BOOTCAMP - AMMI 2021

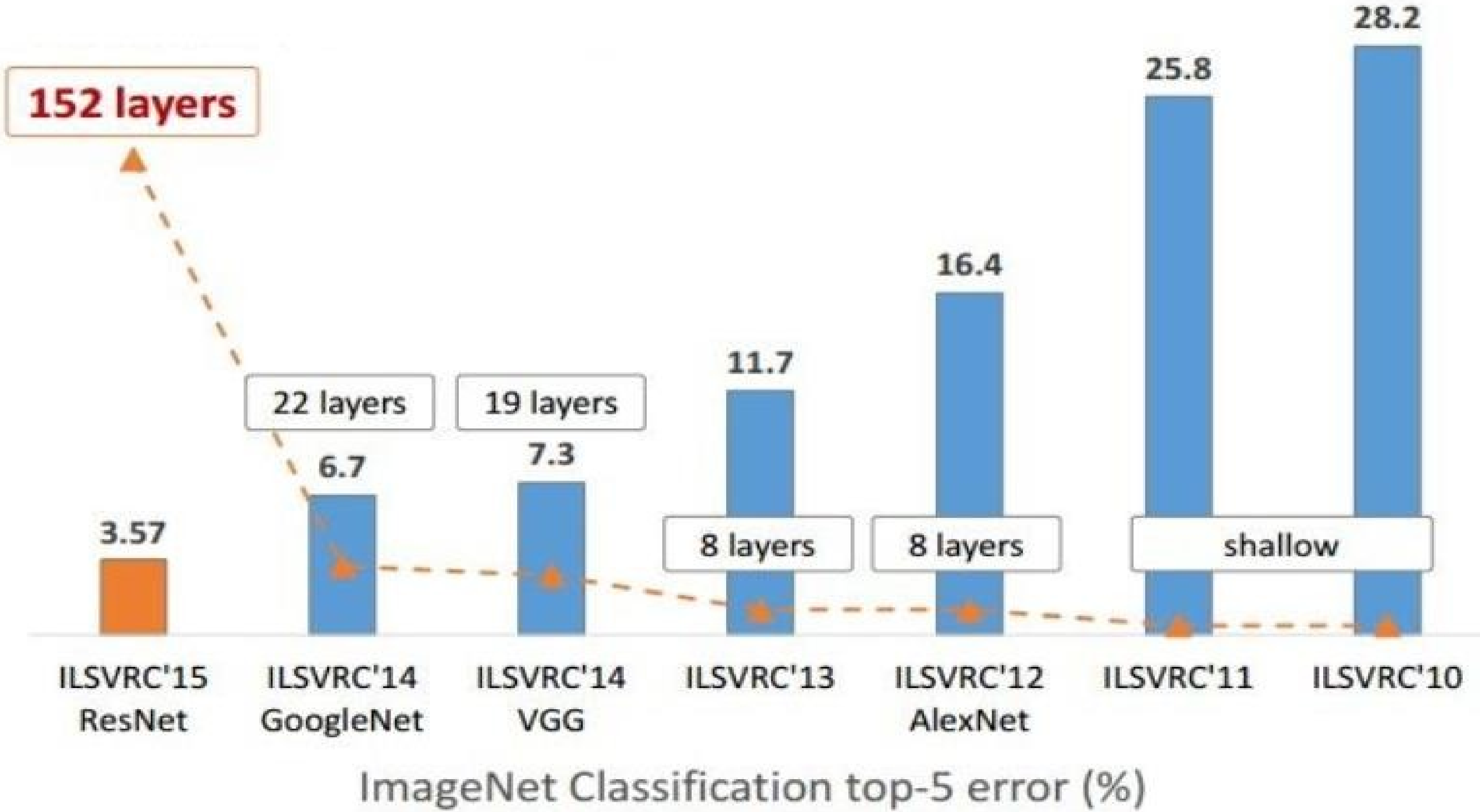
Densely Connected Convolutional Networks: DenseNet

Presented by:

