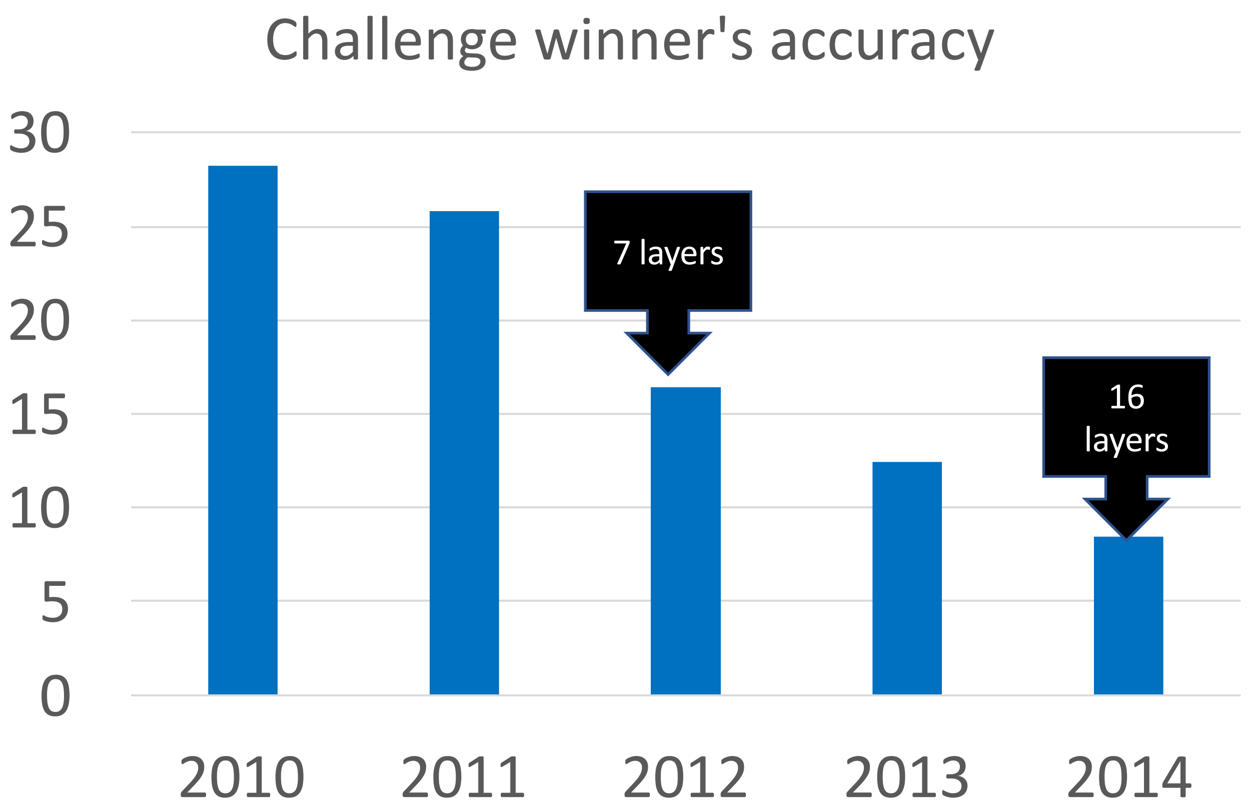
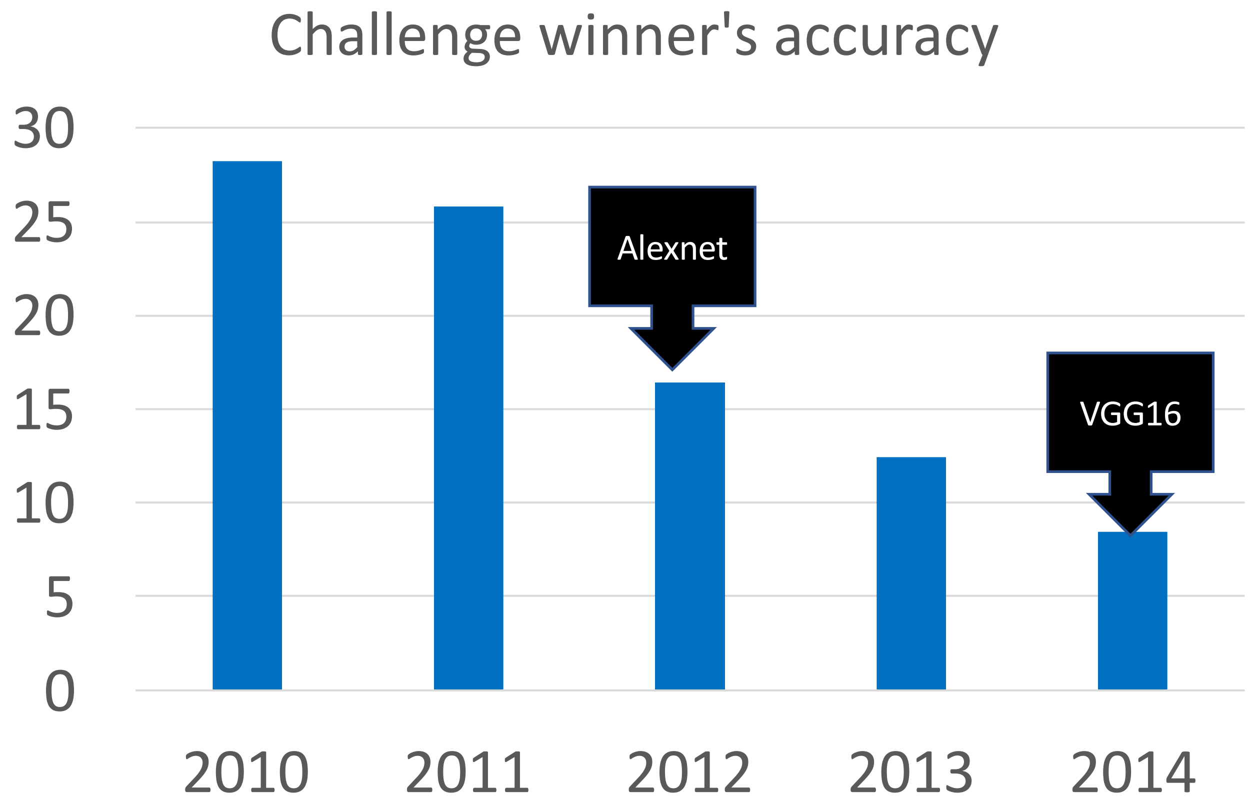
Exploring convnet architectures

