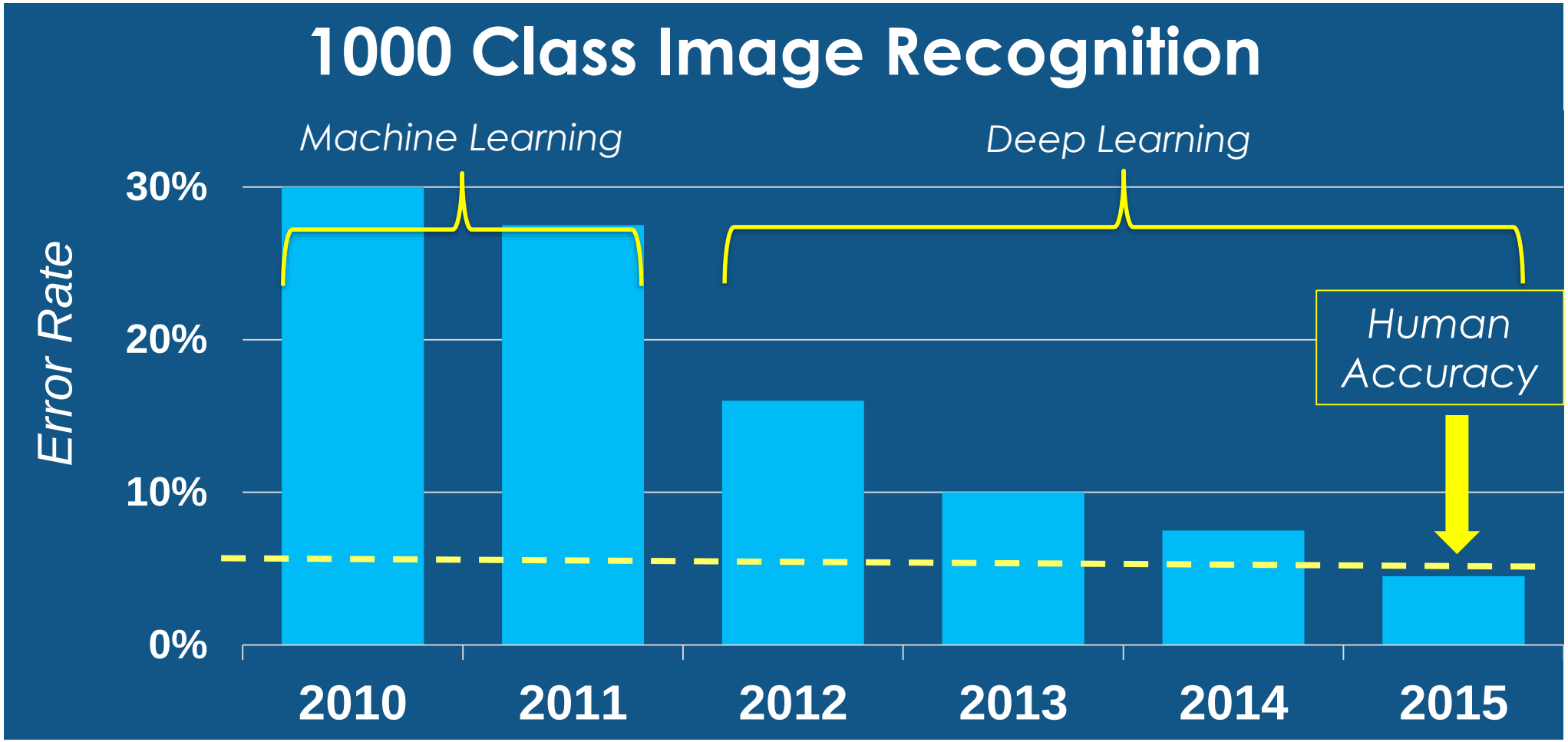
What is Deep Learning ?

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



Why is Deep Learning So Popular Now?

