International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

An Analytical Approach on Various Deep Learning Models for Image Classification

Roshani Raut[1*] , Sonali Patil[2] , Rudraksh Naik[3*], Pradnya Borkar[4], Dhirajkumar Lal[5]

Submitted: 24/03/2023 **Revised**: 27/05/2023 **Accepted**: 12/06/2023

Abstract: Image classification is a fundamental computer vision task that is essential to many applications, including autonomous driving, object detection, and medical diagnostics. This paper presents a comprehensive study on image classification techniques, focusing on deep learning models. We review and analyze prominent architectures, including AlexNet, VGGNet, ResNet, and evaluate their performance on benchmark datasets such as CIFAR-10. Experimental results demonstrate the effectiveness of these models in achieving high accuracy and robustness in image classification tasks. Furthermore, we delve into the training process, hyperparameter tuning, and regularization techniques to optimize the performance of these models.

Keywords: *CNN, Image Classification, AlexNet, VGGNet, ResNet, CIFAR-10*

1. Introduction

Image classification is the process of categorizing or labeling images into different predefined classes or categories. In computer vision, image classification is used in various applications such as object recognition, facial recognition, content-based image retrieval, medical imaging, and autonomous vehicles.

The goal of image classification is to develop a model or algorithm that can automatically analyze the visual features of an image and assign it to one or more predefined classes. This is typically achieved through machine learning techniques, specifically using supervised learning.

The ability of deep learning convolutional neural networks (CNNs) to automatically learn hierarchical features from raw pixel data has made CNNs the most popular method for classifying images. Architectures like AlexNet, VGGNet, GoogLeNet, and ResNet have achieved significant success in various image classification tasks.

2. Literature Survey

2.1 AlexNet

In the realm of computer vision, AlexNet is a deep convolutional neural network with a sophisticated design that has made major contributions. Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton were the ones that first presented it in the year 2012. It competed in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), and it did quite well in that competition.

The architecture incorporates LRN layers after certain convolutional layers. LRN promotes competition among neighboring features, enhancing the model's ability to generalize and improving performance.

Dropout, a form of regularization, is applied to fully connected layers in AlexNet to prevent overfitting. It randomly sets a fraction of the layer's output to zero during training, forcing the network to learn more robust features.

AlexNet's success was partly due to its parallelization on GPUs, which significantly sped up the training process. It highlighted the possibilities of hardware-accelerated deep learning techniques.

AlexNet has eight layers, including three fully connected layers, five convolutional layers, and max-pooling layers. It is a sophisticated and expressive model with approximately 60 million parameters.

[1,2] Department of Information Technology, Pimpri Chinchwad College of Engineering, Pune, MS ,India [3]Department of Computer Engineering ,Pimpri Chinchwad College of Engineering,Pune, MS ,India

[4]Department of Computer Science and Engineering, Symbiosis Institute of Technology, Nagpur , Symbiosis International (Deemed University) MS ,India

[5]Department of Civil Engineering, Pimpri Chinchwad College of Engineering,Pune, MS ,India

** Corresponding author's Email: rudrakshnaik8990@gmail.com roshani.raut@pccoepune.org*

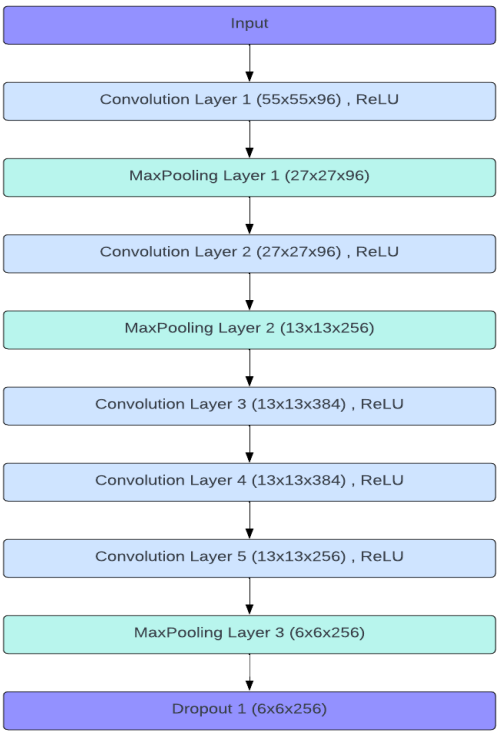


Fig 1: Architecture of AlexNet

The high-level overview of the architecture (Refer Figure 1) :

