Efficientnet Architecture

Last Updated : 03 Jun, 2024

In the field of deep learning, the quest for more efficient neural network architectures has been ongoing. EfficientNet has emerged as a beacon of innovation, offering a holistic solution that balances model complexity with computational efficiency. This article embarks on a detailed journey through the intricate layers of EfficientNet, illuminating its architecture, design philosophy, training methodologies, performance benchmarks, and more.

Table of Content

Efficientnet

EfficientNet is a family of convolutional neural networks (CNNs) that aims to achieve high performance with fewer computational resources compared to previous architectures. It was introduced by Mingxing Tan and Quoc V. Le from Google Research in their 2019 paper "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks." The core idea behind EfficientNet is a new scaling method that uniformly scales

all dimensions of depth, width, and resolution using a compound coefficient.

EfficientNet-B0 Architecture Overview

The EfficientNet-B0 network consists of:

1. **Stem**
   - Initial layer with a standard convolution followed by a batch normalization and a ReLU6 activation.
   - Convolution with 32 filters, kernel size 3x3, stride 2.

2. **Body**
   - Consists of a series of MBConv blocks with different configurations.
   - Each block includes depthwise separable convolutions and squeeze-and-excitation layers.
   - Example configuration for MBConv block:
     - Expansion ratio: The factor by which the input channels are expanded.
     - Kernel size: Size of the convolutional filter.
     - Stride: The stride length for convolution.
     - SE ratio: Ratio for squeeze-and-excitation.

3. **Head**
   - Includes a final convolutional block, followed by a global average pooling layer.
   - A fully connected layer with a softmax activation function for classification.

EfficientNet-B0 Detailed Architecture

EfficientNet uses a technique called compound coefficient to scale up models in a simple but effective manner. Instead of randomly scaling up width, depth, or resolution, compound scaling uniformly scales each dimension with a certain fixed set of scaling coefficients. Using this scaling method and AutoML, the authors of EfficientNet developed seven models of various dimensions, which surpassed the state-of-the-

art accuracy of most convolutional neural networks, and with much better efficiency.

From the table, the architecture of EfficientNet-B0 can be summarized as follows:

Stage	Operator	Resolution	#Channels	#Layers
1	Conv3x3	224 × 224	32	1
2	MBConv1, k3x3	112 × 112	16	1
3	MBConv6, k3x3	112 × 112	24	2
4	MBConv6, k5x5	56 × 56	40	2
5	MBConv6, k3x3	28 × 28	80	3
6	MBConv6, k5x5	14 × 14	112	3
7	MBConv6, k5x5	14 × 14	192	4
8	MBConv6, k3x3	7 × 7	320	1
9	Conv1x1 & Pooling & FC	7 × 7	1280	1

Compound Scaling Method

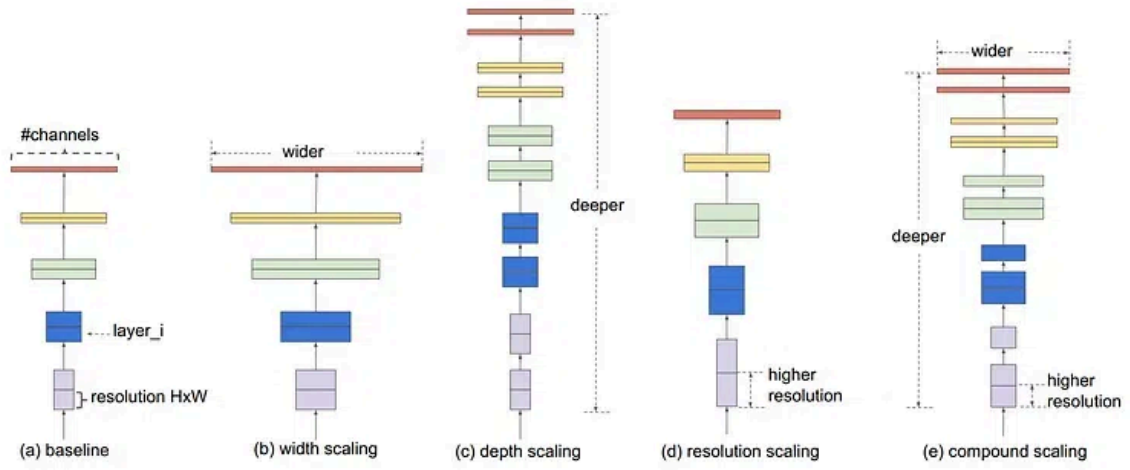
At the heart of EfficientNet lies a revolutionary compound scaling method, which orchestrates the simultaneous adjustment of network width, depth, and resolution using a set of fixed scaling coefficients. This approach ensures that the model adapts seamlessly to varying computational constraints while preserving its performance across different scales and tasks.