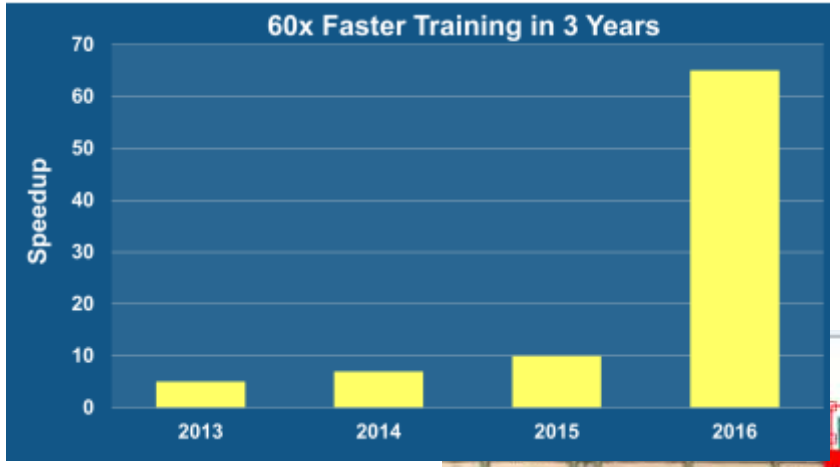
Unparalleled Accuracy



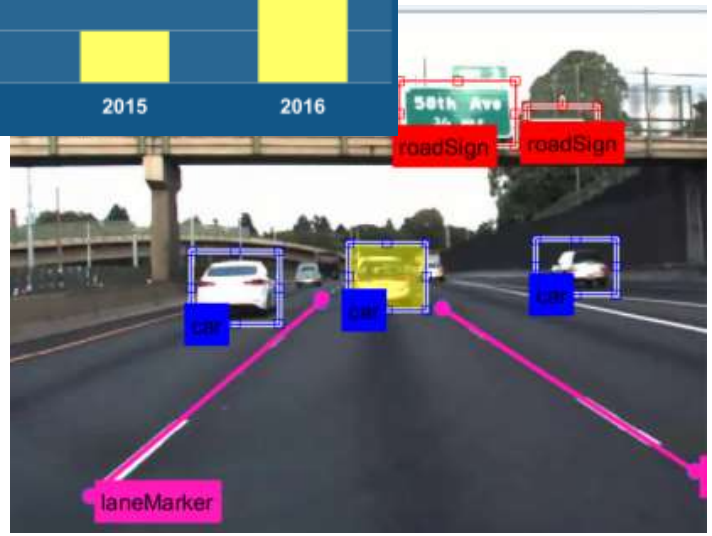
Source: ILSVRC Top-5 Error on ImageNet

Deep Learning Enablers

Acceleration with GPU's



Massive sets of labeled data



Availability of state of the art models from experts

AlexNet
PRETRAINED MODEL
VGG-16
PRETRAINED MODEL
Caffe
MODELS

Deep Learning Workflow

Leverages MATLAB Platform Strengths

Access & Explore

Training Data
- Large open datasets
- Recorded labeled data (e.g., images, video)

Pre-Trained Models
- Access state-of-art networks already trained on large datasets

Preprocess Data

Data Augmentation
- Crop, resize and rotate images – create more training data

Label Training Data
- For image data: draw ROI's, label individual pixels with labels

Develop Predictive Models

Train from Scratch
- Configure and train a deep network with massive amount of training data

Transfer Learning
- Tune pre-trained model for a different task with smaller datasets

Integrate Analytics to Systems

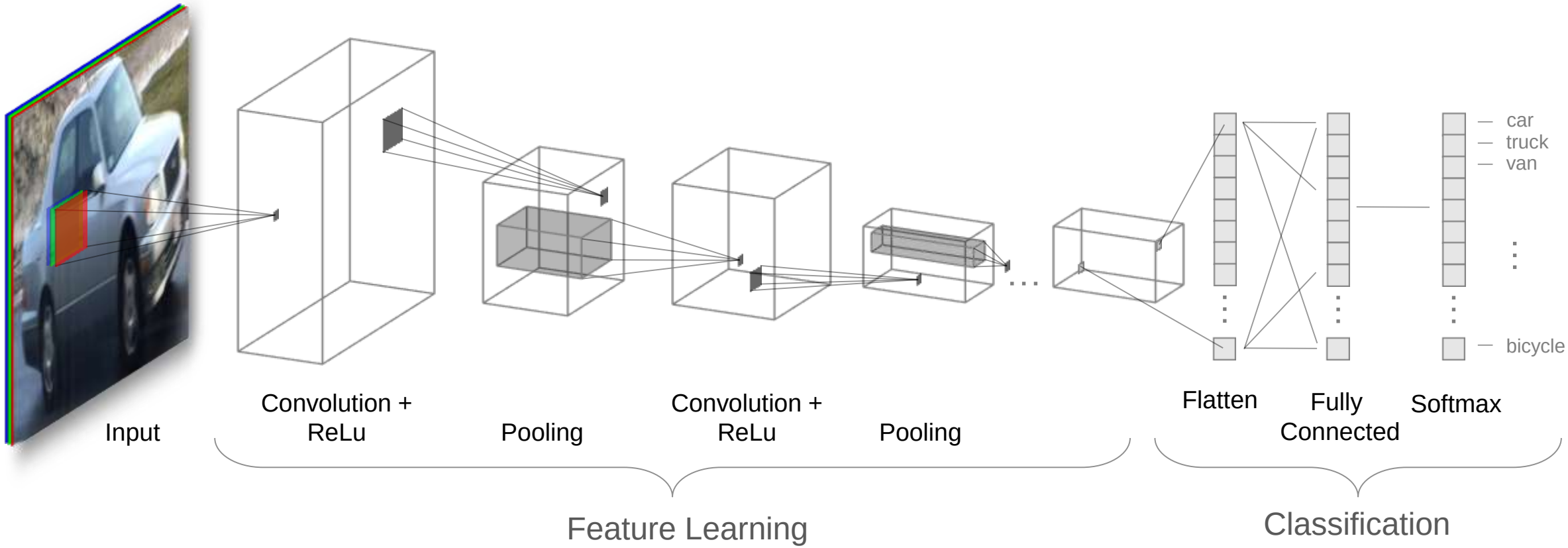
Share Models
- Publish models for others to use

