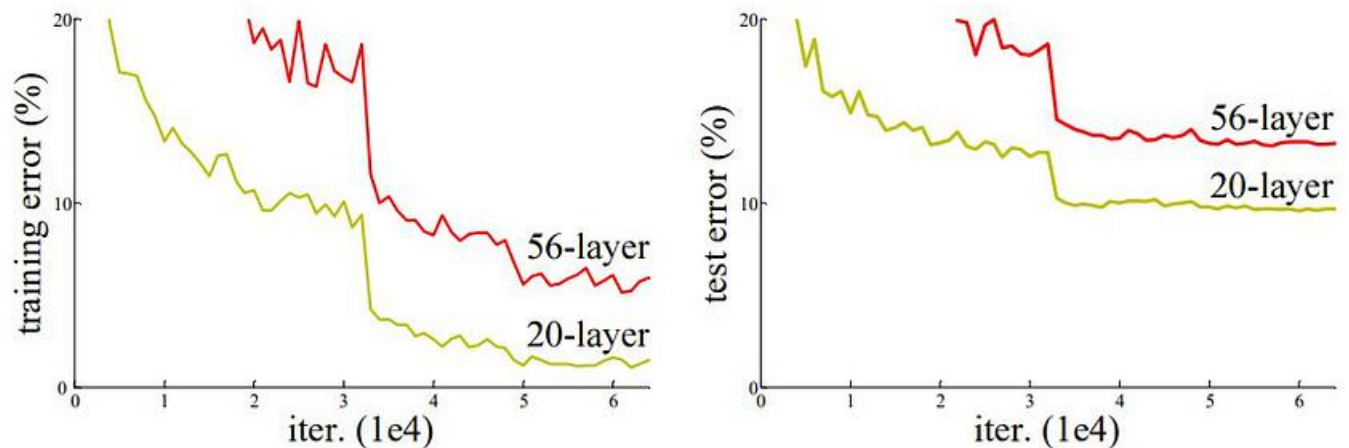




Module 5

ResNet : Architecture

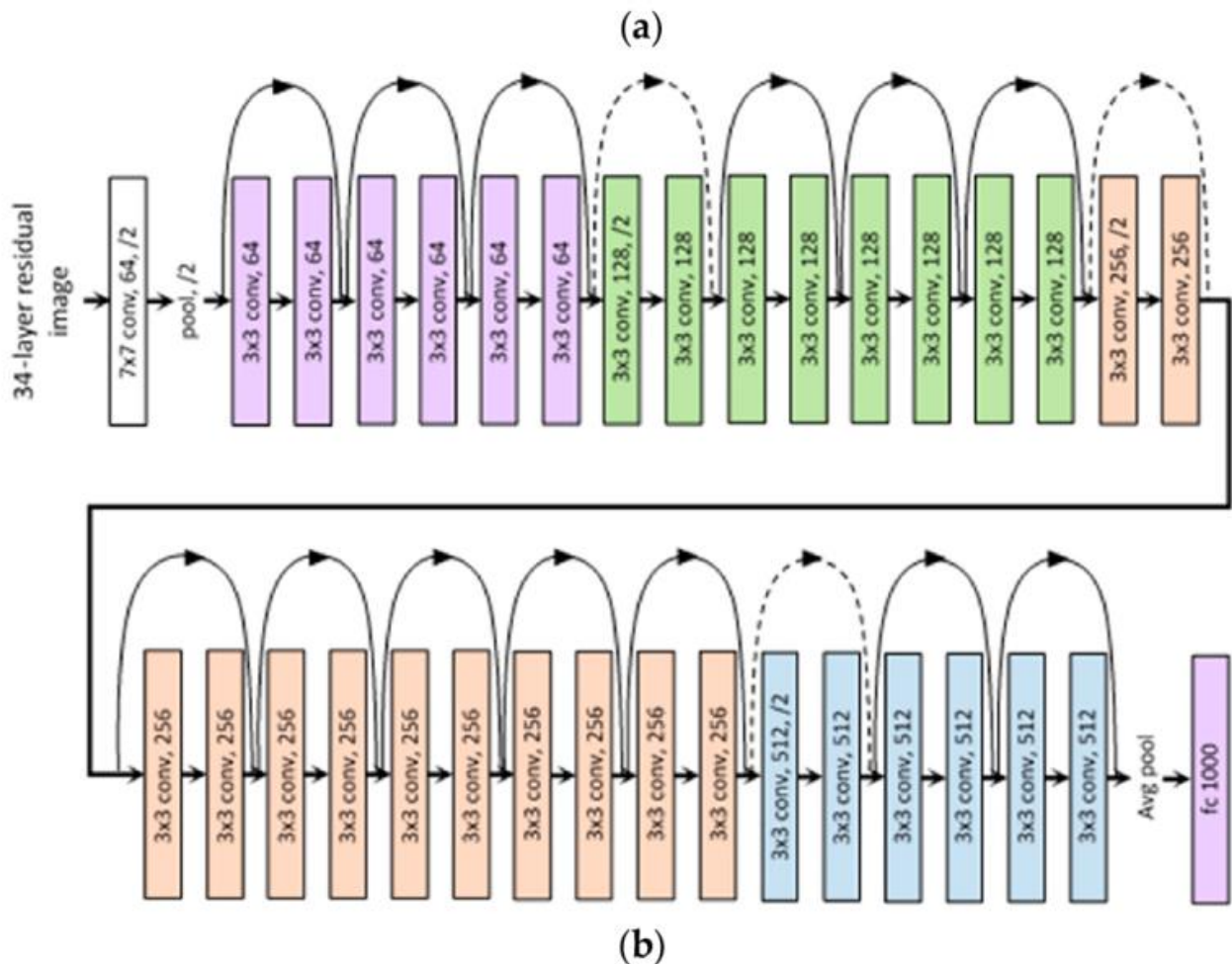
Every consecutive winning architecture uses more layers in a deep neural network to lower the error rate after the first CNN-based architecture (AlexNet) that won the ImageNet 2012 competition. This is effective for smaller numbers of layers, but when we add more layers, a typical deep learning issue known as the Vanishing/Exploding gradient arises. This results in the gradient becoming zero or being overly large. Therefore, the training and test error rate similarly increases as the number of layers is increased.



