VGG

VGG Models and Their Impact

- VGG models (VGG-16 and VGG-19) have significantly influenced computer vision
- Developed by the **Visual Geometry Group (VGG)** from the University of Oxford
- Gained prominence in the **2014 ImageNet Large Scale Visual Recognition Challenge (ILSVRC)**
- Known for their **deep convolutional neural networks (CNNs) with a uniform architecture**
- **VGG-19**, the deeper variant, is recognized for its **simplicity and effectiveness**

Evolution of VGG Models

- **Predecessors of VGG Models**
  - **LeNet-5 (1990s)**: One of the first successful CNNs, used for handwritten digit recognition
  - **AlexNet (2012)**: Won ILSVRC by leveraging deeper architectures and GPU acceleration
- **Introduction of VGG Models**
  - Developed by **Karen Simonyan and Andrew Zisserman** in their 2014 paper
  - Titled **"Very Deep Convolutional Networks for Large-Scale Image Recognition"**
  - Investigated the impact of increasing CNN depth on image recognition tasks

Key Features of VGG-16 and VGG-19

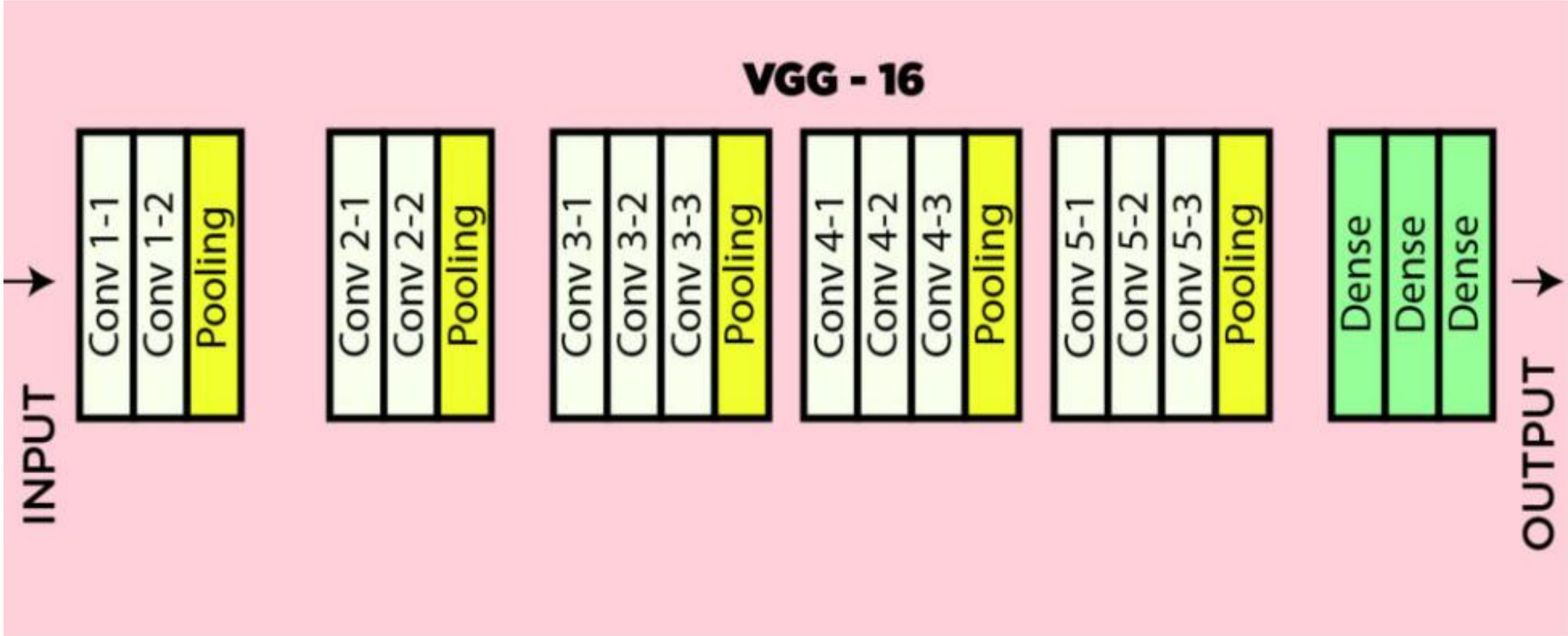
- **VGG-16**: 16 weight layers
- **VGG-19**: 19 weight layers
- Consistent use of **3×3 convolution filters** across all layers
- Simplified network structure leading to **improved performance**