Embedded Deployment
- Embedded processors
- FPGA

HPC
- Servers (multi-GPU)
- Clusters

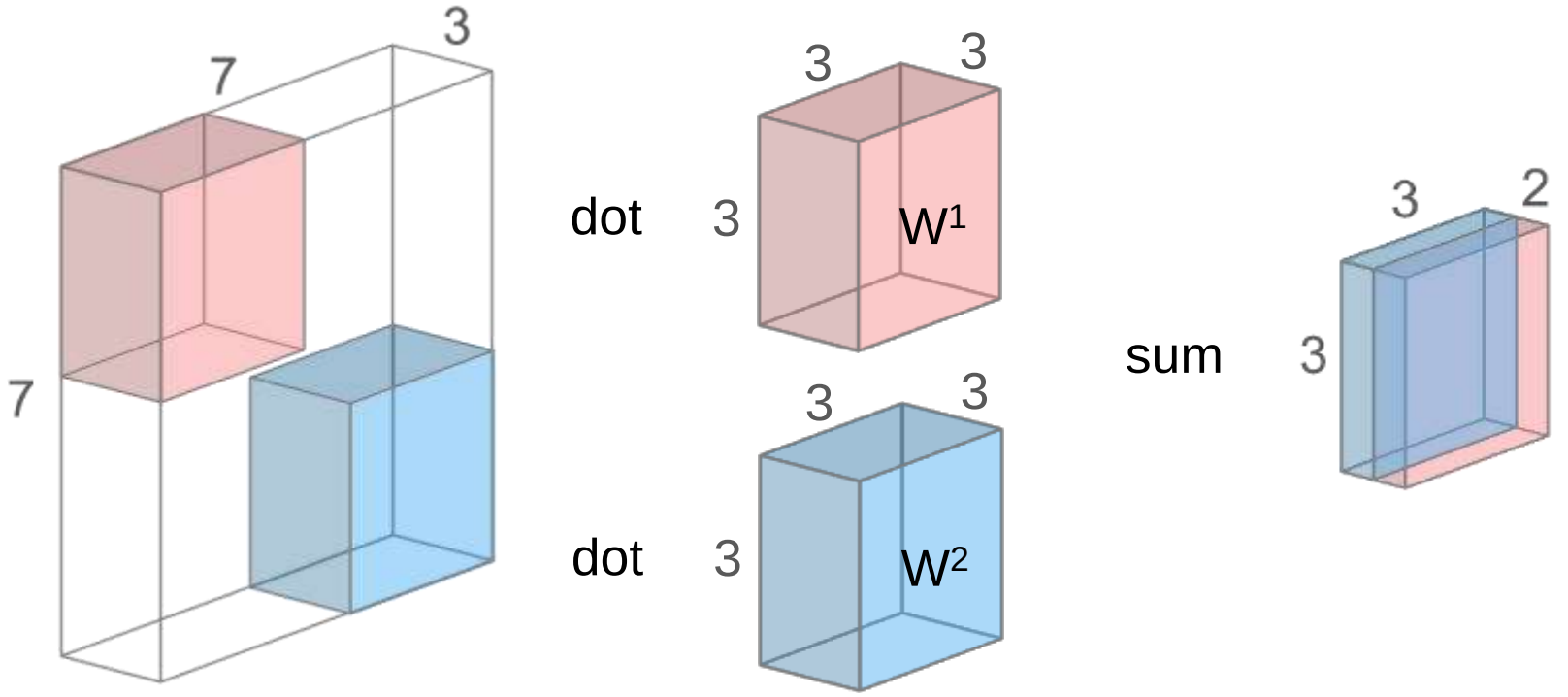
Convolutional Neural Networks

- Train “deep” neural networks on structured data (e.g. images, signals, text)
- Implements Feature Learning: Eliminates need for “hand crafted” features
- Trained using GPUs for performance



