# Exploratory Architectures Analysis of Various Pre-trained Image Classification Models for Deep Learning

S. Deepa [1], J. Loveline Zeema [1], and S. Gokila [2,*]

[1] Department of Computer Science, Christ University, Bangalore, India
[2] Department of Computer Applications, Hindustan Institute of Technology and Science, Chennai, India
Email: sdeepa369@gmail.com (S.D.); j.lovelinezeema@gmail.com (J.L.Z.); sgokilas@gmail.com (S.G.)
*Corresponding author

***Abstract*—The image classification is one of the significant applications in the area of Deep Learning (DL) with respective to various sectors. Different types of neural network architectures are available to perform the image classification and each of which produces the different accuracy. The dataset and the features used are influence the outcome of the model. The research community is working towards the generalized model at least to the domain specific. On this gesture the contemporary survey of various Deep Learning models is identified using knowledge information management methods to move further to provide optimal architecture and also to generalized Deep Learning model to classify images narrow down to the sector specific. The study systematically presents the different types of architecture, its variants, layers and parameters used for each version of Deep Learning model. Domain specific applications and limitations of the type of architecture are detailed. It helps the researchers to select appropriate Deep Learning architecture for specific sector.**

***Keywords*—image classification, deep learning, neural network**

## I. Introduction

One of the most significant applications of Deep Learning (DL) and Artificial Intelligence (AI) is image classification. The process of labeling images based on specific characteristics or features is known as image classification. These characteristics are discovered by the algorithm, which makes use of them to classify and differentiate between various images [1].

The process of classifying images is known as image classification. This is accomplished by utilizing similar features found in images belonging to various classes to identify and label the images [2]. Neural networks are useful in categorizing images, and DL algorithms utilize them as the core mechanism. Essentially, DL algorithms are based on neural networks.

Prediction, classification, and other functions are carried out by the neurons in multiple layers of a neural network. Each neuron's output is sent to the neurons in the next layer, where it is tweaked until reaching the final output layer Fig. 1.

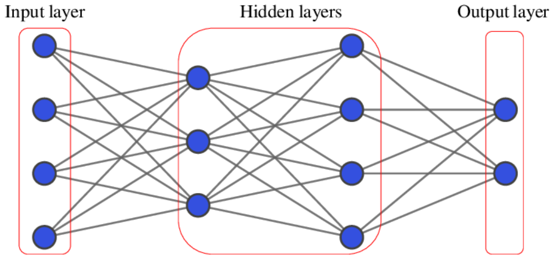

Fig. 1. Neural network.

In the neural network, the Initial data for the neural network is in the input layer. Hidden layers are between the input and output layers, which serve as the hub for all computing. The output layer generates the outcome given the inputs. A pre-trained machine learning model has already undergone extensive training on a large dataset to solve a similar problem. Using pre-trained models saves time and resources. Some available image classification models with their architecture and training methods are the VGG, ResNet [3], Inception, and MobileNet families. These models have resulted in exceptional performance in various image classification tasks, and they are extensively employed in DL frameworks such as Keras, TensorFlow, and PyTorch. Developers and researchers can use these frameworks to build and train their image classification models with ease.

Optimal Architecture: Identify key architectural elements and parameters for each Deep Learning model version to guide the development of generalized and sector-specific image classification models.

Domain-Specific Adaptation: Provide insights into domain-specific applications, limitations, and suitability of different DL architectures, aiding researchers in selecting the most suitable model for specific sectors.

Growing Importance of Image Classification: In modern civilization, image categorization is crucial in a variety of industries, including healthcare, banking, agriculture, and manufacturing. The efficiency of

Manuscript received May 8, 2023; revised July 11, 2023; accepted August 8, 2023; published January 18, 2024.

doi: 10.12720/jait.15.1.66-78

operations and decision-making are directly impacted by the capacity to classify photos effectively.

Expanding DL Model Landscape: There are numerous neural network topologies that display different accuracy, computational efficiency, and adaptability properties in the domain of DL. For academics and practitioners attempting to harness the power of DL for specific applications, navigating this complicated environment can be overwhelming.

Need for Sector-Specific Models: Generalised DL models provide a basis, but sector-specific models can considerably improve the effectiveness of picture categorization. Comparing a model created for autonomous vehicles to one optimised for medical image classification, for example, may necessitate different architectural considerations.

Lack of Consolidated Information: Current resources frequently offer shards of information on distinct DL models. By providing a systematic and comprehensive overview of several DL architectures, their features, applications, and limits, this article aims to fill this knowledge gap.

Future Research and Development Guidance: This article aims to assist researchers and developers in selecting or creating models by illuminating the subtleties of various DL architectures. By encouraging effective model selection and development procedures, this in turn advances the field.

The rest of the paper present as: Section II reviewed method of collecting the relevant research papers; Section III explained technical details of pre-trained CNN Architectures; Section IV reviewed research papers used the pre-trained architectures; Section V introduced data sets used to train and apply the pre trained CNN architectures; Section VI discussed the CNN architectures on different perspective which really support for next level of state-of-art proposal in developing the generalized domain specific CNN model to classify the images.

## II. Methodology for Acquring Knowledge

In order to conduct a thorough analysis of the literature, various databases including Google Scholar, Scopus, Springer, IEEE Xplore, and ArXiv are examined. The key terms used for the search are “DL”, “image classification” and also used the name of the architectures.

Thirty-five related papers were identified and removed the ten irrelevant papers by reading the abstracts and excluding the ones that did not meet our inclusion criteria as per Table I.

TABLE I. Knowledge Source

| Source of Article Collection | Number of Relevant Papers Identified |
|---|---|
| IEEE | 14 |
| Scopus | 5 |
| Google Scholar | 4 |
| ArXiv | 11 |
| Springer | 3 |

## III. Architecture of Pre-trained Models in Image Classification

### A. ResNet (Residual Networks)

Microsoft Research introduced Residual Network architecture with the proposal of ResNet [4]. The idea of residual blocks was introduced in this architecture to address the issue of the vanishing/exploding gradient. Skip connections are a method that are employed in this network Table II. By skipping some layers in between, the skip connection links a layer’s activations to other layers. This creates a block that remains. By stacking these residual blocks together, ResNets [5] are created.

TABLE II. Residual Network Models

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| ResNet50 | 50 | 23.6 | Simpler architecture, easier to optimize, better accuracy than previous models | Not suitable for small datasets |
| ResNet50V2 | 50 | 25.6 | Faster convergence, better accuracy than ResNet50 | Not suitable for small datasets |
| ResNet101 | 101 | 42.6 | Deeper architecture, better accuracy than ResNet50 | Slower training time |
| ResNet101V2 | 101 | 44.3 | Faster convergence, better accuracy than ResNet101 | Not suitable for small datasets |
| ResNet152 | 152 | 60.2 | Deeper architecture, better accuracy than ResNet101 | Slower training time |
| ResNet152V2 | 152 | 62.4 | Faster convergence, better accuracy than ResNet152 | Not suitable for small datasets |

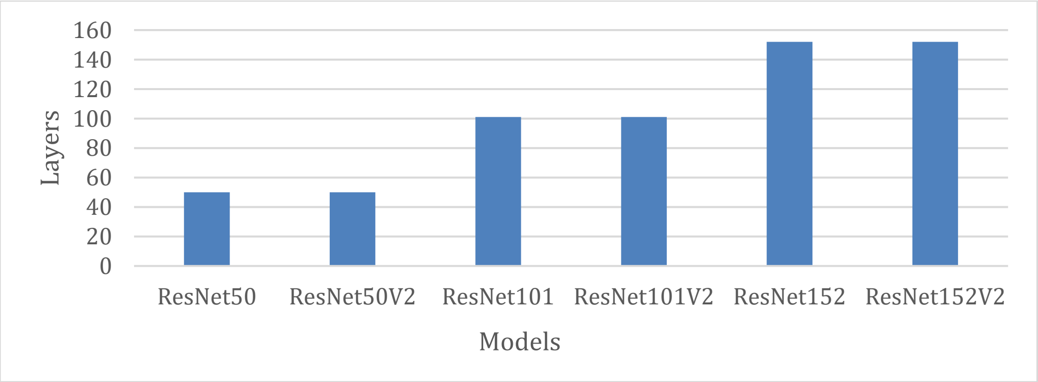

Fig. 2. Types of ResNet architecture vs layers.

