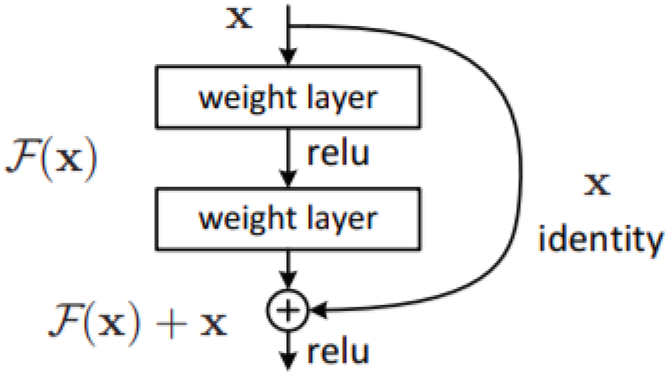
Deep Residual Learning for Image Recognition (ResNet) Summary

9 November 2023

The ResNet paper, titled "Deep Residual Learning for Image Recognition," was authored by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. The authors present a residual learning framework for training deeper neural networks, focusing on learning residual functions with reference to layer inputs.They show that these networks are easier to optimize and can gain accuracy from increased depth. They evaluated the residual nets with 152 layers on the ImageNet dataset, achieving 3.57% error and winning first place on the ILSVRC 2015 classification task.

Introduction

- Deep convolutional neural networks have revolutionized image classification by integrating low/mid/high-level features and classifiers in an end-to-end multilayer fashion.
- Very deep networks have the problem of vanishing/exploding gradients but this has been largely addressed by normalized initialization and intermediate normalization layers.
- A degradation problem has been exposed which occurs in deeper networks which is different from error due to over-fitting and adding more layers doesn't help.
- This paper addresses the degradation problem by introducing a deep residual learning framework.
- Instead of hoping each few stacked layers directly fit a desired underlying mapping, the layers are explicitly fit to a residual mapping


x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Let the desired underlying mapping be H(x), we let the stacked non-linear layers fit another mapping of F(x) = H(x) - x. The original mapping is recast into F(x) + x.

- It is hypothesized that it is easier to optimize the residual mapping than to optimize original, unreferenced mapping.
- The formulation F(x) + x can be realized in a network by feed-forward networks with shortcut connections (identity mapping) and adding to outputs of stacked layers. It doesn't add computational complexity, nor parameters of the network.
- The original ResNet is trained using end-to-end SGD with backprop.
- Easy to optimize without introducing degradation issue unlike other (plain) networks.

Residual Representations

- Vector of Locally Aggregated Descriptors (VLAD) and Fischer Vector (probabilistic representation of VLAD) encodes residual vectors and are powerful shallow representations for image retrieval and classification.
- Encoding residual vectors is more effective than encoding original vectors.
- Solvers that rely on residual vectors converge much faster than standard solvers that are unaware of the residual nature of solutions which suggests that good reformulation or preconditioning can simplify the optimization.

Shortcut Connections

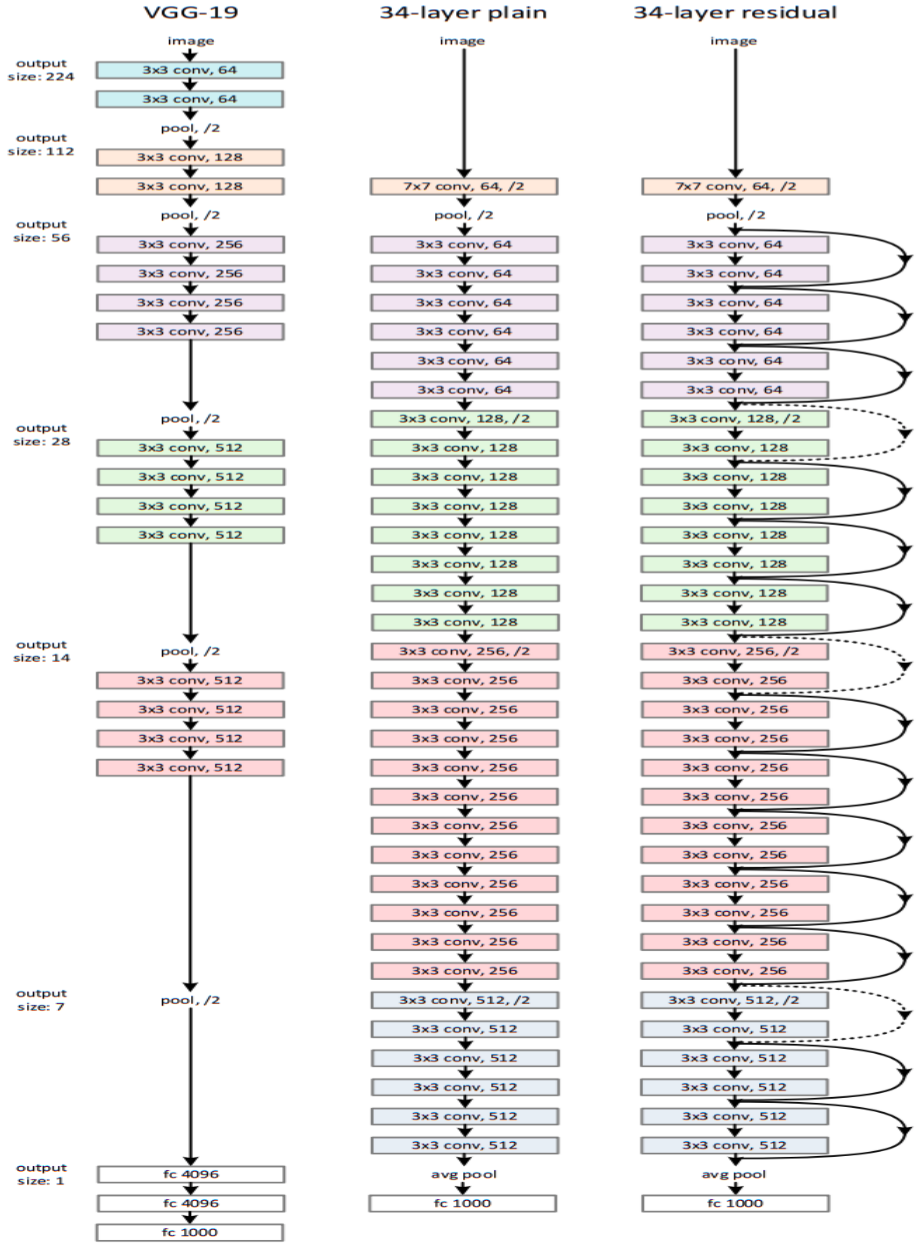
- Connections from intermediate layers are directly connected to auxiliary classifiers for addressing vanishing/exploding gradients.
- The “inception” paper showed the use of shortcut branches and a few deeper branches to create an “inception” layer.
- “Highway networks” present shortcut connections with gating functions but those have parameters whereas resnet’s shortcuts are parameter-free. Also, When “closed” (approaching zero) the shortcut, the layers in highway networks represented non-residual functions, whereas ResNet shortcuts (identity) are never closed, and all information is always passed through, with additional residual functions being learned.
- Resnet have accuracy gains unlike in highway networks upon increasing depths.

