DeepFake Detection Using Deep Learning

A Project report submitted in partial fulfilment

of the requirements for the award of a degree in

Master of Science

Data Science

By

KasiReddy Daggula

(2023002295)

Under the esteemed guidance of

Mr. Shaik Shahid

Assistant Professor

Department of Computer Science

GITAM School of Science

GITAM(Deemed to be University)

Visakhapatnam-530045, AP

(2024-2025)

CERTIFICATE

This is to certify that the project entitled **“DEEPFAKE DETECTION USING DEEP LEARNING”** is Bonafide work done by **KasiReddy Daggula,** Regd.No:2023002295 during December 2024 to April 2025 in partial fulfilment of the requirement for the award of the degree of **Master of Science, Data Science** in the Department of Computer Science, GITAM School of Science, GITAM (Deemed to be University), Visakhapatnam.

Internal Guide	**Head of the Department**
Mr. Shaik Shahid	**Dr. T. Uma Devi**
Assistant Professor	**Professor**
Dept. of Computer Science	**Dept. of Computer Science**

DECLARATION

I (**KasiReddy Daggula)** Regd. No:2023002295 hereby declare that the project entitled **“DEEPFAKE DETECTION USING DEEP LEARNING” (Project Title)** is an original work done in the partial fulfilment of the requirements for the award of the degree of **Master of Science, Data Science** in **GITAM School of Science, GITAM (Deemed to be University), Visakhapatnam.** I assure that this project work has not been submitted towards any other degree or diploma in any other colleges or universities.

KASIREDDY DAGGULA

(Regd. No:2023002295)

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without mentioning the people who made it possible and whose constant guidance and encouragement crown all the effects with success.

I would like to express my sincere gratitude to our honourable Principal and **Prof. K. Vedavathi**, GITAM School of Science, GITAM (Deemed to be University), for giving me an opportunity to work on this project.

I consider it as a privilege to express our deepest gratitude to **Prof. T. Uma Devi**, Head of the Department, Department of Computer Science for her valuable suggestions and constant motivation that greatly helped us to successfully complete the project work. I would like to thank my project guide **Mr. Shaik Shahid, Assistant Professor,** Dept. of Computer Science for his stimulating guidance and profuse assistance.

We would like to thank our Project Coordinator **Mr. B. Ravi Kumar, Associate Professor and Dr. K. Vanitha Assistant Professor** Dept. of Computer Science and in helping to complete the project by taking frequent reviews and for their valuable suggestions throughout the progress of this work.

I thank all the Teaching and Non-teaching Staff who has been a constant source of support and encouragement during the study tenure.

I thank all my friends who helped me in sharing their knowledge and support.

KASIREDDY DAGGULA

(Regd. No:2023002295)

Abstract

Deepfake technology, powered by advanced generative models, produces highly convincing synthetic media, posing significant challenges to digital authenticity. This study proposes a robust deepfake detection framework leveraging the strengths of DenseNet201 and InceptionV3, two convolutional neural network architectures known for their feature extraction capabilities. DenseNet201, with its dense connectivity pattern, enhances feature reuse and captures intricate patterns in manipulated visuals, while InceptionV3's multi-scale convolutional modules excel in identifying subtle discrepancies across diverse image resolutions. By integrating these models in an ensemble approach, the framework analyzes spatial and textural anomalies in video frames to distinguish authentic content from deepfake forgeries. Evaluated on benchmark datasets, the proposed method demonstrates high accuracy and resilience against sophisticated deepfake generation techniques. This research underscores the potential of combining DenseNet201 and InceptionV3 for scalable, real-time deepfake detection, addressing ethical and societal concerns surrounding synthetic media proliferation.

Keywords: Deepfake detection, DenseNet201, InceptionV3.

Index

1.Introduction

Deepfake technology refers to the use of machine learning algorithms to produce synthetic media that can convincingly mimic the appearance and conduct of real people. The term” deepfake” is a combination of” deep learning” and” fake,” pertaining to the fact that the technology uses deep learning algorithms to produce fake media[14]. Deepfakes can be used to produce realistic videos, images, and audio recordings that appear to feature real people saying or doing effects that they no way actually said or did. Deepfake technology has come increasingly advanced in recent times, with new algorithms and ways being developed to produce more realistic and satisfying fake media.[2,8] One of the most generally used ways is called” deep learning,” which involves training a neural network on a large dataset of real media to learn how to induce analogous media. Deep learning algorithms can be used to produce high- quality deepfakes that are delicate to distinguish from real media. The implicit uses of deepfake technology are wide- ranging and include both positive and negative operations. For illustration, deepfakes can be used in the film and entertainment industries to produce realistic special goods and computer- generated imagery. They can also be used in training simulations, similar as for aviators or surgeons, to produce realistic scripts that pretend real- life situations[8]. Still, deepfake technology also has negative operations, particularly in the realm of intimation and propaganda. Deepfakes can be used to spread false information or to manipulate public opinion by creating fake media that appears to feature real people saying or doing effects that they noway actually said or did. For illustration, deepfakes can be used to produce fake newspapers or videos that are designed to sway public opinion on a particular content or issue. One of the most high-profile uses of deepfake technology in recent times has been in the creation of fake pornography. Deepfakes can be used to superimpose the face of a real person onto the body of a porn actor, creating a fake videotape that appears to feature the real person engaging in sexual exertion. This type of deepfake has been used to kill and defame individualities, particularly women and celebrities[15]. The implicit pitfalls associated with deepfake technology have led to growing enterprises among policymakers and experts. In 2018, the US Congress passed the Deepfakes Responsibility Act, which authorized backing for exploration on deepfake discovery technology and criminalized the creation and dispersion of certain types of deepfakes[17]. Other countries, including Canada and Australia, have also introduced legislation to regulate the use of deepfake technology. The development of deepfake technology has also led to the creation of new tools and ways for detecting deepfakes. Experimenters and technology companies are working to develop algorithms and tools that can identify fake media by assaying visual and audio cues. For illustration, some deepfake detection algorithms dissect facial expressions and movements to identify inconsistencies that may indicate a deepfake[26]. Other algorithms use metadata analysis 5 to determine the authenticity of a media train, similar as the time and position where it was created[14,31]. Despite the implicit pitfalls associated with deepfake technology, experimenters and assiduity experts continue to explore new operations for the technology[8]. For illustration, deepfake technology can be used to produce more realistic virtual sidekicks or to ameliorate the quality of computer- generated imagery in videotape games and flicks[3]. still, it’s important that policymakers and experts work together to assure that the technology is developed and used responsibly and that acceptable safeguards are put in place to help its abuse[10].

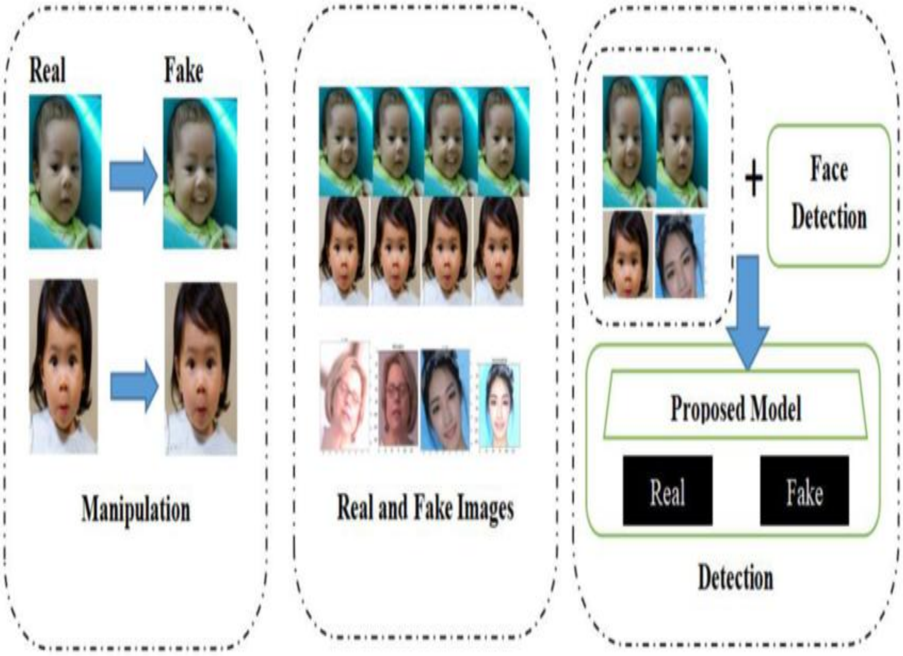


Figure 1. High Level view of deepfake creation and detection model [1]

The project falls within the broader field of Computer Vision and Artificial Intelligence (AI), with a specific focus on Deepfake Detection. This domain leverages machine learning (particularly deep learning) techniques to analyze and classify visual data—images in this case—to distinguish between authentic and synthetically generated content[15]. It addresses real-world challenges related to misinformation, digital security, and media authenticity.

