Best paper award

# DENSELY CONNECTED CONVOLUTIONAL NETWORKS

Gao Huang*, Zhuang Liu*, Laurens van der Maaten, Kilian Q. Weinberger

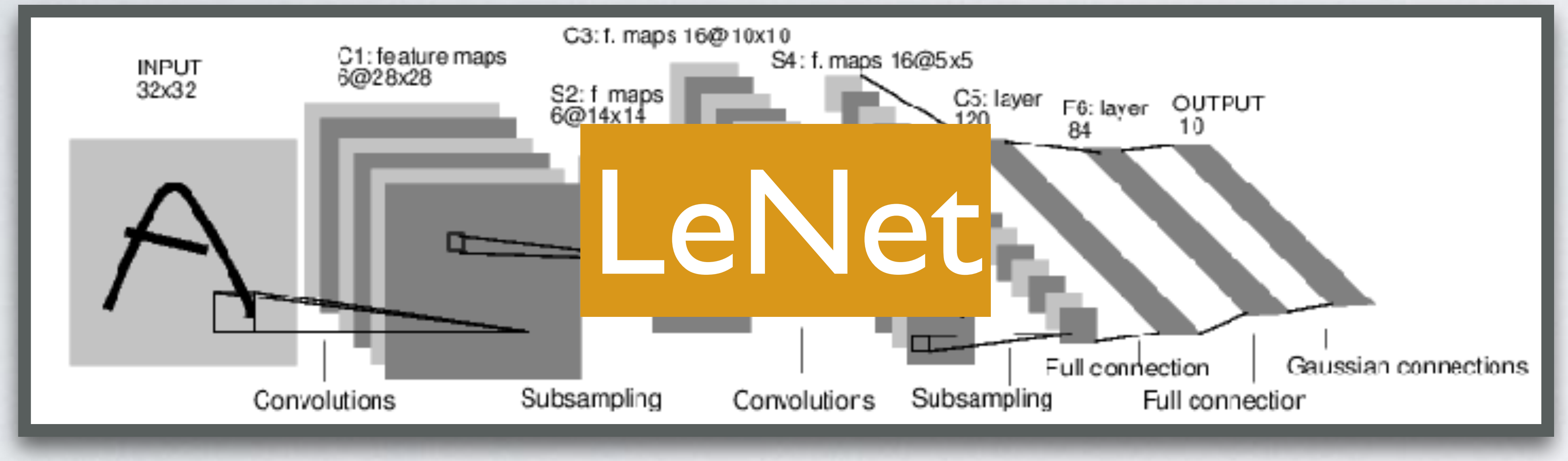
Cornell University

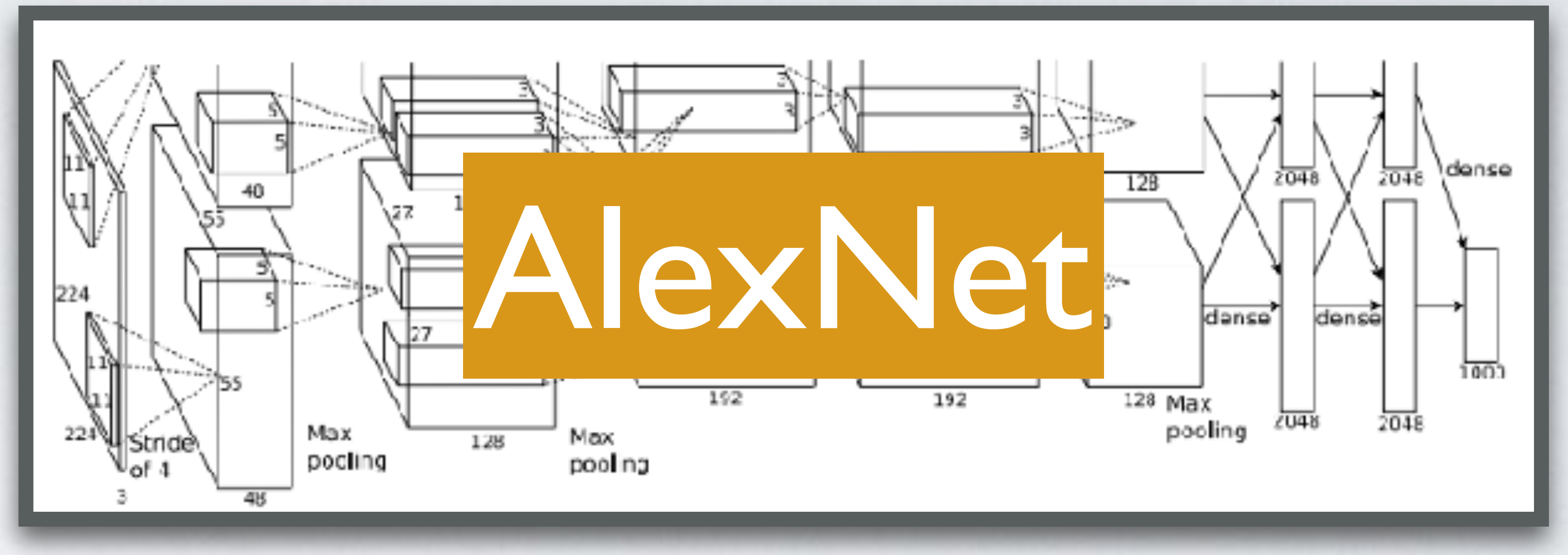
Tsinghua University

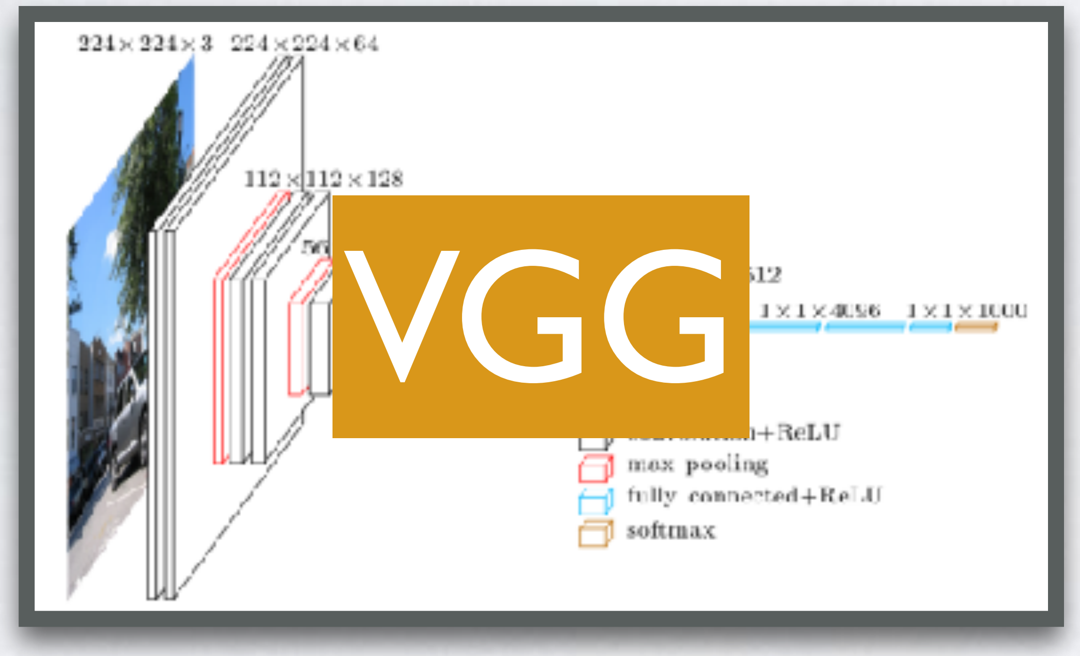
Facebook AI Research

