Improved Residual Networks for Image and Video Recognition

Ionut Cosmin Duta Li Liu Fan Zhu Ling Shao

Inception Institute of Artificial Intelligence (IIAI)

Code and models are publicly available at: https://github.com/iduta/iresnet

Introduction

Problem statements

1. The degradation problem is still an open issue for deep learning (including in ResNets): with the increasing of network depth, optimization/learning difficulties grow as well.
2. Projection shortcuts can play an important role in the network architecture, as they are found on the main information propagation path and can thus directly perturb the signal or cause information loss.
3. In the original ResNet, in the bottleneck building block the only convolution responsible for learning spatial filters receives the least number of input/output channels.

Contributions

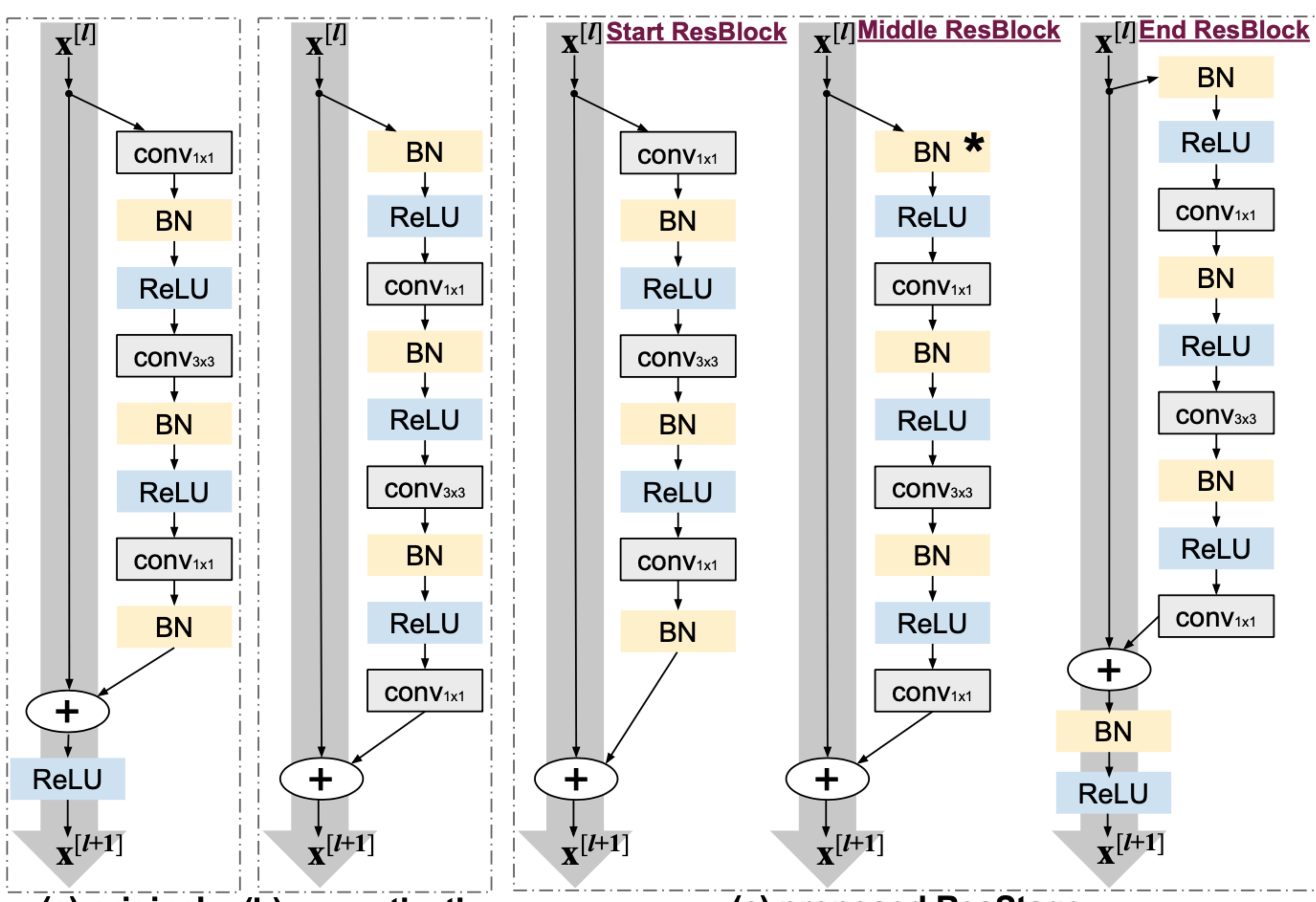
We propose an improved version of residual networks with three main points:

1. We introduce a network architecture for residual learning based on stages.
2. We propose an improved projection shortcut that reduces the information loss.
3. We present a building block that considerably increases the spatial channels for learning more powerful spatial patterns.

