Visual Reality Showdown
AI vs. Reality

Jaladurgam Navya
Department of Data Science
Kent State University,USA
jnavya@kent.edu

Aruna Mokara
Department of Computer Science
Kent State university,USA
amokara@kent.edu

Bharathi Donku
Department of Computer Science
Kent State university,USA
bdonku@kent.edu

Satya Mouli Nikhila Vadlamani
Department of Computer Science
Kent State university,USA
svadlam1@kent.edu

Abstract—**This In the past people were the only ones generating pictures ,but now AI can generate a wide variety of images like paintings, landscapes, human faces and portraits etc.; However, having so many AI made images around makes people worry about their privacy, authenticity and the spread of misinformation. So, our goal is to classify the images of human and AI generated images. First, we collected a dataset of different types of images consisting of human and AI generated images, then we trained three different models including CNN,ResNet50 and InceptionV3 , which achieved accuracies of 78 percentage,83 percentage and 83 percentage, respectively.**

Keywords—Classification,CNN,ResNet50,images ,InceptionV3, real, accuracy

I. INTRODUCTION

The development of artificial intelligence has changed the process of creating images . AI can generate diverse types of images , which look like real images such as paintings, human faces etc.; But there are few significant problems with this innovation. Because the AI can create photos of people without their consent, many are concerned with privacy. Not only human faces , but also AI can generate different types of images , we are unable to distinguish images created by human or AI. We're classifying which pictures were created by people and which ones were generated by AI.

First, we collected the different images from Kaggle dataset. This dataset has human and AI generated images. Then, to train the data. We built three different models are Convolutional Neural Networks, ResNet50, and InceptionV3. These models help us tell apart images made by people and those made by AI. For ResNet50 and InceptionV3, we used pre-trained weights. They already learned a lot from looking at tons of pictures in earlier training .They know things like shapes, colors, and patterns well. So, when we use them in our models, they help the models learn faster and better. These pre-trained models learnt from huge collections of images, like ImageNet.

To evaluate model performance, we used loss and validation curves, which are graphs showing how well the model learns. Then, the confusion matrix which is like a table that helps us to see if our model is guessing images correctly. It shows how many images were classified correctly and how many were mistaken. Lastly, we used a classification report to summarize how well our model performs for different types of images. It tells us things like how accurate our model is for each category of images.

II. ARCHITECTURES

A. CNN Architecture

Convolutional neural networks are a kind of deep learning network that are often used for picture classification, Object identification, and segmentation in computer vision tasks. CNNs are great at understanding the spatial layout and patterns in images. CNN consists of three main layers convolutional layers, pooling layers, and a fully connected layer.

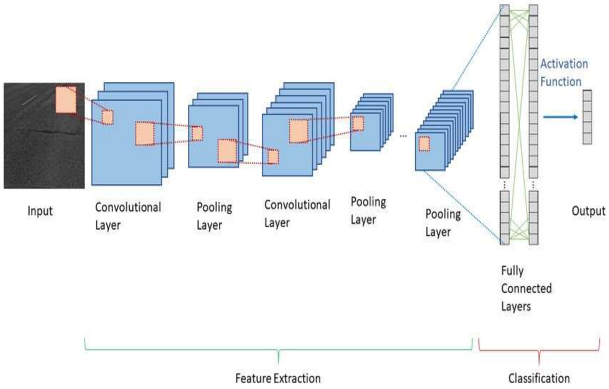


Fig 1 : Architecture of CNN

B. ResNet50 Architecture:

The ResNet50 (Residual Network) model has 50 layers . ResNet50 introduces the concept of residual connections, a pivotal innovation allowing the network to develop into more intricate feature learning. This mechanism involves the direct transmission of information from earlier layers to subsequent ones, circumventing certain layers and thus mitigating the vanishing gradient problem.

