DeepInsight:

Navigating the realm of Deepfake detection

Barnana Ganguly

Abhijit Anil

Amey Ghate

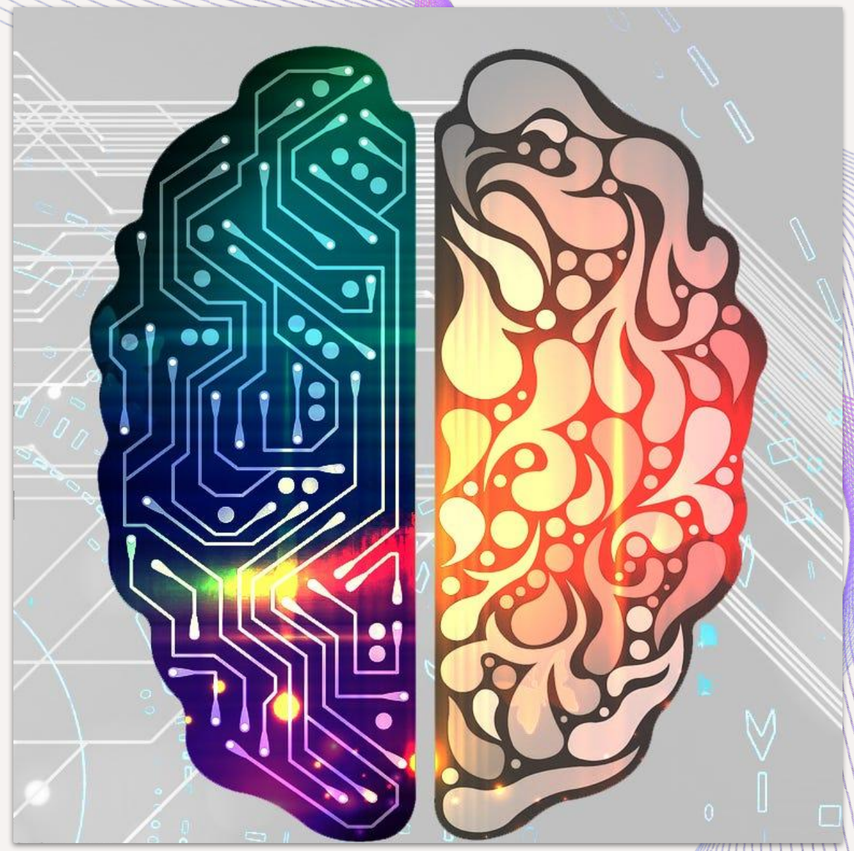
Rathil Madihalli

Sankalp Kulkarni

Shrishty Mishra

Agenda

1. Introduction to Deepfakes
2. Why are Deepfakes dangerous?
3. Data
4. Methodology and Models
5. Results
6. Comparison of Results
7. Conclusion
8. Next Steps



Introduction to Deepfakes

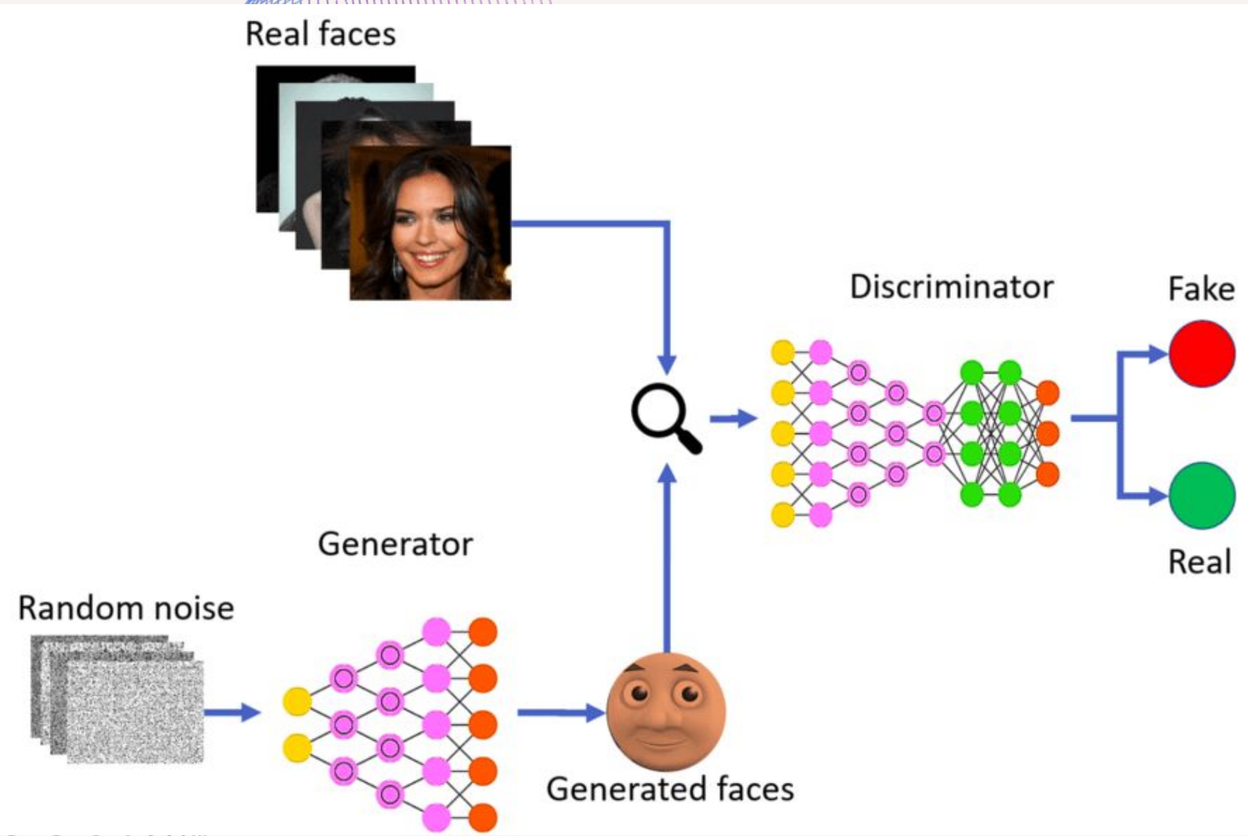
What is a Deepfake?