VGG Architecture

- **VGG-16** is a deep **convolutional neural network (CNN)** designed for image classification
- Developed by the **Visual Geometry Group (VGG) at the University of Oxford**
- Known for its **simplicity and uniform architecture**, making it easy to understand and implement

VGG-16 Configuration

- **16 layers** in total:
  - **13 convolutional layers**
  - **3 fully connected layers**
- Layers are organized into **blocks**:
  - Each block contains **multiple convolutional layers**
  - Followed by a **max-pooling layer** for downsampling


VGG - 16
INPUT
Conv 1-1
Conv 1-2
Pooling
Conv 2-1
Conv 2-2
Pooling
Conv 3-1
Conv 3-2
Conv 3-3
Pooling
Conv 4-1
Conv 4-2
Conv 4-3
Pooling
Conv 5-1
Conv 5-2
Conv 5-3
Pooling
Dense
Dense
Dense
OUTPUT

VGG-16

1. Input Layer:
    1. Input dimensions: (224, 224, 3)
2. Convolutional Layers (64 filters, 3×3 filters, same padding):
    2. Two consecutive convolutional layers with 64 filters each and a filter size of 3×3.
    3. Same padding is applied to maintain spatial dimensions.
3. Max Pooling Layer (2×2, stride 2):
    3. Max-pooling layer with a pool size of 2×2 and a stride of 2.

4. Convolutional Layers (128 filters, 3×3 filters, same padding):

   - Two consecutive convolutional layers with 128 filters each and a filter size of 3×3.

5. Max Pooling Layer (2×2, stride 2):

   - Max-pooling layer with a pool size of 2×2 and a stride of 2.

6. Convolutional Layers (256 filters, 3×3 filters, same padding):

   - Two consecutive convolutional layers with 256 filters each and a filter size of 3×3.

7. Convolutional Layers (512 filters, 3×3 filters, same padding):

   - Two sets of three consecutive convolutional layers with 512 filters each and a filter size of 3×3.

8. Max Pooling Layer (2×2, stride 2):

Max-pooling layer with a pool size of 2×2 and a stride of 2.