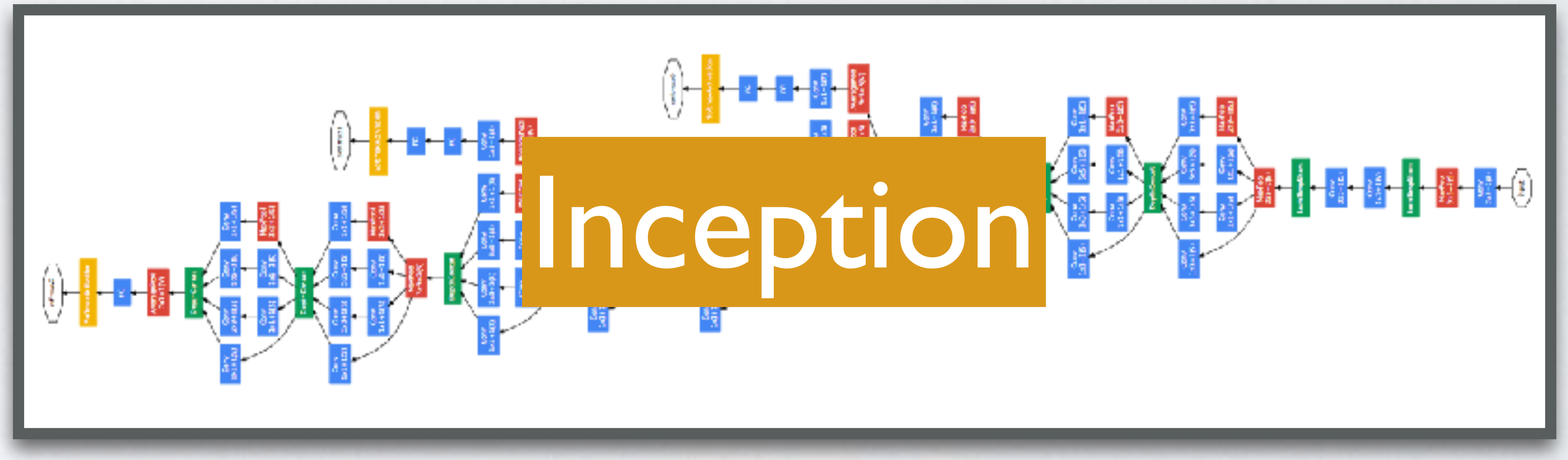
CVPR 2017

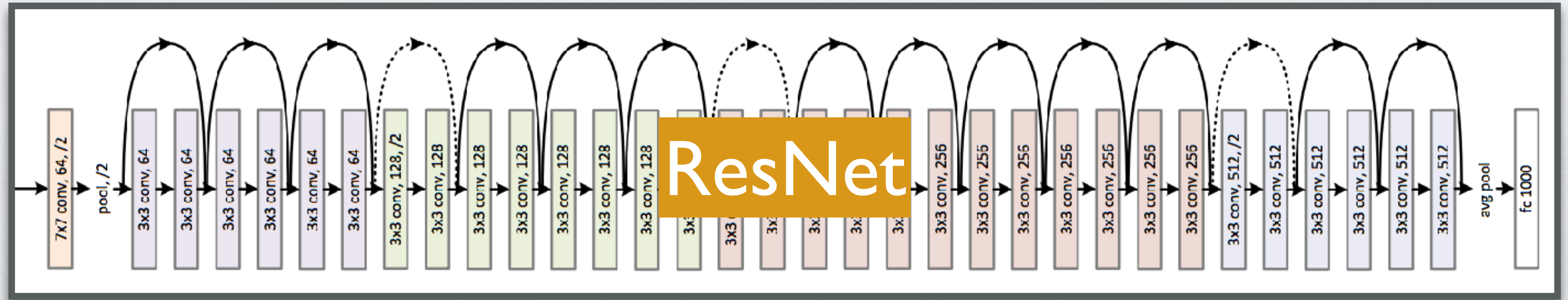
# CONVOLUTIONAL NETWORKS



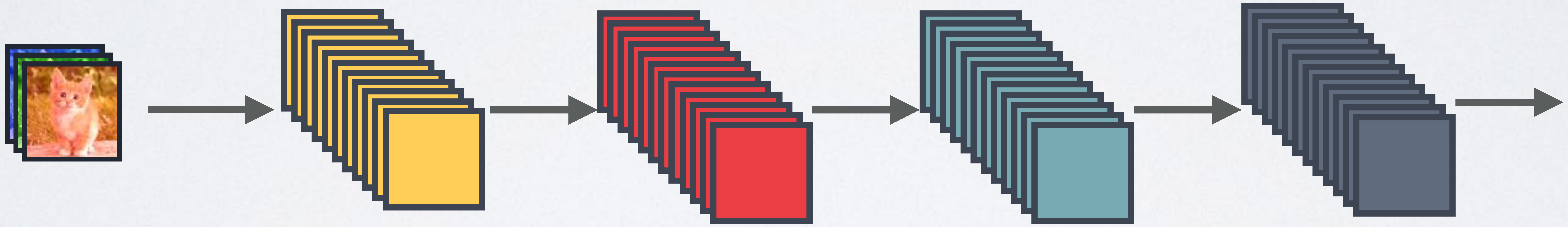



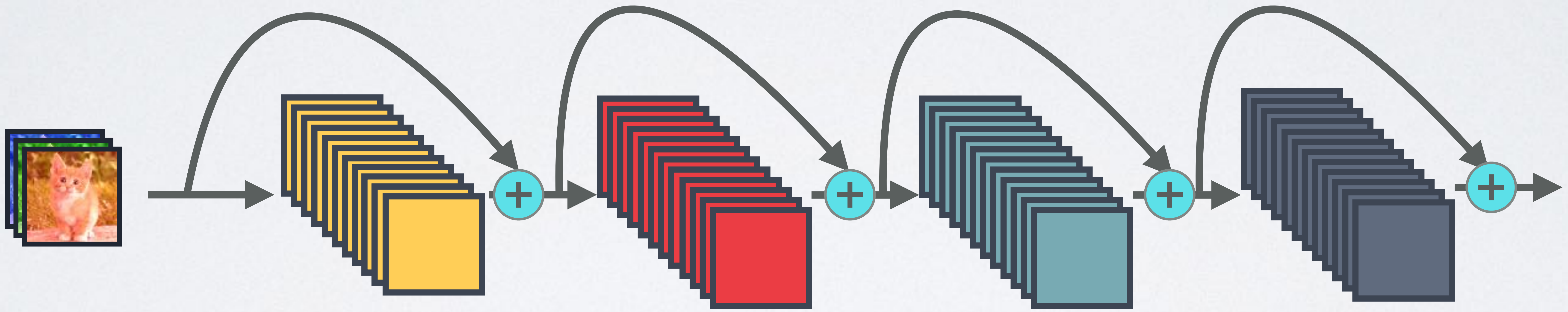



# STANDARD CONNECTIVITY



# RESNET CONNECTIVITY

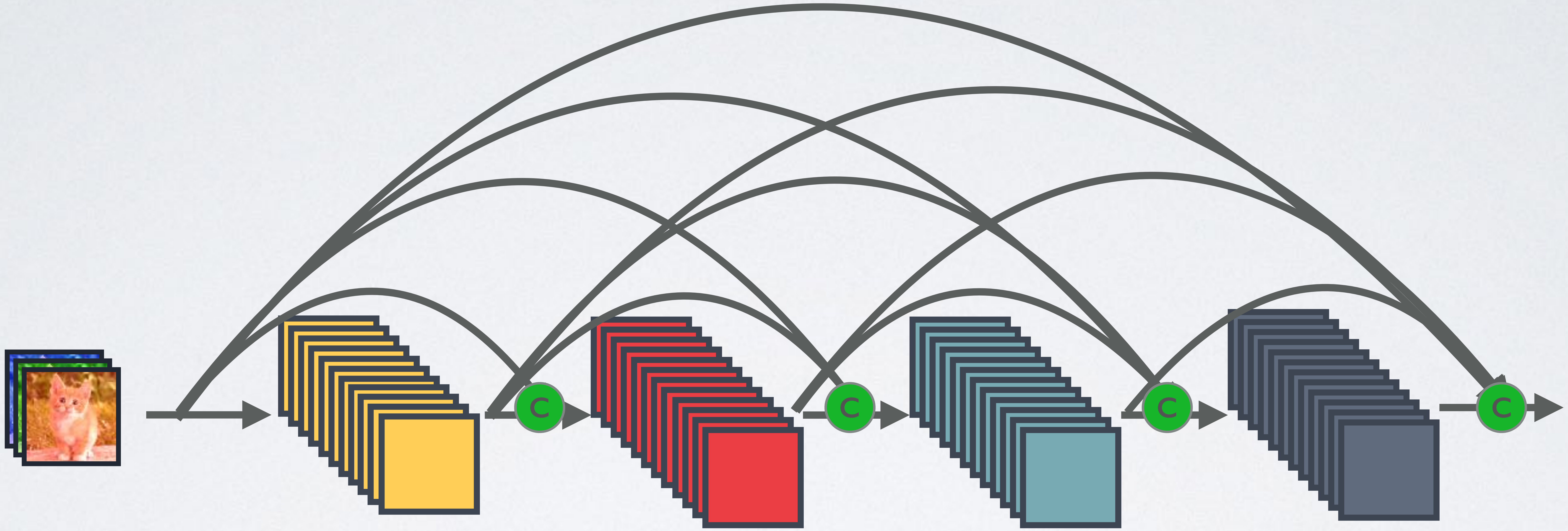
Identity mappings promote gradient propagation.