The domain of **Deepfake Detection in Computer Vision and AI** is a perfect fit for a project using this Kaggle dataset. It leverages the dataset's focus on real vs. fake image classification, aligns with cutting-edge deep learning techniques (like transfer learning with ImageNet and DenseNet201), and addresses a timely societal challenge. If you'd like a more specific project outline (e.g., methodology, tools), or help refining the scope, let me know!

- **Dataset Alignment:** The dataset contains images labeled as "real" or "fake" (deepfake), directly supporting a binary classification task, a common problem in computer vision and AI.
- **Relevance:** Deepfake detection is a pressing issue in today's digital landscape, impacting areas like social media integrity, cybersecurity, and forensic analysis[19].
- **Technological Fit:** It aligns with transfer learning (e.g., using ImageNet-pre-trained models like MaxViT, as you mentioned earlier), convolutional neural networks (CNNs), or vision transformers, all of which are staples in this domain[6].

1.1.1 Problem statement

Through deepfake technology achieved thanks to advanced Generative Adversarial Networks (GANs) artificial intelligence models one can now generate hyper-realistic synthetic images and videos[28,29]. Used content that cannot be distinguished from the real thing threatens to create major risks regarding privacy together with security and public confidence. People become victims of impersonation attacks while disinformation circulates while elections get manipulated through false evidence fabrication resulting in damage from the personal to the international level. Deepfake technology has become accessible to all digital users since its recent democratization so its misuse has increased to match its growing popularity thus the necessity for robust detection systems remains urgent. The virtual environment requires authentic digital media because it serves as the base for documentation and communication thereby maintaining credibility[5].

Utility of existing CNN-based detection methods remains promising yet their effectiveness is extremely limited. The detection approaches depend on detecting specific local attributes such as pixel or facial anomalies which limits their ability to detect various datasets, high-quality forgeries and novel modification techniques[9]. The detection systems collapse when facing discrepancies between different platforms as well as when they encounter compressed media content or moderate deepfake manipulations. The current rapid growth of deepfake production makes existing detection methods inadequate so new methods must be developed to keep pace with new emerging threats. The present capabilities deficiency empowers researchers to develop complex systems similar to Multi-Axis Vision Transformer (DeneseNet201) which binds transformer-based global attentions with standard convolutions to discover subtle features together with widespread contextual relationships[32].

The main challenge involves developing a detection model with precision and enhanced scalability features alongside deployable computational speed. The attention mechanism within DenseNet201 solves the issue by examining images at multiple scales to enhance features and make forgery detection more effective[25]. The research employs DenseNet201 to address deepfakes danger and protect digital authenticity from expanding synthetic media threats.

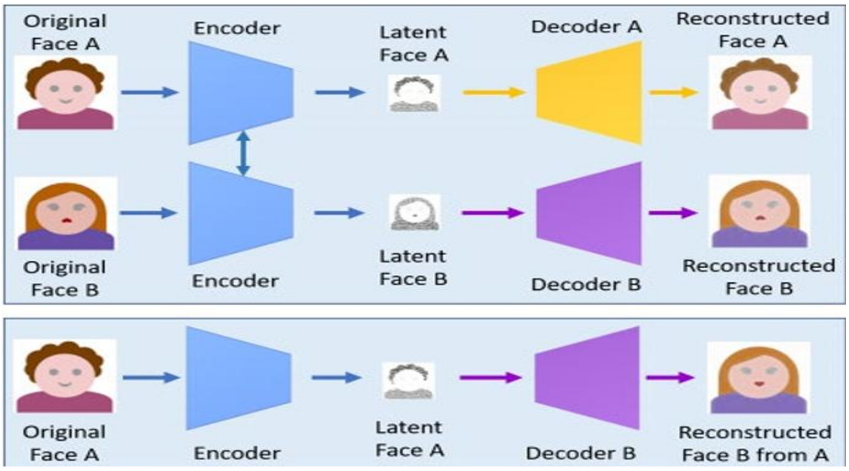


Figure 2.[32]

This Figure shows a Deepfake creation proces where the feature set of face A is connected with the decoder B to reconstruct face B from the original face A. This approach is applied in several works such as DeepFaceLab, DFaker, and DeepFake TensorFlow.

1.1.2Stats for deepfake

Since you've referenced the Kaggle dataset "Deep Fake Image Classification" and asked about ImageNet and transfer learning earlier, I'll assume you're interested in deepfake statistics relevant to your project domain—deepfake detection in computer vision and AI. Below, I'll provide a compilation of key deepfake statistics as of April 9, 2025, drawing from available data and trends up to this point. Note that exact stats for 2025 are projections based on prior growth rates, as real-time data collection for this year is ongoing.

Deepfake Statistics (Up to April 2025)

Prevalence and Growth:

- **Volume of Deepfakes:**
  - In 2023, approximately 500,000 video and voice deepfakes were shared on social media globally, according to Deep Media.
  - Projections suggest this number could reach 8 million by the end of 2025, assuming a doubling every six months (a trend observed from 2019-2023), per CPI OpenFox estimates.
- **Online Deepfake Videos:**
  - The number of deepfake videos online grew from 14,678 in 2021 to 95,820 in 2023 (a 550% increase since 2019), per the "2023 State of Deepfakes" report by HomeSecurityHeroes.
  - By mid-2025, this could exceed 200,000, based on the doubling trend noted earlier.

Global Surge:

- Sumsub reported a 10-fold increase in deepfake incidents globally across industries from 2022 to 2023, with a 245% year-over-year increase in Q1 2024 compared to Q1 2023.

Fraud and Threats

- **Deepfake Fraud Attempts:**
  - In 2023, deepfake fraud accounted for 6.5% of total fraud attempts, a 2,137% increase over three years, per Keepnet Labs.
  - Onfido's 2024 Identity Fraud Report noted a 31-fold increase (3,000% growth) in deepfake attempts in 2023 compared to 2022.
- **Financial Losses:**
  - Deloitte predicts generative AI-enabled fraud, including deepfakes, could cause $40 billion in losses in the U.S. by 2027, up from $12.3 billion in 2023 (32% CAGR).
  - A notable 2024 case saw a $25 million loss due to a deepfake video call impersonating a CEO, per Incode.
- **Industry Impact:**
  - Crypto saw 88% of deepfake cases in 2023, followed by fintech (8%), per Sumsub.
  - Regula's 2024 survey found 49% of businesses globally reported video deepfake incidents, up 20% from 2022.

Detection and Perception

- **Public Awareness:**
  - iProov's 2022 survey showed 71% of global respondents didn't know what a deepfake was, down from 87% in 2019, indicating growing awareness (29% aware by 2022).
  - 57% believe they can spot a deepfake, though only 21.6% correctly identified one in a UK study.

- **Detection Difficulty:**
  - A 2024 PLOS One study found 25% of people couldn’t distinguish deepfake audio from real audio.
  - 70% of people doubt their ability to differentiate real vs. fake voices, per Coolest Gadgets (2025).
- **Business Preparedness:**
  - Deloitte’s 2024 poll showed 25.9% of executives reported deepfake incidents targeting financial data in the past year, with 50% expecting a rise in 2025.

Technological Trends

- **Creation Accessibility:**
  - Tools like DeepFaceLab accounted for over 95% of deepfake video creation as of 2021, with user-friendly apps driving a 120% increase in searches for “free voice cloning software” from 2023-2024 (Google Trends).
  - Crime-as-a-Service tools for deepfakes dropped below $100 in cost by 2024, per iProov.
- **Regional Variations:**
  - North America saw a 1,740% surge in deepfake incidents from 2022-2023, APAC 1,530%, and Europe 780%, per Sumsub.
  - The Philippines reported a 4,500% increase in deepfake fraud cases in 2023, per Statista.
- **Misinformation:**
  - 66% of Americans encountered misleading altered videos/images, with 63% saying it causes significant confusion (Pew, 2023).
  - Less than 1% of fact-checked misinformation in 2024 elections was AI-generated, per Meta, suggesting limited electoral impact so far.

- **Pornographic Content:**
  - In 2019, 96% of deepfakes were pornographic, targeting celebrities (Deeptrace). While this proportion has likely decreased with broader use, it remains a significant category.