Convolution Layer

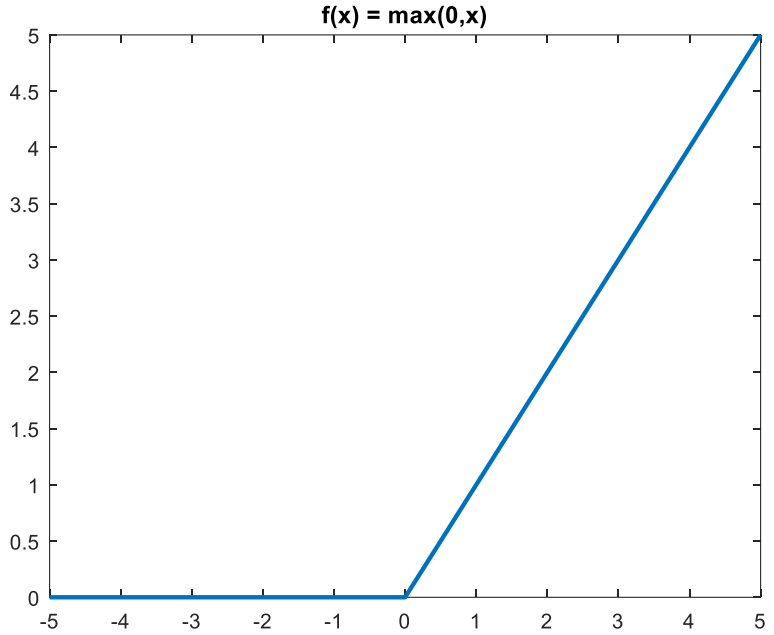
- Core building block of a CNN
- Convolve the filters sliding them across the input, computing the dot product