9. Stack of Convolutional Layers and Max Pooling:

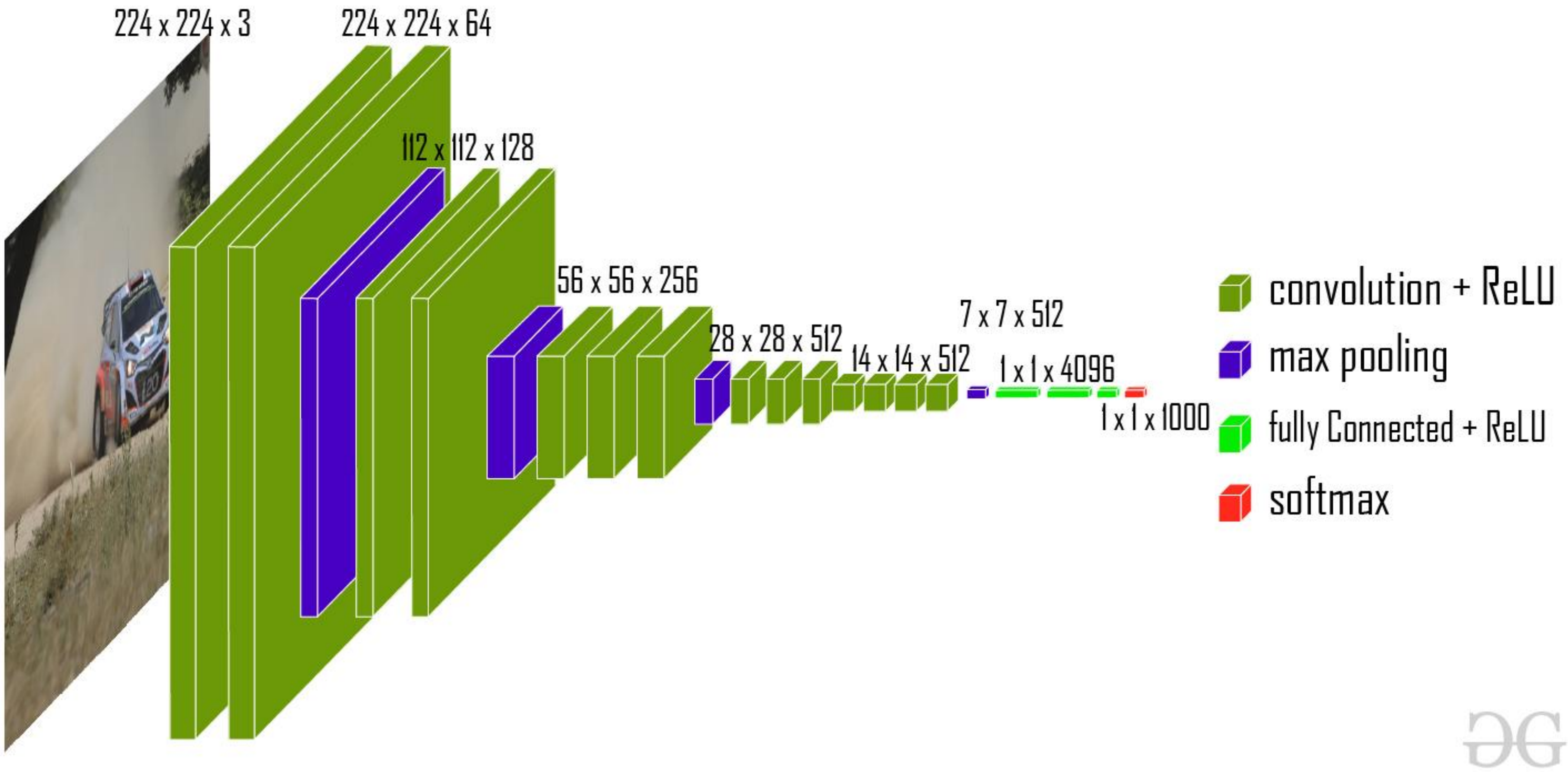
- Two additional convolutional layers after the previous stack.
- Filter size: 3×3.Flattening:

- Flatten the output feature map (7x7x512) into a vector of size 25088.

11. Fully Connected Layers:

- Three fully connected layers with ReLU activation.
- First layer with input size 25088 and output size 4096.
- Second layer with input size 4096 and output size 4096.
- Third layer with input size 4096 and output size 1000, corresponding to the 1000 classes in the ILSVRC challenge.
- Softmax activation is applied to the output of the third fully connected layer for classification.

This architecture follows the specifications provided, including the use of ReLU activation function and the final fully connected layer outputting probabilities for 1000 classes using softmax activation.


224 x 224 x 3
224 x 224 x 64
112 x 112 x 128
56 x 56 x 256
28 x 28 x 512
14 x 14 x 512
7 x 7 x 512
1 x 1 x 4096
1 x 1 x 1000
convolution + ReLU
max pooling
fully Connected + ReLU
softmax

Limitations Of VGG 16:

- It is very slow to train (the original VGG model was trained on Nvidia Titan GPU for 2-3 weeks).

- The size of VGG-16 trained imageNet weights is *528* MB. So, it takes quite a lot of disk space and bandwidth which makes it inefficient.

- 138 million parameters lead to exploding gradients problem.

Further advancements: Resnets are introduced to prevent exploding gradients problem that occurred in VGG-16.