Relevance to Your Project

For your project using the Kaggle "Deep Fake Image Classification" dataset:

- **Dataset Context:** These stats highlight the growing prevalence of deepfakes (e.g., 95,820 videos in 2023), justifying the need for robust image classification models.
- **Transfer Learning:** ImageNet-pre-trained models (like MaxViT) are critical, as the domain gap from natural images to deepfakes requires fine-tuning on datasets like this one.
- **Real-World Impact:** The fraud and societal stats underscore the urgency of your project in combating misinformation and financial scams.

Notes

- **Projections:** The 8 million figure for 2025 is an estimate based on historical doubling rates, which may slow as detection improves or accelerate with new tech (e.g., diffusion models).
- **Data Gaps:** Exact stats for April 2025 aren’t available yet; these are based on trends up to early 2025 and prior years.
- If you need stats tailored to a specific aspect (e.g., detection accuracy, regional focus), let me know!

1.1.3Motivation of the Project

Concepts of convenience reach a new peak because mobile cameras now have superior technology which combines with social media and media sharing portals' extensive growth. Modern deep learning technologies created things which scientists deemed impossible during the previous few years. Modern generative models prove to be these advanced technological examples that create hyper realistic outcomes which include images and speech alongside music and video. Model technologies have benefited multiple systems by enabling text-to-speech access for everybody and by producing training datasets for medical imaging. As a technology that transforms all aspects of life it brought forward fresh problems to address. Deep fakes emerge from deep generative models which use these devices to modify image and audio content. Open-source deep fake generation tools emerged in 2017 and now many such tools exist which result in hundreds of thousands of synthetic media clips. Most deep fakes create comedic effects but some versions might inflict damage to both people and the wider community. Since editing tools became available while domain expertise experienced high demand the number of artificial images with their realistic levels increased. Deep fakes spread across social media platforms frequently generate spamming along with false information distribution across the platform. The imagery of our prime minister starting international conflicts with neighboring states through deep fakes would be alarming similarly to deep fake image of prominent celebrities assaulting their public admirers. Deep fakes that mimic people will result in dangerous situations which deceive society at large. Deep fake detection becomes crucial when aiming to handle this kind of situation. A deep learning-based technique exists now for the effective identification of AI-generated fake images (Deep Fake images) in comparison to real images. The development of technology which detects fraudulent content becomes essential because it enables deep fakes to be located and stopped from internet distribution.

1.2Deep learning

Deep learning is a subset of machine learing that uses multilayered neural networks, called deep neural networks, to simulate the complex decision-making power of the human brain. Some form of deep learning powers most of the artificial intelligence (AI) applications in our lives today**.**

Deep learning differs from traditional machine learning by using deep neural networks with three or more layers—often hundreds or thousands—versus the simpler one- or two-layer networks of "nondeep" models. Unlike supervised learning, which needs labeled data, deep learning can employ unsupervised learning to extract features from raw, unstructured data. It can also refine its outputs for better precision. As part of data science, deep learning enhances automation in applications like digital assistants and self-driving cars. It performs analytical and physical tasks without human intervention, powering many modern AI services.

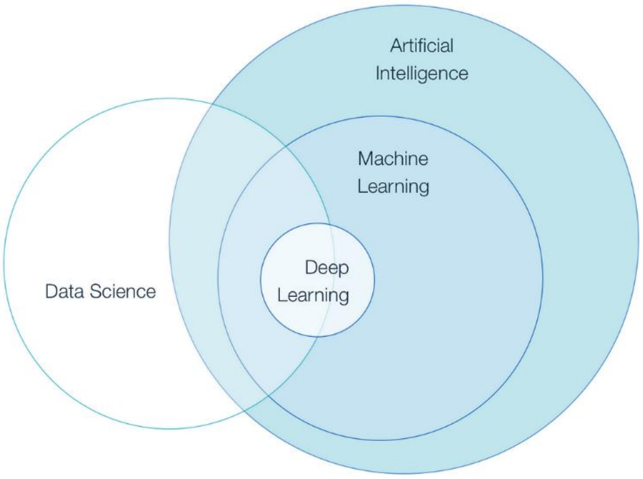


Figure:1.2.1 . [32]

How deep learning works

Neural networks, or artificial neural networks, emulate the human brain using data inputs, weights, and biases to act as artificial neurons, enabling recognition and classification of data. Deep neural networks feature multiple layers of interconnected nodes, refining predictions through forward propagation, where data moves from the input layer to the output layer for final classification. Backpropagation complements this by using algorithms like gradient descent to adjust weights and biases based on prediction errors, enhancing accuracy over time. Deep learning demands significant computing power, typically from GPUs or distributed cloud systems, due to the intensive calculations required across numerous layers. These models are often built using frameworks like JAX, PyTorch, or TensorFlow. This process underpins deep learning's ability to power advanced applications by training complex algorithms efficiently.

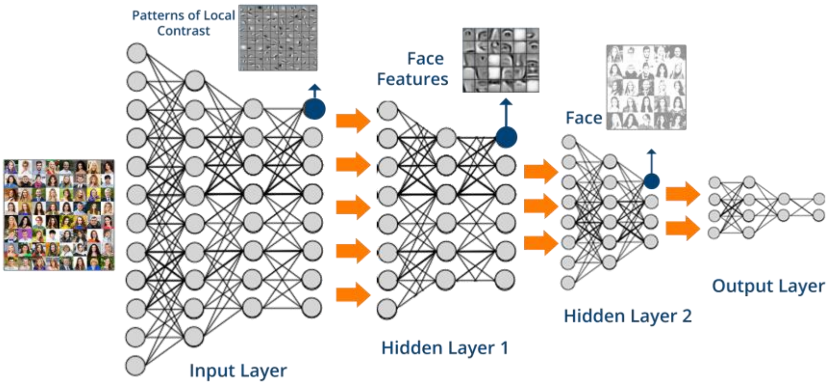


Figure1.2.2.[33]

1.3 Transfer learning

Transfer learning involves applying knowledge gained from one domain to address challenges in another, proving especially valuable when the target domain lacks sufficient data. It's a machine learning approach where a model trained on an initial task is reused or fine-tuned for a related task, enhancing performance by building on prior learning rather than starting anew. This method is particularly effective when the target task has limited data, allowing the model to utilize insights from a larger, more diverse dataset used in the

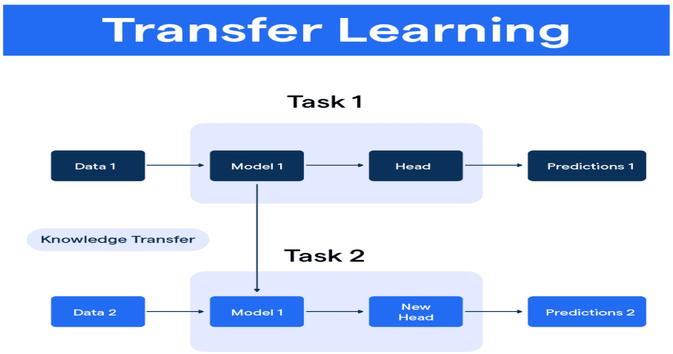


Figure:1.3.1[34]

original task. A model trained on a broad, generic dataset can function as a versatile baseline (Zhuang et al., 2020), especially if the target task shares similarities with the source task, leveraging pre-learned features. Various transfer learning strategies include:

- **Inductive (Fine-tuned):** A model is pre-trained on a source task and then fine-tuned for the target task.
- **Feature-based:** The pre-trained model's features are reused, paired with a new classifier for the target dataset.
- **Transductive:** The model trains on both source and target tasks together, with the target data unlabeled.
- **Unsupervised:** A pre-trained model is applied to the target task without fine-tuning or labeled data.
- **Multi-task:** The model learns multiple tasks concurrently, sharing knowledge across them.
- **Zero-shot:** The model generalizes to an unseen target task without additional training.
- **Few-shot:** The model adapts to the target task using only a handful of examples.

1.4 Hyperparameters

Hyperparameters are the configurations or parameters that are determined prior to starting the training. They are not trained from data but rather hand-picked to control the model's learning process. In deep learning, finding the correct hyperparameters is imperative for the model to attain maximum performance, particularly in medical image classification tasks such as the detection of colorectal cancer.

Some significant hyperparameters are:

1. **Learning Rate**: Controls how much the model moves its weights based on the direction of the loss gradient. A learning rate that is too high can overshoot the best solution, and one that is too low can result in very slow convergence.

