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**

Problem Statements

- 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.

- Projection shortcuts in ResNets 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.

- 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.

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

Contributions

- ➢We introduce a network architecture for residual learning based on stages

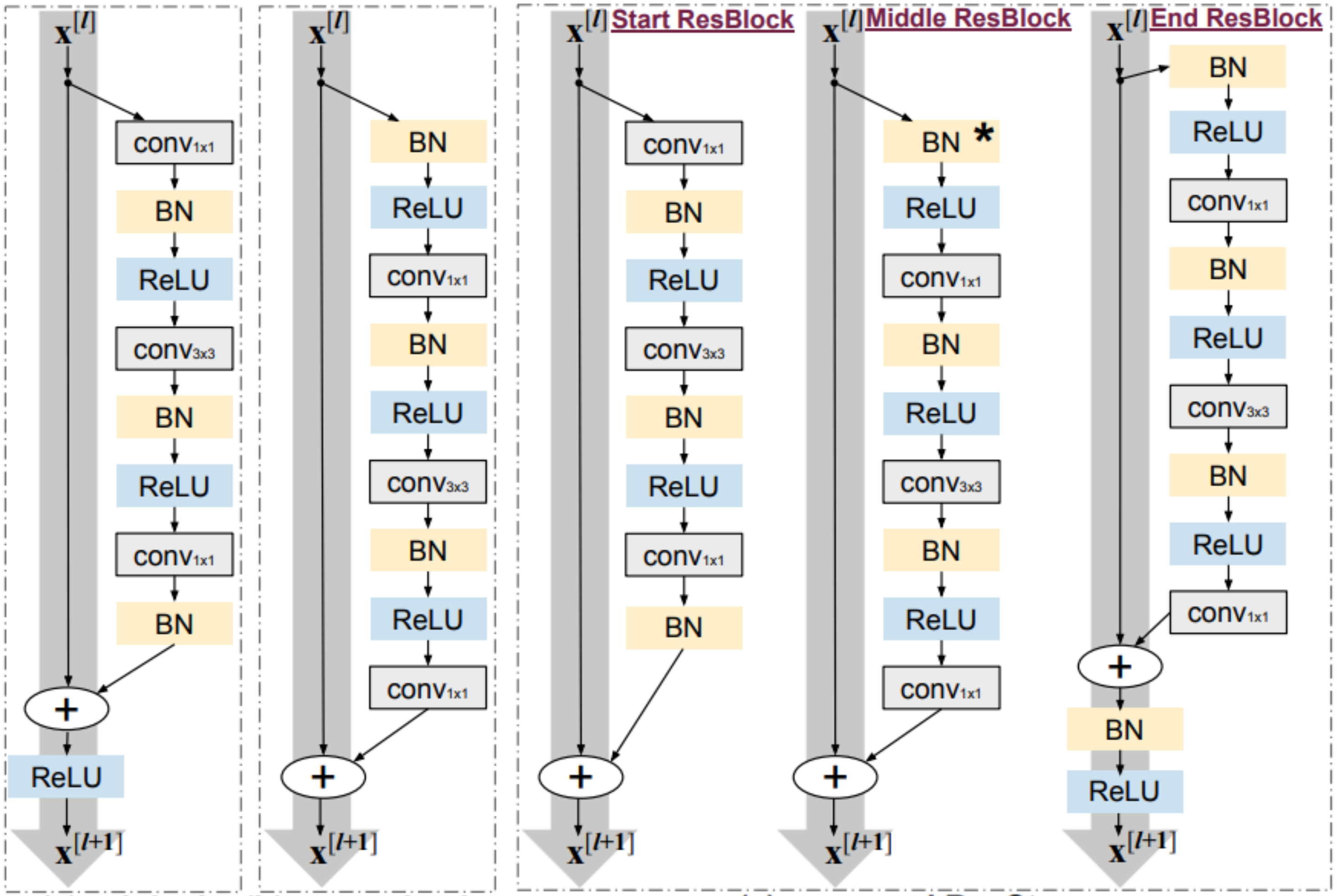
- ➢We propose an improved projection shortcut that reduces the information loss

- ➢We present a building block that considerably increases the spatial channels for learning more powerful spatial patterns

We successfully train a 404-layer deep CNN on the ImageNet dataset and a 3002-layer network on CIFAR-10 and CIFAR-100