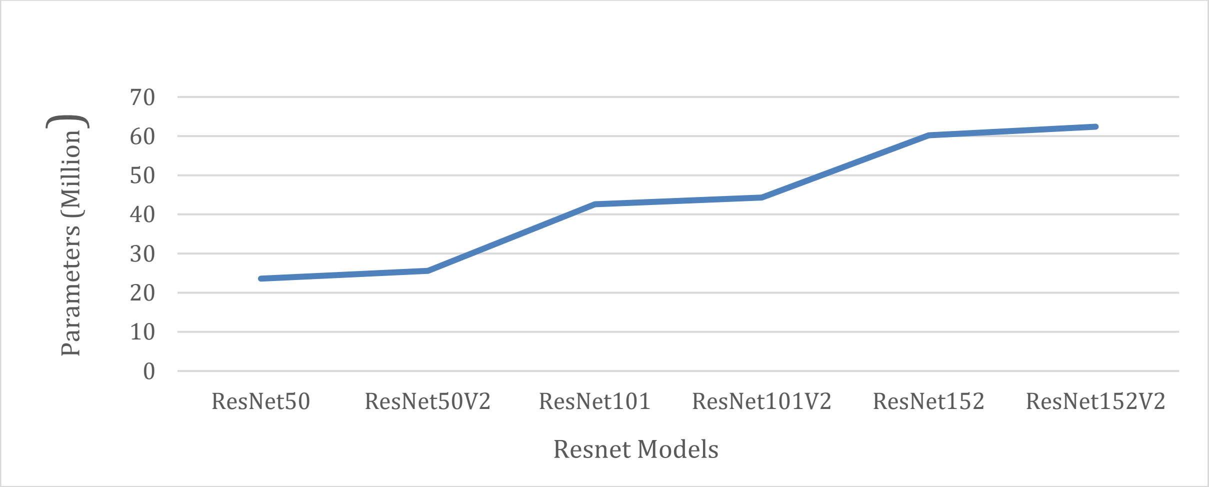

Fig. 3. Types of ResNet architecture vs number of parameters.

This network follows a strategy in which, as opposed to layers learning the underlying mapping and let the network fit the residual mapping. Each step of the model's five convolutional layers contains a different number of residual blocks Figs. 2 and 3. Final layer of fully connected layer is used for classification [3].

## B. *EfficientNet*

Convolutional Neural Network (CNN) architecture and scaling method EfficientNet [3] uses a compound coefficient to uniformly scale all width, depth and resolution dimensions Table III.

TABLE III. EFFICIENTNET MODELS

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| EfficientNetB0 | 7 | 5.3 | Light weight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetB1 | 10 | 7.8 | Light weight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetB2 | 19 | 9.2 | Light weight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetB3 | 25 | 12.3 | Light weight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetB4 | 28 | 19.3 | Light weight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetB5 | 40 | 30.6 | Lightweight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetB6 | 52 | 43.0 | Lightweight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetB7 | 66 | 66.3 | Lightweight, high accuracy, efficient | May not perform well on complex datasets with large images |
| EfficientNetV2B0 | 21 | 20.0 | Improved performance compared to EfficientNetB0 | May not perform well on complex datasets with large images |
| EfficientNetV2B1 | 27 | 30.0 | Improved performance compared to EfficientNetB1 | May not perform well on complex datasets with large images |
| EfficientNetV2B2 | 33 | 40.0 | Improved performance compared to EfficientNetB2 | May not perform well on |
| EfficientNetV2L | 20 | 480 | This model has a higher accuracy rate and can handle complex image datasets. It also provides better generalization performance and faster inference time. | The model requires a high computational cost due to its large size, and it may be difficult to train on low-end devices. |

In addition to squeeze-and-excitation blocks, the base EfficientNet-B0 network is built on MobileNetV2's inverted bottleneck residual blocks. EfficientNet is a family of CNN architectures that have been designed to achieve better performance with fewer parameters and less computational resources than other architectures Figs. 4 and 5.

The choice of which EfficientNet model to use depends on the specific task requirements, available computational resources, and accuracy needs. Overall, the EfficientNet family of models provides a range of options for DL practitioners to choose from based on their specific needs. The different variants of EfficientNet have different performances between accuracy and efficiency, making them suitable for different applications.

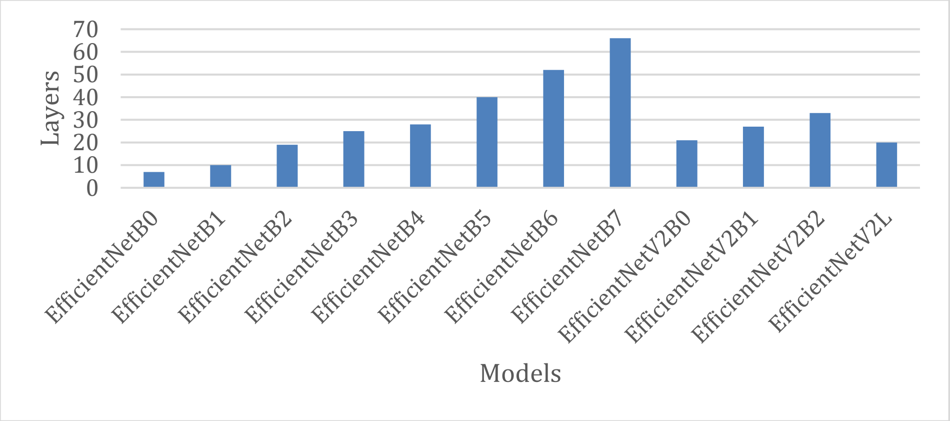

Fig. 4. Types of EfficientNet architecture vs layers.

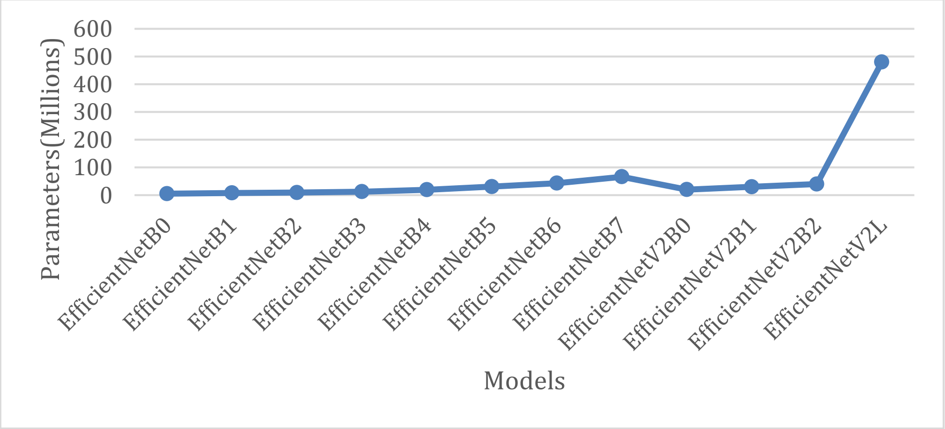

Fig. 5. Types of EfficientNet architecture vs number of parameters.