Emmanuel NGENZIRABONA

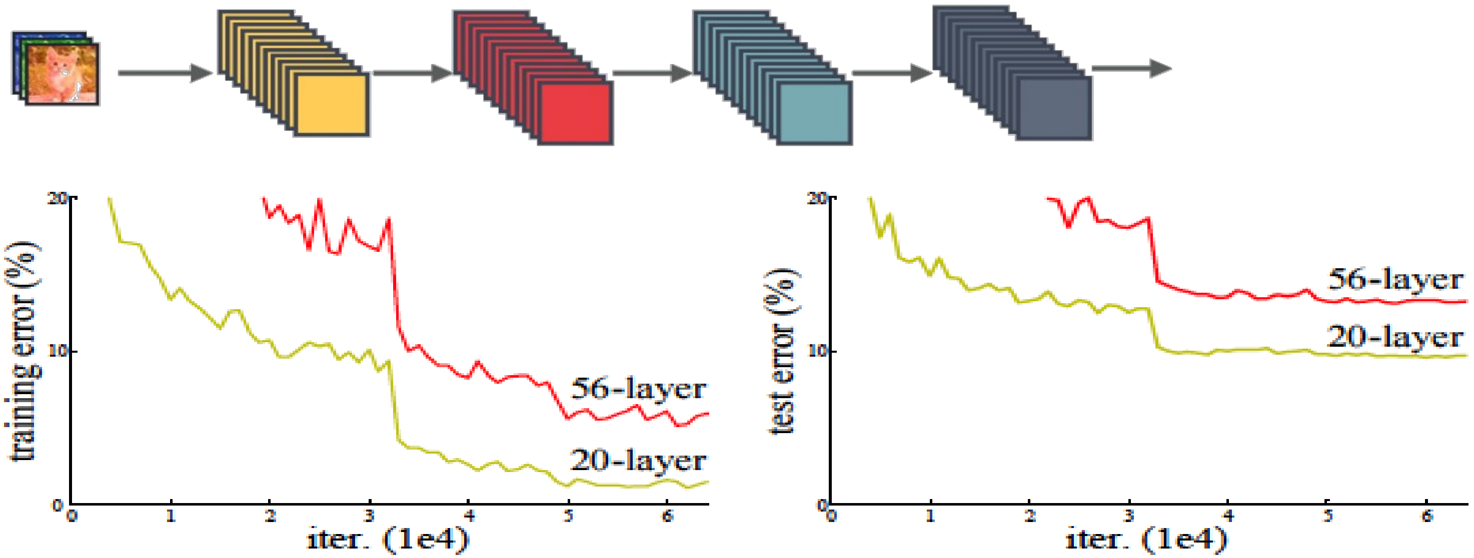
August 20, 2021

Evolution of Depth in CNNs

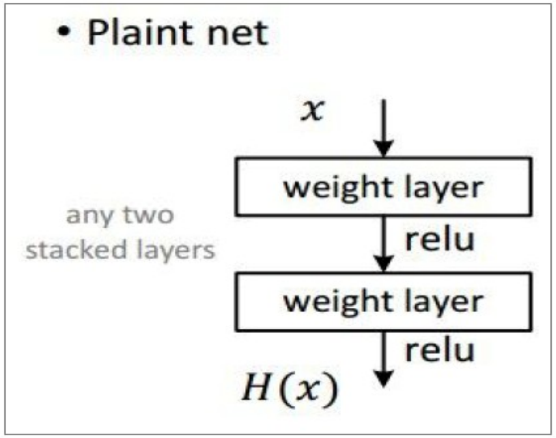


The Degradation

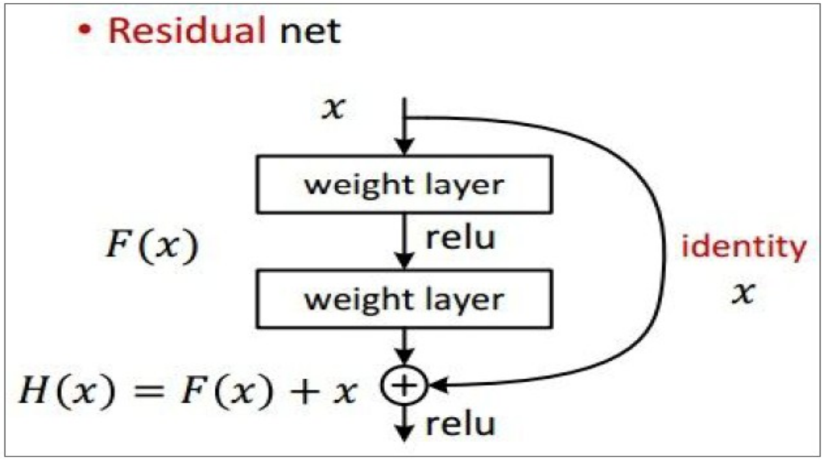
- Normalized initialization and intermediate normalization layers
- The main culprit : Vanishing/exploding gradients
- Not caused by overfitting



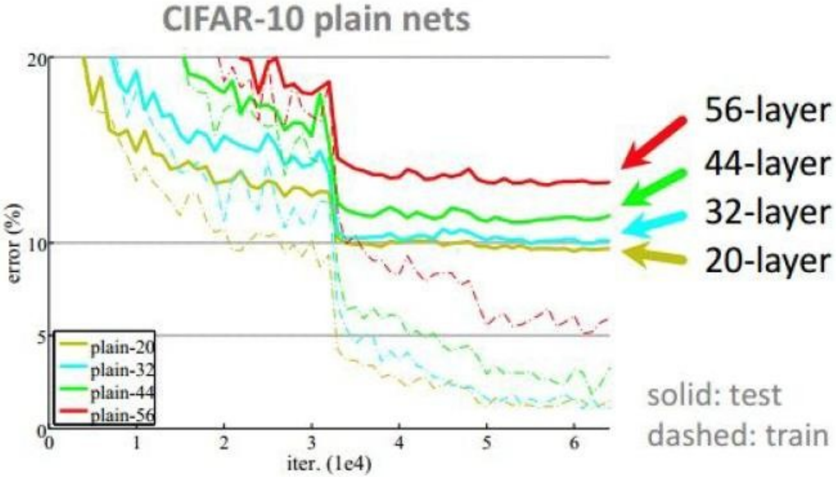
RESNET



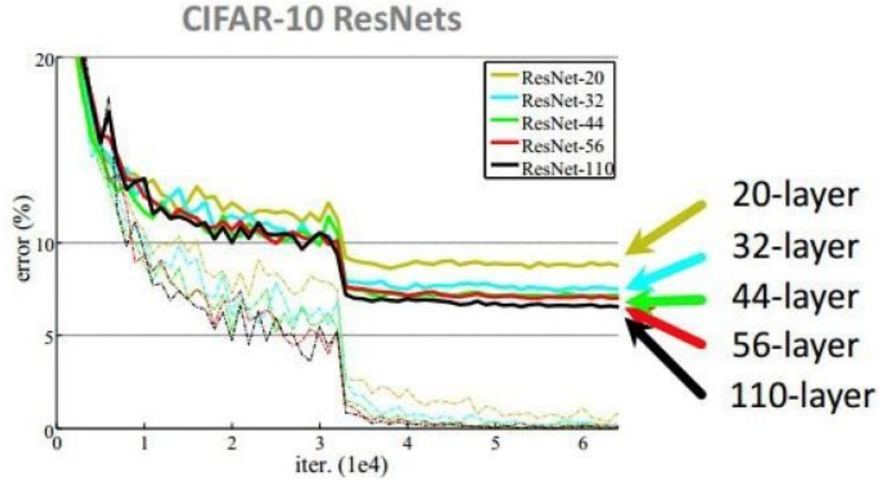
• Plaint net
x
weight layer
relu
weight layer
relu
any two
stacked layers
H(x)



• Residual net
x
weight layer
relu
weight layer
F(x)
identity
x
H(x) = F(x) + x
relu



CIFAR-10 plain nets
error (%)
iter. (1e4)
plain-20
plain-32
plain-44
plain-56
56-layer
44-layer
32-layer
20-layer
solid: test
dashed: train