Deeper is better



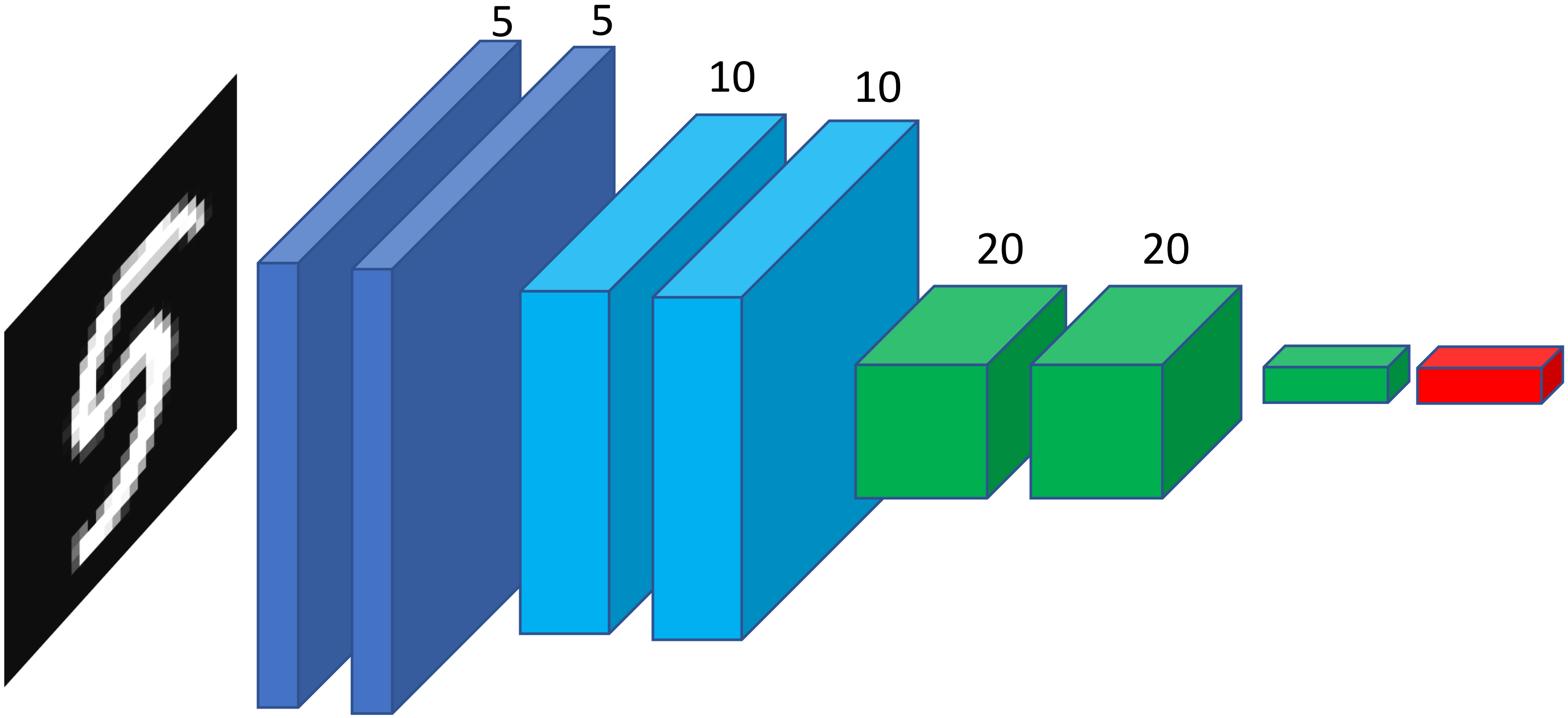
Deeper is better



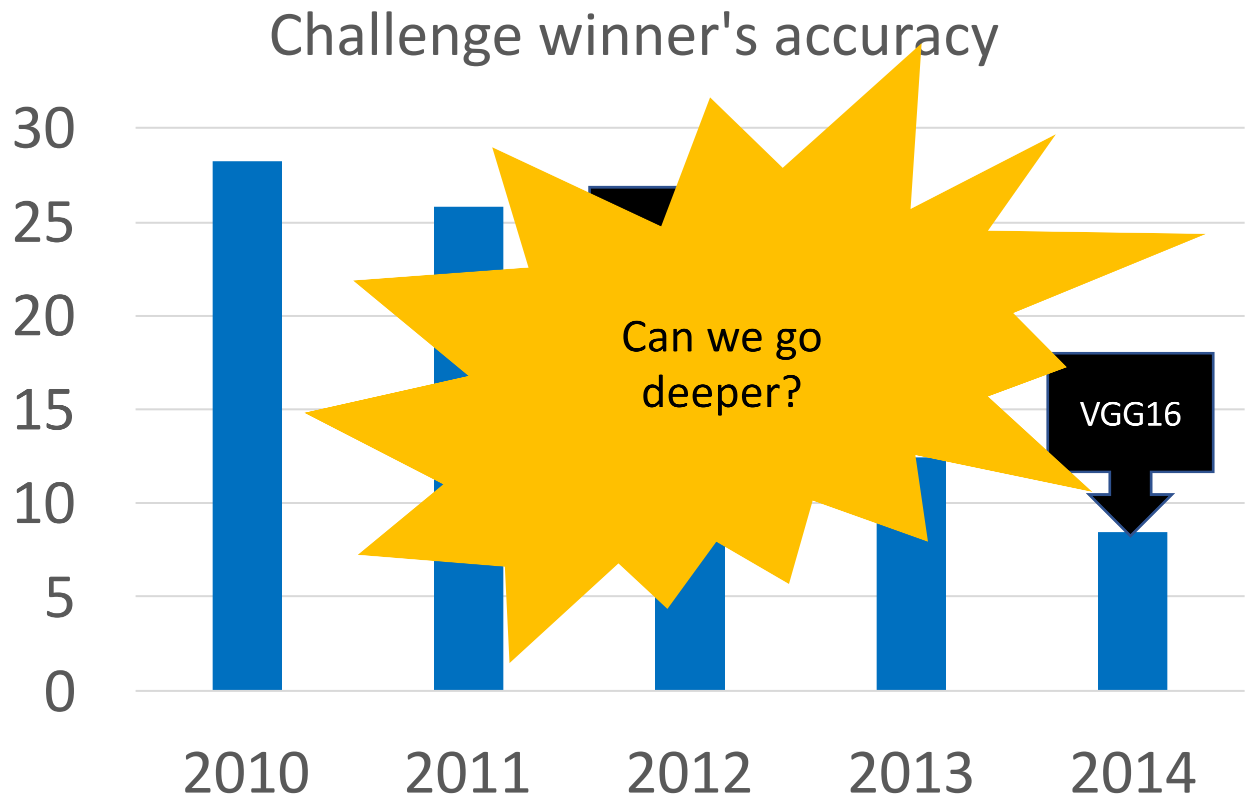
The VGG pattern

- Every convolution is 3x3, padded by 1
- Every convolution followed by ReLU
- ConvNet is divided into "stages"
  - Layers within a stage: no subsampling
  - Subsampling by 2 at the end of each stage
- Layers within stage have same number of channels
- Every subsampling → double the number of channels

Example network



Deeper is better



Is deeper better?