⊕ : Element-wise addition

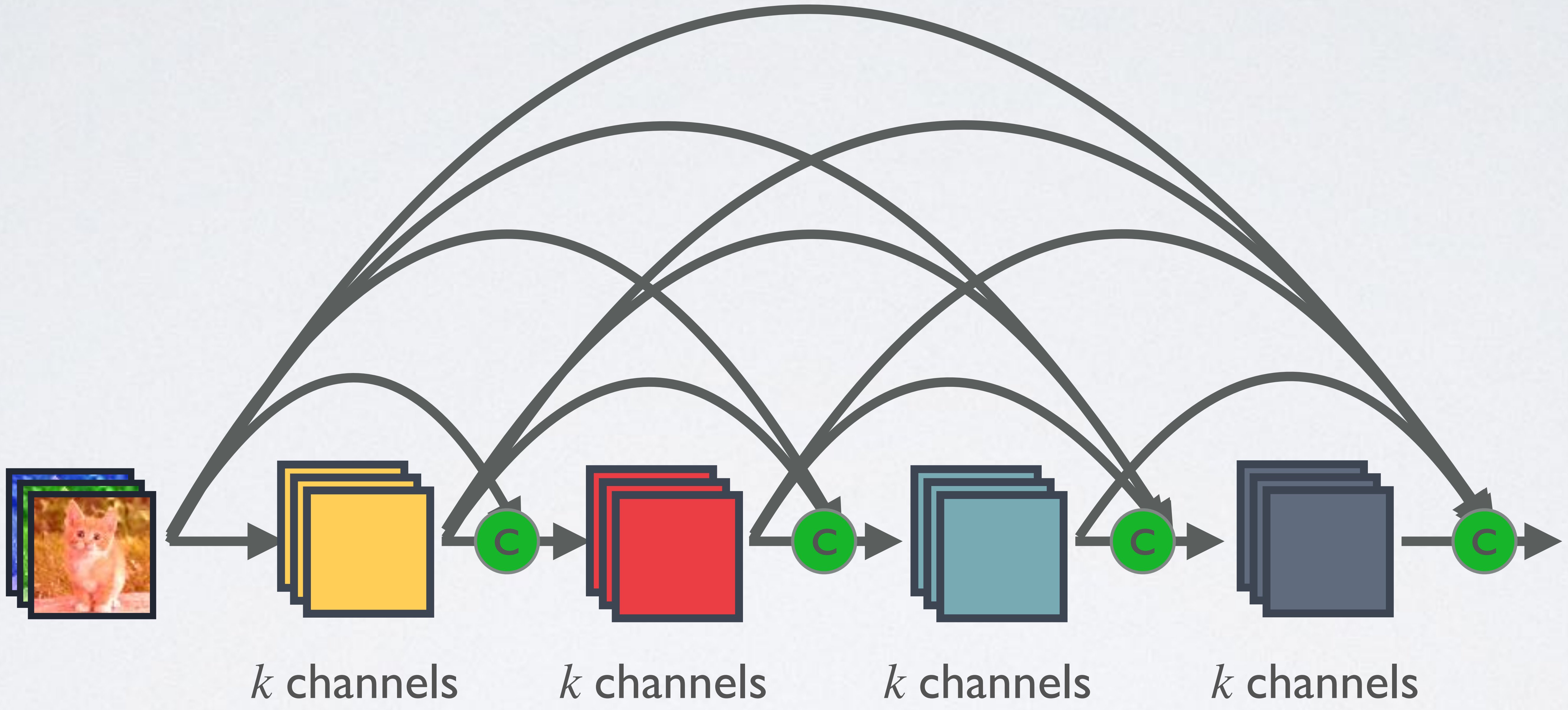
*Deep residual learning for image recognition: [He, Zhang, Ren, Sun] (CVPR 2015)*