Our proposed approach allows us to train extremely deep networks. We successfully train a 404-layer deep CNN on the ImageNet dataset and a 3002-layer network on CIFAR-10 and CIFAR-100 dataset.

Improved Residual Networks

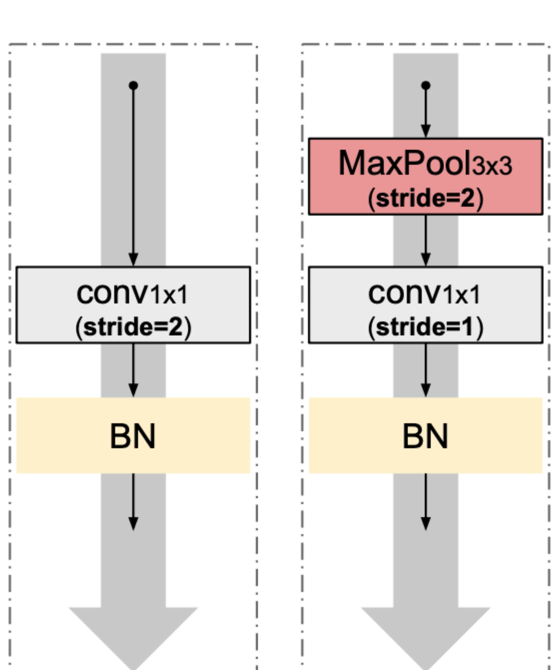
1. Improved information flow through the network



(a) original (b) pre-activation (c) proposed ResStage

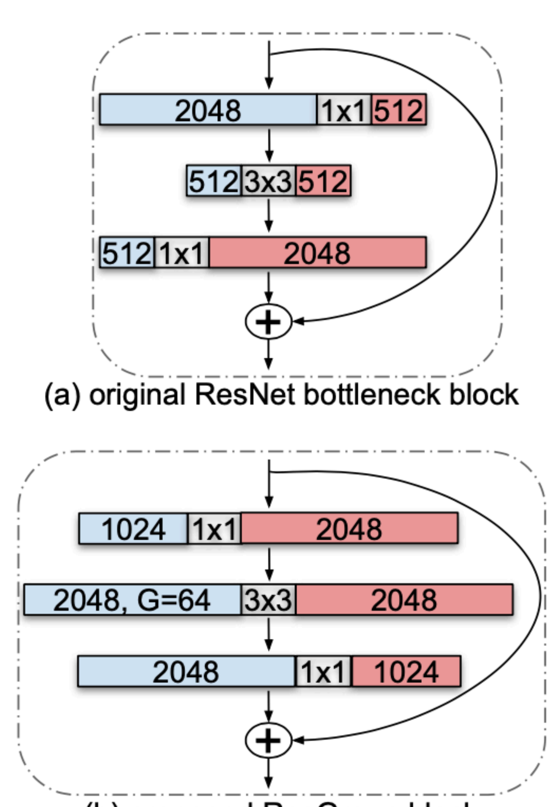
Fig. 1: Residual Building block architectures: (a) original resnet; (b) pre-activation resnet; (c) proposed ResStage. (* the first BN in the first Middle Resblock is eliminated in each stage).

2. Improved projection shortcut



(a) original (b) proposed

3. Grouped building block



(a) original ResNet bottleneck block

(b) proposed ResGroup block

Proposed ResGroup and ResGroupFix architectures:

stage	output	ResNet-50	ResGroupFix-50	ResGroup-50
starting	112×112	7×7, 64, stride 2	7×7, 64, stride 2	7×7, 64, stride 2
	56×56	3×3 max pool, stride2	3×3 max pool, stride2	3×3 max pool, stride2
1	56×56	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 256; 3×3, 256, G=64; 1×1, 128] ×3	[1×1, 256; 3×3, 256, G=8; 1×1, 128] ×3
2	28×28	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 512; 3×3, 512, G=64; 1×1, 256] ×4	[1×1, 512; 3×3, 512, G=16; 1×1, 256] ×4
3	14×14	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 1024; 3×3, 1024, G=64; 1×1, 512] ×6	[1×1, 1024; 3×3, 1024, G=32; 1×1, 512] ×6
4	7×7	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 2048; 3×3, 2048, G=64; 1×1, 1024] ×3	[1×1, 2048; 3×3, 2048, G=64; 1×1, 1024] ×3
ending	1×1	global avg pool 1000-d fc	global avg pool 1000-d fc	global avg pool 1000-d fc
# params		**25.56** × 10^6	**23.37** × 10^6	**24.89** × 10^6
FLOPs		**4.14** × 10^9	**4.30** × 10^9	**5.43** × 10^9

Results

Network	50 layers				101 layers			
	top-1	top-5	params	GFLOPs	top-1	top-5	params	GFLOPs
baseline [6]	23.88	7.06	25.56	4.14	22.00	6.10	44.55	7.88
pre-activation [7]	23.77	7.04	25.56	4.14	22.11	6.26	44.55	7.88
ResStage	23.25	6.81	25.56	4.14	21.75	6.01	44.55	7.88
iResNet	**22.69**	**6.46**	25.56	4.18	**21.36**	**5.63**	44.55	7.92
	152 layers				200 layers			
	top-1	top-5	params	GFLOPs	top-1	top-5	params	GFLOPs
baseline [6]	21.55	5.74	60.19	11.62	22.45	6.39	64.67	15.16
pre-activation [7]	21.41	5.78	60.19	11.62	21.29	5.67	64.67	15.16
ResStage	21.03	5.65	60.19	11.62	20.88	5.57	64.67	15.16
iResNet	**20.66**	**5.43**	60.19	11.65	**20.52**	**5.36**	64.67	15.19

Table 1: Validation error rates (%) comparison results of iResNet on ImageNet.

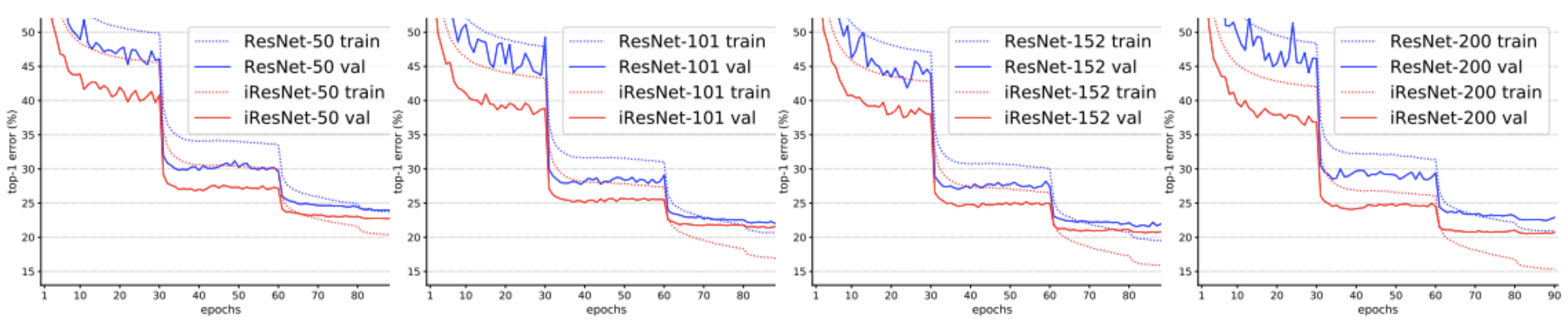
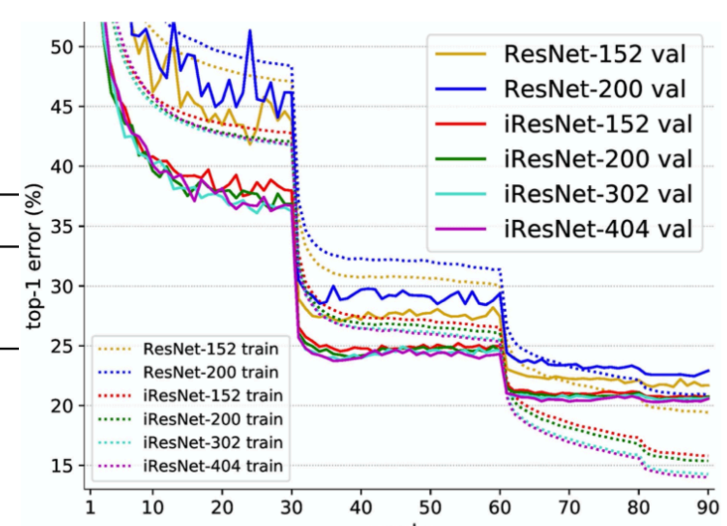