1. Input Layer: The network accepts a 227x227 pixel picture with three RGB color channels as input.

2. Convolutional Layers: The first convolutional layer contains 96 filters with a stride of 4 pixels and a size of 11x11. Following local response normalisation (LRN) and max pooling layers with a pool size of 3x3 and a stride of 2 pixels, rectified linear unit (ReLU) activation is used. Different filter sizes are used in the successive convolutional layers: the second layer has 256 5x5 filters, and the third, fourth, and fifth layers each have 384 3x3 filters. ReLU activation and max pooling are the layers that follow after each convolutional layer.

3. Fully Connected Layers: After the output of the most recent convolutional layer has been flattened, it is then fed into three layers that are fully linked. Following the 4096 neurons that make up the first completely connected layer is a dropout layer, which plays an important role in preventing overfitting. In addition, there are 4096 neurons in the second fully connected layer, which is followed by yet another dropout layer. The output layer is the final fully connected layer, and it contains one thousand neurons to represent each of the one thousand classifications in the ImageNet dataset.

4. Softmax Activation: When applied to the output layer, the softmax activation function produces a distribution of probability over the classes that indicates the network's level of confidence for each class prediction. This distribution may be thought of as a measure of the network's accuracy.

ReLU Activation: The rectified linear unit (ReLU) activating function, which provides non-linearity into the network and accelerates up training by addressing the problem of vanishing gradients, was initially and effectively employed by AlexNet, one of the earliest models. This function was designed to bring non-linearity into the network and handle the vanishing gradient problem.

AlexNet’s success spurred the development and adoption of deeper and more complex neural network architectures. It paved the way for future developments like GoogLeNet, VGGNet, ResNet, and DenseNet and allowed for computer vision research and innovation to flourish.

2.2 VGGNet

VGGNet's straightforward architecture and impressive performance on the ImageNet challenge contributed to its popularity and had a deep impact on the field of computer vision. It highlighted the importance of deep networks with small convolutional filters, inspired further research in the field, and served as a baseline for subsequent architectures.

Overview of the VGGNet architecture (Refer Figure 2):

1. Input Layer: The network accepts a 227x227 pixel picture with three RGB color channels as input.

2. Convolutional Layers: VGGNet consists of 12 convolutional layers, each followed by a ReLU activation function. The convolutional layers use small filter sizes of 3x3 with a stride of 1 pixel and zero-padding to keep the spatial dimensions of the feature maps unchanged. The network is able to capture complex features as the number of filters gradually increases. Typically, VGGNet has 16-19 layers, depending on the variant.

3. Max Pooling Layers: After each set of two or three convolutional layers, VGGNet applies max pooling layers (pool size of 2x2) and a stride of 2 pixels). The most noticeable features are preserved while the spatial dimensions of the feature maps are reduced using max pooling.

4. Fully Connected Layers: The final set of layers in VGGNet consists of fully connected layers. The flattened feature maps from the last convolutional layer are fed into one or more fully connected layers. These layers act as a classifier, transforming the learned features into class probabilities. Depending on the particular VGGNet variant used, the number of neurons in the fully linked layers can change.

5. Softmax Activation: The output of the fully connected layer is fed into a softmax activation function, which results in the production of a probability distribution across all of the classes.

