CNN Architectures and Evolution

Parth Agrawal

January 2026

1 Introduction

In this report, we will review the timeline of the evolution of CNN Architectures. From LeNet-5 in 1998 to EfficientNet in 2019, we will cover all the significant milestones that led these simple feature extractors to transform the world.

2 LeNet-5: The Start of CNN's Rise (1998)

LeNet-5 was initially developed by Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner to recognize handwritten characters. It established the classic sequence of convolution block -> pooling layers -> fully connected neural network.

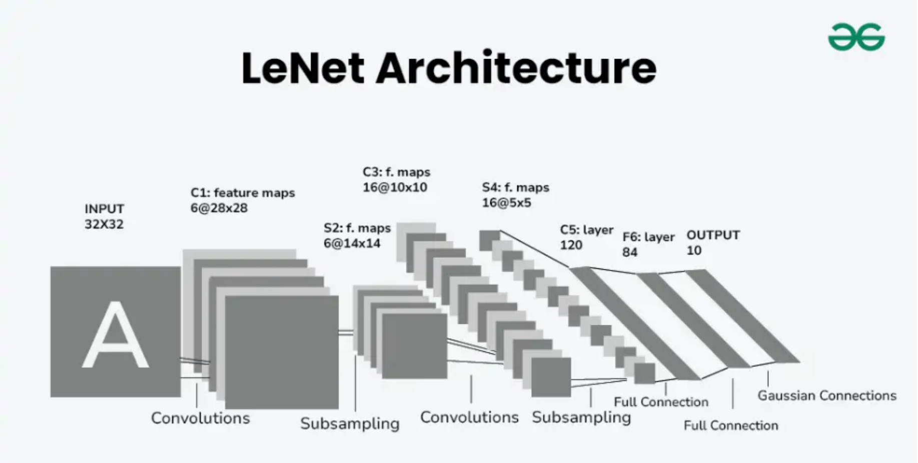


Figure 1: LeNet-5 Architecture

Its architecture is something like this - 1. Input Layer (32 x 32 pixels) 2. C1 Layer (Convolution Layer) 3. S2 Layer (Subsampling / Pooling Layer)

4. C3 Layer (Convolution Layer) 5. S4 Layer (Pooling Layer) 6. C5 layer (Convolution layer) 7. F6 Layer (Fully Connected layer) 8. Output layer

For an input $W \times W$, filter $F \times F$, padding P, and stride S:

$$\text{Output Size} = \frac{W - F + 2P}{S} + 1$$

It was highly successful in recognizing handwritten digits.

3 AlexNet: The Major Breakthrough (2012)

AlexNet was the first breakthrough that made the world see deep learning as a reliable field. AlexNet won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) by a considerable margin over its competitors.

ImageNet was a dataset containing 12 million labelled images in over 1,000 categories. In 2012, AlexNet became the first to achieve an error rate of less than 25% (16.4%). Interestingly, AlexNet achieved this by using Convolutional Neural Networks, which were deemed impractical by most until that point.

AlexNet was one of the first models to utilize a GPU for training, enabling faster results and unlocking the potential for larger models. AlexNet itself was trained on two GPUs with 3GB of memory each.

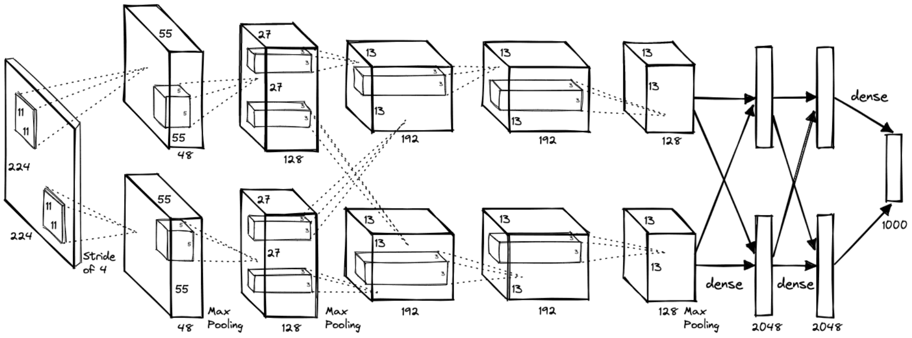


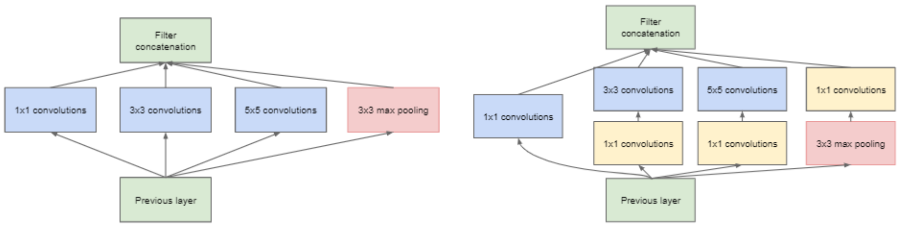
Figure 2: Alexnet Architecture

4 InceptionNet/GoogLeNet: Further Progress (2014)

In ILSVRC 2014, GoogLeNet emerged as the winner, with a significant improvement. The network architecture, however, differed significantly from that of other models. It used 1x1 convolution and Global Average Pooling from the *Network*

In Network paper, and also used a technique called the inception module. Let us discuss these three techniques.

