COMP/EECE 7/8740 Neural Networks

Topics:

- CNN Architectures
  - Variant of ResNet architectures
  - Res2Net
  - Hybrid Networks
  - FactralNet and DenseNet
  - MobileNet
  - ConvNext and JEPA and so on

Md Zahangir Alom
Department of Computer Science
University of Memphis, TN

What were the trends until 2020?

- The backbone convolutional neural networks (CNNs) continually demonstrate stronger **multi-scale representation ability**
- Leading to **consistent performance gains** in a wide range of applications.

- Trends (~ 2020):
  - **Minor modifications to ResNets**
  - Biggest trend is to **split of into several branches, and merge through summation**
  - A couple architectures **go crazy with branch & merge**, without explicit identity connections

What were the trends until 2020?

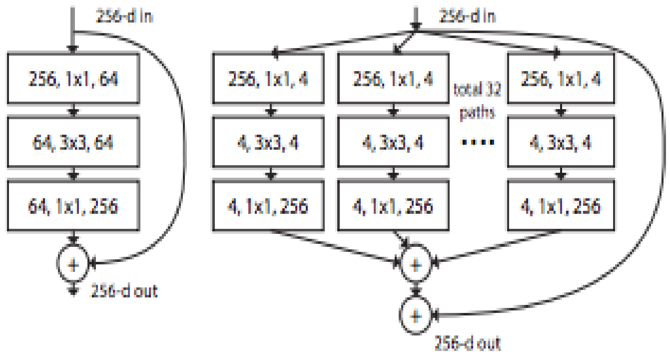


Figure 1. **Left**: A block of ResNet [13]. **Right**: A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels).

ResNeXt

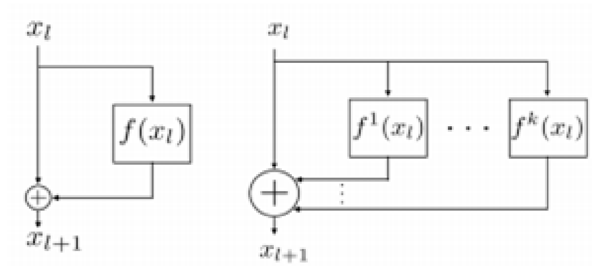


Figure 2: A residual block (left) versus a multi-residual block (right).

MultiResNet

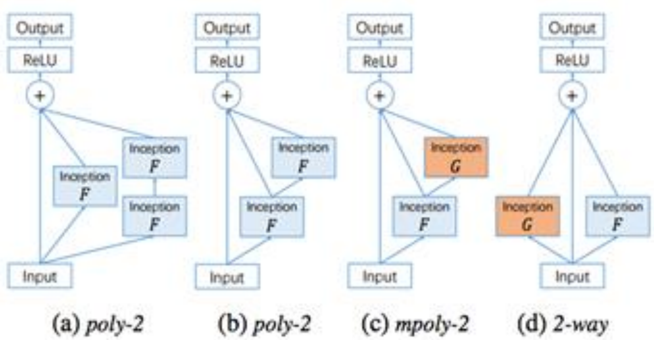
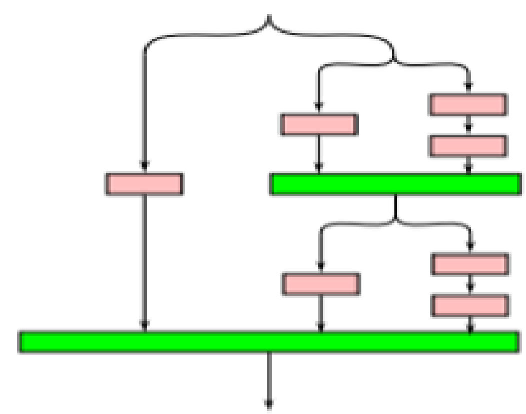


Figure 4: Examples of PolyInception structures.

PolyNet



FactralNet

Aggregated Residual Transformations

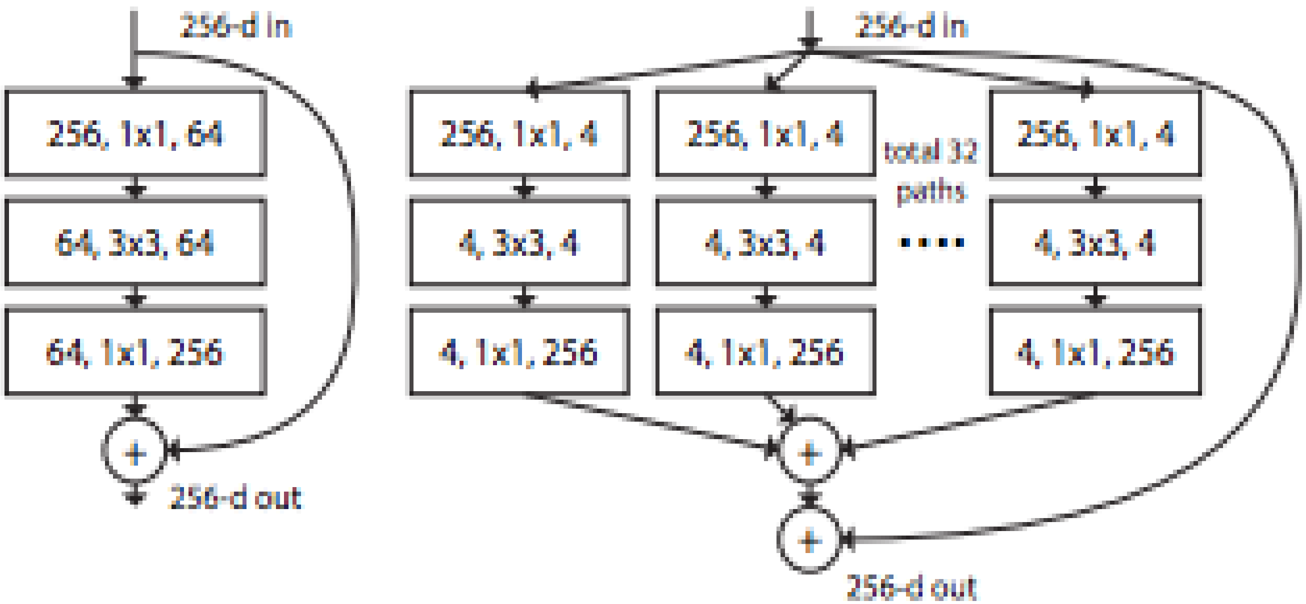
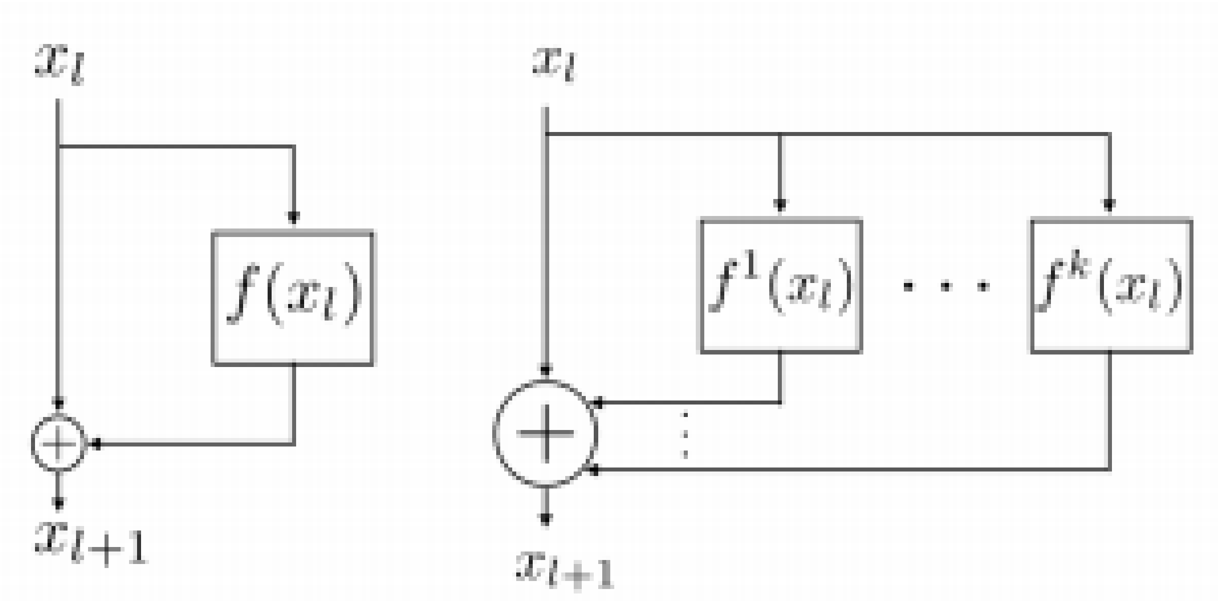