Comparison of 26-layer vs 56-layer architecture

We can see from the following figure that a 20-layer CNN architecture performs better on training and testing datasets than a 56-layer CNN architecture. The authors came to the conclusion that the error rate is caused by a vanishing/exploding gradient after further analysis of the error rate.

Module 5



ResNet-34 Layered architecture

A novel architecture called **Residual Network** was launched by Microsoft Research experts in 2015 with the proposal of **ResNet**.

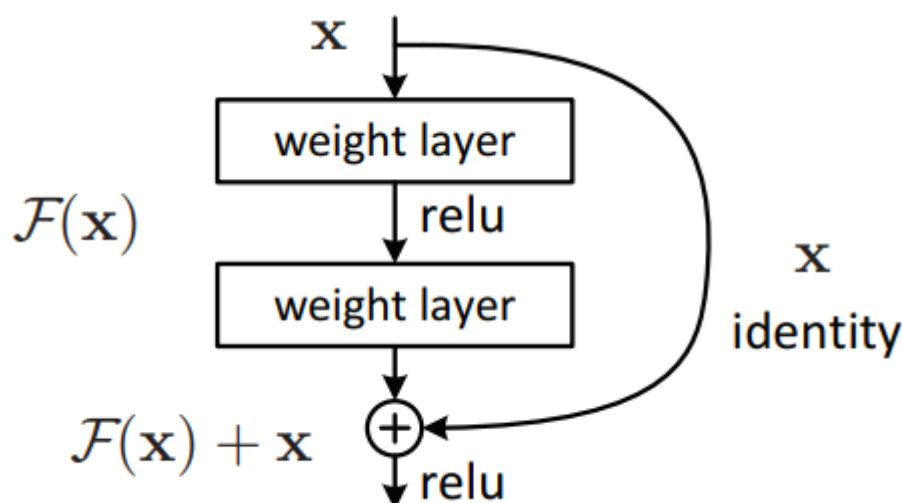
What is ResNet?

The Residual Blocks idea was created by this design to address the issue of the vanishing/exploding gradient. We apply a method known as skip connections in this network. The skip connection bypasses some levels in between to link-layer activations to subsequent layers. This creates a leftover block. These leftover blocks are stacked to create resnets.

The strategy behind this network is to let the network fit the residual mapping rather than have layers learn the underlying mapping. Thus, let the network fit instead of using, say, the initial mapping of $H(x)$,

Module 5

$F(x) := H(x) - x$ which gives $H(x) := F(x) + x$.



Skip connection or Shortcut

The benefit of including this kind of skip link is that regularisation will skip any layer that degrades architecture performance. As a result, training an extremely deep neural network is possible without encountering issues with vanishing or expanding gradients. The CIFAR-10 dataset's 100–1000 layers were used for experimentation by the paper's authors.

Similar techniques exist under the name “highway networks,” which also employ skip connections. These skip connections also make use of parametric gates, just as LSTM. The amount of data that flows across the skip connection is controlled by these gates. However, this design has not offered accuracy that is superior to ResNet architecture.

Salient features of ResNet Architecture:

- With a top-5 mistake rate of 3.57 per cent, won first prize in the ILSVRC classification competition in 2015. (A model ensemble)
- Won first prize in the categories of ImageNet detection, ImageNet localization, Coco detection, and Coco segmentation at the 2015 ILSVRC and COCO competition.
- ResNet-101 is used in Faster R-CNN to replace the VGG-16 layers. They noticed a 28 per cent relative improvement.
- Networks of 100 layers and 1000 layers that are well trained.

Why do we need ResNet?