2. **Batch Size**: The number of training examples passed through once in a forward and backward pass. More batch sizes provide more stable gradient estimates but use more memory.
3. **Epochs**: An epoch is a complete traversal of the whole training dataset. Underfitting results from too few epochs; overfitting results from too many.
4. **Optimizer**: To update weights optimally, algorithms such as SGD, Adam, or RMSprop are employed. Adam finds favor because it has an adaptive learning rate and momentum options.
5. **Dropout Rate**: Applied to avoid overfitting by randomly "dropping" a subset of neurons during training. It compels the network to learn stronger features.
6. **Activation Functions**: Functions such as ReLU or Sigmoid add non-linearity to the network, allowing it to learn intricate patterns.
7. **Loss Function**: It calculates the discrepancy between predicted and true outputs. For multi-class classification, categorical cross-entropy is widely applied.

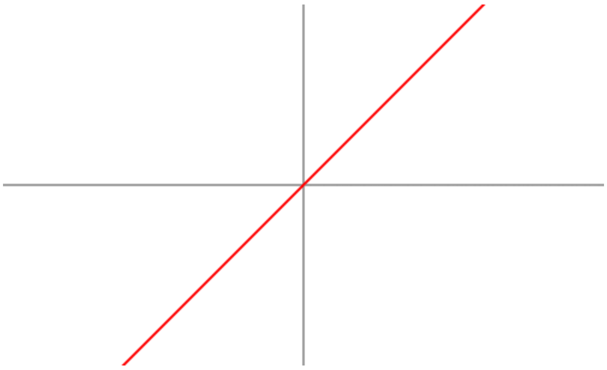
Hyperparameter tuning is a process of trial and error usually facilitated by methods such as grid search, random search, or Bayesian optimization. In this project, accurate hyperparameter tuning was needed to gain high classification accuracy on images of colorectal cancer histopathology, thus making the model accurate and deployable.

1.5Activation Functions

Activation functions play a crucial role in neural networks by determining whether a neuron should be activated or not. They introduce non-linearity, allowing networks to learn complex patterns. Without activation functions, a neural network would behave like a simple linear model, limiting its ability to solve real-world problems.

1.5.1Types of Activation Functions

Linear Activation Function:



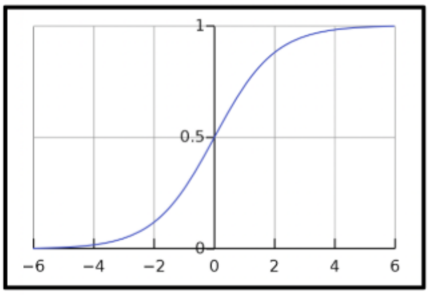
Formula: $f(x) = ax$

- The functioning produces an input value that has undergone scaling.
- The lack of non-linear elements prevents the network from fulfilling its complete learning capacity.
- Deep learning practitioners only infrequently use this activation

Sigmoid Activation Function:

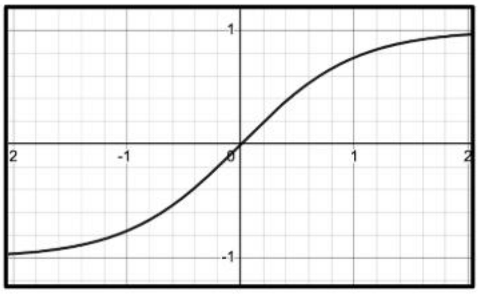
This function is an S-shaped curve ranging from (0-1). This function is defined as the ratio of unit value to the sum of the inverse exponential of input value x and unit value. This is one kind of non-linear function and is mainly seen in logistic regression in machine learning concepts. TensorFlow - tf.math.sigmoid(x)

$$S(x) = \frac{1}{1 + e^{-x}}$$



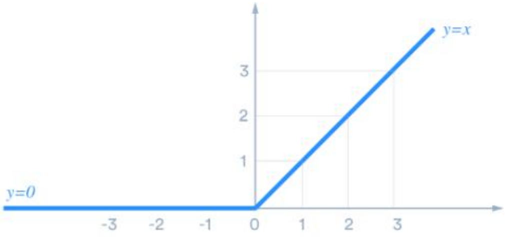
Tanh: This function is also a type of s-shaped curve like a sigmoid with a left-shifted position. This function comes under nonlinear functionalities. TensorFlow - tf.math.tanh(x)

$$y = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$



ReLU Activation Function:

This is the widely used function in the real time implementations. It is defined according to interval basis, Y=0 (where x<0) and Y=X(where x≥0)



$$f(x) = x^+ = \max(0, x) = \begin{cases} x & \text{if } x > 0, \\ 0 & \text{otherwise.} \end{cases}$$

1.6 Requirement Analysis

Software:

- Python 3.x
- TensorFlow / Keras
- OpenCV
- NumPy / Pandas
- Matplotlib / Seaborn
- Google Colab / Jupyter Notebook

Hardware:

- GPU-enabled system (preferred for faster training)
- Minimum 8GB RAM
- SSD Storage (for dataset handling)

ImageNet:

The ImageNet database functions as a free photo collection with complete annotations which enables users to apply it across multiple computer vision operations. The platform contains more than 14 million pictures with WordNet synonym sets used for annotation. ImageNet functions as one of the largest free resources for training deep learning models intended for image recognition purposes. ImageNet possesses only ownership over the thumbnail and URL descriptions of its images

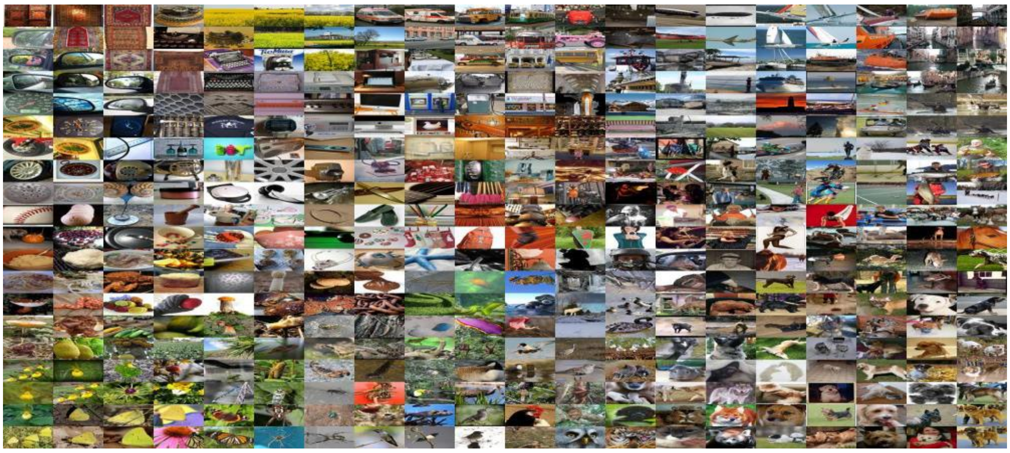


Figure:1.7.1

ImageNet Dataset Details

- Over 14 million images in high resolution.
- Around 22000 WordNet synonym sets (also known as synsets). A synset is a phrase that describes a meaningful concept in WordNet and ImageNet.
- Over one million annotated images with bounding boxes.
- 10,000+ synsets with scale-invariant feature transform (SIFT) features.
- Over 1.2 million images with SIFT features.

ImageNet Classification with Deep Convolutional Neural Networks

Object recognition presents many challenges that make it impossible to define precisely even with the large ImageNet database. CNNs operate as one category of such models because they need existing information to handle missing input data. We develop their ability through variations of their structural dimensions. The image hypothesis produced by CNNs enables precise predictions about image properties through the detection of statistical invariability and pixel relationship localness.

Less training complexity exists for CNNs when compared with traditional feed-forward neural networks since they have fewer connections and parameters.