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

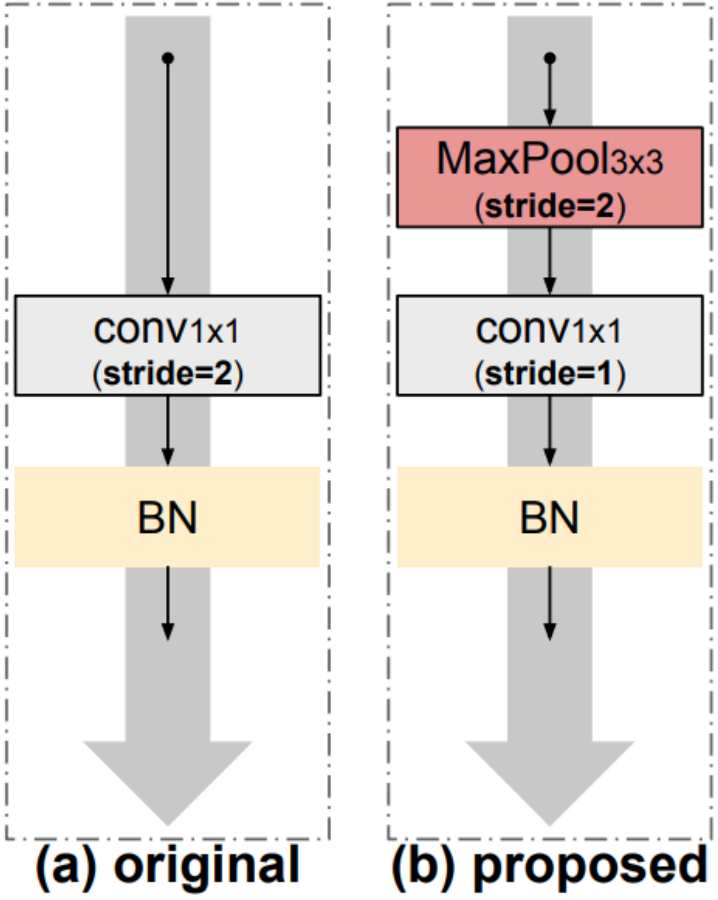
Improved information flow through the network



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

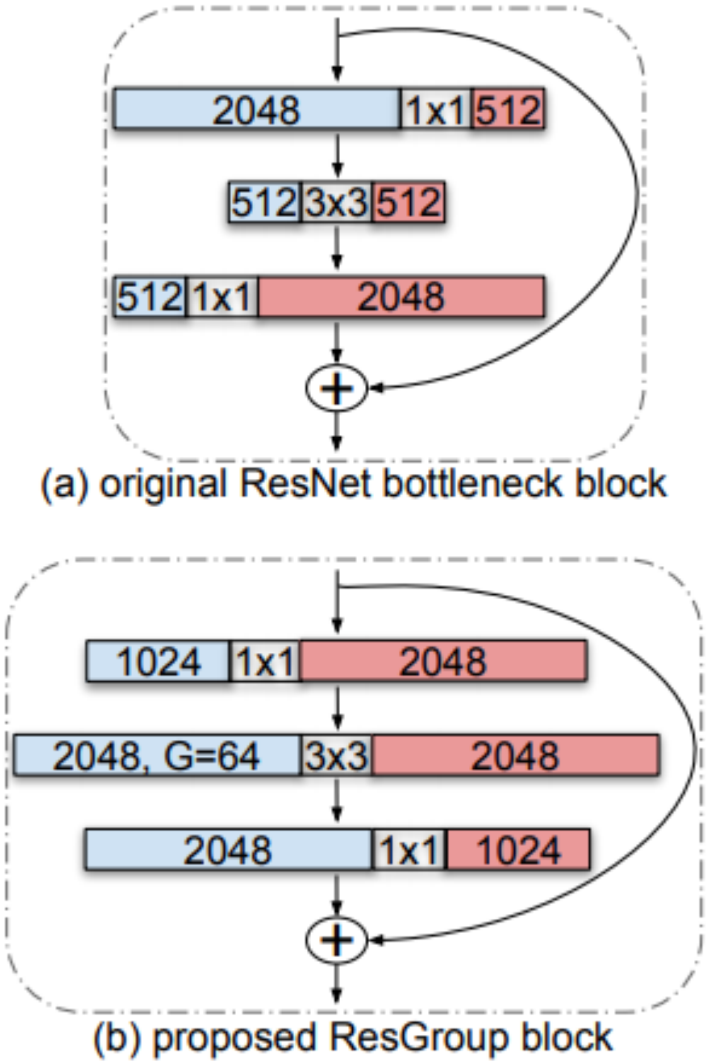
Code and models are publicly available at: https://github.com/iduta/iresnet