Figure 1. **Left:** A block of ResNet [13]. **Right:** A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels).

Xie, Saining, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. "Aggregated residual transformations for deep neural networks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492-1500. 2017.

Multi-Residual Networks (ResNet)

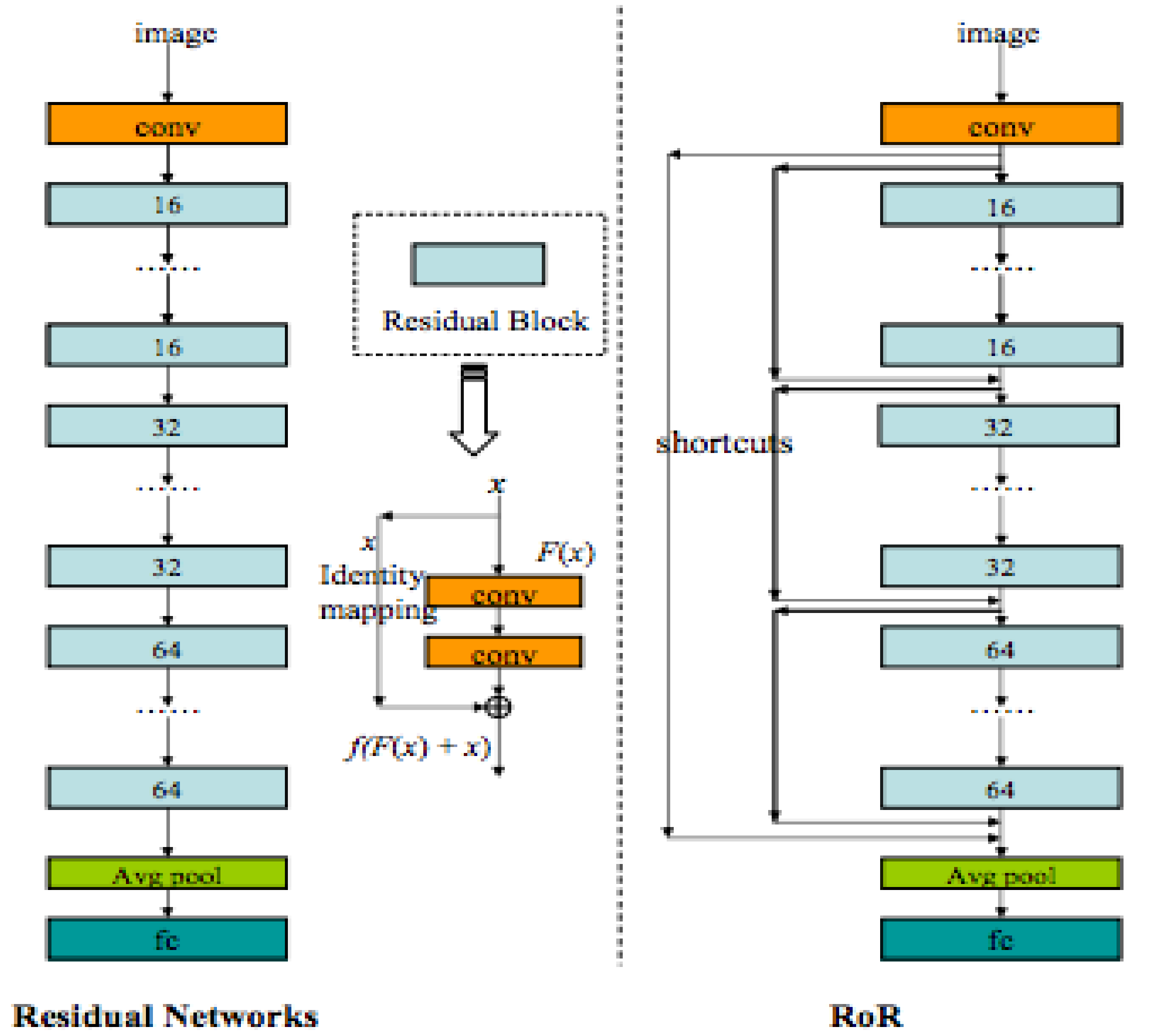


Multi-Residual Networks (ResNet) : residual block (left) versus a multi-residual block (right)

Masoud Abdi, Saeid Nahavandi. **Multi-Residual Networks: Improving the Speed and Accuracy of Residual Networks**