Deepfakes are synthetic media where advanced artificial intelligence, specifically Generative Adversarial Networks (GANs), is used to replace or manipulate a person's likeness in video or audio content. This technology can create realistic videos or audio recordings that convincingly depict individuals saying or doing things they never actually did.

How are they created?

One of the core technologies used to create deepfakes is Generative Adversarial Networks (GANs)

The generator creates images or videos, while the discriminator evaluates their authenticity. This rivalry drives improvements, enhancing the realism of generated content.



The core of GANs lies in their iterative training process

The generator learns to produce increasingly realistic data, while the discriminator becomes better at detecting fakes.

This adversarial process ensures that GANs can create highly convincing deepfakes by mimicking the subtleties of human expressions and environments

Why are Deepfakes dangerous?

Potential Concerns

Accessibility and Advancement: The evolution of AI, particularly through Generative Adversarial Networks (GANs), has made creating realistic deepfakes more accessible, requiring fewer resources and less effort

Social Media's Role: The rapid spread and viral nature of deepfakes on social media amplify their reach, making them a powerful tool for humor, commentary, and misinformation

Public Awareness and Media Attention: High-profile deepfakes have heightened awareness about the technology, sparking debates on its ethical use, potential for misinformation, and the need for legal frameworks

Dangers and Ethical Concerns: While deepfakes hold promise for entertainment and creative applications, they also pose significant risks related to misinformation, privacy violations, and the manipulation of public opinion, underscoring the need for ethical guidelines and protective regulations

Data

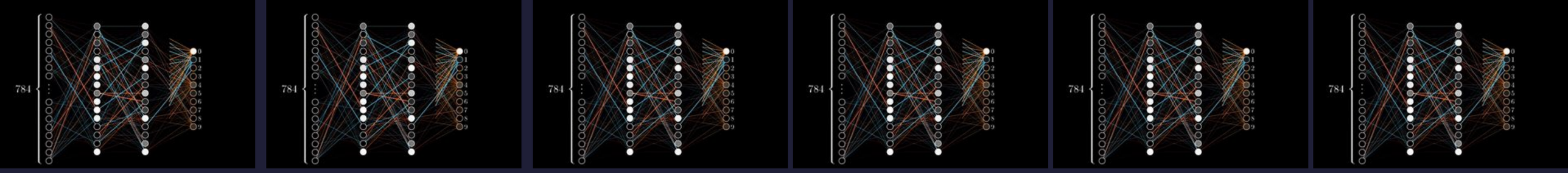
Data

We used Kaggle’s **140k Real and Fake Faces** dataset for our project

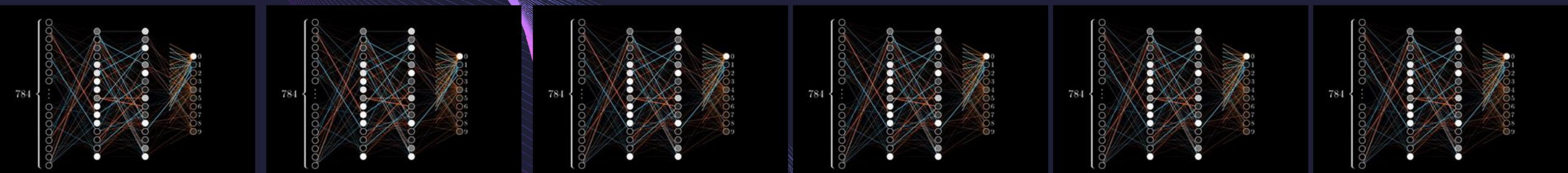
It consists of **70k real faces (from Flickr)** and **70k fake faces (GAN-generated)**

We used **100k images to train** the model and used **20k each for Validation and Test sets**

