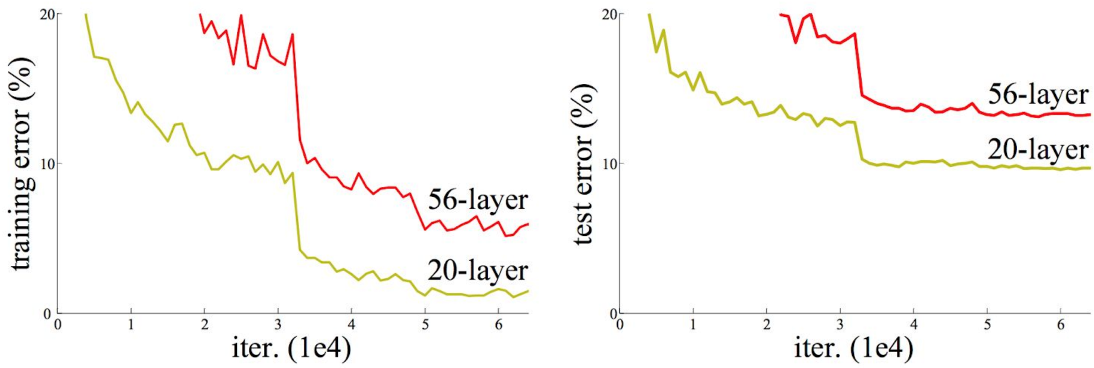
Residual Networks

Problems With Other Networks

Adding more layers to the networks leads to a degradation problem i.e it leads to the model's accuracy saturating, then rapidly decaying, and higher training errors. They don't overfit but nonetheless perform worse than shallower networks on both training and test data due to being more difficult to optimize.
Much of the success of Deep Neural Networks has been achieved by adding layers. The intuition behind their function is that these layers progressively learn more complex features. The first layer learns edges, the second layer learns shapes, the third layer learns objects, the fourth layer learns eyes, and so on. But the author shows that there is a maximum threshold for depth with the traditional CNN model.



Clearly this is not overfitting as training and test error both are higher than shallow networks.

The other problem of adding more layers is obviously Vanishing and Exploding gradients.But it can be addressed by Normalised initialization and intermediate normalization(Batch Norm).

The degradation (of training accuracy) indicates that not all systems are similarly easy to optimize. Let us consider a shallower architecture and its deeper counterpart that adds more layers onto it. There exists a solution by construction to the deeper model: the added layers are identity mapping, and the other layers are copied from the learned shallower model. The existence of this constructed solution indicates that a deeper model should produce no higher training error than its shallower counterpart. But experiments show that our current solvers on hand are unable to find solutions that are comparably good or better than the constructed solution i.e optimizers have problems learning identity(or nearly identity) mappings.

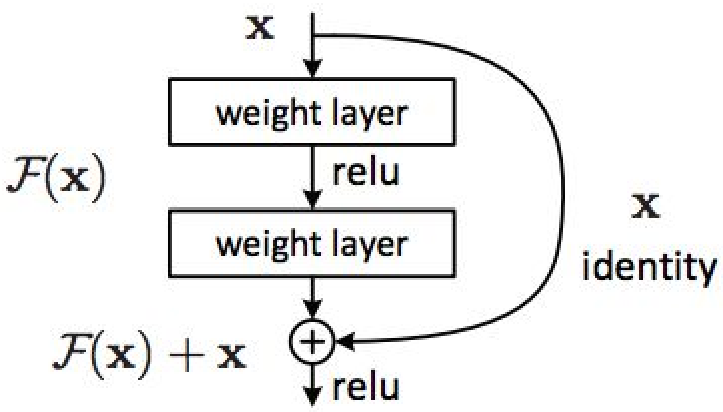
Solution

This degradation problem can be addressed by Deep Residual learning framework.

If the desired mapping is H(x), let the layers learn F(x) = H(x) - x and add x back through a shortcut connection H(x) = F(x) + x. An identity mapping can then be learned easily by driving the learned mapping F(x) to 0.

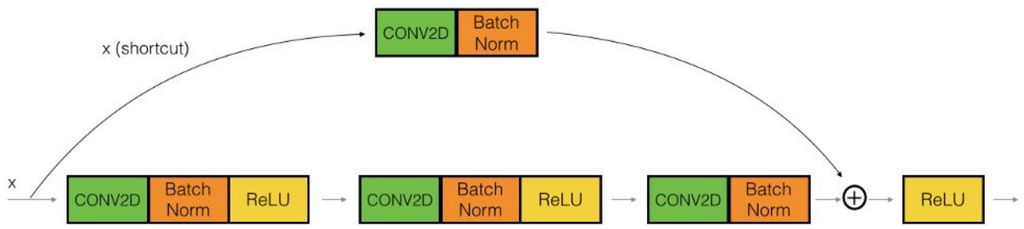
The approach is to add a shortcut or a skip connection that allows info to flow, i.e. you bypass data along with normal CNN flow from one layer to the next layer after the immediate next.