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

Rectified Linear Unit (ReLU) Layer

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



Pooling Layer

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

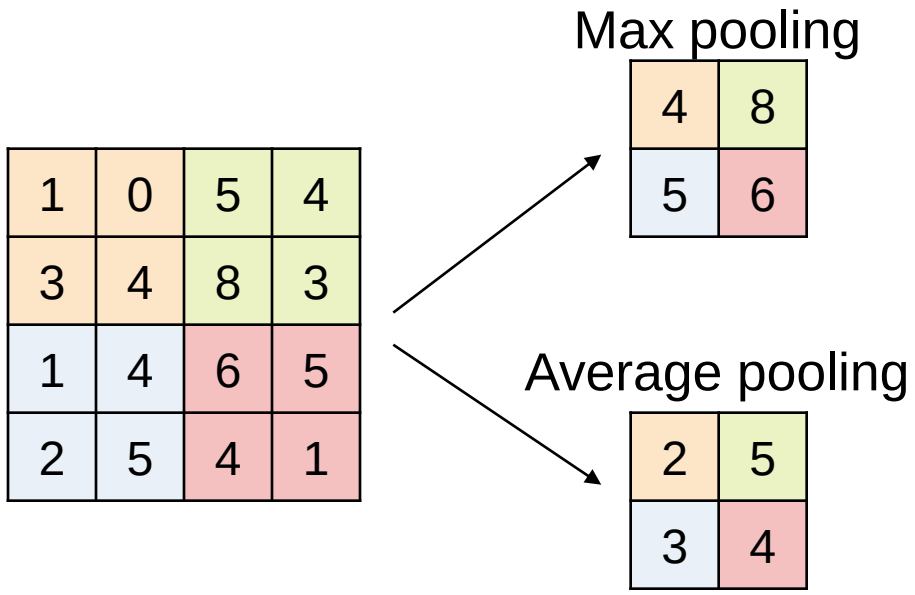
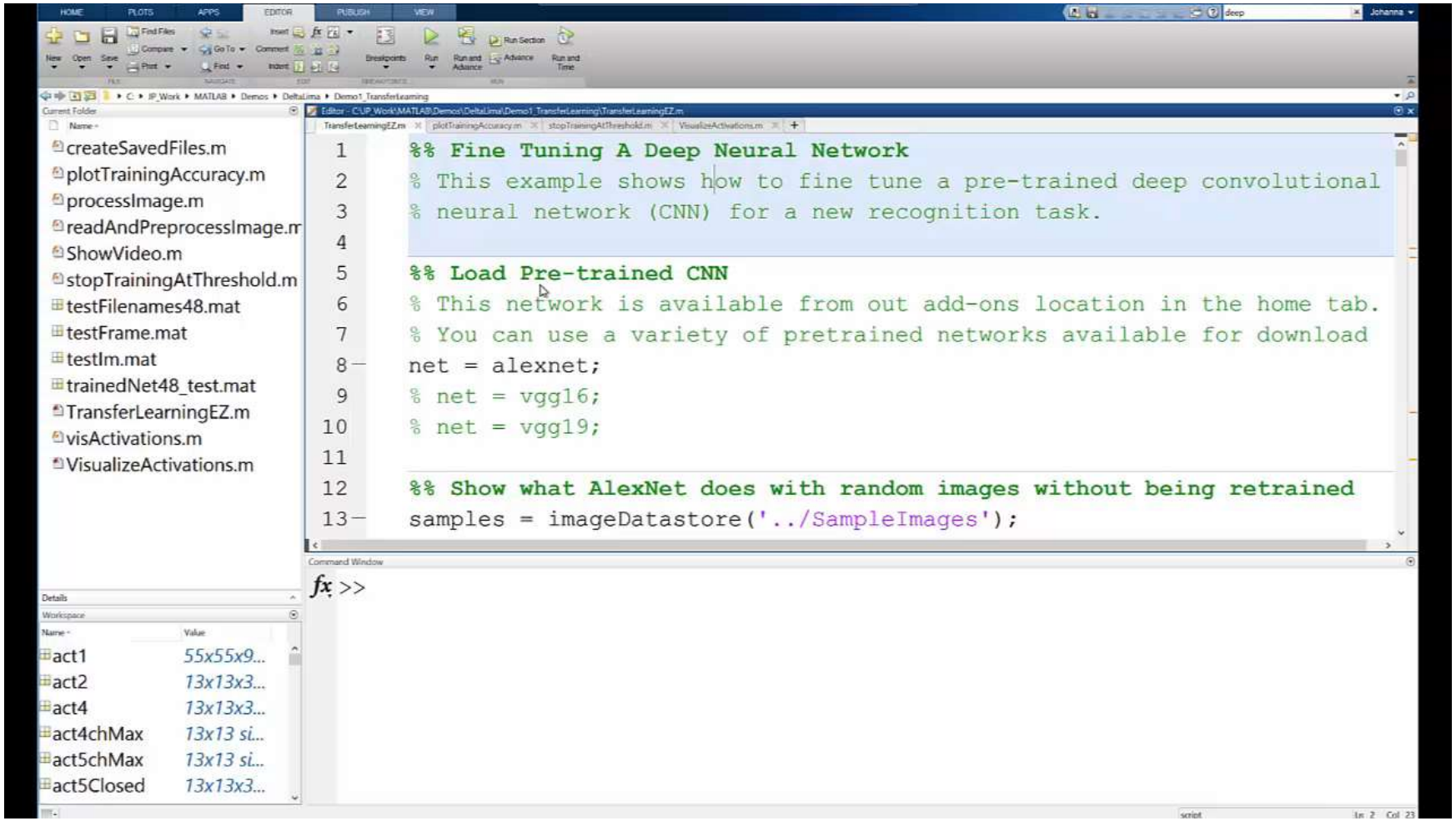
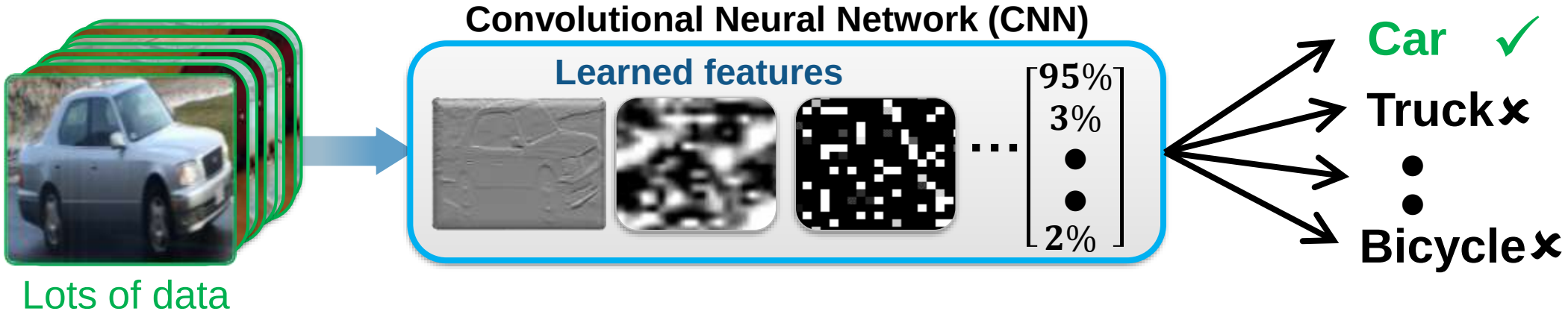


Image Classification Using Pre-trained Network (Video)



