# Deep Residual Learning for Image Recognition

Summary written by Mahban Gholijafari
March 8, 2025

**Summary:** This paper addresses the problem of decreasing accuracy with increasing depth of the network by presenting a residual learning framework to ease the training of networks that are substantially deeper than those used previously. They tested their framework on networks as deep as 152 layers and achieved 3.57% error on ImageNet test set.

**Related work:** Recent work showed the importance of network depth for the accuracy of results on the challenging ImageNet dataset[3, 4, 10, 11] with depth of 16 to 30. With deeper depth, the problem of vanishing/exploding gradients happened and they have been largely addressed by normalized initialization [6, 2, 9, 3] and intermediate normalization layers [4] and allowed deeper networks. With deeper networks started to converge, a degradation problem has been exposed where accuracy gets saturated and then degrades rapidly after adding certain layers and it is happening because of overfitting. Inspired by VLAD and Fisher Vectors [5, 7] in image recognition, and Multigrid method for solving Partial Differential Equations (PDE) in low-level vision and computer graphics, they reformulated and preconditioned the residual nature of these methods for simplifying and optimization of networks. Also, practices and theories that lead to shortcut connections have been studied for a long time [1, 8, 12].

**Approach:** Let $\mathcal{H}(\mathrm{x})$ be an underlying mapping to be fit by a few stacked layers with x denoting the inputs to the first of these layers. They let these layers approximate a residual function $\mathcal{F}(\mathrm{x}) := \mathcal{H}(\mathrm{x}) - \mathrm{x}$ and the original function becomes $\mathcal{F}(\mathrm{x}) + \mathrm{x}$. Their approach was to adopt residual learning to every few stacked layers and they considered a building block defined as $y = \mathcal{F}(\mathrm{x}, \{W_i\}) + \mathrm{x}$ where x and y are the input vectors of the layers considered. The function $\mathcal{F}(\mathrm{x}, \{W_i\})$ represents the residual mapping to be learned and the dimensions of x and $\mathcal{F}$ must be equal and the shortcut connection is performed by element-wise addition that introduces neither extra parameter nor computation complexity. If the dimensions do not match, they propose to perform a linear projection $W_s$ by the shortcut connections to match the dimensions $y = \mathcal{F}(\mathrm{x}, \{W_i\}) + W_s\mathrm{x}$. In this work, they used a residual function $\mathcal{F}$ that has two or three layers and more layers are also possible.

*Datasets, Experiments and Results:* This paper presents a plain network as a baseline that is inspired by VGG nets but has fewer filters and lower complexity. The convolution layers mostly have 3x3 filters, they performed downsampling directly by convolutional layers that have a stride of 2, and the network ends with a global average pooling layer and a 1000-way fc layer with softmax. Based on this plane network, they insert shortcut connections and the identity shortcuts can be directly used when the input and output are of the same dimensions and if they are not, they consider using zero padding or projection shortcut. Their implementation is for ImageNet 2012 classification dataset that consists of 1000 classes, a 224x224 crop is randomly sampled from an image or its horizontal flip, with the per-pixel mean subtracted, BN is adopted, standard color augmentation, and SGD with mini-batch size of 256 was used. Next, they evaluate 18-layer and 34-layer residual nets (ResNets), the baseline architectures are the same. The experiments showed that 34-layer ResNet has a better performance than 18-layer ResNet and exhibits considerably lower training error and is generalizable to validation data. ResNet reduces the top-1 error by 3.5%. This work shows projection shortcuts are better than plain counterpart. Next, modification to the building block as a bottleneck design is introduced and resulted in 50/101/152-layer ResNets and they improved the accuracy by considerable margins. This study also studied extremely deep networks on the CIFAR-10 dataset trained with a minimum batch size of 128 on two GPUs. This dataset has 50k training images and 10k testing images in 10 classes. They compared 20, 32, 44 and 56-layer networks and they overcome the optimization difficulty and demonstrate accuracy gains when the depth increases. In this case, using 110-later ResNet with initial learning eate of 0.1 is slightly too large to start converging. This method has a good generalization performance on other recognition tasks (PASCAL VOC 2007 and 2012 and COCO)

**Strengths:** 34-layer ResNets have achieved very competitive accuracy and 152-layer ResNet has a single-model top-5 validation error f 4.49%. This single model result outperforms all previous ensemble results.

**Weaknesses:** There are still open problems on over 1000 layers networks. The testing results of 1202-layer network is worse than that of their 110-layer network, although both have similar training error, which they think is because of overfitting.

**Reflections:** Future work may be applying maxout/dropout to these architecture designs to improve results.

## References

[1] C. M. Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.

[2] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[3] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[4] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

[5] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE transactions on pattern analysis and machine intelligence*, 34(9):1704–1716, 2011.

[6] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–50. Springer, 2002.

[7] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007.

[8] B. D. Ripley. *Pattern recognition and neural networks*. Cambridge university press, 2007.

[9] A. M. Saxe, J. L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

[10] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[12] W. N. Venables and B. D. Ripley. *Modern applied statistics with S-PLUS*. Springer Science & Business Media, 2013.