Compound Scaling:

The authors thoroughly investigated the effects that every scaling strategy has on the effectiveness and performance of the model before creating the compound scaling method. They came to the conclusion that, although scaling a single dimension can help improve model performance, the best way to increase model performance overall is to balance the scale in all three dimensions (width, depth, and image resolution) while taking the changeable available resources into consideration.

The below images show the different methods of scaling:

1. **Baseline**: The original network without scaling.
2. **Width Scaling**: Increasing the number of channels in each layer.
3. **Depth Scaling**: Increasing the number of layers.
4. **Resolution Scaling**: Increasing the input image resolution.
5. **Compound Scaling**: Simultaneously increasing width, depth, and resolution according to the compound scaling formula.



Different scaling methods vs. Compound scaling

This is achieved by uniformly scaling each dimension with a compound coefficient φ. The formula for scaling is:

$$Width \times Depth^2 \times Resolution^2 \approx Constant$$

The principle behind the compound scaling approach is to scale with a constant ratio in order to balance the width, depth, and resolution parameters.

Depth-wise Separable Convolution

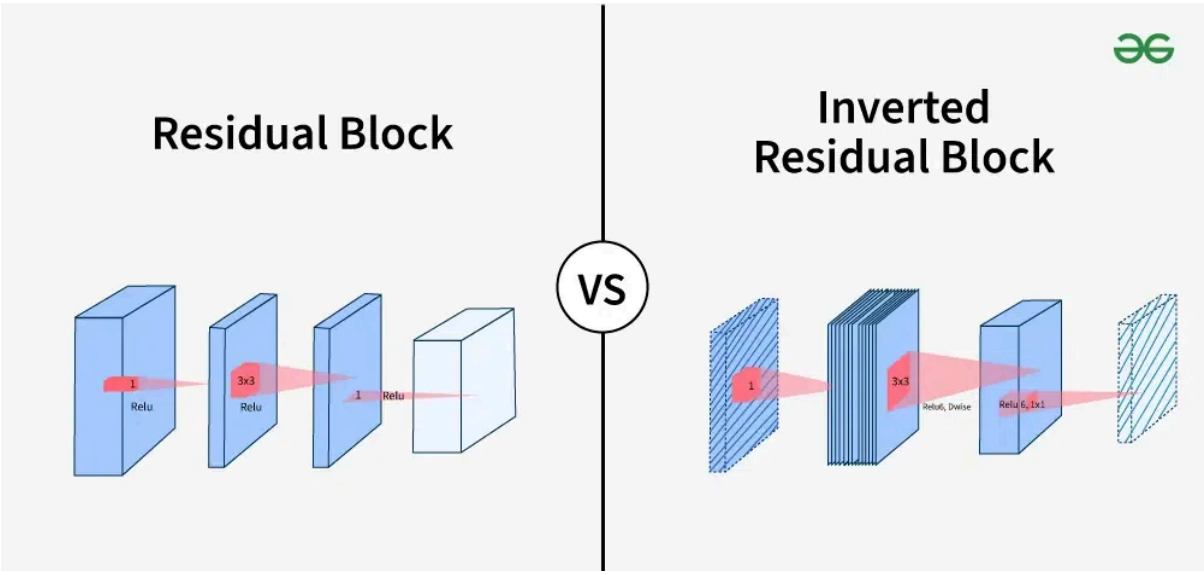
EfficientNet uses depth-wise separable convolutions to lower computational complexity without sacrificing representational capability. This is achieved by splitting the normal convolution into two parts:

1. **Depth-wise Convolution**: Applies a single filter to each input channel.
2. **Point-wise Convolution**: Aggregates features from different channels.