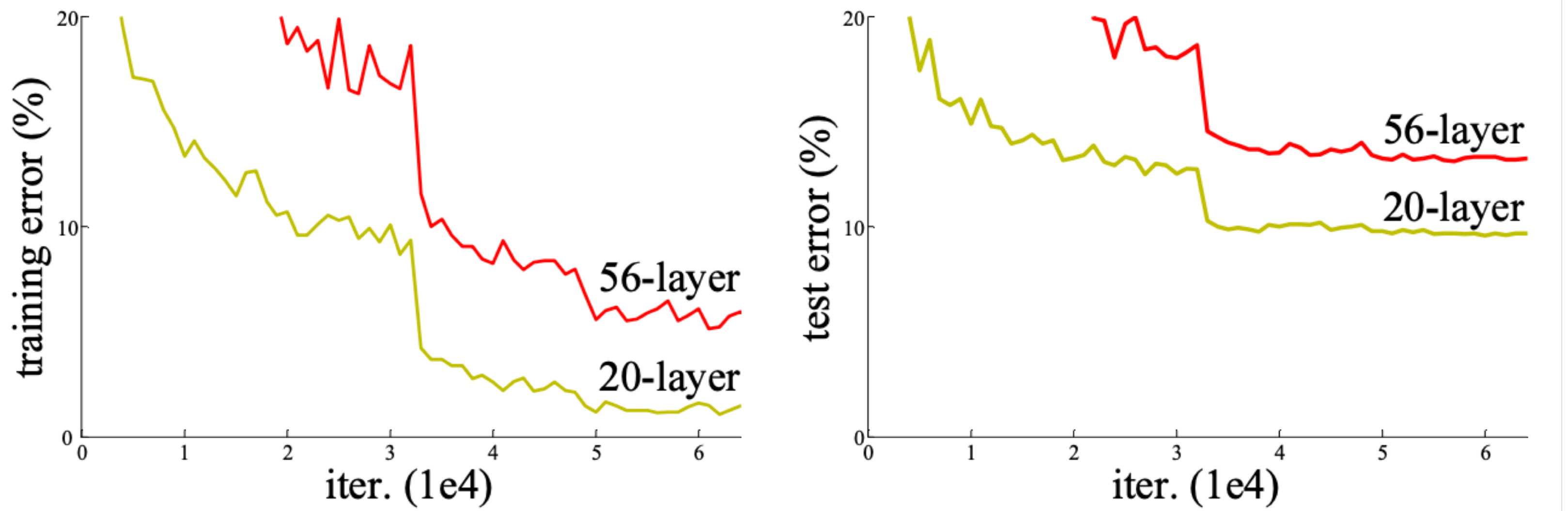


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer "plain" networks. The deeper network has higher training error, and thus test error. Similar phenomena

Challenges in training: exploding / vanishing gradients

- Vanishing / exploding gradients

$$\frac{\partial z}{\partial z_i} = \frac{\partial z}{\partial z_{n-1}} \frac{\partial z_{n-1}}{\partial z_{n-2}} \cdots \frac{\partial z_{i+1}}{\partial z_i}$$

  - If each term is (much) greater than 1 → *explosion of gradients*
  - If each term is (much) less than 1 → *vanishing gradients*

Challenges in training: noisy gradients

- Vanishing / exploding gradients

$$\frac{\partial z}{\partial z_i} = \frac{\partial z}{\partial z_{n-1}} \frac{\partial z_{n-1}}{\partial z_{n-2}} \cdots \frac{\partial z_{i+1}}{\partial z_i}$$

- Gradient for i-th layer depends on all subsequent layers
- But subsequent layers are initially random
  - Implies noisy gradients for earlier layers

Residual connections

- Instead of:

$$z_{i+1} = f_{i+1}(z_i, w_{i+1})$$

- We will have:

$$z_{i+1} = g_{i+1}(z_i, w_{i+1}) + z_i$$

With and without residual connections

- Without residual connections

$$z_{i+1} = f_{i+1}(z_i, w_{i+1})$$

$$\frac{\partial z_{i+1}}{\partial z_i} = \frac{\partial f_{i+1}(z_i, w_{i+1})}{\partial z_i}$$

$$\frac{\partial z}{\partial z_i} = \frac{\partial z}{\partial z_{i+1}} \frac{\partial f_{i+1}(z_i, w_{i+1})}{\partial z_i}$$