Fig. 2: Training and validation curves on ImageNet for ResNet and iResNet on 50, 101, 152 and 200 layers.

Extreme depths results on ImageNet dataset:

	top-1	top-5	params	GFLOPs
iResNet-302	20.45	5.28	96.59	22.67
iResNet-404	**20.30**	**5.26**	124.5	30.15



Network	50 layers				101 layers				152 layers			
	top-1	top-5	params	GFLOPs	top-1	top-5	params	GFLOPs	top-1	top-5	params	GFLOPs
baseline [6]	23.88	7.06	25.56	4.14	22.00	6.10	44.55	7.88	21.55	5.74	60.19	11.62
ResNeXt [35]	22.44	6.25	25.03	4.30	21.03	5.66	44.18	8.07	20.98	5.48	59.95	11.84
ResGroupFix	21.96	6.15	23.37	4.30	20.94	5.56	43.79	8.33	20.70	5.48	60.61	12.35
ResGroup	21.73	5.94	24.89	5.43	20.98	5.46	47.81	9.94	20.81	5.48	66.99	14.70
iResGroupFix	21.88	5.99	23.37	4.47	20.92	5.54	43.79	8.49	20.75	5.51	60.61	12.53
iResGroup	**21.55**	**5.75**	24.89	5.60	**20.55**	**5.45**	47.81	10.11	**20.34**	**5.20**	66.99	14.87

Table 2: Validation error rates (%) comparison results of ResGroup on ImageNet.

Network	164 layers		1001 layers		2000 layers		3002 layers	
	top-1	P/GFLOPs	top-1	P/GFLOPs	top-1	P/GFLOPs	top-1	P/GFLOPs
CIFAR-10:								
baseline [6]	5.23 (5.54±0.37)	1.70/0.26	7.43	10.33/1.59	fail	20.62/3.17	fail	30.93/4.75
iResNet	**4.80** (5.00±0.14)	1.70/0.26	**4.61**	10.33/1.59	**4.40**	20.62/3.17	**4.95**	30.93/4.75
CIFAR-100:								
baseline [6]	23.86 (24.48±0.39)	1.73/0.26	26.98	10.35/1.59	fail	20.65/3.17	fail	30.96/4.75
iResNet	**22.26** (22.37±0.13)	1.73/0.26	**20.92**	10.35/1.59	**21.12**	20.65/3.17	**21.46**	30.96/4.75

Table 2: Classification error (%) on CIFAR-10/100. For 164 layers train the model five times and show "best(mean±std)". P stands for parameters (in millions).

Method	224×224		320×320†	
	top-1	top-5	top-1	top-5
ResNet-200 [7]	21.7	5.8	20.1	4.8
Inception-v3 [31]	-	-	21.2	5.6
Inception-v4 [29]	-	-	20.0	5.0
Inception-ResNet[29]	-	-	19.9	4.9
DenseNet-264 [11]	22.15	6.12	-	-
Attention-92 [32]	-	-	19.5	4.8
NASNet-A [36]	-	-	17.3	3.8
SENet-154 [10]	18.68	4.47	17.28	3.79
iResNet-200	20.52	5.36	19.36	4.56
iResNet-404	20.30	5.26	19.35	4.61
iResGroup-152	20.34	5.20	19.09	4.59

Table 3: Single-crop error rates (%) comparison with other networks on ImageNet validation set.

Conclusions

- We proposed an improved version of residual networks with **improved learning convergence and recognition performance without increasing the model complexity.**
- Our improvements address all three main components of a ResNet: information propagation through the network, the projection shortcut, and the building block.
- Our proposed approach facilitates learning of extremely deep networks, showing no optimization issues when training networks with over 400 layers (on ImageNet) and over 3000 layers (on CIFAR-10/100).