This makes the network more efficient by requiring fewer computations and parameters.

Inverted Residual Blocks

Inspired by MobileNetV2, EfficientNet employs inverted residual blocks to further optimize resource usage. These blocks start with a lightweight depth-wise convolution followed by point-wise expansion and another depth-wise convolution. Additionally, squeeze-and-excitation (SE) operations are incorporated to enhance feature representation by recalibrating channel-wise responses.



Inverted Residual Block Structure

An inverted residual block follows a narrow -> wide -> narrow structure:

1. **Expansion Phase**: Increase the number of feature maps with a 1x1 convolutional layer.
2. **Depth-wise Convolution**: Use a 3x3 convolutional bottleneck layer.
3. **Projection Phase**: Shrink the number of feature maps back to the original input number with a 1x1 convolutional layer.

Efficient Scaling:

EfficientNet achieves efficient scaling by progressively increasing model depth, width, and resolution based on the compound scaling coefficient φ. This allows for the creation of larger and more powerful models without significantly increasing computational overhead. By carefully balancing these dimensions, EfficientNet achieves state-of-the-art performance while remaining computationally efficient.

Efficient Attention Mechanism:

EfficientNet incorporates efficient attention mechanisms, such as squeeze-and-excitation (SE) blocks, to improve feature representation. SE blocks selectively amplify informative features by learning channel-wise attention weights. This enhances the discriminative power of the network while minimizing computational overhead.

Variants of EfficientNet Model:

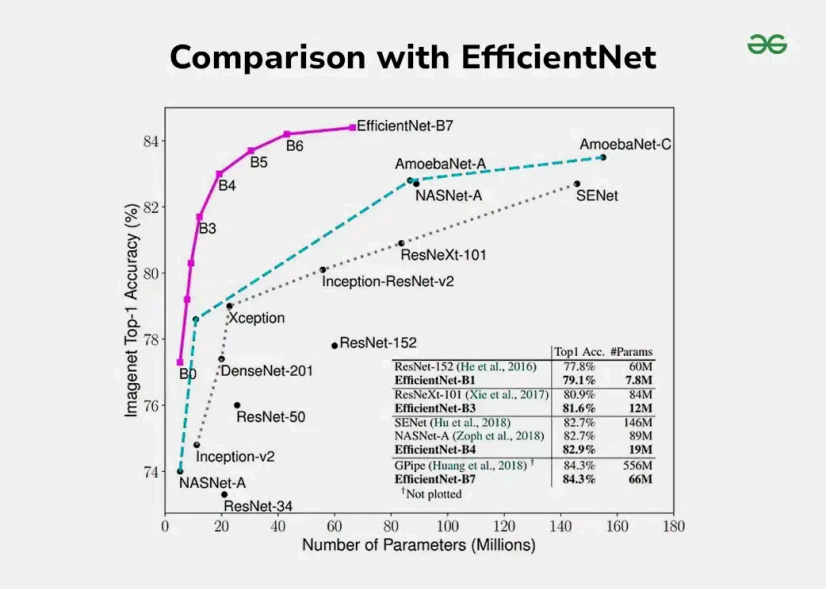
EfficientNet offers several variants, denoted by scaling coefficients like B0, B1, B2, etc. These variants differ in depth, width, and resolution based on the compound scaling approach. For example:

- **EfficientNet-B0:** The baseline model with moderate depth, width, and resolution.
- **EfficientNet-B1 to B7:** Successively larger variants achieved by increasing the compound scaling coefficient φ.
- **EfficientNet-Lite:** Lightweight variants designed for mobile and edge devices, achieving a good balance between performance and efficiency.

Each variant of EfficientNet offers a trade-off between model size, computational cost, and performance, catering to various deployment scenarios and resource constraints.

Performance Evaluation and Comparison

Evaluating the efficacy of EfficientNet involves subjecting it to various performance benchmarks and comparative analyses. Across multiple benchmark datasets and performance metrics, EfficientNet demonstrates outstanding efficiency, outperforming its predecessors in terms of accuracy, computational cost, and resource utilization.