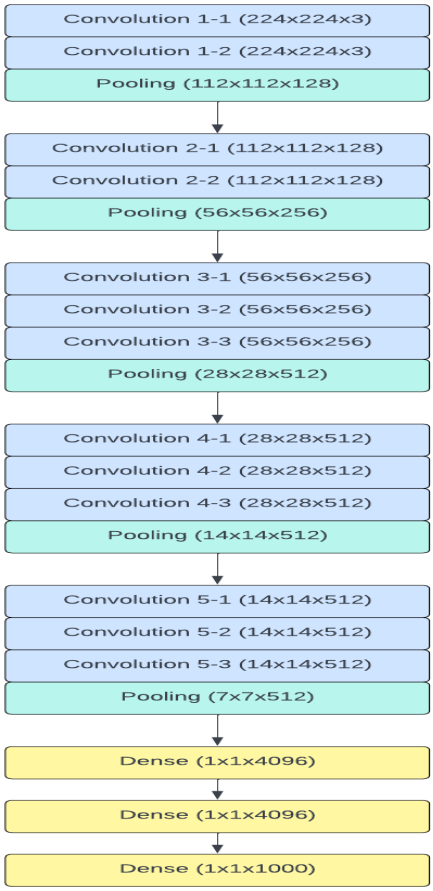


Fig 2: Architecture of VGGNet

2.3 ResNet

The ResNet (Residual Network) architecture was presented for the first time in the 2015 publication "Deep Residual Learning for Image Recognition" written by Kaiming He and colleagues. By introducing residual connections, it overcame the difficulty of training very deep neural networks. This allowed for the formation of networks with hundreds of layers while alleviating the issue of vanishing gradients.

Overview of the ResNet architecture (Refer Figure 3):

1. Input Layer: The network takes an input image of arbitrary size with three color channels (RGB).

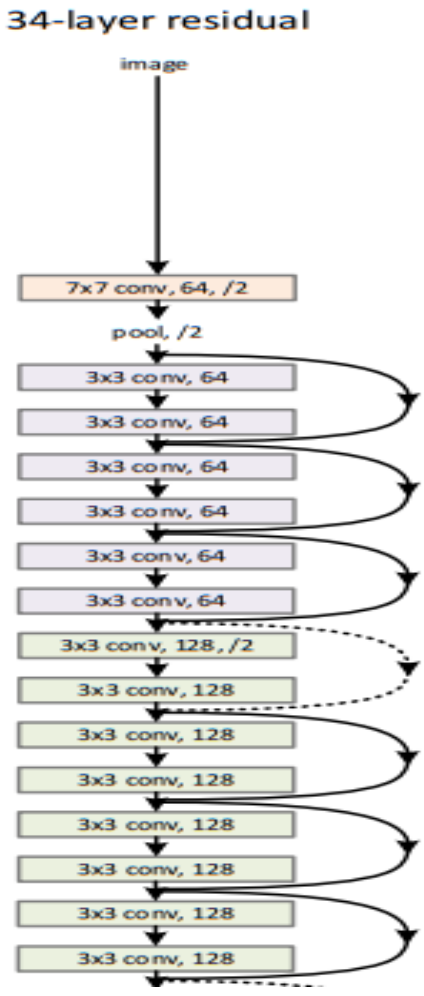
2. Convolutional Layers: Extraction of low-level features from the input image is performed by initial layers of ResNet, by performing convolution operations. Following these convolutional layers, batch normalization is applied and is passed through a ReLU activation function.

3. Residual Blocks: The residual blocks are the most important contribution to innovation made by ResNet. One or more convolutional layers may make up a residual block, and shortcut connections may be present between them. Through the use of a shortcut link, one or more convolutional layers can be skipped, and the input from an earlier layer can be added directly to the output of the residual block. This strategy teaches the network residual translations rather than the real feature mappings, which contributes to the reduction of the vanishing gradient problem.

4. Bottleneck Blocks: In more advanced variations of ResNet, bottleneck blocks are utilised. The computational cost is brought down by a bottleneck block, which works by first applying 1x1 convolutional layers in order to reduce the dimensionality of the input, then applying 3x3 convolutional layers, and finally applying 1x1 convolutional layers in order to recover the dimension of the input. This option in design helps to cut down on the total number of parameters while yet preserving their capacity for representation.

5. Pooling and Fully Connected Layers: The spatial dimensions of the feature maps are combined using global average pooling in order to get a single value for each individual feature map.

6. Towards the end of the network, global average pooling is applied, which reduces the spatial dimensions of the feature maps to a single value per feature map. These features are then fed into fully connected layers, which classifies the image by producing the class probabilities using softmax activation.



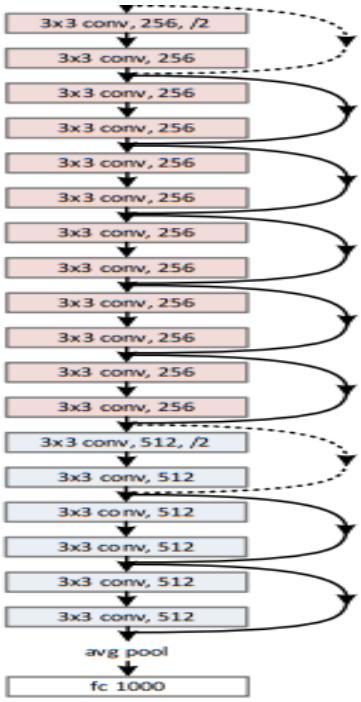


Fig 3: Architecture of ResNet

The main advantage of the ResNet architecture is its ability to train very deep networks effectively, even up to hundreds of layers. The residual connections allow for better gradient flow and enable the network to learn more intricate and abstract features. By mitigating the vanishing gradient problem, ResNet achieved improved accuracy on various image classification tasks.