Residuals of Residual: try going meta

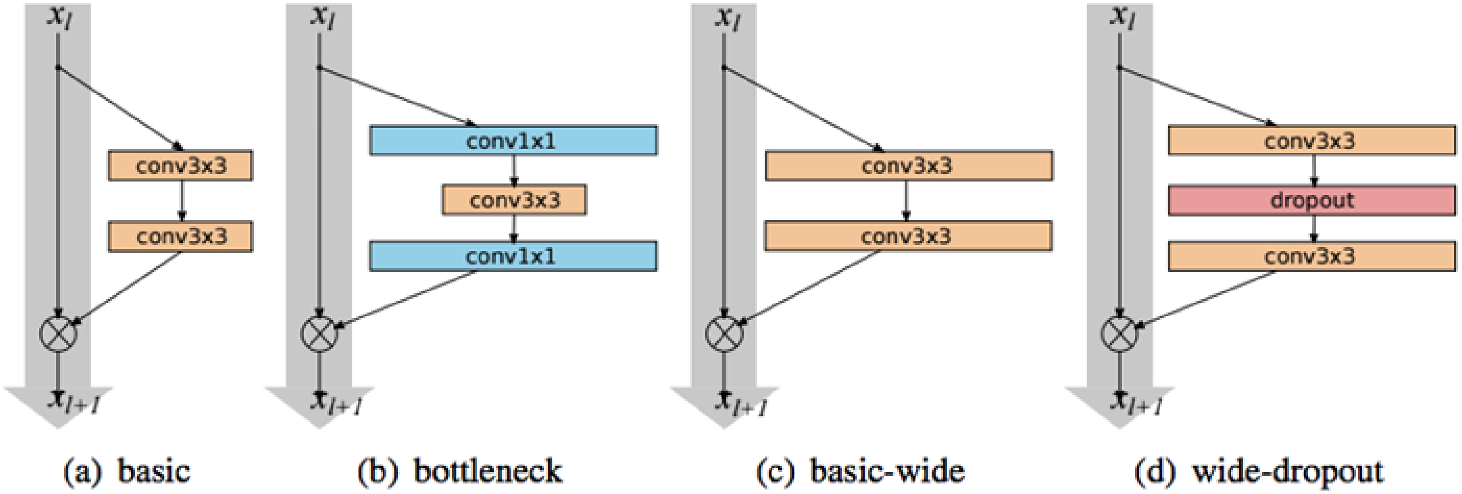
- Residuals of Residuals:



ResNet tweaks: Wide ResNets

- **Use pre-activation ResNet's basic block with more feature maps**

- Used parameter *"k"* to encode width
- Investigated relationship between width and depth to find a good tradeoff



Various residual blocks used in the paper. Batch normalization and ReLU precede each convolution (omitted for clarity)

Sergey Zagoruyko, Nikos Komodakis Wide Residual Networks

ResNet tweaks: Wide ResNets

- Use pre-activation ResNet's basic block with more feature maps
- **Used parameter "k" to encode width**
- Investigated relationship between width and depth to find a good tradeoff

group name	output size	block type = $B(3,3)$
conv1	32×32	$[3\times3, 16]$
conv2	32×32	$\begin{bmatrix} 3\times3, 16\times k \\ 3\times3, 16\times k \end{bmatrix} \times N$
conv3	16×16	$\begin{bmatrix} 3\times3, 32\times k \\ 3\times3, 32\times k \end{bmatrix} \times N$
conv4	8×8	$\begin{bmatrix} 3\times3, 64\times k \\ 3\times3, 64\times k \end{bmatrix} \times N$
avg-pool	1×1	$[8 \times 8]$

Structure of wide residual networks. ResNet block of type B(3,3). Network width is determined by factor k. . Groups of convolutions are shown in brackets where N is a number of blocks in group

Sergey Zagoruyko, Nikos Komodakis Wide Residual Networks

ResNet tweaks: Wide ResNets

- Use pre-activation ResNet's basic block with more feature maps

- Used parameter "k" to encode width
- **Investigated relationship between width and depth to find a good tradeoff**

depth	k	# params	CIFAR-10	CIFAR-100
40	1	0.6M	6.85	30.89
40	2	2.2M	5.33	26.04
40	4	8.9M	4.97	22.89
40	8	35.7M	4.66	-
28	10	36.5M	**4.17**	20.50
28	12	52.5M	4.33	**20.43**
22	8	17.2M	4.38	21.22
22	10	26.8M	4.44	20.75
16	8	11.0M	4.81	22.07
16	10	17.1M	4.56	21.59

Test error (%) of various wide networks on CIFAR-10 and CIFAR-100 (ZCA preprocessing).

Zagoruyko, Sergey, and Nikos Komodakis. "Wide residual networks." *arXiv preprint arXiv:1605.07146* (2016).

ResNet tweaks: Wide ResNets

- These obtained **state of the art results on CIFAR datasets**
- Were outperformed by bottlenecked networks on ImageNet
- **Best results on ImageNet were obtained by widening ResNet-50**

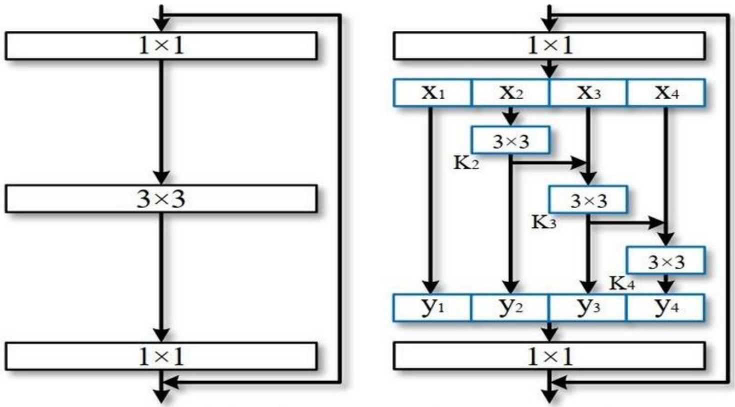
Model	top-1 err, %	top-5 err, %	#params	time/batch 16
ResNet-50	24.01	7.02	25.6M	49
ResNet-101	22.44	6.21	44.5M	82
ResNet-152	22.16	6.16	60.2M	115
WRN-50-2-bottleneck	21.9	6.03	68.9M	93
pre-ResNet-200	21.66	5.79	64.7M	154

"With widening factor of 2.0 the resulting WRN-50-2-bottleneck outperforms ResNet- 152 having 3 times less layers, and being significantly faster."

Zagoruyko, Sergey, and Nikos Komodakis. "Wide residual networks." *arXiv preprint arXiv:1605.07146* (2016).

Res2Net: A New Multi-scale Backbone Architecture

- **Res2Net** model represents the multi-scale features in a layer-wise manner

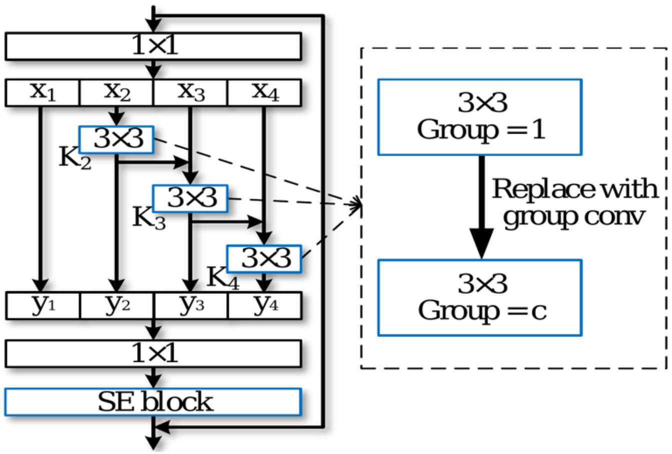


(a) Bottleneck (b) Res2Net module

- A novel building block for CNNs, namely **Res2Net**, by constructing **hierarchical residual-like connections within one single residual block**.
- The **Res2Net represents** multi-scale features at a **granular level and increases the range of receptive fields** for each network layer