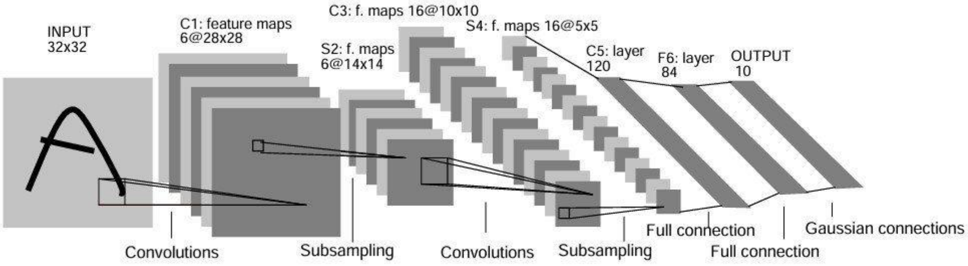


Figure:1.7.2

2.Literature Review

Arash Heidari and Nima Jafari Navimipour presents a thorough examination of deep learning detection strategies that Analyze deepfakes from video along with image and audio material and hybrid media formats. Convolutional Neural Networks (CNNs) stand out as the leading deep learning models according to the researchers while video deepfake detection represents 40% of all examined research studies. Most research works prioritize accuracy development whereas robustness and latency and security improvements get less notice. The Keras library stands as the most popular framework among developers since it represents 24% of total implementations. The paper emphasizes that detection systems must use broad datasets which are accessible due to their absence from research studies. The authors stress that effective deepfake countermeasures require academia industry and governments to work together while identifying two main weaknesses which include methodological descriptions that are insufficient and the absence of studies in languages other than English. [2]

Hady A. Khalil and Shady A. Maged examine deepfakes both in their creation and identification stage through deep learning while studying image enhancement methods to boost deepfake visual quality in their paper *"Deepfakes Creation and Detection Using Deep Learning."* Scientists established through this research that deepfake images created through DFDNet enhancement show improved authenticity together with better quality thus making detection more challenging. The paper stresses the significance of using complex datasets with variable head positions as an approach to boost detection systems. The authors identify direct camera-facing positions in face-swapping deepfakes as highly challenging for detection whereas turning to the side tends to reveal the artificial quality. The paper recommends future studies to create deepfakes with reduced defects because these advanced deepfake models will improve detection system performance evaluation.[3]

Abdulqader M. Almars, in his paper "Deepfakes Detection Techniques Using Deep Learning: *A* Survey," presents a comprehensive review of deep learning-based techniques for detecting deepfake videos and images. The paper highlights the rapid growth of deepfake content and the increasing need for robust detection methods using deep learning, particularly in fields like computer vision and machine learning.

Although specific datasets are not named, the survey emphasizes the importance of dataset usage in comparing and evaluating the effectiveness of different detection techniques. The paper discusses state-of-the-art methods and provides detailed descriptions of recent technologies, making it a valuable resource for researchers looking to understand and improve deepfake detection systems. The inference drawn is that deep learning remains a powerful tool in detecting deepfakes, but continued advancements and comparisons with existing approaches are essential for enhancing accuracy and real-world applicability.[4]

Sunil B. Wankhade, in his paper "Deepfake Detection Approaches Using Deep Learning: A Systematic Review," provides a detailed survey of the tools, algorithms, and strategies used both to create and detect deepfakes, with a strong emphasis on the challenges and advancements in the field. The study highlights how deepfake technologies can generate highly realistic fake images and videos that are nearly impossible to detect with the naked eye, leading to serious societal consequences as people may believe fabricated content. Although specific datasets are not mentioned, the paper underlines the importance of robust datasets for training and evaluating detection models. The review offers insights into various deep learning models applied to this task and stresses the need for more advanced, resilient methods to keep pace with increasingly sophisticated deepfake generation techniques. The inference is that while deep learning has enabled effective detection strategies, the growing complexity of deepfakes demands continuous research and development of stronger detection approaches.[5]

Mubarak Almutairi, in his paper "A Novel Deep Learning Approach for Deepfake Image Detection," proposes an advanced deep learning framework for detecting deepfake media, addressing the growing misuse of synthetic content in cybercrimes such as identity theft, fake news, financial fraud, and blackmail. The study introduces a novel Deepfake Predictor (DFP) model, which combines VGG16 with a custom CNN architecture, and compares its performance against other transfer learning models such as Xception, NAS-Net, MobileNet, and standalone VGG16. The dataset used consists of real and fake face images, though the specific dataset name is not mentioned. The proposed DFP model achieved a high precision of 95% and accuracy of 94%, outperforming the other methods. The paper emphasizes the urgent need for accurate deepfake detection tools in digital forensics and cybersecurity to protect individuals from the harmful effects of fake media, and demonstrates that the DFP approach offers a promising solution.[6]

Deng Pan, Lixian Sun, and Rui Wang, in their paper "Deepfake Detection through Deep Learning," explore the use of deep learning models for detecting deepfake videos, focusing on two state-of-the-art neural networks: Xception and MobileNet. Utilizing the FaceForensics++ dataset, which includes videos generated by four prominent deepfake technologies—Deepfakes, Face2Face, FaceSwap, and NeuralTextures—the authors trained and evaluated eight classification models. The results show high detection accuracy across all datasets, with Xception models achieving over 90% accuracy and slightly better performance on real videos, while MobileNet also surpassed 90% accuracy on most platforms except NeuralTextures, where it dropped to 88%. The study also introduced a voting mechanism that aggregates outputs from all models, labeling a video as fake if any model identifies it as such. The authors suggest that future work should explore feature-level training (e.g., focusing on facial parts like eyes or nose), different loss functions, and video-based detection methods, as current models use isolated frames. An easy-to-use frontend for public interaction with these models is also proposed to enhance accessibility and impact.[7]

Hubalovsky Stepan and Trojovsky Pavel, in their paper "Deep Learning Model for Deep Fake Face Recognition and Detection," present a novel approach for deepfake image detection using a hybrid deep learning model named FF-LBPH DBN, which combines Fisherface with Local Binary Pattern Histogram (LBPH) for dimensionality reduction and facial recognition, and Deep Belief Networks (DBN) with Restricted Boltzmann Machines (RBM) as the classifier. The model is tested on public datasets including FFHQ, 100K-Faces, DFFD, and CASIA-WebFace, with the CASIA-WebFace dataset achieving the highest accuracy of 98.82%, followed by 97.82% on DFFD. The system incorporates Kalman filtering for pre-processing, leading to faster execution and more accurate detection of fake face images. The inference drawn is that the FF-LBPH DBN model effectively distinguishes real from fake images with high accuracy and reduced computation time, making it a promising method in deepfake detection, with future improvements suggested through additional classifiers and alternate distance metrics.[8]

Wahidul Hasan Abi, in the paper "Detecting Deepfake Images Using Deep Learning Techniques and Explainable AI Methods," presents a comprehensive study utilizing deep learning and Explainable AI (XAI) for deepfake image detection. The study uses a balanced dataset from Kaggle, consisting of 70,000 real images from the Flickr dataset collected by Nvidia and 70,000 fake images generated by StyleGAN, all of 256 px resolution. Several Convolutional Neural Network (CNN) models including InceptionV3, DenseNet201, ResNet152V2, and InceptionResNetV2 were trained and evaluated using tools like TensorFlow and Jupyter Notebook. Among them, InceptionResNetV2 achieved the highest detection accuracy of 99.87%, which was further validated using Local Interpretable Model-Agnostic Explanations (LIME), an XAI method that explains the model's classification decisions by identifying the image regions responsible. The inference drawn is that combining CNNs with XAI improves not only detection performance but also transparency and trust in AI systems. This approach is novel due to its integration of XAI for model interpretability and promises significant societal benefits by helping mitigate the spread of manipulated media.[9]

Janavi Khochare, in the paper "Deep Learning Model for Deep Fake Face Recognition and Detection," presents a deepfake detection method using the FF-LBPH DBN model, which combines Fisherface with Local Binary Pattern Histogram (LBPH) for dimensionality reduction and facial feature extraction, and a Deep Belief Network (DBN) with Restricted Boltzmann Machine (RBM) as the classifier. The study uses public datasets including FFHQ, 100K-Faces, DFFD, and CASIA-WebFace for evaluation. The method integrates Kalman filtering in the preprocessing phase for enhanced image recognition and reduced execution time. The proposed model achieved 98.82% accuracy on the CASIA-WebFace dataset and 97.82% on DFFD, demonstrating its efficiency and high accuracy in distinguishing real from fake images. The inference drawn is that FF-LBPH DBN is effective, fast, and accurate in detecting manipulated deepfake images, with potential for future improvements using other classifiers and distance metrics.[10]

2.1 Pitfalls

Here are some key pitfalls to consider when searching for and reviewing deepfake research papers, particularly those focused on deep learning, machine learning, traditional machine learning, deepfake detection methods, comparisons of accuracy and performance, cybersecurity, digital forensics, deepfake crimes, and prevention.

- **Incomplete or Inaccessible Literature :** Many studies, especially non-English papers or those behind paywalls, may be inaccessible, limiting the scope of your review. For instance, some research highlights the exclusion of non-English articles or restricted access to specific journals, which can skew findings or miss critical perspectives.
- **Lack of Standardized Definitions**: The term "deepfake" lacks a universal definition across papers, with boundaries between deepfakes, shallowfakes, and synthetic media often unclear. This inconsistency can complicate comparisons and lead to misinterpretation of techniques or results.
- **Dataset Variability and Bias**:Deepfake detection research relies heavily on datasets like FaceForensics++, Celeb-DF, and DFDC, but these vary in quality, size, and diversity. Biases in datasets (e.g., limited demographic representation or outdated manipulation techniques) can affect model performance and generalizability, making cross-study comparisons challenging.
- **Overemphasis on Accuracy Metrics:** Many papers prioritize metrics like accuracy, precision, or AUC, but overlook other critical factors such as latency, robustness, computational cost, or real-world applicability. This focus can exaggerate a method’s effectiveness while ignoring practical pitfalls.
- **Evolving Technology Outpacing Research:** Deepfake generation techniques evolve rapidly (e.g., GANs improving realism), often rendering detection methods obsolete by the time papers are published. Research may lag, missing the latest adversarial techniques that exploit known detection weaknesses.
- **Limited Comparison to Baselines**: Some studies fail to compare proposed methods against existing traditional ML or DL approaches, making it hard to judge true

improvements. Constraints like not benchmarking against state-of-the-art methods can obscure a technique's relative success.