- With residual connections

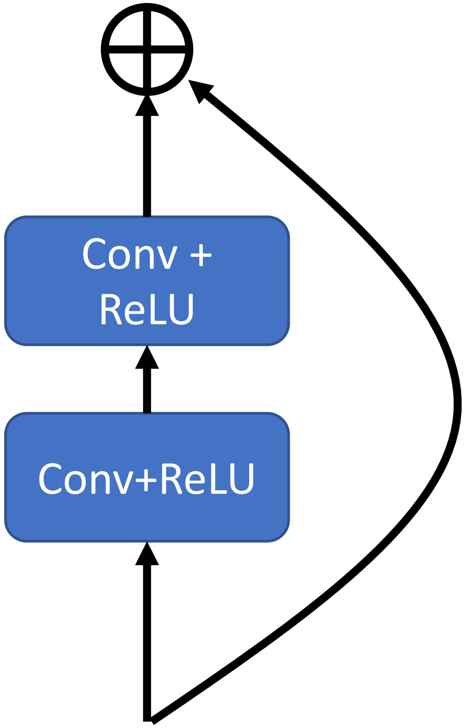
$$z_{i+1} = g_{i+1}(z_i, w_{i+1}) + z_i$$

$$\frac{\partial z_{i+1}}{\partial z_i} = \frac{\partial g_{i+1}(z_i, w_{i+1})}{\partial z_i} + I$$

$$\frac{\partial z}{\partial z_i} = \frac{\partial z}{\partial z_{i+1}} \frac{\partial g_{i+1}(z_i, w_{i+1})}{\partial z_i} + \frac{\partial z}{\partial z_{i+1}}$$

Noisy

Residual block



Residual connections

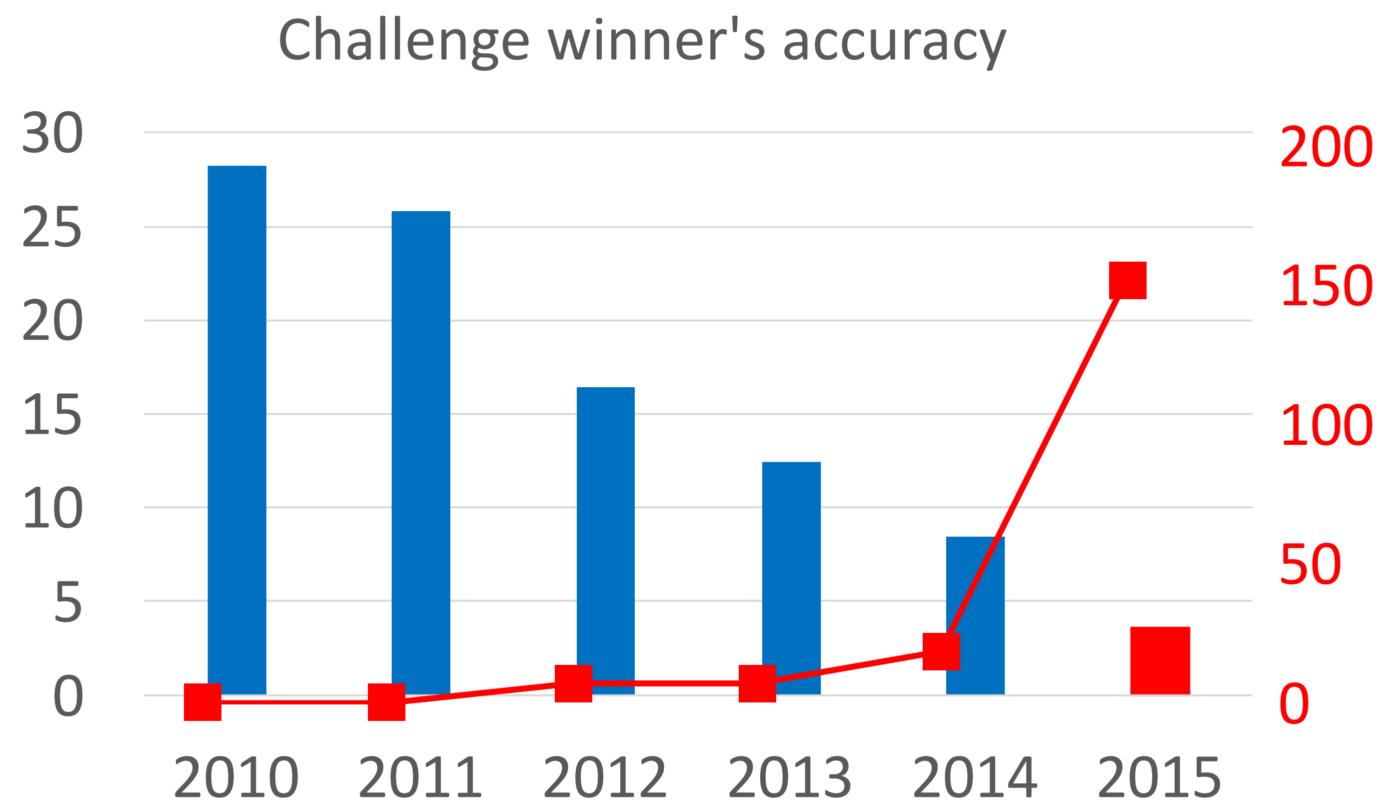
- Assumes all z_i have the same size
- True within a stage
- Across stages?
  - Doubling of feature channels
  - Subsampling
- Increase channels by 1x1 convolution
- Decrease spatial resolution by subsampling

