Deep Residual Learning for Image Recognition

Link

arxiv

Summary

- 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**(not just validation error). This suggests 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.
- Instead of hoping each few stacked layers directly fit a desired underlying mapping, we 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 $F(x) + x$. We hypothesize that it is easier to optimize the residual mapping than to optimize the original, unreferenced mapping. To the extreme, if an identity mapping were optimal, it would be easier to push the residual to zero than to fit an identity mapping by a stack of nonlinear layers. This formulation is realized by the shortcut connections as seen in the following figure.

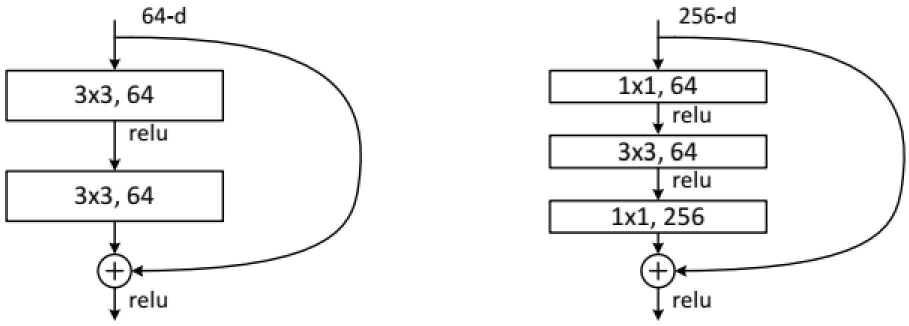

Figure 1: Residual block

- When adding shortcut connection between input and output that have same dimensions we use identity mapping i.e.,

  $$y = F(x, W_i) + x$$

  When dimensions increase we try two option

  1. The shortcut still performs identity mapping, with extra zero entries padded for increasing dimensions. This option introduces no extra parameter.
  2. We can perform a linear projection W_s by the shortcut connections to match the dimensions:

     $$y = F(x, W_i) + W_s x$$

     This adds extra para mater. We could also do this for matching dimensions but it didn't show significant gain in performance.

- For training deeper nets they use a bottleneck structure(Figure 1. right) where they use a stack of three layers instead of two.

- This ResNet approach doesn't show training accuracy degradation like just linearly stacking layer does as evident in Figure 2.

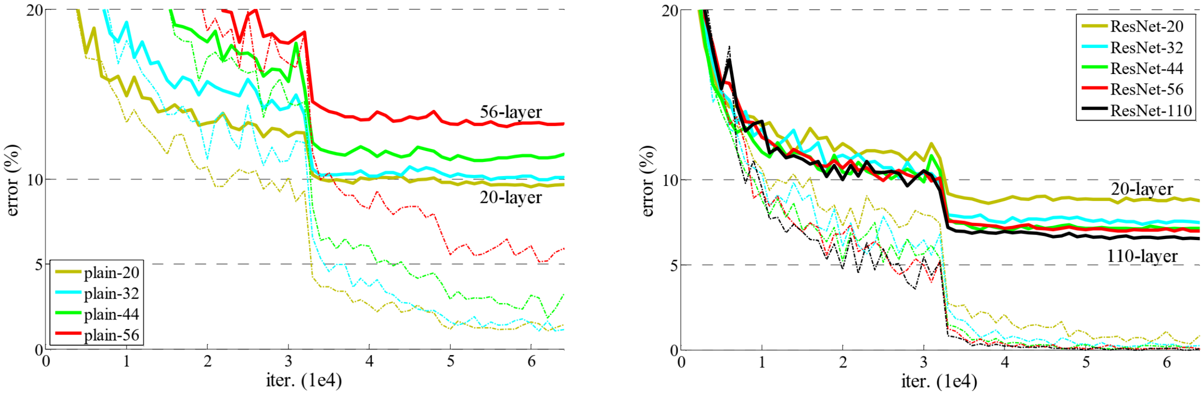

Figure 2: Training on CIFAR-10. Dashed lines denote training error, and bold lines denote testing error.

- An ensemble of these residual nets achieves 3.57% error on the ImageNet test set. This result won the 1st place on the ILSVRC 2015 classification task. Their ResNet based approach also won the 1st places on the tasks of ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation in 2015.