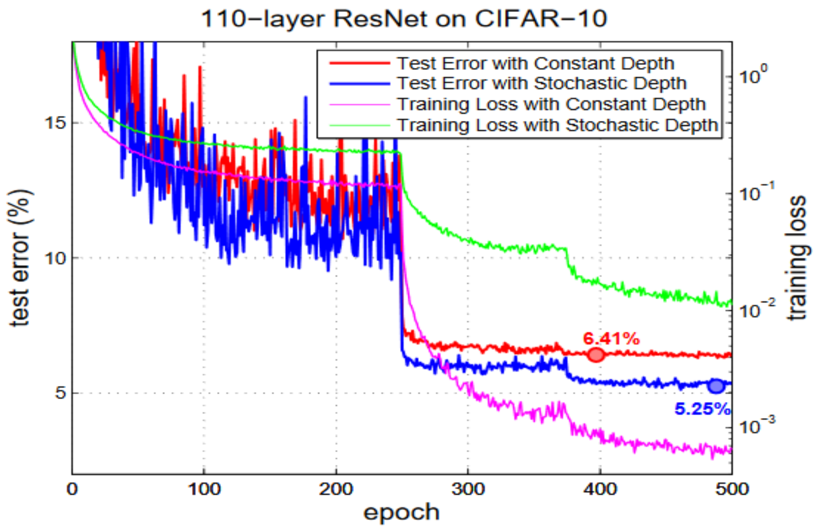


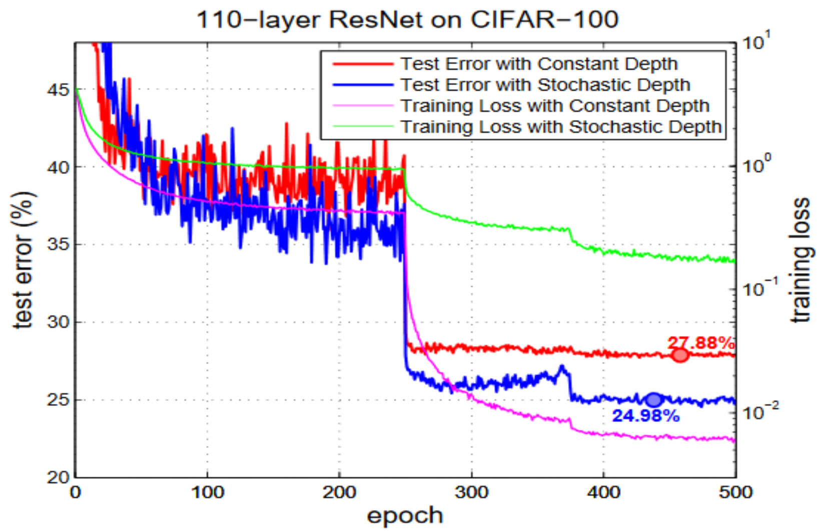
CIFAR-10 ResNets
error (%)
iter. (1e4)
ResNet-20
ResNet-32
ResNet-44
ResNet-56
ResNet-110
20-layer
32-layer
44-layer
56-layer
110-layer

STOCHASTIC DEPTH

Deep network during testing, but shallower network during training.

$$H_l = ReLU(b_l f_l(H_{l-1}) + id(H_{l-1})) \qquad b_l \in \{0,1\}$$

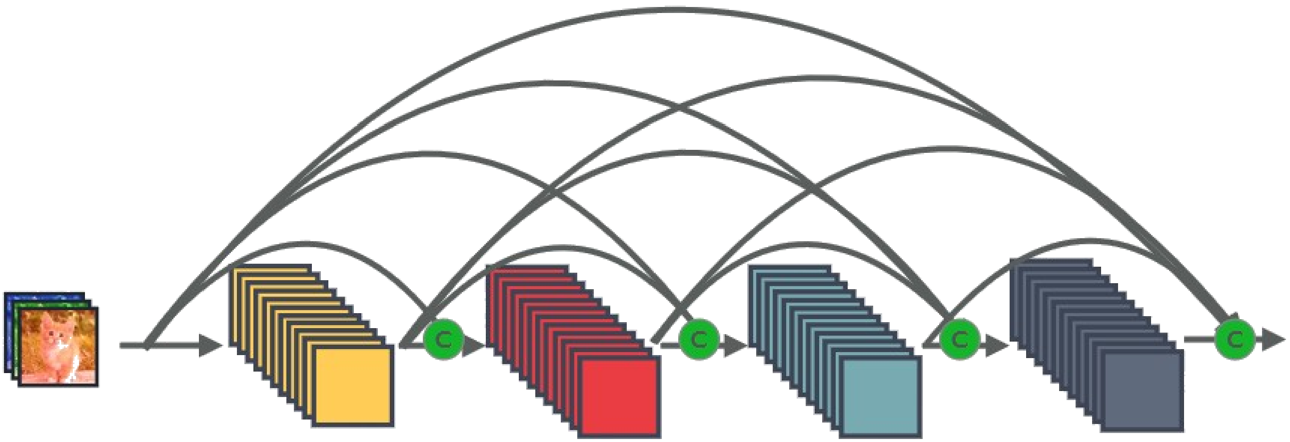




They all share a key characteristic:
They create short paths from early layers to later layers

DENSE CONNECTIVITY

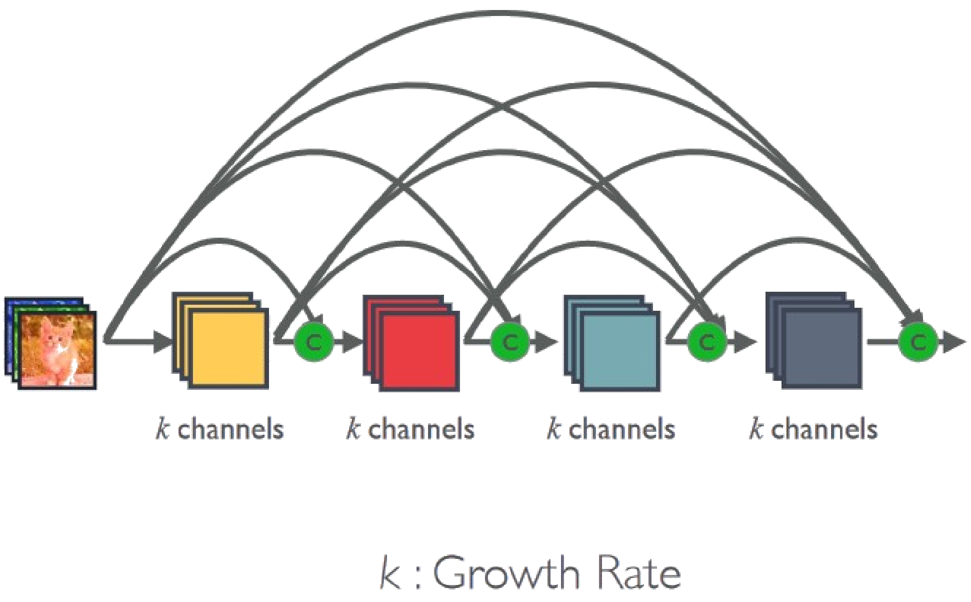
$\frac{l\ (l+1)}{2}$ direct connections



: Channel-wise concatenation

DENSE AND SLIM

- The growth rate regulates how much new information each layer contributes to the global state.



- ResNets :

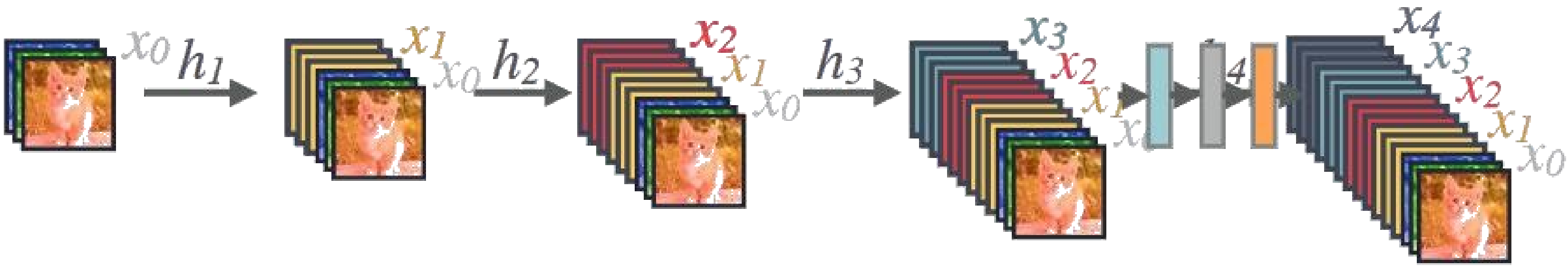
$$x_l = H_l\,(x_{l-1}) + x_{l-1}$$

- DenseNets :