# DENSE CONNECTIVITY




C : Channel-wise concatenation

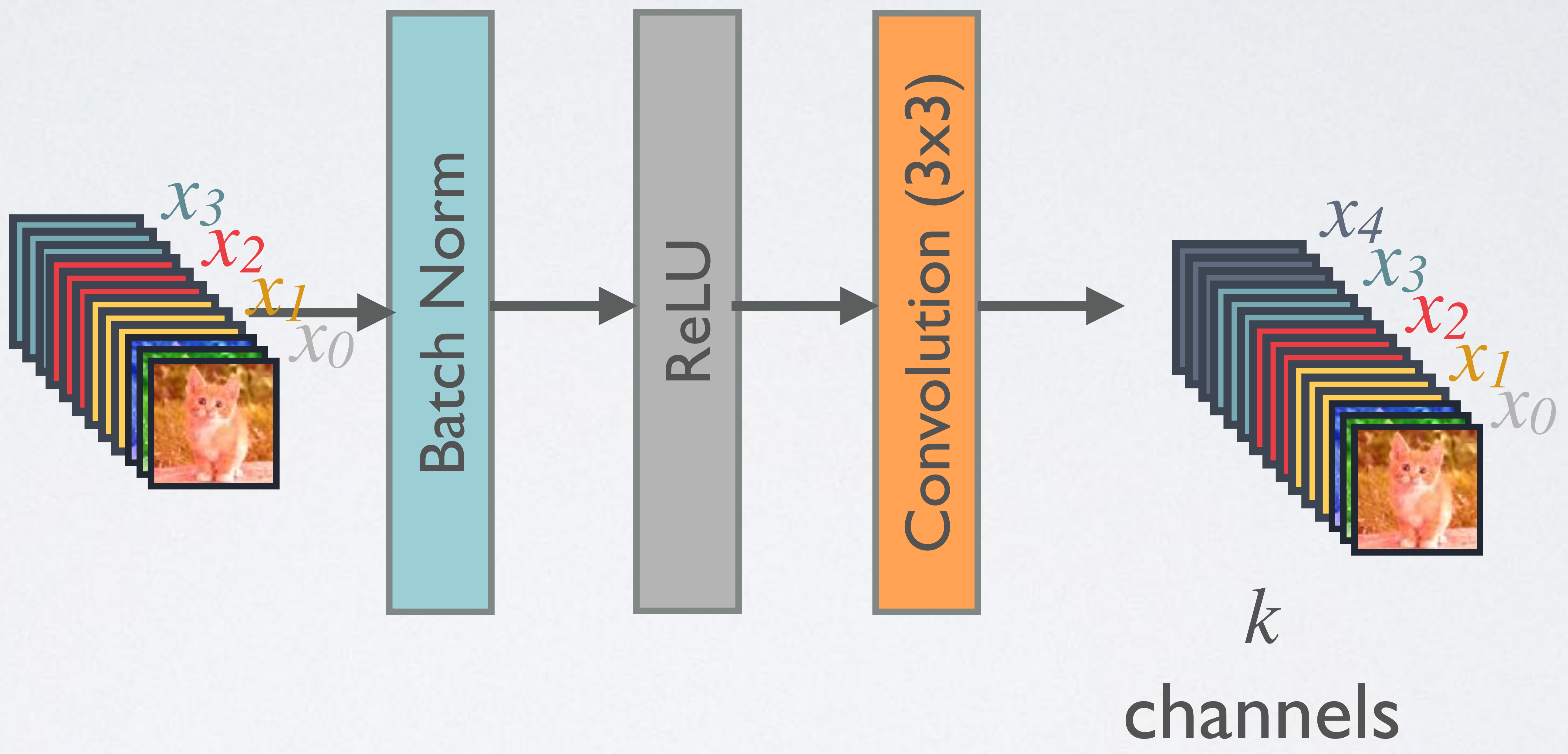
# DENSE AND SLIM




$k$ : Growth Rate

# FORWARD PROPAGATION

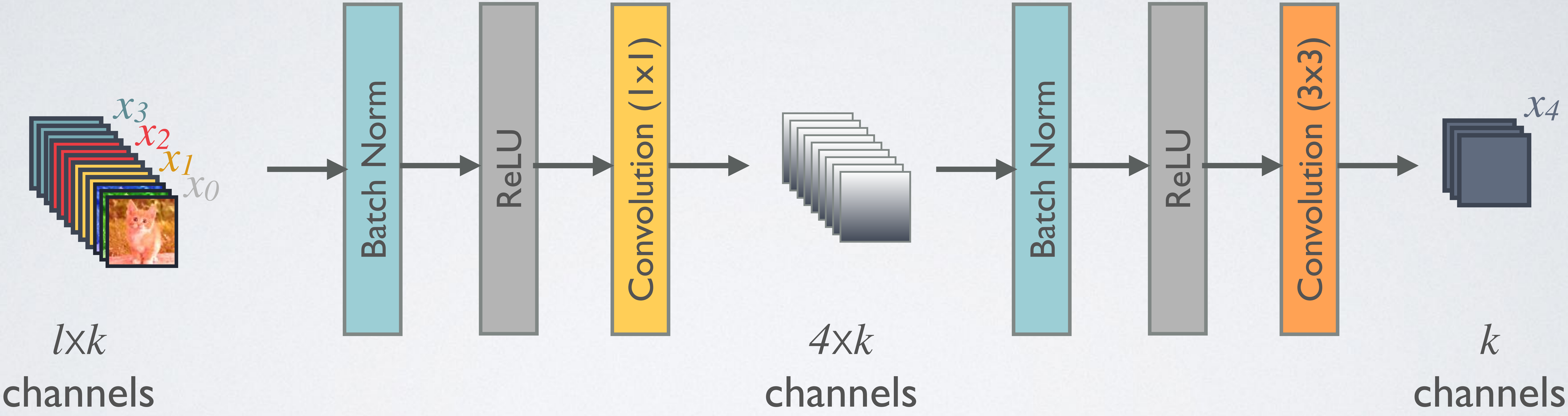
# COMPOSITE LAYER IN DENSENET




$$x_5 = h_5([x_0, \ldots, x_4])$$

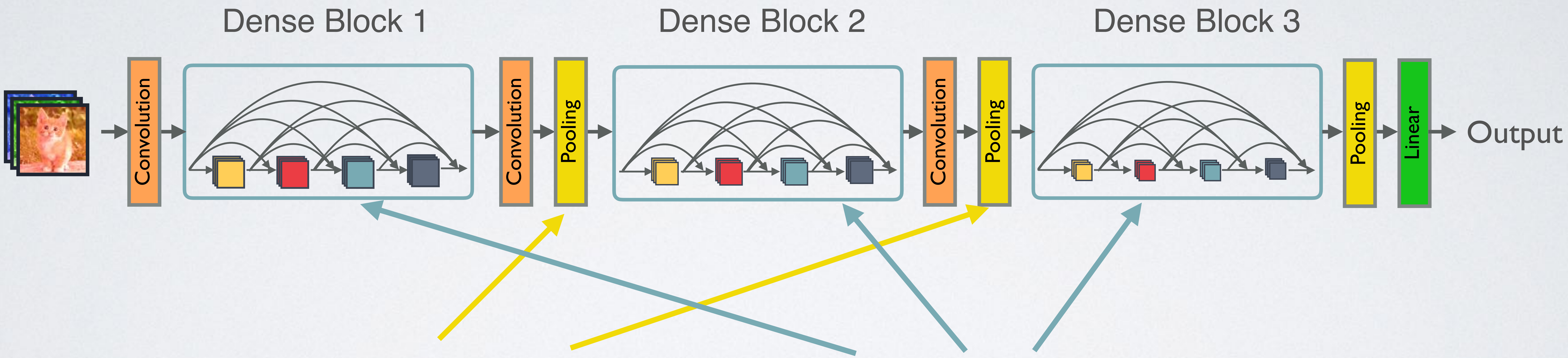
# COMPOSITE LAYER IN DENSENET

## WITH BOTTLENECK LAYER




Higher parameter and computational efficiency

# DENSENET

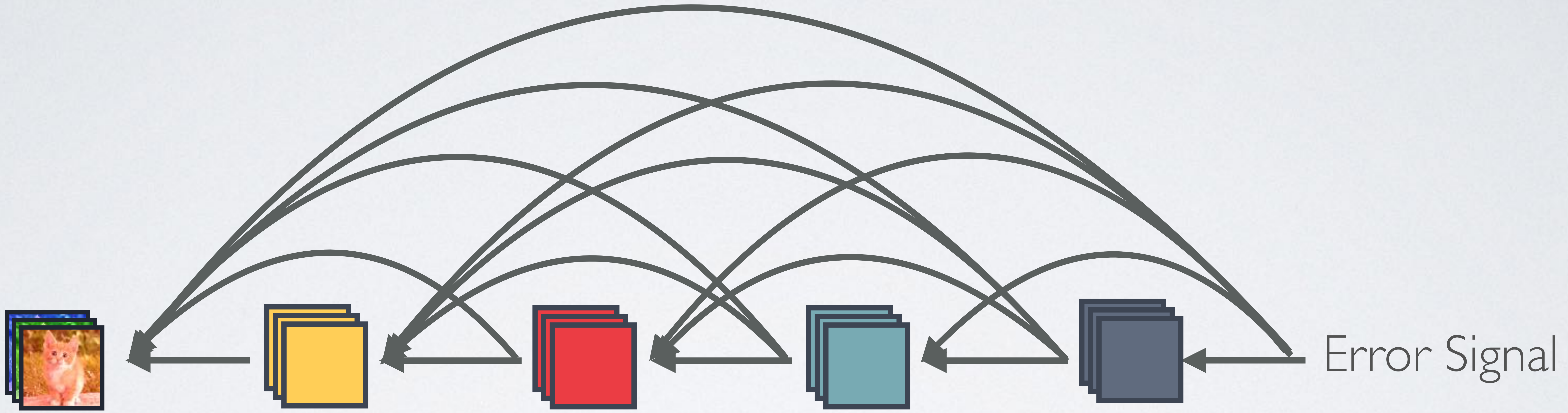



Pooling reduces feature map sizes

Feature map sizes match within each block

# ADVANTAGES OF DENSE CONNECTIVITY

# ADVANTAGE 1: STRONG GRADIENT FLOW

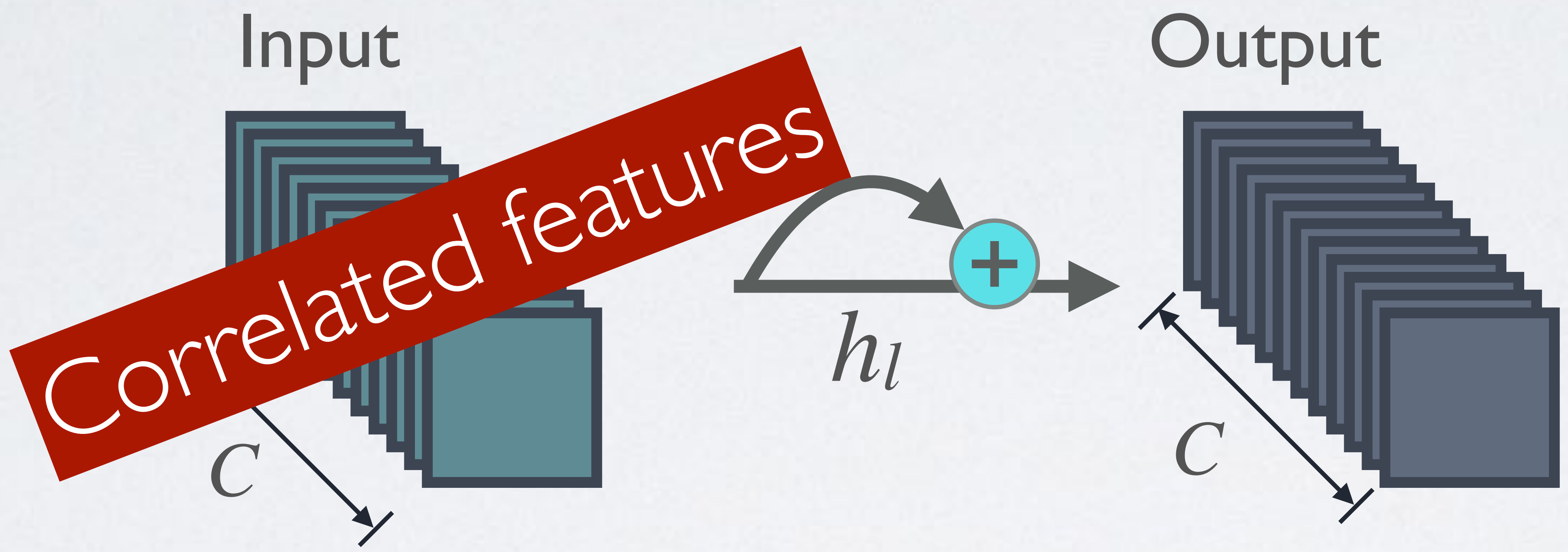



Implicit "deep supervision"