- **1x1 convolution -** It is a technique to reduce dimensions. It is performing a 1x1 convolution before the actual convolution operation. It can reduce the required number of operations by a considerable margin and also creates a bottleneck, which reduces the risk of overfitting.
- **Inception module -** The inception module is basically, rather than assigning a fixed layer like 3x3 or 5x5 convolution, we let the network pick and choose the best option. This usually would increase parameters and operations by a significant factor, but with 1x1 convolution, we can practically implement this.
- **Global Average Pooling -** Instead of using a fully connected layer and a softmax function, what GoogLeNet did was to use a Global Average Pooling at the end. This vastly reduced the risk of overfitting.



(a) Inception module, naïve version

(b) Inception module with dimension reductions

Figure 3: Inception Module

This was a significantly deeper model compared to AlexNet and VGG.

5 ResNet: Solving the Depth Barrier (2015)

Kaiming He and his team introduced ResNet (Residual Networks). It addressed the problem of vanishing gradients (repeated multiplications causing small values).

It features a deep and bottleneck architecture, utilizing Batch Normalisation and ReLU. It mainly introduced two new features -

- **Residual Learning -** Rather than learning from the output block, the model learns from a residual function (difference between output and input block).
- **Skip Connections -** It was the key innovation in ResNet. These connections skip one or more layers and add the output of the earlier layer directly to the later layer. This helps mitigate the problem of vanishing gradients in deep networks.

Instead of learning a direct mapping $H(x)$, the block learns the Residual $F(x) = H(x) - x$.

The output is:

$$y = F(x) + x$$

. If $F(x)$ becomes zero, the layer performs an Identity Mapping, ensuring performance does not degrade.

The deeper networks and easier training allow ResNet to achieve much better performance than GoogLeNet.

6 MobileNet: Increasing Efficiency (2017)

MobileNet was a neural network launched by Google in 2017 for mobile devices. The primary goal was to provide high-performance, low-latency image classification for mobile devices.

It achieves this by using breaking convolutions in two steps - Depthwise separable convolutions and Pointwise separable convolutions.

- **Depthwise Separable Convolutions -** In this step, a small filter is used, and a convolution at each channel is performed. The output is the same size but with fewer channels.
- **Pointwise Separable Convolutions -** In this type of convolution, a single filter (usually 1x1) is applied across all the channels in both input and output layers. It is an alternative to a fully connected layer, making it suitable for devices having limited computational resources.

7 EfficientNet: Compound Scaling (2019)

Until this point, people tried to improve the models by adding more layers or increasing the size of layers. However, EfficientNet (made by Researchers at Google AI in 2019) introduced something called Compound Scaling.

In Compound Scaling, the depth (number of layers), width (number of units in each layer) and image resolution (the detail level of input images) are increased together in a balanced way (fixed proportions). This resulted in a smaller and faster model that can perform better than ResNet.

These models achieved high accuracy with fewer parameters, and scaled from B0 to B7 (small to large). They proved to be a good fit for Transfer learning. However, these models required more memory and worked more slowly on some hardware.

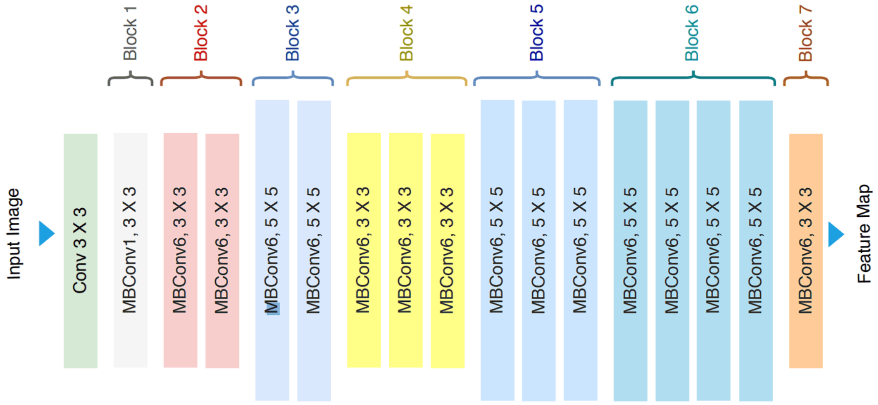


Figure 4: Architecture of EfficientNet-B0

8 References

- GeeksforGeeks. (2025, December 10). LeNet-5 architecture. GeeksforGeeks. https://www.geeksforgeeks.org/computer-vision/lenet-5-architecture/
- Pinecone. (n.d.). AlexNet and ImageNet: The birth of deep learning. Pinecone. https://www.pinecone.io/learn/series/image-search/imagenet/
- Tsang, S.-H. (2018, August 24). Review: GoogLeNet (Inception v1)—Winner of ILSVRC 2014 (Image Classification). Medium. https://medium.com/coinmonks/paper-review-of-googlenet-inception-v1-winner-of-ilsvlc-2014-image-classification-c2b3565a64e7
- GMI Cloud. (n.d.). ResNet. GMI Cloud. https://www.gmicloud.ai/glossary/resnet
- Hugging Face. (n.d.). MobileNet. Hugging Face. https://www.huggingface.co/learn/computer-vision-course/unit2/cnns/mobilenet
- Vina, A. (2025, August 29). What is EfficientNet? A quick overview. Ultralytics. https://www.ultralytics.com/blog/what-is-efficientnet-a-quick-overview