Methodology



The DenseNet Model

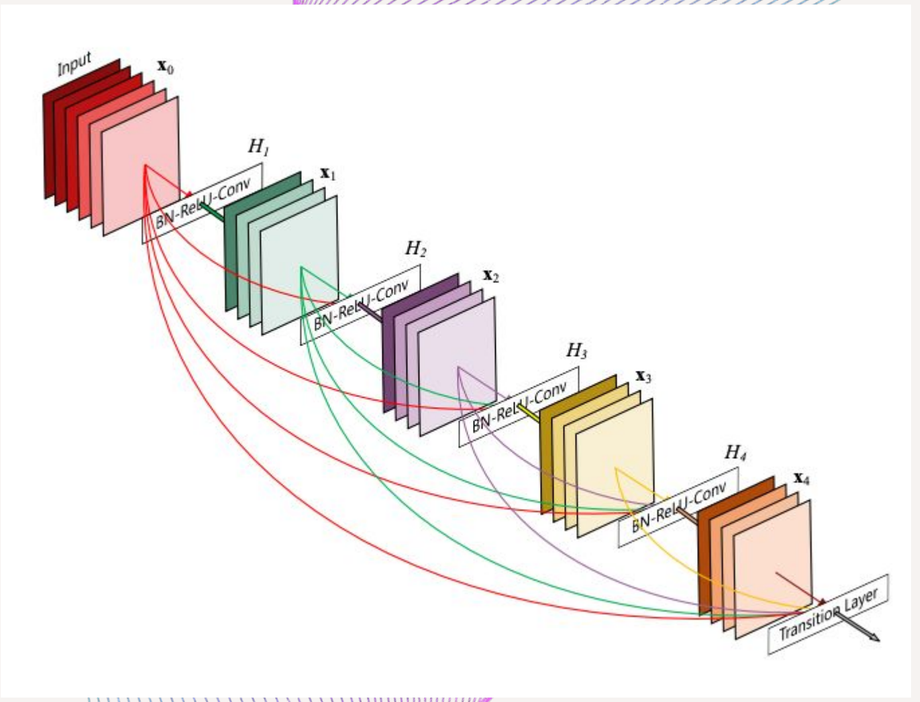


The DenseNet121 Model

DenseNet121, short for Dense Convolutional Network with 121 layers, is a deep learning model that has shown significant effectiveness in image recognition tasks and is a part of the DenseNet family

Dense Connectivity: Unlike traditional CNNs that connect layers sequentially, DenseNet121 connects each layer to every other layer in a feed-forward fashion. This dense connectivity pattern ensures maximum information flow between layers in the network.

Features and Architecture: The architecture is divided into dense blocks and transition layers. Dense blocks contain layers that are densely connected, and transition layers are used to reduce the dimensions of the feature maps.

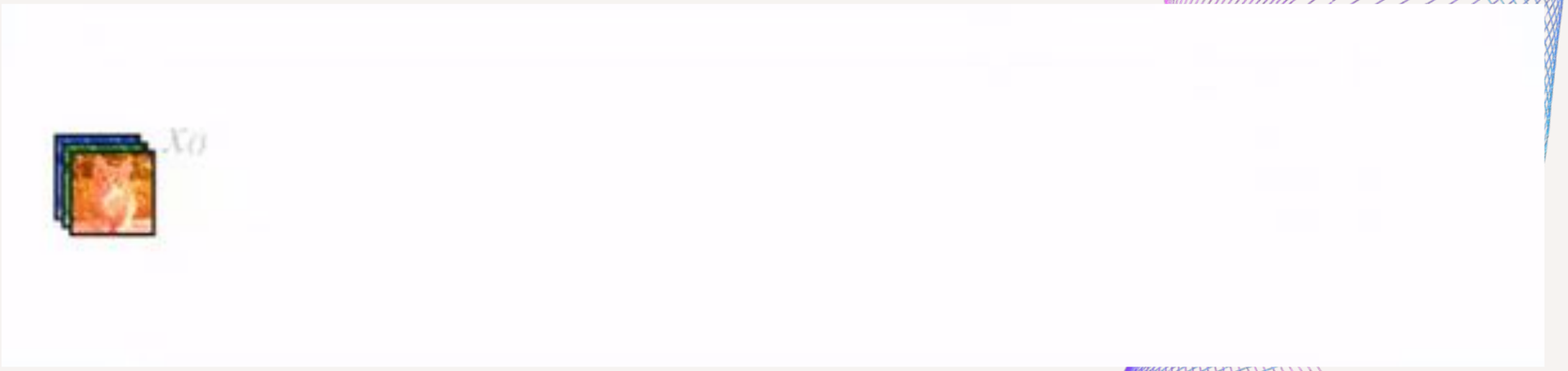


It uses composite functions that include Batch Normalization, ReLU activation, and Convolution followed by a pooling layer to reduce the size of the feature maps.

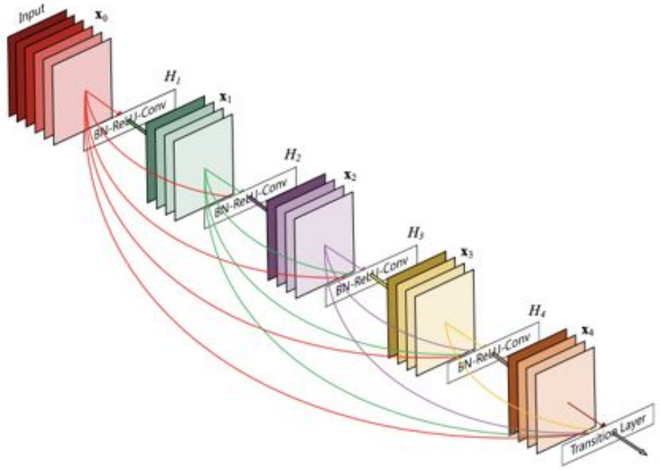
Advantages of DenseNet

Unlike traditional convolutional networks, DenseNet has several compelling advantages:

1. **Parameter Efficiency:** Requires fewer parameters than traditional CNNs, making it more efficient and reducing the risk of overfitting
2. **Improved Feature Propagation:** Due to dense connectivity, features from earlier layers can be reused in later layers, improving the model's ability to learn detailed features
3. **Feature Reuse:** Encourages feature reuse, making the network more efficient and reducing the number of parameters
4. **Better Gradient Flow:** Facilitates the backpropagation of gradients, making the network easier to train

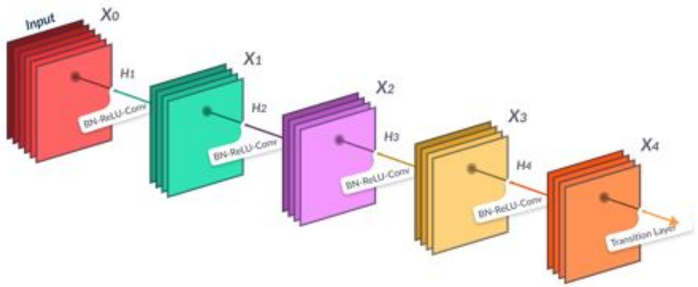


DenseNet vs ResNet



DenseNet Structure

$$a^{[l]} = g\left([a^{[0]}, a^{[1]}, a^{[2]}, \dots\dots\dots\dots, a^{[l-1]}\right)$$



ResNet Structure

$$a^{[l]} = g\left(z^{[l+1]} + a^{[l]}\right)$$

In **DenseNet**, every layer is directly connected to all subsequent layers, increasing feature reuse

In **ResNet**, Layers have skip connections that bypass one or more layers, which mitigates the vanishing gradient problem

DenseNet is also more parameter-efficient due to feature reuse, leading to a reduction in the total number of parameters

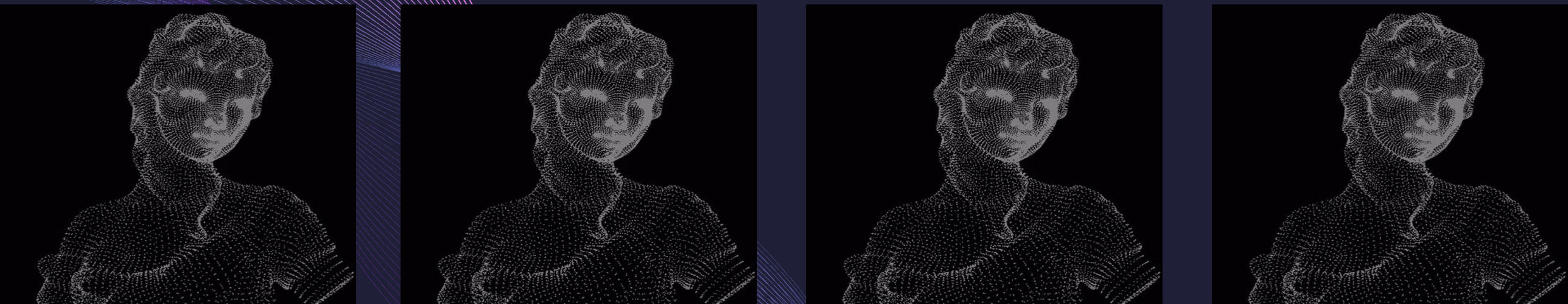
DenseNet also works best when model size and computational efficiency are crucial

Core Model Architecture

At its core, the model leverages the DenseNet121 convolutional neural network as a powerful feature extractor, renowned for its efficiency and depth. Here are the critical details that define our model's processing and learning capabilities:

Feature Extraction with DenseNet121	**The backbone of architecture, DenseNet121, operates without pre-loaded weights, ensuring that every feature it learns is directly relevant to our specific challenge of identifying deepfakes.**
Global Average Pooling 2D (GAP)	**Following the DenseNet121 layer, the GAP layer reduces each feature map to a single summary value. This step is crucial for condensing the feature information into a more manageable form, focusing on the most salient features extracted from the input images**
Dense Output Layer with Sigmoid Activation	**The architecture culminates in a dense layer with a single neuron. This layer utilizes a sigmoid activation function to produce a probability score, effectively classifying the input image as real or fake.**
Training Parameters and Batch Processing	**To balance performance and computational expenses, we trained the model with a batch size of 32 for 20 epochs using the Adam Optimizer**

The DenseNet Model with Augmented Images



Data Augmentation

These augmentations were carefully selected to improve the robustness of our base DenseNet model, and enhance its ability to generalize from the training data to unseen images by diversifying the visual features it encounters during training

Rescaling	**Normalizes pixel values across images to a 0-1 range, ensuring consistent input scale for the model.**
Rotation	**Randomly rotates images within a specified range (e.g., 20 degrees) to simulate variations in camera angle.**
Horizontal Flip	**Mirrors images horizontally, useful for simulating different perspectives**
Shear Range	**Applies shear transformations, skewing the image to simulate a more diverse set of angles and perspectives**
Zoom Range	**Randomly zooms inside the pictures, allowing the model to focus on different parts of the image and improving its ability to recognize features at various scales**

The DenseNet Model with Grayscale Images

DenseNet with Grayscale

Our objective was to explore the influence of color on differentiating authentic images from manipulated ones.

So we adapted our DenseNet model for grayscale analysis which involved converting all input images to grayscale, utilizing the 'color mode' setting

Despite this modification, we retained the original DenseNet layers, ensuring a consistent baseline for comparison.

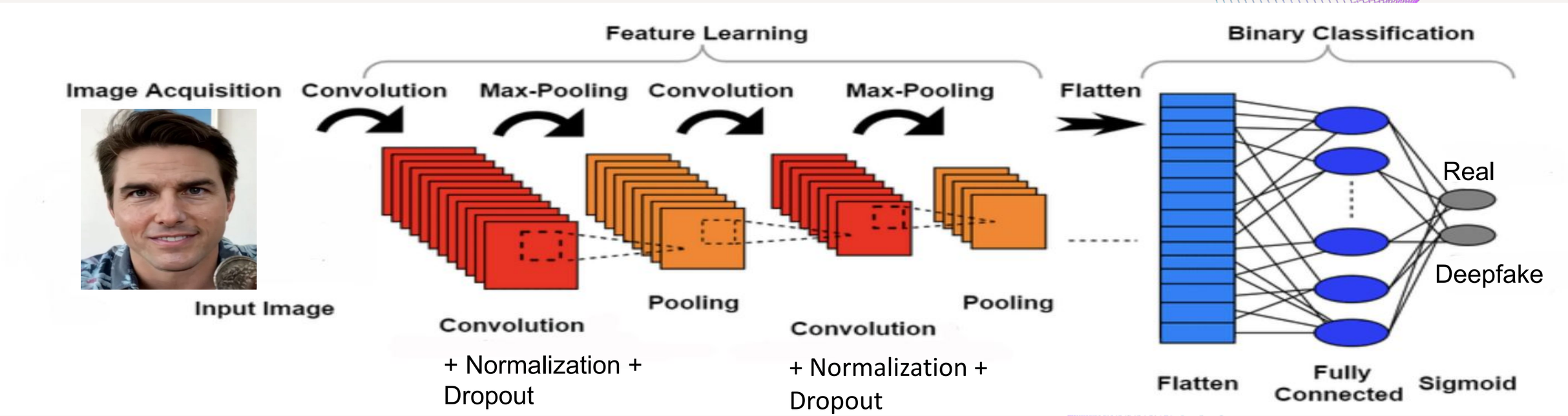
This adjustment serves as an additional layer of augmentation, enriching our investigation into the model's adaptability and the role of color in image authentication.

The Custom CNN Model

Model Overview

This architecture leverages the power of CNNs to effectively identify deepfake content, combining feature extraction, normalization, and regularization techniques to achieve high accuracy

1. **Input Shape:** (224, 224, 3) - Tailored for processing standard RGB images
2. **Layers:** Sequential arrangement of Convolutional, Max Pooling, Batch Normalization, Dropout, and Global Average Pooling layers to extract and refine features
3. **Output:** Binary classification (Real or Deepfake) with a sigmoid activation function



Key Components

Convolutional Layers: Extracts a hierarchy of visual features starting from simple edges to complex textures and patterns, using filters of sizes progressively increasing from 16 to 512

Pooling and Batch Normalization: Reduces dimensionality and standardizes the activations, respectively, improving the model's efficiency and stability

Dropout: Mitigates overfitting by randomly omitting subsets of features at each iteration during training

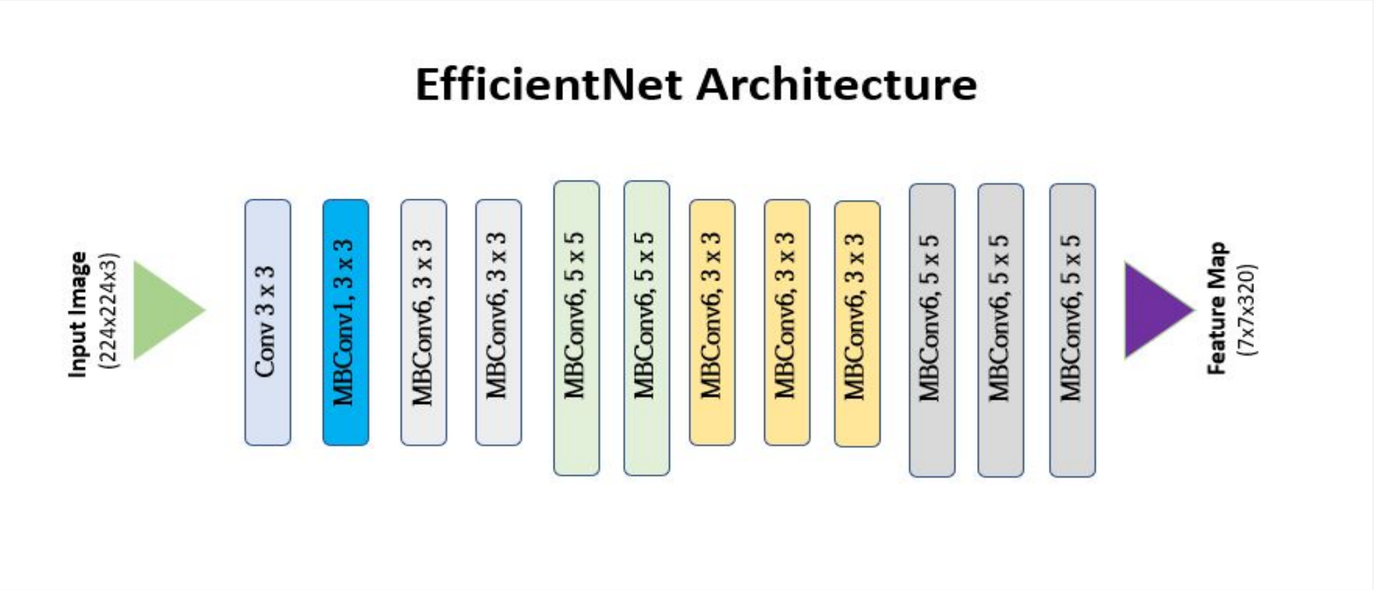
Global Average Pooling: Condenses the feature maps into a single global feature for each map, reducing the total number of parameters and hence the risk of overfitting

