Very Deep ConvNets for Large-Scale Image Recognition

Karen Simonyan, Andrew Zisserman

University of Oxford

Overview

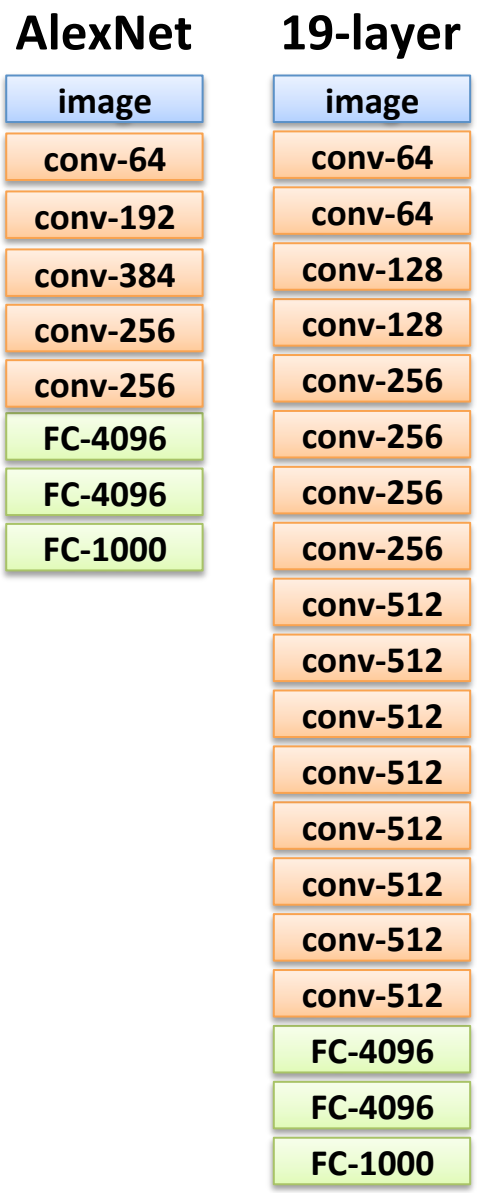
- ConvNet design choices: depth, width, receptive fields
- Variety of models with different depth:

ConvNet Model	Dataset	Conv. & FC layers
LeNet-5 [LeCun et al., 1998]	MNIST	5
[Ciresan et al., 2012]	CIFAR	7
AlexNet [Krizhevsky et al., 2012] [Zeiler & Fergus, 2012]	ImageNet Challenge	8
OverFeat [Sermanet et al., 2012]		9
[Goodfellow et al., 2013]	SVHN	11

- How does ConvNet depth affect accuracy on ImageNet?

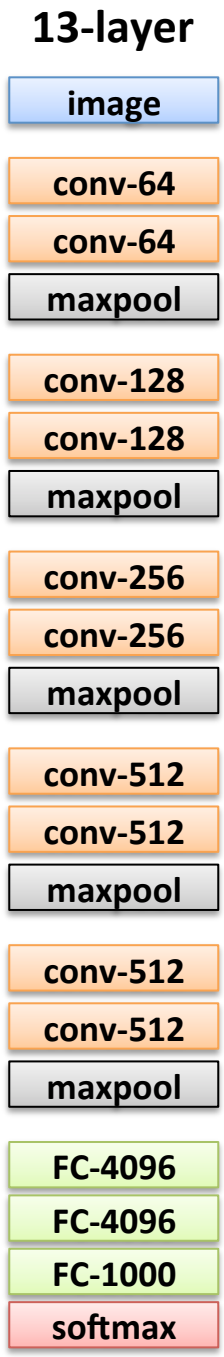
Contributions

- Comparison of very deep ConvNets
  - simple architecture design
  - increasing depth: from 11 to 19 layers
- Evaluation of very deep features on other datasets
- The models are publicly available



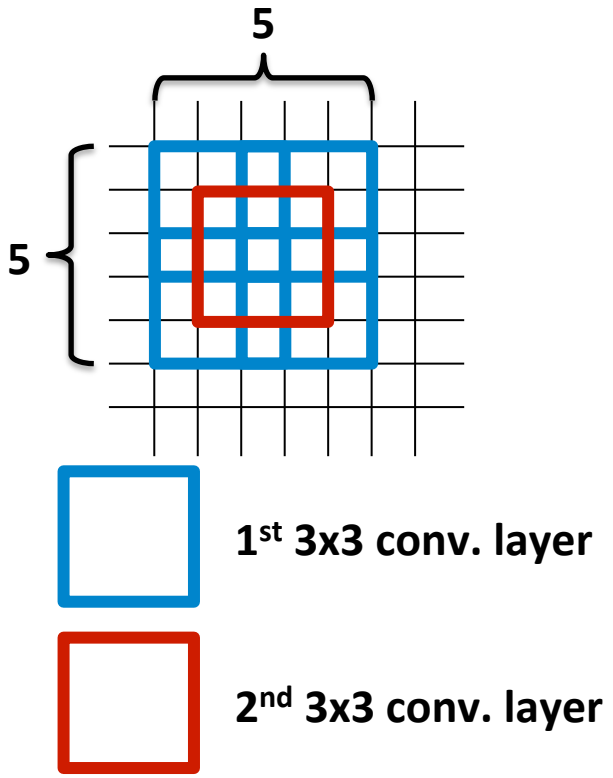
Network Design

- Single family of networks
- Key design choices:
  - 3x3 conv. kernels – very small
  - stacks of conv. layers w/o pooling (but with ReLU)
  - conv. stride 1 – no skipping
- Other details are conventional:
  - Rectification (ReLU) non-linearity
  - 5 max-pool layers (x2 downsampling)
  - no normalisation
  - 3 fully-connected layers

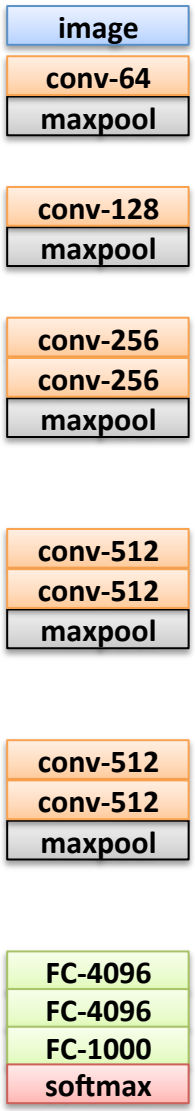


Why Stacks of 3x3 Filters?

