ResNet-18 Model Documentation

Introduction to ResNet

ResNet (Residual Network) is a groundbreaking deep learning architecture that introduced the concept of residual learning to overcome the vanishing gradient problem commonly encountered in deep neural networks. Developed by **Kaiming He et al.** in 2015, ResNet's primary innovation is the **skip connection**, which allows information to bypass certain layers, leading to better gradient flow and more efficient training in deep architectures.

ResNet-18 is one of the simpler versions of ResNet, consisting of **18 layers**. It strikes a balance between computational efficiency and performance, making it widely used in tasks such as image classification, object detection, and even transfer learning for a wide variety of applications.

Architecture

The ResNet-18 architecture follows a standard structure of convolutional layers, batch normalization, activation functions, pooling layers, and fully connected layers, but what sets it apart is the **residual blocks**.

Residual Block

A residual block contains two convolutional layers followed by a **skip connection**. The key idea is to learn the residual mapping rather than directly mapping the input to the output. This can be mathematically expressed as:

y=F(x,{Wi})+xy = F(x, \{W_i\}) + xy=F(x,{Wi})+x

Where:

- F(x,{Wi})F(x, \{W_i\})F(x,{Wi}) represents the residual mapping to be learned.
- xxx is the input to the block.
- yyy is the output.
- The addition of xxx (input) ensures that the network retains useful information and allows gradients to flow more freely during backpropagation.

Layer Breakdown of ResNet-18

ResNet-18 consists of:

1. **Convolutional Layer 1**:
    - Input: Image (typically 224x224x3 for RGB images)
    - A 7x7 convolution filter with 64 kernels and stride 2.
    - Output: 64 feature maps of size 112x112.
    - **Batch Normalization** and **ReLU Activation**.
    - Max Pooling layer with a 3x3 filter and stride 2.
2. **Residual Blocks**: ResNet-18 consists of 4 stages, each containing 2 residual blocks.
    - **Stage 1**:
        - 2 residual blocks with 64 filters, each followed by a 3x3 convolution.
    - **Stage 2**:
        - 2 residual blocks with 128 filters, applied with 3x3 convolutions.
        - First residual block reduces the feature map size by half, i.e., 56x56.
    - **Stage 3**:
        - 2 residual blocks with 256 filters.
        - The feature map is further reduced in size to 28x28.
    - **Stage 4**:
        - 2 residual blocks with 512 filters, with the final output of 14x14 feature maps.
3. **Average Pooling Layer**:
    - Global average pooling is applied over the 14x14 feature maps, reducing them to a 1x1 spatial dimension (effectively flattening the output).
4. **Fully Connected Layer**:
    - The final fully connected layer outputs the class predictions.
    - For an image classification task, the output is typically passed through a softmax function to produce probabilities for each class.

The total number of layers in ResNet-18 is:

- 1 initial convolutional layer
- 16 layers in the form of residual blocks (with two 3x3 convolutional layers per block)
- 1 fully connected layer

Architecture Summary

Layer	Output Size	Details
Conv1	112 x 112	7x7, 64 filters, stride 2
MaxPool	56 x 56	3x3 max pooling, stride 2
Conv2_x (2 residual blocks)	56 x 56	3x3, 64 filters
Conv3_x (2 residual blocks)	28 x 28	3x3, 128 filters

Conv4_x (2 residual blocks)	14 x 14	3x3, 256 filters
Conv5_x (2 residual blocks)	7 x 7	3x3, 512 filters
Average Pooling	1 x 1	Global average pooling
Fully Connected (FC)	1 x 1	1000 (for ImageNet classification)

Key Features of ResNet-18

1. **Residual Learning**: The skip connections in ResNet-18 help avoid the vanishing gradient problem by letting gradients flow directly through the network, making it easier to train deep models.
2. **Shallow but Powerful**: Although ResNet-18 has only 18 layers, it delivers strong performance on many image classification tasks. Its simplicity allows it to be trained faster while still achieving good accuracy.
3. **Improved Gradient Flow**: The identity shortcut connections (skip connections) allow gradients to bypass one or more layers, reducing the chances of exploding or vanishing gradients during training.
4. **Transfer Learning**: ResNet-18 is often used as a backbone for transfer learning tasks. Pre-trained models on large datasets like ImageNet can be fine-tuned for domain-specific applications.
5. **Generalization**: The residual connections help the model generalize better across tasks, which is useful when applying ResNet-18 to different datasets and problems.

Training ResNet-18

1. **Data Augmentation**:
   - Common augmentations like random cropping, flipping, and color jittering are used to prevent overfitting and improve generalization.
2. **Learning Rate Scheduling**:
   - Learning rate schedules like step decay or cyclical learning rates can be applied to ensure efficient convergence.
3. **Loss Function**:
   - Typically, **cross-entropy loss** is used for classification tasks.
4. **Optimization**:
   - Optimizers like **SGD** with momentum or **Adam** are often employed during training.
5. **Batch Normalization**:
   - ResNet-18 incorporates batch normalization after each convolution, stabilizing the learning process and improving convergence.

Applications of ResNet-18

1. **Image Classification**: ResNet-18 has been highly successful on classification tasks, achieving high accuracy on benchmark datasets like ImageNet.
2. **Object Detection**: As a backbone in detection networks (e.g., Faster R-CNN), ResNet-18 is often used for detecting objects in real-time scenarios.
3. **Medical Image Analysis**: ResNet-18 has been adapted for tasks like tumor detection, lung nodule classification, and other diagnostic image tasks due to its strong generalization capability.
4. **Transfer Learning**: Pre-trained ResNet-18 models can be fine-tuned for various tasks like scene recognition, facial recognition, and other specialized domains.

Advantages of ResNet-18

1. **Efficient**: ResNet-18 is computationally lighter than deeper ResNet models like ResNet-50 or ResNet-101, making it more suitable for use in embedded or mobile devices.
2. **Better Training Stability**: Thanks to residual connections, ResNet-18 is easier to train compared to traditional deep networks, even with fewer layers.
3. **Transferability**: Pre-trained ResNet-18 models on large datasets like ImageNet can be fine-tuned on smaller, domain-specific datasets, leading to strong performance with minimal data.
4. **High Performance**: Despite its relatively low depth, ResNet-18 delivers competitive results on a wide range of image-based tasks.