$$z_{i+1} = g_{i+1}(z_i, w_{i+1}) + \text{subsample}(W z_i)$$

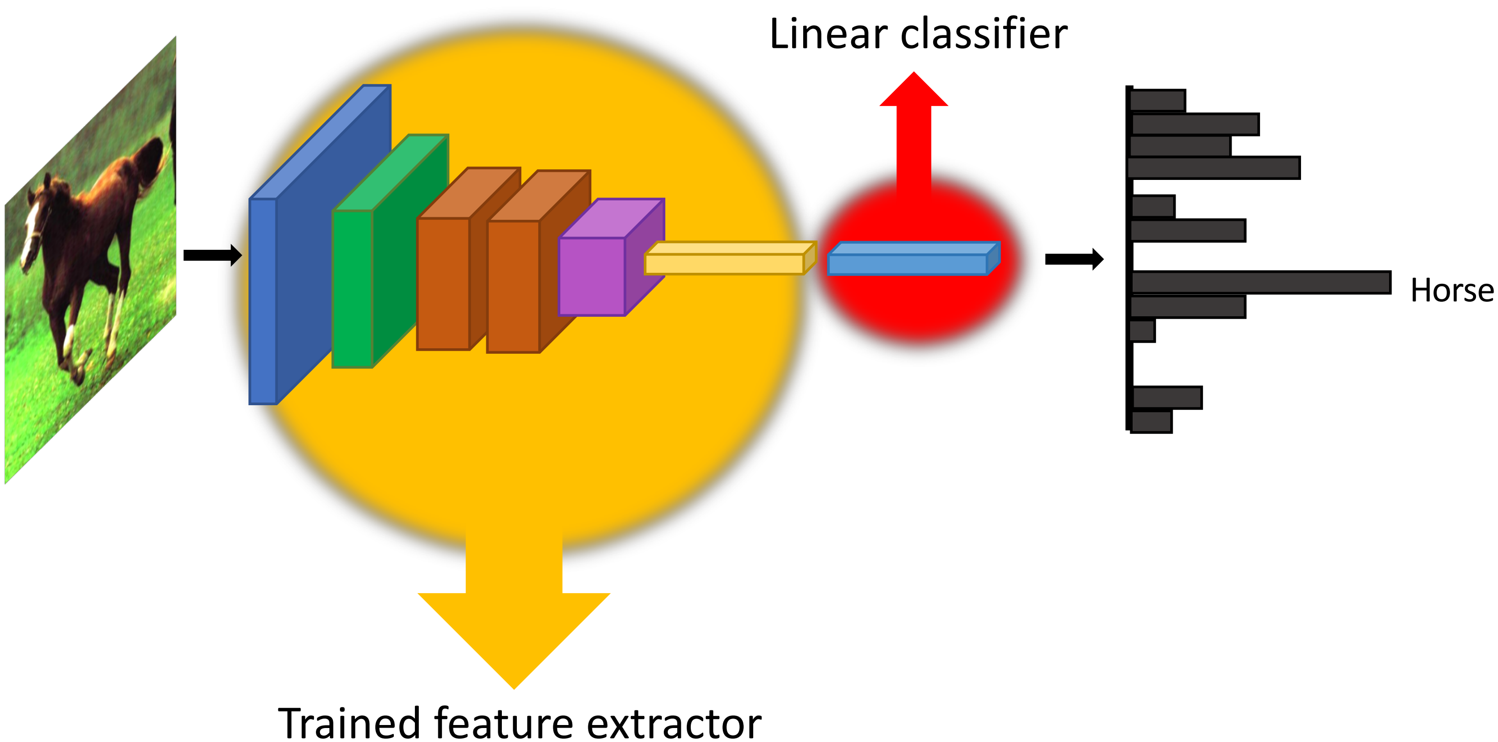
The ResNet pattern

- Decrease resolution substantially in first layer
  - Reduces memory consumption due to intermediate outputs
- Divide into stages
  - maintain resolution, channels in each stage
  - halve resolution, double channels between stages
- Divide each stage into residual blocks
- At the end, compute average value of each channel to feed linear classifier