### C. *VGG16*

Very Deep Convolutional Networks for Large-Scale Image Recognition (VGG-16) [6], one of the most well-liked pre-trained image classification models, is the VGG-16. It was the model to compete when it was introduced at the ILSVRC 2014 Conference. VGG-16, developed by the University of Oxford's Visual Graphics Group, outperformed AlexNet at the time and was quickly adopted by researchers and industry for image classification tasks Table IV.

TABLE IV. VGG MODELS

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| VGG16 | 16 | 138 | Simple and easy to understand architecture | Large number of parameters, slow training time |
| VGG19 | 19 | 143 | Improved accuracy compared to VGG16 | Large number of parameters, slow training time |

VGG16 and VGG19 are deep CNNs with a simple architecture of 3×3 filters and max pooling layers. VGG16 has 16 layers and 138 million parameters, while VGG19 has 19 layers and 143 million parameters Fig. 6.

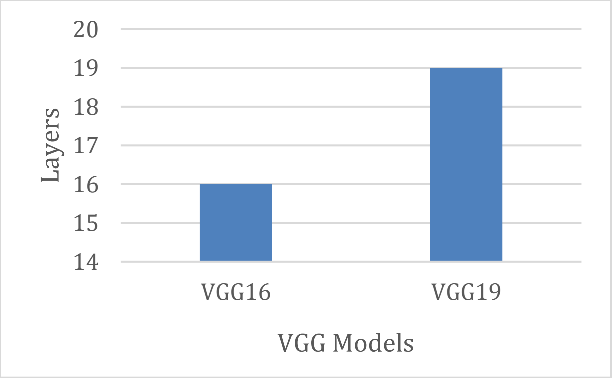

Fig. 6. Types of VGG architecture vs layers.

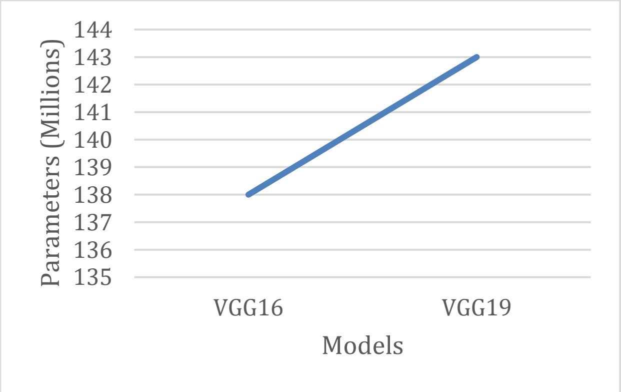

Fig. 7. Types of VGG architecture vs number of parameters.

These two models are appropriate for image classification, object detection, and also segmentation tasks. VGG models are simple and easy to understand architecture, but it also have a many number of parameters and it uses a slow training time Fig. 7. VGG19 has improved accuracy compared to VGG16, but both models are not suitable for small datasets or limited computing resources.

### D. *Xception*

Xception is a deep CNN with depth wise separable convolutions [7] that allow for a more efficient use of parameters and faster training than traditional convolutions Table V.

TABLE V. XCEPTION MODELS

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| Xception | 71 | 22.9 | High accuracy, faster training than Inception models | High computational cost |
| InceptionV3 | 48 | 23.85 | Good accuracy with fewer parameters, can be used for transfer learning, and can handle different input sizes. | Computationally expensive, can lead to over fitting, and requires careful tuning of hyper parameters |
| InceptionResNetV2 | 572 | 55.87 | High accuracy, fewer parameters than previous versions, robustness to adversarial examples, and can be used for transfer learning. | Computationally expensive, can lead to over fitting, and requires careful tuning of hyper parameters. |

It has 71 layers and 22.9 million parameters, making it suitable for image classification, object detection, and face recognition tasks. Xception has high accuracy and faster training than Inception models, but it also has a high computational cost.

### E. *ConvNeXt*

ConvNeXt is a notable advancement in CNN architecture that utilizes parallelized equivalent convolutional layers to improve accuracy and computational efficiency. It has gained significant attention and application in various computer vision tasks Table VI. ConvNeXt, a group of CNN architectures, shares many similarities with ResNeXt. “Aggregated Residual Transformations for Deep Neural Networks” is introduced in [8]. The primary concept behind ConvNeXt is to replace standard convolutional layers with a group of equivalent convolutional layers, which are then combined to generate the final output Fig. 8.

TABLE VI. CONVNEXT MODEL

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| ConvNeXtTiny | 10 | 0.6 | Small and fast with good accuracy, efficient on mobile devices | Limited depth, may not perform as well as larger models on complex datasets |
| ConvNeXtSmall | 14 | 1.3 | Small and fast with good accuracy, efficient on mobile devices | Limited depth, may not perform as well as larger models on complex datasets |
| ConvNeXtBase | 26 | 6.9 | Good balance of accuracy and speed, efficient on a range of devices | Larger than the MobileNet-based models, may be slower on some devices |
| ConvNeXtLarge | 38 | 22.3 | High accuracy, competitive with state-of-the-art models on a range of datasets | Larger than the MobileNet-based models, may be slower on some devices |
| ConvNeXtXLarge | 64 | 56.6 | Highest accuracy, competitive with state-of-the-art models on a range of datasets | Largest model in the series, requires more memory and processing power than the other models, may not be practical on some devices |

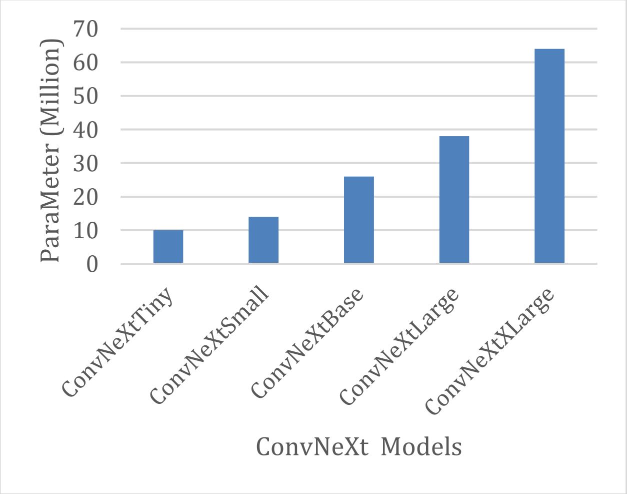

Fig. 8. Types of ConvNeXt architecture Vs Layers.

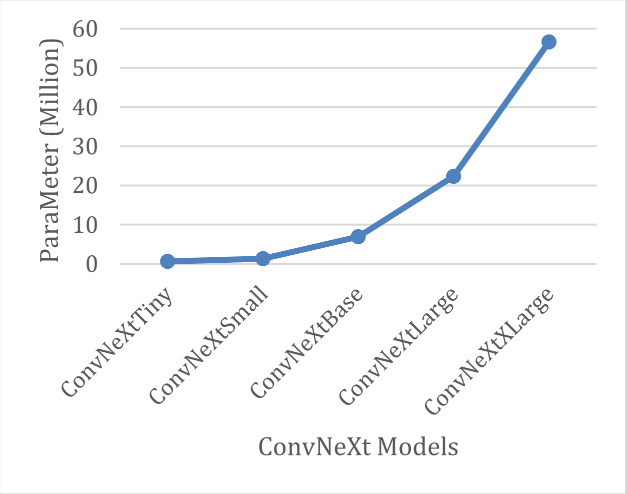

Fig. 9. Types of ConvNeXt architecture vs number of parameters.

Overall, the ConvNeXt family of models provides a range of options for DL practitioners to choose from based on their specific needs and making them suitable for different applications and use cases Fig. 9.

