

Survey of DNN Development Resources

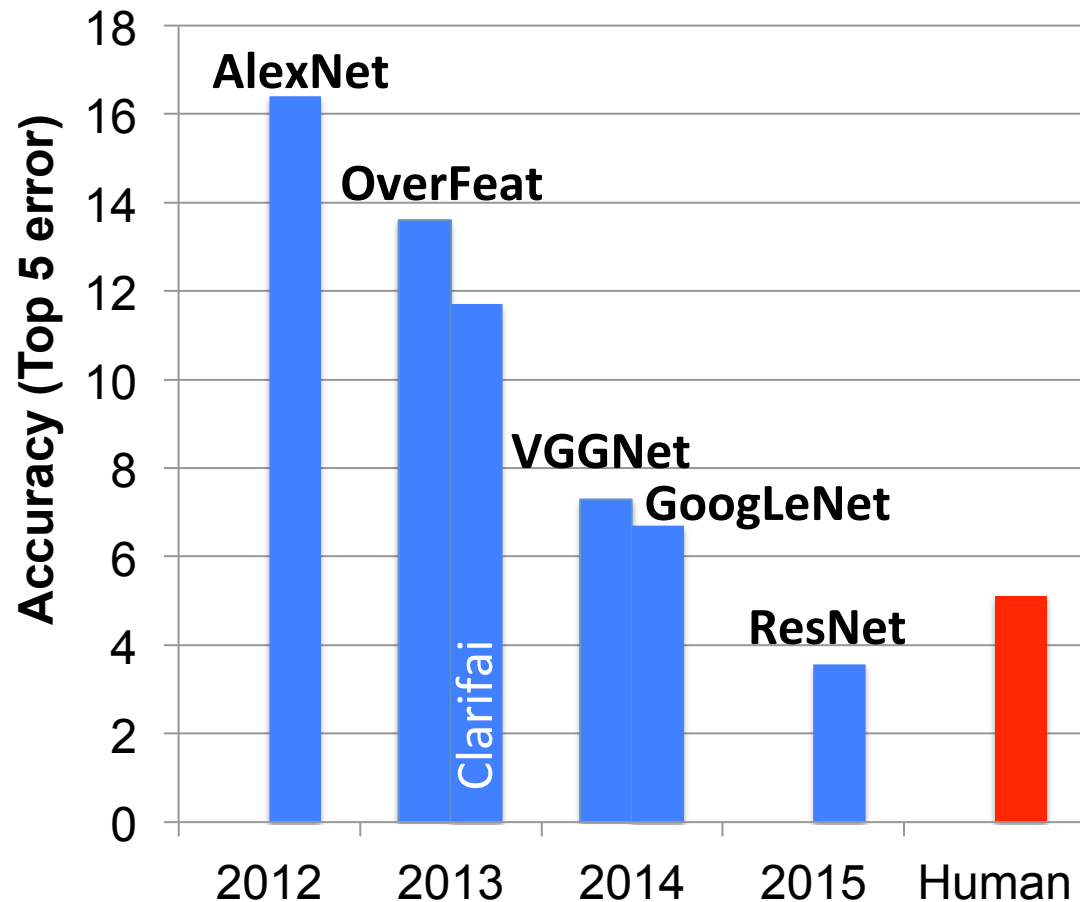
CICS/MTL Tutorial (2017)

Website: <http://eyeriss.mit.edu/tutorial.html>

Popular DNNs

- **LeNet (1998)**
- **AlexNet (2012)**
- **OverFeat (2013)**
- **VGGNet (2014)**
- **GoogLeNet (2014)**
- **ResNet (2015)**

ImageNet: Large Scale Visual Recognition Challenge (ILSVRC)



LeNet-5

CONV Layers: 2

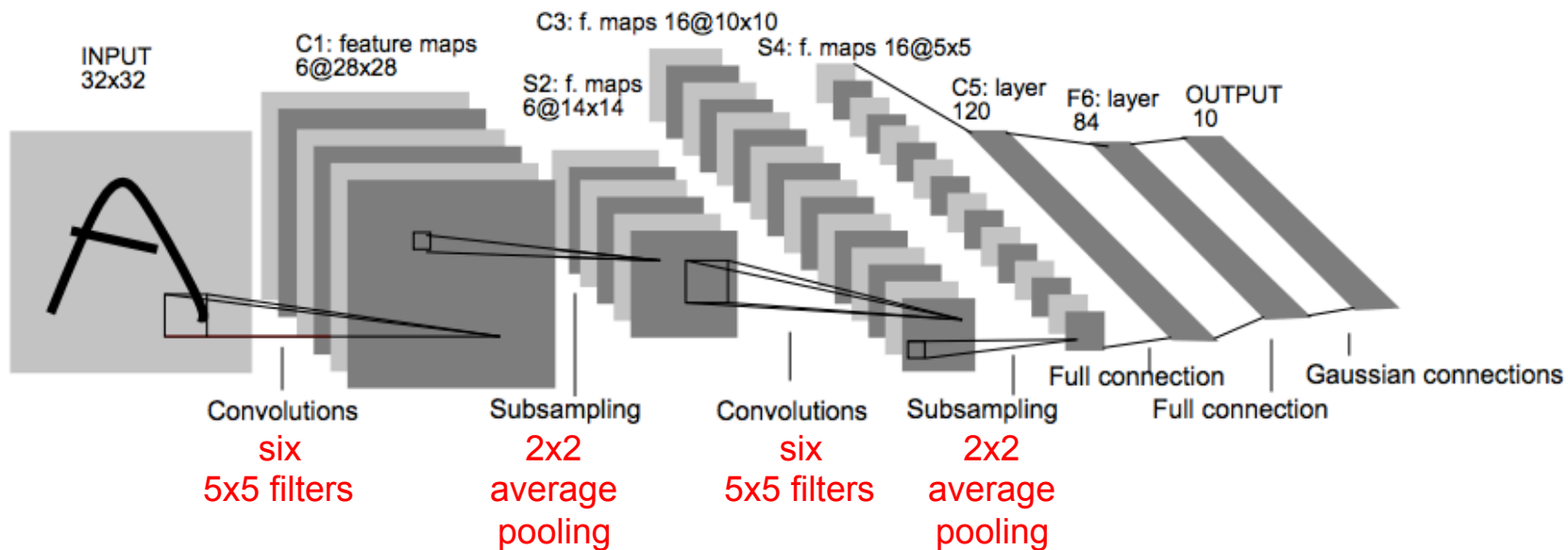
Fully Connected Layers: 2

Weights: 60k

MACs: 341k

Sigmoid used for non-linearity

Digit Classification!



AlexNet

CONV Layers: 5

Fully Connected Layers: 3

Weights: 61M

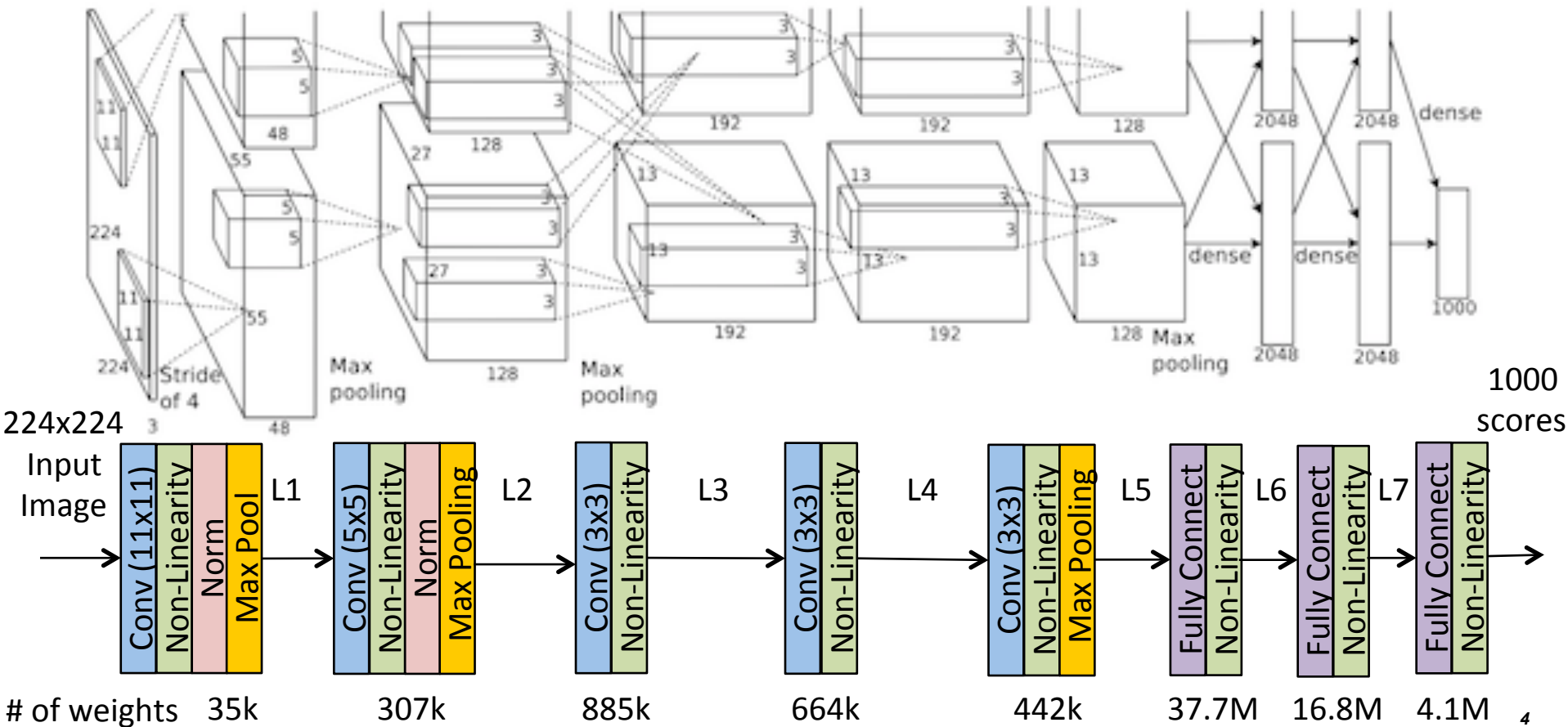
MACs: 724M

ReLU used for non-linearity

ILSCVR12 Winner

Uses Local Response Normalization (LRN)

[Krizhevsky et al., NIPS, 2012]

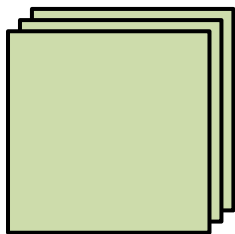


Large Sizes with Varying Shapes

AlexNet Convolutional Layer Configurations

Layer	Filter Size (RxS)	# Filters (M)	# Channels (C)	Stride
1	11x11	96	3	4
2	5x5	256	48	1
3	3x3	384	256	1
4	3x3	384	192	1
5	3x3	256	192	1

Layer 1



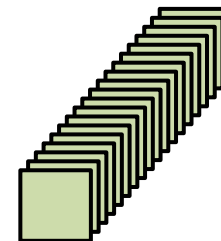
34k Params
105M MACs

Layer 2



307k Params
224M MACs

Layer 3



885k Params
150M MACs

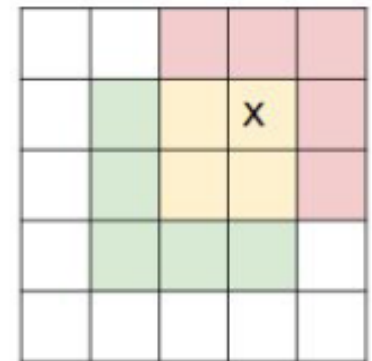
VGG-16

CONV Layers: 13
Fully Connected Layers: 3
Weights: 138M
MACs: 15.5G

Also, 19 layer version

Reduce # of weights

stack 2
3x3 conv



for a 5x5
receptive field

[figure credit
A. Karpathy]

More Layers → Deeper!

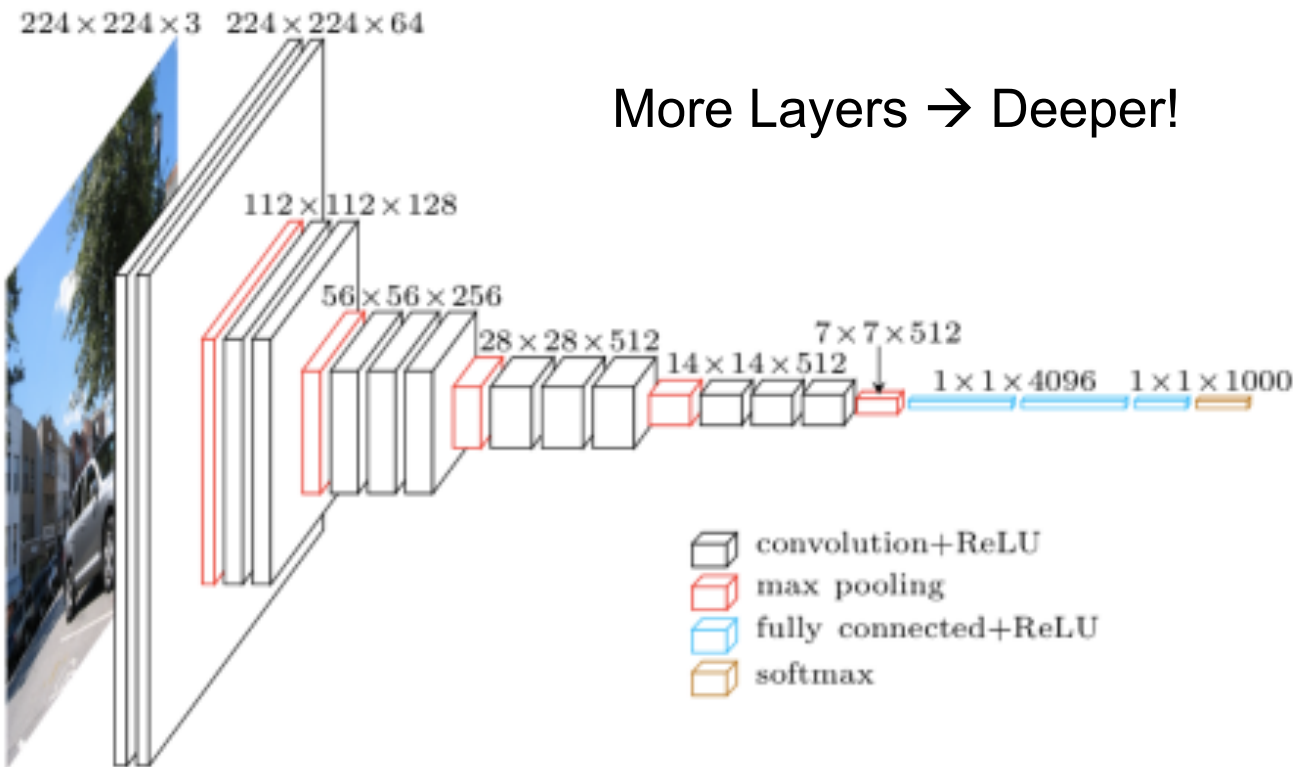


Image Source: <http://www.cs.toronto.edu/~frossard/post/vgg16/>

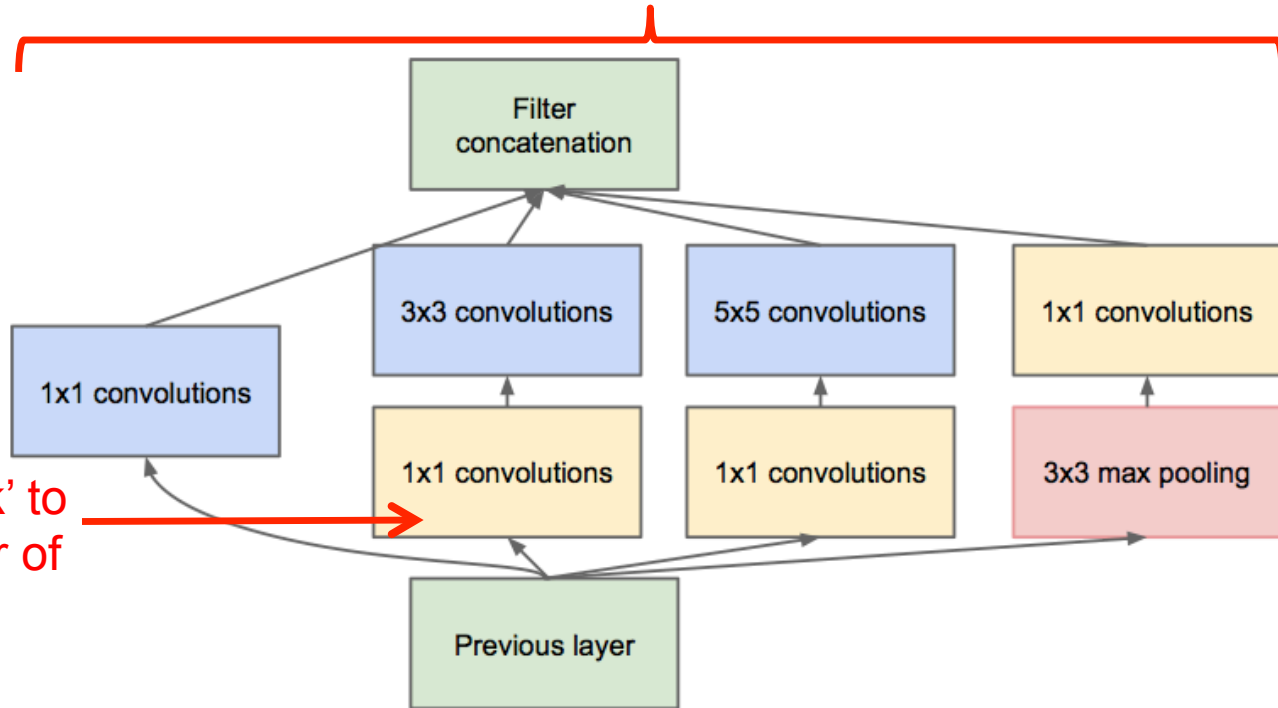
GoogLeNet (v1)

CONV Layers: 21 (depth), 57 (total)
Fully Connected Layers: 1
Weights: 7.0M
MACs: 1.43G

Also, v2, v3 and v4
ILSVRC14 Winner

parallel filters of different size has the effect of processing image at different scales

Inception Module



1x1 'bottleneck' to reduce number of weights

GoogLeNet (v1)

CONV Layers: 21 (depth), 57 (total)

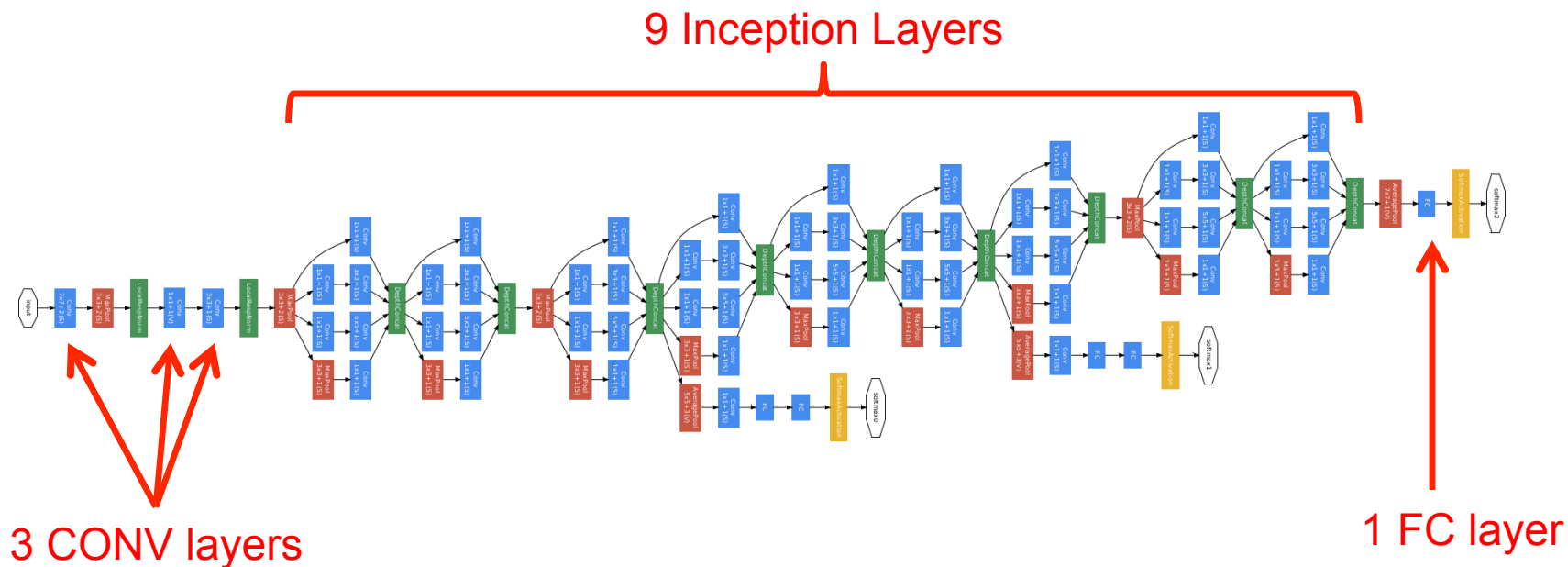
Fully Connected Layers: 1

Weights: 7.0M

MACs: 1.43G

Also, v2, v3 and v4

ILSVRC14 Winner



ResNet-50

CONV Layers: 49

Fully Connected Layers: 1

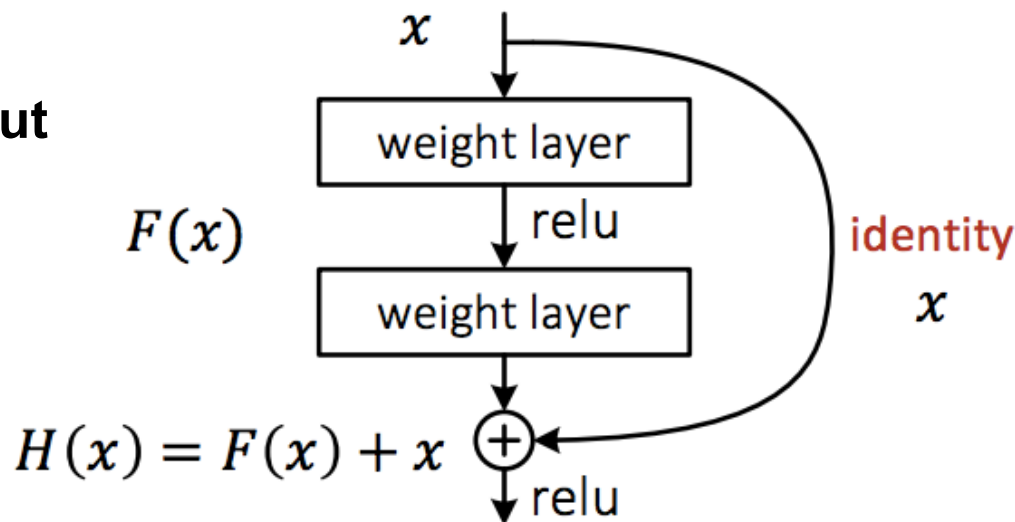
Weights: 25.5M

MACs: 3.9G

Also, 34, **152** and 1202 layer versions

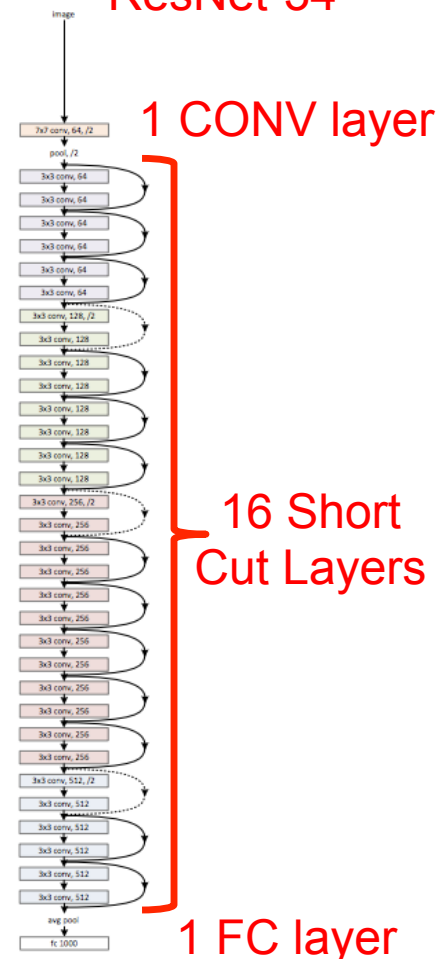
ILSVRC15 Winner

Short Cut Module



Helps address the vanishing gradient challenge for training very deep networks

ResNet-34



Revolution of Depth

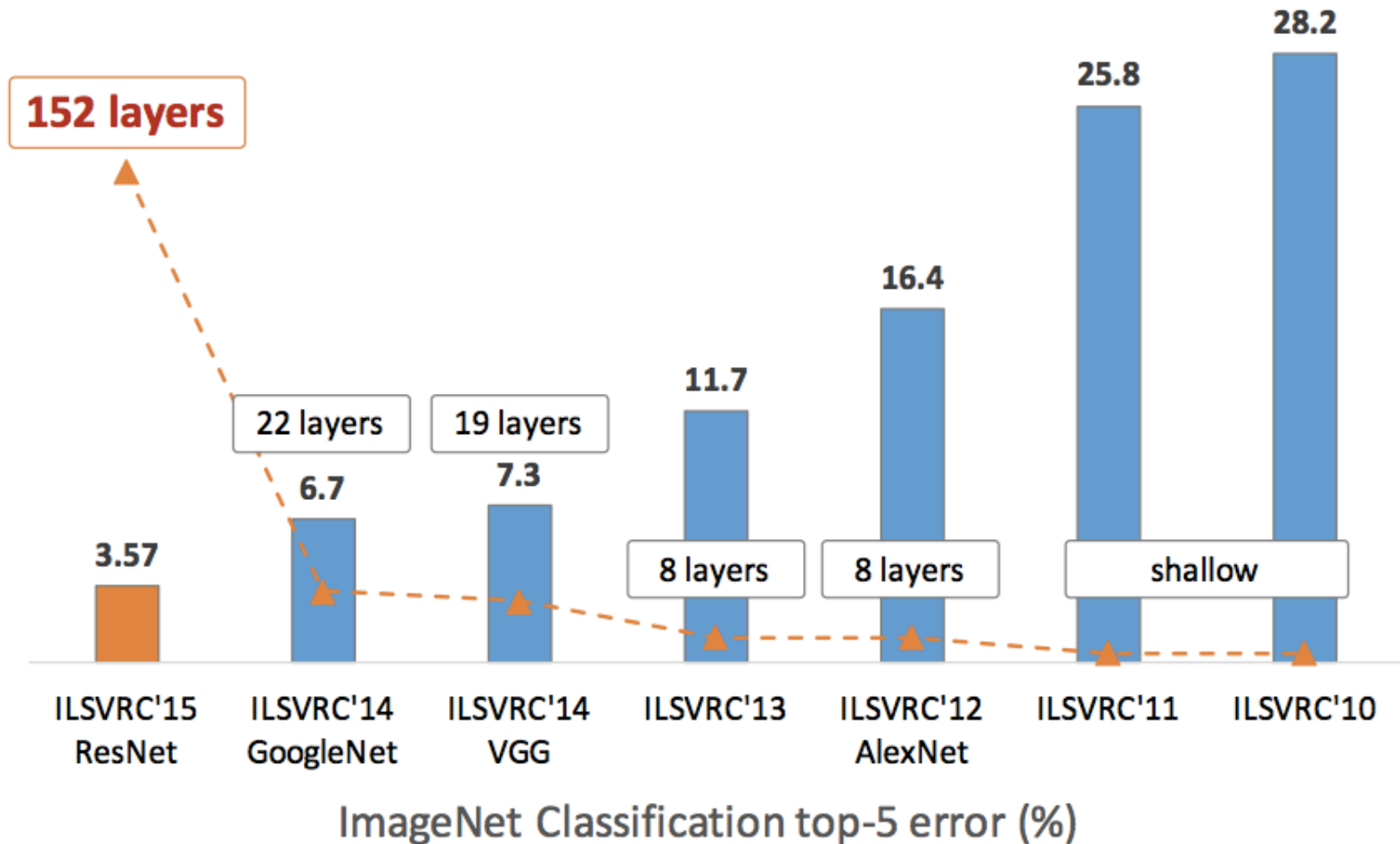


Image Source: http://icml.cc/2016/tutorials/icml2016_tutorial_deep_residual_networks_kaiminghe.pdf

Summary of Popular DNNs

Metrics	LeNet-5	AlexNet	VGG-16	GoogLeNet (v1)	ResNet-50
Top-5 error	n/a	16.4	7.4	6.7	5.3
Input Size	28x28	227x227	224x224	224x224	224x224
# of CONV Layers	2	5	16	21 (depth)	49
Filter Sizes	5	3, 5, 11	3	1, 3, 5, 7	1, 3, 7
# of Channels	1, 6	3 - 256	3 - 512	3 - 1024	3 - 2048
# of Filters	6, 16	96 - 384	64 - 512	64 - 384	64 - 2048
Stride	1	1, 4	1	1, 2	1, 2
# of Weights	2.6k	2.3M	14.7M	6.0M	23.5M
# of MACs	283k	666M	15.3G	1.43G	3.86G
# of FC layers	2	3	3	1	1
# of Weights	58k	58.6M	124M	1M	2M
# of MACs	58k	58.6M	124M	1M	2M
Total Weights	60k	61M	138M	7M	25.5M
Total MACs	341k	724M	15.5G	1.43G	3.9G

CONV Layers increasingly important!

Summary of Popular DNNs

- **AlexNet**
 - First CNN Winner of ILSVRC
 - Uses LRN (deprecated after this)
- **VGG-16**
 - Goes Deeper (16+ layers)
 - Uses only 3x3 filters (stack for larger filters)
- **GoogLeNet (v1)**
 - Reduces weights with Inception and only one FC layer
 - Inception: 1x1 and DAG (parallel connections)
 - Batch Normalization
- **ResNet**
 - Goes Deeper (24+ layers)
 - Shortcut connections

Frameworks

Caffe

Berkeley / BVLC
(C, C++, Python, MATLAB)



TensorFlow

Google
(C++, Python)

theano

U. Montreal
(Python)



Facebook / NYU
(C, C++, Lua)

Also, CNTK, MXNet, etc.

More at: <https://developer.nvidia.com/deep-learning-frameworks>

Example: Layers in Caffe

Convolution Layer

```
layer {  
  name: "conv1"  
  type: "Convolution"  
  bottom: "data"  
  top: "conv1"  
  ...  
  convolution_param {  
    num_output: 20  
    kernel_size: 5  
    stride: 1  
    ...  
  }  
}
```

Non-Linearity

```
layer {  
  name: "relu1"  
  type: "ReLU"  
  bottom: "conv1"  
  top: "conv1"  
}
```

Pooling Layer

```
layer {  
  name: "pool1"  
  type: "Pooling"  
  bottom: "conv1"  
  top: "pool1"  
  pooling_param {  
    pool: MAX  
    kernel_size: 2  
    stride: 2 ...  
  }  
}
```

Benefits of Frameworks

- **Rapid development**
- **Sharing models**
- **Workload profiling**
- **Network hardware co-design**

Image Classification Datasets

- **Image Classification/Recognition**
 - Given an entire image → Select 1 of N classes
 - No localization (detection)

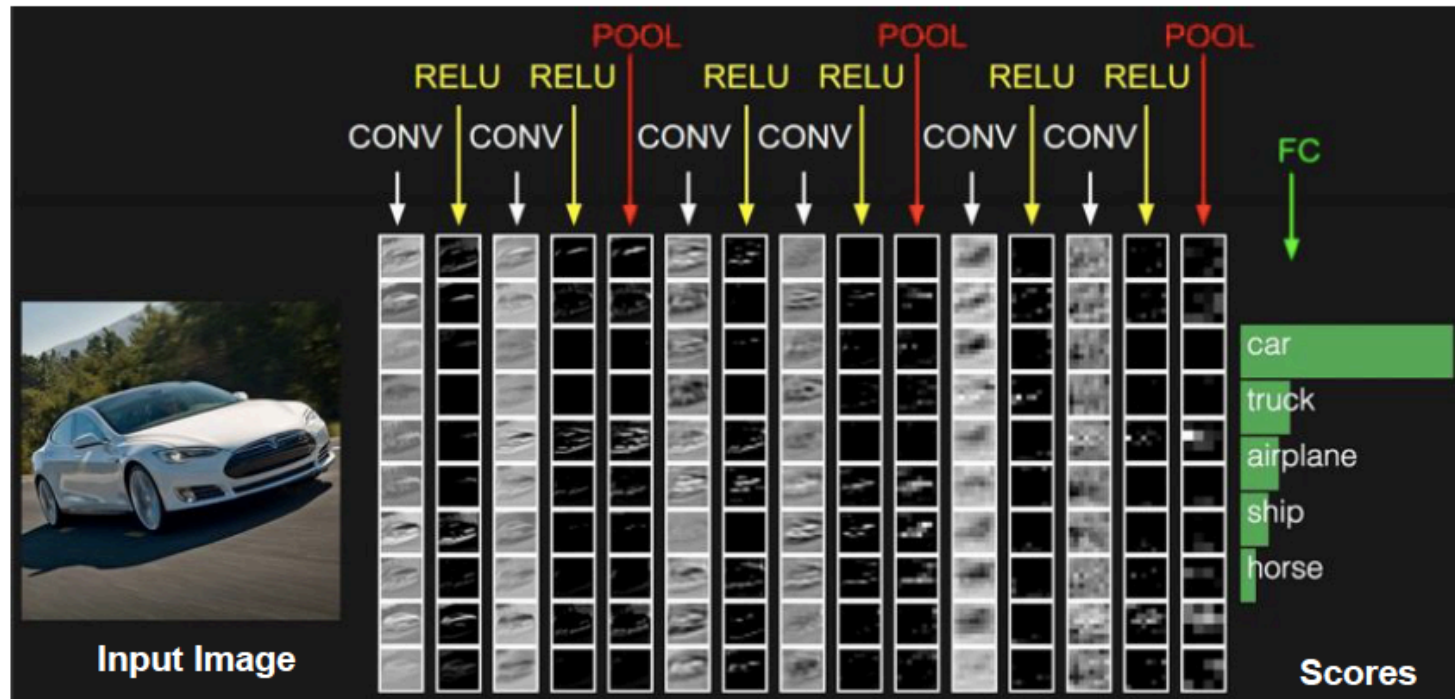


Image Source: Stanford cs231n

Datasets affect difficulty of task

MNIST

Digit Classification

28x28 pixels (B&W)

10 Classes

60,000 Training

10,000 Testing

LeNet in 1998

(0.95% error)



ICML 2013

(0.21% error)



Object Classification

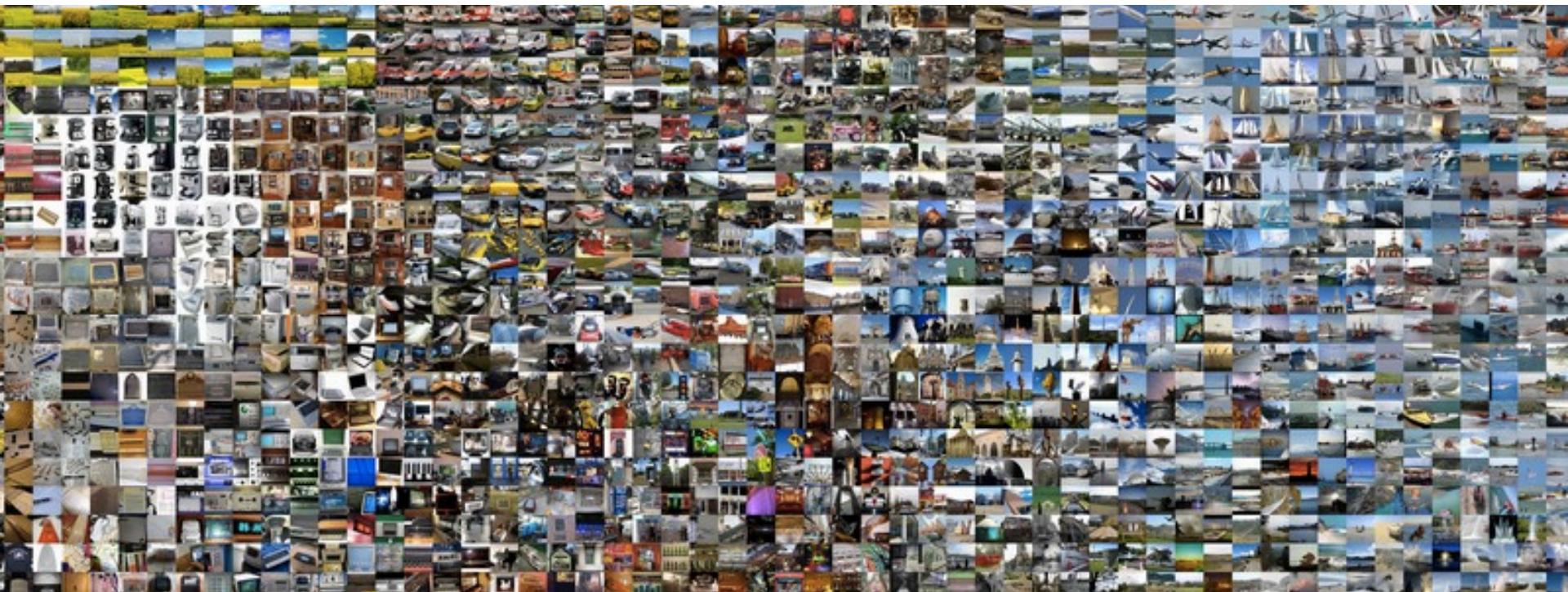
~256x256 pixels (color)

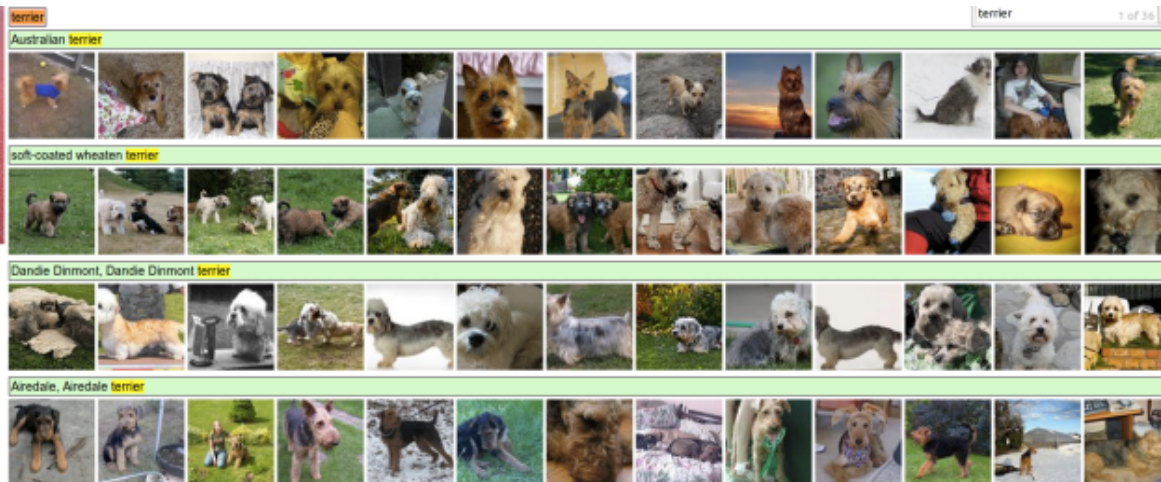
1000 Classes

1.3M Training

100,000 Testing (50,000 Validation)

Image Source: <http://karpathy.github.io/>





**Fine grained
Classes**
(120 breeds)

Image Source: <http://karpathy.github.io/>

Image Source: Krizhevsky et al., NIPS 2012

Top-5 Error

Winner 2012
(16.42% error)



Winner 2016
(2.99% error)

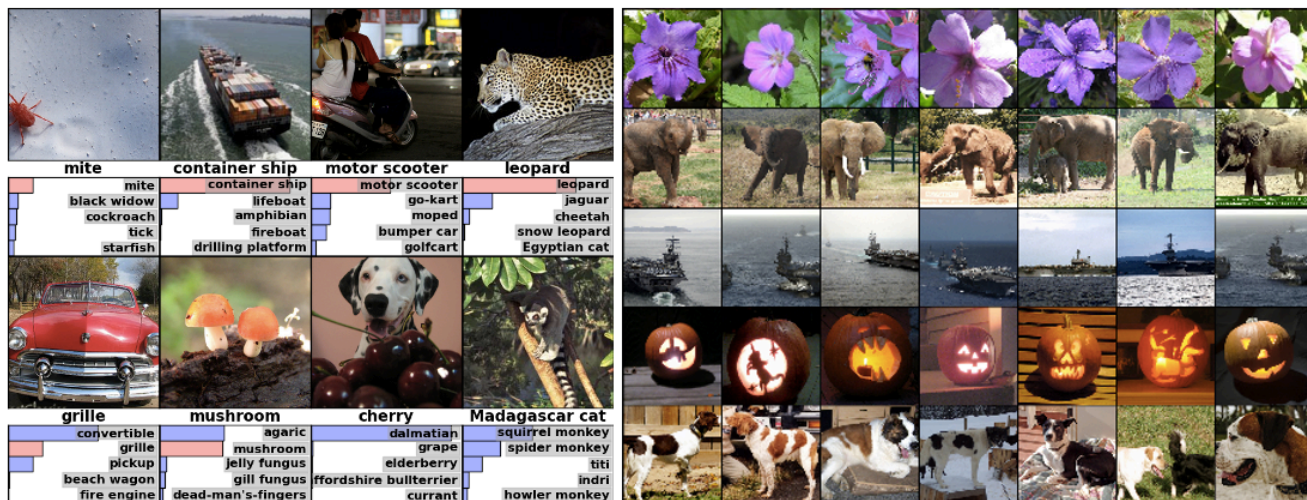


Image Classification Summary

	MNIST	IMAGENET
Year	1998	2012
Resolution	28x28	256x256
Classes	10	1000
Training	60k	1.3M
Testing	10k	100k
Accuracy	0.21% error (ICML 2013)	2.99% top-5 error (2016 winner)

http://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results.html

Next Tasks: Localization and Detection

Image classification

Steel drum



Ground truth

Steel drum
Folding chair
Loudspeaker

Accuracy: 1

Scale
T-shirt
Steel drum
Drumstick
Mud turtle

Accuracy: 1

Scale
T-shirt
Giant panda
Drumstick
Mud turtle

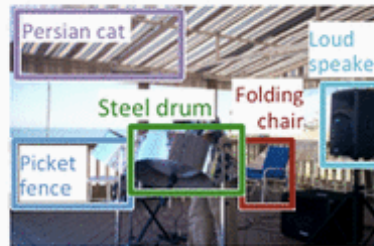
Accuracy: 0

Single-object localization

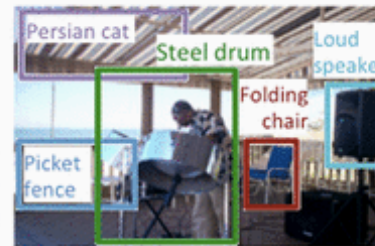
Steel drum



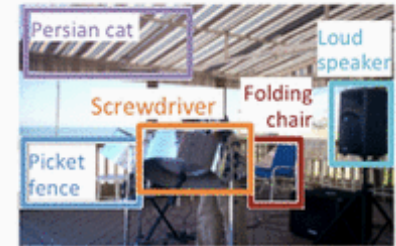
Ground truth



Accuracy: 1

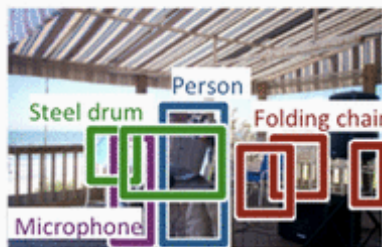


Accuracy: 0

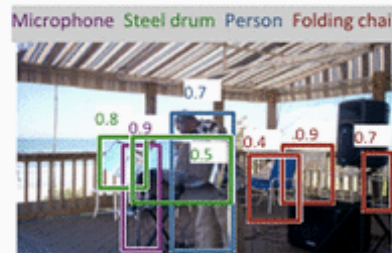


Accuracy: 0

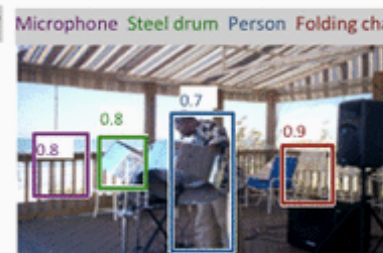
Object detection



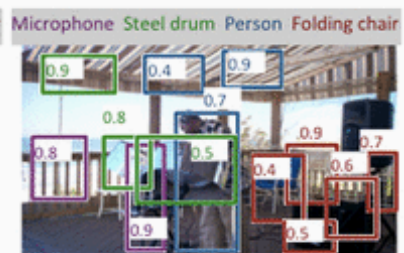
Ground truth



AP: 1.0 1.0 1.0 1.0



AP: 0.0 0.5 1.0 0.3



AP: 1.0 0.7 0.5 0.9

Recently Introduced Datasets

- **Google Open Images (~9M images)**
 - <https://github.com/openimages/dataset>
- **Youtube-8M (8M videos)**
 - <https://research.google.com/youtube8m/>
- **AudioSet (2M sound clips)**
 - <https://research.google.com/audioset/index.html>

Summary

- **Development resources presented in this section enable us to evaluate hardware using the appropriate DNN model and dataset**
 - **Difficult tasks typically require larger models**
 - **Different datasets for different tasks**
 - **Number of datasets growing at a rapid pace**