3. Motivation

The motivation behind image classification is rooted in the need to automate the process of understanding and interpreting visual data. Images are rich sources of information, and humans can effortlessly recognize and categorize objects, scenes, and patterns in visual data. However, teaching computers to perform the same task is challenging and time-consuming.

The key motivations for image classification are:

1. Automation: Image categorization allows for the automated study of enormous amounts of visual data. With the increasing availability of digital photos from numerous sources such as cameras, social media, and medical imaging devices, manually analyzing and categorizing each image is impracticable and time-consuming.

2. Object Recognition: Image classification plays a crucial role in object recognition, where the goal is to identify and label specific objects within an image. Applications such as autonomous vehicles, surveillance systems, and robotics require accurate and efficient object recognition capabilities. Image classification models enable machines to detect and classify objects in real-time, enabling advanced functionalities in these domains.

3. Visual Search and Retrieval: Image classification enables efficient visual search and retrieval systems. By categorizing images into different classes, it becomes easier to organize and retrieve images based on their content. This has applications in areas such as content-based image retrieval, digital asset management, and e-commerce, where users can search for similar images or products based on visual similarity.

4. Decision Making: Image classification provides a foundation for decision-making systems based on visual data. For example, in medical imaging, classifying medical images (e.g., X-rays, MRIs) can assist in diagnosing diseases or conditions. Similarly, in satellite imagery analysis, image classification can aid in identifying land use, vegetation, or detecting changes over time.

5. Data Understanding and Insights: Image classification helps in understanding large-scale image datasets by providing insights into the distribution and characteristics of the images. It allows researchers and analysts to identify patterns, trends, and correlations within the image data, leading to discoveries and advancements in various fields.

4. Methodology

Image classification involves the following steps:

4.1 Data Collection

Data collection is the process of gathering a dataset of labeled images, where each image is associated with a known class or category. The dataset that you are going to classify should have a wide range of characteristics and be representative of those classes. We have utilised the CIFAR dataset for this particular experiment. The CIFAR-10 dataset is a standard for evaluating performance in image classification tasks. It is made up of 60,000 colour photos with a resolution of 32 by 32 pixels, organised into 10 classes with 6,000 pictures in each class.

Dataset Table	**CIFAR-10**
Classes	airplane, automatic, bird, cat, deer, dog, frog, horse, ship, truck
Number of Images	60,000
Image Size	32x32 pixels
Color Channels	RGB (3 channels)
Data Split	50,000 training images, 10,000 test images
Evaluation Metric	Classification Accuracy

Table 1: Dataset Table

4.2 Image Preprocessing

Image Preprocessing is the process of preparing the images for analysis by performing preprocessing steps such as resizing, normalization, and noise reduction. This ensures that the images are in a suitable format and that any irrelevant variations are minimized.

4.3 Feature Extraction

Feature extraction is an essential step that involves extracting meaningful and discriminative features from raw input images that can be used to distinguish between different classes. Feature Extraction is image classification models in performed with the help of below elements :

Here's a general overview of how feature extraction is performed in image classification models:

Convolutional Layers: The initial layers of an image classification model typically consist of convolutional layers. Convolution operations are performed on the input image using a set of learnable filters. Each filter detects different visual patterns or features in the image, such as edges, textures, or shapes. Multiple filters are applied to generate a set of feature maps, where each map represents the response of a specific filter across the entire image.

Activation Functions: Activation functions are applied to the feature maps element by element to create nonlinearity and allow the network to learn more complex representations. Activation functions help in capturing more abstract and higher-level features.