- A stack of 3x3 layers has a large receptive field
  - two 3x3 layers – 5x5 receptive field
  - three 3x3 layers – 7x7 receptive field
- More ReLU non-linearities
  - more discriminative
- Less parameters than a single layer
- Design simplicity
  - no need to tune layer parameters



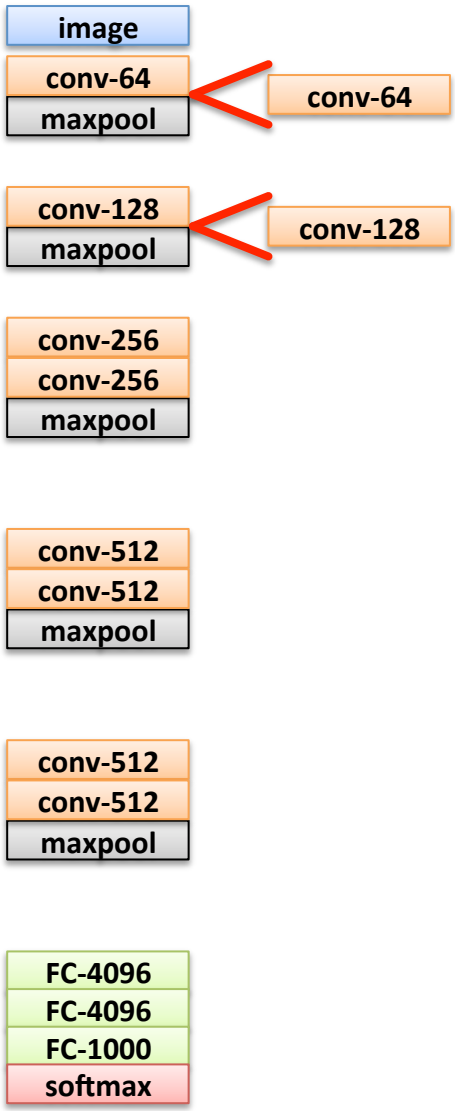
The Networks



11-layer

- Start from 11 layers, inject more conv. layers

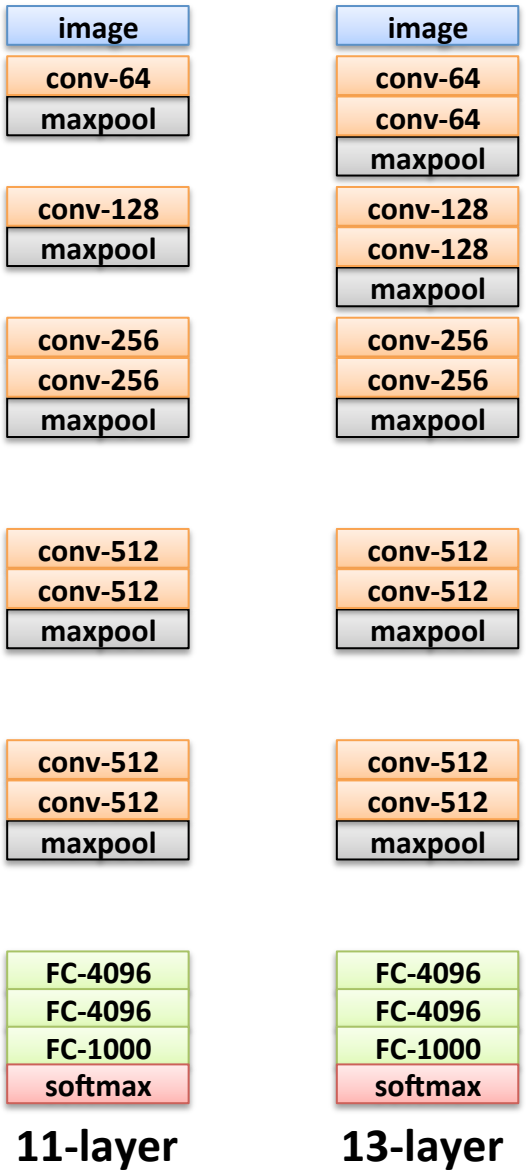
The Networks



11-layer

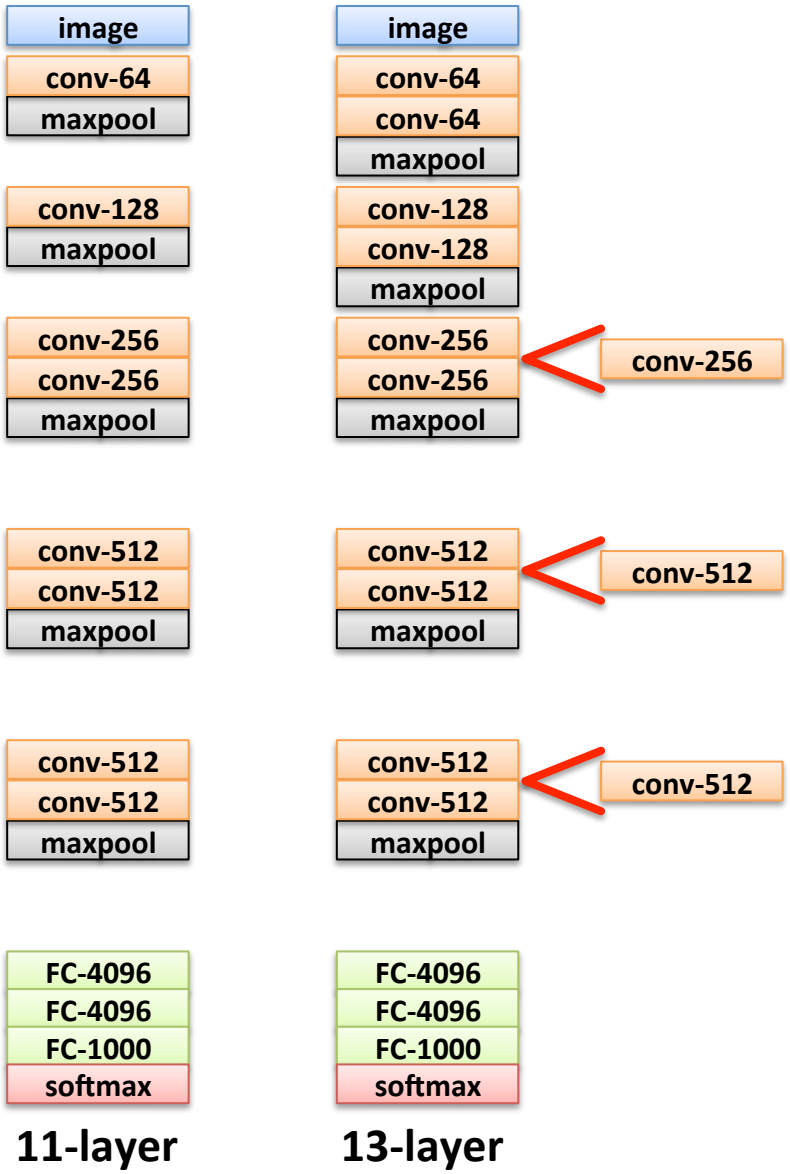
- Start from 11 layers, inject more conv. layers

The Networks



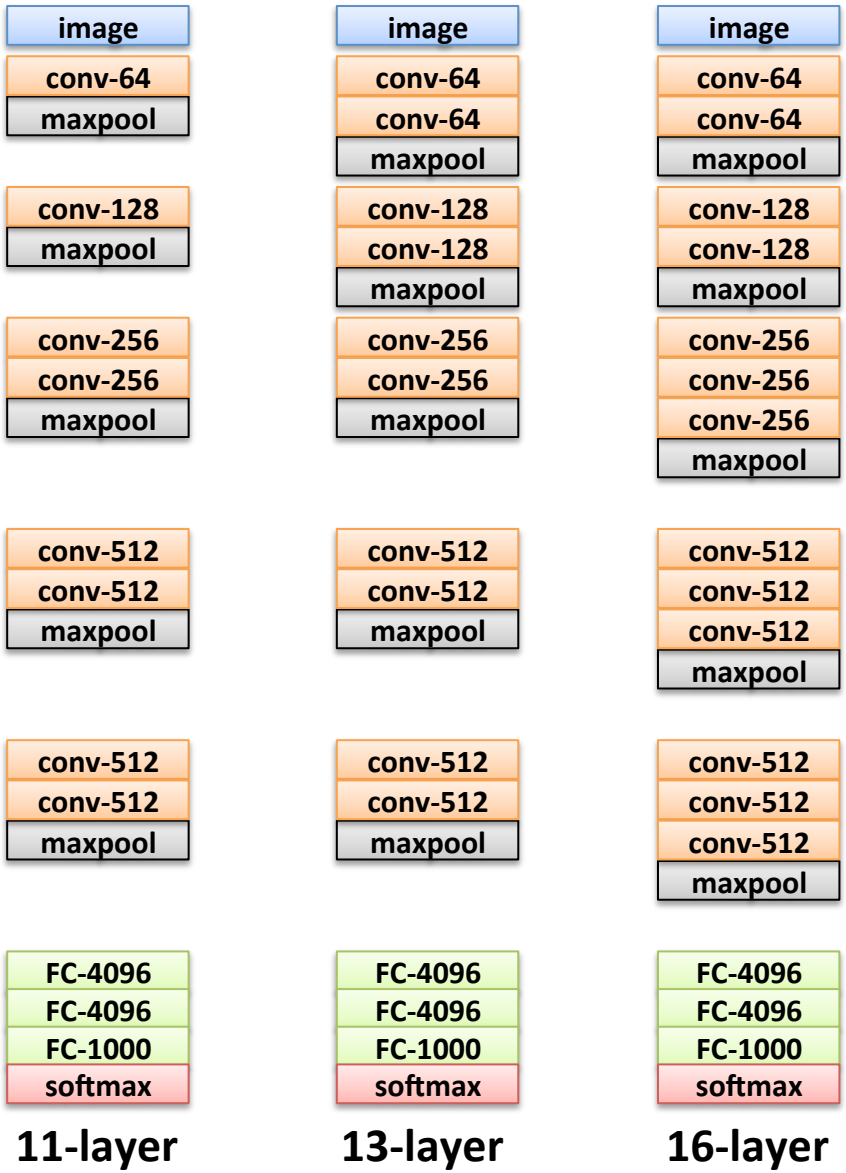
- Start from 11 layers, inject more conv. layers

The Networks



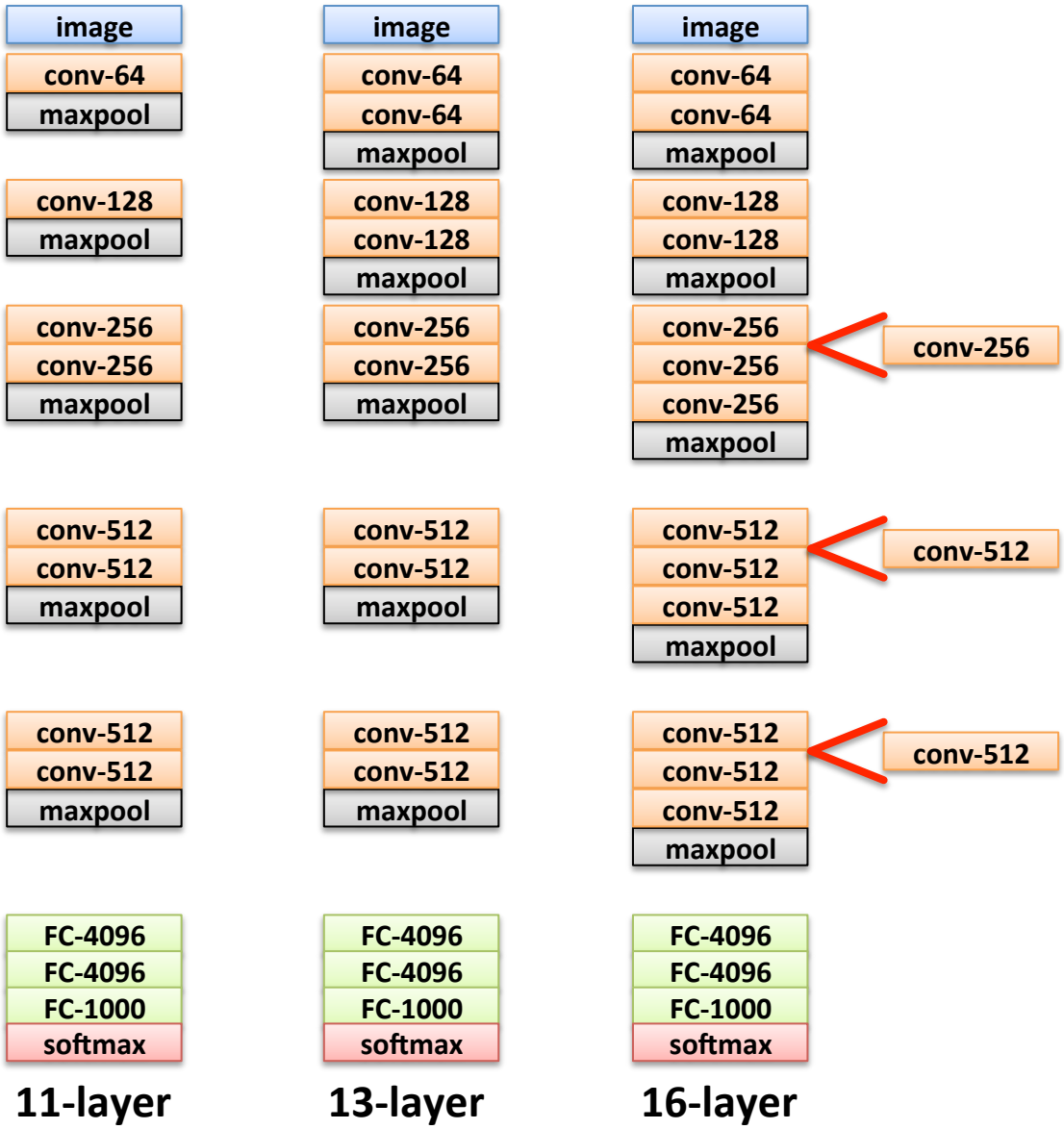
- Start from 11 layers, inject more conv. layers

The Networks



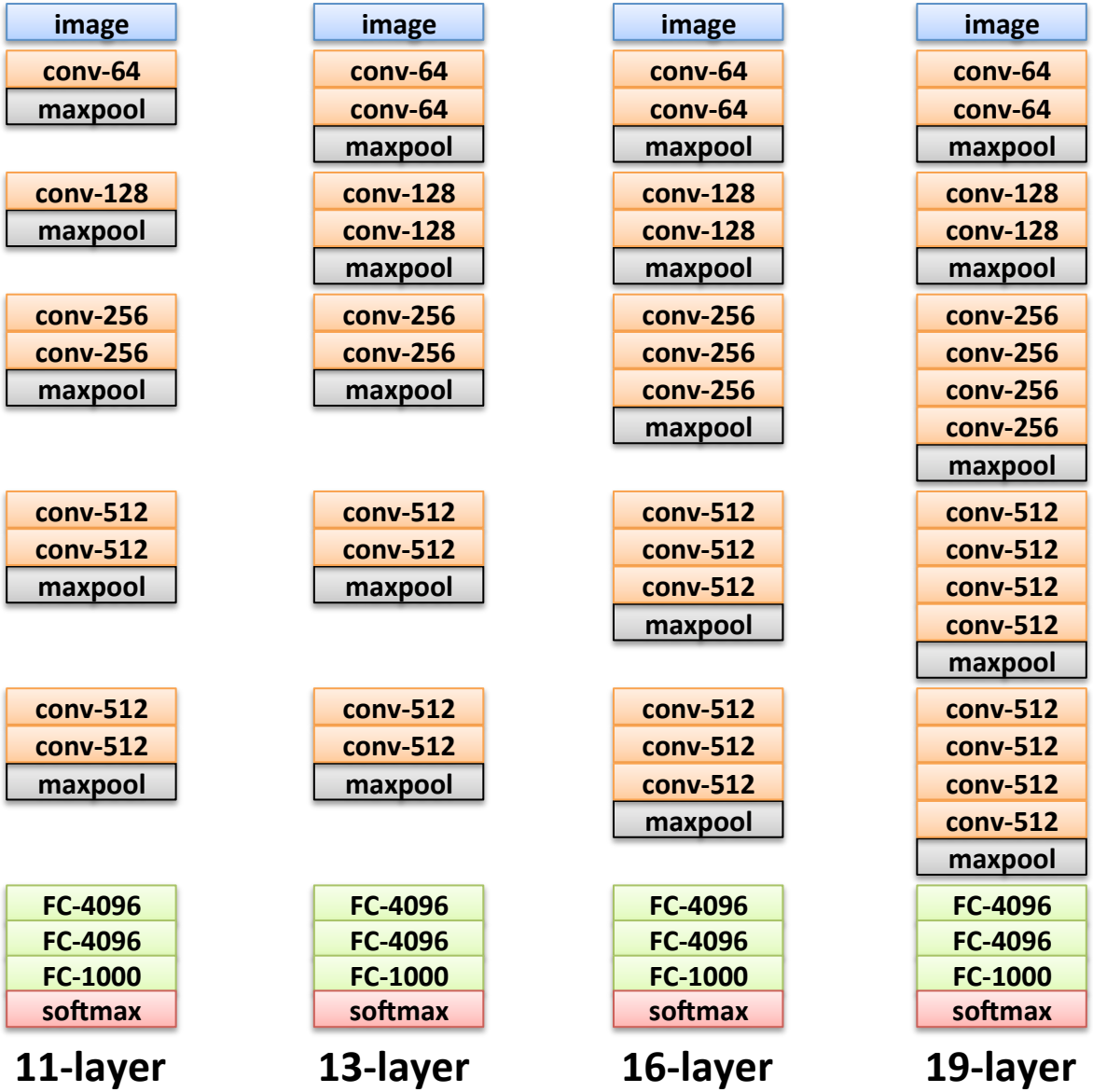
- Start from 11 layers, inject more conv. layers

The Networks



- Start from 11 layers, inject more conv. layers

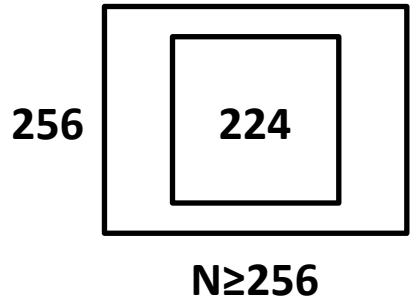
The Networks

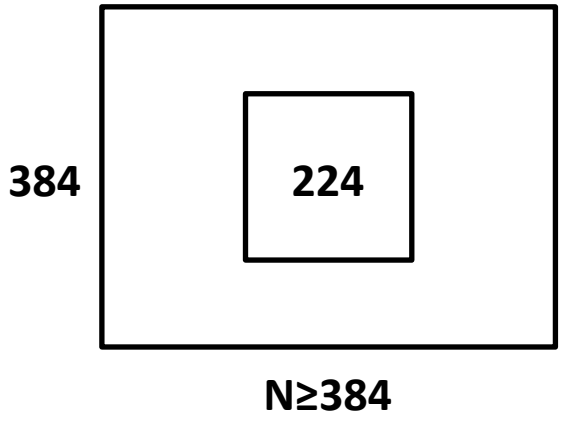


- Start from 11 layers, inject more conv. layers

Training

- ConvNet input during training: fixed-size 224x224 crop
- But images have various sizes…
- Solution:
  - rescale images to a certain size (preserving aspect ratio)
  - random 224x224 crop
- Image size affects the scale of image statistics seen by a ConvNet
  - single-scale: 256xN or 384xN
  - multi-scale: randomly sample the size for each image from 256xN to 512xN
- Standard augmentation: random flip and RGB shift



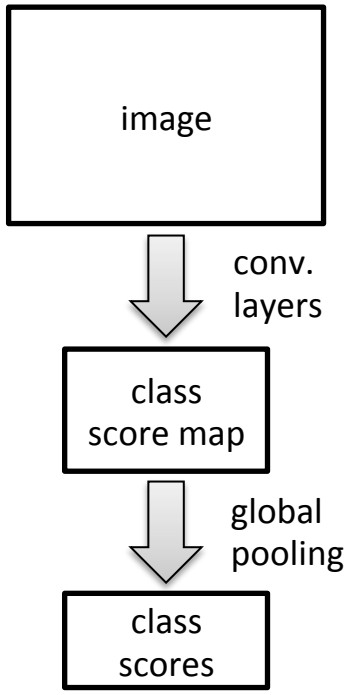


Training (2)

- Mini-batch gradient descent with momentum
- Regularisation: dropout and weight decay
- Fast convergence (74 training epochs)
- Initialisation
  - deep nets are prone to vanishing/exploding gradient
  - 11-layer net: random initialisation from $\mathcal{N}(0;0.01)$
  - deeper nets
    - top and bottom layers initialised with 11-layer net
    - other layers – random initialisation
  - also possible to initialise all layers randomly [Glorot & Bengio, 2010]

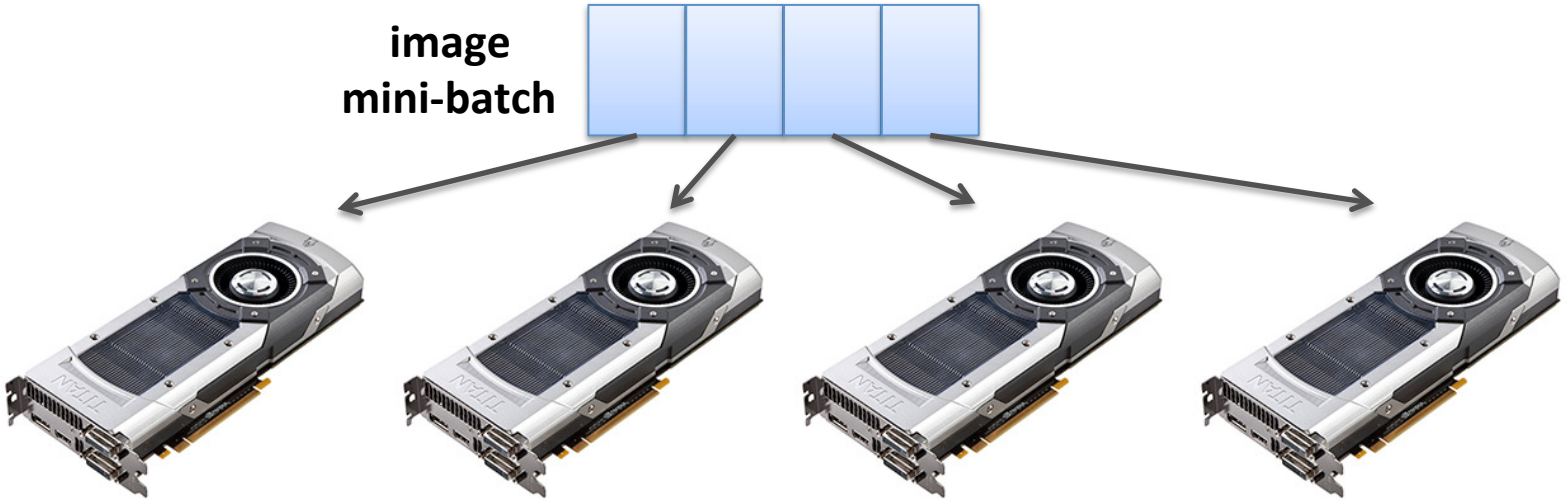
Testing

- ConvNet evaluation on variable-size images:
  - Testing on multiple 224x224 crops [AlexNet]
  - Dense application over the whole image: [OverFeat]
    - all layers are treated as convolutional
    - sum-pooling of class score maps
    - more efficient than multiple crops
  - Combination of both
- Multiple image sizes:
  - 256xN, 384xN, 512xN
  - class scores averaged



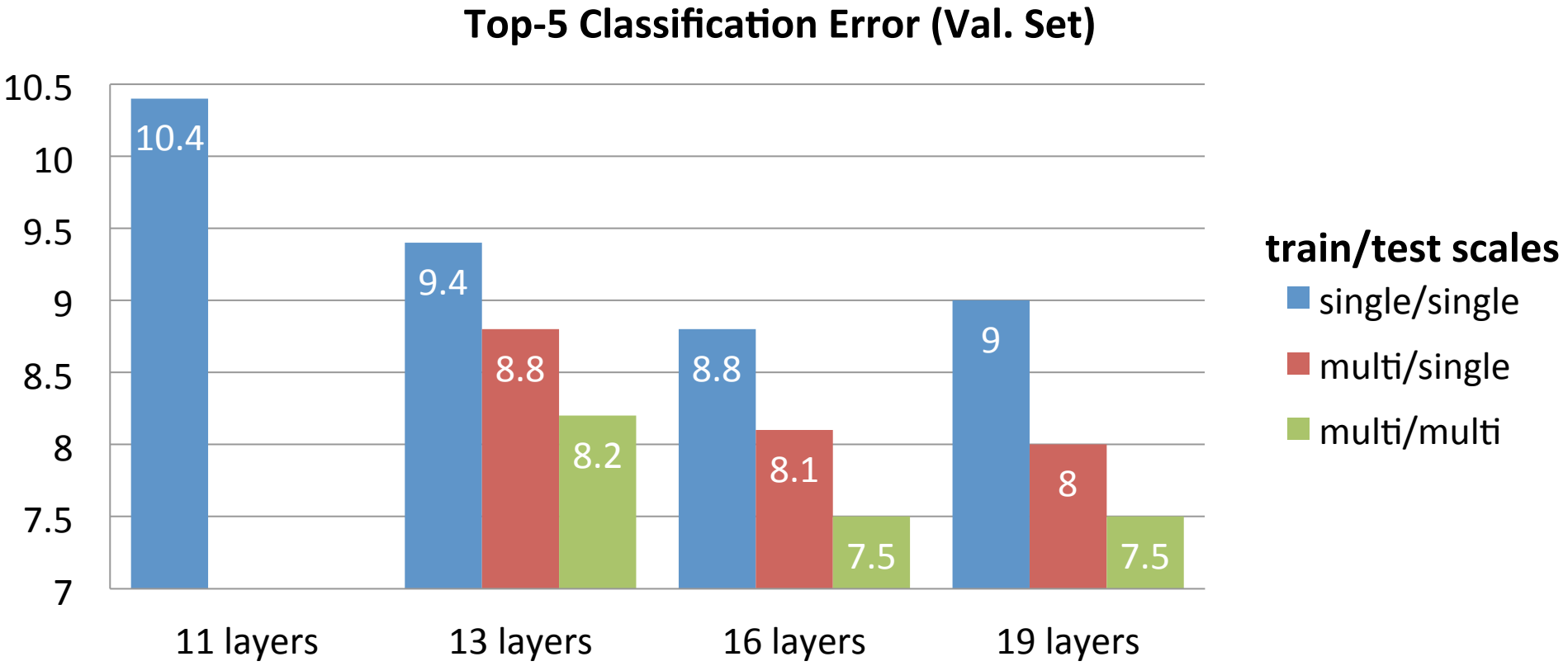
Implementation

- Heavily-modified Caffe toolbox
- Multiple GPU support
  - 4 x NVIDIA Titan in an off-the-shelf workstation
  - synchronous data parallelism
  - ~3.75 times speed-up, 2-3 weeks for training



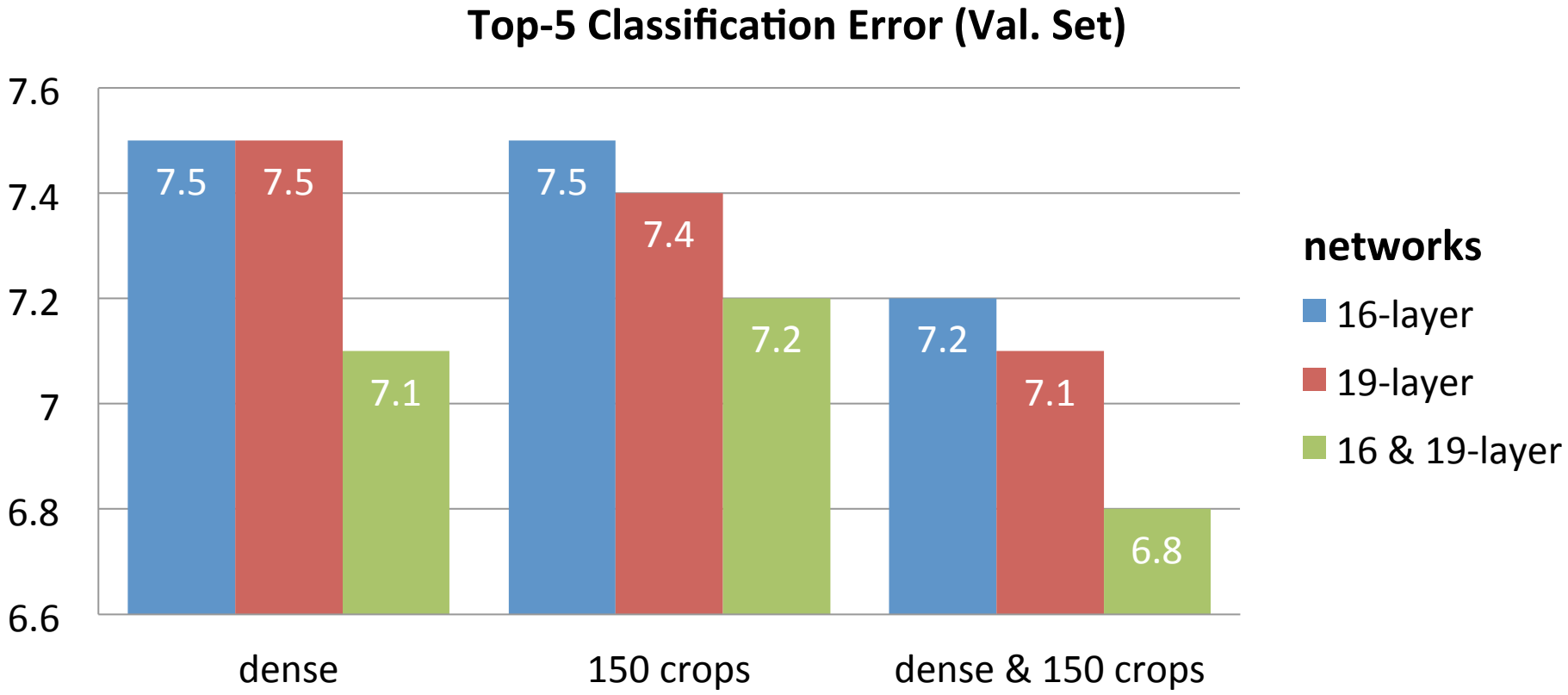
Evaluation on ImageNet Classification

Effect of Depth and Scale



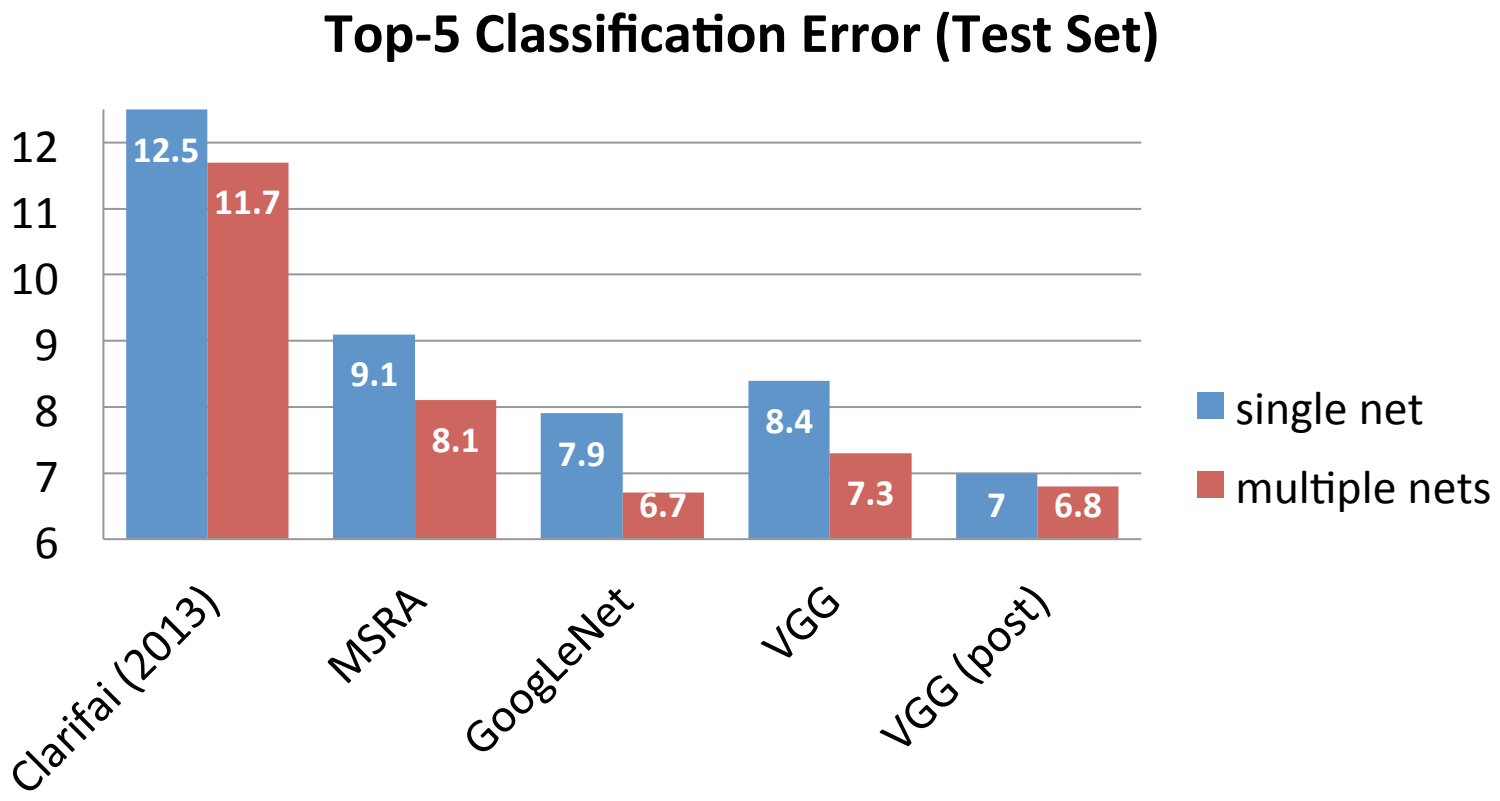
- Error decreases with depth
- Using multiple scales is important
  - multi-scale training outperforms single-scale
  - multi-scale testing further improves the results

Evaluation: Dense vs Multi-Crop



- Dense evaluation is on par with multi-crop
- Dense & multi-crop are complementary
- Combining predictions from 2 nets is beneficial

ImageNet Challenge 2014 (Classification)

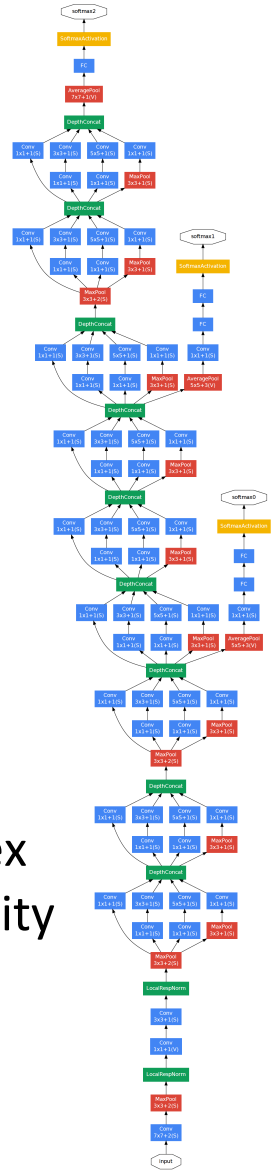


- 1st place: GoogLeNet (6.7% with 7 nets, 7.9% with 1 net)
- 2nd place: this work
  - submission: 7.3% with 2 nets, 8.4% with 1 net
  - post-submission: 6.8% with 2 nets, 7.0% with 1 net