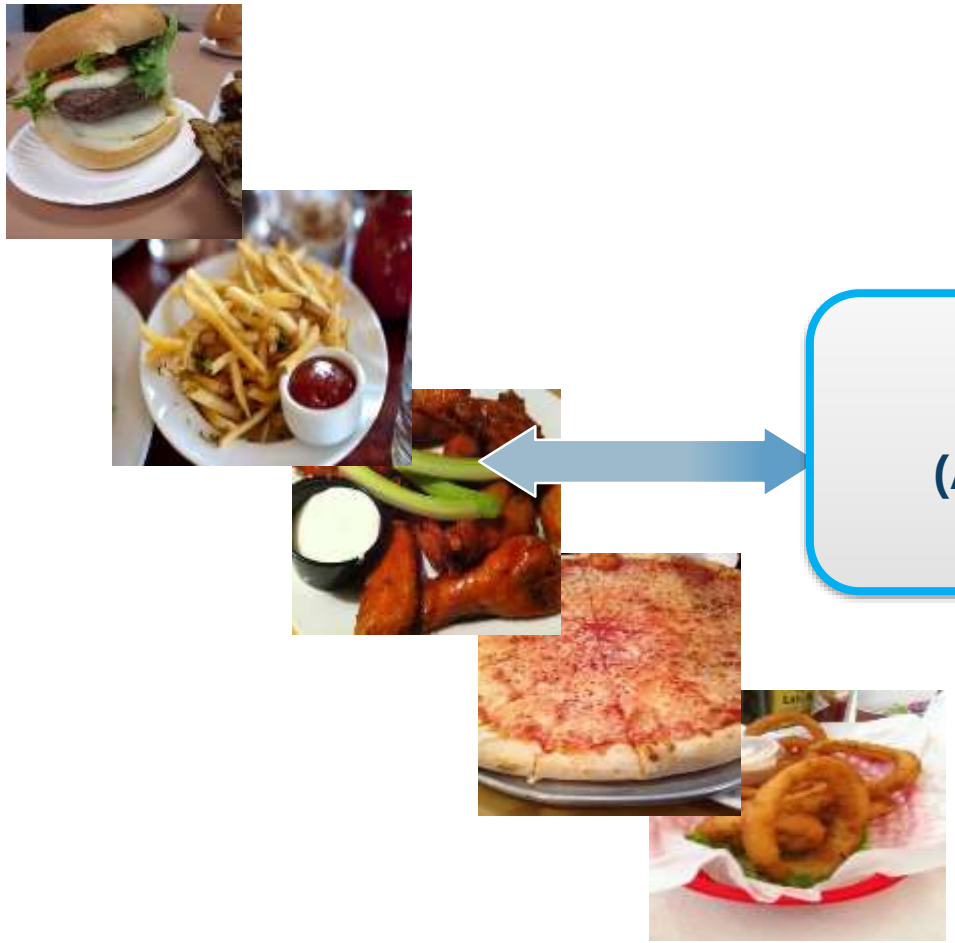
Approaches for Deep Learning

1. Train a Deep Neural Network from Scratch



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

Example: Fine-tune a pre-trained model (Transfer learning)



Pre-trained CNN
(AlexNet – 1000 Classes)

