Problem with deep neural networks - Deep networks are hard to train because of the notorious vanishing gradient problem — as the gradient is back-propagated to earlier layers, repeated multiplication may make the gradient infinitely small. When deeper networks are able to start converging, a degradation problem has been exposed: with the network depth increasing, accuracy gets saturated and then degrades rapidly. Unexpectedly, such degradation is not caused by overfitting, and adding more layers to a suitably deep model leads to higher training error.

Proposed solution in ResNet - The core idea of ResNet is introducing a so-called “identity shortcut connection” that skips one or more layers.The paper present a residual learning framework to ease the training of networks that are substantially deeper than those used previously.The paper explicitly reformulate the layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions. It provide comprehensive empirical evidence showing that these residual networks are easier to optimize, and can gain accuracy from considerably increased depth.Instead of hoping each few stacked layers directly fit a desired underlying mapping, it explicitly let these layers fit a residual mapping. Formally, denoting the desired underlying mapping as H(x), we let the stacked nonlinear layers fit another mapping of F(x) := H(x)−x. The original mapping is recast into F(x)+x. The formulation of F(x) +x can be realized by feedforward neural networks with “shortcut connections” . Shortcut connections are those skipping one or more layers. In our case, the shortcut connections simply perform identity mapping, and their outputs are added to the outputs of the stacked layers . Identity shortcut connections add neither extra parameter nor computational complexity. The entire network can still be trained end-to-end by SGD with backpropagation.

Architecture of Residual Network -To a plain network, we insert shortcut connections which turn the network into its counterpart residual version.The identity shortcuts can be directly used when the input and output are of the same dimensions.When the dimensions increase, we consider two options: (A) The shortcut still performs identity mapping, with extra zero entries padded for increasing dimensions. This option introduces no extra parameter; (B) The projection shortcut is used to match dimensions (done by 1×1 convolutions). For both options, when the shortcuts go across feature maps of two sizes, they are performed with a stride of 2.

Bottleneck blocks - Because of concerns on the training time that we can afford, we modify the building block as a bottleneck . For each residual function F, we use a stack of 3 layers instead of 2 . The three layers are 1×1, 3×3, and 1×1 convolutions, where the 1×1 layers are responsible for reducing and then increasing dimensions, leaving the 3×3 layer a bottleneck with smaller input/output dimensions.

Results -On ImageNet dataset Resnet-34 A gives 25.03% error,Resnet-34 B gives 24.52% error,Resnet-34 A gives 25.03% error,Resnet-50 gives 22.85% error,Resnet-101 gives 21.75% error and Resnet-152 gives 21.43 % error in classification tasks.On CIFAR10 dataset ResNet with 20 layers give 8.75% error,with 32 layers give 7.51% error,with 44 layers give 7.17% error,with 56 layers give 6.97% error,with 110 layers give 6.43% error,with 1202 layers give 7.93% error.

Github Repo for model- <https://github.com/rishabhd786/ResNet_CIFAR10.git>