Putting it all together - Residual networks

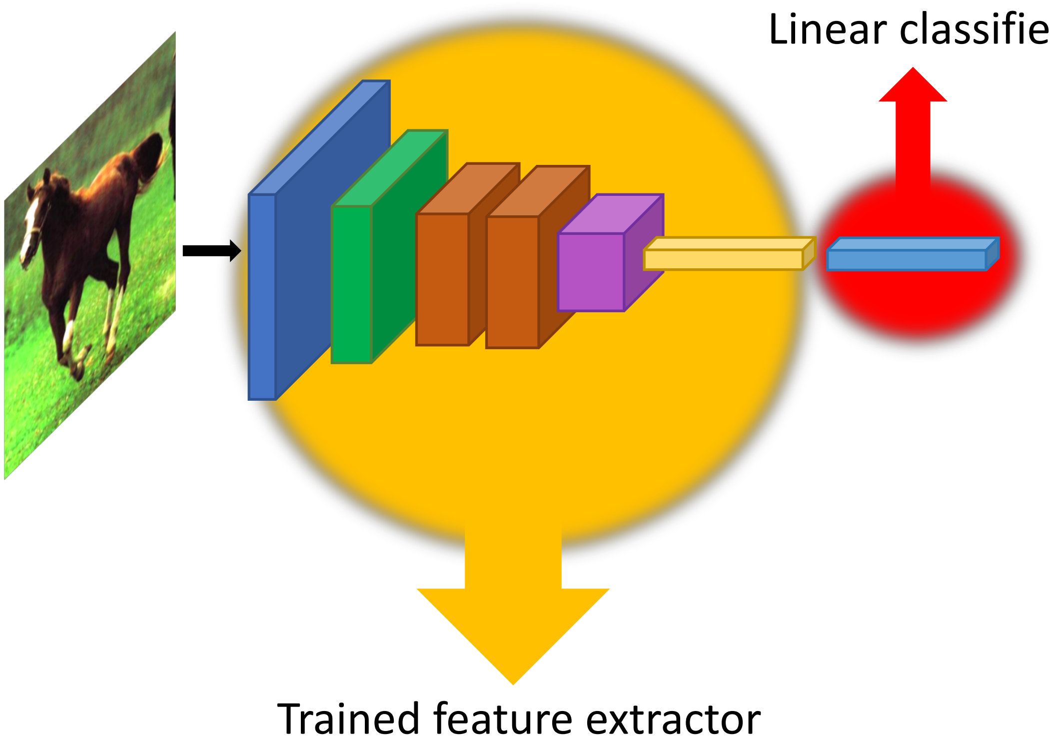


Transfer learning with convolutional networks



Transfer learning with convolutional networks

- What do we do for a new image classification problem?
- Key idea:
  - *Freeze* parameters in feature extractor
  - *Retrain* classifier



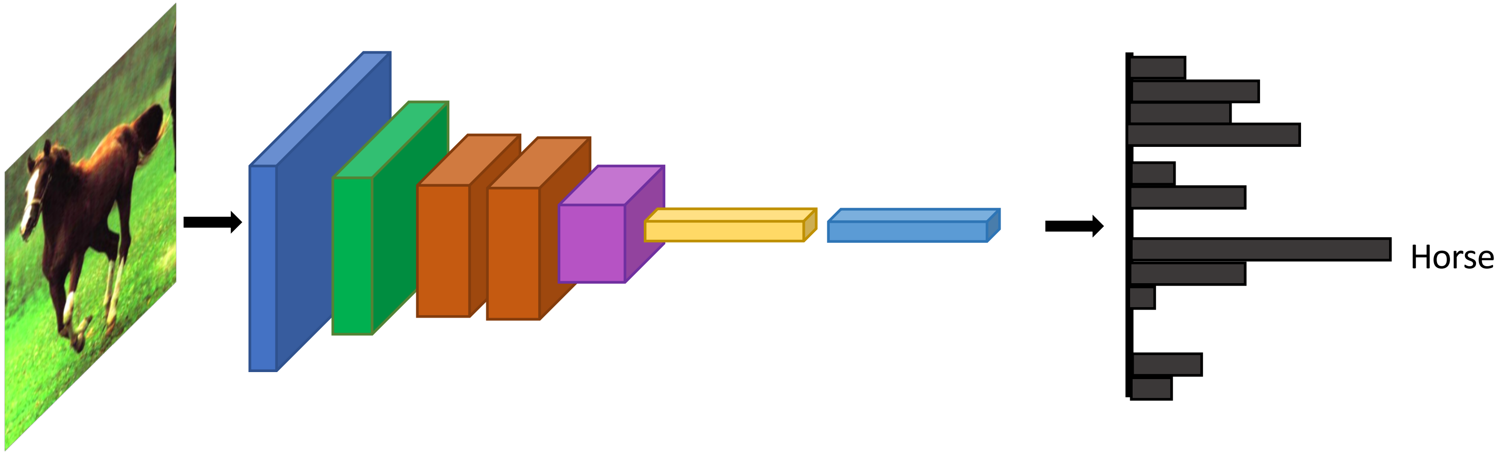
Transfer learning with convolutional networks

