Animal Species Image Classification

INTRODUCTION:

In a world teeming with biodiversity, the accurate identification and classification of animal species play a pivotal role in ecological research, conservation efforts, and wildlife management. This project delves into the realm of Animal Species Classification using Artificial Neural Networks (ANNs), leveraging the power of machine learning to enhance our ability to discern and categorize diverse species.

The primary focus of this endeavor is on the intricate process of Feature Extraction within the model. By unraveling the unique characteristics and patterns inherent in diverse species, we aim to develop a robust classification system that transcends the limitations of traditional methods. Additionally, the project endeavors to shed light on the efficacy of Feature Extraction in comparison to utilizing pre-trained models, providing valuable insights into the trade-offs and advantages offered by each approach.

PROBLEM SPECIFICATION:

The project revolves around Classification of Images of Animals into 4 different classes. There are 4000 different images available of these animals which are further divided into training and testing sets. The main aim is to showcase the results of feature extraction which helps in increasing the accuracy of models. There are primarily 3 models which projects the increase in accuracy and the results after using feature extraction. Finally, we have pre-trained model which helps us to compare the results of self-designed model with complex pre-trained models. Further, We have to show the comparison studies of number of classes vs F1 score as well as results of prediction on different models.

CLASSIFICATION MODELS:

1. SIMPLE ANN:

A simple artificial neural network (ANN) typically comprises a fundamental building block known as a dense layer. In this architecture, neurons, or artificial nodes, within a layer are densely connected to all nodes in the adjacent layer. Each connection is assigned a weight, and the layer incorporates an activation function to introduce non-linearity. The dense layer is crucial for learning complex patterns and relationships within the data. During training, the weights are adjusted through backpropagation, optimizing the network to make accurate predictions or classifications. This basic structure forms the backbone of many neural network architectures and serves as a foundation for more complex models. The accuracy and loss achieved are:

loss: 1.3153
accuracy: 0.3571
val_loss: 1.2619
val_accuracy: 0.4115

This neural network architecture follows a sequential structure for a classification task. It starts with a Flatten layer for input processing, followed by three Dense layers of decreasing neuron counts (512, 256, 128) with ReLU activation. Each dense layer includes dropout for regularization (0.5, 0.3, 0.2, respectively), reducing overfitting. The second and third dense layers employ L2 regularization (0.01) to control weights. The final Dense layer (4 neurons) with softmax activation outputs probabilities for multi-class classification. Overall, the architecture is designed for robust feature learning and prevention of overfitting.

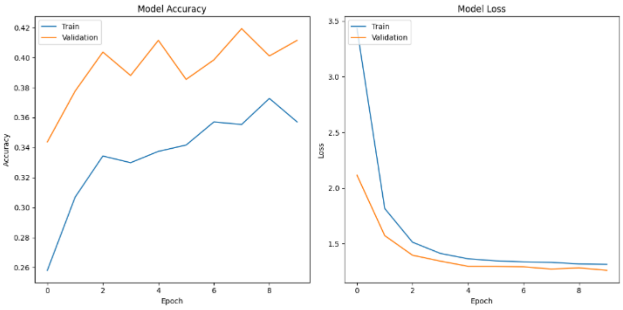

Fig 1: Accuracy and Loss curves of Simple ANN

2. WAVELET TRANSFORM :

Wavelet transformation is a mathematical technique employed for signal and image processing that decomposes a signal into different frequency components. Unlike traditional Fourier transforms, wavelet transformation allows for both time and frequency localization, making it particularly effective in capturing transient features in a signal. The transformation involves convolving the signal with wavelet functions, which are scaled and translated versions of a base wavelet. This process results in a multi-resolution analysis, breaking down the signal into approximation and detail coefficients across different scales. Wavelet transformation finds applications in diverse fields, including image compression, denoising, and feature extraction, offering a versatile tool for analyzing and representing complex signals with both high and low-frequency components.