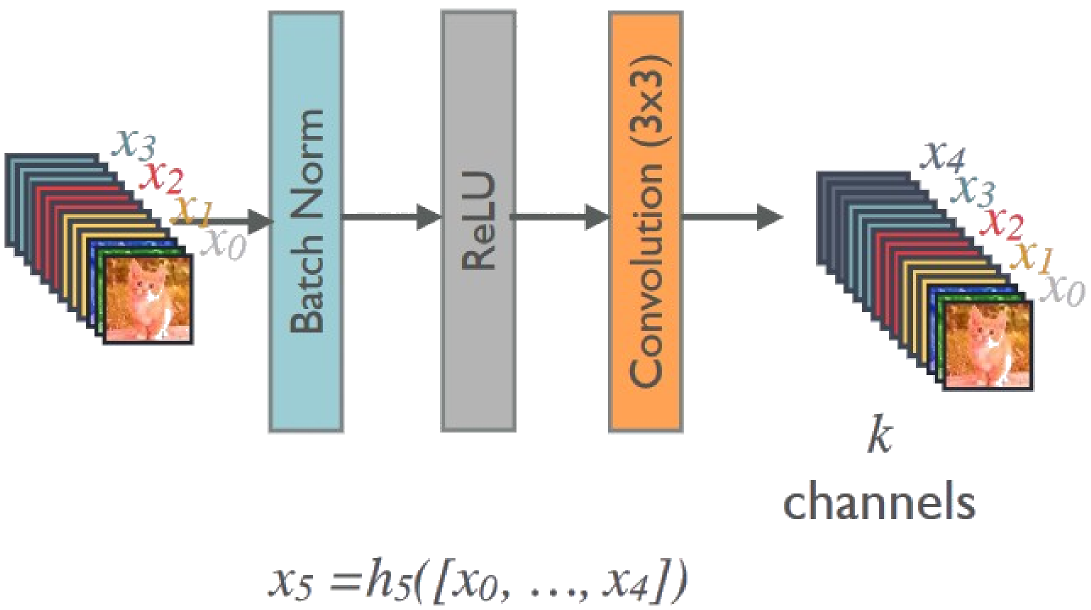
$$x_l = H_l\,([x_0, x_1, \dots, x_{l-1}])$$

Where $[x_0, x_1, \dots, x_{l-1}]$ refers to the concatenation of the feature-maps p
in layers 0….. l-1.