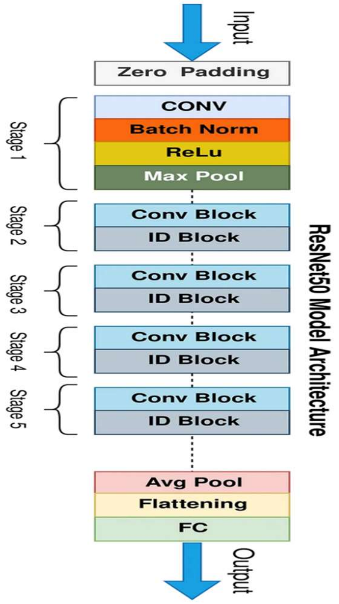


Fig 2: Architecture of ResNet50

C. *InceptionV3 Architecture:*

InceptionV3 is a convolutional neural network architecture , this is developed by Google .It a part of the Inception family of models. It has been trained on the ImageNet dataset and designed for image classification. The main building blocks of InceptionV3 are the Inception modules. These modules consist of parallel convolutional layers of different filter sizes (1x1, 3x3, 5x5) and a max-pooling layer. By having these parallel paths the network can learn to capture features at various scales effectively.

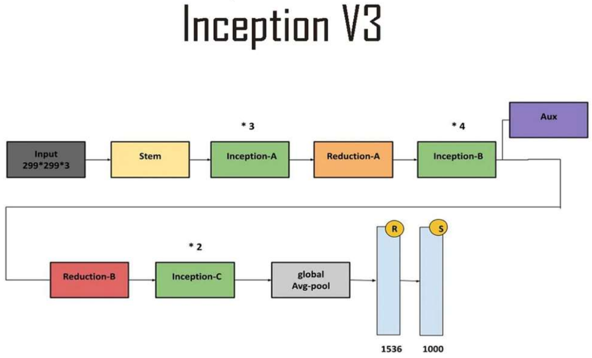


Fig 3: Architecture of InceptionV3

III. Survey (Literature Review)

1.In 2023, Gaye Ediboglu Bartos and Serel Akyol conducted research on "Deep Learning for Image Authentication: A Comparative Study on Real and AI-Generated Image Classification" presenting a comparative analysis of deep learning techniques for distinguishing between real and AI-generated images. They utilized the CIFAKE image dataset and employed ResNet and VAE models, achieving good accuracy. The accuracy of ResNet was 94%, while the accuracy of VAE was 71%.[1]

2.In 2018, Weize Quan, Kai Wang, Dong-Ming Yan, and Xiaopeng Zhang conducted research on Distinguishing Between Natural and Computer-Generated Images Using Convolutional Neural Networks. They utilized two datasets comprising images with different pixel sizes. The first dataset consisted of computer-generated images, while the second dataset comprised natural images. For training, evaluating, and analyzing the model, they employed CNN. The accuracy achieved with a pixel size of 240 was 93.20%.[2]

3.In 2024, Ruchira Purohit, Yana Sane, and Devashree Vaishampayan conducted research aimed at differentiating between AI-generated and natural images. They utilized two datasets: the first dataset was obtained from the 'Google' image library, containing AI images, while the second dataset comprised images captured using their own camera. Their approach involved employing a CNN model with two output layers utilizing sigmoid activation for binary classification. The accuracy achieved for the Google image dataset was 88%, while for their own dataset was 81%.[3]

4.In 2023 ,Shivani Atul Bhinge and Piyush Nagpal conducted a study on "Quantifying the Performance Gap between Real and AI-Generated Synthetic Images in Computer Vision " aiming to investigate the performance disparity between computer vision systems when analyzing real images versus AI-generated synthetic images. They utilized the CIFAKE dataset and employed EfficientNet and CNN models to train and evaluate the model. The accuracy of the CNN model was 75.76%, while the EfficientNet model achieved an accuracy of 95.97%.[4]

5.Jordan J. Bird and Ahmad Lotfi conducted research on Image Classification and Explainable Identification of AI-Generated Synthetic Images in 2023. In this paper, the authors utilized the CIFAKE dataset and employed CNN models to train and evaluate the model, yielding good results. The accuracy of the CNN model is 92.985%.[5]

IV. Dataset

The dataset contains images generated by AI and humans, this dataset has various categories such as landscapes, paintings ,animals, people etc.; These images are collected from web scraping and AI-generated . There are 975 images in the dataset. It has two folders with the AI and Real images . The AI images are 539 images and Real images are 436 images. The dataset has diverse types of images as shown below. This dataset is splitting into training and testing sets.