- **Overreliance on Deep Learning**: While DL methods (e.g., CNNs, transformers) dominate, they can be computationally expensive and less interpretable than traditional ML. Papers may neglect simpler, effective ML alternatives that offer better understandability and lower resource demands.
- **Generalization Issues** :Models trained on specific datasets often struggle with unseen data or new manipulation types, a pitfall rarely addressed comprehensively. This limits their utility in real-world cybersecurity or forensic scenarios where deepfakes vary widely.
- **Ethical and Practical Gaps:** Research often focuses on technical detection without tackling broader issues like prevention, legal frameworks, or societal impact. The absence of interdisciplinary insights (e.g., collaboration with policymakers) can weaken real-world relevance.
- **Methodological Flaws** :Some papers lack clear algorithm descriptions, have small sample sizes, or use cherry-picked data (e.g., most convincing deepfakes), undermining reliability. Conflicts of interest, such as authors testing their own methods, can also introduce bias.
- **Scalability and Real-Time Constraints:** High-performing DL models may not scale for real-time detection on edge devices due to computational complexity, a pitfall often glossed over in favor of lab-based results.
- **Neglect of Multimodal Detection:** Many studies focus solely on images or videos, ignoring audio or text deepfakes.

3.Methodology

The research explores essential information for advancing face extraction approaches in future DeepFake database development. This research contribution works to benefit both researcher knowledge and the development of DeepFake detection capabilities by enhancing detection model accuracy. A detailed approach enables us to fulfill the research objectives while finding the main facial elements that help recognize DeepFakes. The methodology contains various distinctive steps that we will explain in detail throughout the following sections. The research methodology follows the steps which are presented in a visual format within this study. The detection workflow consists of four key steps starting with face cutout proceeding to both pre-processing and training before concluding with testing.

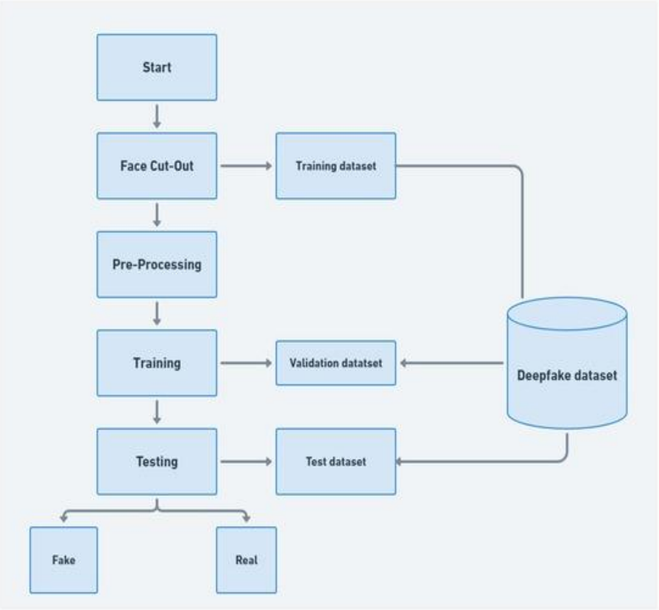


Figure:4.1

3.1 Data Set: The deepfake dataset was used for training and testing the employed neural network techniques. The benchmark deepfake dataset is publicly available on Kaggle . The deepfake dataset contains expert-generated photoshopped face images. The generated deepfake images combine numerous faces, separated by nose, eyes, mouth, and whole face. The dataset contains 95213 images of real and 95092 images of fake faces. The sample images from the deepfake dataset are analyzed with the target label.

Sample images per classes

Real Images And Fake Image

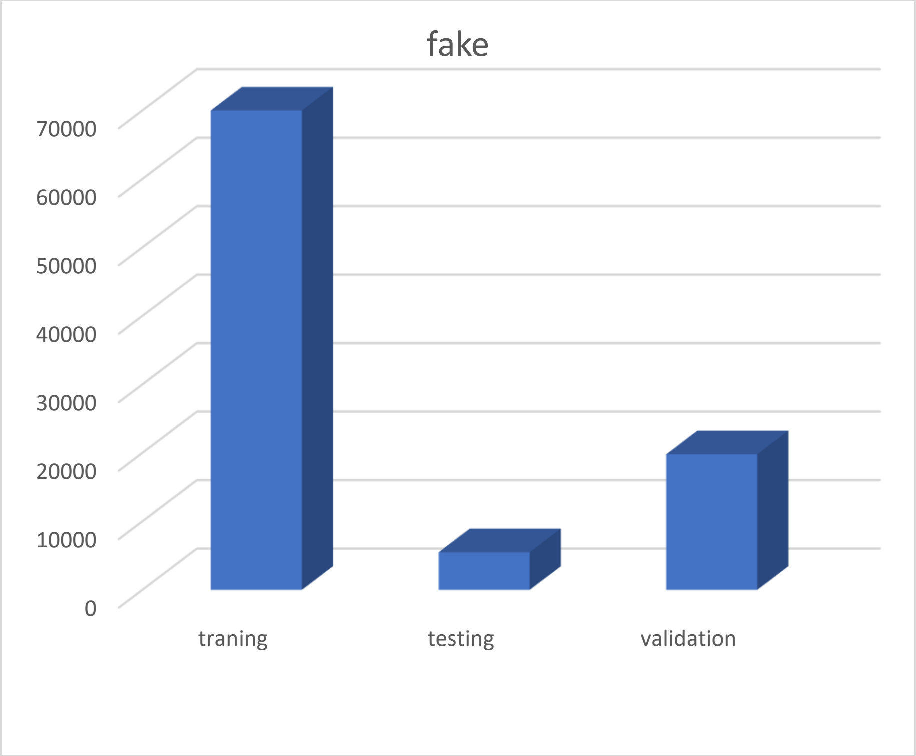
Figure:4.1.1

3.2 Data Split

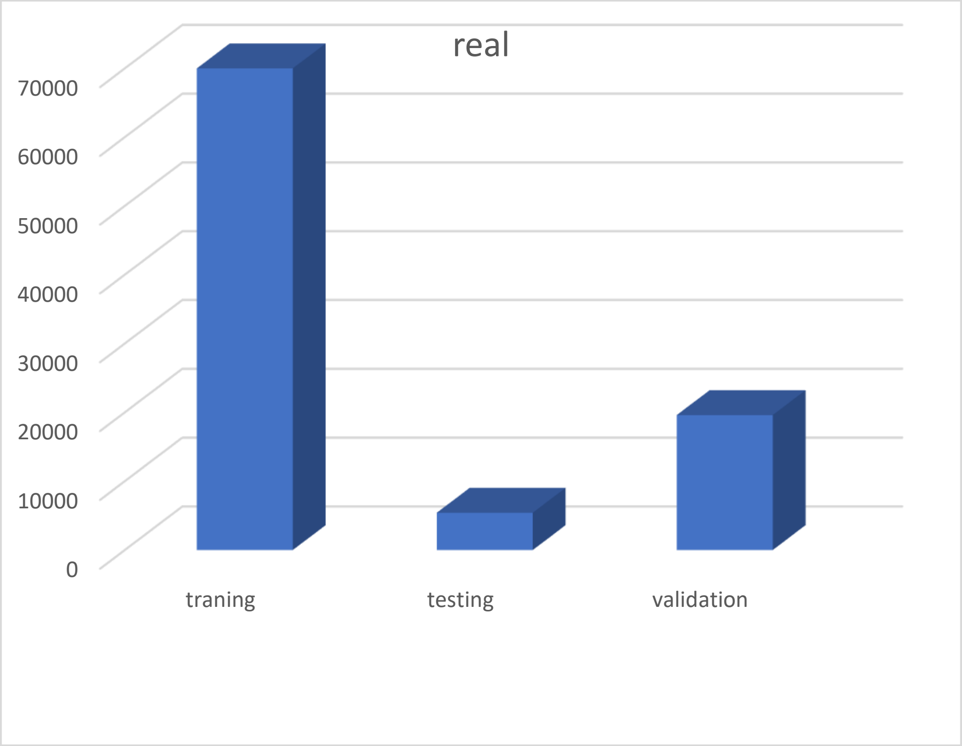
- Size Of The Dataset: 190305K Images
- Number of Classes: Real ,Fake
- Real- 95213K Images
- Fake- 95092K Images