### *F. DenseNet (Densely Connected Convolutional Network)*

Densely Connected Convolutional Network (DenseNet) and Neural Architecture Search Network (NASNet) are two families of DL models that have achieved good performance on various computer vision tasks Table VII [9]. The varying parameter and layers are enhancing the performance Figs. 10 and 11.

TABLE VII. DENSENET MODELS

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| DenseNet121 | 121 | 8 | Efficient use of parameters, high accuracy on small and medium-sized datasets, strong feature reuse across layers | High memory consumption, slower training speed than other models |
| DenseNet169 | 169 | 14 | Strong feature reuse across layers, lower memory consumption than DenseNet201 and ResNet models | Slower training speed than other models |
| DenseNet201 | 201 | 20 | Strong feature reuse across layers, high accuracy on large datasets, efficient use of parameters | High memory consumption, slower training speed than other models |

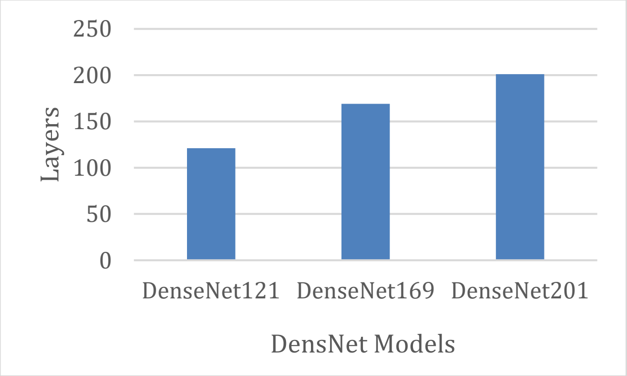

Fig. 10. Types of DenseNet architecture vs layers.

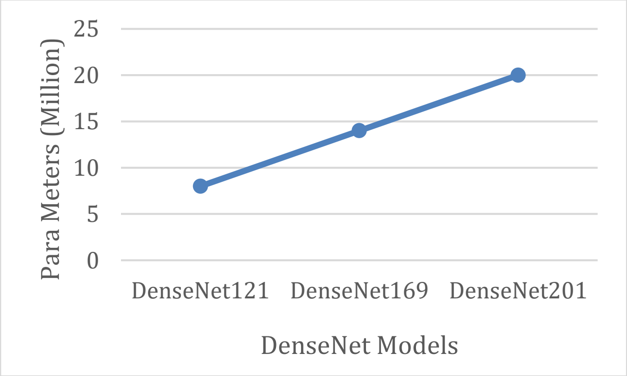

Fig. 11. Types of DenseNet architecture vs number of parameters.

### *G. MobileNet*

MobileNet is a family of DL models that were introduced by Google in 2017. MobileNet is a family of DL models, it was introduced by Google in 2017. The MobileNet [10, 11] architecture is mainly designed to be computationally efficient and also lightweight, making it suitable for mobile devices and other resource constrained environments Table VIII. MobileNet achieves this by using depth wise separable convolutions, which reduce the number of parameters in the model while still preserving accuracy Figs. 12 and 13.

TABLE VIII. MOBILENET MODELS

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| MobileNet | 28 | 4.2 | Small size, fast, low-latency, low-power, efficient on mobile devices, low memory footprint | Lower accuracy compared to larger models |
| MobileNetV2 | 53 | 3.4 | Small size, fast, low-latency, low-power, efficient on mobile devices, improved accuracy compared to MobileNet | Still lower accuracy compared to larger models, may require additional fine-tuning to improve accuracy on certain tasks |

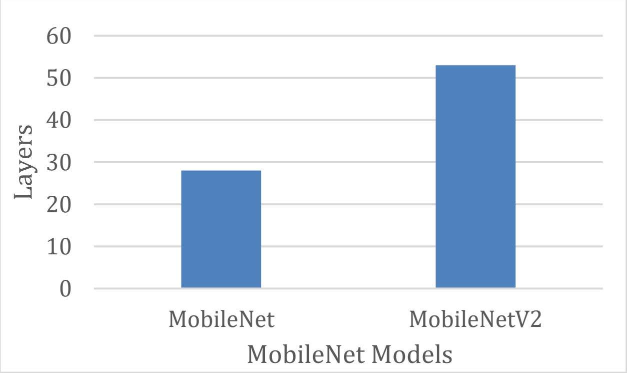

Fig. 12. Types of MobileNet architecture vs layers.

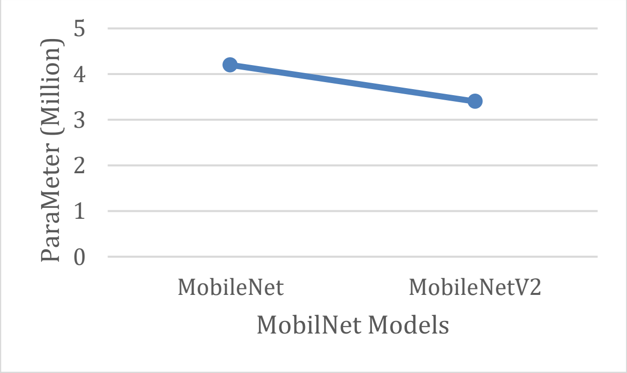

Fig. 13. Types of MobileNet architecture vs number of parameters.

TABLE IX. INCEPTION

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| InceptionV3 | 48 | 23.85 | Good accuracy with fewer parameters, can be used for transfer learning, and can handle different input sizes. | Computationally expensive, can lead to over fitting, and requires careful tuning of hyper parameters |
| InceptionResNetV2 | 572 | 55.87 | High accuracy, fewer parameters than previous versions, robustness to adversarial examples, and can be used for transfer learning. | Computationally expensive, can lead to over fitting, and requires careful tuning of hyper parameters |

## H. *Inception*

InceptionV3, InceptionResNetV2, MobileNet, and MobileNetV2 are the DL models and is broadly used in the area of computer vision Table IX. Each model has its specific architecture and characteristics, making them suitable for different use cases based on the layer and parameter change Figs. 14 and 15.

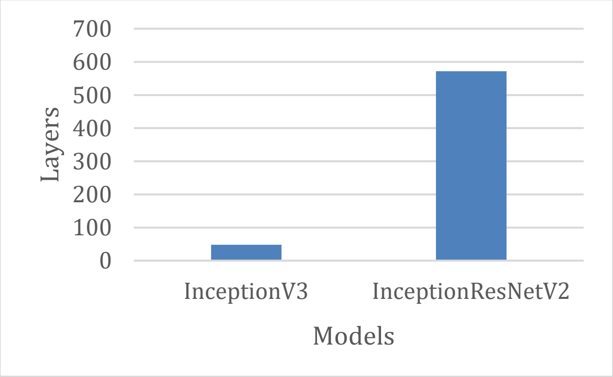

Fig. 14. Types of inception architecture vs layers.

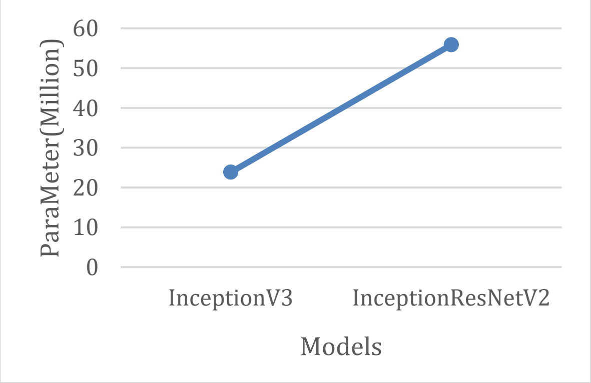

Fig. 15. Types of inception architecture vs number of parameters.