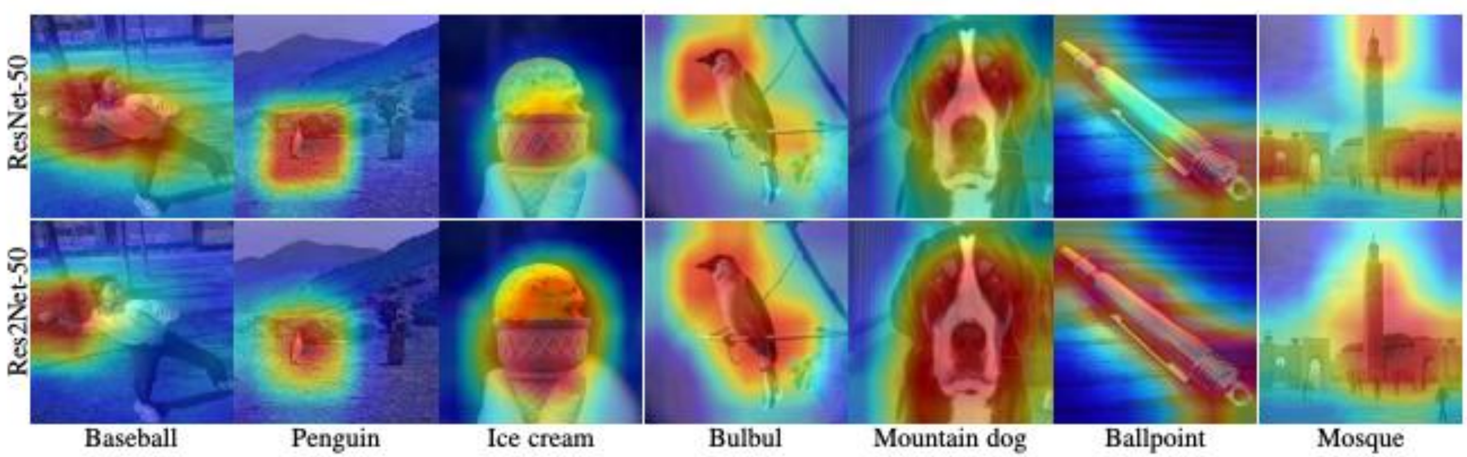
Gao, Shanghua, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip HS Torr. "Res2net: A new multi-scale backbone architecture." *IEEE transactions on pattern analysis and machine intelligence* (2019).

Res2Net: A New Multi-scale Backbone Architecture



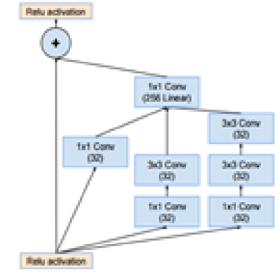
The Res2Net module can be integrated with the dimension cardinality (replace conv with group conv) and SE blocks.

- The proposed **Res2Net block can be plugged into the state-of-the-art backbone CNN models**, e.g., ResNet, ResNeXt, and DLA

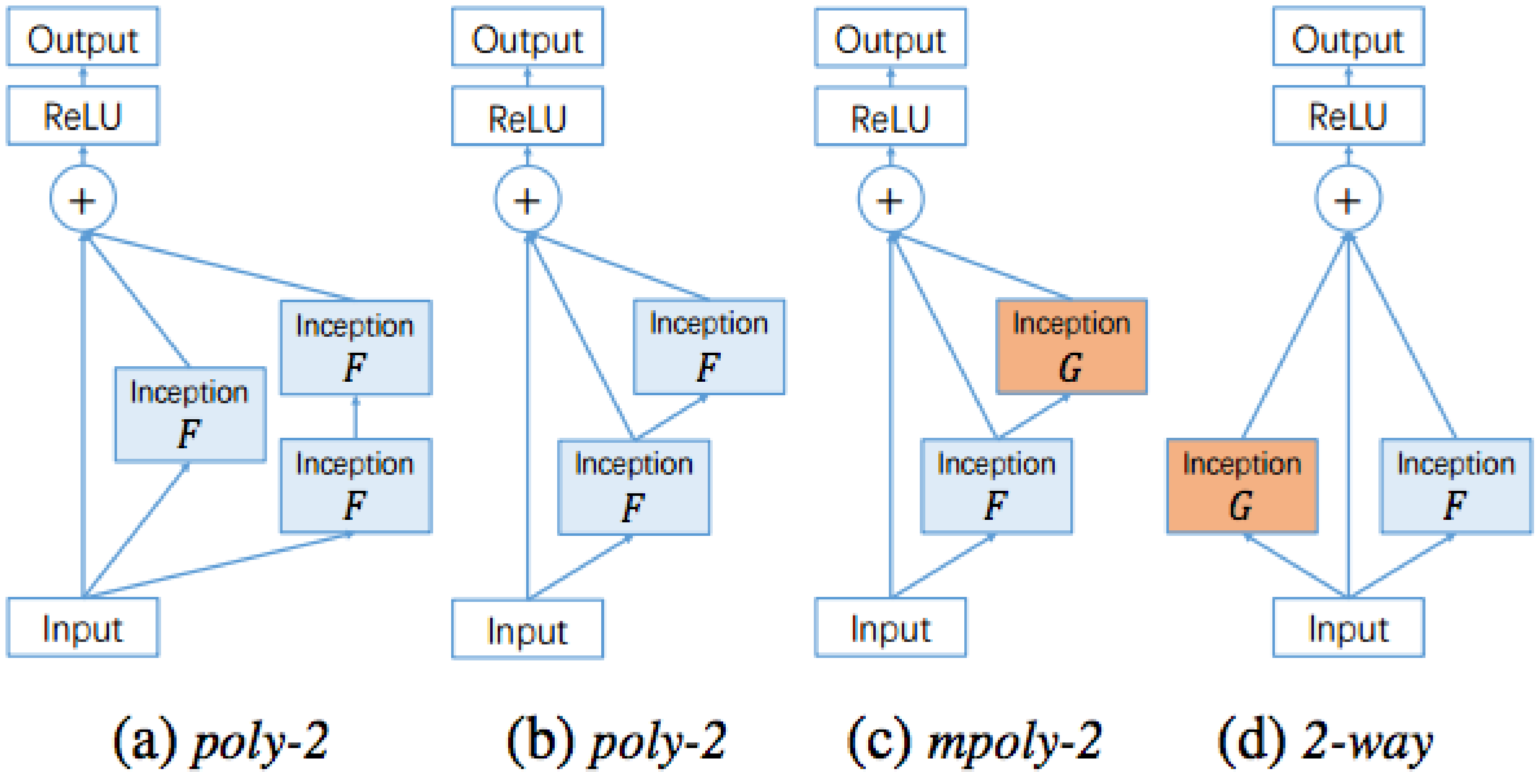


Visualization of class activation mapping, using ResNet-50 and Res2Net-50 as backbone networks.