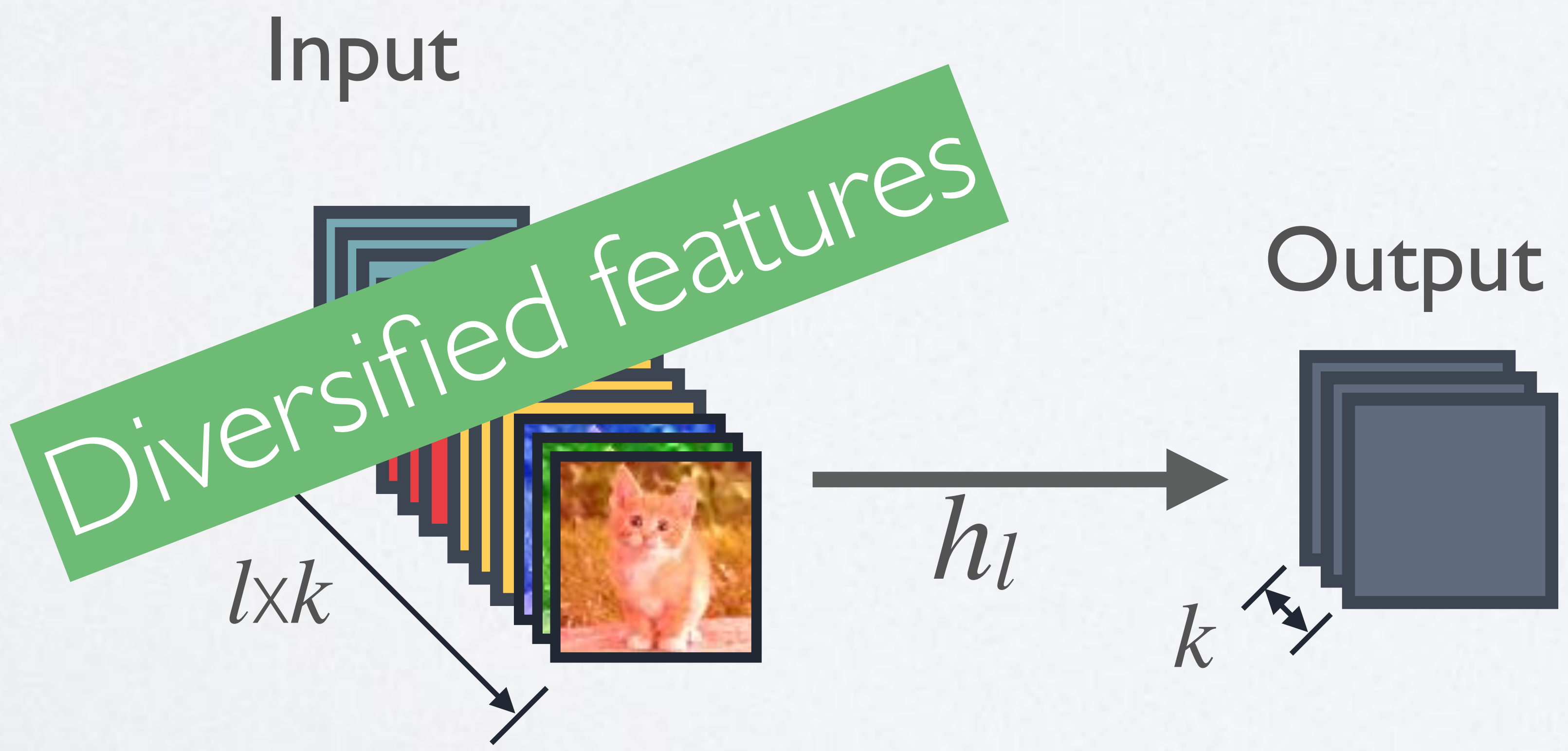
***Deeply supervised Net: [Lee, Xie, Gallagher, Zhang, Tu] (2015)***

# ADVANTAGE 2: PARAMETER & COMPUTATIONAL EFFICIENCY

**ResNet connectivity**:




**DenseNet connectivity:**




#parameters:

$\boxed{O(C \times C)}$

$k \ll C$

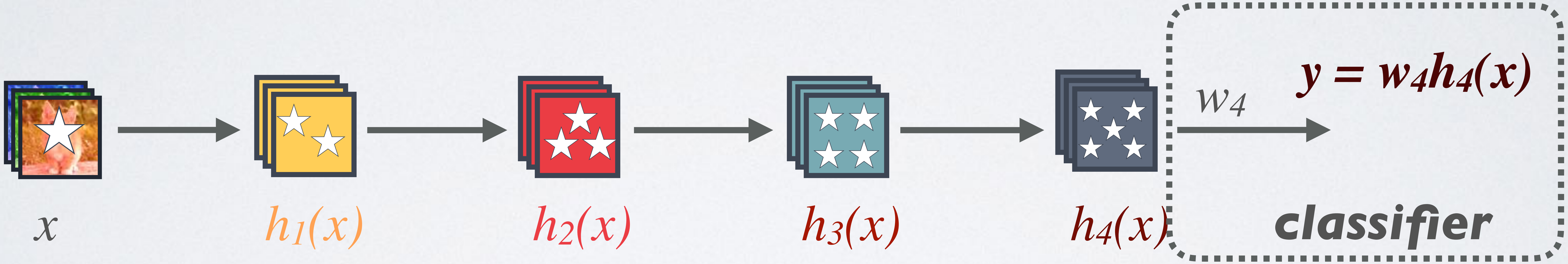
$\boxed{O(l \times k \times k)}$

$k$: *Growth rate*

# ADVANTAGE 3: MAINTAINS LOW COMPLEXITY FEATURES

**Standard Connectivity:**

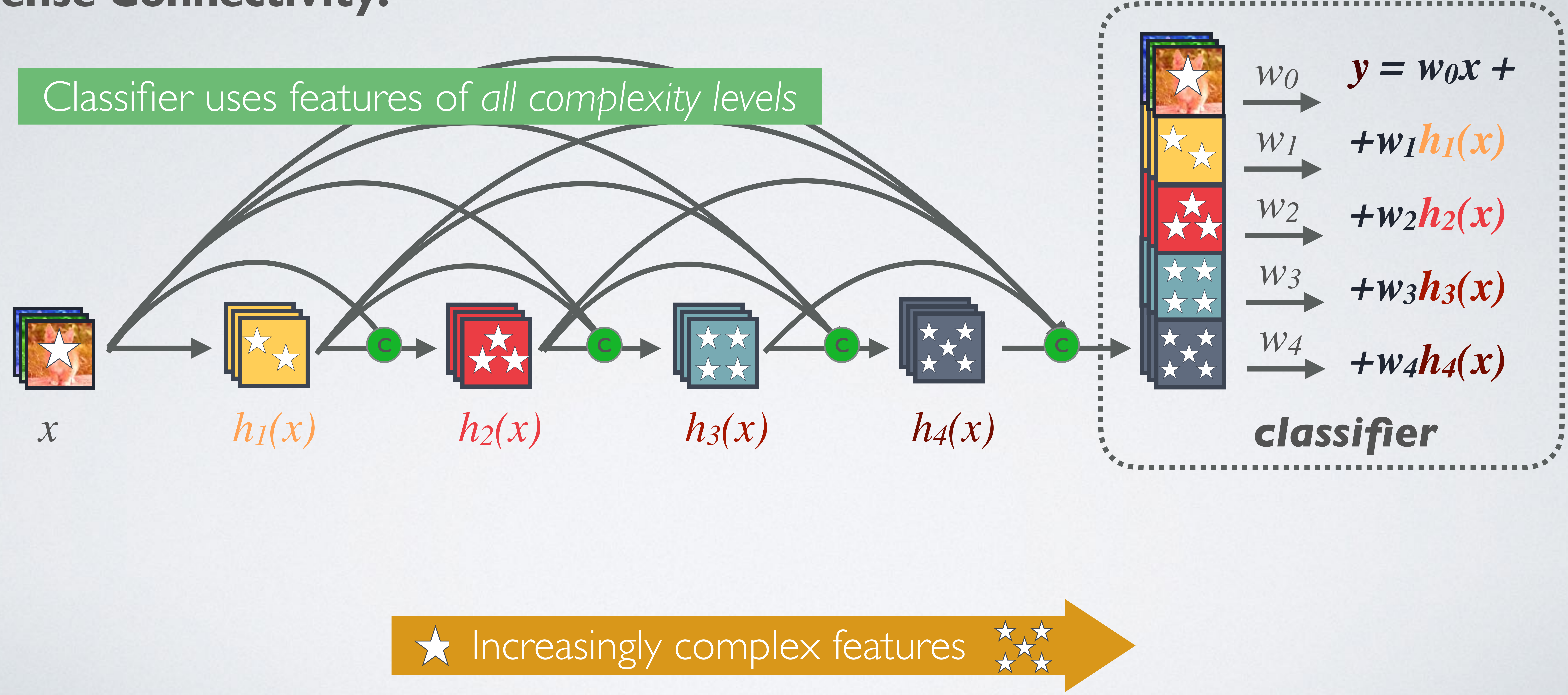
Classifier uses most complex (high level) features