Improved projection shortcut



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

Grouped building block



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

Proposed 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	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256, G{=}64 \\ 1\times1, 128 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256, G{=}8 \\ 1\times1, 128 \end{bmatrix} \times 3$
2	28×28	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512, G{=}64 \\ 1\times1, 256 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512, G{=}16 \\ 1\times1, 256 \end{bmatrix} \times 4$
3	14×14	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 1024 \\ 3\times3, 1024, G{=}64 \\ 1\times1, 512 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 1024 \\ 3\times3, 1024, G{=}32 \\ 1\times1, 512 \end{bmatrix} \times 6$
4	7×7	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 2048 \\ 3\times3, 2048, G{=}64 \\ 1\times1, 1024 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 2048 \\ 3\times3, 2048, G{=}64 \\ 1\times1, 1024 \end{bmatrix} \times 3$
ending	1×1	global avg pool 1000-d fc	global avg pool 1000-d fc	global avg pool 1000-d fc
# params		**25.56** $\times 10^6$	**23.37** $\times 10^6$	**24.89** $\times 10^6$
FLOPs		**4.14** $\times 10^9$	**4.30** $\times 10^9$	**5.43** $\times 10^9$

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

Results

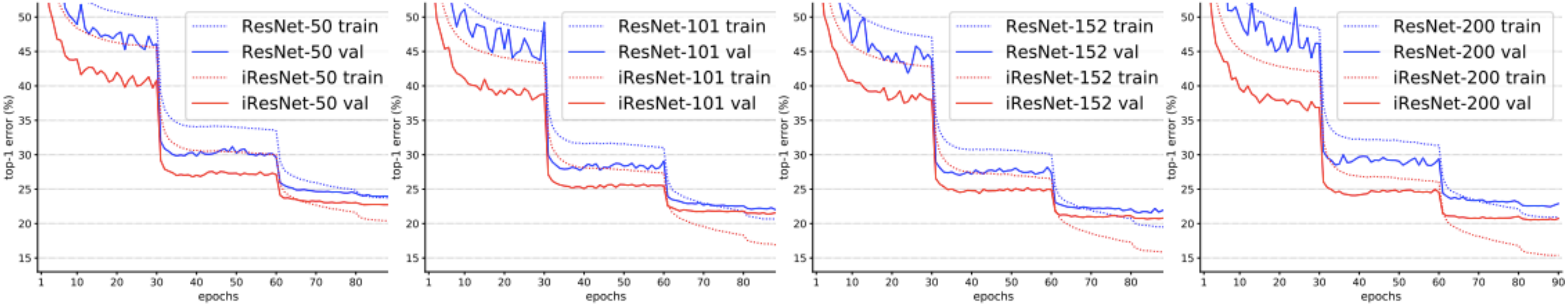
Validation error rates (%) comparison results of iResNet on ImageNet

