What Is Residual Neural Network (ResNet)?

Residual Network (ResNet) is a deep learning model used for computer vision applications. It is a Convolutional Neural Network (CNN) architecture designed to support hundreds or thousands of convolutional layers. Previous CNN architectures were not able to scale to a large number of layers, which resulted in limited performance. However, when adding more layers, researchers faced the “vanishing gradient” problem.

Neural networks are trained through a backpropagation process that relies on gradient descent, shifting down the loss function and finding the weights that minimize it. If there are too many layers, repeated multiplications will eventually reduce the gradient until it “disappears”, and performance saturates or deteriorates with each layer added.

ResNet provides an innovative solution to the vanishing gradient problem, known as “skip connections”. ResNet stacks multiple identity mappings (convolutional layers that do nothing at first), skips those layers, and reuses the activations of the previous layer. Skipping speeds up initial training by compressing the network into fewer layers.

Then, when the network is retrained, all layers are expanded and the remaining parts of the network—known as the residual parts—are allowed to explore more of the feature space of the input image.

Most ResNet models skip two or three layers at a time with nonlinearity and batch normalization in between. More advanced ResNet architectures, known as HighwayNets, can learn “skip weights”, which dynamically determine the number of layers to skip.

What Is a Residual Block?

Residual blocks are an important part of the ResNet architecture. In older architectures such as VGG16, convolutional layers are stacked with batch normalization and nonlinear activation layers such as ReLu between them. This method works with a small number of convolutional layers—the maximum for VGG models is around 19 layers. However, subsequent research discovered that increasing the number of layers could significantly improve CNN performance.

The ResNet architecture introduces the simple concept of adding an intermediate input to the output of a series of convolution blocks. This is illustrated below.

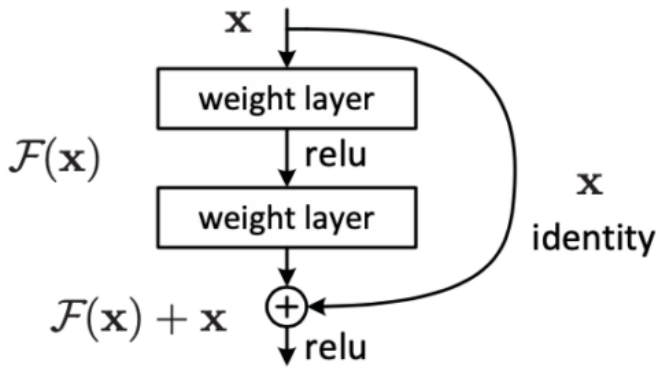

Figure 2. Residual learning: a building block.

The image above shows a typical residual block. This can be expressed in Python code using the expression *output = F(x) + x* where x is an input to the residual block and output from the previous layer, and F(x) is part of a CNN consisting of several convolutional blocks.

This technique smooths out the gradient flow during backpropagation, enabling the network to scale to 50, 100, or even 150 layers. Skipping a connection does not add additional computational load to the network.