For instance, on the ImageNet dataset, the largest EfficientNet model, EfficientNet-B7, achieved approximately 84.4% top-1 and 97.3% top-5 accuracy. Compared to the previous best CNN model, EfficientNet-B7 was 6.1 times faster and 8.4 times smaller in size. On the CIFAR-100 dataset, it achieved 91.7% accuracy, and on the Flowers dataset, 98.8% accuracy.

Efficiency and Performance

- **Efficiency**: EfficientNet achieves state-of-the-art accuracy on ImageNet with significantly fewer parameters and FLOPS compared to previous models like ResNet, DenseNet, and Inception.

- **Performance**: Due to the balanced scaling method, EfficientNet models provide an excellent trade-off between accuracy and computational efficiency, making them suitable for deployment in resource-constrained environments.

Conclusion

EfficientNet stands as a testament to the ingenuity of modern deep learning architectures. Its scalable design, coupled with efficient training methodologies, positions it as a versatile tool for a myriad of computer vision tasks. As we navigate the ever-expanding landscape of artificial intelligence, EfficientNet serves as a guiding light, illuminating the path towards more efficient and effective neural network designs.

Next Article

Website Architecture Optimization

Similar Reads

Spine-Leaf Architecture

Spine-Leaf Architecture is a two-layer architecture. This architecture has two layers i.e The spine layer and the Leaf layer. The Spine layer and Le...

15+ min read

Website Architecture Optimization

Just like a well-organized store makes it easier for customers to find what they need, a strong site architecture is fundamental for a successful...

15+ min read

Architecture of a System

Architecture is a critical aspect of designing a system, as it sets the foundation for how the system will function and be built. It is the process...

15+ min read

Client-Server Architecture - System Design

Client-server architecture is a fundamental concept in system design where a network involves multiple clients and a server. Clients are devic...

15+ min read

Data-Driven Architecture - System Design

Data-driven architecture is an emerging paradigm in system design that prioritizes data as a core element in shaping applications and services. B...

15+ min read

Kappa Architecture - System Design

The Kappa Architecture is a streamlined approach to system design focused on real-time data processing. Unlike the Lambda Architecture,...

15+ min read

Complete Guide to Clean Architecture

Clean Architecture is a software design approach that promotes the separation of concerns, ensuring systems are maintainable, scalable, and...

15+ min read

Hexagonal Architecture - System Design

Hexagonal Architecture, also known as Ports and Adapters Architecture, is a design pattern used in system development. It focuses on making...

15+ min read

Federated Architecture - System Design

A Federated Architecture in system design is a decentralized approach where independent components or services collaborate to achieve a...

15+ min read

Master-Slave Architecture

One essential design concept is master-slave architecture. Assigning tasks between central and subordinate units, it transforms system coordinatio...

15+ min read

GeeksforGeeks
SanchhayaEducation Private Limited

Corporate & Communications Address:

A-143, 7th Floor, Sovereign Corporate Tower, Sector- 136, Noida, Uttar Pradesh (201305)

Registered Address:

K 061, Tower K, Gulshan Vivante Apartment, Sector 137, Noida, Gautam Buddh Nagar, Uttar Pradesh, 201305

Python Tutorial

Python Programming Examples
Django Tutorial
Python Projects
Python Tkinter
Web Scraping
OpenCV Tutorial
Python Interview Question

Computer Science

GATE CS Notes
Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths

DevOps

Git
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar

Databases

SQL
MYSQL
PostgreSQL
PL/SQL
MongoDB

Preparation Corner

Company-Wise Recruitment Process
Aptitude Preparation
Puzzles
Company-Wise Preparation

More Tutorials

Software Development
Software Testing
Product Management
Project Management
Linux
Excel
All Cheat Sheets

Machine Learning/Data Science

Complete Machine Learning & Data Science Program - [LIVE]
Data Analytics Training using Excel, SQL, Python & PowerBI - [LIVE]
Data Science Training Program - [LIVE]
Data Science Course with IBM Certification

Programming Languages

C Programming with Data Structures
C++ Programming Course
Java Programming Course
Python Full Course

Clouds/Devops

DevOps Engineering
AWS Solutions Architect Certification
Salesforce Certified Administrator Course

GATE 2026

GATE CS Rank Booster
GATE DA Rank Booster
GATE CS & IT Course - 2026
GATE DA Course 2026
GATE Rank Predictor