## I. *NASNetMobile*

NASNetMobile is a neural architecture search network that was introduced in "Learning Transferable Architectures for Scalable Image Recognition" [12]. It has 4.2 million parameters and is optimized for mobile devices Fig. 16.

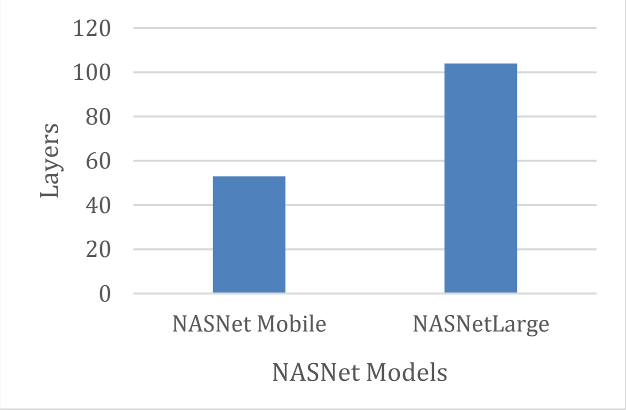

Fig. 16. Types of NASNetMobile architecture vs layers.

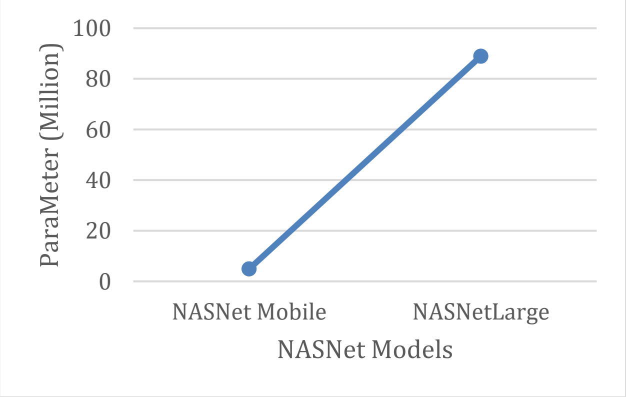

Fig. 17. Types of NASNetMobile architecture vs number of parameters.

NASNetMobile achieves better accuracy on the ImageNet dataset while being efficient enough to run on mobile devices Fig. 17. The architecture of NASNetMobile was discovered using a neural architecture search algorithm, automatically it searches the best architecture for a given task Table X.

TABLE X. NASNETMOBILE

| Model | Layers | Parameters (Million) | Advantages | Limitations |
|---|---|---|---|---|
| NASNet Mobile | 53 | 4.9 | High accuracy, lightweight, efficient on mobile devices | Large model size compared to other mobile models |
| NASNetLarge | 104 | 88.9 | State-of-the-art performance, modular architecture allows for customization, transfer learning | High computational and memory requirements, not suitable for mobile devices |

NASNetLarge is another neural architecture search network that has more parameters than NASNetMobile, with 88 million parameters. It achieves even higher accuracy than NASNetMobile on the ImageNet dataset. NASNetLarge is optimized for high-performance computing environments and is suitable for applications that require the highest accuracy, such as object detection and image segmentation. The architecture of NASNetLarge was also discovered using a neural architecture search algorithm

## IV. Pretrained Models-Argumentative Review

Literature review provides a summary of several popular DL architectures and their performance on various datasets. These architectures include VGG, ResNet, Inception, MobileNet, and ConvNeXt. The review also discusses the benefits and limitations of each architecture, providing valuable insights into their applications and potential use cases.

The Deep residual network architecture is designed to enable the training of very deep neural networks. The vanishing gradient problem, occurs because of small gradient, it cannot effectively propagate through the network, and it is difficult to train very deep networks [4]. To solve this problem residual connections are used architecture, it allows the network to learn the variance between the input as well as output of a particular layer, instead of directly learn the output. this method is shown to considerably improve the accuracy of the deep neural networks for image recognition tasks.

The concept of identity mappings is introduced along with residual connection. It states that the residual connections create additional complexity which makes the optimization process difficult and by allowing the network to learn identity mappings [5]. The residual method simply passes the input through a layer without making any changes, the authors show that it is possible to maintain simplify the architecture with improving performance.

The existing architectures affects from a transaction between depth and width in deeper networks, it requires more computation to achieve accuracy, for wider networks it can be faster with less accurate. The proposes a new architecture for CNNs and it is called as inception-v4 is proposed to achieve high accuracy [1]. The inception-v4 architecture is designed to address the problem of combining both depth and width in an integrated way. The article also introduced the new features, like factorized 7×7 convolutions and an auxiliary classification, which helps to improve the performance on image classification tasks.

The inception-v4 and inception-ResNet models which are the deep CNNs constructed on inception module [2]. The Inception-v4 architecture, which is an extension of the Inception architecture with added residual connections [12]. Residual connections are added in the inspectional ResNet between the inception modules and it helps for the gradient flow problem. This study stats that adding of the residual connections improves the performance of the model on the ImageNet classification task.

The MobileNets architecture is mainly designed for efficient inference on mobile and also embedded devices [10]. This model uses depth-wise separable convolution layers and it helps to reduce the number of parameters and also procedures required for inference. the authors also says that the MobileNet architecture is computationally more efficient and it can achieve good accuracy for image classification as well as object detection.

The MobileNetv2 architecture which is an extended version of the MobileNet architecture [11]. This architecture uses inverted residual blocks with linear bottlenecks for improvin the accuracy of the model and it maintains the computational effectiveness of the original MobileNet. The study shows that MobileNetv2 accomplishes better accuracy on the ImageNet classification.

The Densely Connected Convolutional Networks (DenseNet) architecture built on dense blocks in its place of traditional convolutional layers [9, 13]. In a dense block, all layer receives feature maps from all previous layers and also passes its own feature maps to its all of the subsequent layers. The DenseNet gives the good accuracy on the ImageNet classification, compare to more parameter-efficient deep CNNs.

The Condensed CNNs (CondenseNets) architecture designed for more parameter-efficient compare to other deep CNNs [14]. This work is a combination of dense blocks with skip connections for decreasing the number of parameters necessary for the model with the same accuracy. The CondenseNets is having significantly fewer parameters than other deep CNNs and also gives the better results for image classification.

The purpose Multi-Scale Dense Networks (MSDNet) architecture is to attain more computationally efficient than other deep CNNs [15]. The architecture uses a mixture of dense blocks with various growth rates and it helps to enable multi-scale feature learning. The MSDNet is also more suitable e Image classification. The CNN architecture based on the reinforcement learning which is automatically searching for neural network parameter is proposed in [16]. The study shows that the method is capable to learn novel neural network architectures which is suitable for image classification and object detection.

The EfficientNet is a combination of model extraction, neural architecture search as well as conditional computation to yield a family of EfficientNet models that are very small and faster than the actual models. Even Smaller method decreasing the dimension of EfficientNet models with accuracy [17].

A novel attention mechanism for semantic segmentation that helps for improvement of the accuracy of existing methods using less computational cost [18]. The channel attention module is easily incorporated into existing segmentation networks and it illustrates the efficiency on several benchmark datasets. Tensor decompositions is used to find the important components of the architecture and shows how it is use full for optimization the architecture for precise tasks. It is used to investigating the architecture [19].

A family of mobile-friendly CNN architectures called MobileNets are designed for resource-constrained devices. The depth wise separable convolutions, which allow for a more reduction in the number of parameters and FLOPS required for mobile vision tasks [20]. Inverted residual blocks and linear bottlenecks is included in a new version of the MobileNet architecture which improves accuracy and efficiency over the original MobileNet [21]. The inverted residual blocks and linear bottlenecks, supports for the efficient use of computational resources and better performance on small devices.