Burger

Fries

Wings

Pizza

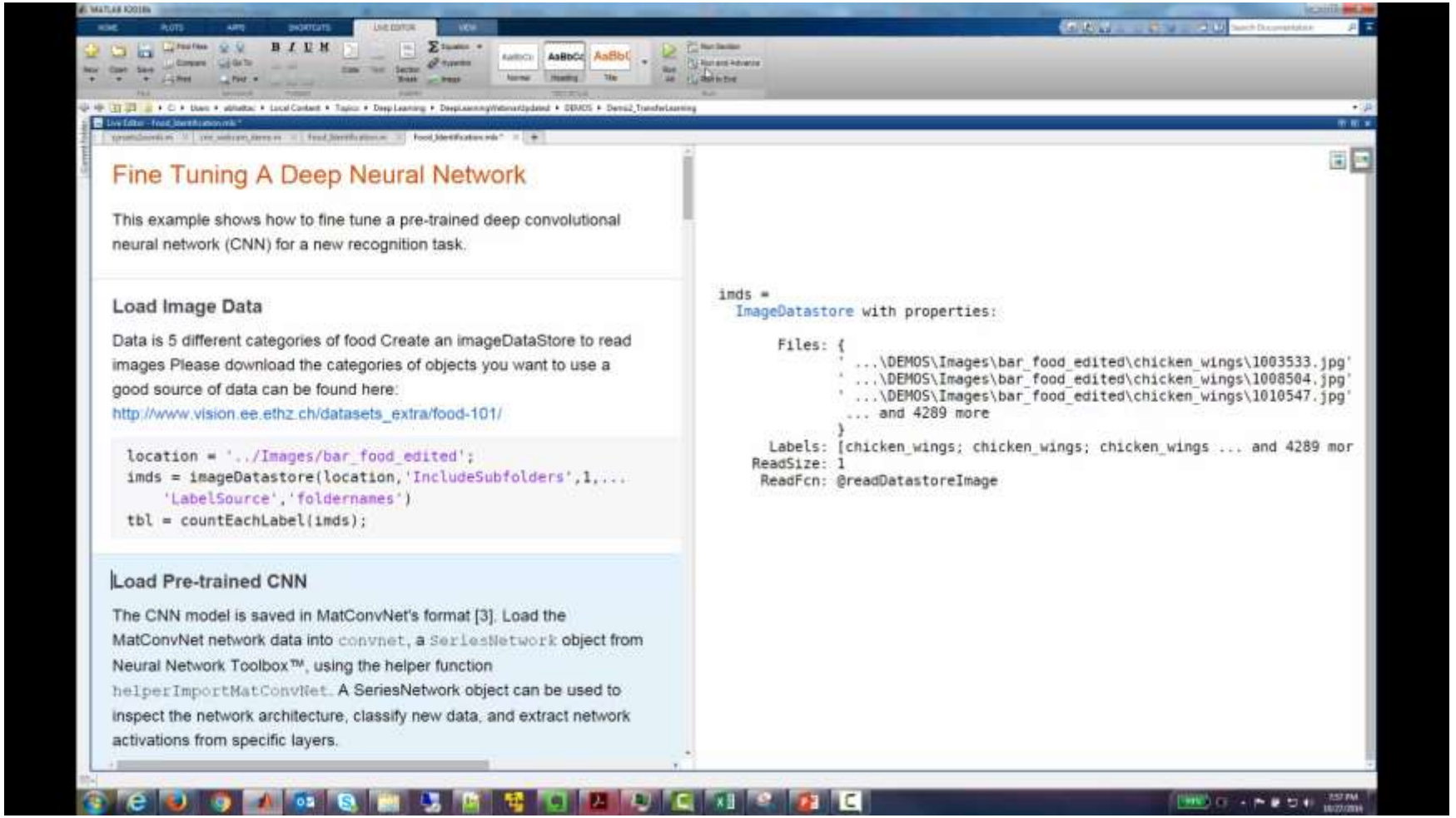
Onion Rings

New Task – 5 Class Classification

Why Perform Transfer Learning

- Requires less data and training time
- Reference models (like AlexNet, VGG-16. VGG-19) are great feature extrac tors
- Leverage best network types from top researchers

Transfer Learning in MATLAB



Manipulate Deep Learning Networks Easily

Perform net surgery

Modify the existing network by deleting later layers and adding new ones.

```
% Here we only need to keep everything except the last 3 layers.
layers = net.Layers(1:end-3);

% Add new fully connected layer for 2 categories.
layers(end+1) = fullyConnectedLayer(64, 'Name', 'special 2');
layers(end+1) = reluLayer;

% the new layer adding non-linearity and improves the network's ability to handle data
layers(end+1) = fullyConnectedLayer(height(tbl), 'Name', 'fc8_2');

% Add the softmax layer and the classification layer
layers(end+1) = softmaxLayer;
layers(end+1) = classificationLayer()
```

DELETE LAYERS

ADD LAYERS

Manipulate Deep Learning Networks Easily

Set options for training

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

Train the network

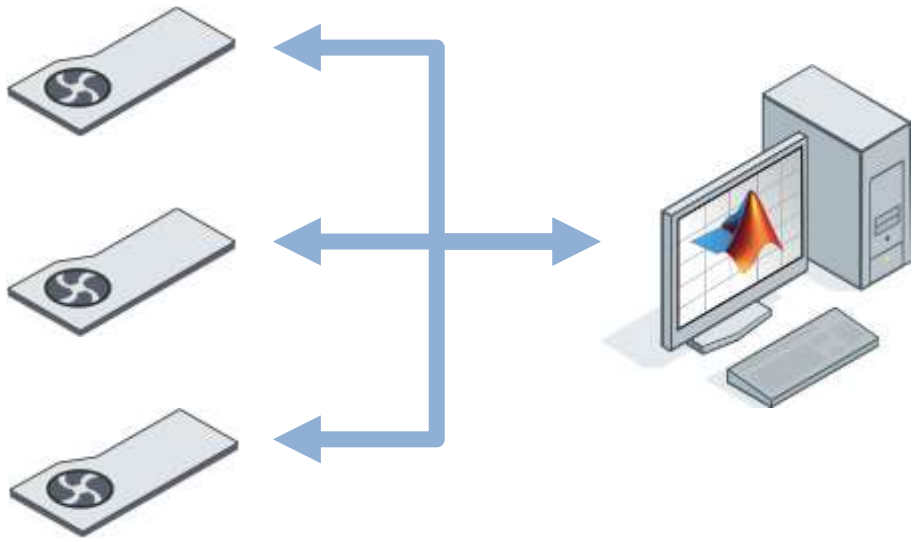
```
net = trainNetwork(imds, layers, opts);
```

Make predictions

```
label = classify(net, im);
```

Extract features

```
features = activations(net, Xtrain, 'fc7');
```



MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

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

```
imageDS = imageDatastore(dir)
```