Comparison with GoogLeNet

image
conv-64
conv-64
maxpool
conv-128
conv-128
maxpool
conv-256
conv-256
conv-256
conv-256
maxpool
conv-512
conv-512
conv-512
conv-512
maxpool
conv-512
conv-512
conv-512
conv-512
maxpool
FC-4096
FC-4096
FC-1000
softmax

- models developed independently
- both are very deep:
  - 19 (VGG) vs 22 (GoogLeNet) weight layers
- mitigating gradient instability:
  - pre-training (VGG) vs auxiliary losses (GoogLeNet)
- multi-scale training of both
- filter configuration:
  - VGG: only 3x3, stride 1 convolution
  - GoogLeNet: 5x5, 3x3, 1x1 filters combined into complex multi-branch modules to reduce computation complexity
- speed: GoogLeNet is several times faster
- single-model accuracy: VGG is ~0.5% better



Recent Advances

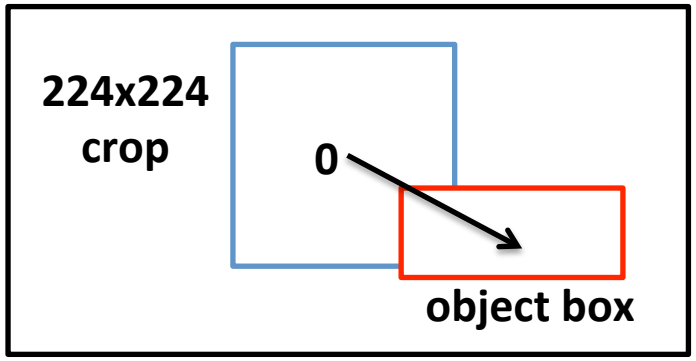
- Deep Image: Scaling up Image Recognition
  - 5.33% error
  - models based on VGG-16, but wider
- Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification
  - 4.94% error
  - models based on VGG-19, but deeper and wider with PReLU
- Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift
  - 4.82% error
  - Inception layers with batch normalisation

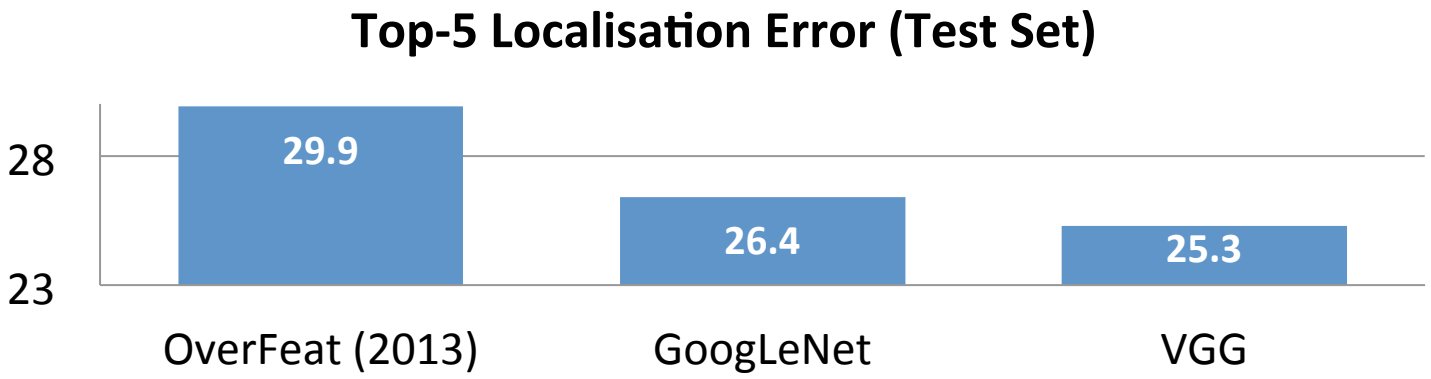
Beyond ImageNet Classification…

ImageNet Challenge 2014 (Localisation)

Object bounding box prediction

- Very deep location regressor (similar to OverFeat)
- Last layer predicts a box for each class
- Initialised with classification models
- Fine-tuning of all layers





- 1st place: VGG (25.3% error), 2nd place: GoogLeNet (26.4%)
- Outperforms OverFeat by using deeper representations

Very Deep Features on Other Datasets

- ImageNet-pretrained ConvNets are excellent feature extractors on other datasets
- We compare against the state of the art, based on less deep ConvNet features, trained on ILSVRC
- Simple recognition pipeline
  - last fully-connected (classification) layer is removed
  - dense evaluation over the whole image → global sum-pooling
  - aggregation over several scales
  - L_2 normalisation → linear SVM
  - no fine-tuning

Results on Caltech & PASCAL VOC

Method	Caltech 101	Caltech 256	PASCAL VOC 2007	PASCAL VOC 2012 (class-n)	PASCAL VOC 2012 (actions)
Best published result using shallow ConvNets	**93.4**	77.6	81.5	81.7	76.3
19-layer	92.3	**85.1**	**89.3**	**89.0**	-
16 & 19-layer	92.7	**86.2**	**89.7**	**89.3**	**84.0**

- Very deep features are competitive or better than the state of the art despite a simple pipeline
- Two ConvNet extractors are complementary
- Even better results have been recently reported using our released models

Other Tasks…

Since the public release of models, they have been successfully applied by the community to:

- object detection
- semantic segmentation
- image caption generation
- texture classification
- etc.

Summary

- Representation depth is important
- Excellent results using very deep ConvNets
  - simple architecture, small receptive fields
  - but very deep → lots of non-linearity
- VGG-16 & VGG-19 models publicly available from:
  - http://www.robots.ox.ac.uk/~vgg/research/very_deep/
  - formats: Caffe and MatConvNet
  - can be used with any package with a cuDNN back-end, e.g. Torch and Theano

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPUs used for this research.