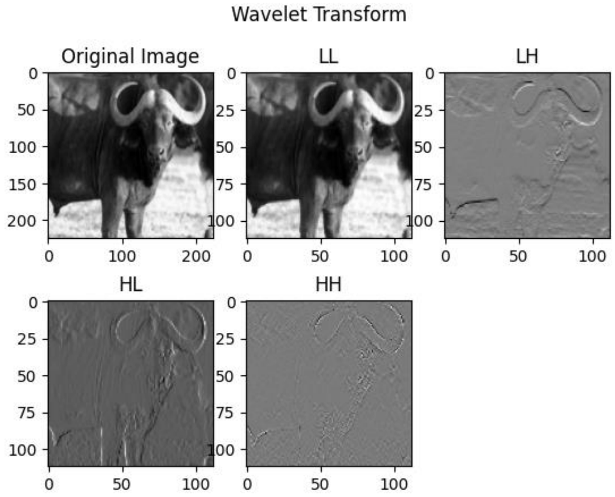

Fig 2: Features extracted from Wavelet Transformation

The accuracy and loss achieved are:

loss: 1.1666
accuracy: 0.4419
val_loss: 1.3484
val_accuracy: 0.3250

The presented neural network architecture utilizes a pi wavelet transform as a preprocessing step for image data. The pi_wavelet_transform function takes a root path, class names, and image size as input parameters. For each image in the specified classes, it loads the image, resizes it, and applies the pi wavelet transform using the Haar wavelet. The resulting wavelet coefficients (LL, LH, HL, HH) are concatenated to form a 4-channel image representation. These transformed images and their corresponding class labels are then used to train a Sequential neural network model. The model architecture consists of a Flatten layer to accommodate the 4-channel input, followed by three Dense layers (512, 256, 128 neurons) with ReLU activation. Dropout layers (0.5, 0.3, 0.2) are employed for regularization, and L2 regularization (0.01) is applied to the second and third dense layers. The final Dense layer with softmax activation produces probabilities for multi-class classification. This architecture integrates wavelet-based feature extraction into the neural network for improved performance on image classification tasks.

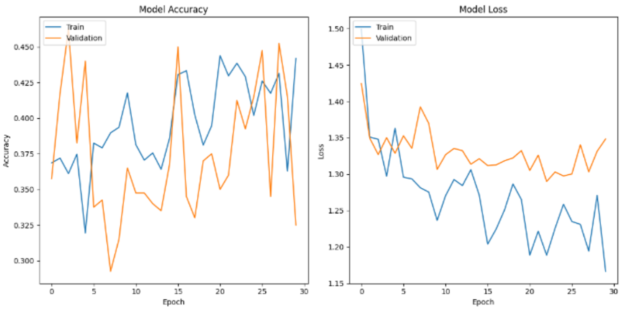

Fig 3: Accuracy and Loss curves of Wavelet Transform

3. Simple CNN:

Convolutional Neural Networks (CNNs) represent a specialized class of deep learning models designed for processing and analyzing grid-like data, such as images and videos. The key innovation of CNNs lies in their use of convolutional layers, which efficiently scan input data through learnable filters to detect hierarchical patterns and spatial relationships. These filters enable the network to automatically learn relevant features, reducing the need for manual feature engineering. CNN architectures typically consist of convolutional layers followed by pooling layers to down sample the spatial dimensions and then fully connected layers for classification or regression tasks. CNNs have demonstrated remarkable success in image recognition, object detection, and other computer vision applications, owing to their ability to capture local patterns while maintaining translational invariance, making them an integral technology in modern artificial intelligence systems. The accuracy and loss achieved are:

loss: 0.7341
accuracy: 0.6993
val_loss: 0.7666
val_accuracy: 0.7031

This neural network architecture is tailored for image classification. It starts with two convolutional layers, each followed by max-pooling, gradually reducing spatial dimensions. The flattened output is then processed through a

Dense layer with 128 neurons and ReLU activation, followed by two additional Dense layers (64 neurons each) with ReLU activation and L2 regularization (0.01). Batch normalization is applied after each Dense layer. The final layer consists of four neurons with softmax activation for multi-class classification. This design efficiently captures hierarchical features through convolutional operations and reduces overfitting with regularization and batch normalization.