Residual Learning

- Instead of learning the original underlying mapping H(x), the layers approximate a residual function F(x) = H(x) – x.Though both forms can be approximated by non linear layers, the ease of learning might be different.
- While adding layers, in doing so, the error must not exceed the shallower counterpart because if the added layer could learn identity mapping, there should be no net effect of adding the layer in the actual learning of model. The existence of “degradation problem” suggests that solvers have difficulty in approximating the non-linear layers to identity mapping.
- In residual mapping, if the optimal mapping is identity, then the weights are driven towards zero, so that the output (with shortcut connection) simply approaches identity mapping. In reality, such identity mappings are not optimal and there is space for the network to learn. What residual network thus gives is a precondition to learn more optimal function. This comes from the statement: “If the optimal function is closer to an identity mapping than to a zero mapping, it should be easier for the solver to find perturbations with reference to an identity mapping, than to learn the function as a new one.”
- Experiments show that such identity mapping provide reasonable preconditioning.

Network Architectures

- The (plain) baseline models are inspired by VGG nets.
- The convolutional layers have 3x3 filters.
- For the same output feature map size, the layers have the same numbers of filters. If the feature map size is halved, the number filters is doubled. This preserves the time complexity per layer.
- Down-sampling directly by convolutional layers with stride of 2.
- Global average pooling layer and 1000-way FC layer with softmax at the end of layer.
- Total 34 weighted layers (as shown in middle of the above figure).
- Lower complexity and only 18% FLOPS than VGG-19.
- Based on the plain network, connections are inserted. (connection of same dimension shown in solid and otherwise in dotted arrows.) While dimensions are added, either extra zeros are padded or a matrix method is used to increase dimension as said earlier.



Implementation for ImageNet:

- Resized with its shorter side sampled in [256,480] for scale augmentation.
- 224x224 crop is sampled from the image or its horizontal flip.
- Per-pixel mean is subtracted.
- Standard color augmentation is used.
- Batch normalization right after each convolutional layer and before activation.
- Trained from scratch, using SGD with a mini-batch size of 256.
- Learning rate starting from 0.1, divided by 10 as error tends saturation.
- 600000 iterations, weight decay of 0.0001, momentum of 0.9 and dropout not used.
- Testing is done by averaging the scores in different scales such that the shorter side is in {224, 256, 384, 480, 640}.

Identity vs Projection Shortcuts

- The study investigates the use of parameter-free, identity shortcuts in ImageNet training and projection shortcuts.
- Three options are compared: zero-padding shortcuts, projection shortcuts, and all shortcuts being projections.
- All three options are significantly better than the plain counterpart.
- However, projection shortcuts are not essential for addressing the degradation problem, and they are not used in the rest of the paper to reduce memory/time complexity and model sizes.
- Identity shortcuts are crucial for minimizing complexity in bottleneck architectures.

Deeper Bottleneck architectures

The text describes deeper nets for ImageNet, which are modified as a bottleneck design due to concerns on training time. The design includes 3 layers for each residual function, with 1×1 layers reducing and increasing dimensions, and 3×3 layers as a bottleneck with smaller input/output dimensions.

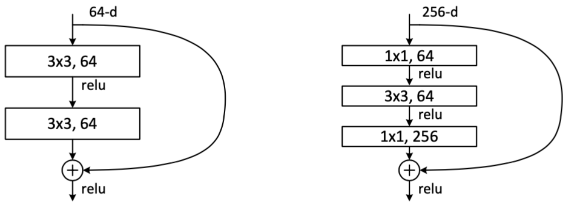


Figure 5. A deeper residual function $\mathcal{F}$ for ImageNet. Left: a building block (on 56×56 feature maps) as in Fig. 3 for ResNet-34. Right: a "bottleneck" building block for ResNet-50/101/152.