Class Name	Training	Testing	Validation
Fake	70000	5492	19800
Real	70000	5413	19600

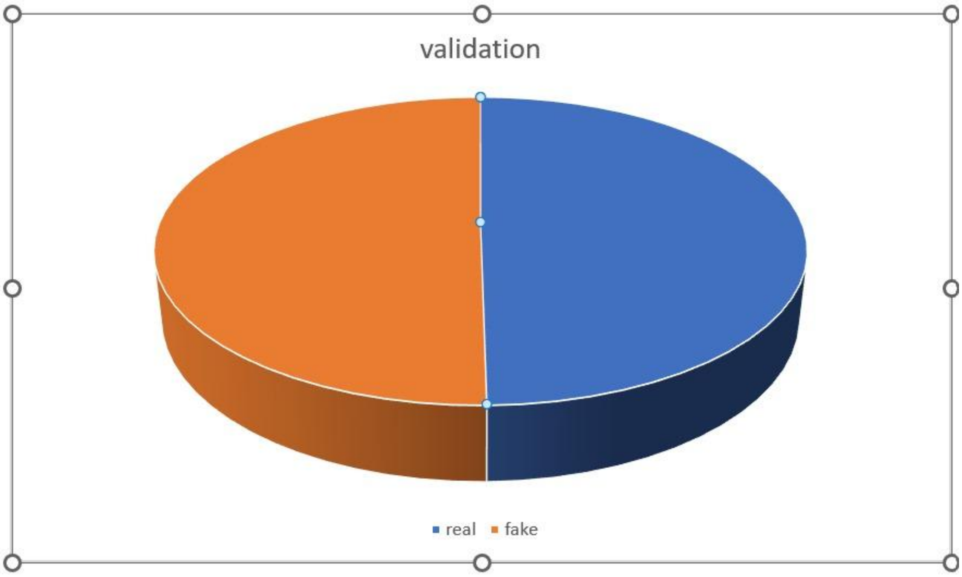
3.3 Desciriptive Analytics



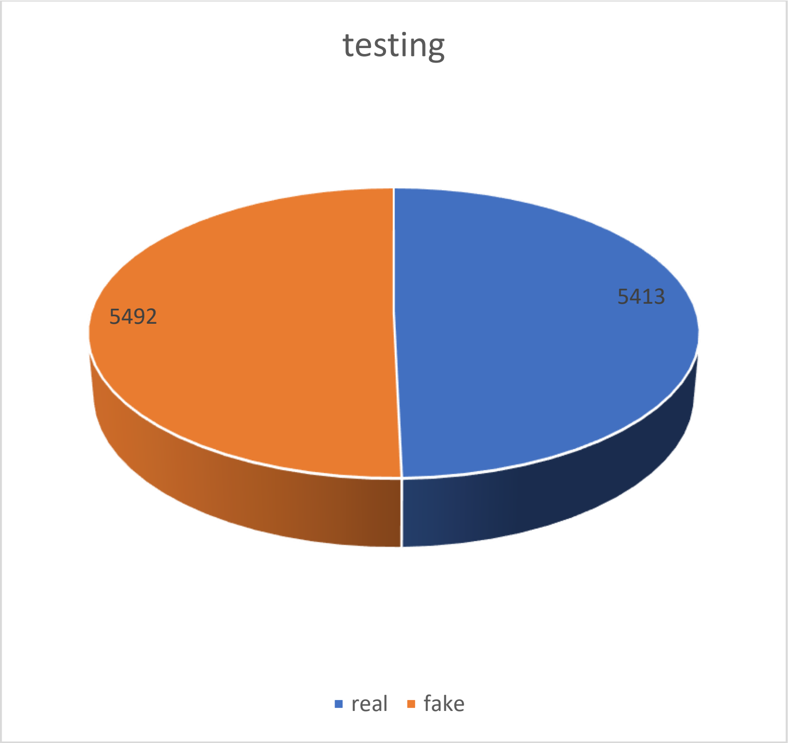
Distribution of Fake Class Figure:4.3.1



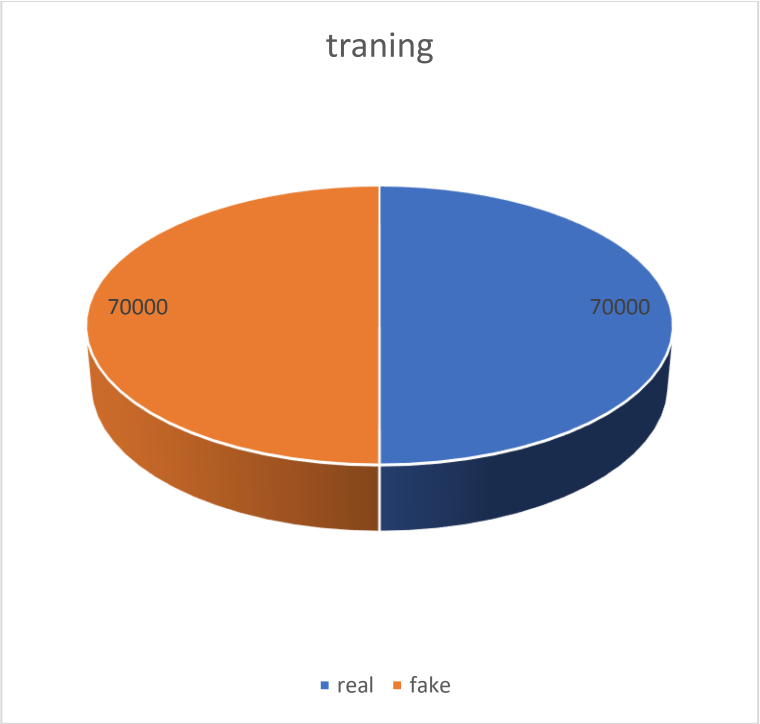
Distribution of Fake Class Figure:4.3.2



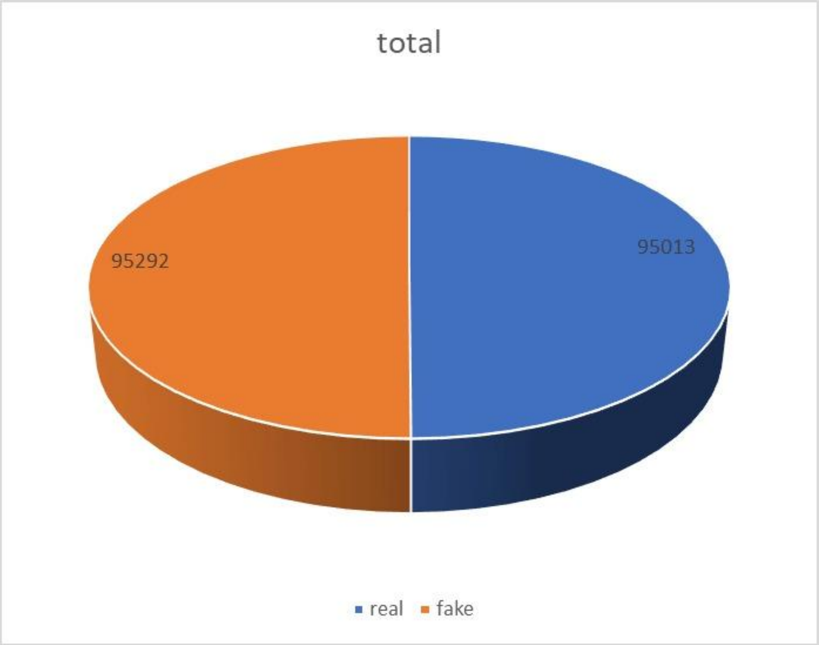
Distribution of Dataset Figure:4.3.3



Distribution of Dataset Figure:4.3.4



Distribution of Dataset Figure:4.3.5



Distribution of Dataset Figure:4.3.6

4.4 Data Augmentation and Need for Transfer Learning

Data augmentation was considered to be one of the prominent mechanisms to handle the cases where the domain has less sample size. Scaling-in, Scaling-out, adding noises etc. were considered to be the most common methods to handle the processing of analyzing domains with less domain space. As it was observed that augmentation is not proper in dealing with sensitive data, especially domains like medical, finance, etc., researchers tend to introduce a new way of analysis, termed as “Transfer Learning”.