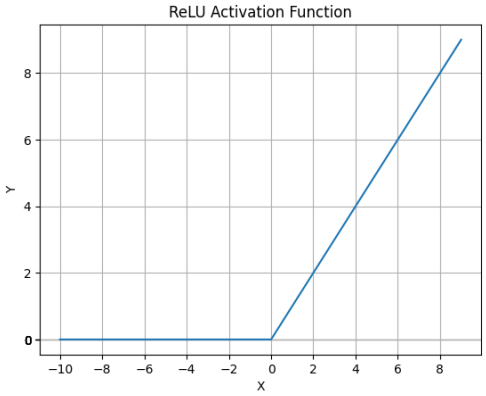


Fig 4: ReLU Activation Function

Pooling Layers: When attempting to downsample the feature maps, it is common practise to add pooling layers after a few convolutional layers have been created. The max pooling algorithm selects the highest possible value that exists inside a short section of the feature map (for example, 2x2) and discards the other values. The spatial dimensions of the feature maps are reduced thanks to pooling, which, in addition to conserving essential features, also increases the network's translation invariance.

Fully Connected Layers: The output of the layer with the convolutions and the layer with the pooling are typically smoothed and fed into one or more layers that are fully linked. These layers are responsible for the aggregation and transformation of high-dimensional features. Because every neuron in the fully connected layer is connected to every other neuron in the layer below it, the network is able to learn complicated correlations between the features that were collected.

Classification Layer: This layer takes the features that have been learned by the layers that came before it and translates them to the total amount of output classes that are involved in the classification process. The likelihood that is given the most weight in determining the class that is predicted. For example, the Softmax layer shown in Figure 5.

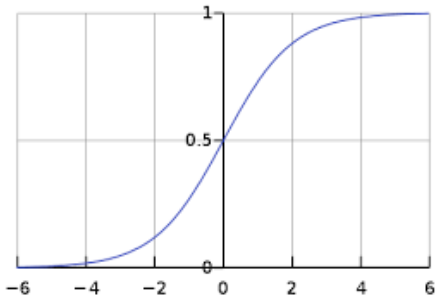


Fig 5 : Softmax Activation Function

4.4 Model Training

Models are trained using the labeled images on a deep learning architecture. The model learns to recognize patterns and features that are indicative of different classes during the training process.

While training a model, it is important to consider several hyperparameters that are mentioned below for optimal performance of the models.

1. s, and zooms, data augmentation techniques can assist improve the diversity and amount of the training data. The model's ability to generalise and adapt to variations in the test data can be improved with the help of augmented data.Learning Rate is the variable that determines the size of the steps that are done throughout the optimisation process. It establishes the rate at which the model adjusts to the data used for training. A low learning rate may result in slow convergence, whereas a high learning rate may cause the model to overshoot the optimal solution. Both of these outcomes are possible. It is essential to choose an optimum learning rate that enables effective and consistent training.
2. Batch Size: The batch size is what defines the number to educate examples that are worked through before the weights of the model are updated. A lower

batch size enables more frequent changes to the weight, but it also increases the likelihood of noisy estimations of the gradient. Larger batch sizes can potentially give more steady updates, but doing so takes more RAM. It is critical to locate an optimal compromise between the batch size and the rate of model convergence.

3. Number of Epochs: During the training process, the whole of the training dataset is processed by the model a certain number of times, which is determined by the number of epochs. In the event that the number of epochs is set too low, underfitting may occur, which is when the model does not fully learn from the data. When it is set too high, there is a risk of overfitting, which is when the model becomes overly specific to the training data. It is essential to monitor the performance of the model on a set of validation events and select a suitable number of epochs.
4. Activation Functions: The selection of activation functions has the potential to have an effect on the capacity of the model to learn complex patterns. The Rectified Linear Unit (ReLU), the Sigmoid Function, and the Tanh Function are all common activation functions. The selection of the proper activation function for each layer can have an effect on the learning capacity and performance of the model. These qualities are unique to each activation function.
5. Regularization Techniques: Through the addition of constraints to the model's architecture or weights, regularisation approaches are able to assist in preventing overfitting. approaches such as L1 and L2 regularisation, dropout, and batch normalisation are examples of common regularisation approaches. Regularisation is a technique that helps improve a model's capacity to generalise its findings and reduces the impact of characteristics in the training data that are noisy or irrelevant.
6. Network Architecture: The design of the model is part of the network architecture, and it includes information about the number of levels, the types of layers, the sizes of each layer, and the relationships between them. It's possible that different picture classification jobs call for the application of distinct architectures, including CNNs, ResNet, or DenseNet, that are better suited to the job. It is essential to take into account the data and computational resources at hand when determining the complexity of the model as well as its capacity to handle the data.
7. Optimization Algorithm: The optimisation algorithm that is used has an effect on the manner in which the model's weights are changed when it is being trained. Among these several optimisation techniques are the Stochastic Gradient Descent (SGD) algorithm, the Adam algorithm, and the RMSprop algorithm. The selection of an algorithm can have an effect on both the speed at which the model can converge and its ability to locate the best solution. Each algorithm has its own update rules and hyperparameters.
8. Data Augmentation: By adding random transformations such as rotations, flipta.
9. Weight Initialization: The initial values of the model's weights can impact the convergence and performance of the model. Choosing appropriate weight initialization techniques, such as Xavier or He initialization, can help the model start in a good region of the weight space and facilitate faster convergence.
10. Early Stopping: Early stopping is a technique where training is stopped early if the model's performance on a validation set does not improve for a certain number of epochs. It helps prevent overfitting and allows for the selection of the best performing model based on validation performance.