A temporally-shifted attention mechanism that reduced the number of parameters and FLOPS is applied in a light-weight version of the Vision Transformer (ViT) architecture for video recognition. Due to the reduced number of parameters and FLOPS required for video recognition tasks simultaneously maintaining accuracy [22]. A new model scaling method for CNNs which is using less parameters and FLOPS associated to all the existing models is used. The compound scaling method that is used to optimizes the scaling of depth, width, and resolution of the network concurrently [23].

The convergence and acceleration properties of Residual Neural Network (ResNet) architectures became popular for image classification tasks. The improved of ResNet architecture that accelerates the convergence and reduces the number of parameters while maintaining accuracy. The proposed modification involves replacing the identity shortcut connections with a linear mapping that is learned during training [6]. The enhanced of ResNet architecture was evaluated on several benchmark datasets and demonstrate improved convergence and performance compared to the standard ResNet architecture.

A new family of neural network architectures, called EfficientNetV2 that are designed to be smaller and faster than the original EfficientNet architecture while maintaining accuracy [3]. The high accuracy is achieved by introducing a new compound scaling method that jointly scales the width, depth and resolution of the network in a balanced way. They also introduce a new training technique, called Stochastic Depth, that improves training speed by randomly dropping network layers during training. The model is evaluated on numerous benchmark datasets and presented with fewer parameters and less computation than previous models.

The architecture of VGGNet, a deep CNN that achieves better performance on the ImageNet dataset. The network consists of a series of convolutional layers with small filters and max-pooling layers, followed by several fully connected layers [24]. The implementation investigates the effect of increasing the depth of the network on performance and find that deeper networks perform better, but with diminishing returns beyond a certain depth. They also compare their network to other state-of-the-art networks and demonstrate superior performance.

The AlexNet involves of many convolutional layers with small filters, max-pooling layers, and many fully connected layers. The AlexNet architecture is deep CNN based model performances well on the ImageNet dataset. Several techniques are used to regularize the network and it prevents from overfitting, it includes dropout and data augmentation. Rectified Linear Units (ReLU) as the activation function, used for improving the performance, compare to traditional activation functions [8, 25]. The efficiency is evaluated using the ImageNet dataset, reaching a top-5 error rate of 15.3%, it is a major improvement compare to existing results.

The deep CNN, VGG-16 architecture achieves better performance on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [26]. This architecture consists of 16 layers, with small 3x3 filters as well as max pooling layers. The proposed method also experimented the effect of depth on the network’s performance and it shows that increasing the depth of the network reaches the improved performance.

The GoogLeNet architecture and it also known as Inception v1 consists of multiple inception modules. It was designed for efficient factorization of the convolutional filters into smaller ones. The auxiliary classifiers of GoogleNet are used to improve the convergence of the network in training. The GoogLeNet architecture is attained best performance on the ILSVRC 2014 challenge [7].

FaceNet is a deep neural network architecture for face recognition and clustering that learns a high-dimensional embedding for each face image [27]. A triplet loss function that encourages the embeddings of images of the same identity to be closer together than embeddings of images of different identities. The FaceNet architecture consists of a deep CNN followed by a fully connected layer that produces the embedding. FaceNet acquires a mapping function starts the high-dimensional space of face images it moves to a low-dimensional embedding space, the distances between the embeddings are correspond to the similarities between the faces This is achieved by training a Deep Neural Network (DNN) on a large-scale face dataset (over 200 million images), using a triplet loss function that encourages the embeddings of the same face to be close together and those of different faces to be far apart. The method can also be used for face clustering and face verification tasks.

The deep CNN architecture was proposed for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) and achieved a top-5 level test error of 15.3% It has 8 layers, along with large 11×11 and 5×5 filters, and the activation function used is Rectified Linear Units (ReLU) [28]. The work illustrated with an 8-layer neural network and it consist of the convolutional and also fully connected layers, with 60 million parameters of a dataset with 1.2 million images from 1000 categories. The main support of this work, it has used Rectified Linear Units (ReLU) as activation functions, and it always allows faster training and better performance compared to traditional activation functions like sigmoid and hyperbolic tangent. The achievement of this architecture generated the extensive use of deep CNNs for image classification and other computer vision tasks.

The deconvolution networks are used to visualize and understand the features. It also helps to activate the specific features in the network. The visualization shows that generating the saliency maps that highlight the main regions of an image and it contribute to the network’s classification decision [29].

A backpropagation and the gradient of the output class score with respect to the input image and it helps to highlight the regions of the image that contributed the most to the classification decision. The same is introduced for visualizing the internal representations of deep CNNs and generating saliency maps to highlight the regions of an input image that are important for classification [30]. They also established a method for visualizing the activations of separate neurons in the network and some of these neurons are responded to advanced semantic concepts like faces and text.

The Wide Residual Networks (WRNs) is significantly improved the performance of deep neural networks on a wide range of image classification tasks [31]. The proposed work shows that by increasing the width of the residual blocks in a deep network, they could achieve better, on the ImageNet dataset while using fewer parameters than previous methods. Experimental results shows that the WRN architecture is robust to solve overfitting problem and can simplify better for new datasets.

The ImageNet Large Scale Visual Recognition is a competition for object detection and classification tasks using deep neural networks [32]. The dataset consists of over twelve lakh high-resolution images and covers thousand object categories are used to analysis and solve the challenges. It also evaluated and compared the performance of different deep neural network architectures on the dataset.

Aggregated Residual Transformations (ART) is new deep neural network architecture, which aims to improve the performance of deep neural networks with reduced computational cost [33]. It is achieved by using residual connections and aggregating the predictions of multiple ART modules within the same network. The experimental results show on various image classification tasks proves that ART achieves more computationally efficient than previous deep neural network architectures.

A CNN architecture and it is designed for mobile devices with less computational resources are proposed in [34]. The work introduced ShuffleNet unit and it substitutes the standard convolutional layer in the network. The ShuffleNet unit consists of three main parts that is pointwise group convolution, channel shuffle, and depth wise convolution. The pointwise group convolution is used for reducing the number of input channels, the channel shuffle is used for randomly shuffle the output channels from different groups and the depth wise convolution is used to accomplish the actual convolutional operation. The ShuffleNet architecture achieves better accuracy on the ImageNet dataset.

ShuffleNet V2 is an improved version of the ShuffleNet architecture [34, 35]. The main contributions of the ShuffleNet V2 is channel shuffle operation and a set of procedures for designing effective CNN architectures. The new channel shuffle operation is designed to improve the efficiency of the network by reducing the communication overhead between different groups of channels. The architecture shows that the channel shuffle operation can be used to replace the standard concatenation operation used in residual networks, leading to significant improvements in both accuracy and computational efficiency. The procedures proposed are based on the idea of “pruning and splitting”, where the network is first pruned to remove redundant connections and then split into smaller sub-networks and it leads to significant improvements in both accuracy and computational efficiency, also reducing the complexity of the architecture design process. It is mainly used for mobile devices with limited computational resources.

The DenseNet-169 mode is used for detecting COVID-19 patients from chest X-ray images [36]. Nearest-neighbour interpolation technique was used for data preprocessing and Adam Optimizer was used for optimization and accuracy achieved is 96.37%.

The inverted-bell-curve-based ensemble of DL (or CNN) models are developed for the detection of COVID-19 from CXR images [37]. The existing transfer learning, pretrained weights are not enough for COVID-19 CXR images. The combination of VGG16, ResNet18 and Densenet161 models are used to train on the available data and confidence score level is combined and used in the proposed ensemble method. It considers a classifier predicts the correct class with a high score value and identifies a wrong class using less score value. The work shows that the ensemble method producing superior results.