Forward Propagation

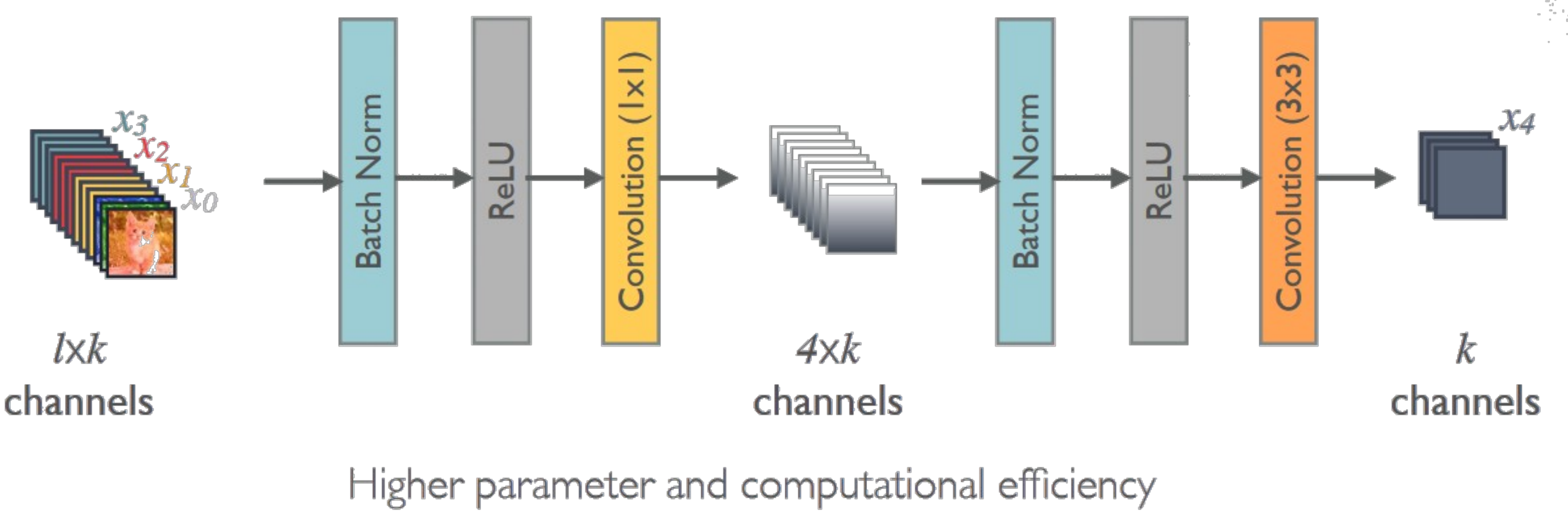


Composite Layer in DENSENET



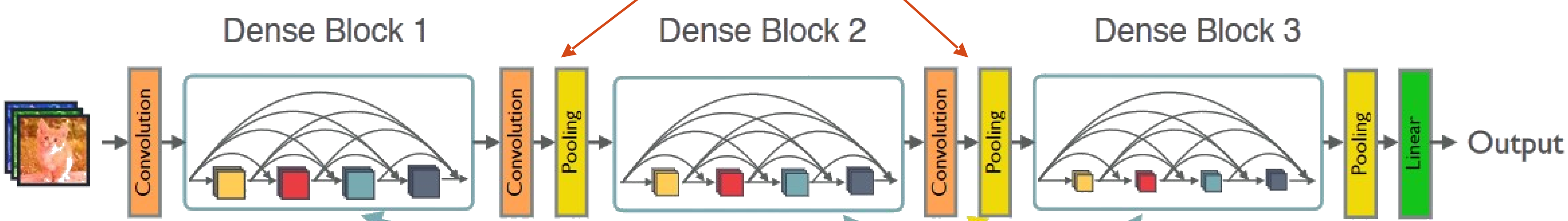
Composite Layer in DenseNet

WITH BOTTLENECK LAYER



DENSENET

Compression in transition layer



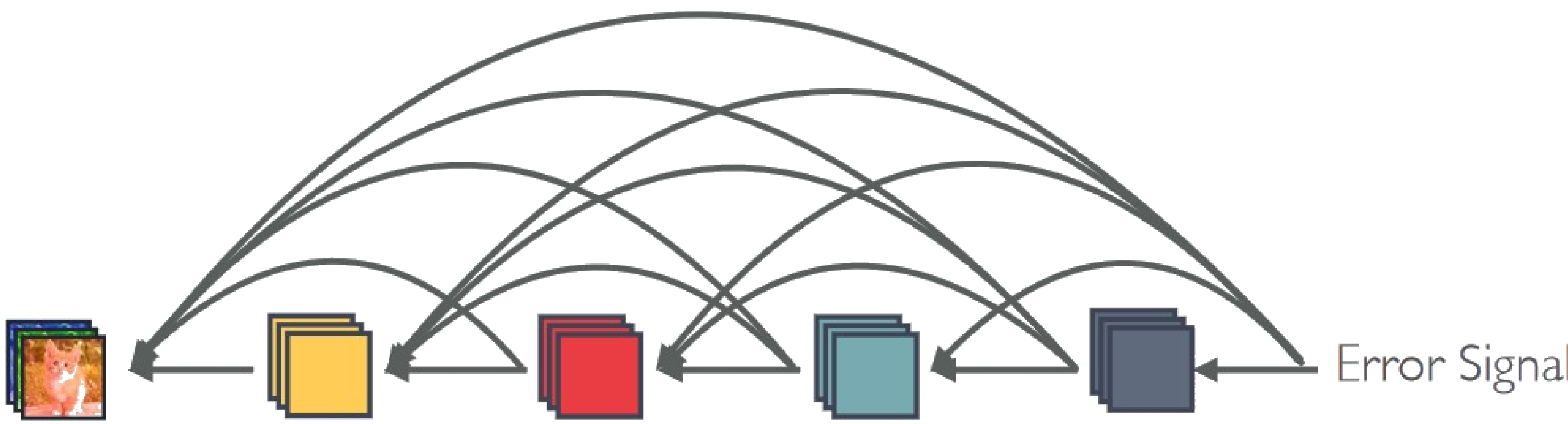
Pooling reduces
Feature map sizes

Feature map sizes match
Within each block

Advantages of Dense Connectivity

Advge1. Strong Gradient Flow

- Direct access: Deep Supervision with single classifier
- Reduces overfitting on tasks with smaller training set sizes

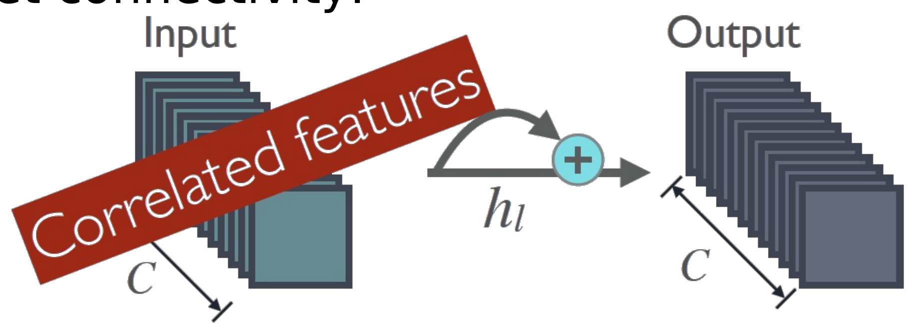


Comparison between Architectures

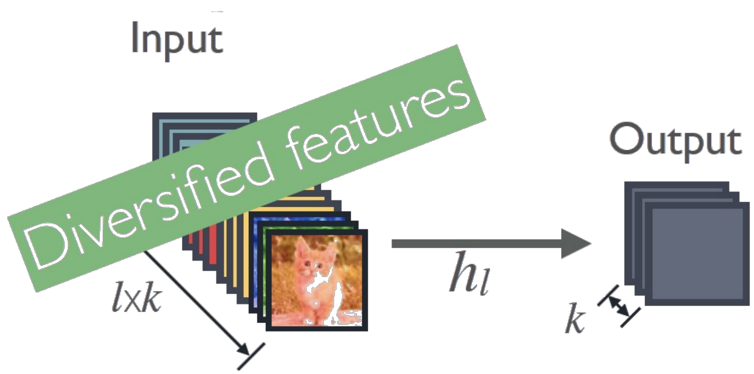
Method	Depth	Params	C10	C10+	C100	C100+	SVHN
Network in Network [22]	-	-	10.41	8.81	35.68	-	2.35
All-CNN [31]	-	-	9.08	7.25	-	33.71	-
Deeply Supervised Net [20]	-	-	9.69	7.97	-	34.57	1.92
Highway Network [33]	-	-	-	7.72	-	32.39	-
FractalNet [17]	21	38.6M	10.18	5.22	35.34	23.30	2.01
with Dropout/Drop-path	21	38.6M	7.33	4.60	28.20	23.73	1.87
ResNet [11]	110	1.7M	-	6.61	-	-	-
ResNet (reported by [13])	110	1.7M	13.63	6.41	44.74	27.22	2.01
ResNet with Stochastic Depth [13]	110	1.7M	11.66	5.23	37.80	24.58	1.75
	1202	10.2M	-	4.91	-	-	-
Wide ResNet [41]	16	11.0M	-	4.81	-	22.07	-
	28	36.5M	-	4.17	-	20.50	-
with Dropout	16	2.7M	-	-	-	-	1.64
ResNet (pre-activation) [12]	164	1.7M	11.26*	5.46	35.58*	24.33	-
	1001	10.2M	10.56*	4.62	33.47*	22.71	-
DenseNet ($k = 12$)	40	1.0M	**7.00**	5.24	**27.55**	24.42	1.79
DenseNet ($k = 12$)	100	7.0M	**5.77**	**4.10**	**23.79**	**20.20**	1.67
DenseNet ($k = 24$)	100	27.2M	**5.83**	**3.74**	**23.42**	**19.25**	**1.59**
DenseNet-BC ($k = 12$)	100	0.8M	**5.92**	4.51	**24.15**	22.27	1.76
DenseNet-BC ($k = 24$)	250	15.3M	**5.19**	**3.62**	**19.64**	**17.60**	1.74
DenseNet-BC ($k = 40$)	190	25.6M	-	**3.46**	-	**17.18**	-

Advge2.
Parameter & Computational efficiency

ResNet connectivity:



DenseNet connectivity:



#parameters:

$O(C \times C)$

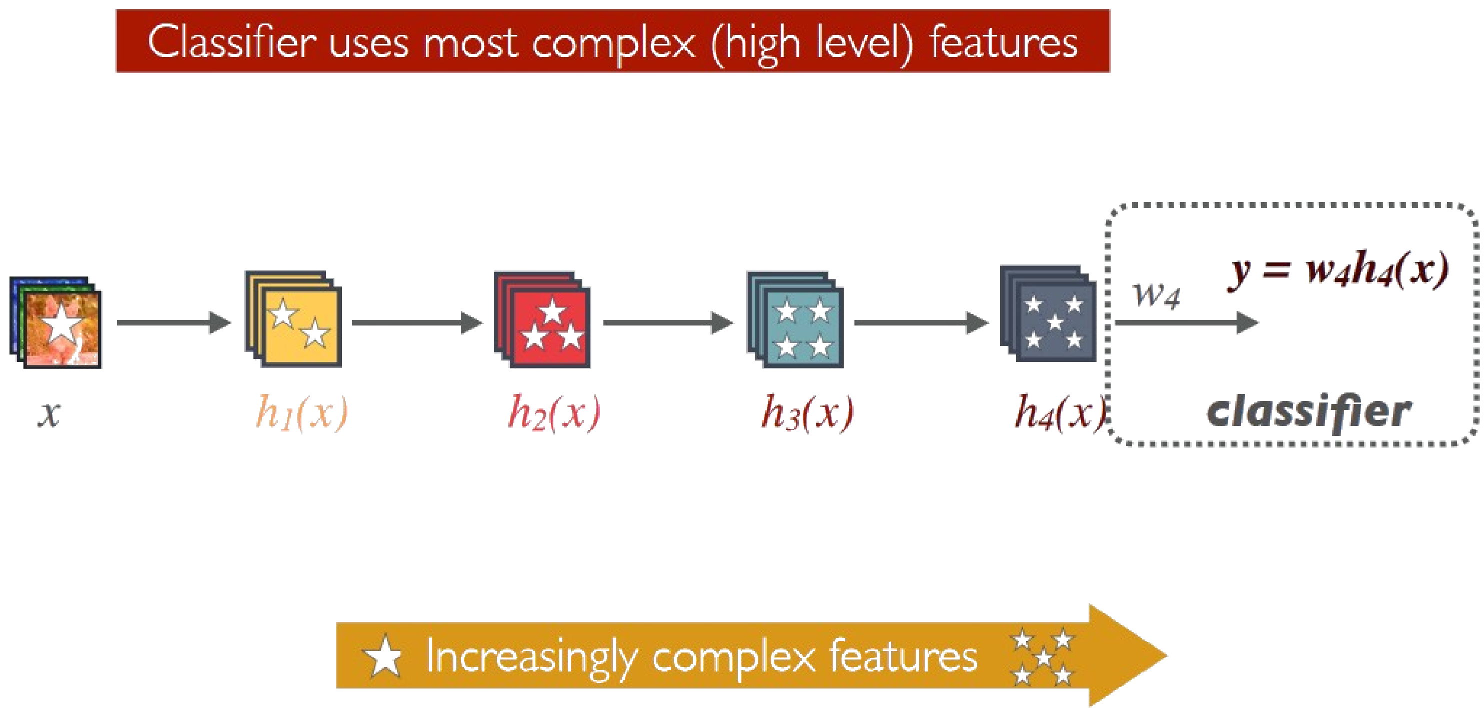
$k << C$

$O(l \times k \times k)$

k: Growth rate

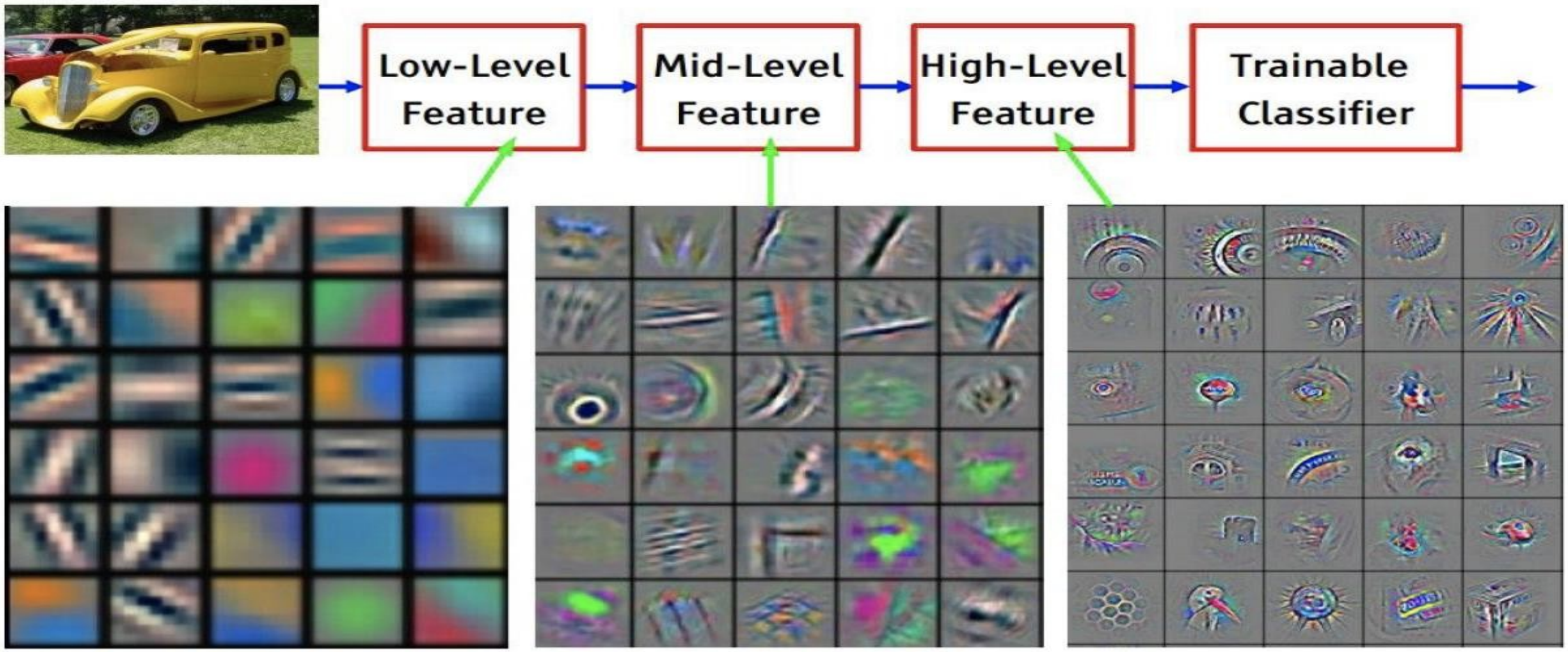
Advge3.
Maintains Low Complexity Features

Standard Connectivity :