4.5 Model Evaluation

Model evaluation is the process of analysing the performance of the trained model by employing a second set of images known as a validation or test set. These sets of pictures are kept distinct from each other. Accuracy, precision, recall, and the F1 score are typical examples of metrics used in the evaluation of image classification systems.

4.6 Prediction

Prediction: After the model has been trained and evaluated, it may then be used to categorise photos that it has not before encountered. The model receives an image as input, and using the previously discovered patterns and characteristics, it makes a prediction about the category to which the image belongs.

5. Result Analysis

After training the models for image classification with the same training data and epochs, the accuracy of AlexNet was recorded to be the highest among the three models.

	Accuracy
AlexNet	76 %
VGGNet	61 %
ResNet	43 %

Table 2: Accuracy Scores of Models

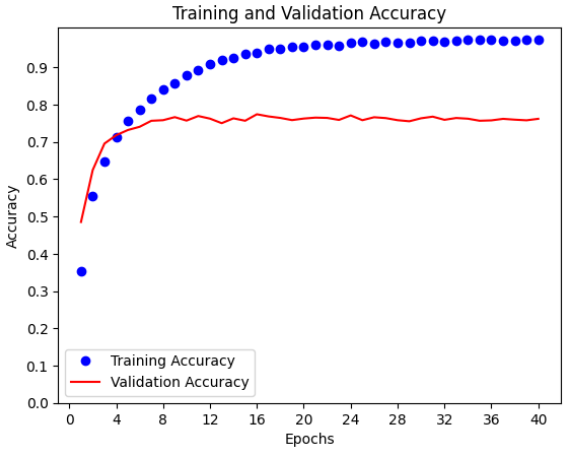


Fig 6 : Accuracy Curve for AlexNet

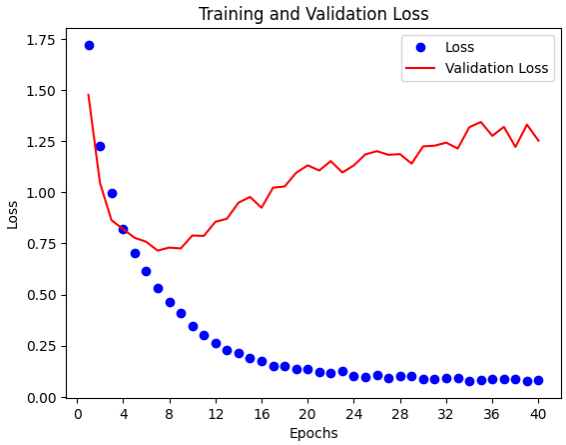


Fig 7: Loss Curve for AlexNet

In Figure 7 and Figure 6 , the validation loss for the AlexNet starts increasing after 8 epochs. After 20 epochs the training accuracy increases by a very small degree at each epoch. The validation accuracy hardly increases after 8 epochs. Hence training the model can be stopped after 10 epochs as there is not much change in the accuracy.The AlexNet model improves and is able to recognise patterns and features present in the images, hence the training accuracy increases and loss decreases.In Figure 8 and Figure 9, the training accuracy increases by a slight amount but the validation accuracy does not change by a big factor. Also after 20 epochs the validation loss starts increasing, resulting in overfitting of the model.