PolyNet



Inception ResNet



PolyNet Structures

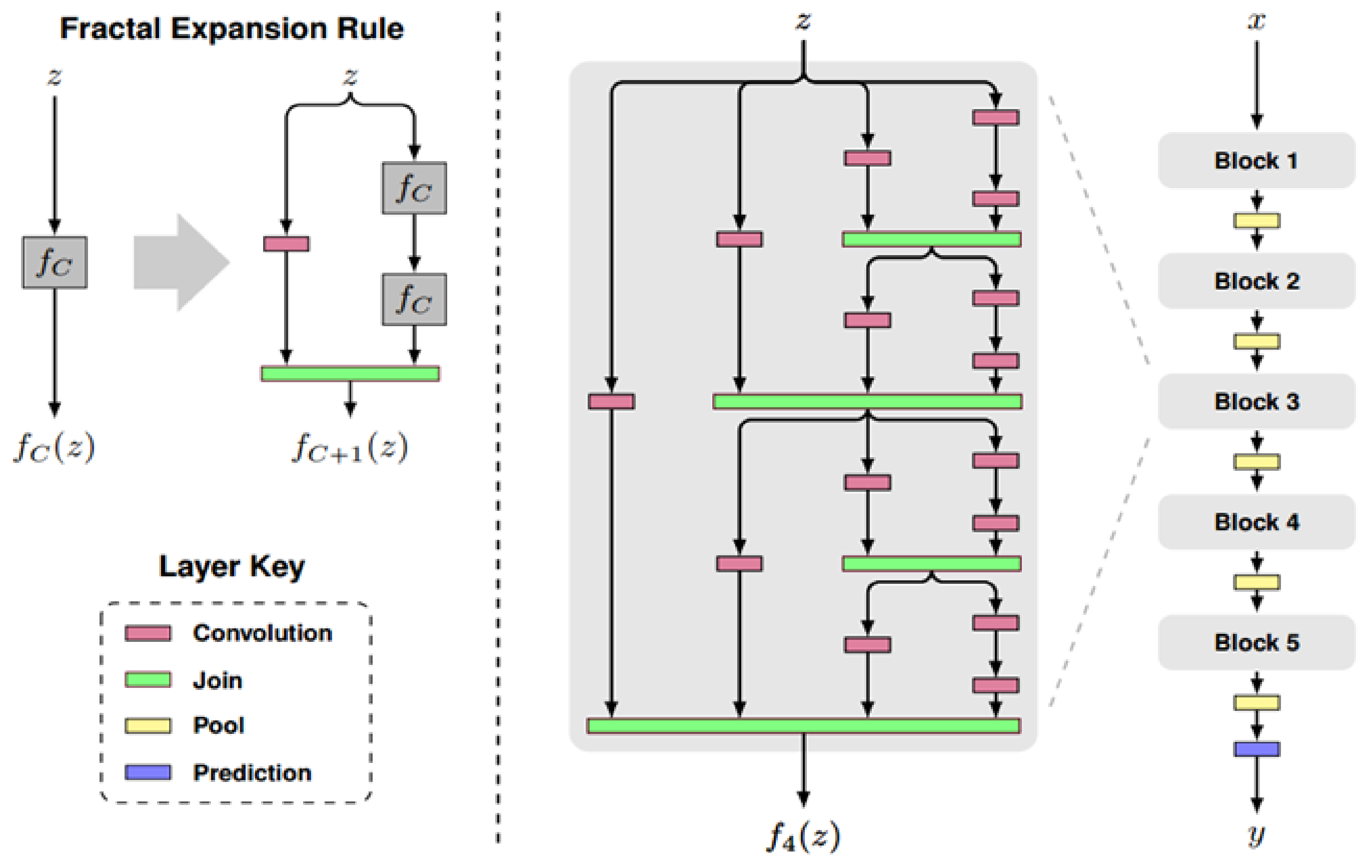
Xingcheng Zhang, Zhizhong Li, Chen Change Loy, Dahua Lin. **PolyNet: A Pursuit of Structural Diversity in Very Deep Networks**

Aside from ResNets, Inception, and Inpcetion-ResNet

FractalNet and DenseNet

FractalNet

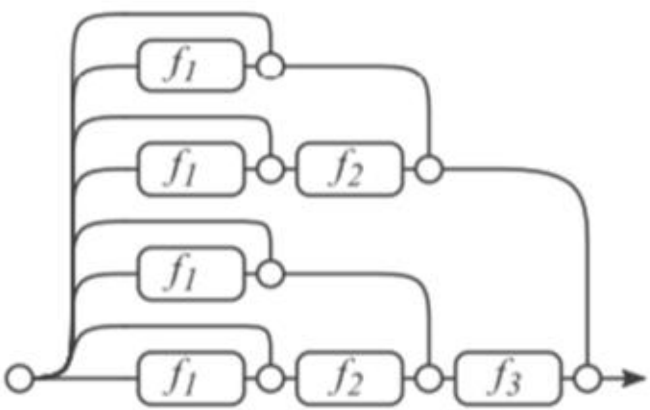
- A extremely deep architecture that does not rely on residuals



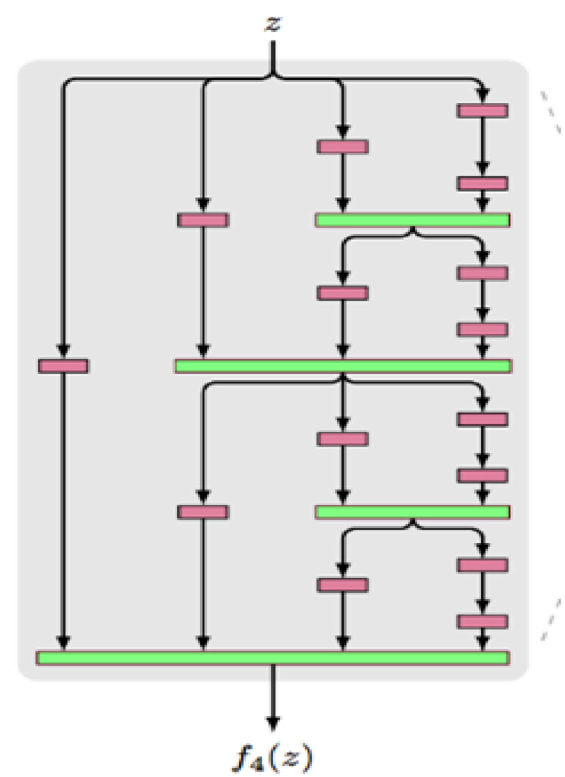
Gustav Larsson, Michael Maire, Gregory Shakhnarovich **FractalNet: Ultra-Deep Neural Networks without Residuals**

FractalNet

- Interestingly, its architecture is similar to an unfolded ResNet



(b) Unraveled view of (a)

