CNN Architecture

CNN architectures, or Convolutional Neural Network architectures, have revolutionized the field of Computer Vision by significantly advancing the state-of-the-art in various tasks, including image classification, object detection, image segmentation, and more. CNNs excel at automatically learning hierarchical representations of visual data, making them highly effective for analyzing images and extracting meaningful features.

CNN architectures are specifically designed to exploit the spatial structure and correlations present in images. Unlike traditional fully connected neural networks, CNNs leverage two key components: convolutional layers and pooling layers.

Convolutional layers perform convolution operations, where small filters or kernels slide over the input image, extracting local features through element-wise multiplication and summing. These filters learn to detect various low-level and high-level visual patterns, such as edges, corners, and textures, by capturing local spatial relationships.

Here is the basic concept regarding the architecture of CNN

1) LeNet-5:

- [ ] LeNet-5 is a pioneering convolutional neural network (CNN) architecture proposed by Yann LeCun et al. in 1998.
- [ ] It was primarily designed for handwritten digit recognition.
- [ ] LeNet-5 consists of multiple layers, including convolutional layers, max-pooling layers, and fully connected layers.
- [ ] It was one of the first CNN architectures to achieve impressive results on image classification tasks.
- [ ] However, compared to more recent architectures, such as ResNet, LeNet-5 is relatively shallow and lacks some of the advanced features like residual connections.

2) AlexNet [2012]:

- [ ] AlexNet is a CNN architecture that gained significant attention after winning the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012.
- [ ] It was developed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. AlexNet consists of five convolutional layers, max-pooling layers, and three fully connected layers.
- [ ] It introduced several key innovations, including the use of Rectified Linear Units (ReLU) as activation functions, local response normalization, and the utilization of GPUs for efficient training.
- [ ] AlexNet demonstrated the potential of deep neural networks for image classification, and it paved the way for subsequent architectures.

3) VGGNet [2014]:

- [ ] VGGNet, proposed by Karen Simonyan and Andrew Zisserman in 2014, is known for its simplicity and depth.
- [ ] It features a uniform architecture with strictly 3x3 convolutional layers stacked on top of each other, followed by max-pooling layers.
- [ ] VGGNet comes in different variants, denoted by the total number of layers (e.g., VGG16, VGG19), and it achieved excellent performance on the ImageNet challenge.

- ☐ VGGNet's deep architecture allowed it to learn more complex features but also made it computationally expensive compared to previous models like AlexNet.

4) GoogleNet [2014]:
- ☐ GoogleNet, also known as Inception-v1, was developed by Christian Szegedy et al. in 2014.
- ☐ It introduced the concept of "inception modules," which are designed to capture features at multiple spatial scales using parallel convolutional operations of different filter sizes.
- ☐ GoogleNet emphasized computational efficiency while maintaining high accuracy.
- ☐ It achieved this by employing 1x1 convolutions to reduce the dimensionality within the network, reducing the number of parameters and computational cost.
- ☐ GoogleNet showed that CNNs could be both deep and efficient.

5) ResNet [2015]:
- ☐ ResNet, introduced by Kaiming He et al. in 2015, addressed the challenge of training very deep neural networks by introducing residual connections.
- ☐ ResNet leveraged the concept of residual learning, where the network learns to predict the residual between the input and the desired output.
- ☐ This approach allowed for the training of extremely deep networks (e.g., ResNet-152) by enabling easier flow of gradients and preventing the vanishing gradient problem.
- ☐ ResNet achieved state-of-the-art performance on various image recognition tasks and became a widely adopted architecture.

6) FractalNet:
- ☐ FractalNet, proposed by Gustav Larsson et al. in 2016, is an architecture that aims to improve the scalability and performance of deep networks.
- ☐ It employs a fractal structure in which lower layers approximate functions of higher layers.
- ☐ FractalNet introduces a set of fractal blocks, where each block consists of a stack of convolutional layers.
- ☐ The architecture allows for increased network depth without increasing the computational cost significantly.
- ☐ FractalNet demonstrated promising results in terms of improved accuracy and reduced memory usage.

7) DenseNet:
- ☐ DenseNet, introduced by Gao Huang et al. in 2016, is designed to address the vanishing gradient problem and encourage feature reuse.
- ☐ In DenseNet, each layer is connected to every other layer in a feed-forward fashion. This connectivity pattern encourages information flow across the network and enables gradient propagation more efficiently.
- ☐ DenseNet's dense connections facilitate feature reuse, allowing each layer to receive gradients from all subsequent layers and contribute to the overall network output.
- ☐ DenseNet has shown strong performance on image classification and segmentation tasks.

8) SqueezeNet:
- ☐ SqueezeNet, proposed by Forrest N. Iandola et al. in 2016, is a highly compact CNN architecture designed for efficient model deployment on resource-constrained devices.

- SqueezeNet achieves a smaller model size by replacing regular convolutions with 1x1 convolutions, reducing the number of parameters.
- It also employs "fire modules," which consist of a mix of 1x1 and 3x3 filters, to strike a balance between model size and accuracy.
- Despite its compact size, SqueezeNet achieves competitive performance on image classification tasks, making it suitable for scenarios with limited computational resources.

ResNet

The main idea behind ResNet is to address the problem of vanishing gradients that can occur when training very deep neural networks. As a network becomes deeper, it becomes harder for the network to learn and propagate gradients back to earlier layers, which can hinder the training process and limit the network's performance. To overcome this issue, ResNet introduces a new type of building block called a "residual block." A residual block consists of multiple convolutional layers, and instead of directly mapping the input to the output of the block, it learns the residual mapping—the difference between the input and the output. This residual mapping is then added back to the input, allowing the network to learn the residual information rather than trying to learn the entire mapping from scratch.

The convolutional layers in the residual block can have different sizes and numbers of filters depending on the specific architecture. The addition operation merges the input and the residual mapping to produce the output. By using this residual connection, the network can easily learn identity mappings (where the input and output are the same), and the gradients can flow through the network more easily, even in very deep architectures.

ResNet also introduces the concept of "skip connections" to further facilitate the flow of gradients. Skip connections allow the input to bypass one or more layers and be directly added to the output of a later layer in the network. This enables the network to learn both low-level and high-level features simultaneously.

In addition to the basic residual block, ResNet employs various configurations of these blocks to create deeper architectures. The original ResNet paper introduced several variants, including ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152, which differ in terms of the number of layers and the size of the convolutional filters.

Overall, ResNet's architecture with residual connections has proven to be highly effective in training very deep neural networks, enabling improved accuracy and faster convergence compared to previous architectures. It has become a widely adopted architecture for various computer vision tasks, including image classification, object detection, and image segmentation.