The purpose of MobileNet architecture is that it uses depth wise separate convolutions [38], it reduces the parameter count compared to the existing models with fixed convolutions. Diving the convolution into 1X1 point wise and 3×3 in depth wise is the novel approach in MobileNet. The classification function ReLU achieves an accuracy of 96.22%. This study presents a system for medical image categorization and Alzheimer’s ailment recognition. Alzheimer’s disease has five stages.

## V. Dataset Used to Train and Test

The authors handled various data set in each model. The data set is another deciding factor of model accuracy. The researcher working towards the generalization of model has focus more on data set also. The reviews summary about the details of various data sets used in model training and the accuracy attained are mentioned in Table XI.

The accuracies are depending on the implementation, hyper parameters, and other experimental settings.

There are various architectures used for DL in image classification. VGG, Inception [12], and ResNet [2, 26] are some of the widely studied models. VGG is a deep CNN that has 19 layers, and it was the winner of the ImageNet Challenge 2014. Inception is a DL architecture that uses multiple filters with different kernel sizes. It has been used in various applications such as image and speech recognition. ResNet [6] is a deep neural network that uses residual blocks to improve the accuracy of the model. It has been widely used in various applications such as object recognition and detection.

TABLE XI. DATA SETS USED IN VARIOUS MODELS

| Dataset | Model | Accuracy |
|---|---|---|
| Caltech-101 | AlexNet | 90.40% |
| Caltech-256 | AlexNet | 57.00% |
| Fashion-MNIST | CNN | 92.50% |
| QuickDraw | CNN | 75.00% |
| FER2013 | CNN | 71.50% |
| Chest X-Ray | DenseNet-121 | 90.40% |
| MURA | DenseNet-169 | 90.50% |
| LFW | FaceNet | 99.63% |
| PASCAL VOC 2012 | FCN | 79.70% |
| SVHN | GoogleNet | 96.80% |
| DTD | GoogLeNet | 47.30% |
| CUB-200-2011 | GoogLeNet | 77.60% |
| Street View House Numbers (SVHN) | GoogLeNet | 96.30% |
| Stanford Dogs | GoogLeNet | 84.90% |
| SUN397 | GoogLeNet | 63.60% |
| Places205 | GoogLeNet | 50.30% |
| Open Images | Inception-v3 | 78.90% |
| Google Landmarks | Inception-v3 | 82.60% |
| Dog Breed Identification (Dogs vs Cats) | Inception-v4 | 93.80% |
| INaturalist | ResNet-101 | 85.40% |
| PASCAL-50S | ResNet-101 | 86.20% |
| Tiny ImageNet | ResNet-50 | 77.50% |
| EuroSAT | ResNet-50 | 98.60% |
| Oxford Flowers | VGG-16 | 98.00% |
| Oxford Pets | VGG-16 | 94.30% |
| Food-101 | VGG-16 | 82.40% |
| UC Merced Land Use | VGG-16 | 89.20% |
| Describable Textures | VGG-16 | 82.70% |
| MIT Indoors | VGG-16 | 70.40% |

These models were trained with ImageNet, CIFAR-10, and CIFAR-100 data sets. CIFAR-10 has 60,000 images with 10 classes, CIFAR-100 has 60,000 images with 100 classes, and the ImageNet data set consist 1.2 million images with 1000 classes. These data sets collected from academic and research institutions. It has a few thousand to more than a million images.

The data set used and the number of layers in the model influenced how accurate these models were. On the ImageNet data set, VGG accuracy is 92.7% while Inception had an accuracy of 95.2%. On the ImageNet data set, ResNet achieved an accuracy of 96.54% [2].

## VI. DISCUSSION ON ARGUMENTATIVE REVIEW

In this section, discusses the different algorithms and models used in DL and also its application in image classification. Also discussed the different data set used, the source, and the volume of data, along with the accuracy achieved by these models. The final layer of a DL model usually contains a classification function, such as SoftMax or sigmoid, that maps the model's outputs to a probability distribution over the possible classes. The specific function used depends on the nature of the classification task.

### *A. Accuracy Attained in Various DL Models Based on Analysis*

Recent advancements in DL and computer vision algorithms have greatly improved the accuracy of image recognition tasks. The development of more effective and efficient DL architectures has played a significant role in this progress.

These architectures have been designed to extract more meaningful features from input data, resulting in better accuracy in image recognition tasks. The emergence of pre-trained models has also made it easier for researchers to develop and deploy new models, leading to further improvements in performance.

Overall, the advancements in DL architectures have opened up new possibilities for the field of computer vision, allowing for more accurate and efficient image recognition. With the continued development of DL algorithms and architectures, even the greater progress can be expected in the future.

Table XII gives a review of many different DL architectures and their accuracies on the ImageNet dataset. It shows that few recent architectures, such as EfficientNet and ConvNeXt [32], be likely to attain higher accuracies than the previous architectures such as ResNet and Inception. The highest is achieved by EfficientNetB7 [24] at 84.4%, using the ImageNet dataset. However, it is significant to note that the accuracies are not directly equivalent across different architectures, as they may have been trained and evaluated using different methods and settings.

Therefore, the optimal of architecture for a particular application should be based on different factors such as computational resources, model complexity, and task requirements.

TABLE XII. ACCURACY ATTAINED IN VARIOUS DL MODELS

| Architecture | Accuracy |
|---|---|
| ResNet101V2 [4] | 80.40% |
| ResNet152 [4] | 78.10% |
| ResNet152V2 [3,5] | 80.30% |
| InceptionV3 [1] | 78.80% |
| InceptionResNetV2 [2] | 80.40% |
| MobileNet [10] | 70.60% |
| MobileNetV2 [11] | 71.80% |
| DenseNet121 [9] | 74.90% |
| DenseNet169 | 76.00% |
| DenseNet201 | 77.30% |
| NASNetMobile [14] | 74.00% |
| NASNetLarge | 82.70% |
| EfficientNetB0 [23] | 77.30% |
| EfficientNetB1 | 79.10% |
| EfficientNetB2 | 80.00% |
| EfficientNetB3 | 81.10% |
| EfficientNetB4 | 82.60% |
| EfficientNetB5 | 83.30% |
| EfficientNetB6 | 84.00% |
| EfficientNetB7 | 78.00% |
| EfficientNetV2B0 [3] | 79.00% |
| EfficientNetV2B1 | 80.30% |
| EfficientNetV2B2 | 81.10% |
| EfficientNetV2B3 | 82.20% |
| EfficientNetV2S [24] | 84.30% |
| EfficientNetV2M | 84.90% |
| EfficientNetV2L | 85.50% |
| ConvNeXtTiny [8, 18] | 73.60% |
| ConvNeXtSmall | 78.50% |
| ConvNeXtBase | 79.80% |
| ConvNeXtLarge | 80.80% |
| ConvNeXtXLarge | 81.20% |

### *B. Criteria for Model Fixation*

This section, discusses about the criteria used in many papers to fix the model, the reasons behind the high accuracy achieved, the unique field in which the models are used, and the domains in which they have been used the most.

The performance of the models is evaluated using Top1 accuracy and it measures the percentage of images for which the predicted class is the same as the ground class. However, some papers also use top-5 accuracy, which measures the percentage of images for which the predicted class is among the top-5 predicted classes [19]. Some other criteria used in many papers to fix the model were the number of layers, the learning rate, the batch size, and the optimizer. The number of layers impact the accuracy of the model. Too many layers can lead to over fitting, and too few layers result in under fitting. The learning rate can be defined the percentage of weights of the model are used in training. The batch size gives the details of number of samples used in each iteration, and the optimizer is used to optimize the loss function. The architectures focused on reducing the number of parameter and improve the scaling [22, 23]. The another criterial focused is FLOPS, which was aimed to reduce. That supports in using the model in handheld devices [20].