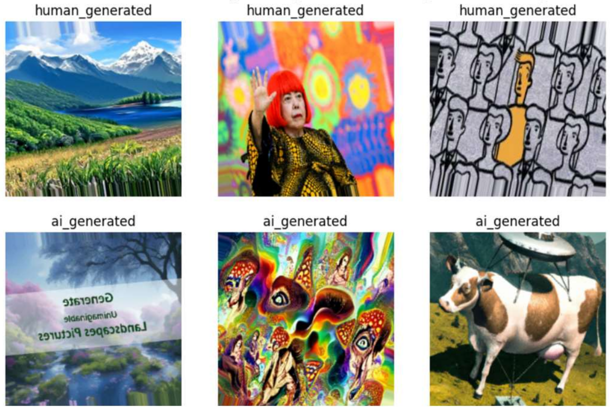


Fig 4: Images in the dataset

V. Methodology

A. Data Preprocessing:

Preparation involves several crucial steps designed to enhance raw data for training deep learning models. The process includes adjusting pixel values within a standard range, utilizing data augmentation methods to increase dataset variety, and converting image formats for framework compatibility. Pre-processing enhances deep learning models' ability to learn effectively and generalize well by maintaining numerical stability, avoiding over fitting with data variations, and standardizing data representation.

Improved Model Performance: Pre-processing optimizes raw data, facilitating more effective learning by deep learning models, ultimately resulting in higher accuracy and better predictive capabilities.

Enhanced Generalization and Robustness: By augmenting dataset diversity and stabilizing training processes, pre-processing techniques mitigate over fitting and improve model adaptability to new, unseen data, bolstering reliability in real-world applications.

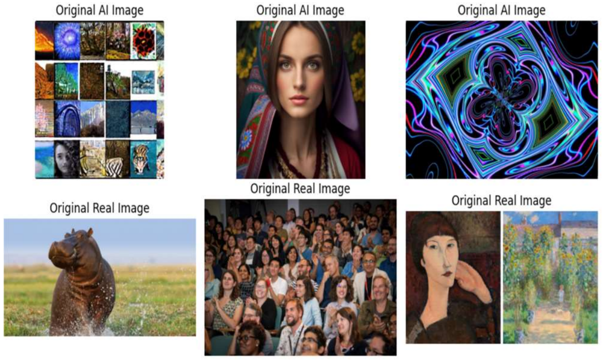


Fig 5: Before Data augmentation

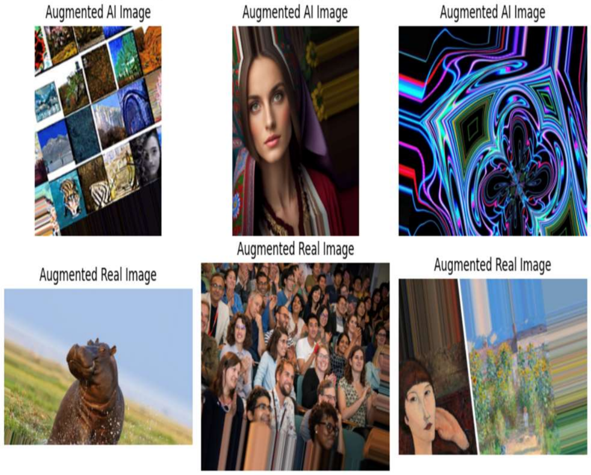


Fig 6: After data augmentation

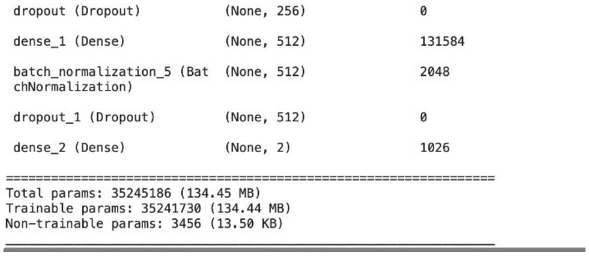
B. Training the model

Convolutional neural network (CNN) model:

A function is defined to create the Convolutional Neural Network model architecture. The model consists of multiple convolutional layers with max-pooling and batch normalization, followed by dense layers with dropout regularization. The output layer uses a softmax activation to provide probability scores for the two classes (AI art and real art).