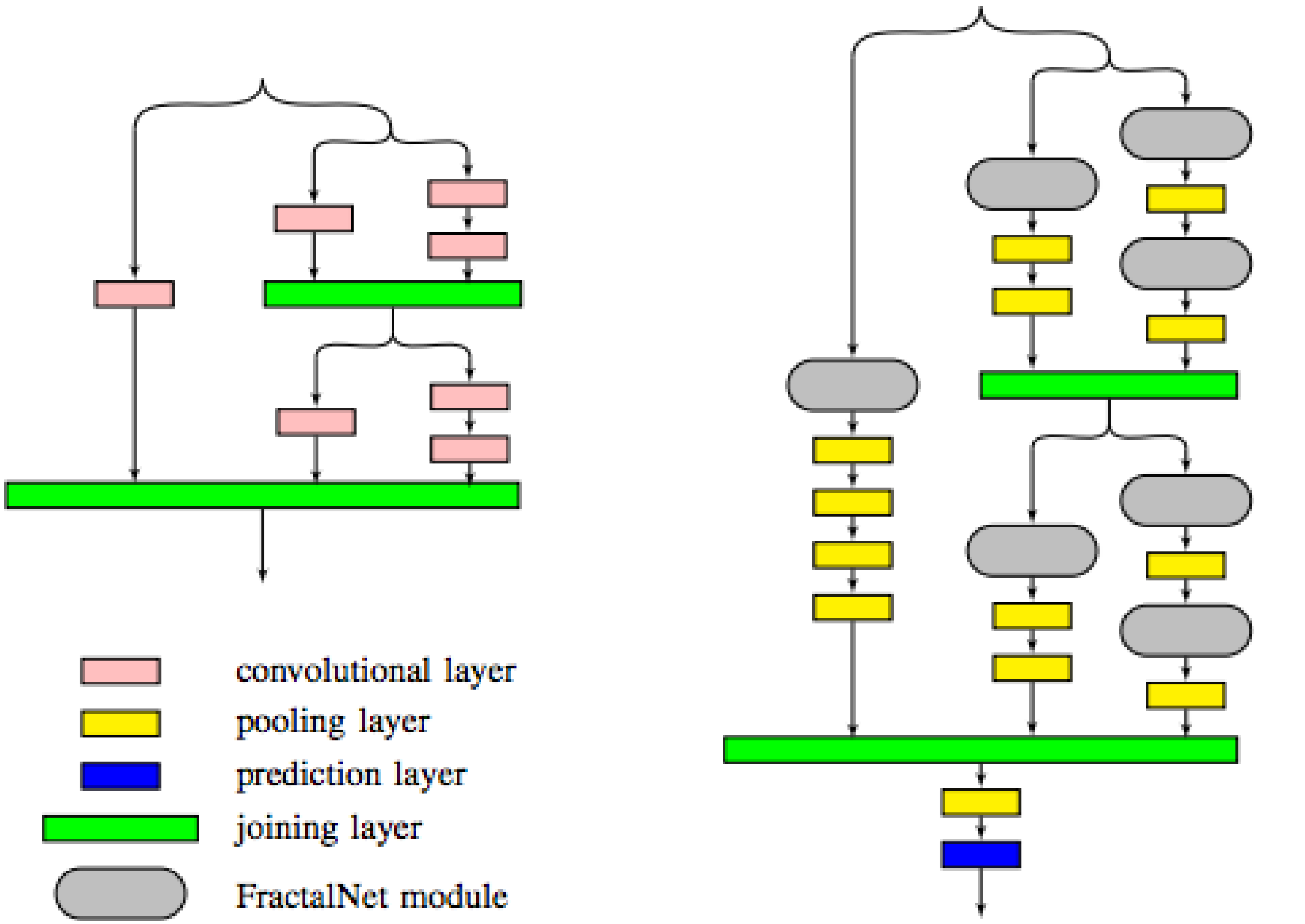


Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, arxiv 2016

Gustav Larsson, Michael Maire, Gregory Shakhnarovich

FractalNet: Ultra-Deep Neural Networks without Residuals

Fractal of Fractals : try going meta



(a) FractalNet module

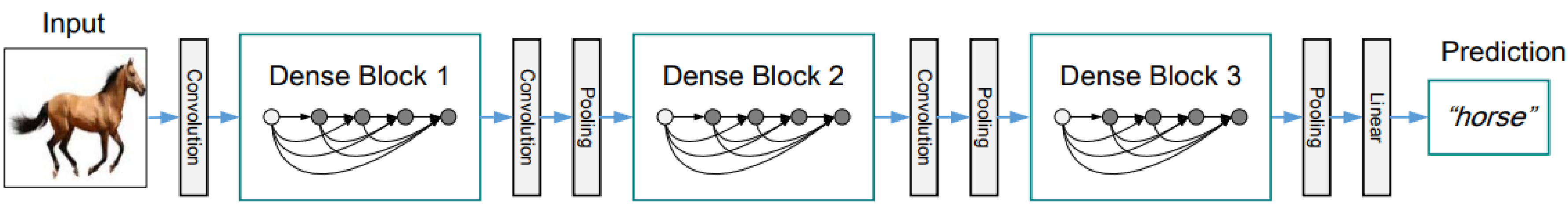
(b) Fractal of FractalNet architecture

Leslie N. Smith, Nicholay Topin, **Deep Convolutional Neural Network Design Patterns**

DenseNet

- Consists with **multiple dense and transition blocks**

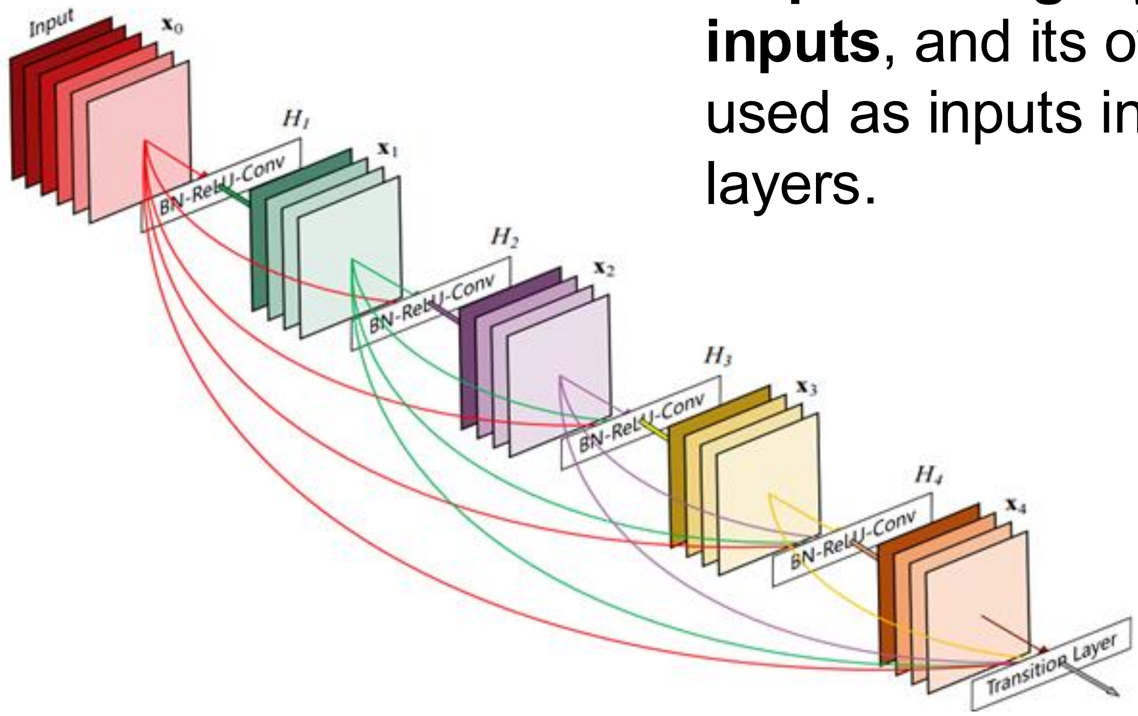
- Basic operations:
  - Convolutions (1x1,3x3)
  - Batch Normalization (BN)
  - ReLU and
  - Pooling.