	precision	recall	f1 score

airplane	0.80	0.82	0.81
automobile	0.86	0.90	0.88
bird	0.64	0.69	0.67
cat	0.58	0.59	0.58
deer	0.73	0.71	0.72
dog	0.68	0.65	0.66
frog	0.81	0.81	0.81
horse	0.80	0.79	0.80
ship	0.88	0.82	0.85
truck	0.85	0.83	0.84
accuracy			**0.76**
macro avg	0.76	0.76	0.76
weighted avg	0.76	0.76	0.76

Table 3: Classification Report of AlexNet

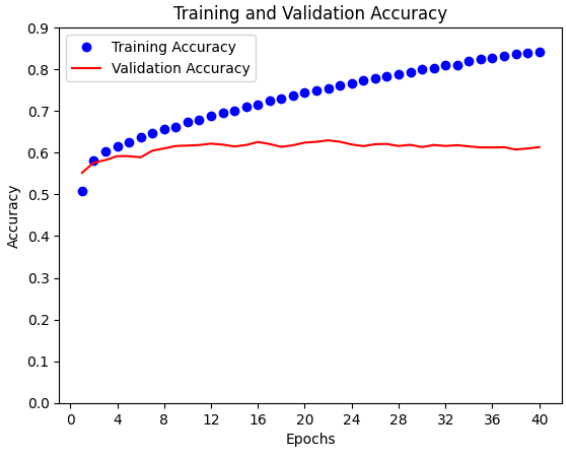


Fig 8: Accuracy Curve for VGGNet

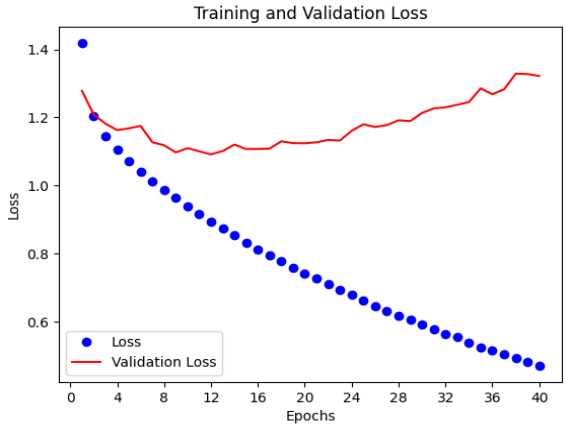


Fig 9: Loss Curve for VGGNet

	precision	recall	f1 score
airplane	0.71	0.66	0.68
automobile	0.65	0.69	0.67
bird	0.53	0.54	0.54
cat	0.48	0.36	0.41
deer	0.58	0.59	0.58
dog	0.52	0.53	0.53
frog	0.65	0.66	0.66
horse	0.64	0.71	0.67
ship	0.75	0.71	0.73
truck	0.61	0.67	0.64
accuracy			**0.61**
macro avg	0.61	0.61	0.61
weighted avg	0.61	0.61	0.61

Table 4: Classification report of VGGNet

The VGGNet model performs well on the training dataset but the same performance is not noticed in the testing dataset. Due to the above reason, the accuracy of VGGNet is lower than the AlexNet.

In Figure 10 and Figure 11, we can see a slight increase in training and validation accuracy after each epoch. The validation loss decreases as the model is trained further.

In this case, the model's accuracy for classification is still low, as it needs to be trained further.