Increasingly complex features

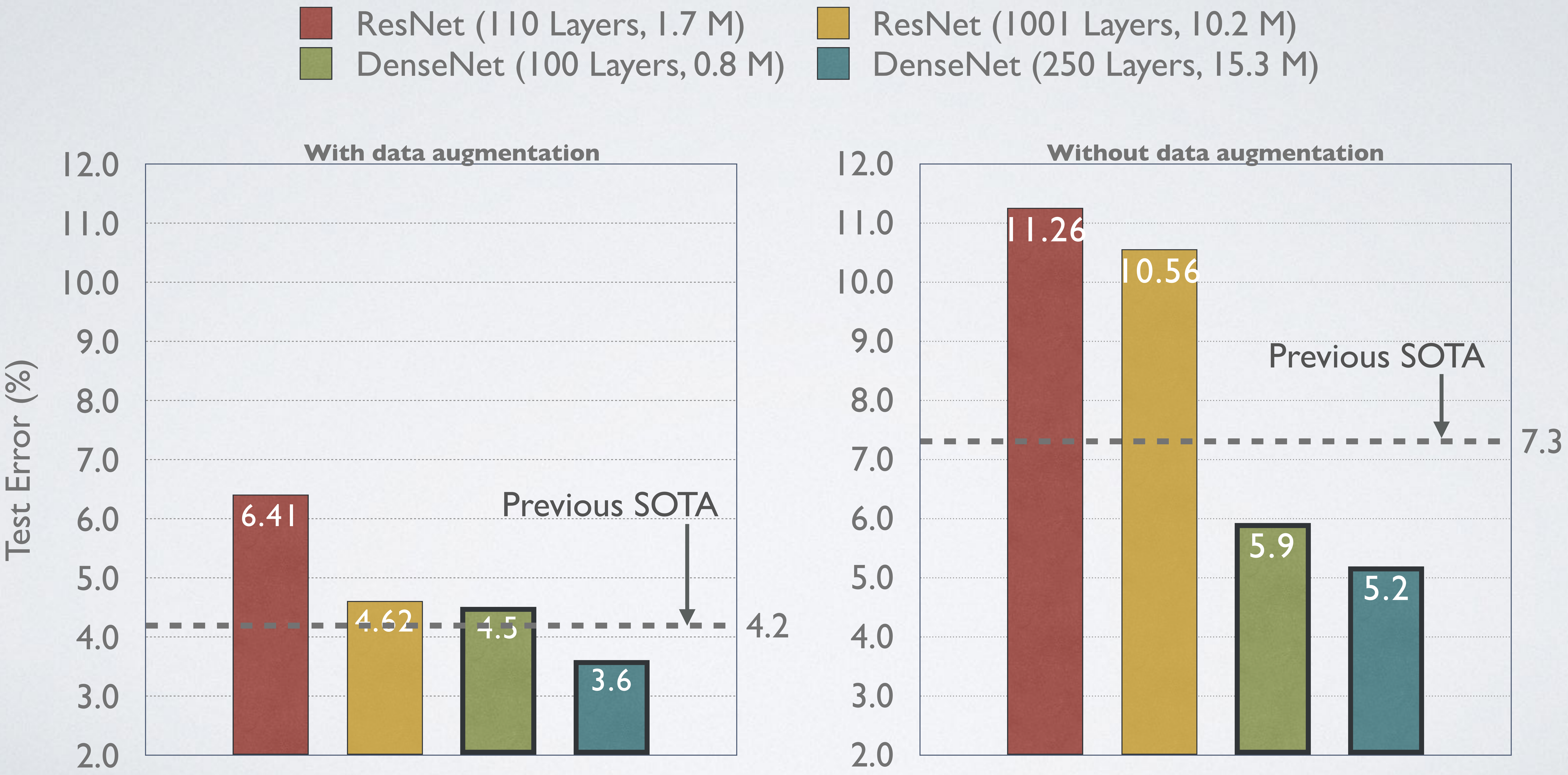
# ADVANTAGE 3: MAINTAINS LOW COMPLEXITY FEATURES

**Dense Connectivity:**

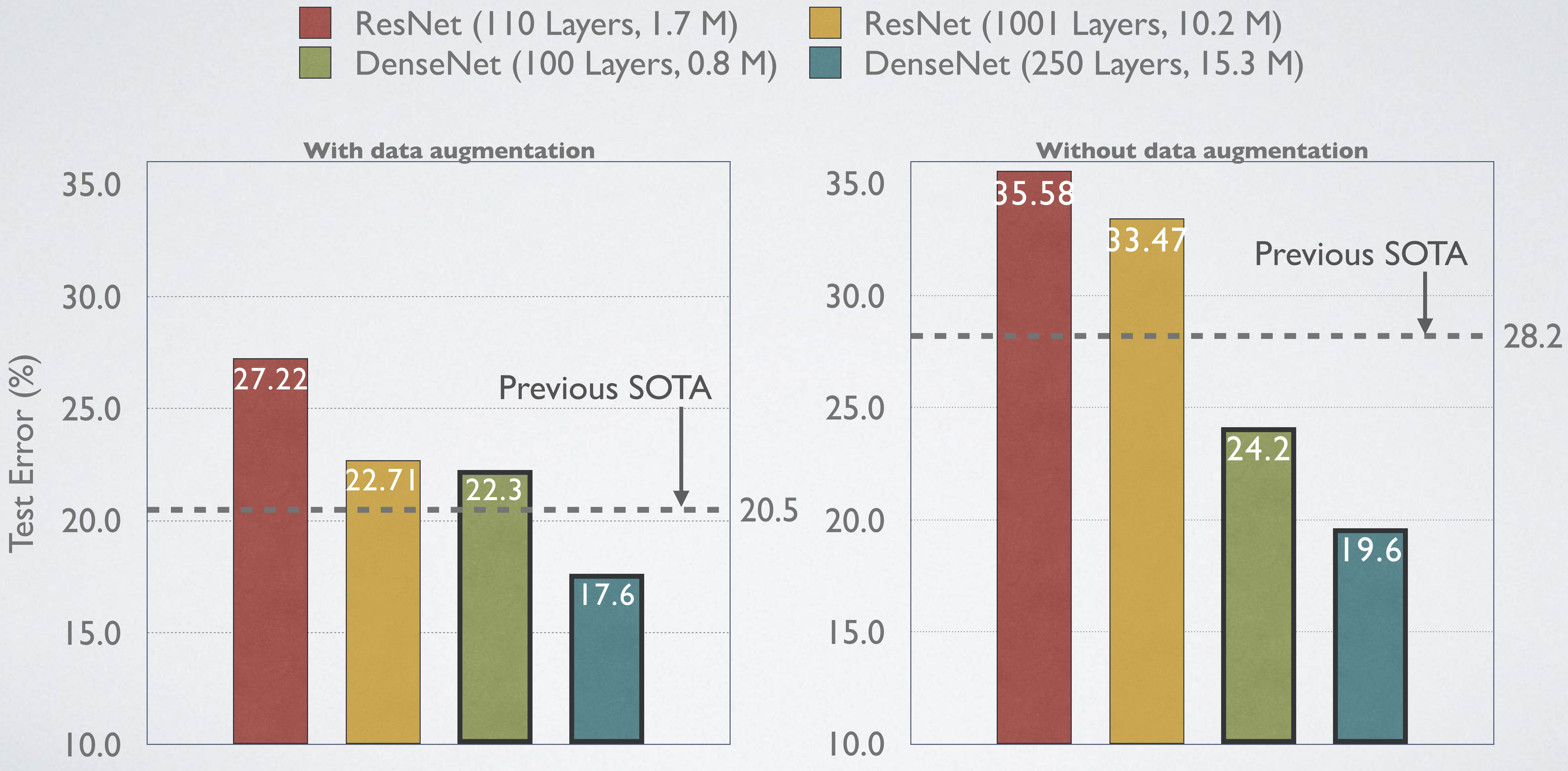


# RESULTS

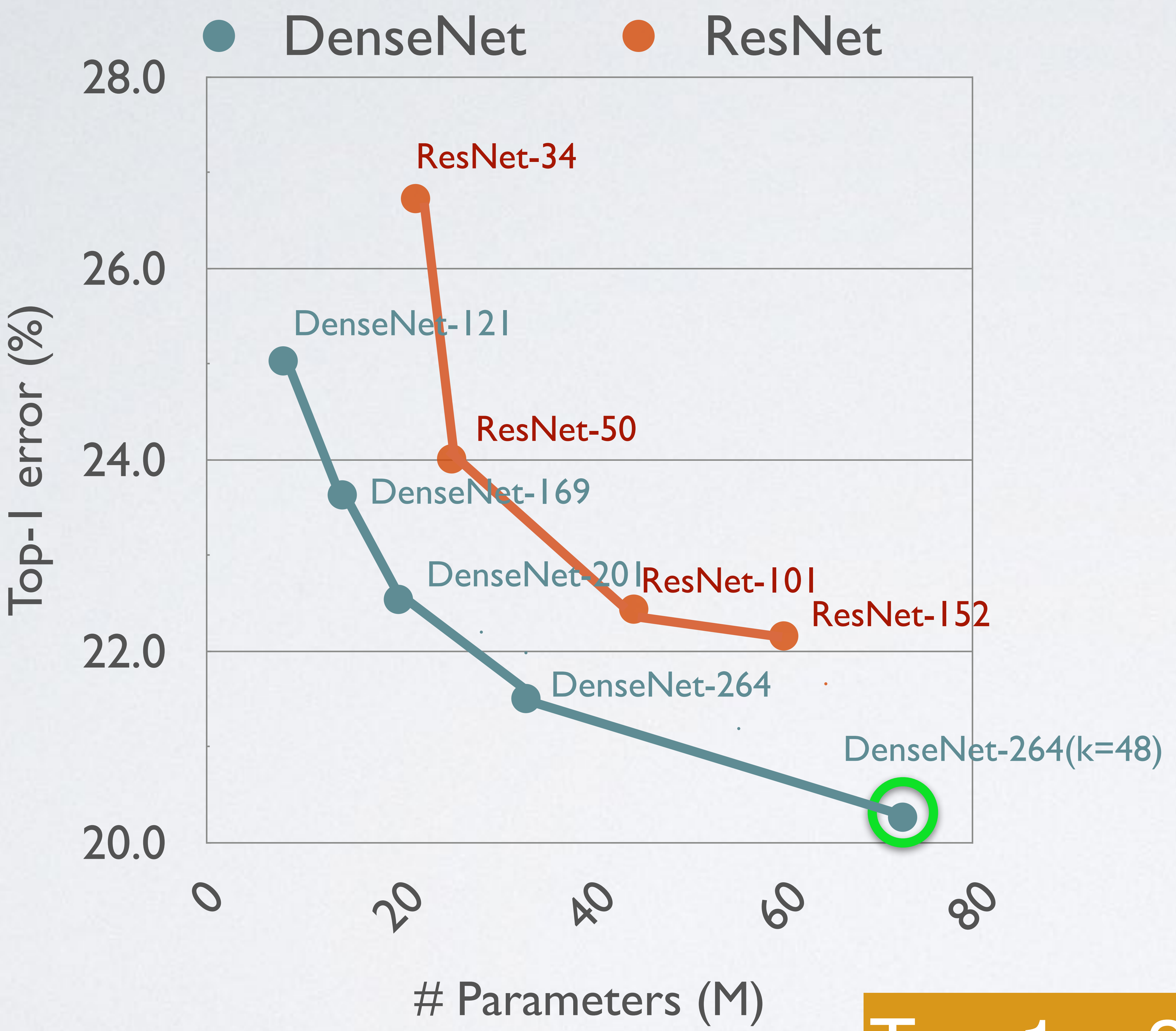
# RESULTS ON **CIFAR-10**

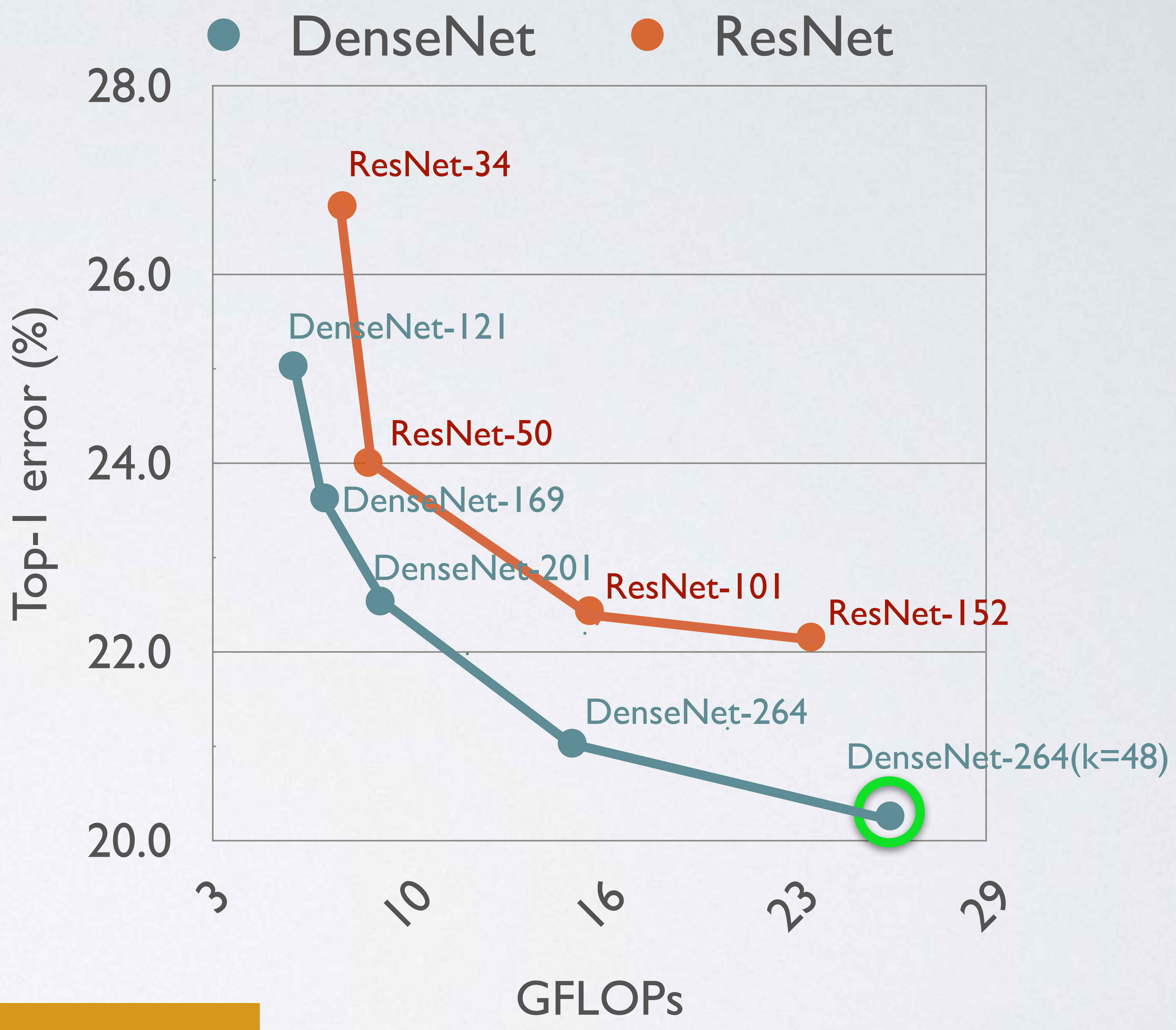


# RESULTS ON **CIFAR-100**



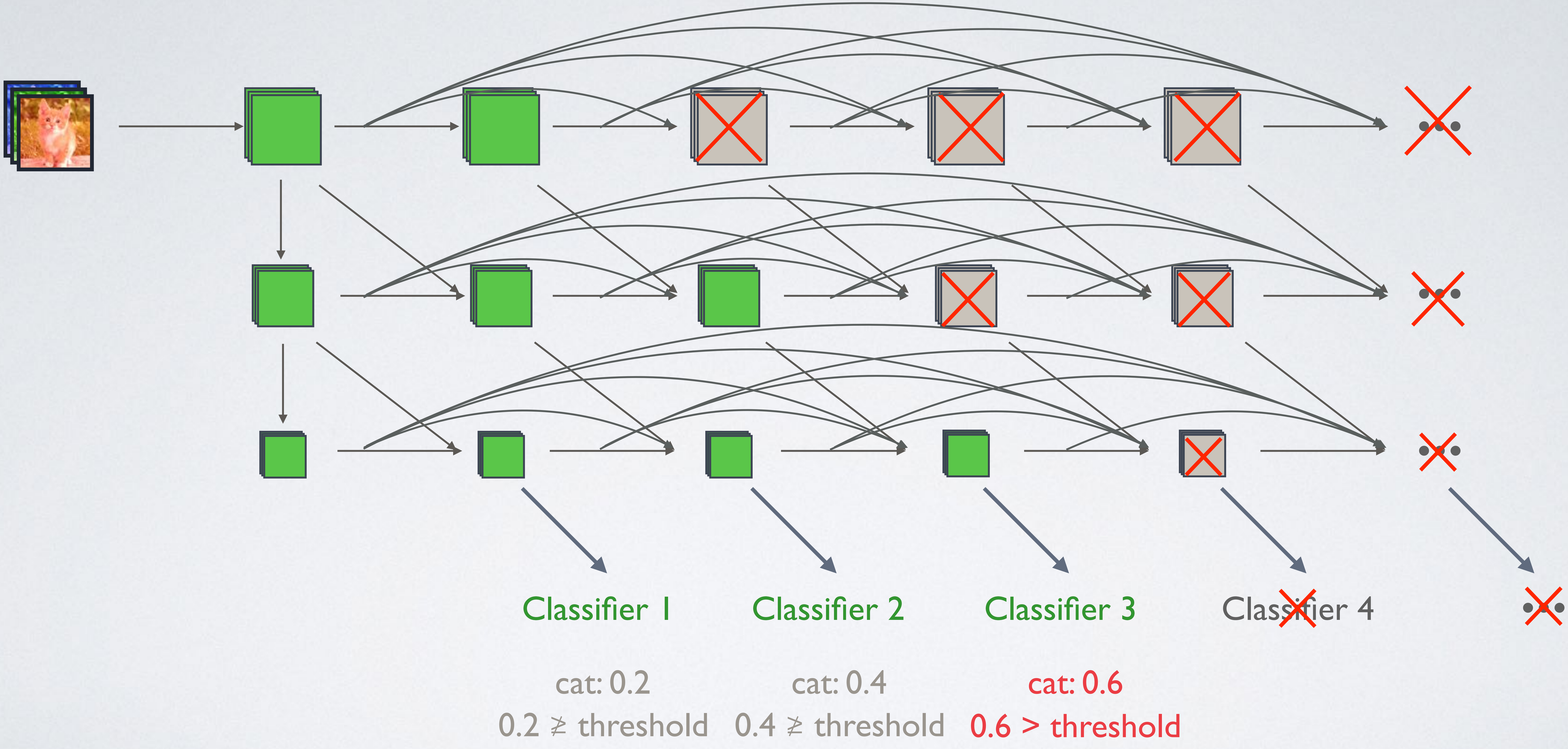
# RESULTS ON **IMAGENET**



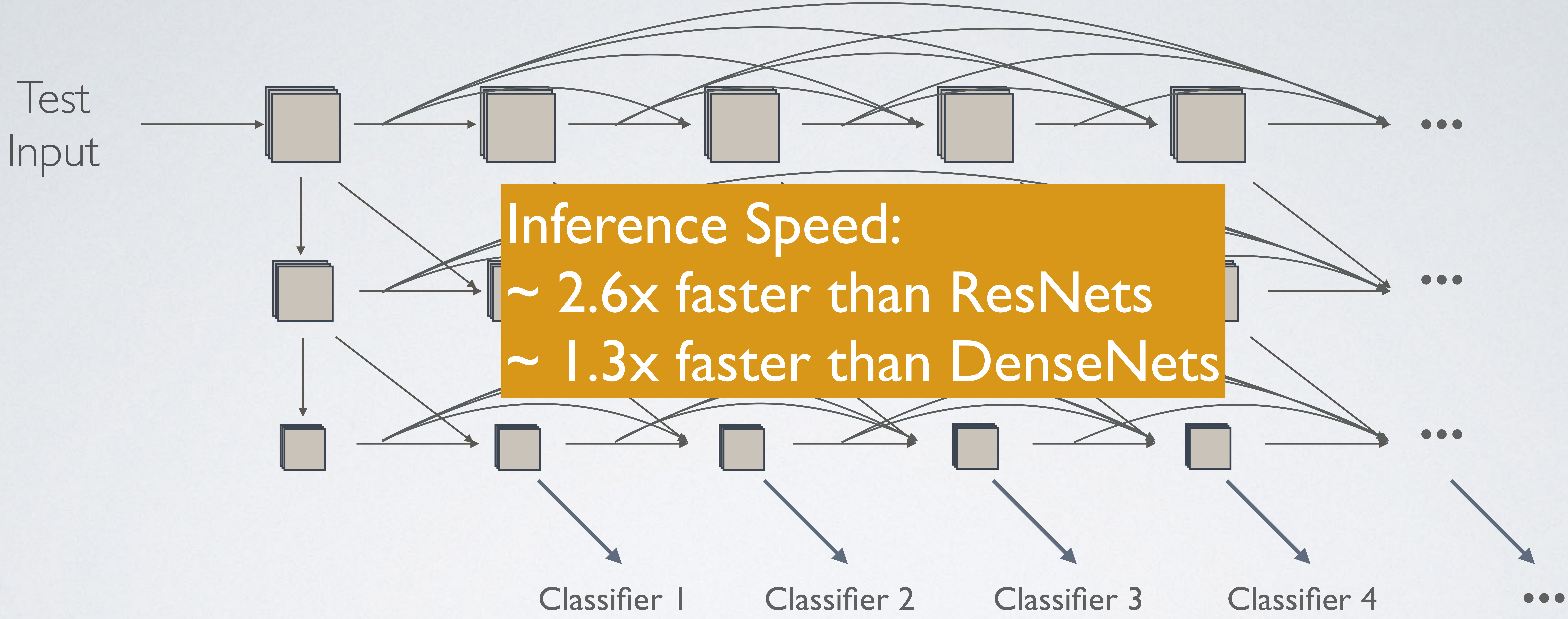

Top-1: 20.27%
Top-5: 5.17%

# MULTI-SCALE DENSENET (Preview)




*Multi-Scale DenseNet: [Huang, Chen, Li, Wu, van der Maaten, Weinberger] (arXiv Preprint: 1703.09844)*

# MULTI-SCALE DENSENET (Preview)




**Inference Speed:**
~ 2.6x faster than ResNets