Overall Network Architecture

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. "Densely connected convolutional networks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700-4708. 2017.

DenseNet (Within a DenseBlock)



- Every layer is connected to all other layers.
- For each layer, the **feature-maps of all preceding layers are used as inputs**, and its own feature-maps are used as inputs into all subsequent layers.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. "Densely connected convolutional networks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700-4708. 2017.

Advantages of DenseNet

- Alleviate the **vanishing-gradient problem**
- Strengthen of **feature propagation**
- Encourage **feature reuse**
- **Substantially** reduce the number of the network parameters.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. "Densely connected convolutional networks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700-4708. 2017.

MobileNets

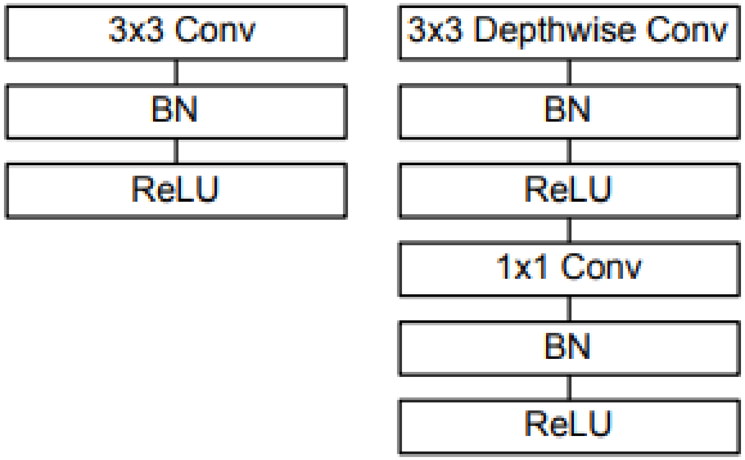
- Introduced two simple global hyperparameters that **efficiently trade off between latency and accuracy**
- The hyper-parameters allow the model builder **to choose the right sized model for their application based on the constraints of the problem**

MobileNet models can be applied to various recognition tasks for efficient on device intelligence

Howard, Andrew G., Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." *arXiv preprint arXiv:1704.04861* (2017).

MobileNets

- The MobileNet model is based on **depth-wise separable convolutions and a pointwise convolution.**
  - **Depth-wise convolution** applies a single filter to each input channel
  - **Point-wise convolution** then applies a 1×1 convolution to combine the outputs the depthwise convolution



Left: Standard convolutional layer with batchnorm and ReLU. Right: Depthwise Separable convolutions with Depthwise and Pointwise layers followed by batchnorm and ReLU.

MobileNets Architecture

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
$5\times$ Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

- Significantly **reduce the network parameters and computational power.**

Table 2. Resource Per Layer Type

Type	Mult-Adds	Parameters
Conv 1×1	94.86%	74.59%
Conv DW 3×3	3.06%	1.06%
Conv 3×3	1.19%	0.02%
Fully Connected	0.18%	24.33%

Bonus Material!

Spatial Transformer Networks

A module to provide spatial transformation capabilities on individual data samples.

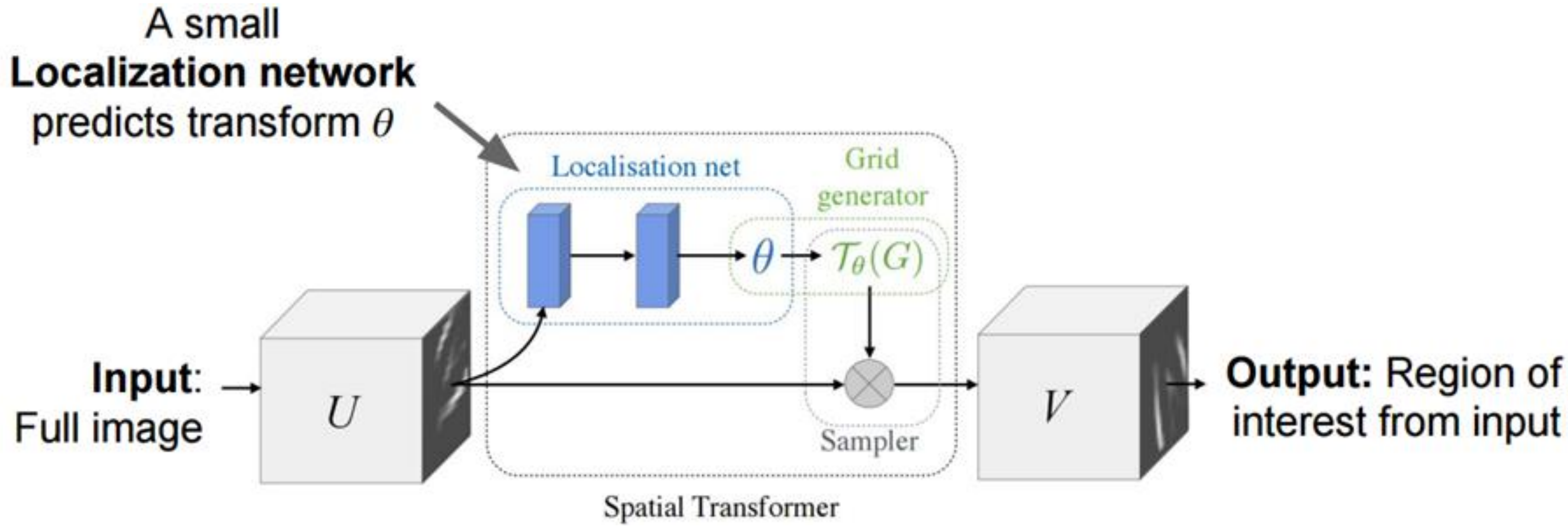
Idea:
Function mapping pixel coordinates of output to pixel coordinates of input.

$$\begin{pmatrix} x_i^s \\ y_i^s \end{pmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix}$$