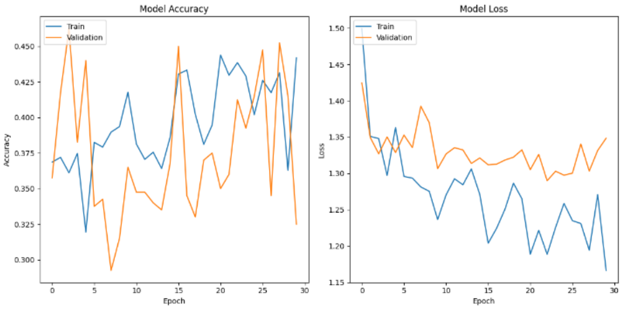

Fig 4: Accuracy and Loss curves of CNN

4. TRANSFER LEARNING VGG-16 :

Transfer learning is a machine learning paradigm that leverages knowledge gained from training a model on one task and applies it to a different but related task. In the context of neural networks, transfer learning involves using a pre-trained model, often on a large dataset, as the starting point for a new task. This approach is particularly beneficial when the new task has limited labeled data. The pre-trained model's learned features can be valuable for capturing general patterns and representations, which are then fine-tuned on the specific task. Transfer learning is widely employed in computer vision, natural language processing, and other domains, facilitating the development of more accurate and efficient models, especially when training data is scarce. It allows practitioners to tap into the knowledge acquired by models from one domain and apply it effectively to improve performance in a related domain. The accuracy and loss achieved are:

loss: 1.1135
accuracy: 0.7618
val_loss: 0.5234
val_accuracy: 0.8863

VGG-16, short for Visual Geometry Group 16-layer, is a widely recognized convolutional neural network (CNN) architecture designed for image classification. Developed by the Visual Geometry Group at the University of Oxford, VGG-16 is characterized by its deep structure and uniform architecture. The model consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. The convolutional layers use small receptive fields (3x3) with a stride of 1, maintaining a compact and consistent design. Max pooling is applied after every two convolutional layers, reducing spatial dimensions. VGG-16 is known for its simplicity and efficacy, achieving competitive results in image classification tasks. While it may have more parameters compared to some contemporary architectures, its straightforward design has made it a valuable benchmark in the development of deeper neural networks.

COMPARISON STUDIES AND EXTENSIONS:

In our comparative study, we are investigating the relationship between the number of classes and the F1 score within a ResNet 50 model. The F1 score is a metric that combines precision and recall, providing a comprehensive assessment of a model's performance, particularly valuable in situations with imbalanced class distribution. Precision measures the accuracy of positive predictions, while recall gauges the model's ability to capture all relevant instances of a given class. The F1 score is the harmonic mean of precision and recall, offering a balanced measure that is especially useful when false positives and false negatives carry different levels of significance. As we vary the number of classes in the ResNet 50 architecture—a deep neural network acclaimed for its innovative use of residual connections—we aim to discern how the model's performance adapts to different classification scenarios, shedding light on the interplay between architectural complexity and classification accuracy.

- 4 Class Model

F1 Scores for Model (one batch):
[0.44444444
0.63157895
0.77777778
0.77777778]

Average F1 Score for Model (one batch): 0.675438596491228l

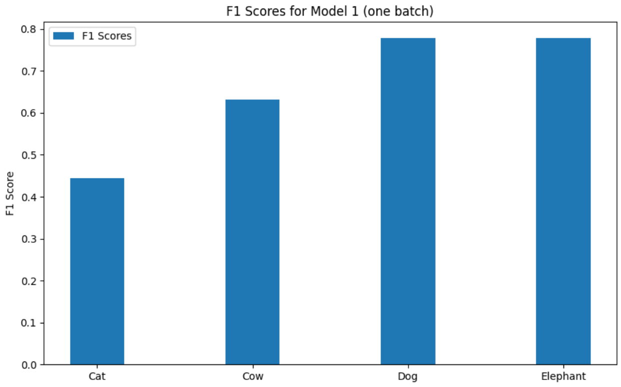

Fig 5: F1 Score for 4 Class Model

- 5 Class Model

F1 Scores for Model (one batch):
[0.8
0.76190476
0.28571429
0.83333333
0.85714286]

Average F1 Score for Model (one batch): 0.7116071428571429

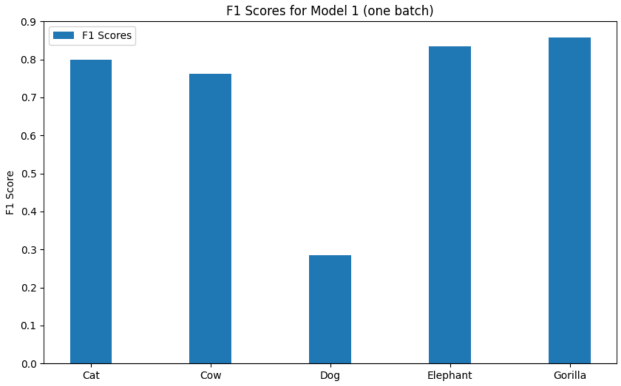

Fig 6: F1 score for 5 Class Model

- 6 Class Model

F1 Scores for Model (one batch):
[0.7
0.66666667
0.28571429
0.66666667
0.92307692
0.66666667]