VGG-19 Architecture

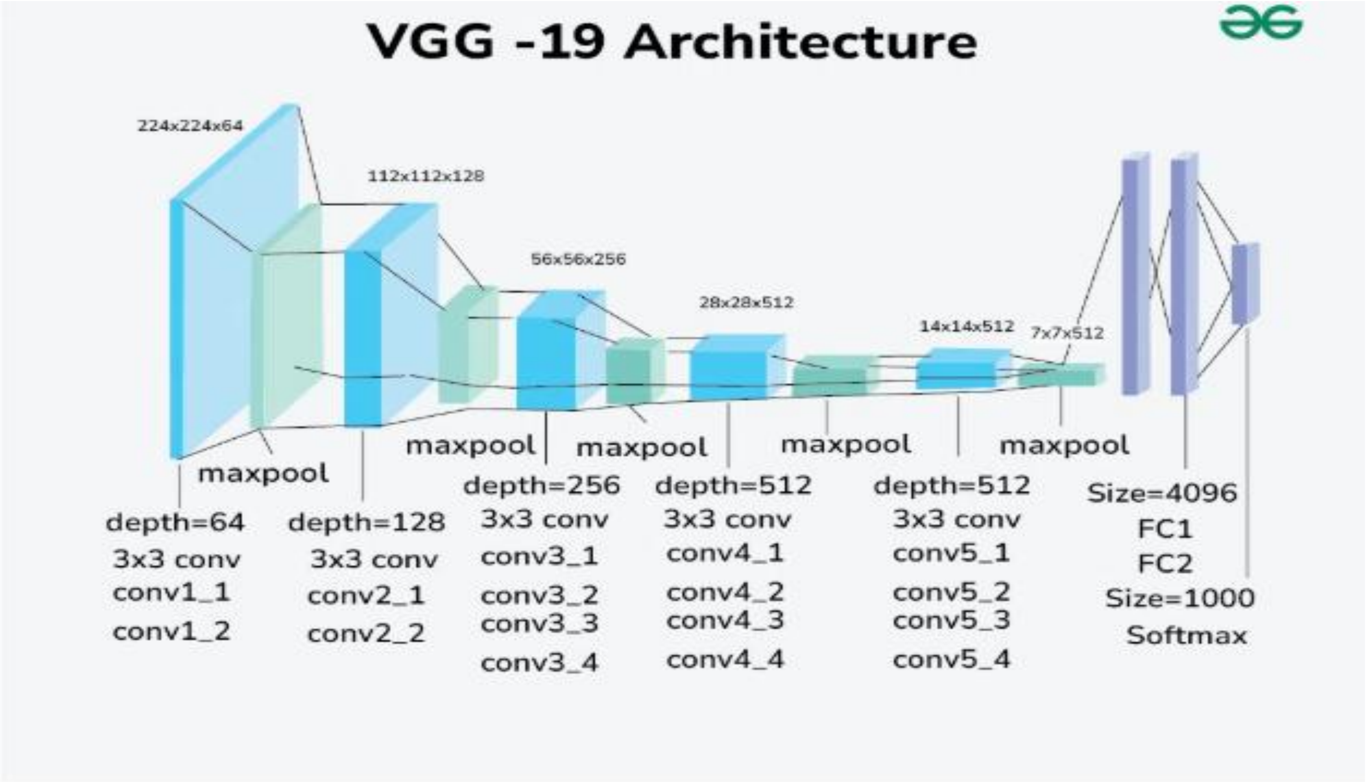
VGG-19 is a deep convolutional neural network with 19 weight layers, comprising 16 convolutional layers and 3 fully connected layers. The architecture follows a straightforward and repetitive pattern, making it easier to understand and implement.

The key components of the VGG-19 architecture are:

1. **Convolutional Layers**: 3x3 filters with a stride of 1 and padding of 1 to preserve spatial resolution.
2. **Activation Function**: ReLU (Rectified Linear Unit) applied after each convolutional layer to introduce non-linearity.
3. **Pooling Layers**: Max pooling with a 2x2 filter and a stride of 2 to reduce the spatial dimensions.
4. **Fully Connected Layers**: Three fully connected layers at the end of the network for classification.
5. **Softmax Layer**: Final layer for outputting class probabilities.