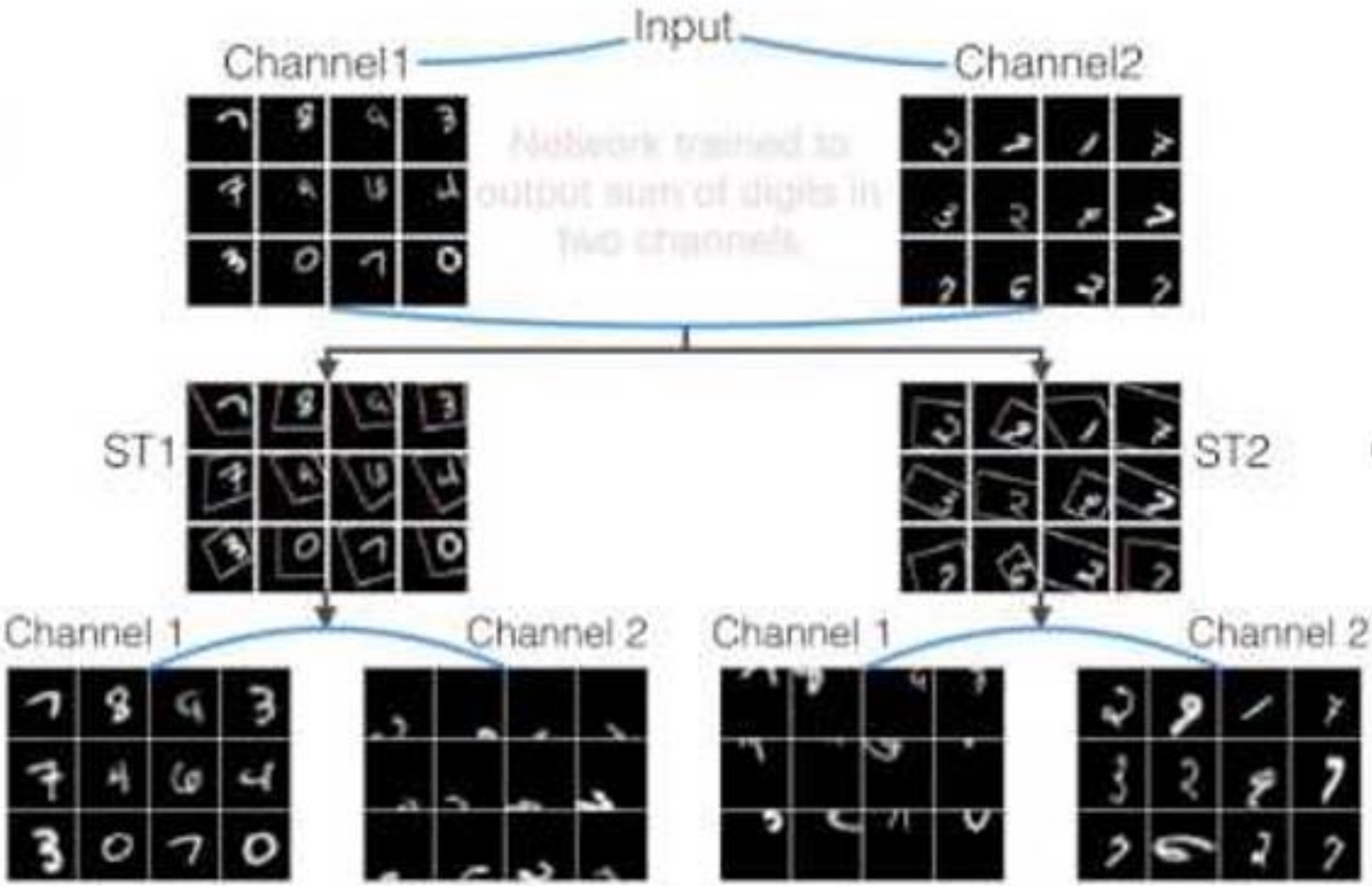
Max Jaderberg, Karen Simonyan, Andrew Zisserman, Koray Kavukcuoglu Spatial Transformer Networks

Spatial transform by how much?

- The localisation network function can take any form, such as a fully-connected network or a convolutional network
- Model includes a final regression layer to produce the transformation parameters θ.



MNIST Addition



Recap (general classification models design principles)

- At surface level, there's **tons of new architectures** that are very different
- Upon closer inspection, **most of them are reapplying well established principles**
- Universal principles seem to be **having shorter subpaths through the networks**
- Use **skip connections and/or create multiple paths** through the network, identity propagation (Residuals, Dense Blocks) seem to make training easier
- **Reduce filter sizes** (except possibly at the lowest layer), factorize filters aggressively
- Use **1x1 convolutions to reduce and expand** the number of feature maps judiciously

References

- Sergey Ioffe,Christian Szegedy,Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, http://jmlr.org/proceedings/papers/v37/ioffe15.pdf
- K. He, X. Zhang, S. Ren, and J. Sun, Deep Residual Learning for Image Recognition, CVPR 2016 https://arxiv.org/abs/1512.03385
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Identity Mappings in Deep Residual Networks https://arxiv.org/abs/1603.05027
- Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, Kaiming He, Aggregated Residual Transformations for Deep Neural Networks, https://arxiv.org/pdf/1611.05431v1.pdf
- Max Jaderberg, Karen Simonyan, Andrew Zisserman, Koray Kavukcuoglu: Spatial Transformer Networks https://arxiv.org/abs/1506.02025
- Leslie N. Smith, Nicholay Topin, Deep Convolutional Neural Network Design Patterns, https://arxiv.org/abs/1611.00847
- Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi: Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning https://arxiv.org/abs/1602.07261
- Gustav Larsson, Michael Maire, Gregory Shakhnarovich, FractalNet: Ultra-Deep Neural Networks without Residuals https://arxiv.org/abs/1605.07648
- Gao Huang, Zhuang Liu, Kilian Q. Weinberger, Laurens van der Maaten: Densely Connected Convolutional Networks https://arxiv.org/pdf/1608.06993v3.pdf
- Rupesh Kumar Srivastava, Klaus Greff, Jürgen Schmidhuber: Highway Networks https://arxiv.org/abs/1505.00387
- Xingcheng Zhang, Zhizhong Li, Chen Change Loy, Dahua Lin, PolyNet: A Pursuit of Structural Diversity in Very Deep Networks, https://arxiv.org/abs/1611.05725
- Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. https://arxiv.org/abs/1412.6806
- Andreas Veit, Michael Wilber, Serge Belongie, Residual Networks Behave Like Ensembles of Relatively Shallow Networks, https://arxiv.org/pdf/1605.06431v2.pdf
- Klaus Greff, Rupesh K. Srivastava & Jürgen Schmidhuber, Highway and Residual Networks Learn Unrolled Iterative Estimation,https://arxiv.org/pdf/1612.07771v1.pdf
- Min Lin, Qiang Chen, Shuicheng Yan, Network In Network, https://arxiv.org/abs/1312.4400
- Brian Chu, Daylen Yang, Ravi Tadinada, Visualizing Residual Networks, https://arxiv.org/abs/1701.02362
- Djork-Arné Clevert, Thomas Unterthiner, Sepp Hochreiter: Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs) https://arxiv.org/abs/1511.07289
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, https://arxiv.org/abs/1502.01852
- Anish Shah, Eashan Kadam, Hena Shah, Sameer Shinde, Sandip Shingade, Deep Residual Networks with Exponential Linear Unit, https://arxiv.org/abs/1604.04112