~ 1.3x faster than DenseNets

"Easy" examples 

"Hard" examples 

NEW *Memory efficient Torch implementation:* NEW

*https://github.com/liuzhuang13/DenseNet*

## Other implementations:

Our Caffe Implementation
Our memory-efficient Caffe Implementation.
Our memory-efficient PyTorch Implementation.
PyTorch Implementation by Andreas Veit.
PyTorch Implementation by Brandon Amos.
MXNet Implementation by Nicatio.
MXNet Implementation (supports ImageNet) by Xiong Lin.
Tensorflow Implementation by Yixuan Li.
Tensorflow Implementation by Laurent Mazare.

Tensorflow Implementation (with BC structure) by Illarion Khlestov.
Lasagne Implementation by Jan Schlüter.
Keras Implementation by tdeboissiere.
Keras Implementation by Roberto de Moura Estevão Filho.
Keras Implementation (with BC structure) by Somshubra Majumdar.
Chainer Implementation by Toshinori Hanya.
Chainer Implementation by Yasunori Kudo.

# REFERENCES


- Kaiming He, et al. "Deep residual learning for image recognition" CVPR 2016
- Chen-Yu Lee, et al. "Deeply-supervised nets" *AISTATS* 2015
- Gao Huang, et al. "Deep networks with stochastic depth" *ECCV* 2016
- Gao Huang, et al. "Multi-Scale Dense Convolutional Networks for Efficient Prediction" *arXiv preprint arXiv:1703.09844* (2017)
- Geoff Pleiss, et al. "Memory-Efficient Implementation of DenseNets'', *arXiv preprint arXiv:1707.06990* (2017)