Average F1 Score for Model (one batch): 0.6824290293040293

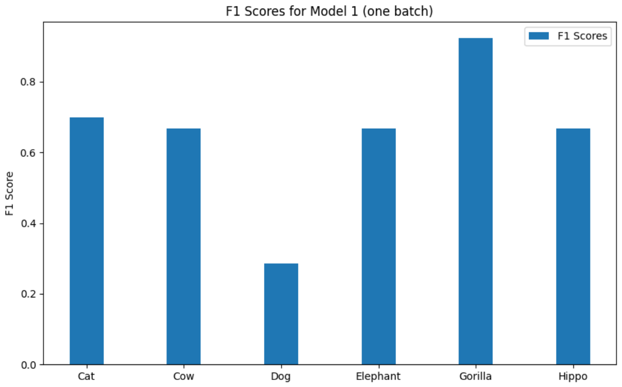

Fig 7: F1 score for 6 Class Model

- 7 Class Model

F1 Scores for Model (one batch):
[0.8
0.5
0.5
0.75
0.57142857
0.57142857
0.58823529]

Average F1 Score for Model (one batch): 0.622203256302521

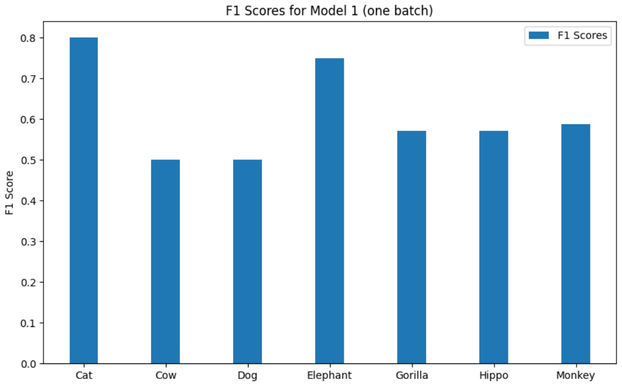

Fig 8: F1 score for 7 Class Model

- 8 Class Model

F1 Scores for Model (one batch):
[0.6
0.8
1.
0.85714286
0.85714286
0.66666667

0.15384615
0.42857143]

Average F1 Score for Model (one batch): 0.61492673992674

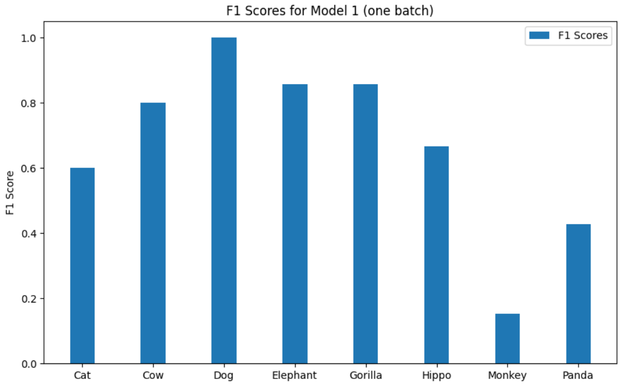

Fig 9: F1 score for 8 Class Model

- 9 Class Model

F1 Scores for Model (one batch):
[1.
0.4
0.8
1.
0.57142857
0.88888889
0.46153846
0.4
1.]

Average F1 Score for Model (one batch): 0.730921855921856

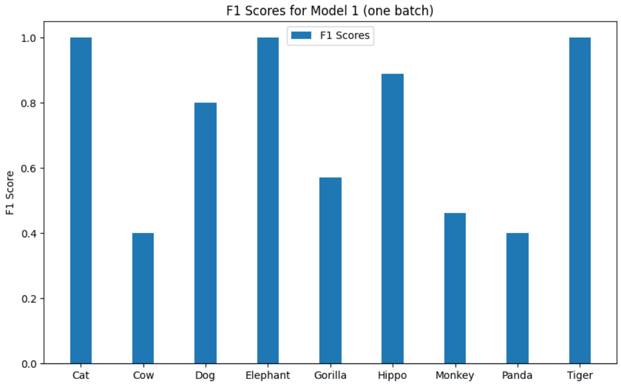

Fig 10: F1 score for 9 Class Model

- 10 Class Model

F1 Scores for Model (one batch):
[0.5
0.72727273
0.28571429
1.
0.85714286
0.8
0.75
1.
0.25
0.75]