Final Layer: Dense layer with 1 unit and a sigmoid activation function to output a probability score indicating the likelihood of the image being a deepfake

Other Models

Challenges faced

- We also experimented with EfficientNet for deepfake detection, intrigued by its state-of-the-art efficiency and scalability
- However, we encountered challenges in achieving optimal performance due to our limited ability to scale the network
- EfficientNet requires significant expansion in depth and width to improve, which in turn demands more computational resources
- Our current GPU hardware limitations prevented us from making these necessary adjustments, leading us to focus on DenseNet and our custom CNN model instead.



EfficientNet's architecture is based on a **compound scaling** method that uniformly scales the network's depth, width, and resolution with a set of fixed scaling coefficients

This scalability enables the deployment of more powerful models capable of detecting subtle manipulations in images and videos that are characteristic of deepfakes

Results

DenseNet (Base Model)

- DenseNet model achieves remarkable performance in deepfake detection:
  - **ROC AUC score**: 0.9975
  - **AP score**: 0.9974
- The model showed balanced and accurate detection of real (**precision**: 0.94, **recall**: 0.99) and deepfake images (**precision**: 0.99, **recall**: 0.94)
- Overall high **accuracy** (0.97) and consistent metrics (precision, recall, f1) demonstrate the robustness of our approach for deepfake detection

```
ROC AUC Score: 0.9975173699999998
AP Score: 0.9973757115620031

              precision    recall  f1-score   support

           0       0.94      0.99      0.97     10000
           1       0.99      0.94      0.96     10000

    accuracy                           0.97     20000
   macro avg       0.97      0.97      0.97     20000
weighted avg       0.97      0.97      0.97     20000
```

The DenseNet base model showcases exceptional efficacy in deepfake detection with remarkable performance metrics and balanced detection capabilities.

DenseNet with Image Augmentation

- Implementation of image augmentations in our DenseNet-based deepfake detection model yields promising outcomes:
  - **ROC AUC score**: 0.9686
  - **AP score**: 0.967
- The model excels at finding deepfakes (**precision**: 0.83, **recall**: 0.97, **f1**: 0.90) while maintaining good accuracy for real images (**overall accuracy**: 0.89)
- These results suggest the model generalizes well to unseen data and is robust against variations in images

```
ROC AUC Score: 0.968640695
AP Score: 0.966609199565477

              precision    recall  f1-score   support

           0       0.96      0.81      0.88     10000
           1       0.83      0.97      0.90     10000

    accuracy                           0.89     20000
   macro avg       0.90      0.89      0.89     20000
weighted avg       0.90      0.89      0.89     20000
```

Inclusion of image augmentations slightly worsens model performance in detecting deepfake content

Model demonstrates solid precision, recall, and accuracy, validating its effectiveness in deepfake detection

DenseNet with Grayscale Images

- DenseNet with grayscale shows incredible results in deepfake detection:
  - **ROC AUC score**: 0.9980
  - **AP score**: 0.9978
- The model accurately and precisely identified both real images (**precision**: 0.99, **recall**: 0.97) and deepfakes (**precision**: 0.97, **recall**: 0.99).
- Consistent metrics (precision, recall, f1) coupled with extremely overall high **accuracy** (0.98) exhibit the robustness of our approach

```
ROC-AUC Score: 0.9980781299999999
AP Score: 0.9978361434792907

              precision    recall  f1-score   support

           0       0.99      0.97      0.98     10000
           1       0.97      0.99      0.98     10000

    accuracy                           0.98     20000
   macro avg       0.98      0.98      0.98     20000
weighted avg       0.98      0.98      0.98     20000
```

The DenseNet with Grayscale show remarkable efficacy in deepfake detection with exceptional performance metrics and balanced detection capabilities.

This model has the **best accuracy,precision and recall** amongst all the models**.**

Custom CNN

- Custom Convolutional Neural Network (CNN) delivers outstanding deepfake detection results:
  - **ROC AUC score**: 0.9919
  - **AP score**: 0.9914
- The model precisely identified both real images (**precision**: 0.92, **recall**: 0.98) and deepfakes (**precision**: 0.97, **recall**: 0.92).
- This translates to a robust and accurate overall performance (**accuracy**: 0.95) for deepfake detection.

```
ROC AUC Score: 0.9919095450000001
AP Score: 0.9914724911124952

              precision    recall  f1-score   support

           0       0.92      0.98      0.95     10000
           1       0.97      0.92      0.94     10000

    accuracy                           0.95     20000
   macro avg       0.95      0.95      0.95     20000
weighted avg       0.95      0.95      0.95     20000
```

Custom CNN demonstrates robustness and accuracy in classifying both authentic and Deepfake images

High precision, recall, and F1-score metrics validate the effectiveness of the model in identifying and mitigating deepfake content