Module 5

We stack extra layers in the Deep Neural Networks, which improves accuracy and performance, often in order to handle a challenging issue. The idea behind layering is that when additional layers are added, they will eventually learn features that are more complicated. For instance, when recognising photographs, the first layer may pick up on edges, the second would pick up on textures, the third might pick up on objects, and so on. However, it has been discovered that the conventional Convolutional neural network model has a maximum depth threshold. This graphic shows the percentage of errors for training and test data for a 20-layer network and a 56-layer network, respectively.

In both the training and testing situations, we can observe that the error per cent for a 56-layer network is higher than that of a 20-layer network. This shows that a network's performance declines as additional layers are added on top of it. This might be attributed to the initialization of the network, the optimization function, and most significantly, the vanishing gradient problem. You may assume that overfitting is also at blame, however, in this case, the 56-layer network's error percentage is the worst on both training and test data, which does not occur when the model is overfitting.

ResNet Architecture

The VGG-19-inspired 34-layer plain network architecture used by ResNet is followed by the addition of the shortcut connection. The architecture is subsequently transformed into the residual network by these short-cut connections, as depicted in the following figure:



The diagram illustrates three different convolutional neural network architectures: VGG-19, 34-layer plain, and 34-layer residual. Each architecture starts with an 'image' input and proceeds through a series of layers, with output sizes indicated on the left.

- VGG-19:** This architecture consists of 19 layers. It starts with two 3x3 conv layers (64 filters) followed by a pool, /2 operation. This is followed by two 3x3 conv layers (128 filters) followed by a pool, /2 operation. Then, there are four 3x3 conv layers (256 filters) followed by a pool, /2 operation. Next, there are four 3x3 conv layers (512 filters) followed by a pool, /2 operation. Finally, there are four 3x3 conv layers (512 filters) followed by a pool, /2 operation. The final output size is 7. The layers are color-coded: blue for the first two layers, orange for the next two, purple for the next four, green for the next four, and red for the last four.
- 34-layer plain:** This architecture consists of 34 layers. It starts with a 7x7 conv layer (64 filters, /2) followed by a pool, /2 operation. This is followed by eight 3x3 conv layers (64 filters). Then, there are eight 3x3 conv layers (128 filters, /2). Next, there are eight 3x3 conv layers (256 filters, /2). Finally, there are eight 3x3 conv layers (512 filters, /2). The final output size is 1000. The layers are color-coded: blue for the first two layers, purple for the next eight, green for the next eight, red for the next eight, and blue for the last eight.
- 34-layer residual:** This architecture consists of 34 layers. It starts with a 7x7 conv layer (64 filters, /2) followed by a pool, /2 operation. This is followed by eight 3x3 conv layers (64 filters). Then, there are eight 3x3 conv layers (128 filters, /2). Next, there are eight 3x3 conv layers (256 filters, /2). Finally, there are eight 3x3 conv layers (512 filters, /2). The final output size is 1000. The layers are color-coded: blue for the first two layers, purple for the next eight, green for the next eight, red for the next eight, and blue for the last eight. Residual connections are shown as curved arrows bypassing the convolutional layers and adding to the output of the previous layer.



Module 5

Resnet 34 Architecture

ResNet using Keras

An open-source, Python-based neural network framework called Keras may be used with TensorFlow, Microsoft Cognitive Toolkit, R, Theano, or PlaidML. It is made to make deep neural network experimentation quick. The following ResNet implementations are part of Keras Applications and offer ResNet V1 and ResNet V2 with 50, 101, or 152 layers,

- ResNet50
- ResNet101
- ResNet152
- ResNet50V2
- ResNet101V2
- ResNet152V2

ResNetV2 and the original ResNet (V1) vary primarily in that V2 applies batch normalisation before each weight layer.

ResNet 50

To implement the ResNet version1 with 50 layers (**ResNet 50**), we simply use the function from Keras as shown below:

Arguments used:

- **include_top**: whether to include the fully-connected layer at the top of the network.
- **weights**: one of None (random initialization), 'ImageNet' (pre-training on ImageNet), or the path to the weights file to be loaded.
- **input_tensor**: optional Keras tensor (i.e. output of layers.Input()) to use as image input for the model.
- **input_shape**: optional shape tuple, only to be specified if include_top is False (otherwise the input shape has to be (224, 224, 3) (with 'channels_last' data format) or (3, 224, 224) (with 'channels_first' data format). It should have exactly 3 inputs channels, and the width and height should be no smaller than 32. E.g. (200, 200, 3) would be one valid value.
- **pooling**: Optional pooling mode for feature extraction when include_top is False.

1. **None** means that the output of the model will be the 4D tensor output of the last convolutional block.



Module 5

2. **avg** means that global average pooling will be applied to the output of the last convolutional block, and thus the output of the model will be a 2D tensor.

3. **max** means that global max pooling will be applied.

- **classes**: optional number of classes to classify images into, only to be specified if `include_top` is True, and if no `weights` argument is specified.

Conclusion

- A robust backbone model called ResNet is utilised often in various computer vision tasks.
- ResNet employs skip connections to transfer output from one layer to another. This aids in reducing the issue of disappearing gradients.
- You may load ResNet 50 that has already been trained using Keras, or you can write ResNet by yourself.