Advge3.
Maintains Low Complexity Features

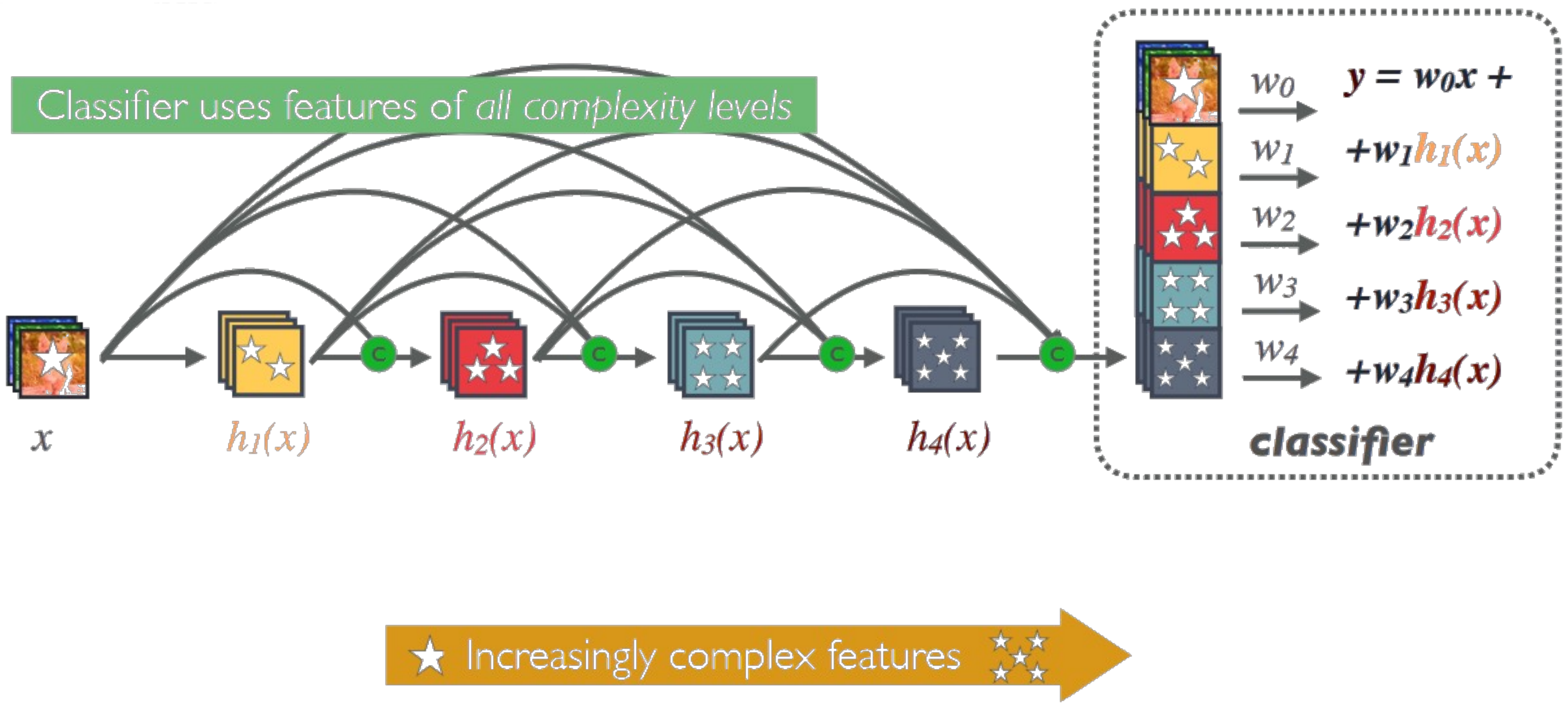
Remember feature visualization



Feature visualization of convolutional net trained on ImageNet

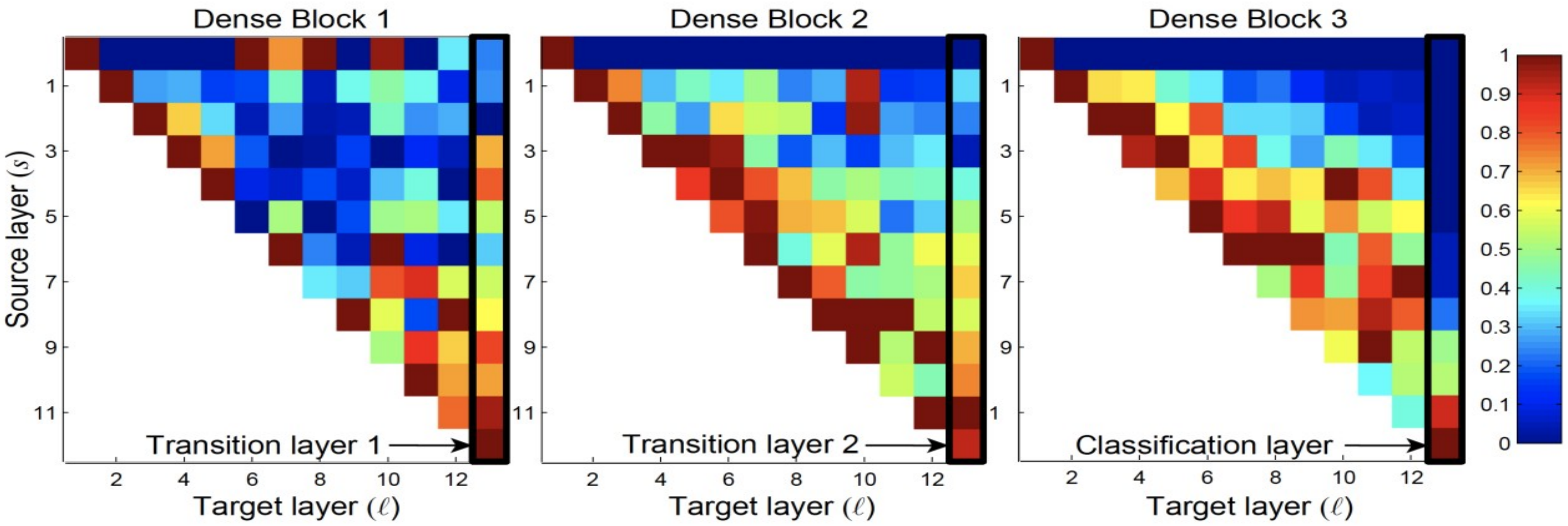
Advge3.
Maintains Low Complexity Features

“Collective Knowledge”



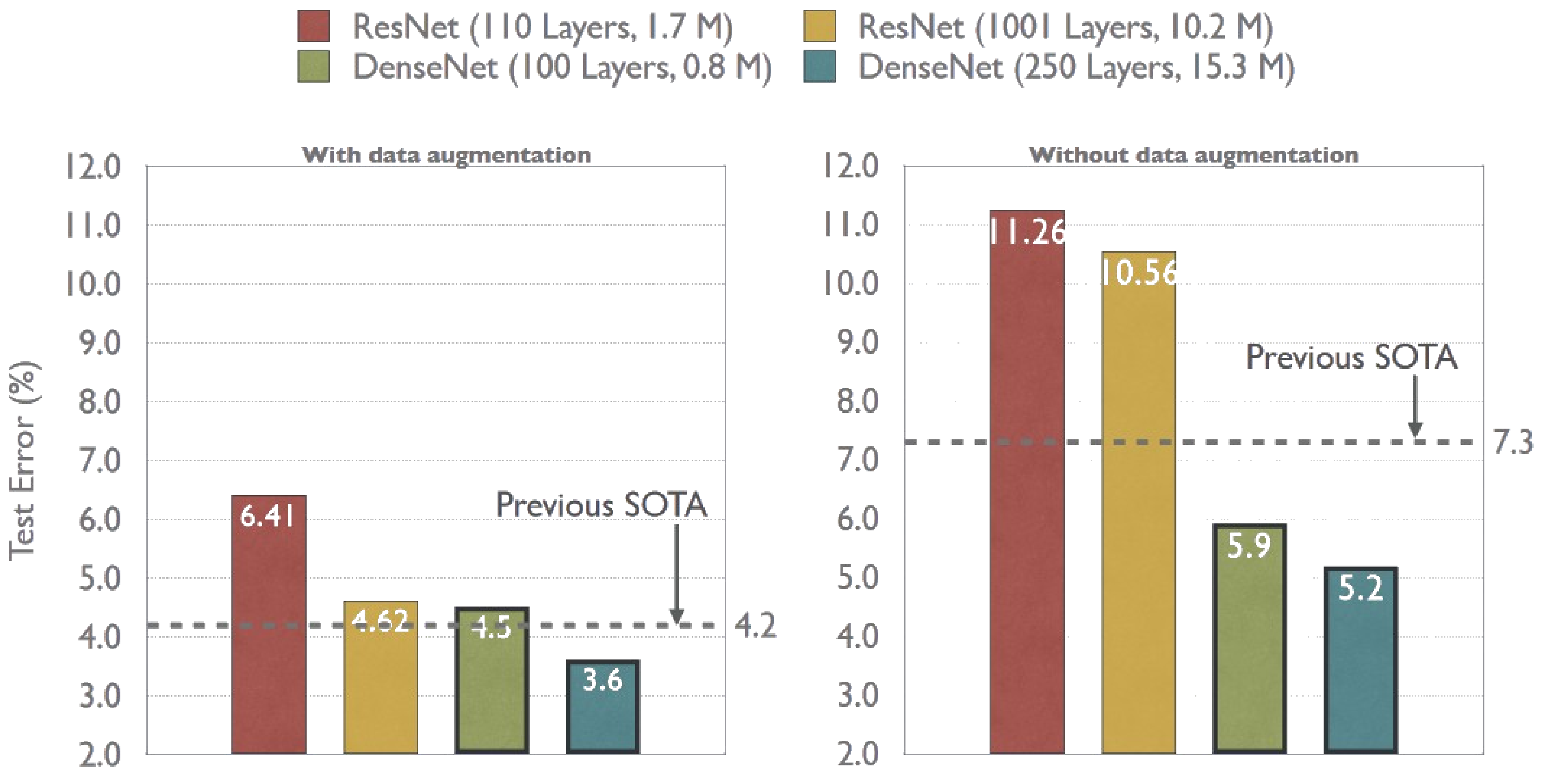
Advge3.
Maintains Low Complexity Features

- Feature reuse
- Information flow from the first to the last layers of the block
- Compression in transition layer
- Concentrate on high level feature for final classification



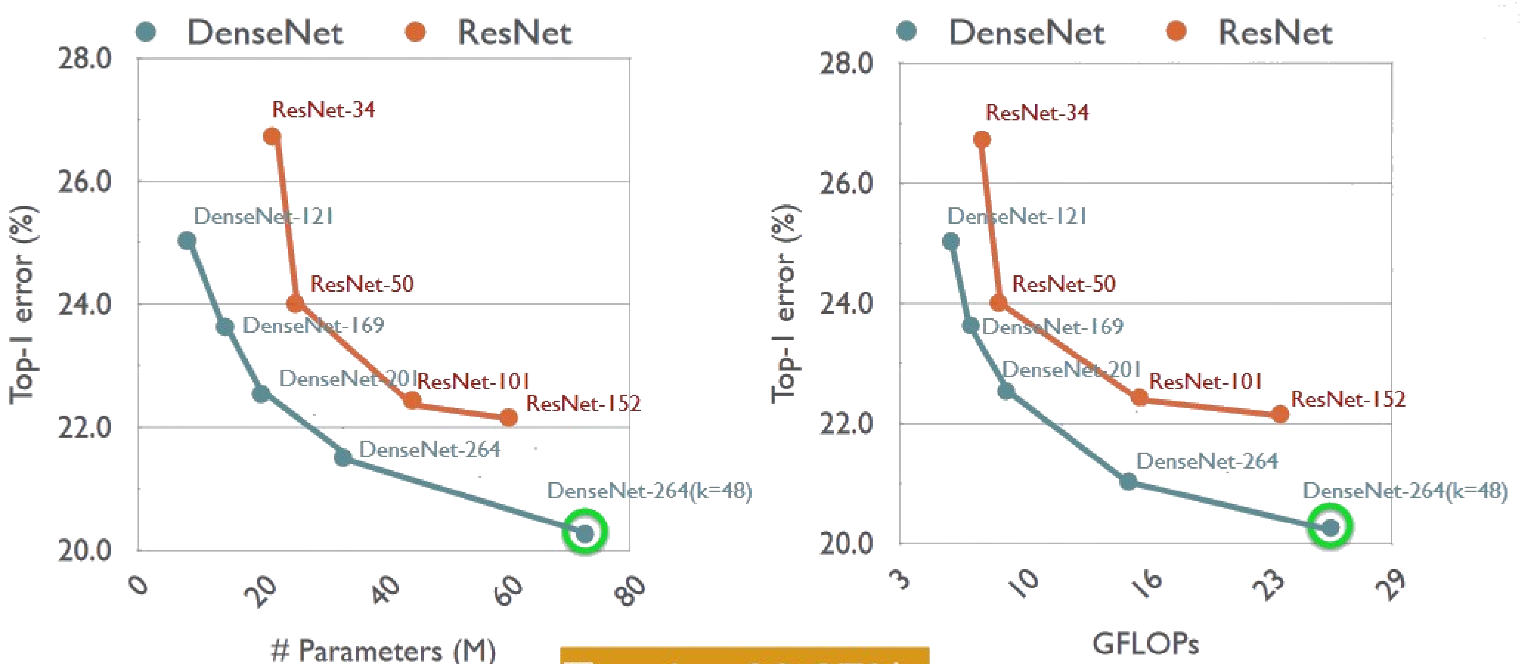
RESULTS

Results on CIFAR-10



Layers	Output Size	DenseNet-121($k = 32$)	DenseNet-169($k = 32$)	DenseNet-201($k = 32$)	DenseNet-161($k = 48$)
Convolution	112×112	7×7 conv, stride 2			
Pooling	56×56	3×3 max pool, stride 2			
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56×56	1×1 conv			
	28×28	2×2 average pool, stride 2			
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28×28	1×1 conv			
	14×14	2×2 average pool, stride 2			
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 36$
Transition Layer (3)	14×14	1×1 conv			
	7×7	2×2 average pool, stride 2			
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$
Classification Layer	1×1	7×7 global average pool			
		1000D fully-connected, softmax			

RESULTS ON IMAGENET



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

References

(1) Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. Conference on Computer Vision and Pattern Recognition, 2017

(2) Kaiming He, et al. "Deep residual learning for image recognition" CVPR

(3) Chen-Yu Lee, et al. "Deeply-supervised nets" AISTATS 2015

(4) CS231n: Convolutional Neural Networks for Visual Recognition

(5) Geoff Pleiss, et al. "Memory-Efficient Implementation of DenseNets", arXiv preprint arXiv:1707.06990 (2017)

THANK YOU!!!