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

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

Results

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



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

Results

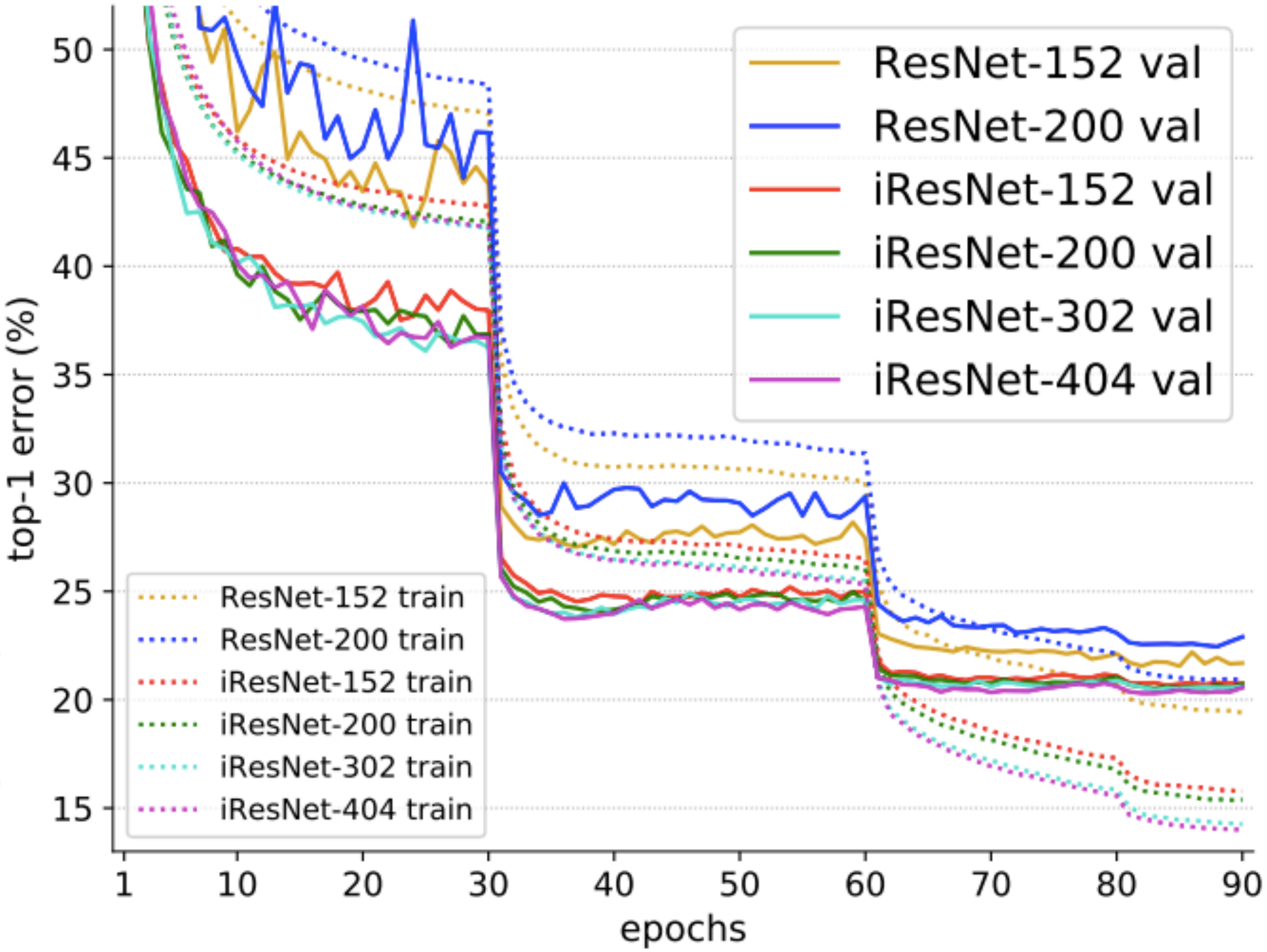
Error rates results of iResNet on ImageNet with extreme depth: 302 and 404 layers.
P stands for parameters.

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

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

Results

Extreme depths



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

Results

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).

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

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

Results

Video recognition error rates (%), parameters are in millions.

Network	Kinetics-400				Something-Something-v2			
	top-1	top-5	params	GFLOPs	top-1	top-5	params	GFLOPs
baseline3D-50 [6]	37.01	15.41	47.00	93.26	46.50	19.02	46.54	93.26
iResNet3D-50	**33.91**	**13.36**	47.00	93.93	**45.56**	**17.73**	46.54	93.93

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

Results

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

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

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

Results

Proposed backbones on SSD object detector with 300×300 input image size (results on COCO val2017).

Backbone	Avg. Precision, IoU:			Avg. Precision, Area:			Avg. Recall, #Dets:			Avg. Recall, Area:			params	GFLOPs
	0.5:0.95	0.5	0.75	S	M	L	1	10	100	S	M	L		
ResNet-50 [6]	26.20	43.97	26.96	8.12	28.22	42.64	24.50	35.41	37.07	12.61	40.76	57.25	22.89	20.92
iResNet-50	27.74	45.85	28.51	8.52	30.07	44.62	25.29	36.90	38.51	13.28	42.79	58.57	22.89	20.99
iResGroupFix-50	28.90	47.44	29.99	9.70	31.49	45.83	25.97	37.84	39.52	14.63	44.17	59.52	21.13	18.61
iResGroup-50	**29.56**	**48.38**	**30.87**	**10.33**	**32.52**	**46.62**	**26.40**	**38.49**	**40.24**	**15.10**	**44.82**	**60.20**	22.66	21.62
ResNet-101 [6]	29.58	47.69	30.80	9.38	31.96	47.64	26.47	38.00	39.64	14.09	43.54	61.03	41.89	48.45
iResNet-101	30.92	49.50	32.29	10.05	34.27	49.13	27.15	39.34	41.08	15.21	45.93	61.90	41.89	48.49
iResGroupFix-101	31.64	50.70	33.28	11.21	34.91	50.20	27.94	40.41	42.22	16.84	46.99	63.64	41.55	48.25
iResGroup-101	**32.81**	**51.78**	**34.55**	**11.81**	**36.56**	**51.72**	**28.37**	**41.43**	**43.22**	**17.20**	**48.54**	**64.08**	45.58	54.87

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

Results

Single-crop error rates (%) comparison with other networks on ImageNet validation set.
† some approaches use larger image crops than 320×320, Inception family uses 299×299.

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

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

Conclusion

- 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 training 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).

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

Thank you!

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