Detailed Layer-by-Layer Architecture of VGG-Net 19

The VGG-19 model consists of five blocks of convolutional layers, followed by three fully connected layers. Here is a detailed breakdown of each block:



VGG-19 Architecture

Block 1

- Conv1_1: 64 filters, 3x3 kernel, ReLU activation
- Conv1_2: 64 filters, 3x3 kernel, ReLU activation
- Max Pooling: 2x2 filter, stride 2

Block 2

- Conv2_1: 128 filters, 3x3 kernel, ReLU activation
- Conv2_2: 128 filters, 3x3 kernel, ReLU activation
- Max Pooling: 2x2 filter, stride 2

Block 3

- Conv3_1: 256 filters, 3x3 kernel, ReLU activation
- Conv3_2: 256 filters, 3x3 kernel, ReLU activation
- Conv3_3: 256 filters, 3x3 kernel, ReLU activation
- Conv3_4: 256 filters, 3x3 kernel, ReLU activation
- Max Pooling: 2x2 filter, stride 2

Block 4

- Conv4_1: 512 filters, 3x3 kernel, ReLU activation
- Conv4_2: 512 filters, 3x3 kernel, ReLU activation
- Conv4_3: 512 filters, 3x3 kernel, ReLU activation
- Conv4_4: 512 filters, 3x3 kernel, ReLU activation
- Max Pooling: 2x2 filter, stride 2

Block 5

- Conv5_1: 512 filters, 3x3 kernel, ReLU activation
- Conv5_2: 512 filters, 3x3 kernel, ReLU activation
- Conv5_3: 512 filters, 3x3 kernel, ReLU activation
- Conv5_4: 512 filters, 3x3 kernel, ReLU activation
- Max Pooling: 2x2 filter, stride 2

Fully Connected Layers

- **FC1**: 4096 neurons, ReLU activation
- **FC2**: 4096 neurons, ReLU activation
- **FC3**: 1000 neurons, softmax activation (for 1000-class classification)

Architectural Design Principles

The VGG-19 architecture follows several key design principles:

1. **Uniform Convolution Filters**: Consistently using 3x3 convolution filters simplifies the architecture and helps maintain uniformity.
2. **Deep Architecture**: Increasing the depth of the network enables learning more complex features.
3. **ReLU Activation**: Introducing non-linearity helps in learning complex patterns.
4. **Max Pooling**: Reduces the spatial dimensions while preserving important features.
5. **Fully Connected Layers**: Combines the learned features for classification.

Impact and Legacy of VGG-19

Influence on Subsequent Models

- Inspired architectures like **ResNet** and **Inception**
- Showcased that **increasing depth improves image recognition performance**

Use in Transfer Learning

- Extensively used for **feature extraction** in various applications
- Pre-trained on **ImageNet** and fine-tuned for tasks like **object detection, segmentation, and style transfer**

Research and Industry Applications

- Used as a **baseline** for academic research and model comparisons
- Applied in industries such as **medical imaging** and **autonomous vehicles**

Additional Information about VGG-19

- **Model Simplicity & Effectiveness**: Uses **3×3 convolution filters** and a repetitive block structure
- **High Computational Demand**: Requires **significant memory and processing power**
- **Robust Feature Extraction**: Captures intricate details, making it ideal for **transfer learning**
- **Data Augmentation**: Techniques like **random cropping, flipping, and color jittering** improve generalization
- **Influence on Network Design**: Inspired **modern deep learning architectures** with small filters and deep networks

Conclusion

In conclusion, VGG-19 stands as a landmark model in the history of deep learning, combining simplicity with depth to achieve remarkable performance. Its architecture serves as a foundation for many modern neural networks, highlighting the enduring impact of its design principles on the field of computer vision.