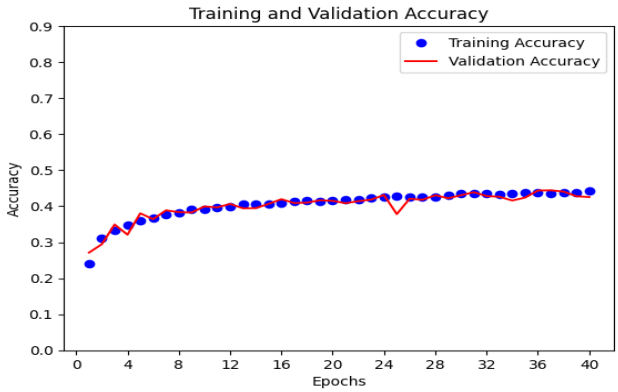


Fig 10 : Accuracy curve for ResNet

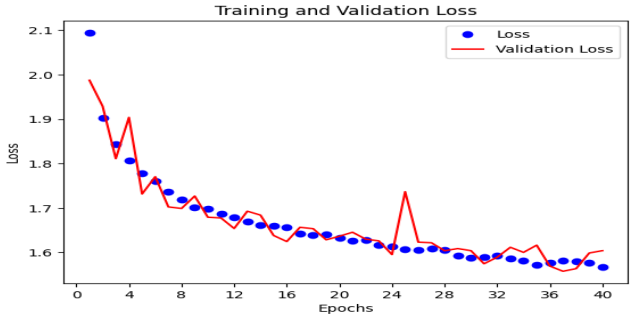


Fig 11: Loss curve for ResNet

	precision	**recall**	**f1 score**
airplane	0.53	0.40	0.46
automobile	0.47	0.46	0.46
bird	0.37	0.26	0.31
cat	0.31	0.20	0.24
deer	0.45	0.36	0.40
dog	0.34	0.54	0.42
frog	0.57	0.34	0.43
horse	0.62	0.36	0.46
ship	0.39	0.75	0.52
truck	0.40	0.58	0.37
accuracy			**0.43**
macro avg	0.44	0.43	0.42
weighted avg	0.44	0.43	0.42

Table 5: Classification Report of ResNet

6. Conclusion and Future Scope

Image Classification was performed by using 3 different models (AlexNet , VGGNet , ResNet). The accuracy score of AlexNet was higher than the other 2 models even though it has a much simpler architecture than the other models.

In the above experiment we can conclude that the model complexity is subject to change with the application it is being used for. A complex model may or may not be accurate. It is necessary to build a suitable model that is efficient as well as accurate. Experimenting with different architectures, hyperparameters and optimization techniques is crucial in order to achieve the desired result.

We may further train the models for more epochs and apply an early stopping function to prevent the models from overfitting by monitoring the accuracy of the model.

The goal of this research is to help academics and practitioners choose the best method for their individual needs by providing a better knowledge of the strengths and limits of various classification models. The insights gained from this work can inspire future advancements in image classification techniques and foster the development of more accurate and efficient models in the field of computer vision.

References

[1] “ImageNet Classification with Deep Convolutional Neural Networks" by Alex Krizhevsky et al. (2012)

[2] “Very Deep Convolutional Networks for Large-Scale Image Recognition" by Karen Simonyan and Andrew Zisserman (2014)

[3] “Going Deeper with Convolutions" by Christian Szegedy et al. (2014)

[4] “Deep Residual Learning for Image Recognition" by Kaiming He et al. (2015)

[5] “Densely Connected Convolutional Networks" by Gao Huang et al. (2017)

[6] Ramana, K. V. ., Muralidhar, A. ., Balusa, B. C. ., Bhavsingh, M., & Majeti, S. . (2023). An Approach for Mining Top-k High Utility Item Sets (HUI). International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 198–203. https://doi.org/10.17762/ijritcc.v11i2s.6045

[7] One weird trick for parallelizing “convolutional neural networks" by Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton (2014)

[8] "Learning Hierarchical Features for Scene Labeling" by Alex Krizhevsky and S. V. Nair (2009)

[9] "Return of the devil in the details: Delving deep into convolutional nets" by Karen Simonyan and Andrew Zisserman (2014)

[10] "Deep Inside Convolutional Networks: Visualizing Image Classification Models and Saliency Maps" by Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman (2013)

[11] Dr. Sandip Kadam. (2014). An Experimental Analysis on performance of Content Management Tools in an Organization. International Journal of New Practices in Management and Engineering, 3(02), 01 - 07. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/27

[12] "Very Deep Convolutional Networks for Large-Scale Visual Recognition" by Karen Simonyan and Andrew Zisserman (2015)

[13] "Explaining and Harnessing Adversarial Examples" by Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy (2015)

[14] "Identity Mappings in Deep Residual Networks" by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016)

[15] Dr. Antino Marelino. (2014). Customer Satisfaction Analysis based on Customer Relationship Management. International Journal of New Practices in Management and Engineering, 3(01), 07 - 12. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/26

[16] "Aggregated Residual Transformations for Deep Neural Networks" by Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He (2017)

[17] https://www.geeksforgeeks.org/residual-networks-resnet-deep-learning/

[18] https://www.analyticsvidhya.com/blog/2021/03/introduction-to-the-architecture-of-alexnet