Average F1 Score for Model 1 (one batch): 0.6752029220779221

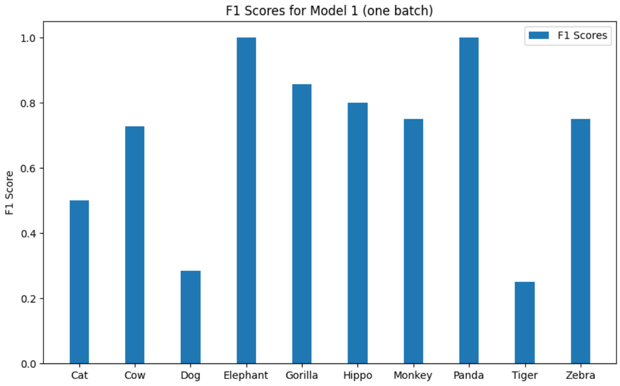

Fig 11: F1 score for 10 Class Model

ANALYSIS OF OTHER MODELS:

1. VGG - 16:

The VGG-16 neural network, crafted by the Visual Geometry Group at the University of Oxford, stands as a pivotal milestone in convolutional neural network (CNN) architecture, particularly tailored for image classification. With an imposing depth, this model comprises 16 layers, featuring 13 convolutional layers and 3 fully connected layers. What distinguishes VGG-16 is its uniform structure, employing consistently small 3x3 convolutional filters with a stride of 1 across the entire network. The application of max pooling after every two convolutional layers ensures an effective reduction in spatial dimensions. Renowned for its straightforward yet powerful design, VGG-16 has become a benchmark in the field, showcasing the effectiveness of deeper architectures in capturing intricate hierarchical features for image recognition tasks. Despite its comparatively larger parameter count in contemporary contexts, VGG-16 remains influential, providing valuable insights that have guided the development of subsequent advanced neural network architectures.

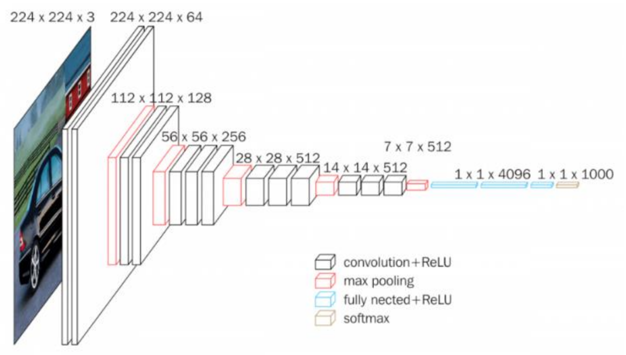

Fig 12: Architecture of VGG – 16 Neural Network

The output obtained from the VGG-16 convolutional neural network reveals promising performance metrics, demonstrating its efficacy in the task at hand. The training metrics exhibit a relatively low loss of 0.4027 and a high accuracy of 89.41%. These figures underscore the model's capability to minimize errors and make accurate predictions on the training dataset. Furthermore, the validation results, represented by a loss of 1.2526 and an accuracy of 77.57%, suggest a robust generalization of the model to unseen data. While there is a slight drop in accuracy compared to the training set, the VGG-16 CNN maintains strong predictive capabilities on the validation data, indicating a well-balanced and effective model that avoids overfitting. Overall, these output metrics attest to the VGG-16 model's competence in achieving high accuracy and generalization performance.

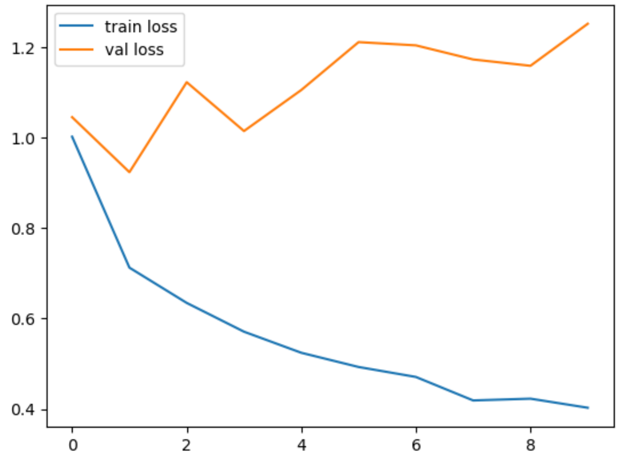

Fig 13: Loss Curve of VGG – 16 Neural Network

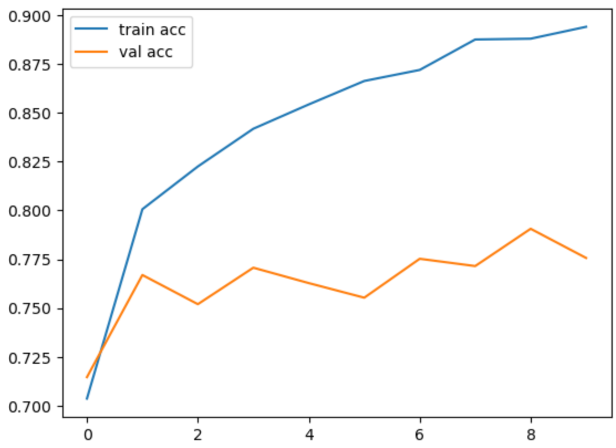

Fig 14: Accuracy Curve of VGG – 16 Neural Network

2. ResNet v50:

The ResNet-50 (Residual Network with 50 layers) is a formidable convolutional neural network architecture that has significantly advanced the field of deep learning. Its key innovation lies in the incorporation of residual connections, addressing the challenge of vanishing gradients in exceedingly deep networks. ResNet-50 consists of 50 layers, featuring a stack of convolutional layers, residual blocks, and fully connected layers. The residual blocks, equipped with shortcut connections, allow the model to directly pass along the original input to subsequent layers, facilitating the flow of gradients during training. This unique architecture enables the construction of remarkably deep networks without suffering from degradation issues. Specifically, ResNet-50 includes a series of convolutional layers, max-pooling, and four sets of residual blocks with varying numbers of convolutional layers. The final layers involve global average pooling and fully connected layers for classification. Renowned for its depth and accuracy, ResNet-50 has become a cornerstone in image classification tasks, showcasing the pivotal role of residual connections in training deep neural networks effectively.

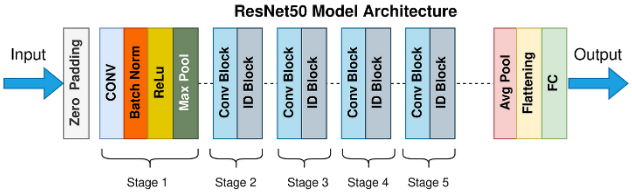

Fig 15: Architecture of ResNet v50

The output from the ResNet-50 neural network reflects robust performance on the training set, with a relatively low loss of 0.4000 and a commendable accuracy of 86.99%. These metrics highlight the model's effectiveness in minimizing errors and making accurate predictions on the training data. The validation results further indicate a strong generalization capability, with a loss of 0.8439 and an accuracy of 74.50%. While there is a slight decrease in accuracy compared to the training set, the ResNet-50 model demonstrates solid predictive power on previously unseen validation data, suggesting a well-balanced and generalizable architecture. The combination of a low training loss and a relatively high validation accuracy attests to the ResNet-50's ability to capture complex features and patterns in the data while avoiding overfitting. Overall, these output metrics underscore the ResNet-50's competence in achieving both accurate training and effective generalization to new data.

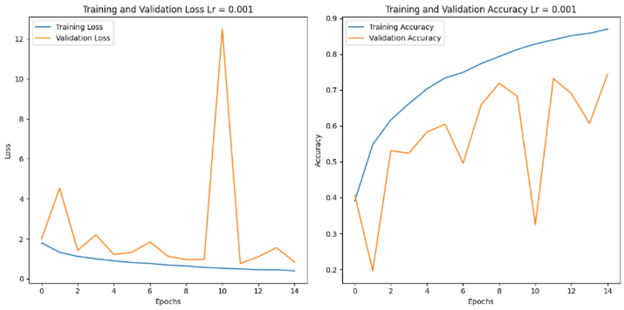

Fig 16: Accuracy and Loss Curve of ResNet v50

3. InceptionNet

InceptionNet, also known as GoogLeNet, represents a groundbreaking convolutional neural network (CNN) architecture designed by Google. The distinguishing feature of InceptionNet lies in its inception modules, which utilize multiple filter sizes within the same layer to capture a diverse range of features. The architecture is characterized by its depth and computational efficiency. It features a stacked sequence of inception modules, which include 1x1, 3x3, and 5x5 convolutions, as well as max-pooling operations. These modules are concatenated to form a rich feature map that captures both local and global patterns. Additionally, 1x1 convolutions are used to reduce dimensionality before more computationally expensive operations. The inception architecture, with its emphasis on parallel processing and efficient use of parameters, enables the construction of deeper networks without an exponential increase in computational cost. This innovation has proven highly effective in image recognition tasks, and InceptionNet remains influential in the evolution of CNN architectures.

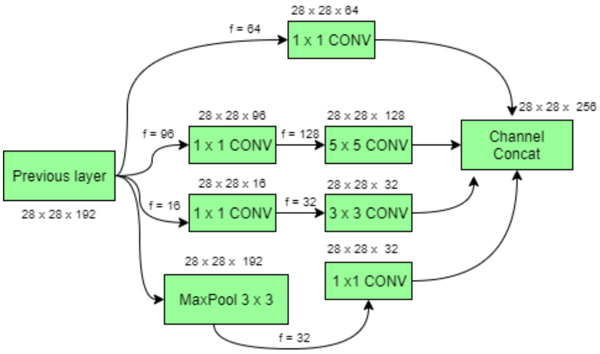

Fig 17: Architecture of InceptionNet

The output results from the InceptionNet neural network showcase outstanding performance on the training set, exhibiting a remarkably low loss of 0.1515 and an impressive accuracy of 95.03%. These metrics underscore the model's proficiency in minimizing errors and accurately classifying the training data. On the validation set, the InceptionNet model maintains strong generalization capabilities with a loss of 0.9709 and an accuracy of 76.65%. While there is a slight drop in accuracy compared to the training set, the model demonstrates a solid ability to make accurate predictions on new, unseen data. The combination of low training loss and respectable validation accuracy suggests that InceptionNet effectively captures intricate patterns and features in the data, showcasing its robustness and suitability for complex image recognition tasks. Overall, these output metrics attest to InceptionNet's capability to balance accuracy and generalization performance.

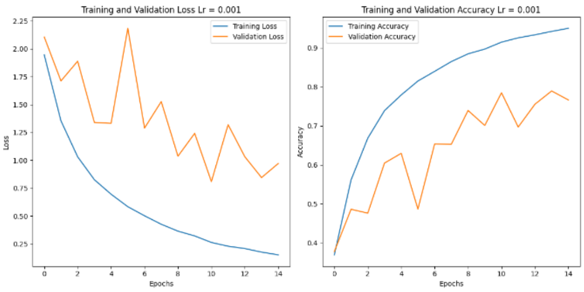

Fig 18: Accuracy and Loss Curve of InceptionNet

4. Transformer Network:

The Transformer neural network represents a groundbreaking architecture introduced for natural language processing tasks, particularly in machine translation, and has since been extended to various other domains. Its architecture relies on a self-attention mechanism, allowing the model to weigh different parts of the input sequence differently during processing. The Transformer architecture consists of an encoder-decoder structure. The encoder processes the input sequence, capturing contextual information through self-attention layers, while the decoder generates the output sequence based on the encoded information. The self-attention mechanism enables the model to consider dependencies between all positions in the input sequence simultaneously, eliminating the need for recurrent or convolutional connections. This parallelization enhances training efficiency and facilitates the modeling of long-range dependencies. The Transformer architecture has become foundational in the field of natural language processing, providing a scalable and efficient solution for sequence-to-sequence tasks.

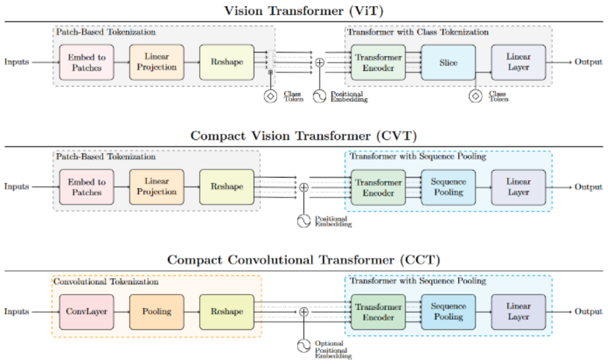

Fig 19: Architecture of ViT, CvT and CCT

- Vision Transformer (ViT):

The Vision Transformer is a neural network architecture that extends the Transformer model, originally designed for natural language processing, to image classification tasks. ViT divides an input image into fixed-size patches, linearly embeds them, and then applies self-attention mechanisms to capture global dependencies and interactions between patches. Positional embeddings are added to retain spatial information. ViT has demonstrated competitive performance, challenging the dominance of convolutional neural networks in computer vision.

- Compact Vision Transformer (CvT):

Compact Vision Transformer is an enhancement of the Vision Transformer, addressing the scalability challenges associated with large-scale models. CvT introduces a hierarchical architecture, incorporating multiple stages of attention, where each stage operates at a different resolution. This enables the model to efficiently process images of various scales. CvT balances performance and efficiency, making it suitable for real-world applications with limited computational resources.

- Compact Convolutional Transformer (CCT):

Compact Convolutional Transformer further refines the integration of convolutional and transformer architectures. It combines local convolutions and global self-attention mechanisms in a single module, allowing the model to capture both local and global context efficiently. This hybrid approach aims to leverage the strengths of both convolutional and transformer architectures, offering improved performance on a diverse range of computer vision tasks while maintaining computational efficiency. CCT represents a bridge between traditional convolutional networks and the emerging transformer-based architectures in computer vision.

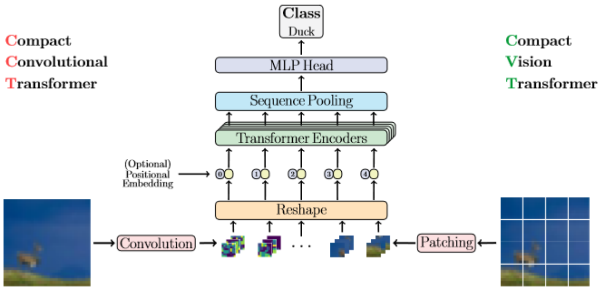

Fig 20: Working of CvT and CCT

The output from the Compact Convolutional Transformer (CCT) model reveals a challenging training scenario, as indicated by a relatively high loss of 2.4275 and an accuracy of 44.23% on the training set. The Top-3 accuracy, measuring the proportion of instances where the correct class is among the top three predicted classes, is slightly higher at 69.18%. Similarly, the validation set metrics display a loss of 2.8195, accuracy of 41.63%, and Top-3 accuracy of 61.74%. The notable discrepancy between training and validation metrics suggests potential overfitting or insufficient model convergence, possibly due to a limited number of training epochs. The low accuracy in a lesser number of epochs could be attributed to resource constraints and longer training times, impeding the model's ability to sufficiently learn complex patterns in the data. Increasing the number of epochs or optimizing the training process could potentially improve the CCT model's accuracy and convergence.

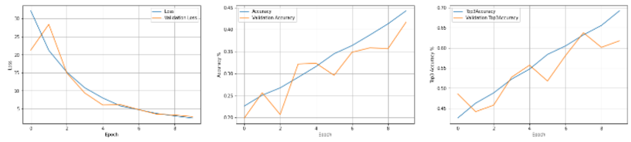

Fig 21: Accuracy and Loss of CCT

CONCLUSION:

In conclusion, this project has provided a comprehensive exploration into the realm of artificial neural networks (ANN) and convolutional neural networks (CNN), shedding light on their fundamental principles and applications. The understanding of feature extraction techniques and transfer learning has allowed for a deeper insight into leveraging pre-trained models for various tasks, enhancing the efficiency of model training on limited datasets. The exploration of advanced neural network architectures, including ResNet, VGG-16, InceptionNet, Vision Transformer, and others, has broadened the perspective on diverse approaches to solving complex problems in computer vision and natural language processing. The journey through transformer networks, originally designed for NLP, and their adaptation to vision tasks further exemplifies the versatility of deep learning models. Overall, this project has been an enriching experience, deepening the understanding of neural networks and paving the way for future endeavors in leveraging advanced architectures for diverse applications.

CONTRIBUTIONS:

- Harsh Khurana (S20210020279):

Worked with CNN and Feature Extraction of Wavelet Transform. Worked on ResNet v50 for Comparison studies of Num. of Classes vs F1 score. Worked on InceptionNet and Transformer Network include ViT, CvT and CCT.

- Ankit Singh (S20210020254):

Worked with primary ANN for prediction, Fine Turning and show casing results. Worked on Data Analysis and Data Preprocessing of all the Networks

- Akash Srivastava (S20210020250):

Worked on Transfer learning for comparison studies. Specially VGG-16 for Fine Turning the Network and Understanding it with depth.

APPENDIX:

Link for Datasets Used:

https://www.kaggle.com/datasets/utkarshsaxenadn/animal-image-classification-dataset

https://www.kaggle.com/datasets/ayushv322/animal-classification