Results Comparison

Model Comparison

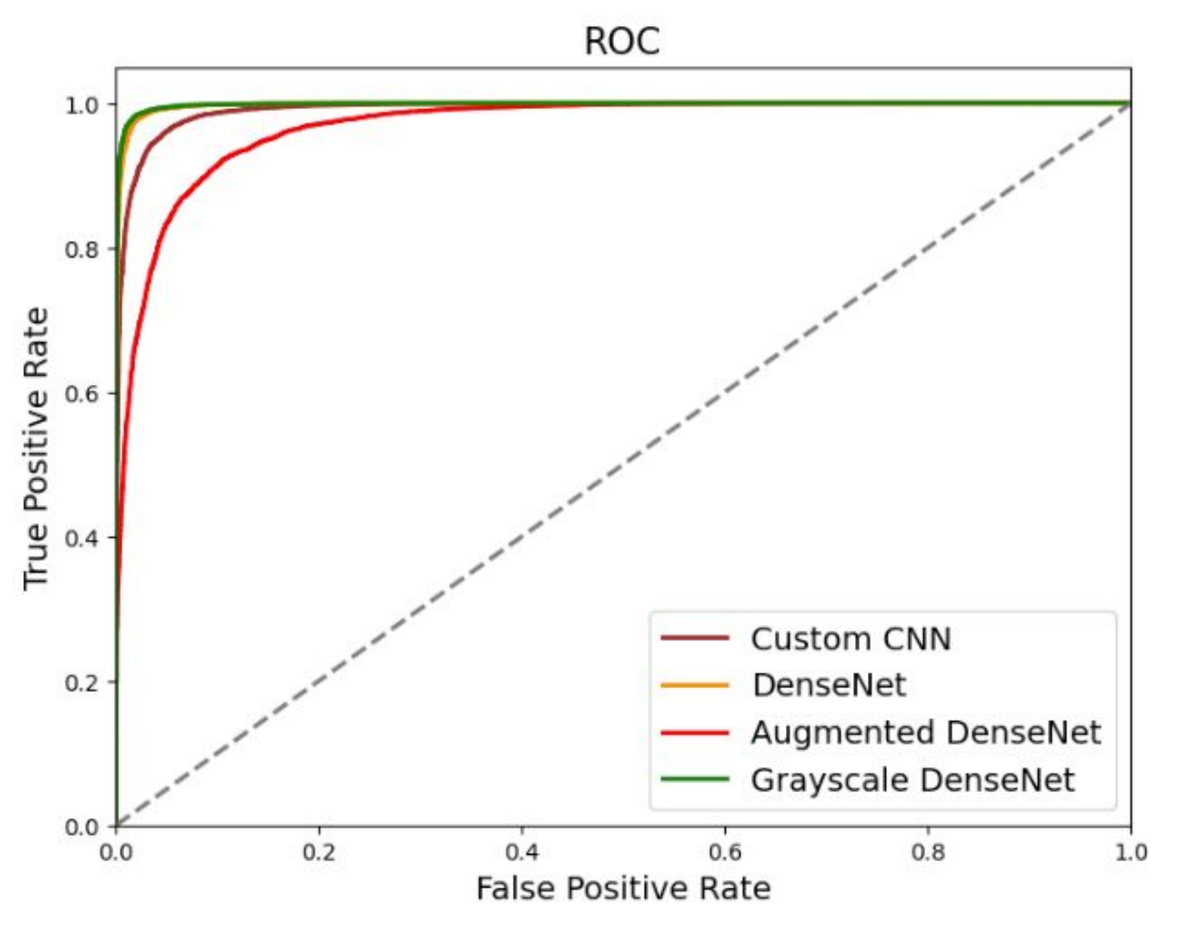
Model	Accuracy	Precision	Recall	F1-Score
DenseNet Model (Base Model)	**0.97**	**0.97**	**0.97**	**0.97**
DenseNet Model with Image augmentations	**0.89**	**0.90**	**0.89**	**0.89**
DenseNet Model with Grayscale Images	**0.98**	**0.98**	**0.98**	**0.98**
Custom CNN	**0.95**	**0.95**	**0.95**	**0.95**

The original DenseNet model showcased exceptional performance but introducing image augmentations led to a slight performance dip

Transitioning to grayscale images significantly improved the model's performance indicating that color information might be less crucial for deepfake detection in this scenario

The custom CNN model, presents a robust alternative, slightly lagging behind the grayscale-enhanced DenseNet but outperforming the augmentation-influenced version. It signifies a solid middle ground, balancing complexity and performance.