Batch Size is 20, the batch size dictates the number of samples processed in each training iteration, balancing computational efficiency and model convergence.
With a predefined value of 30 epochs, the training process iterates over the entire dataset, allowing the model to progressively refine its parameters and enhance classification performance.

The model is compiled with the Adam optimizer and categorical cross-entropy loss. A model checkpoint callback is set up to save the best model weights based on validation accuracy. The training process is initiated using the fit function with the training and validation data generators. Loss and accuracy values for each epoch are logged during training.



```
 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 512)               131584

 batch_normalization_5 (Bat  (None, 512)               2048
 chNormalization)

 dropout_1 (Dropout)         (None, 512)               0

 dense_2 (Dense)             (None, 2)                 1026

=================================================================
Total params: 35245186 (134.45 MB)
Trainable params: 35241730 (134.44 MB)
Non-trainable params: 3456 (13.50 KB)
_________________________________________________________________
```

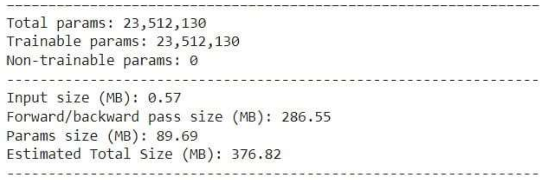
Fig 7: Total parameters

After training, the best model weights are loaded from the checkpoint, and the final accuracy is 78 % evaluated on the test data. The trained model is saved as an HDF5 file.

ResNet50 model:

ResNet-50 model, a highly effective convolutional neural network architecture for image classification. By utilizing pre-trained weights from Image Net, The input the model is adapted for the specific binary classification task by replacing its top fully connected layer with a new linear layer tailored for binary classification. To streamline training, the parameters of the pre-trained layers are frozen, ensuring that only the new classification layer is trainable, thereby optimizing computational resources and accelerating convergence.

The model architecture is instantiated using the pre-trained ResNet-50 model provided by torch vision. The fully connected layer at the top of the ResNet-50 model is replaced with a new linear layer consisting of 2 output neurons, corresponding to the two classes in the dataset. This modification enables the model to perform binary classification.



```
----------------------------------------------------------------
Total params: 23,512,130
Trainable params: 23,512,130
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.57
Forward/backward pass size (MB): 286.55
Params size (MB): 89.69
Estimated Total Size (MB): 376.82
----------------------------------------------------------------
```

Fig 8: Total parameters

The training process involves iterating through 50 epochs, each comprising both training and validation phases. During training, the model's parameters are optimized using stochastic gradient descent (SGD) with momentum. Learning rate is 0.0001 scheduling is employed to adjust the learning rate dynamically. Early stopping is used in this model, but total epochs are running and give good accuracy of 83%.

InceptionV3 model :

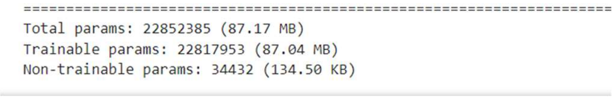
Before training the model , loading the InceptionV3 model with pre-trained weights of ImageNet and utilized as a base model. Model is set up to not include its top layer, as specified by the include_top=False parameter. The input images are 299 * 299 pixels in size, with three color channels.