### C. Cause of High Accuracy

The papers that produced high accuracy due to various techniques it includes data augmentation, transfer learning, and regularization [8] used for implementation. Data augmentation mainly used for creating new training data by applying random transformations to the existing data. Transfer learning is responsible for a pre-trained model on a large dataset as a starting point and fine-tuning it on a smaller dataset. Regularization helps adding constraints to the model to prevent over fitting. The model with more layers will not produce always higher accuracy. The performance of the model depends on many different factors such as the architecture, the dataset, and the optimization algorithm.

The deeper models be likely to capture more complex features, which helps to improve the accuracy. The model with more layers produced higher accuracy, but the limitation is many layers can be a source of over fitting. The optimal number of layers varies depending on the data set. The learning rate and the batch size also have a great impact on the accuracy of the model.

### D. Domains Applied

The models applied in the healthcare, autonomous driving, and agriculture. In healthcare, these models have been used for cancer detection, brain tumor detection, CT SCAN of COVID-19 and diagnosis. In autonomous driving, these models are used for object detection and tracking. In agriculture, these models have been used for crop classification and yield estimation. The domains in which these models have been used the most are computer vision, image classification, and object recognition. The initial learning rates, and learning saturation levels of different models for various applications is in Table XIII.

These parameters can also vary based on the convergence speed of selected model and the specific problem.

TABLE XIII. LEARNING RATE OF MODEL WITH EPOCH IN VARIOUS APPLICATIONS

| Application | Transfer Model | Initial Learning Rate | Learning Saturation Level | Number of Epochs |
|---|---|---|---|---|
| Image Classification | VGG16 | 0.001 | 0.0001 | 40 |
| | ResNet50 | 0.01 | 0.0005 | 30 |
| | DenseNet | 0.0005 | 0.00005 | 50 |
| | EfficientNet | 0.001 | 0.0001 | 50 |
| | Xception | 0.001 | 0.0001 | 40 |
| | ConvNet | 0.0002 | 0.00002 | 60 |
| Object Detection | ResNet50 | 0.01 | 0.0005 | 30 |
| | Xception | 0.01 | 0.0005 | 40 |
| Semantic Segmentation | U-Net | 0.0001 | 0.00001 | 80 |
| | EfficientNet | 0.0002 | 0.00002 | 60 |
| Speech Recognition | ResNet50 | 0.0005 | 0.0001 | 50 |
| | Xception | 0.0005 | 0.0001 | 40 |
| Sentiment Analysis | VGG16 | 0.0002 | 0.00002 | 60 |
| | EfficientNet | 0.0002 | 0.00002 | 60 |

## VII. CONCLUSION

This study's scope included a thorough investigation of pre-trained deep learning models in a variety of applications. These models are priceless resources that speed up training procedures and provide the basic building blocks for the creation of complex models. The ability to customize model selection to individual use cases enables programmers to create unique deep learning applications more quickly. These results make it clear that the current study plays a crucial function in assisting both beginning and experienced developers. This study accelerates the creation of deep learning applications by incorporating a wide variety of pre-trained models. The subtle conclusions drawn from our research provide weight to the dynamic interaction between model architectures and application environments.

## CONFLICT OF INTEREST

The authors declare no conflict of interest.

## AUTHOR CONTRIBUTIONS

Core concept and design of study: S.D., J.L.Z., and S.G. Knowledge acquiescing and filter out the relevant information: S.D. and J.L.Z. Collecting and presenting the architecture of pre trained image processing models: S.D., J.L.Z., and S.G. Argumentative review of pre trained image processing models: J.L.Z. and S.G. Wide discussion of each models presented with respected to various criteria: S.D. and S.G. Draft manuscript preparation: S.D., J.L.Z., and S.G. Review of entire structure of work and prepare the final version of manuscript: S.D., J.L.Z. and S.G. All authors had approved the final version.

## REFERENCES

[1] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[2] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. AAAI*, 2017, pp. 4278–4284.

[3] X. Li and L. Chen, "EfficientNetV2: Smaller models and faster training," arXiv preprint, arXiv:2104.00298, 2021.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. European Conference on Computer Vision*, 2017, pp. 630–645.

[6] S. Beyramysoltan and R. Klette, "On the convergence and acceleration of ResNet architectures," *Journal of Mathematical Imaging and Vision*, vol. 63, no. 3, pp. 387–407, 2021.

[7] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, and A. Rabinovich, "GoogLeNet: Going deeper with convolutions," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "AlexNet: ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[9] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.

[10] M. Andreetto and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint, arXiv:1704.04861, 2017.

[11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[12] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[13] P. N. Srinivasu, J. Shafi, T. B. Krishna, C. N. Sujatha, S. P. Praveen, and M. F. Ijaz, "Using recurrent neural networks for predicting type-2 diabetes from genomic and tabular data," *Diagnostics*, vol. 12, no. 12, 3067, 2022.

[14] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Condensed convolutional neural networks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6838–6846.

[15] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Multi-scale dense networks for resource efficient image classification," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3468–3476.

[16] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," arXiv preprint, arXiv:1611.01578, 2017.

[17] M. Tan and Q. V. Le, "Making EfficientNet even smaller," arXiv preprint, arXiv:2104.00298, 2021.

[18] S. Gao, Z. Li, Y. Li, H. Liu, J. Han, and Y. Sun, "Channel attention is all you need for semantic segmentation," arXiv preprint, arXiv:2102.11550, 2021.

[19] Y. A. B. Cardoso, T. R. Azevedo, D. S. D. Souza, and L. S. Oliveira, "Understanding the inception architecture with tensor decompositions," arXiv preprint, arXiv:2103.08474, 2021.

[20] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint, arXiv:1704.04861, 2017.

[21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[22] S. Targ, A. R. Zamir, M. A. Shahar, and A. Makhzani, "MobileViT: Light-weight, temporally-shifted ViT for efficient video recognition," arXiv preprint, arXiv:2011.09094, 2020.

[23] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. International Conference on Machine Learning*, 2019, pp. 6105–6114.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint, arXiv:1409.1556, 2015.

[25] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. 2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[26] C. Bai, L. Huang, X. Pan, J. Zheng, and S. Chen, "Optimization of deep convolutional neural network for large scale image retrieval," *Neurocomputing*, vol. 303, pp. 60–67, 2018.

[27] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

[28] A. Rizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[29] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. European Conference on Computer Vision*, 2014, pp. 818–833.

[30] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," arXiv preprint, arXiv:1312.6034, 2014.

[31] S. Zagoruyko and N. Komodakis, "Wide residual networks," in *Proc. the British Machine Vision Conference (BMVC)*, 2016, pp. 1–12.

[32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, and A. C. Berg, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[33] K. Zhang, L. Tan, Z. Li, and Y. Qiao, "Aggregated residual transformations for deep neural networks," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[34] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6848–6856.

[35] N. Ma, X. Zhang, H. T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. the European Conference on Computer Vision (ECCV)*, 2018, pp. 116–131.

[36] P. P. Dalvi, D. R. Edla, and B. R. Purushothama, "Diagnosis of coronavirus disease from chest x-ray images using DenseNet-169 architecture," *Sn. Comput. Sci.*, vol. 4, 214, 2023.

[37] A. Paul, A. Basu, M. Mahmud *et al*., "Inverted bell-curve-based ensemble of deep learning models for detection of COVID-19 from chest X-rays," *Neural Comput. & Applic.*, 2022.

[38] C. Ouchicha, O. Ammor, and M. Meknassi, "A novel deep convolutional neural network model for Alzheimer's disease classification using brain MRI," *Autom. Control Compu. Sci.*, vol. 56, pp. 261–271, 2022.