The ROC curves

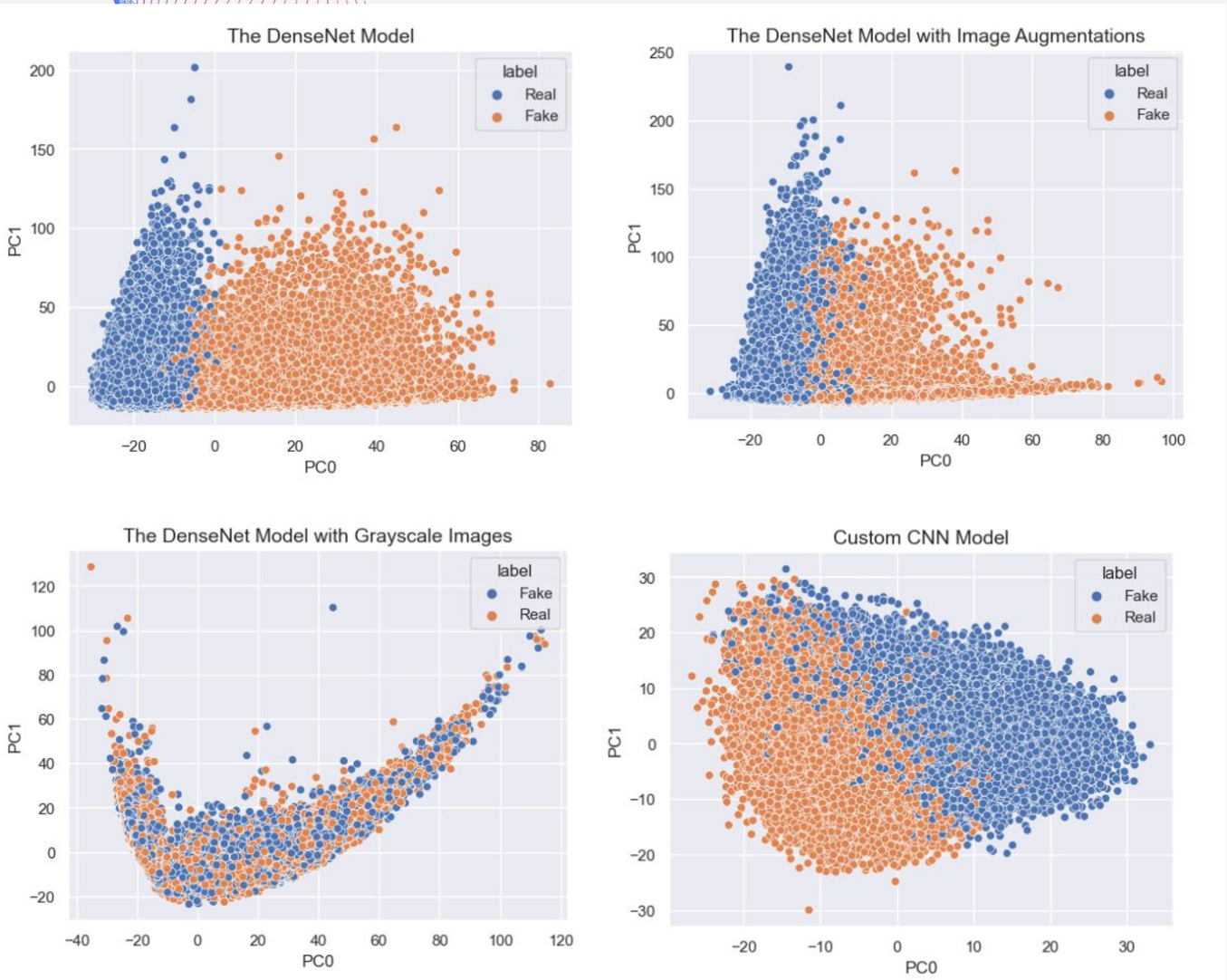


Grayscale Densenet's Superiority: Leverages texture over color, focusing on structural details crucial for spotting deepfakes, leading to better performance

Standard Densenet & Custom CNN: Lag behind grayscale due to less effective use of color information in detecting deepfakes

Densenet with Augmentations' Underperformance: Augmentations may introduce noise, leading to overfitting or poor generalization, hindering its ability to accurately detect deepfakes

Principal Component Plots



- Scatter plots of principal components for various models (except DenseNet with Grayscale) reveal their effectiveness in distinguishing between real and fake images, highlighting the success of the feature extraction process

- The first two principal components prominently display distinct clusters for real versus fake images, indicating the models' proficiency in differentiating between the two

- The clear separation between clusters in these plots suggests the models' strong potential in accurately classifying DeepFakes, enabling effective application of techniques like SVM.

Conclusion

Conclusion

In wrapping up our exploration of deepfake detection methodologies, we distill our findings into these pivotal insights that define our journey and future outlook:

DenseNet121's Superior Performance	Exceptional in deepfake detection, especially with grayscale images, showcasing high accuracy and ROC AUC scores
Image Augmentation vs. Grayscale	Augmentation reduced performance slightly, while grayscale images significantly improved detection, highlighting texture's importance over color
Model Comparisons	Original and grayscale DenseNet models outperformed others, suggesting grayscale processing is highly effective for deepfake detection
Technical and Hardware Challenges	Despite successes, challenges such as hardware limitations and the need for computational resources were noted, especially with models like EfficientNet. Future work could explore scaling and optimizing models further to enhance deepfake detection capabilities

Next Steps

Our project faced challenges due to limited computational resources, which affected our ability to enhance the robustness and generalization of our deepfake detection model. These constraints restricted the scale and complexity of both data processing and model training. However, gaining access to more computational power would offer us an opportunity to advance our efforts in four key areas:

1. **Efficient Architectures:** Leverage advanced, computation-intensive models like EfficientNet for superior performance in deepfake detection, utilizing their scalable architecture for improved robustness and generalization

2. **Varied Datasets:** Access and process larger, more diverse datasets, crucial for training our models on a broad spectrum of deepfake techniques and improving detection across various evolving methods

3. **Advanced Augmentation:** Implement a wider array of sophisticated data augmentation techniques to better simulate varying conditions, enhancing the model's ability to generalize and accurately identify deepfakes under diverse scenarios

4. **Longer Training:** Utilize the increased computational resources for more extensive training periods, including adversarial training, to continuously challenge and enhance the model's detection capabilities against a dynamically evolving set of deepfake examples.

Thank you!