Subsequently, for further feature extraction and classification additional layers are added on the top of base model. These new layers include a max pooling layer which helps to make the images smaller in size , and global average pooling layer , which makes the data simpler by finding the average of each feature map. Also, a dropout layer is included with a dropout of 0.5. Following that , there is a dense layer with 512 neurons and ReLU activation, this dense layer is also regulated using L_2 regularization parameter of 0.001 to prevent overfitting .Lastly , a dense output layer with sigmoid activation is added for binary classification, distinguishing between AI generated and Real images.[6]

The final model, which includes both the original InceptionV3 model and the additional layers, is created by using the Model class. This shows how data moves through the model .It sets up the inputs and outputs. To enable fine tuning of the model , the top layers to layer 249 are frozen , preventing their weights from being updated during training, while the layers starting from layer 250 are unfrozen , allowing their weights to be adjusted.

Once model architecture is setup, it is compiled using Adam optimizer, which helps optimize the model's performance during training . The learning rate is 0.0001 for evaluating model's performance , binary cross- entropy is used as the loss function, and accuracy is used as the metric to measure.

Finally, two callbacks are defined to monitor the validation loss during training. The EarlyStopping callback ,it monitors the validation loss during training and waits for 10 epochs. If the validation loss does not improve for 10 epochs, training will stop early. The ReduceLROnPlateau callback lowers the learning rate if the validation loss does not improve after a certain number of tries.



```
=================================================================
Total params: 22852385 (87.17 MB)
Trainable params: 22817953 (87.04 MB)
Non-trainable params: 34432 (134.50 KB)
```

Fig 9: Total parameters of the model

The model is trained using a generator on the training data for 50 epochs, with early stopping and learning rate reduction callbacks. It evaluates the performance on the test data, calculating the test loss and accuracy.

After training the model the accuracy is 0.8256. The training is stopped at epoch 13 by the early stopping because there is no progress.

VI. Results

In this section, we present the outcomes of our experiments utilizing three distinct convolutional neural network (CNN) architectures: Convolutional Neural Network (CNN), ResNet50, and Inception V3. We commence by furnishing an outline of the performance metrics attained by each model, followed by visualizations of accuracy and loss trends. Subsequently, we provide comprehensive classification reports encompassing confusion matrices, precision, recall, and F1-scores.

A. Model Performance Metrics:

The table below summarizes the accuracy and loss metrics obtained for each model:

Table 1

Model	Accuracy
CNN	78%
ResNet50	83%
Inception V3	83%

Accuracy Plots:

The following figures display the accuracy achieved by each model over epochs:

CNN:

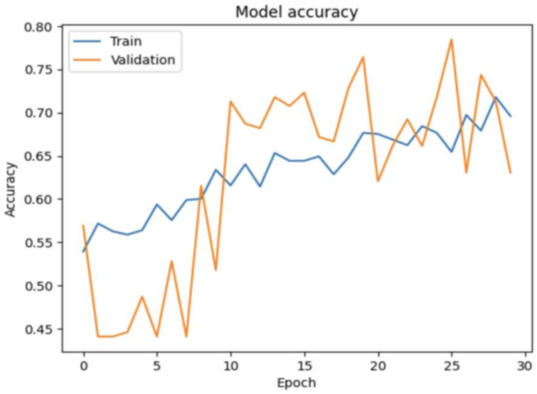


Fig 10: Accuracy plot for CNN

The plot above illustrates a consistent upward trajectory, signifying enhanced model performance with increasing training epochs, albeit displaying some leveling off towards the conclusion. This trend hints that the model could potentially derive advantages from further regularization methods or adjustments in its architecture.

ResNet50:

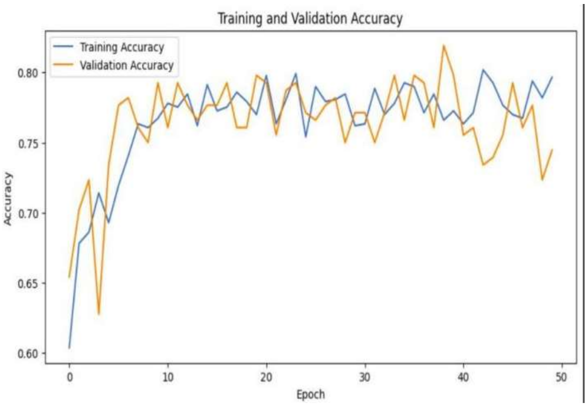


Fig 11: Accuracy plot for ResNet50

The accuracy plot above exhibits a more gradual and uniform ascent compared to CNN, presumably due to the residual connections addressing the vanishing gradient issue. This trend implies that there is still room for enhancement through continued training.

Inception V3:

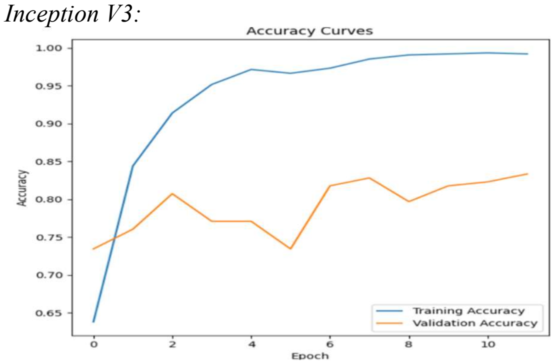


Fig 12: Accuracy plot for InceptionV3

The accuracy plot depicted above follows a trajectory akin to that of ResNet50, displaying a consistent and gradual rise in accuracy over epochs. This observation corresponds with the recognized capabilities of the Inception architecture in effectively utilizing multi-scale feature extraction.

Loss Plots:
The following figures illustrate the loss incurred by each model during training:

CNN:

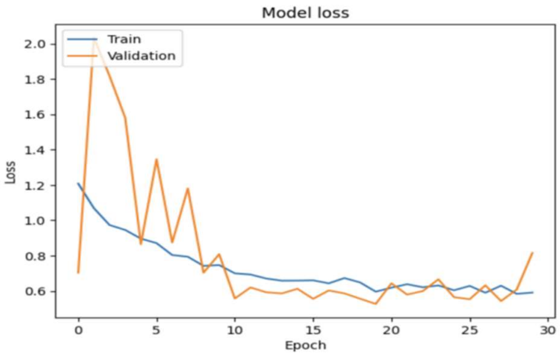


Fig 13: Loss plot for CNN

The loss plot above illustrates a progressive decline over time, correlating with the model's increasing accuracy throughout training. Nonetheless, towards the conclusion, the curve seems to level off, possibly suggesting convergence or overfitting concerns.

ResNet50:

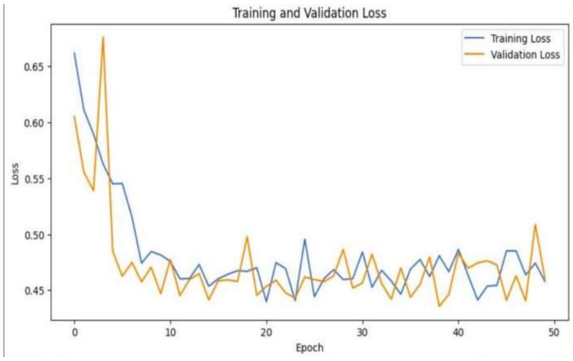


Fig 14: Loss plot for ResNet50

Unlike the CNN, the loss plot for ResNet50 exhibits a smoother and more uniform decrease over epochs, mirroring the steady rise observed in the accuracy plot. This implies more stable and efficient training dynamics enabled by the residual connections.

Inception V3:

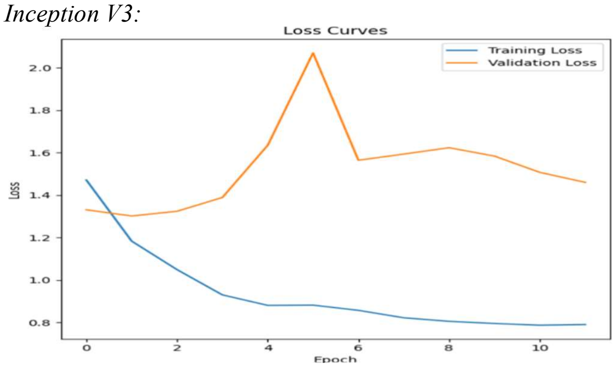
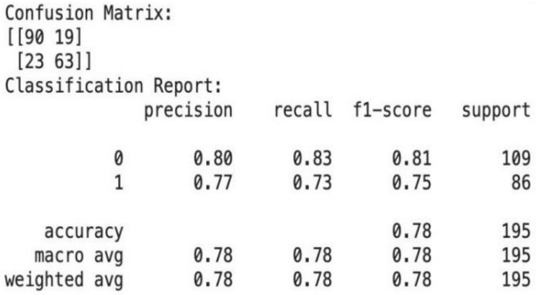


Fig 15: Loss plot for InceptionV3

Like the ResNet50 model, the Inception V3 loss plot exhibits a consistent and smooth decrease in loss as the training progresses, indicating that the model is effectively learning from the training data. The plot suggests that the model's loss may continue to decrease with further training epochs.

B. Classification Reports:
For evaluating the models' classification performance, we examined the validation classification reports.

CNN:

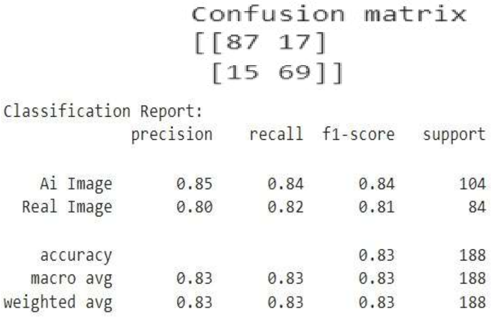


```
Confusion Matrix:
[[90 19]
 [23 63]]
Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.83      0.81       109
           1       0.77      0.73      0.75        86

    accuracy                           0.78       195
   macro avg       0.78      0.78      0.78       195
weighted avg       0.78      0.78      0.78       195
```

Fig 16: CNN Classification report and confusion matrix

Above confusion matrix and classification report suggest a binary classification problem with a reasonable overall accuracy of 0.78, but some room for improvement in correctly classifying the Real images as indicated by the lower precision and recall scores for that class.

ResNet50:

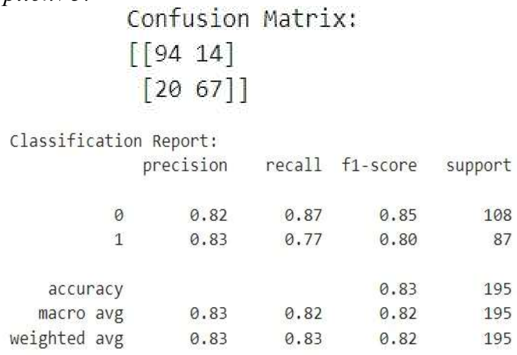


```
Confusion matrix
[[87 17]
 [15 69]]
Classification Report:
              precision    recall  f1-score   support

    Ai Image       0.85      0.84      0.84       104
  Real Image       0.80      0.82      0.81        84

    accuracy                           0.83       188
   macro avg       0.83      0.83      0.83       188
weighted avg       0.83      0.83      0.83       188
```

Fig 17 : ResNet50: Classification report and confusion matrix

The confusion matrix indicates a reasonable performance in classifying the two classes, while the classification report shows balanced precision and recall scores for both "AI Image" and "Real Image" classes, resulting in an overall accuracy of 0.83.

InceptionV3:



```
Confusion Matrix:
[[94 14]
 [20 67]]
Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.87      0.85       108
           1       0.83      0.77      0.80        87

    accuracy                           0.83       195
   macro avg       0.83      0.82      0.82       195
weighted avg       0.83      0.83      0.82       195
```

Fig 18:InceptionV3 Classification report and confusion matrix

The above confusion matrix shows a good performance in classifying the Ai images, but some difficulty in correctly identifying the Real images, which is further reflected in the classification report with a lower recall score of 0.77 for class 1 compared to 0.87 for class 0, while maintaining an overall accuracy of 0.83.

Overall, the CNN model achieved a decent accuracy of 0.78 but struggled with the Real images. Both ResNet50 and Inception V3 outperformed CNN with higher overall accuracy of 0.83 and better handling of class imbalance. Inception V3 showed slightly higher precision while ResNet50 had marginally better recall for the Real images. The residual and inception architectures proved more effective for this classification task.

VII.DISCUSSION

The experiments carried out utilizing various convolutional neural network architectures—specifically CNN, ResNet50, and Inception V3—highlight the complexities encountered in achieving precise image classification owing to the intrinsic intricacy and variability of visual data. Although the models attained reasonably high overall accuracy, their effectiveness varied across different classes, particularly when confronted with class imbalances and rare categories.

The CNN model, despite its straightforwardness, exhibited a commendable overall accuracy of 78%. However, it faced challenges in accurately classifying minority classes, as evidenced by lower precision and recall scores. This discrepancy in performance among classes underscores the complexities associated with managing class imbalances, where certain categories possess notably fewer instances than others.

It's crucial to acknowledge that the intricacies and diversities within visual data extend beyond class imbalances Variations in lighting conditions, perspectives, and the presence of diverse objects or backgrounds can have a notable impact on model performance. Continuous progress in domains like data augmentation, domain adaptation, and multi-task learning is crucial for bolstering the robustness and generalization abilities of image classification models.

VIII. CONTRIBUTIONS

Table 2

Individual Task Contributions	Name
CNN model: Data preprocessing, model training, evaluation, and analysis	Aruna
ResNet50: Data preprocessing, model training, evaluation, and analysis	Bharathi
InceptionV3:Data preprocessing, model training, evaluation, and analysis	Navya
Collaborative Contributions	Dataset Selection, Survey , Project Presentation and Final Project Report

IX. Conclusion

In conclusion, the CNN, ResNet50, and Inception V3 models showcased competitive performance in classifying the dataset. While CNN demonstrated balanced metrics, ResNet50 and Inception V3 exhibited superior accuracy and overall performance. The confusion matrices and various plots provided insights into the classification performance, training behavior, and potential issues like over fitting for each model. Overall, the analysis suggests that ResNet50 and Inception V3 are the most accurate models for the given task and dataset.

References

[1] G. E. Bartos and S. Akyol, "Deep Learning for Image Authentication: A Comparative Study on Real and AI-Generated Image Classification," in Proc. Conference, Nov. 2023.

[2] W. Quan, K. Wang, D. -M. Yan and X. Zhang, "Distinguishing Between Natural and Computer-Generated Images Using Convolutional Neural Networks," in IEEE Transactions on Information Forensics and Security, vol. 13, no. 11, pp. 2772-2787, Nov. 2018, doi: 10.1109/TIFS.2018.2834147.keywords: {Nickel;Training;Visualization;Image forensics;Videos;Feature extraction;Image forensics;natural image;computer-generated image;convolutional neural network;robustness;local-to-global strategy;visualization},

[3] R. Purohit, Y. Sane, D. Vaishampayan, S. Vedantam and M. Singh, "AI vs. Human Vision: A Comparative Analysis for Distinguishing AI-Generated and Natural Images," 2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 2024, pp. 1-7, doi: 10.1109/ICAECT60202.2024.10469620. keywords: {Computational modeling;Finance;Cyberbullying;Libraries;Internet;Fraud;Artificial intelligence;AI images;Image Classification;CNN},

[4] Bhinge, Shivani Atul, and Piyush Nagpal. "Quantifying the Performance Gap between Real and AI-Generated Synthetic Images in Computer Vision." (2023). SSRN Electronic Journal.

[5] J. J. Bird and A. Lotfi, "CIFAKE: Image Classification and Explainable Identification of AI-Generated Synthetic Images," in IEEE Access, vol. 12, pp. 15642-15650, 2024, doi: 10.1109/ACCESS.2024.3356122. keywords: {Artificial intelligence;Visualization;Data models;Image recognition;Computational modeling;Synthetic data;Image classification;AI-generated images;generative AI;image classification;latent diffusion},

[6] https://medium.com/@armielynobinguar/simple-implementation-of-inceptionv3-for-image-classification-using-tensorflow-and-keras-6557feb9bf53