Residual Block:



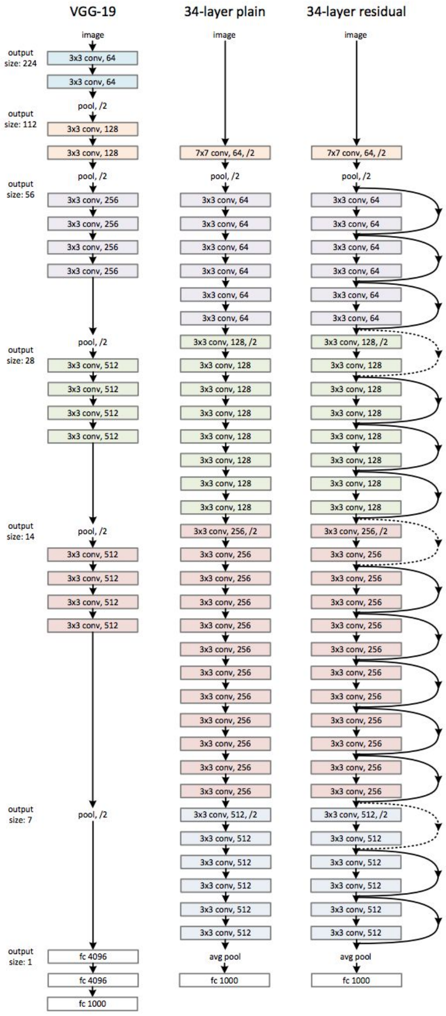
- Identity shortcut connections add neither extra parameter nor computational complexity. The entire network can still be trained end-to-end by SGD with backpropagation, and can be easily implemented using common libraries.

- If the dimensions of x and F(x) are not equal e.g when changing the input/output channels, We can do size transformation by zero-padding or projections. Projections introduce additional parameters. Authors found that projections perform slightly better, but are "not worth" the large number of extra parameters.
- The identity shortcut can be directly used when the input and output are of the same dimension(dimensions of F(x) and x are same).
- Bottleneck design is used to further reduce computational complexity, i.e. 1x1 convolutional layers to change the dimensions .This skip connection can be represented as:

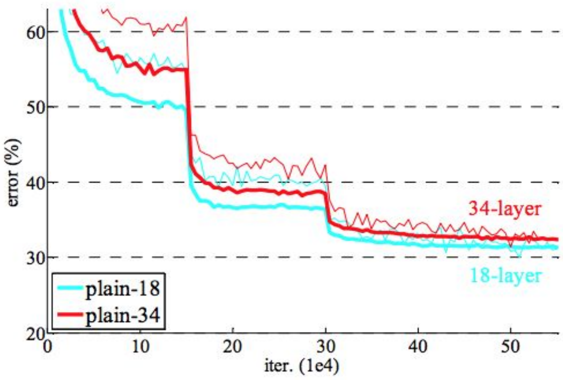


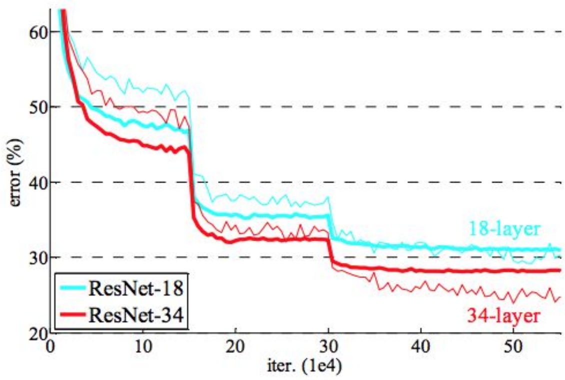
- The 1x1 Conv block helps to modify the incoming data and restructure so that the output of the first layer matches the dimensions of the third layer and can be added together.
- They use batch normalization before each nonlinearity.
- Optimizer is SGD.
- They don't use dropout.

Network Architecture



Results





- They also test on very deep residual networks with 50 to 152 layers.
- In further tests on CIFAR-10 they can observe that the activations of the convolutions in residual networks are lower than in plain networks.
- So the residual networks default to doing nothing and only change (activate) when something needs to be changed.
- They also test on COCO and get significantly better results than a Faster-R-CNN+VGG implementation.