Dataset	Best Non-Convnet perf	Pretrained convnet + classifier	Improvement
Caltech 101	84.3	87.7	+3.4
VOC 2007	61.7	79.7	+18
CUB 200	18.8	61.0	+42.2
Aircraft	61.0	45.0	-16
Cars	59.2	36.5	-22.7

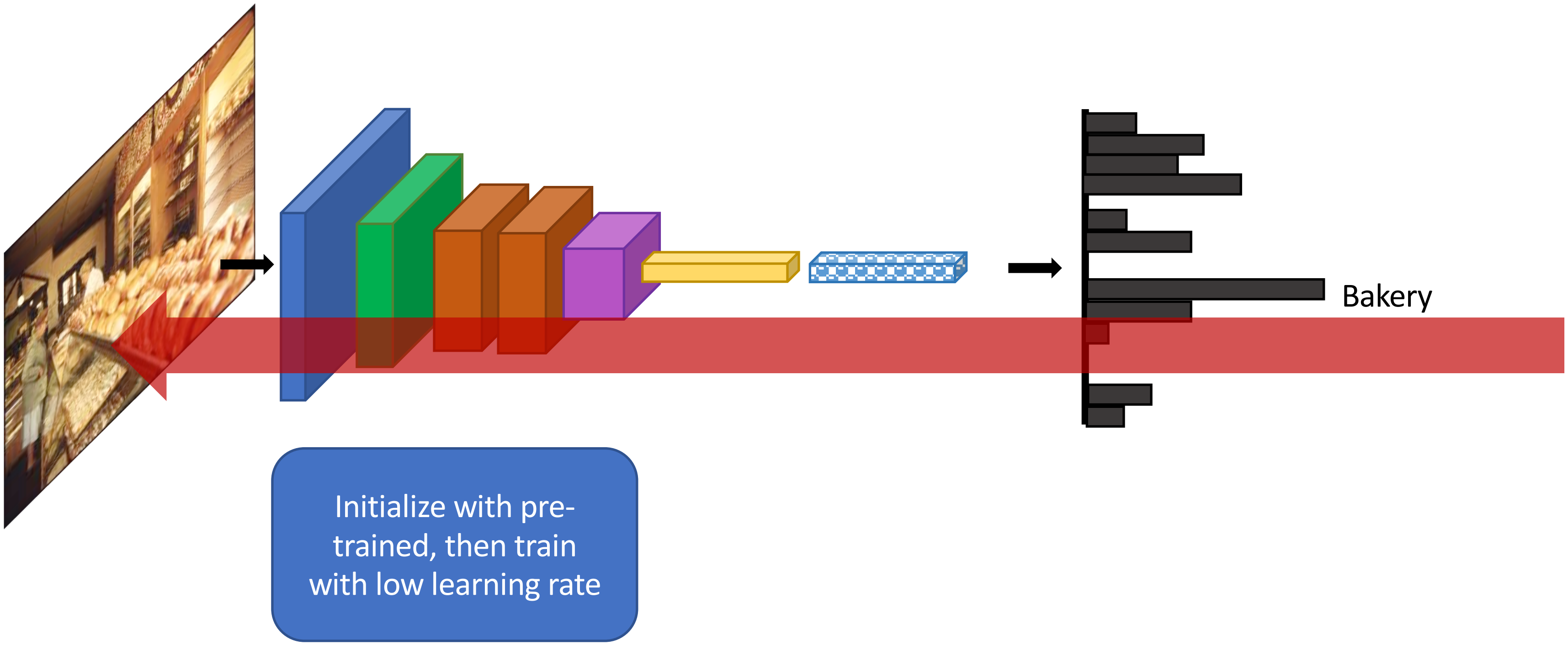
Why transfer learning?

- Availability of training data
- Computational cost
- Ability to pre-compute feature vectors and use for multiple tasks
- *Con: NO end-to-end learning*

Finetuning



Finetuning



Bakery
Initialize with pre-trained, then train with low learning rate

Finetuning

Dataset	Best Non-Convnet perf	Pretrained convnet + classifier	Finetuned convnet	Improvement
Caltech 101	84.3	87.7	88.4	+4.1
VOC 2007	61.7	79.7	82.4	+20.7
CUB 200	18.8	61.0	70.4	+51.6
Aircraft	61.0	45.0	74.1	+13.1
Cars	59.2	36.5	79.8	+20.6