Easily manage large sets of images

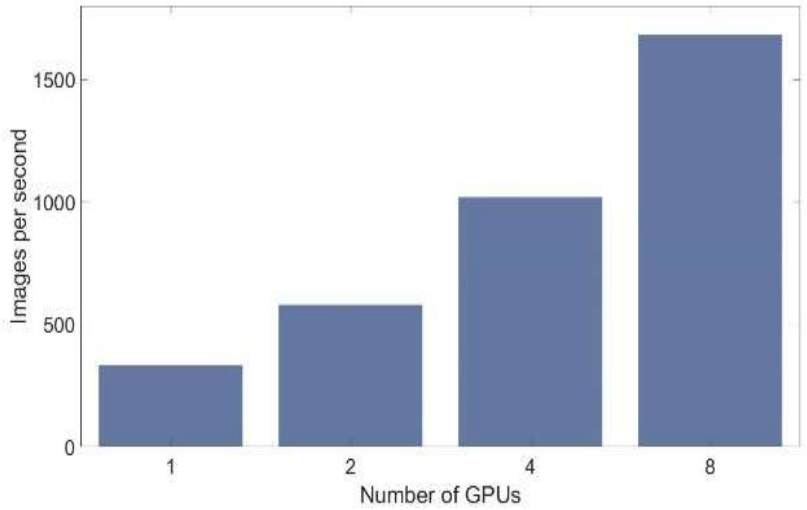
MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

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

Training modes supported:

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

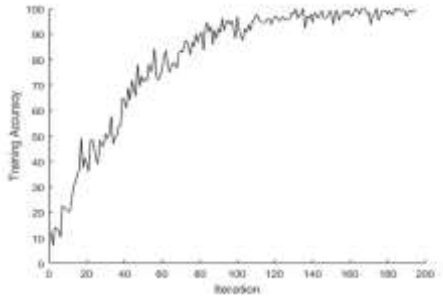


Acceleration with Multiple GPUs

MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

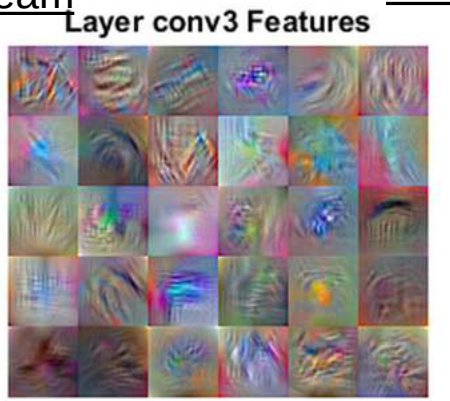
- Handle and label large sets of images
- Accelerate deep learning with GPU's
- Visualize and debug deep neural networks
- Access and use models from experts



Training Accuracy Plot

Deep Dream

Network Activations

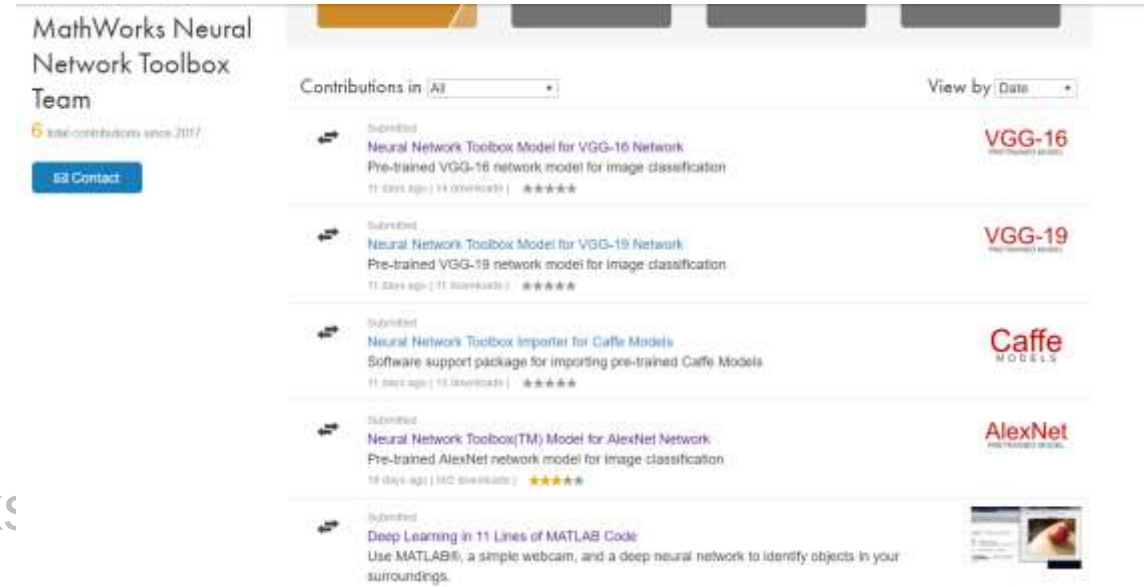


Feature Visualization

MATLAB makes Deep Learning Easy and Accessible

Learn about new MATLAB capabilities to

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



Curated Set of Pretrained Models

Access Models with 1-line of MATLAB Code

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

Regression Support for Deep Learning

Classification vs. Regression

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

Supported by new regression layer:

```
routputlayer = regressionLayer('Name','routput')
```

Example predict facial key-points: