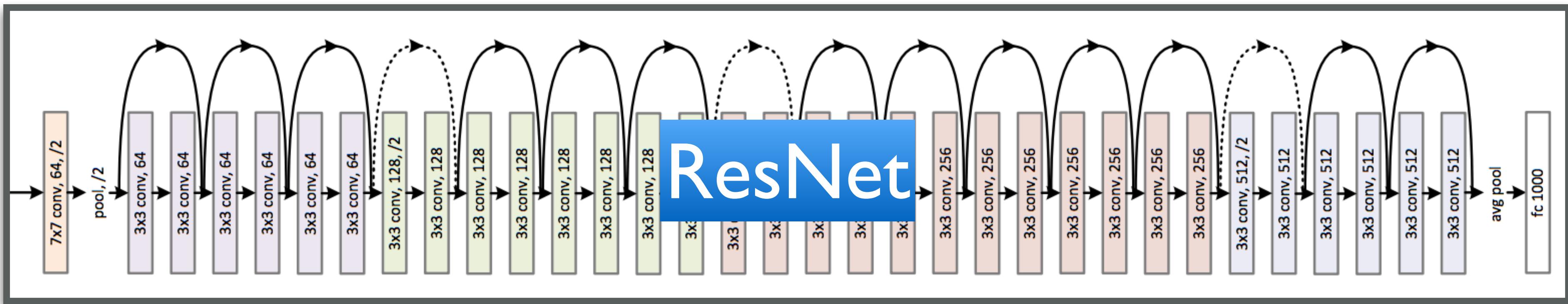
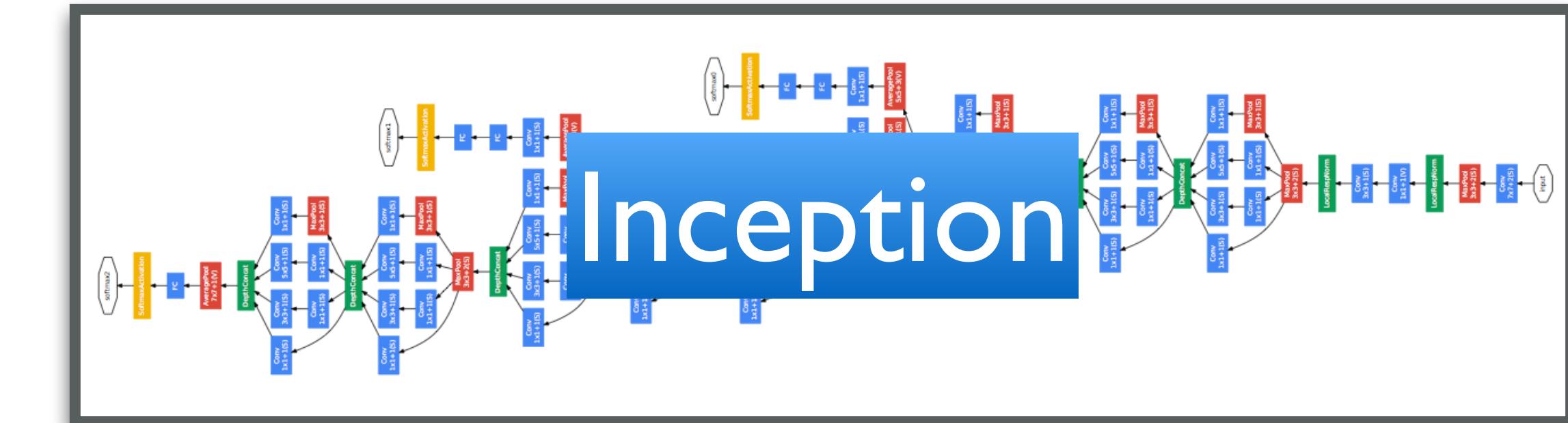
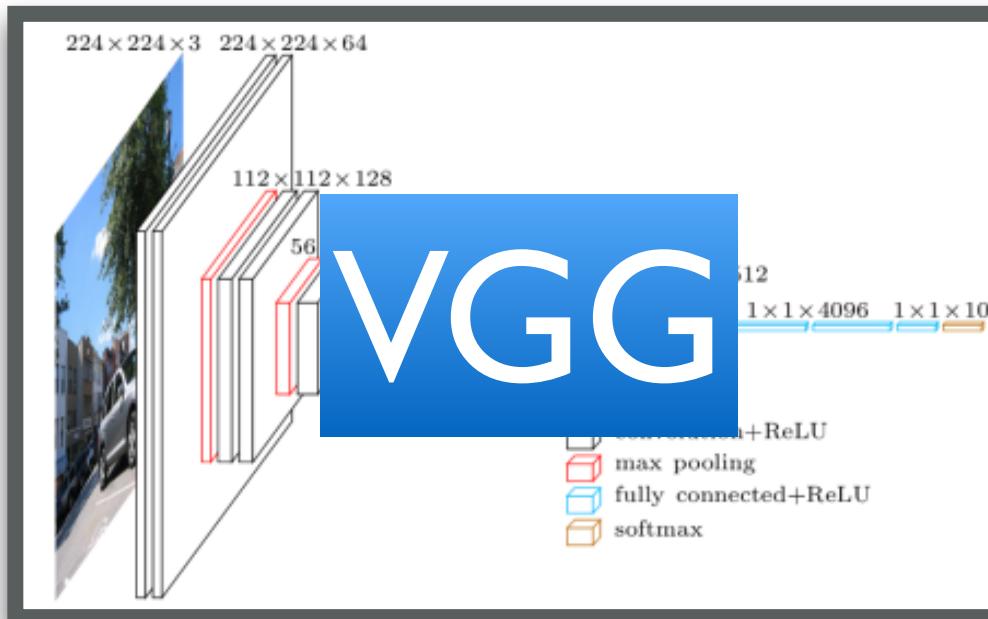
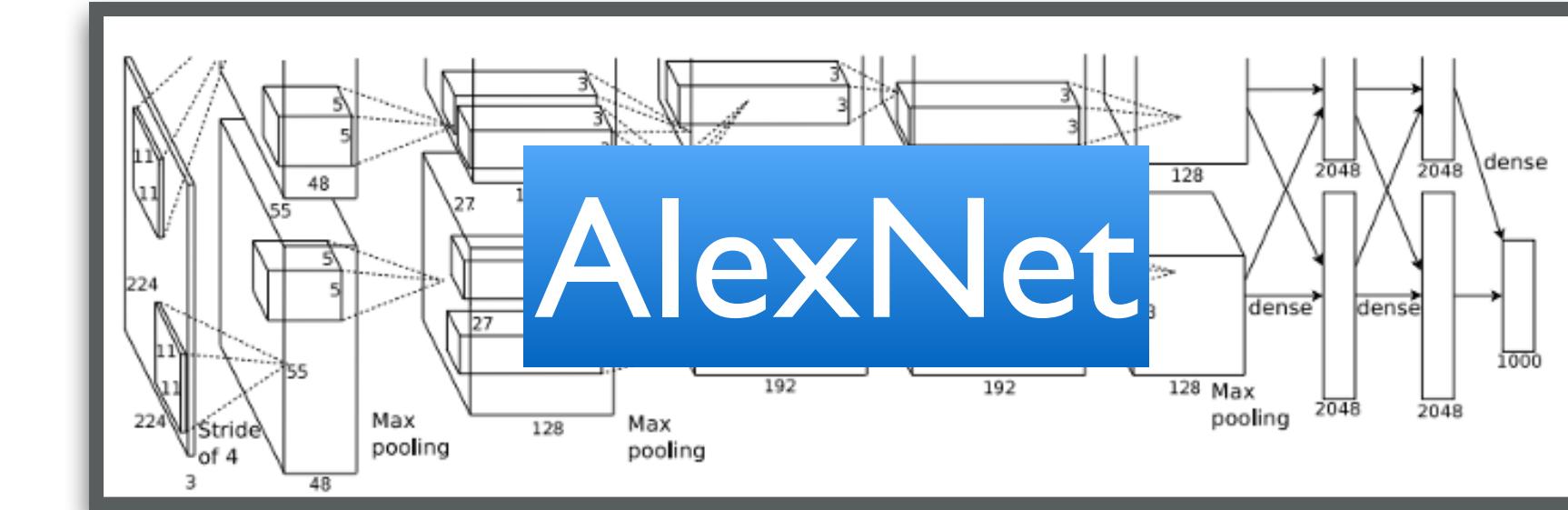
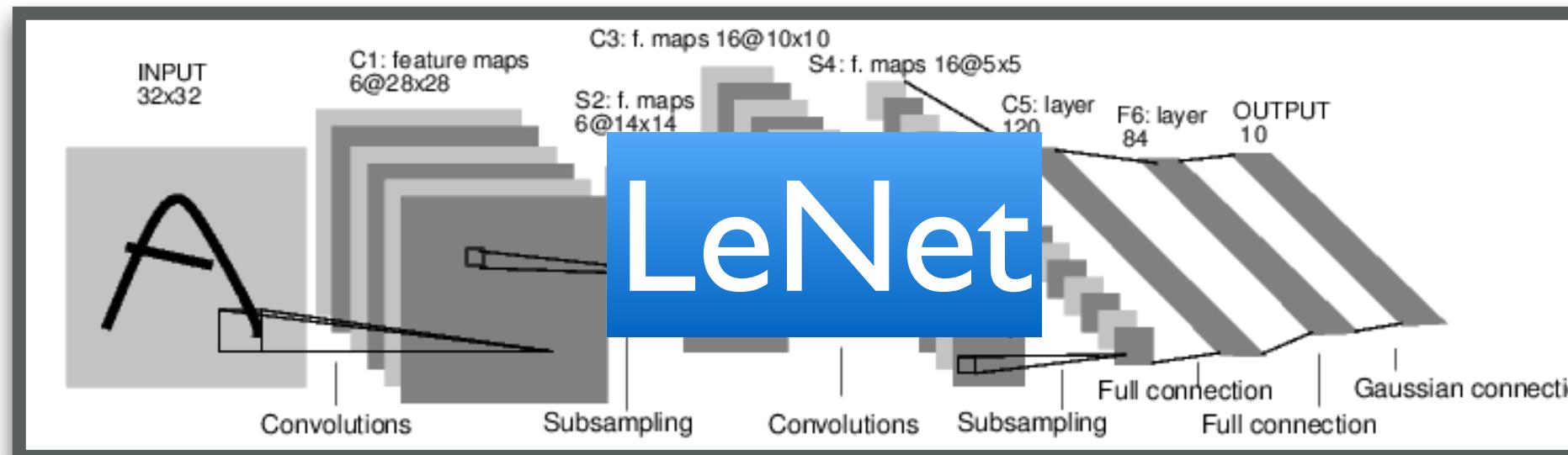


Introduction to Computer Vision: Convolutional Networks (Part 3)

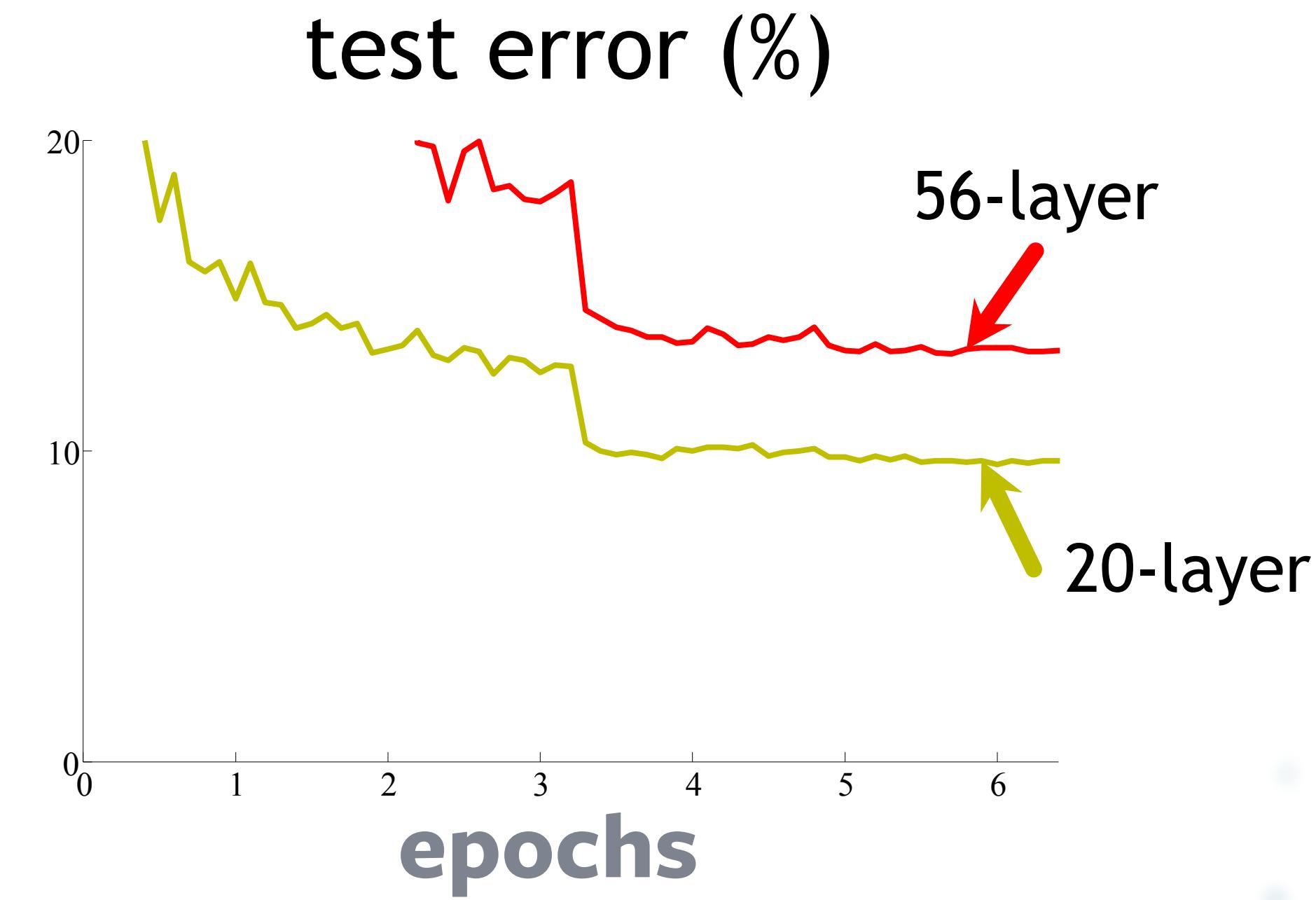
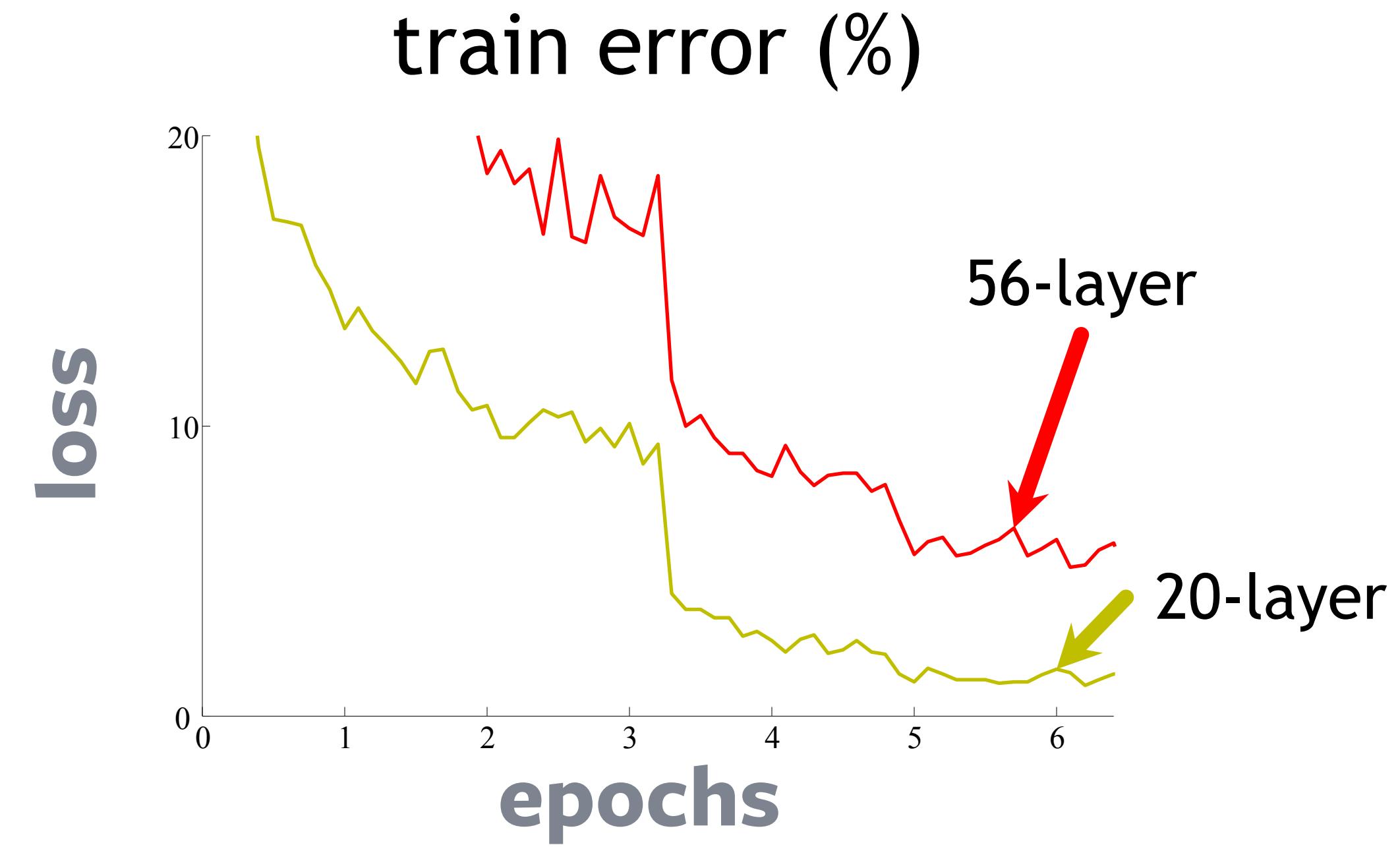
Laurens van der Maaten

Convolutional networks



Stacking layers

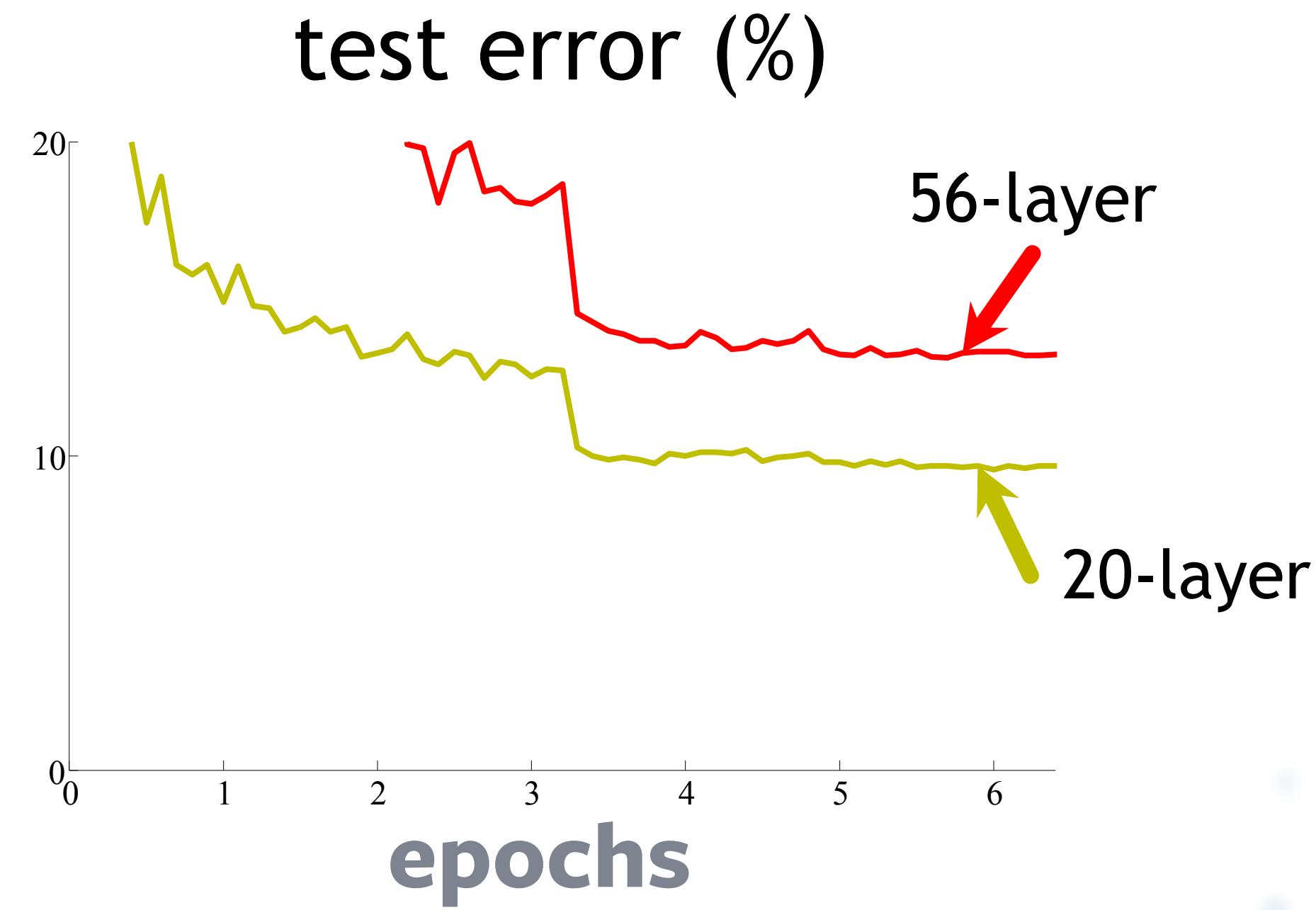
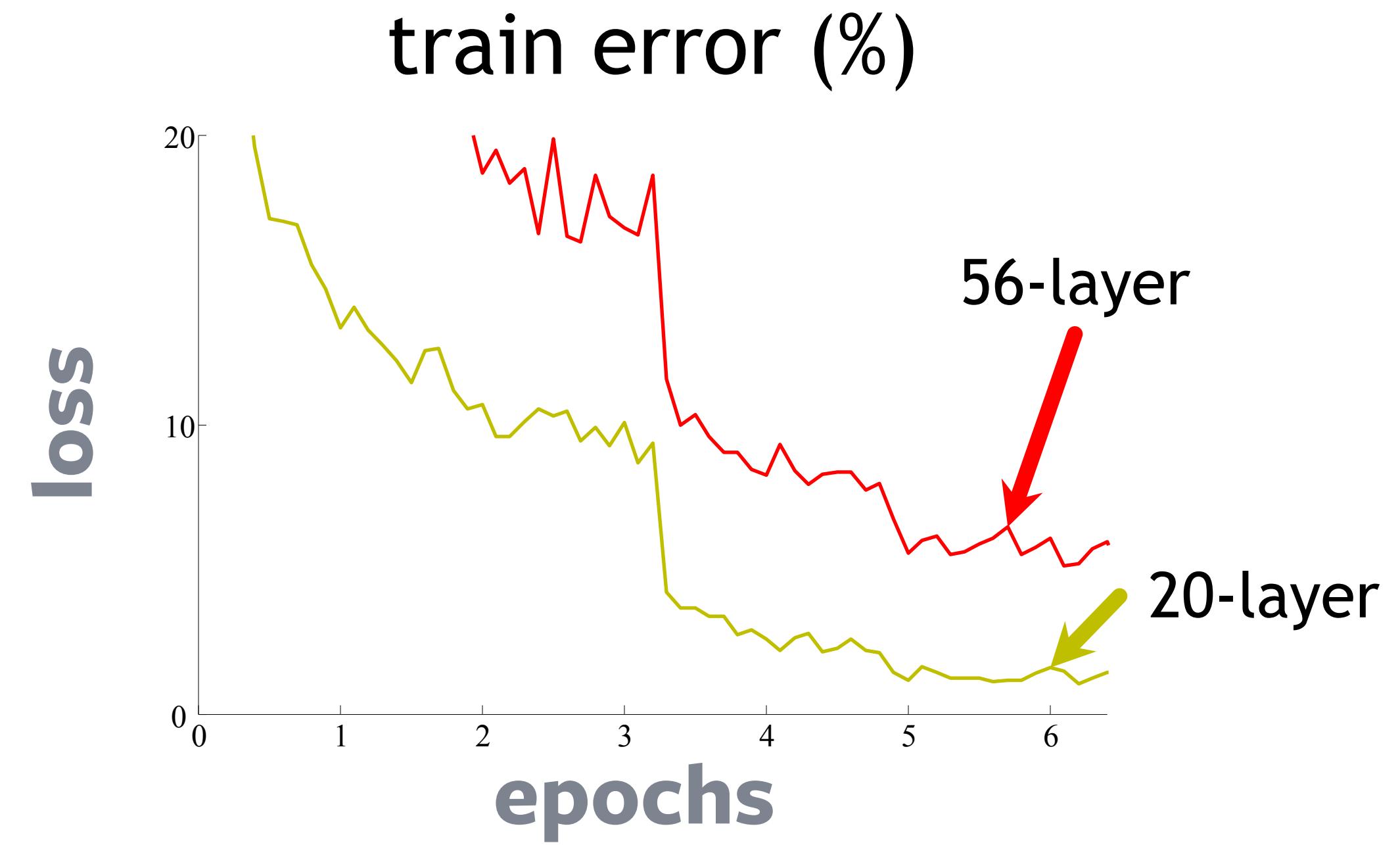
- Simple experiment with stacking many 3x3 convolutional blocks:



* Figure credit: Kaiming He

Stacking layers

- Simple experiment with stacking many 3x3 convolutional blocks:



- Does this suggest overfitting?

* Figure credit: Kaiming He

Stacking layers

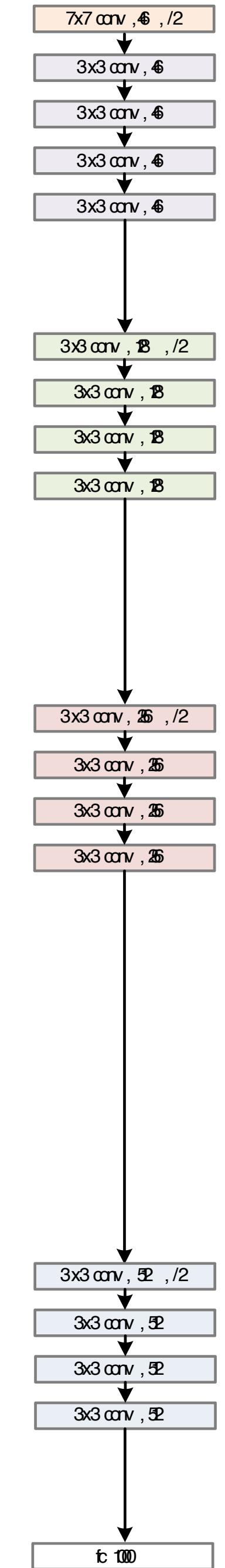
- Deeper models should not have higher training error

* Figure credit: Kaiming He

Stacking layers

"shallow"
model

- Deeper models should not have higher training error
- Solution by construction:
 - Train shallow model

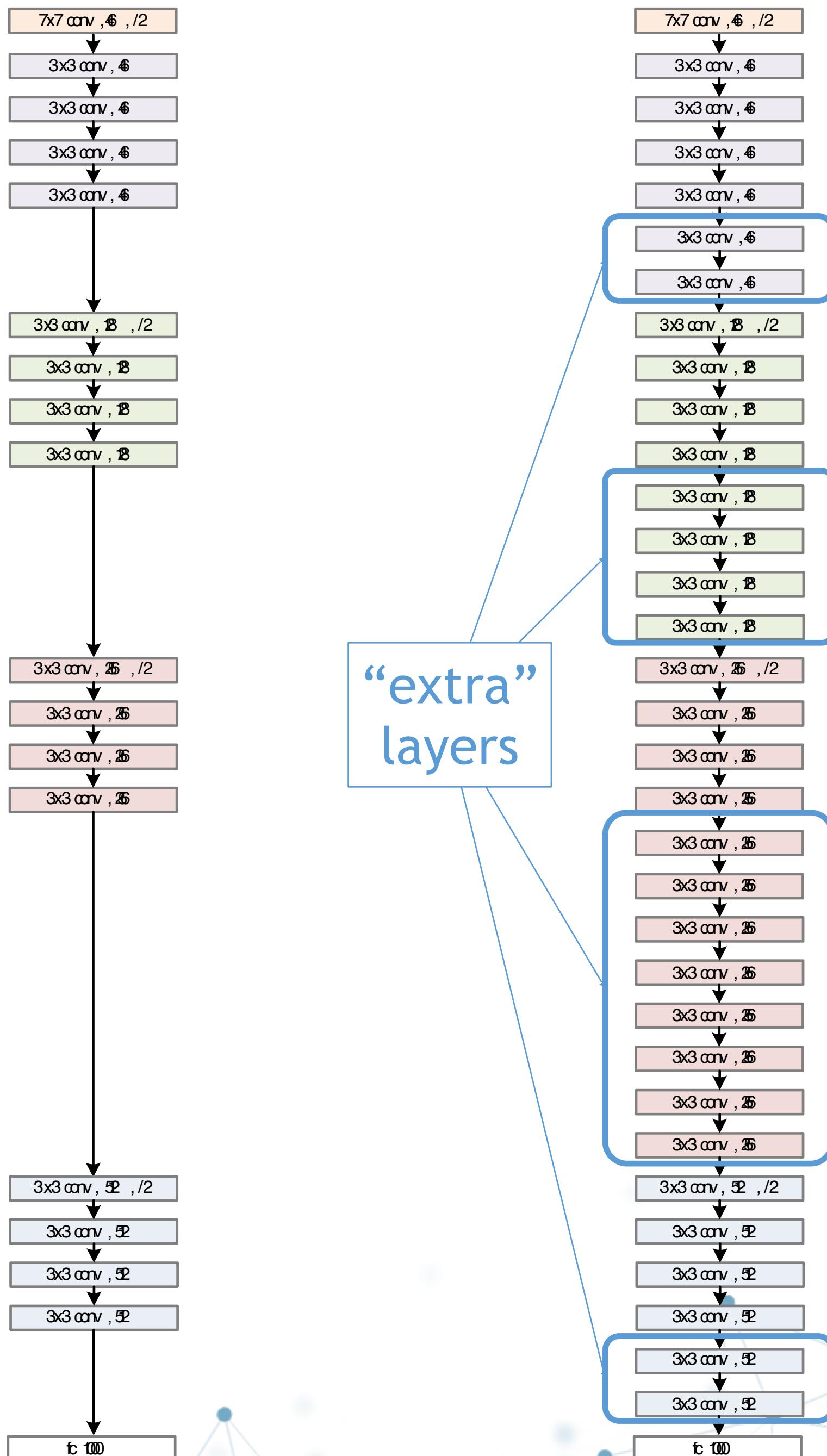


* Figure credit: Kaiming He

Stacking layers

“shallow”
model

- Deeper models should not have higher training error
- Solution by construction:
 - Train shallow model
 - Add additional layers set to identity
 - Train the deeper model further



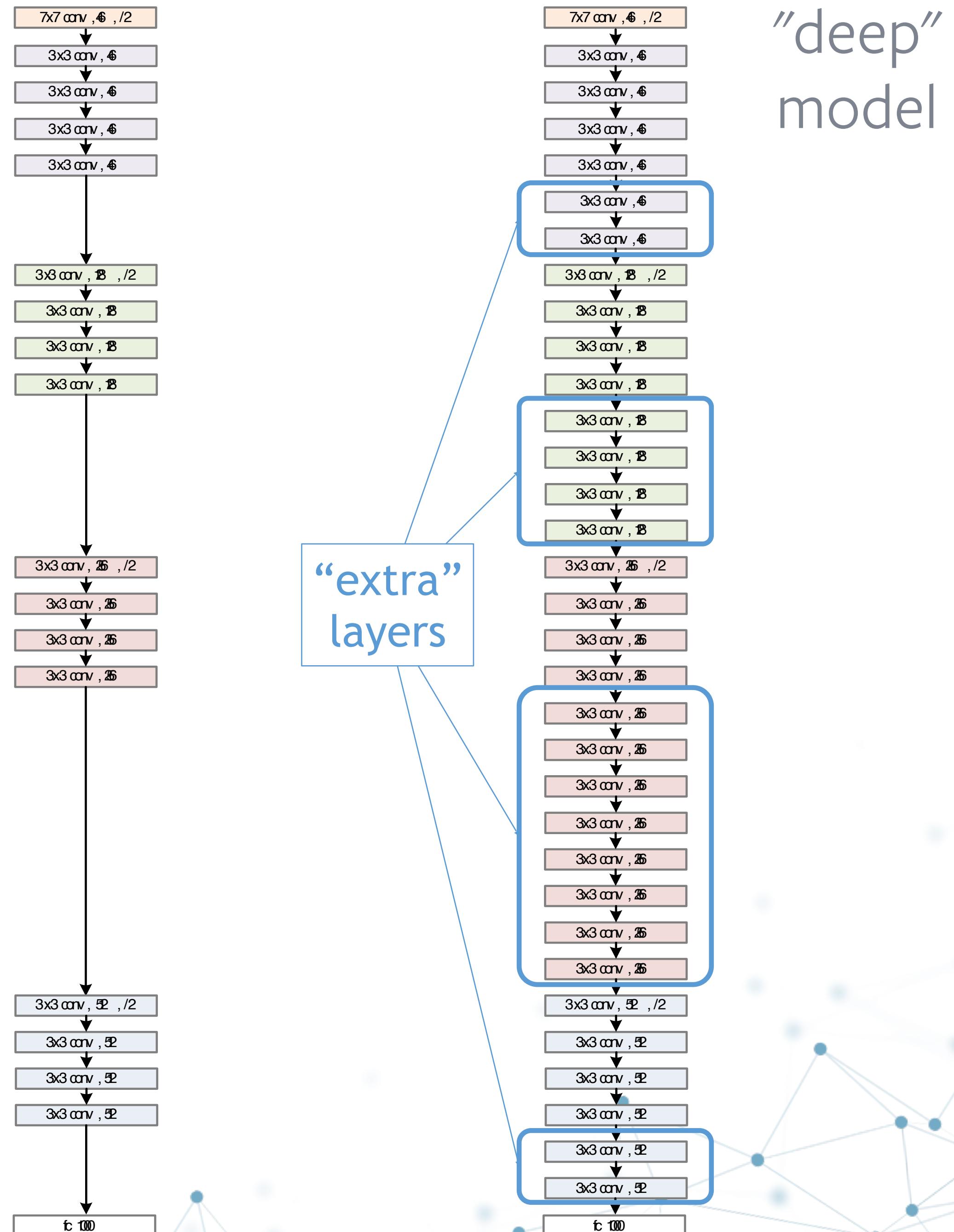
“deep”
model

* Figure credit: Kaiming He

Stacking layers

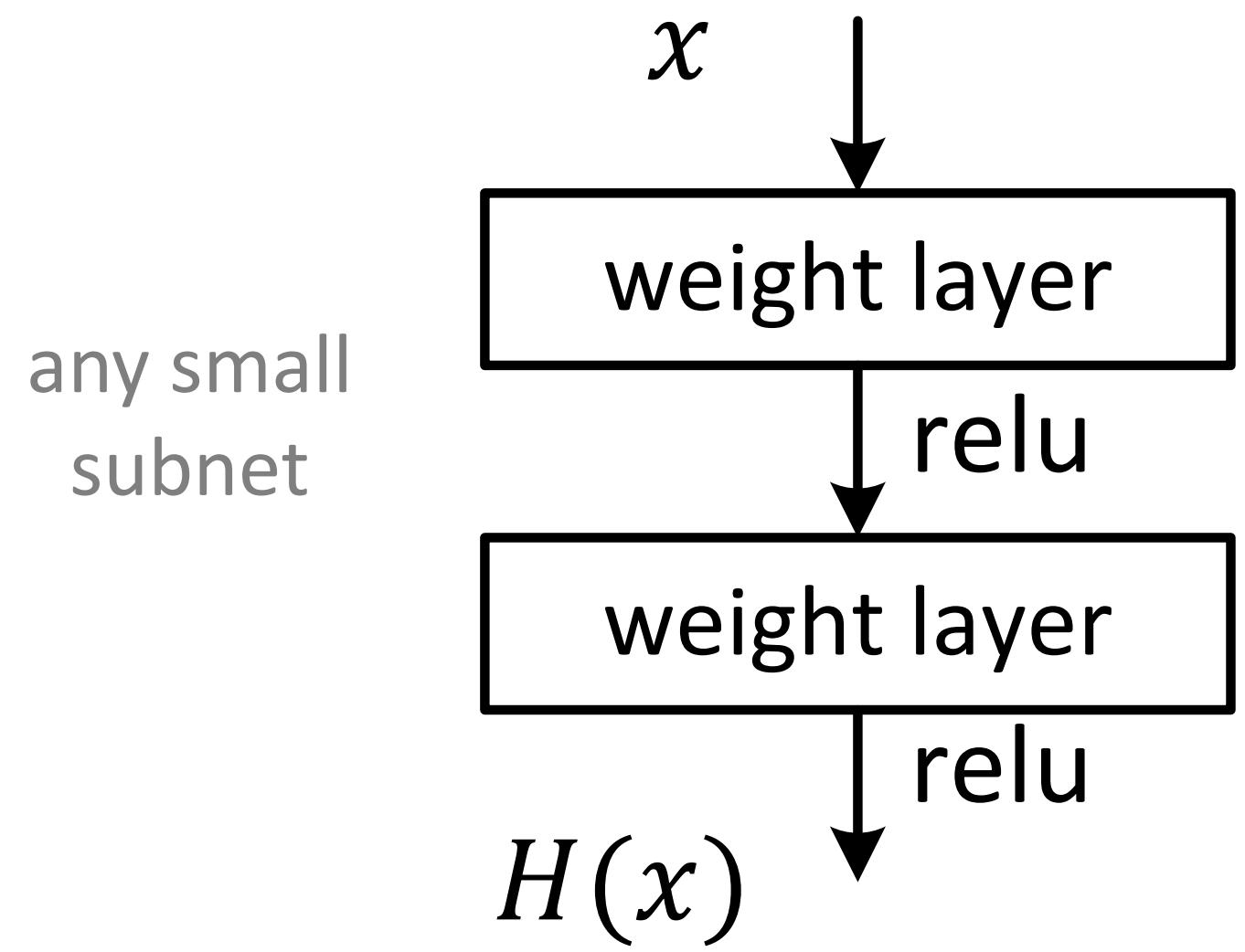
“shallow”
model

- Deeper models should not have higher training error
- Solution by construction:
 - Train shallow model
 - Add additional layers set to identity
 - Train the deeper model further
- Learning does not work well :(



Residual connections

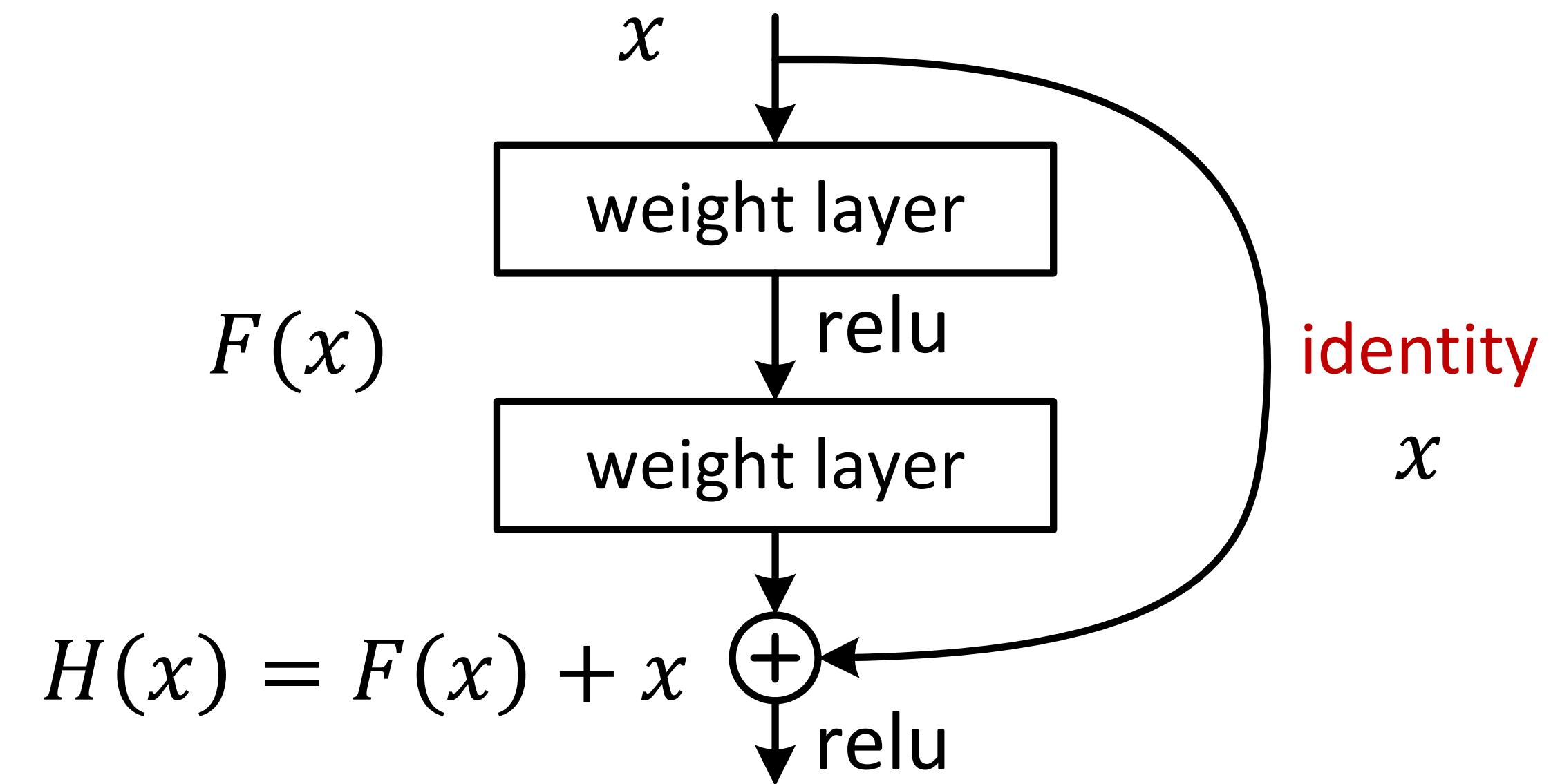
- Block architecture that achieves similar goal
- Take any “regular” network block...



* Figure credit: Kaiming He

Residual connections

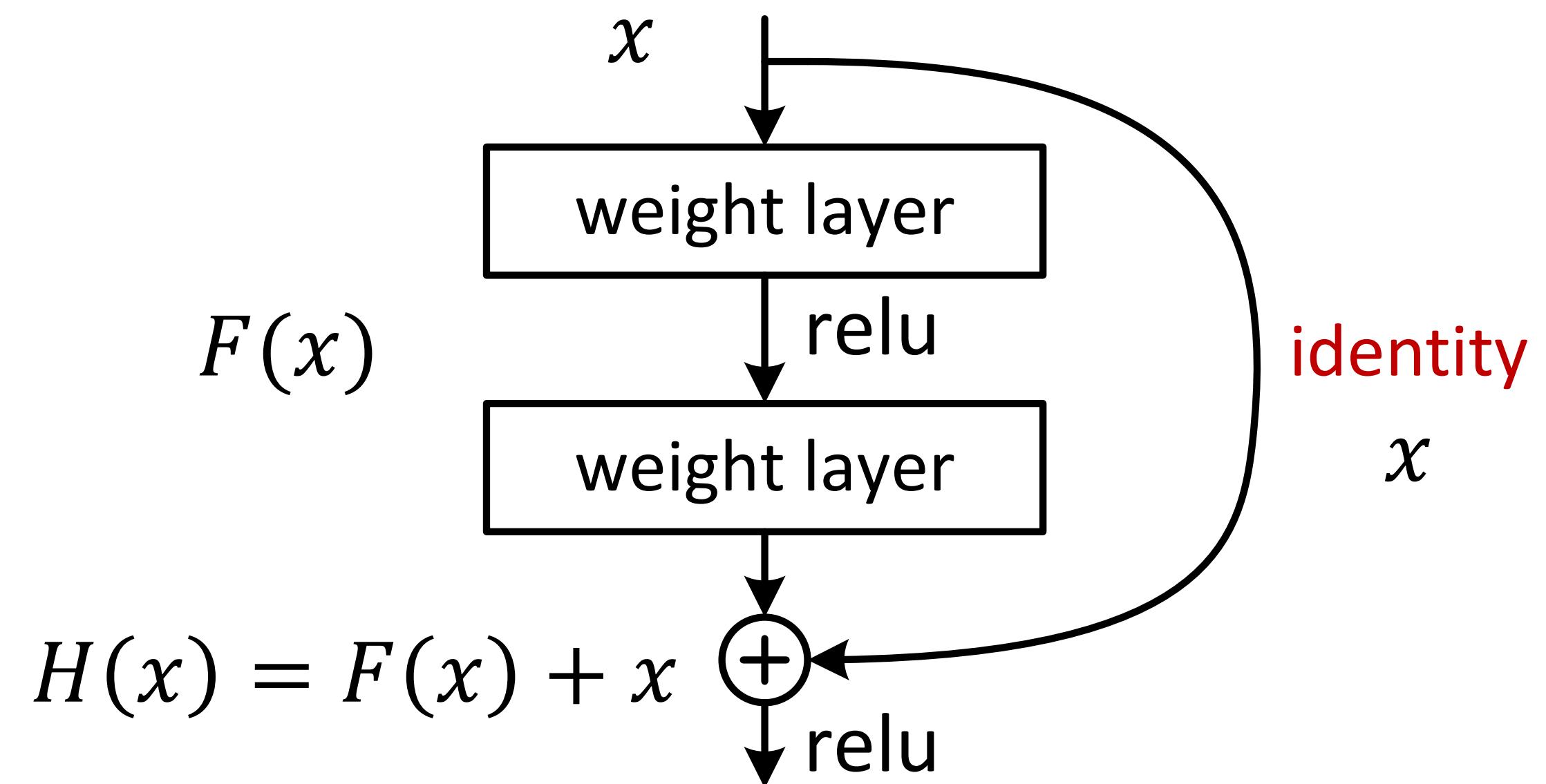
- Block architecture that achieves similar goal
- Take any “regular” network block... and make it **residual**:



* Figure credit: Kaiming He

Residual connections

- Block architecture that achieves similar goal
- Take any “regular” network block... and make it **residual**:

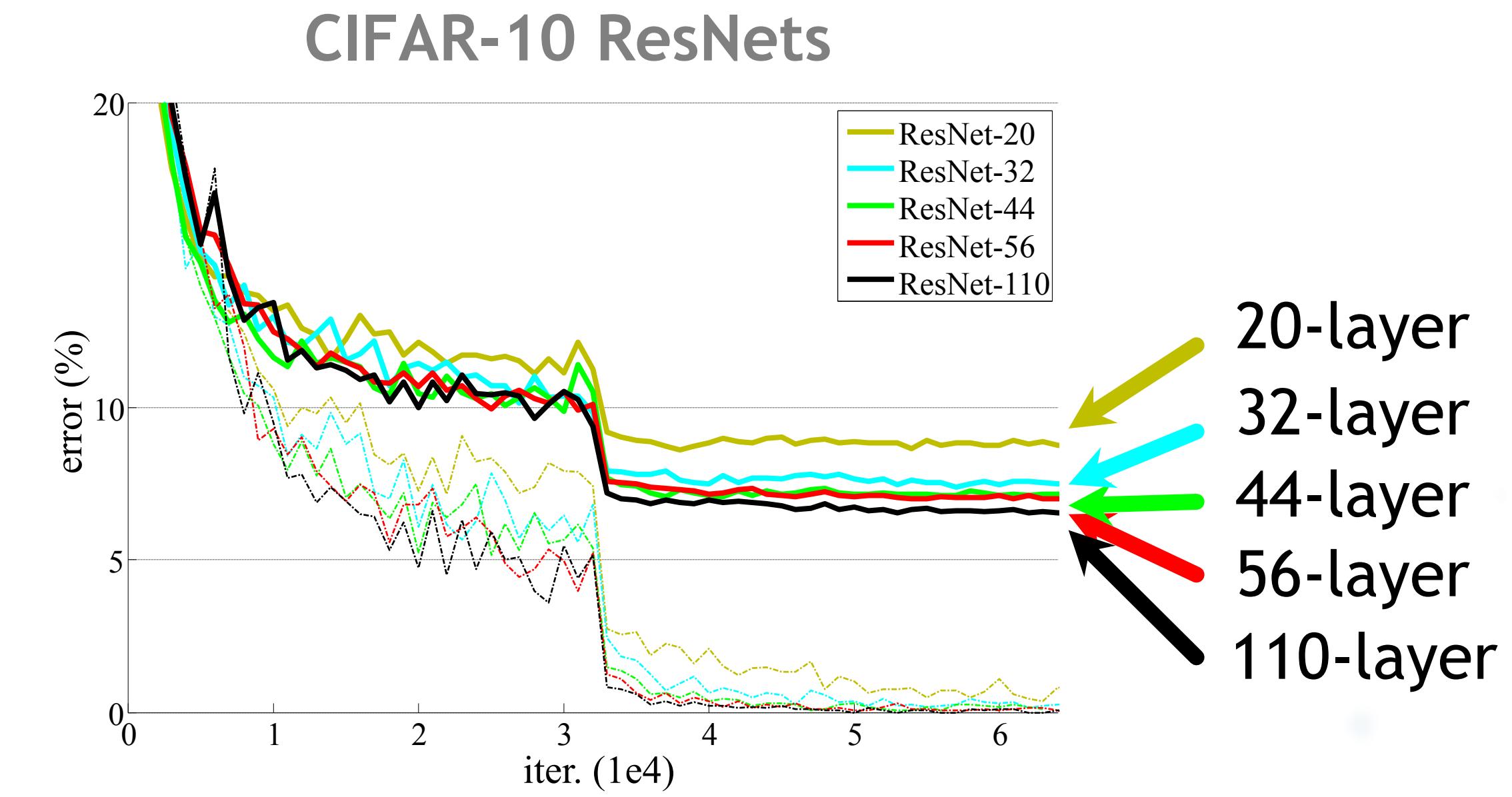
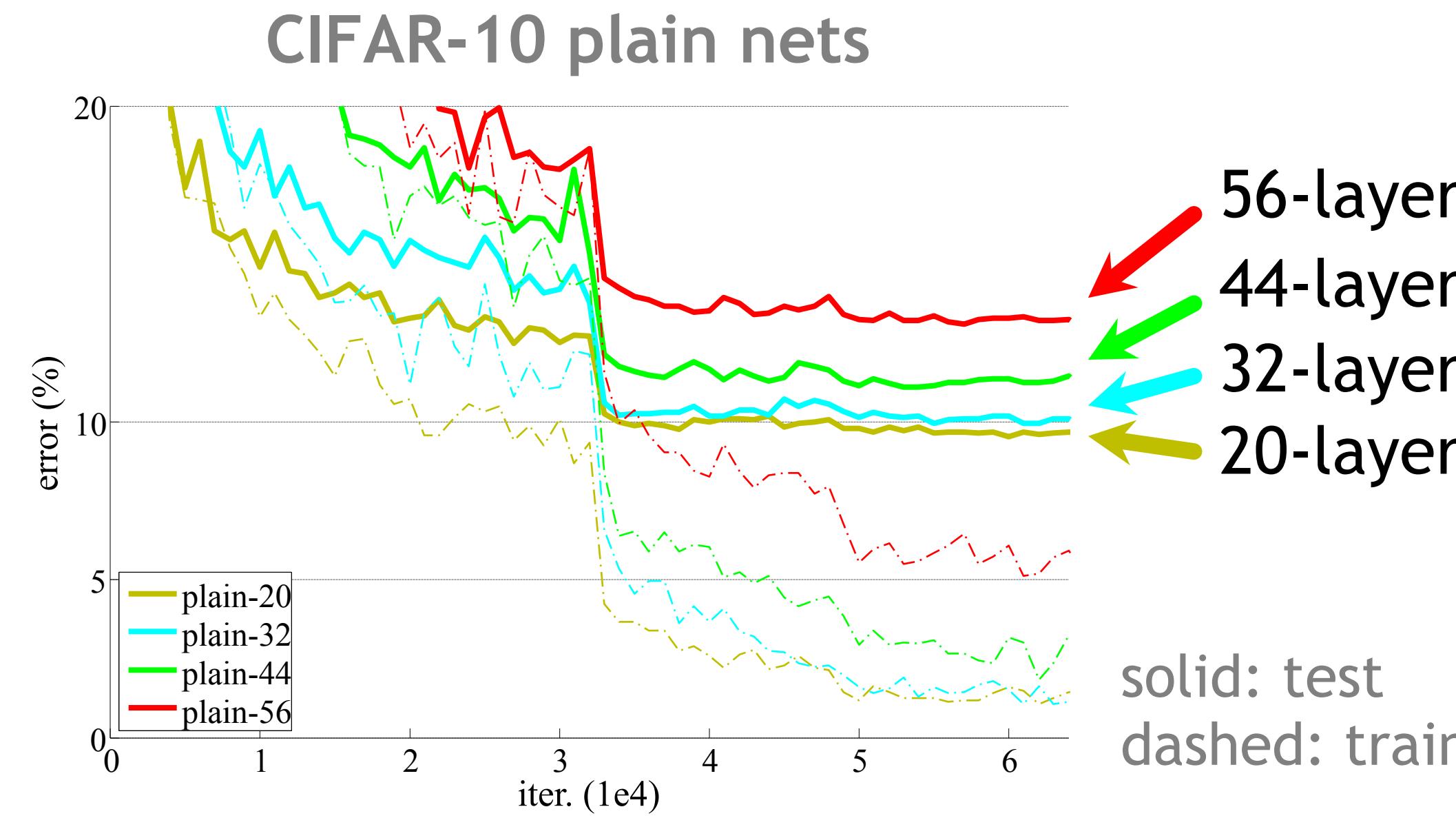


- Initializing weights to zero achieves the desired effect

* Figure credit: Kaiming He

Residual connections

- Deeper models can now be trained without problems:

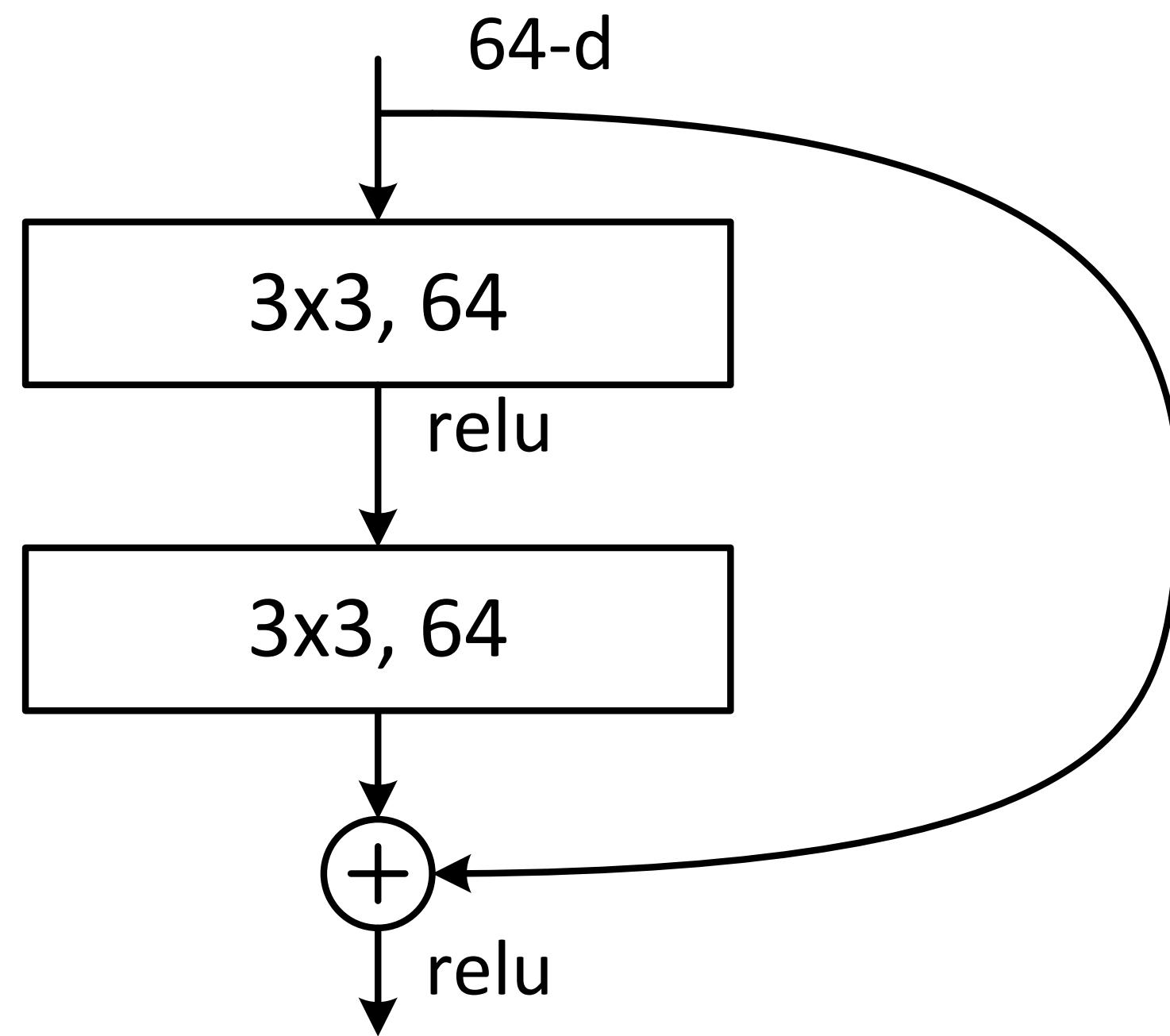


- Overfitting may still happen, but at least training error does not go up

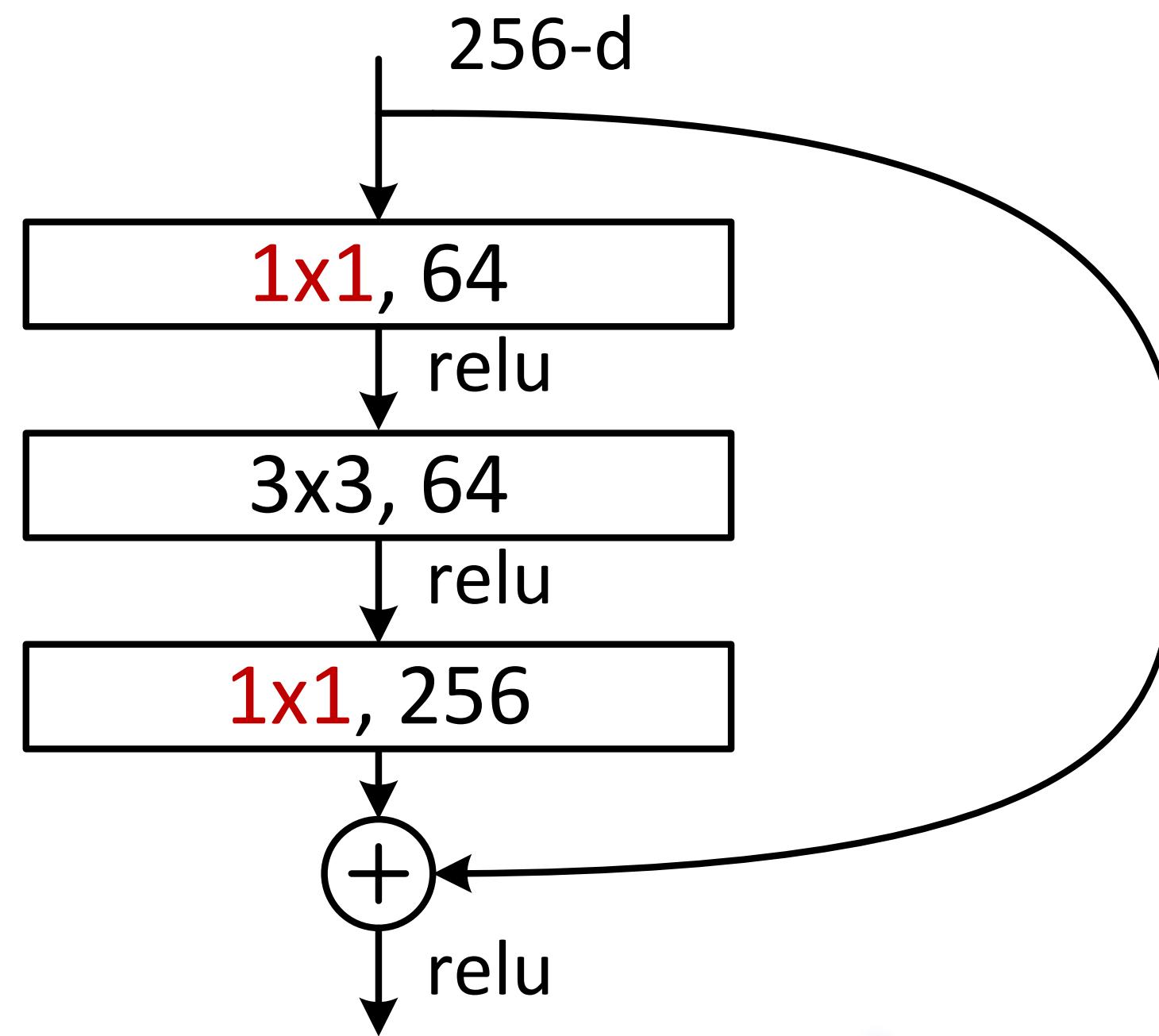
* Figure credit: Kaiming He

Residual networks

- In practice, residual networks (ResNet) use the following block design:



simple design

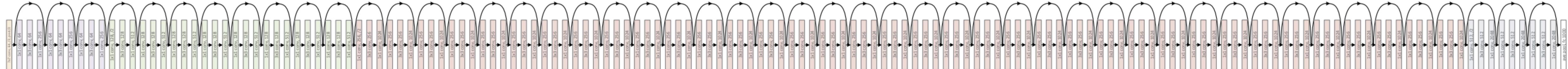


bottleneck design

* Figure credit: Kaiming He

Residual networks

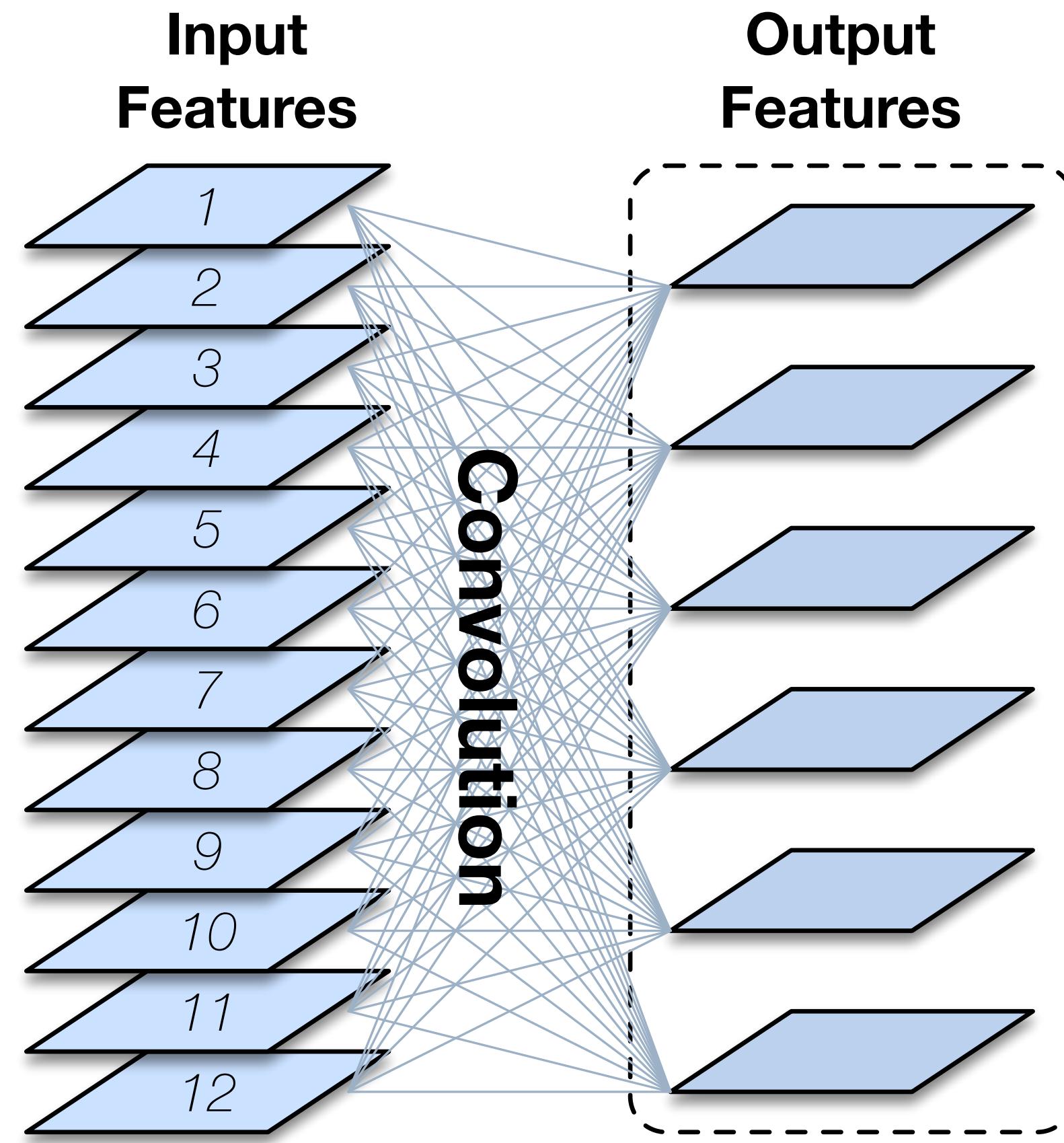
- Full model has pooling layers for downsampling
- All pooling layers do max-pooling but the last one does average-pooling



- Every time the image size halves, the number of channels is doubled

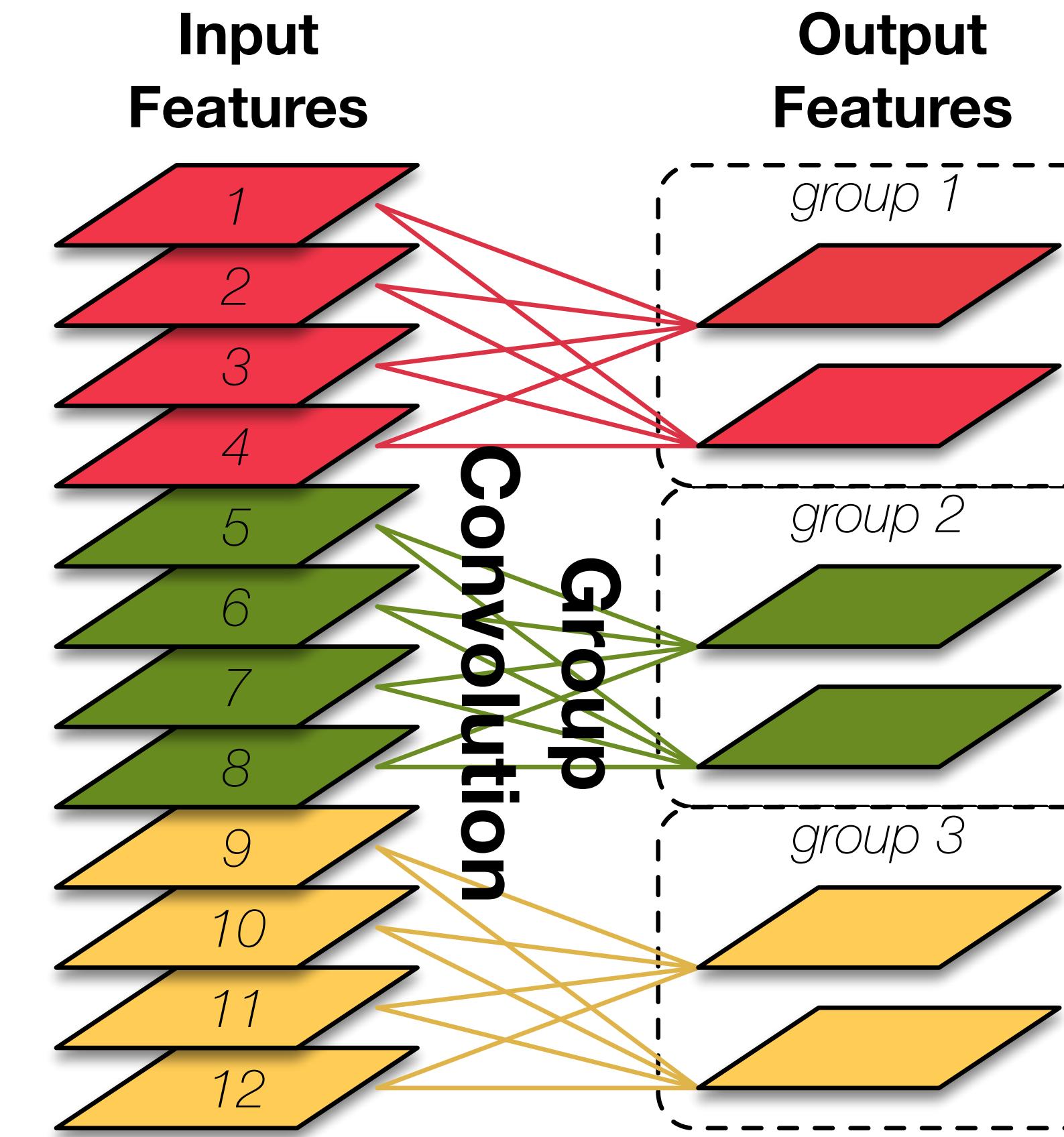
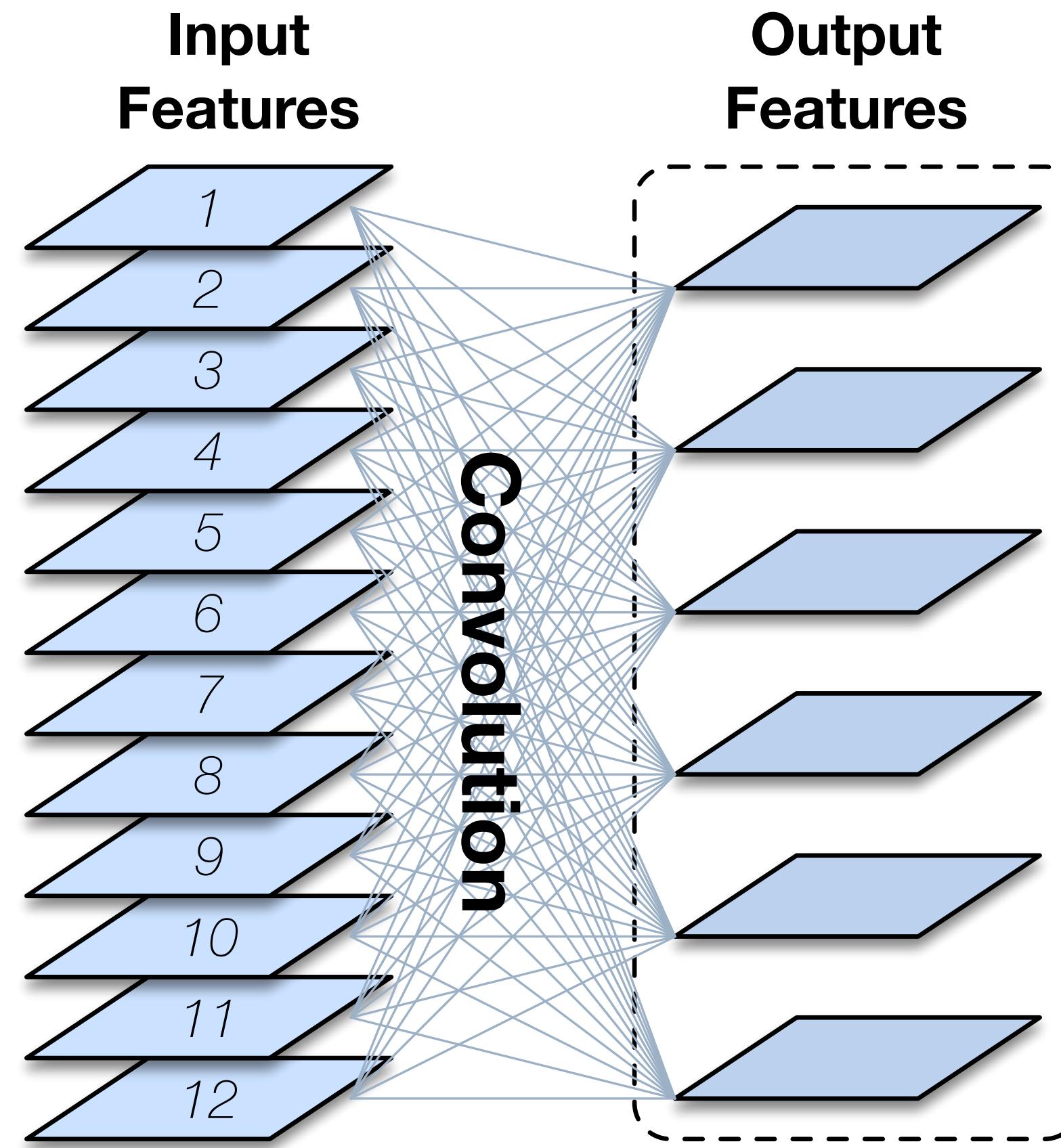
Group Convolutions

- In **group convolutions**, not all input channels feed into all output channels



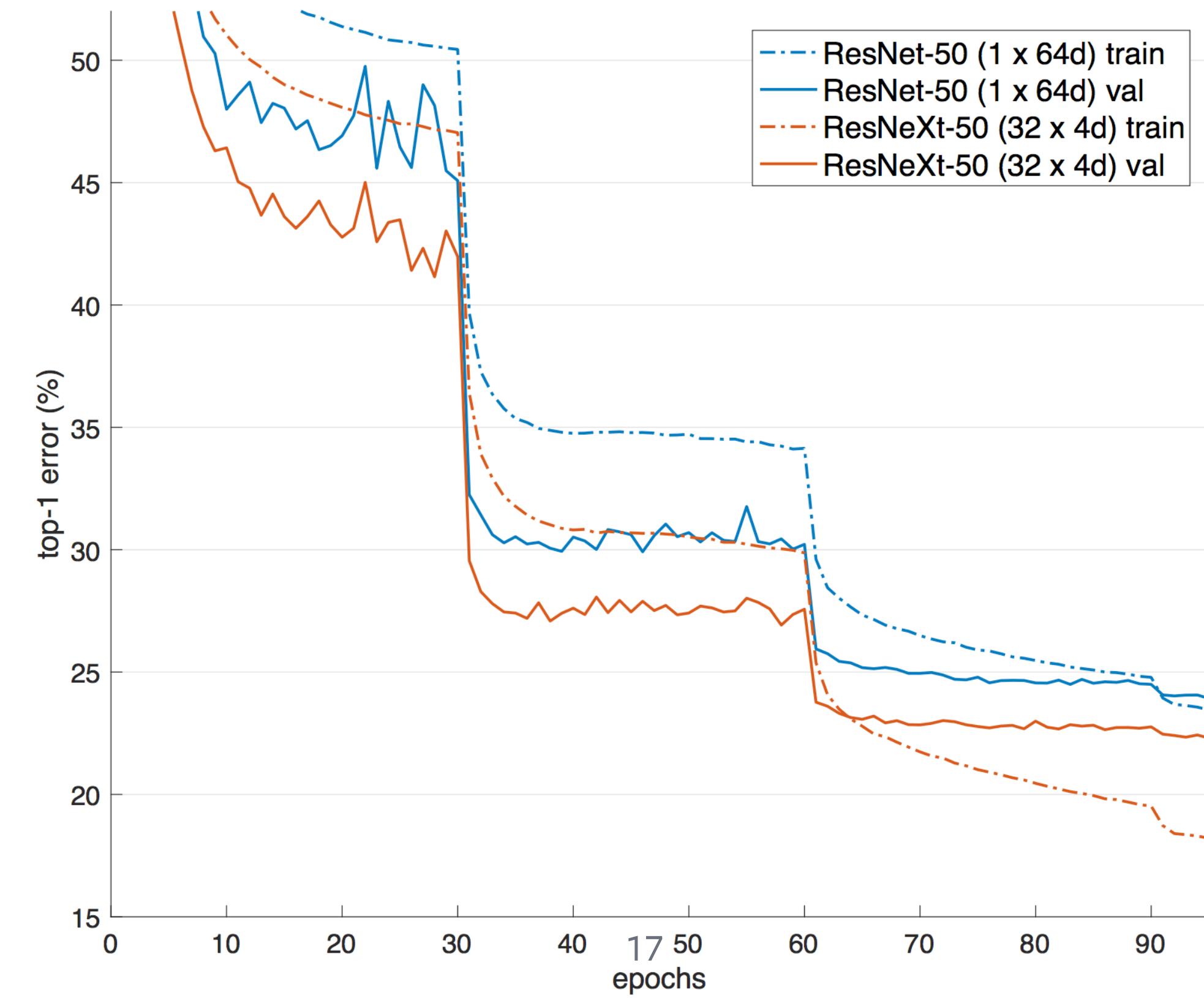
Group Convolutions

- In **group convolutions**, not all input channels feed into all output channels:



Group Convolutions

- **ResNeXt** is a popular model that uses group convolutions
- Group convolutions generally give a better **compute-accuracy trade-off**



* Figure credit: Kaiming He

Model compression

- Can we train a small model to **mimic** a large model?

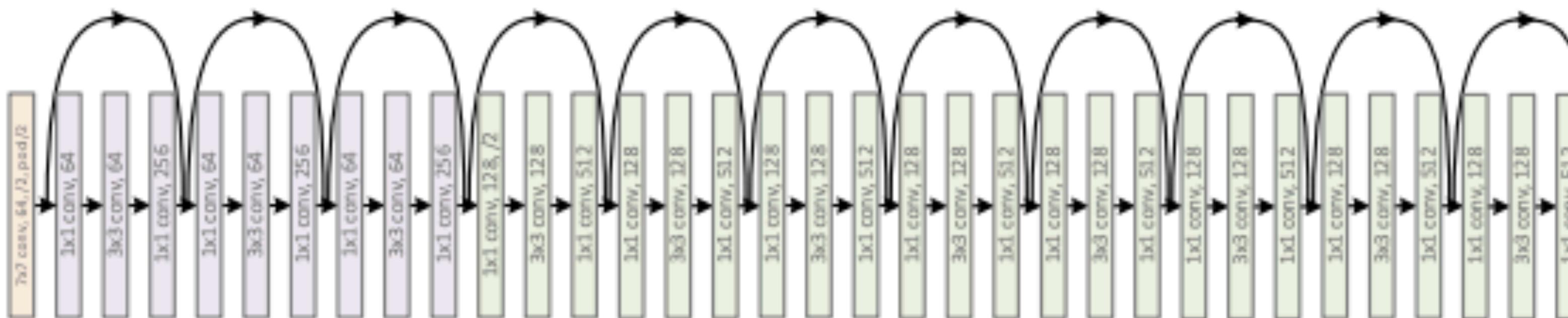
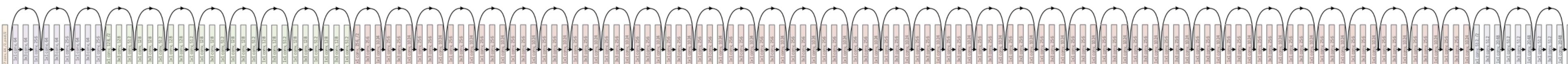
Model compression

- Can we train a small model to **mimic** a large model? Yes!

Model compression

- Can we train a small model to **mimic** a large model? Yes!

large “teacher” model



small “student” model

Model compression

- Can we train a small model to **mimic** a large model? Yes!

large “teacher” model



→ **teacher output**



→ **student output**

small “student” model

Model compression

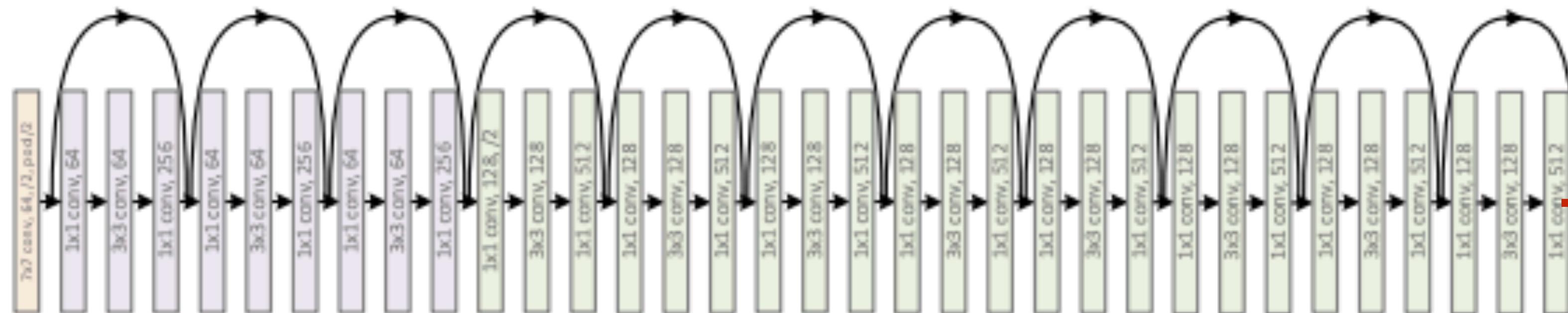
- Can we train a small model to **mimic** a large model? Yes!

large “teacher” model



→ **teacher output**

train student to minimize
cross-entropy with teacher



→ **student output**

small “student” model

Model compression

- This actually works very well in practice:

System	Test Frame Accuracy	WER
Small model (normal training)	58.9%	10.9%
Big model	61.1%	10.7%
Small model ("distillation")	60.8%	10.7%

Model compression

- This actually works very well in practice:

System	Test Frame Accuracy	WER
Small model (normal training)	58.9%	10.9%
Big model	61.1%	10.7%
Small model ("distillation")	60.8%	10.7%

- Why do you think this works?

Model compression

- This actually works very well in practice:

System	Test Frame Accuracy	WER
Small model (normal training)	58.9%	10.9%
Big model	61.1%	10.7%
Small model ("distillation")	60.8%	10.7%

- Why do you think this works?
- **Soft targets!** Teacher tells student which examples to "ignore"...

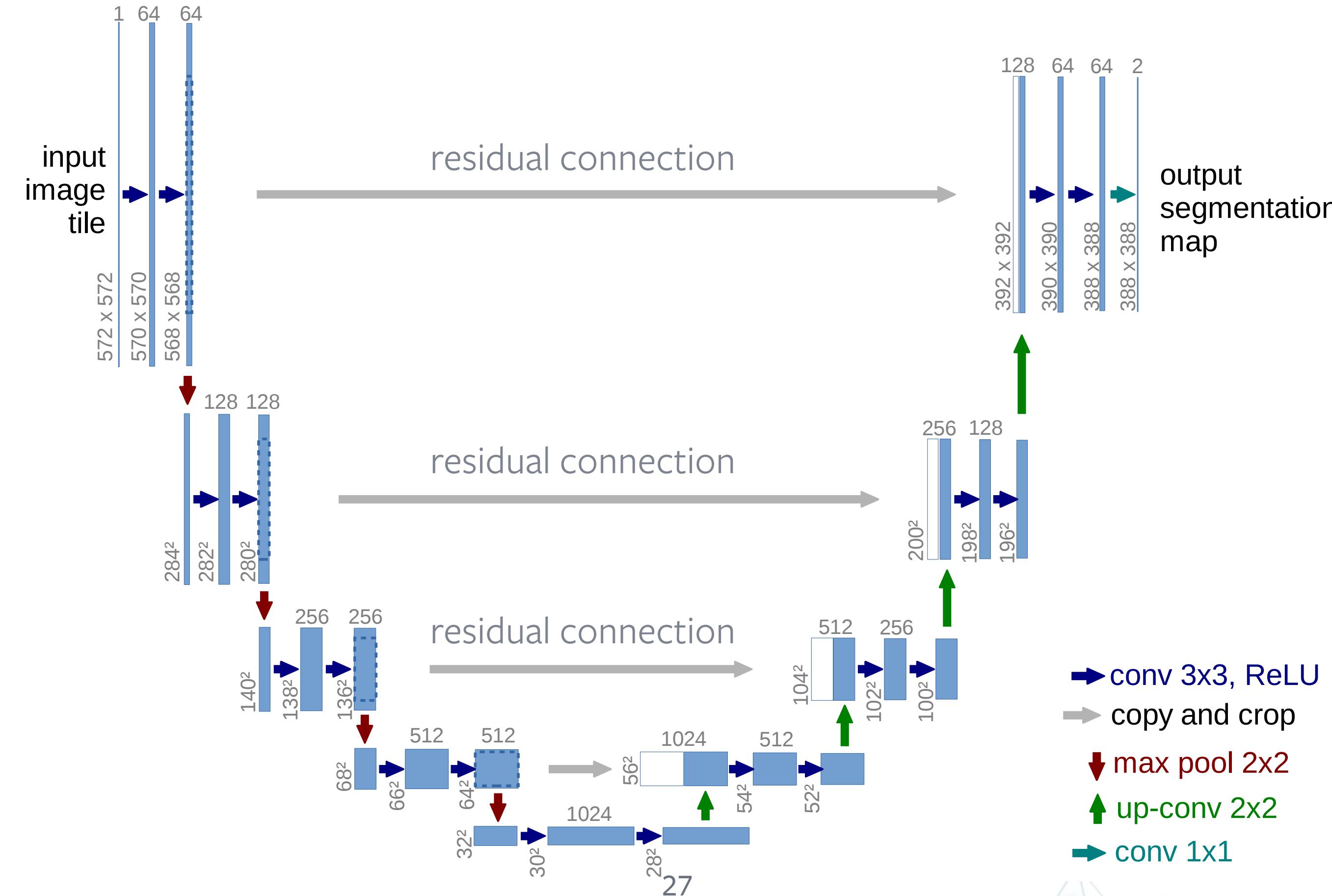
Beyond image classification

- **Semantic segmentation:** Predict label for each pixel in the image



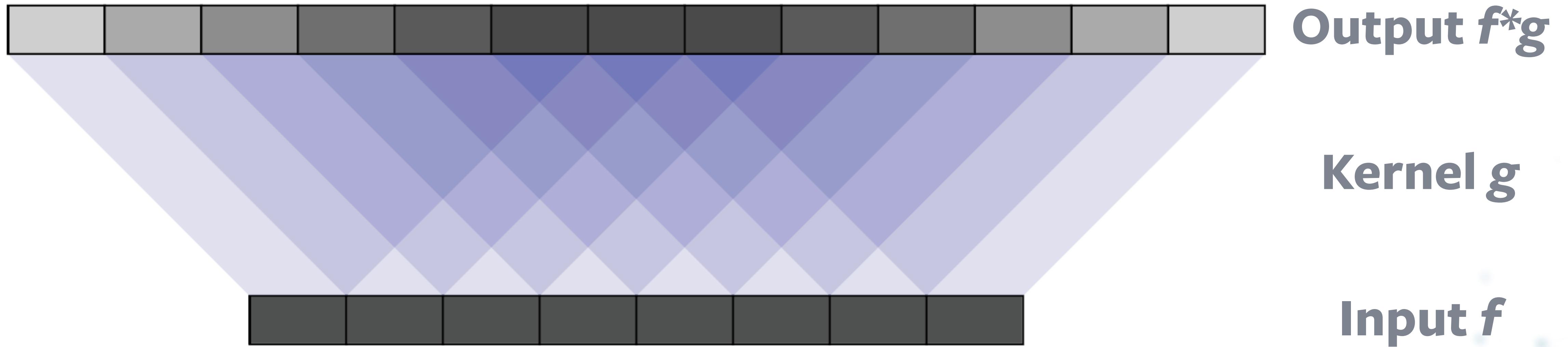
Beyond image classification

- **U-Nets** are an architecture designed for such problems:

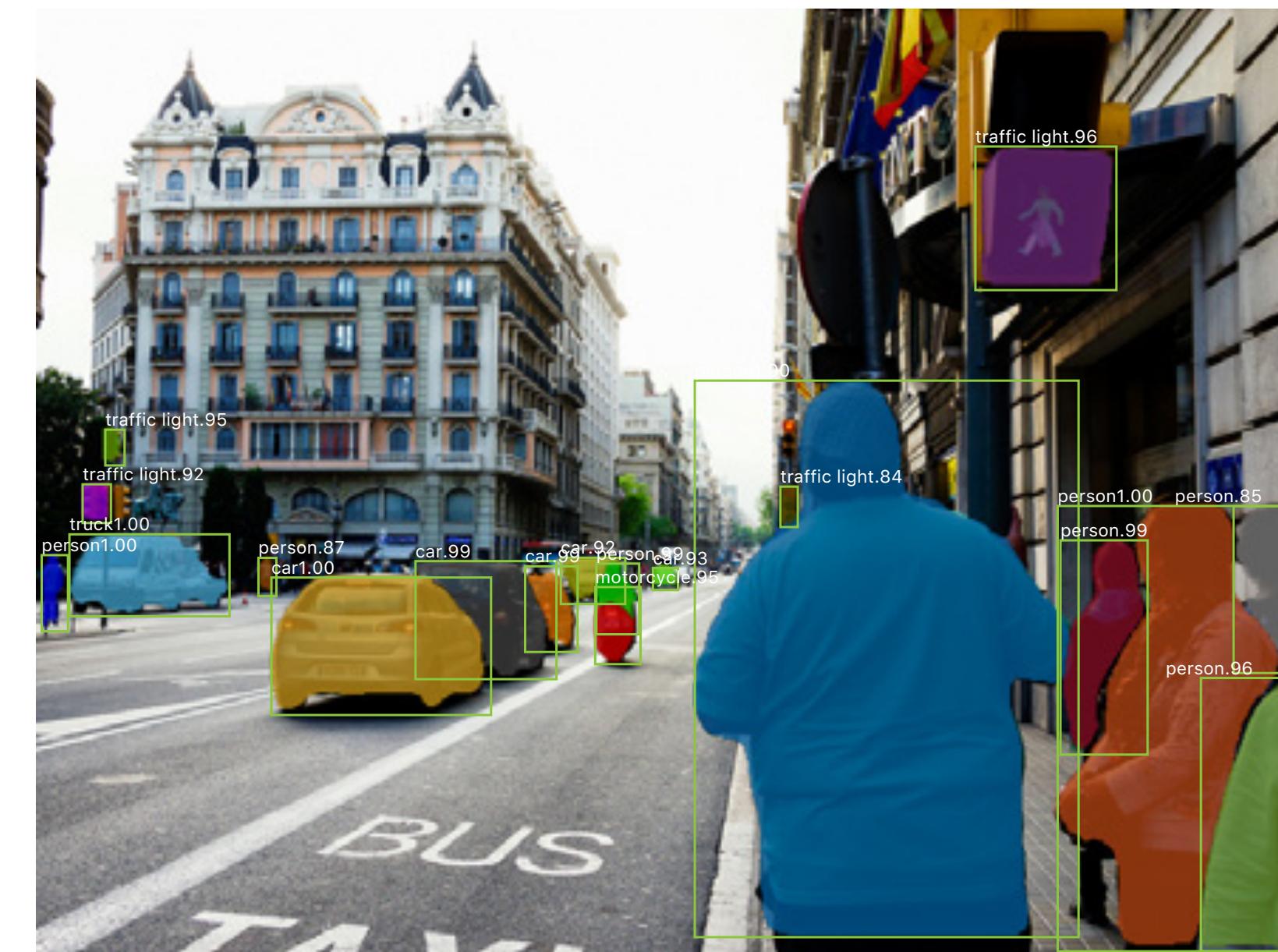
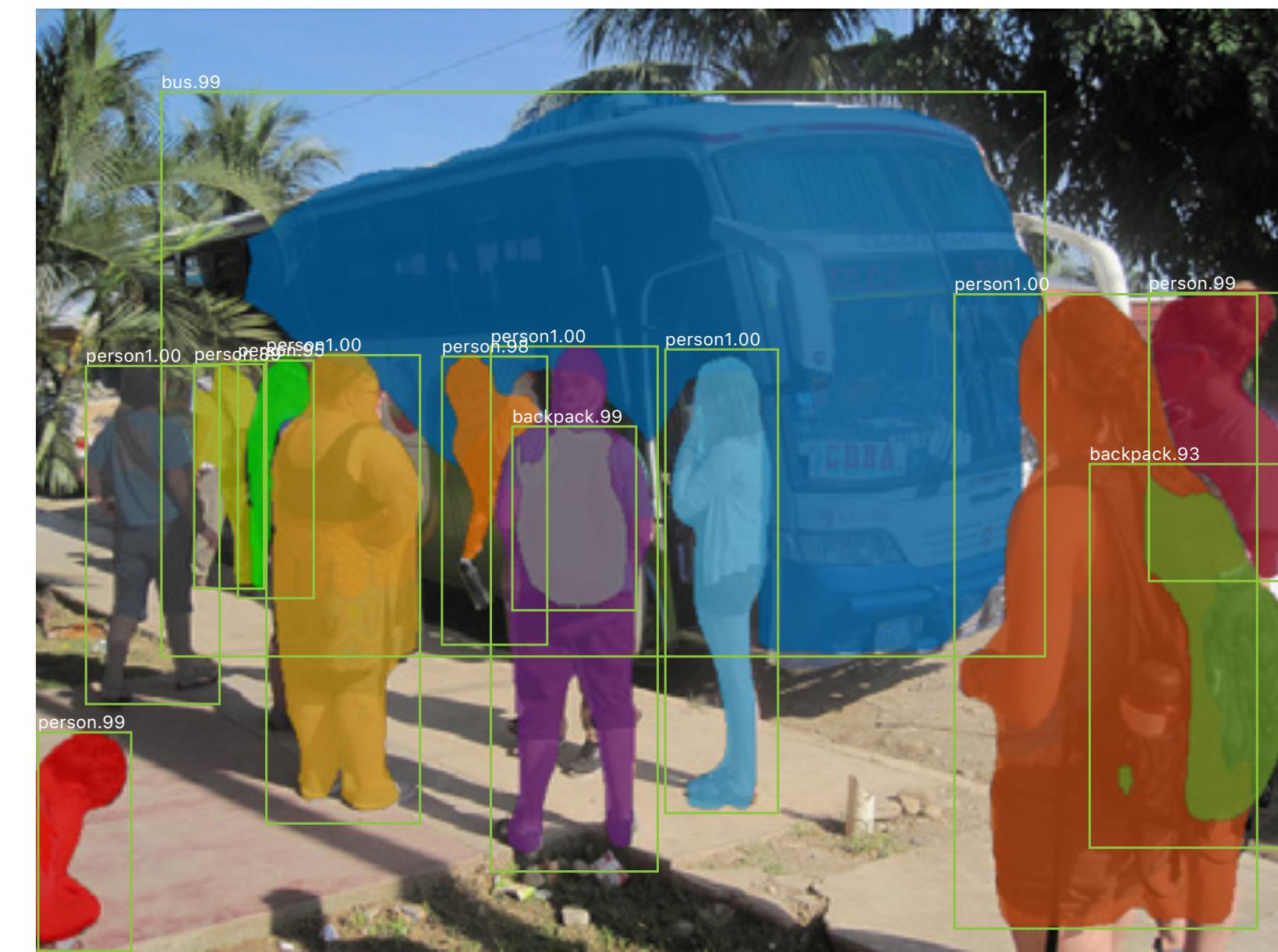
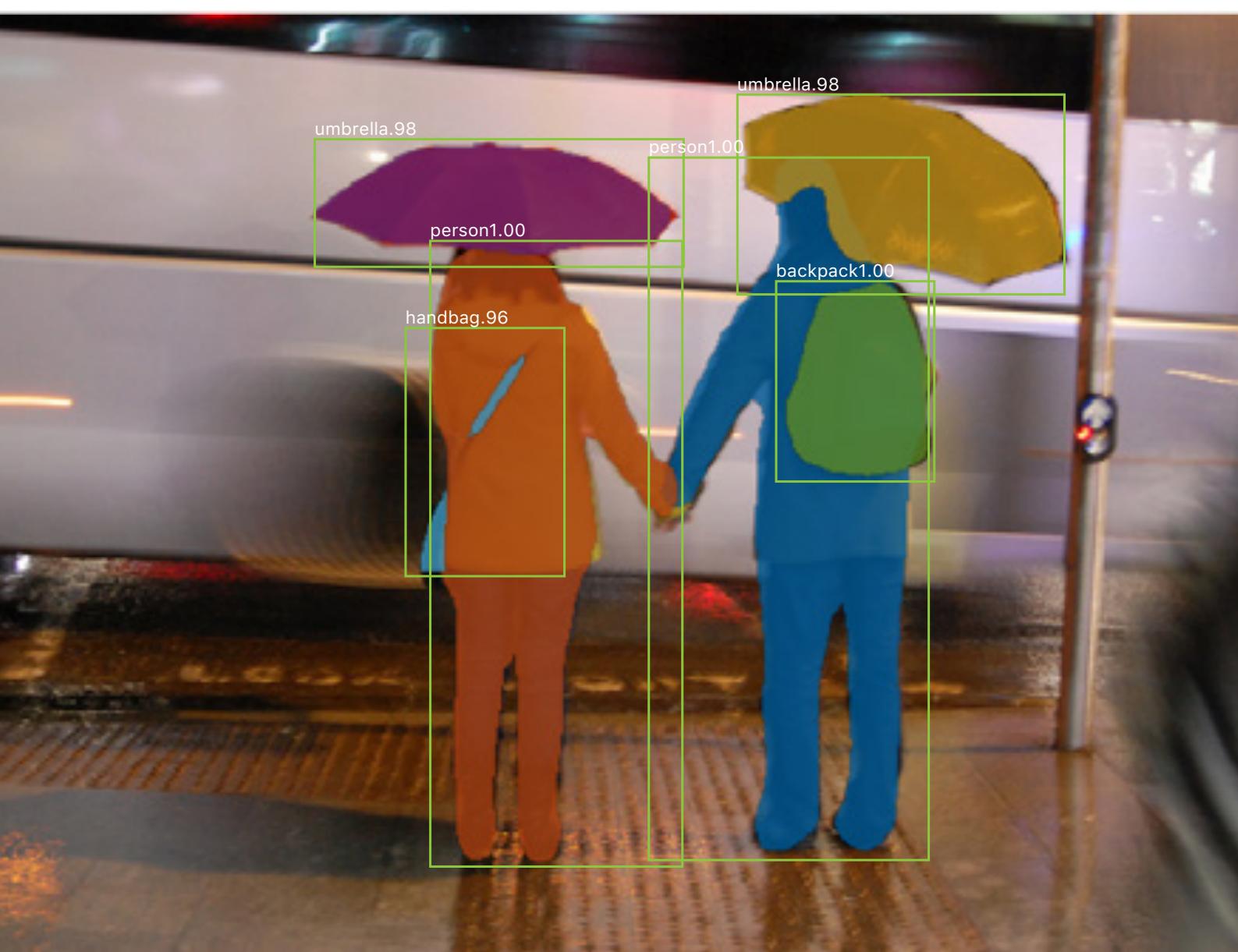
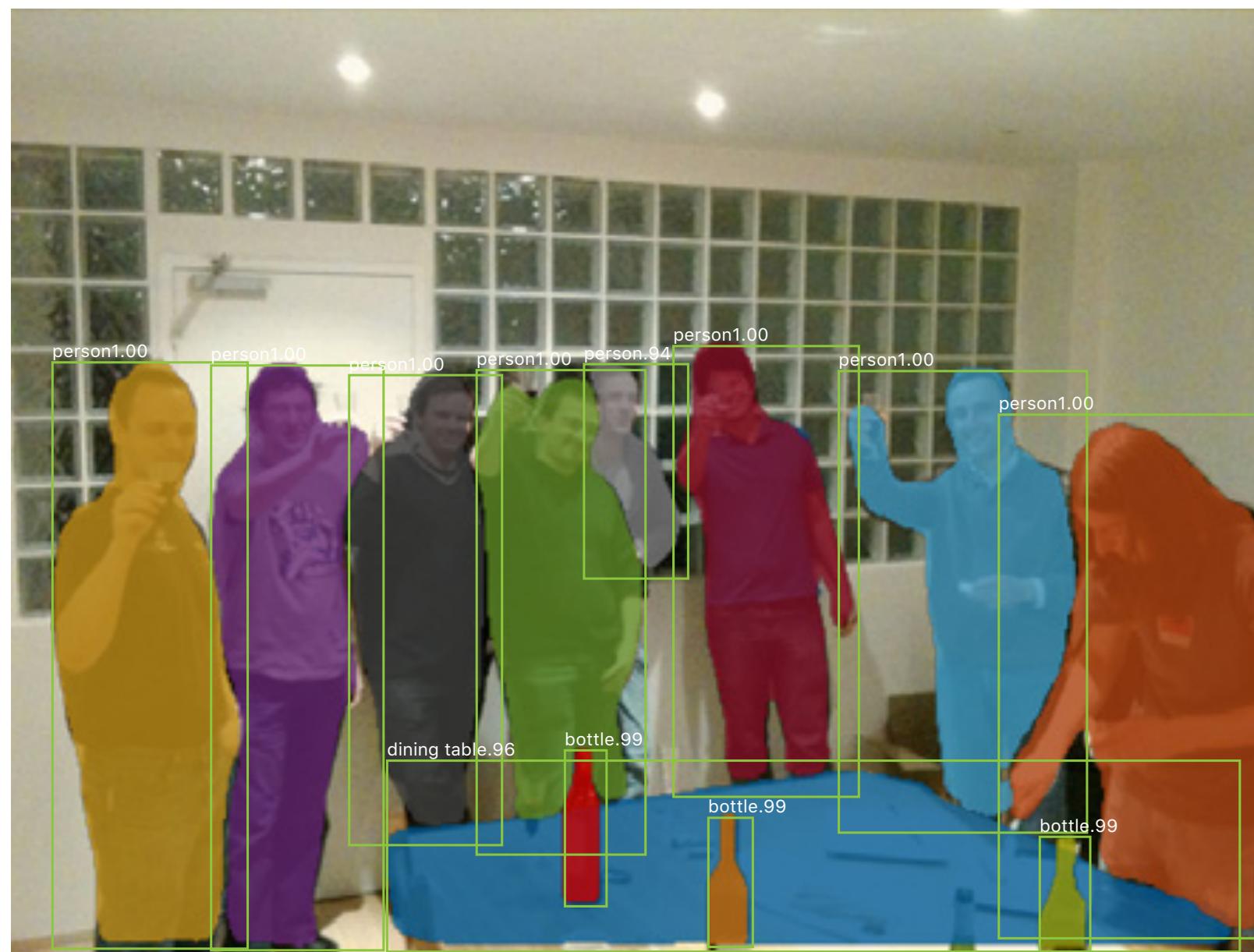


Beyond image classification

- Main additional ingredient in U-Nets is **deconvolution**:



* Credits: Chris Olah



* Results obtained with Mask R-CNN.

Summary

- Residual connections help loss propagation through network, allowing for successfully training deeper networks
- Group convolutions reduce parameters and computation by reducing number of interactions between input and output channels
- Deconvolution allows for building segmentation networks

Reading material

- S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. **Aggregated Residual Transformations for Deep Neural Networks**. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017
- O. Ronneberger, P. Fischer, and T. Brox. **U-Net: Convolutional Networks for Biomedical Image Segmentation**. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

Questions?