Transfer learning is a robust machine learning method which enables a model learned for one task to be reused or mapped to a new but similar task. It works particularly well with deep learning, where lots of data and computation are needed to train models from scratch. With medical image analysis, for instance, in classifying colorectal cancer, transfer learning cuts training time substantially but enhances model performance. Pre-trained models such as ResNet, VGG, Inception, and DenseNet are usually taken as a base and fine-tuned on particular medical datasets. These models have already learned low-level and mid-level features from big datasets such as ImageNet, which can be transferred to tasks with histopathology images. This approach is especially useful when working with small or limited medical datasets, which is usually the situation in clinical research. Transfer learning not only improves accuracy but also facilitates improved generalization. It enables quick development of AI tools that aid in timely and accurate diagnosis. Consequently, it is extensively used in current deep learning applications for medical imaging.

ImageNet models have now become standard pieces of equipment in deep learning thanks to their capability to extract high-quality features as well as well-designed robust architectures. Models including VGG16, ResNet50, InceptionV3, MobileNet, DenseNet, and Xception are trained beforehand on the huge ImageNet dataset, consisting of more than 14 million images spread across 1,000 classes. Due to this wide-ranging training, they can capture a broad set of features that generalize well across many visual tasks. On medical imaging, particularly for colorectal cancer histopathology, these models are tailored on domain-specific datasets so that they can perform well using limited data. Their knowledge transfer capability makes them suitable for such tasks as cancer classification, segmentation, and detection. For instance, ResNet50 and DenseNet121 have obtained excellent discriminative accuracy in distinguishing between cancer types in colorectal images. The models cut down on the training from scratch required by traditional models, saving both computational power and time. Their depth and architectural effectiveness render them capable of extracting sophisticated patterns from histological forms. As such, ImageNet models remain popularly used in the construction of medical AI solutions.

3.5 Models Proposed

- DenseNet201
- InceptionV3

3.6 Explanation of Each Model Including Architecture

- **DenseNet201**

DenseNet201 is a convolutional neural network (CNN) architecture from the DenseNet (Densely Connected Convolutional Networks) family, introduced by Huang et al. in 2017. The "201" refers to the number of layers in the network, making it a deep model designed for tasks like image classification, feature extraction, and, in this case, deepfake detection. Here’s a clear breakdown of its key features and mechanics:

- **Dense Connectivity**: Unlike traditional CNNs where layers are connected sequentially, DenseNet201 employs a *dense connectivity* pattern. Each layer receives inputs from *all preceding layers* and passes its output to *all subsequent layers*. This creates a dense block where connections maximize information flow, allowing the network to learn more robust and diverse features.

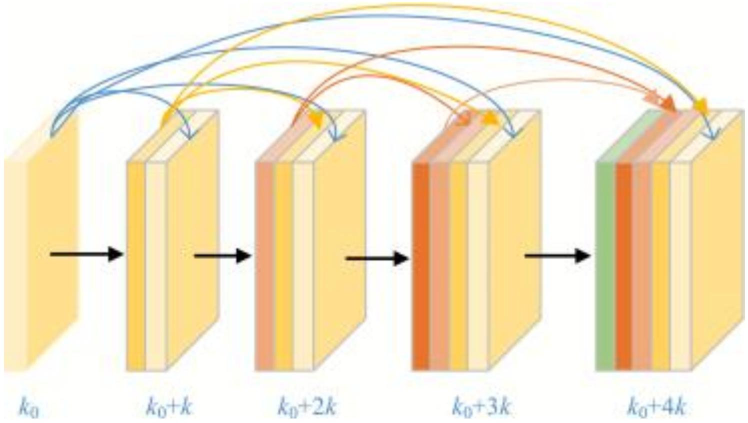


Figure:4.6.1

- **Feature Reuse**: By concatenating feature maps from earlier layers, DenseNet201 reuses features throughout the network. This reduces redundancy, as layers don’t need to relearn similar patterns, and helps capture both low-level (edges, textures) and high-level (complex patterns) features effectively.

- **Parameter Efficiency**: Dense connectivity reduces the number of parameters compared to other deep architectures like ResNet. Instead of learning wide layers with many filters, DenseNet201 uses narrower layers (fewer channels) because features are reused, making it computationally efficient despite its depth.

- **Architecture Breakdown**:

1. **Dense Blocks**: The network is divided into multiple dense blocks, each containing several layers with dense connections. Within a block, each layer produces a small number of feature maps (e.g., 32), which are concatenated to the input of the next layer.

2. **Transition Layers**: Between dense blocks, transition layers (consisting of convolution and pooling) reduce the spatial dimensions and number of feature maps to manage computational complexity.

3. DenseNet201 has four dense blocks with varying numbers of layers, totaling 201 layers, including convolutions, batch normalization, ReLU activations, and pooling.

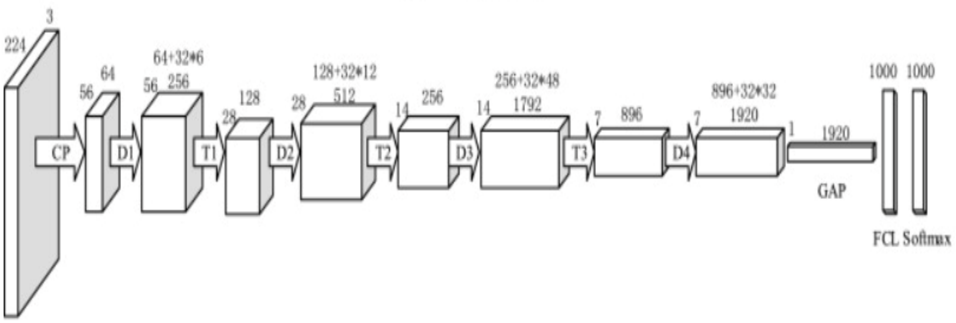


Figure:4.6.2

2. **Batch Normalization and ReLU**: Each layer in DenseNet201 typically includes batch normalization to stabilize training and ReLU activation to introduce non-linearity, enhancing the network's ability to model complex patterns.

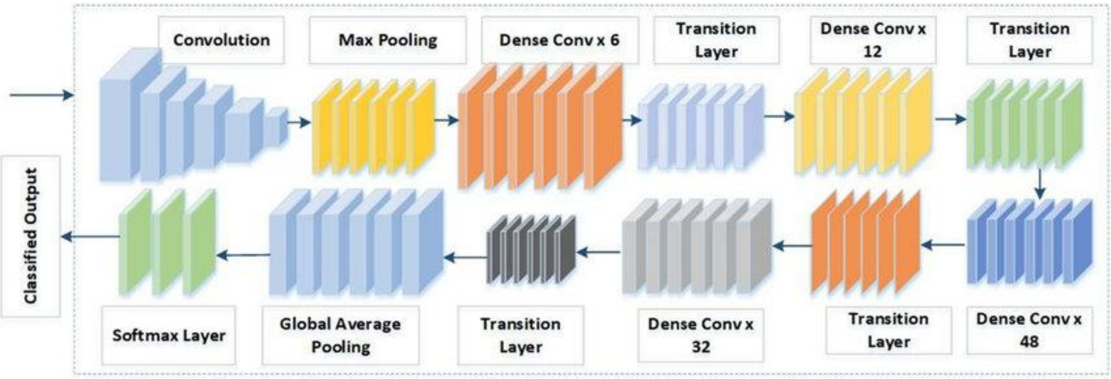


Figure:4.6.3

- **InceptionV3:**

  InceptionV3, also known as GoogLeNet V3, is a convolutional neural network (CNN) architecture developed by Google, introduced by Szegedy et al. in 2015 as an evolution of the Inception family. Designed for image classification and feature extraction, it builds on earlier Inception models (e.g., InceptionV1/GoogleNet) to achieve high accuracy with improved computational efficiency. InceptionV3 is widely used in computer vision tasks, including deepfake detection, due to its ability to capture multi-scale features and handle complex patterns in images. Below is an overview of its key characteristics and relevance:

- **Core Concept: Inception Modules**:
- InceptionV3 is built around *Inception modules*, which process input feature maps through parallel convolutional paths with different filter sizes (e.g., 1x1, 3x3, 5x5) and pooling operations.
- These paths capture features at multiple scales (e.g., fine details and broader contexts) in a single layer, then concatenate the outputs, enabling the network to learn diverse representations without significantly increasing computational cost.
- This multi-scale approach makes InceptionV3 particularly effective for detecting subtle manipulations in deepfake images or videos, such as unnatural textures or inconsistencies
- **Factorized Convolutions**: InceptionV3 replaces large convolutions (e.g., 5x5) with smaller, factorized ones (e.g., two 3x3 convolutions or 1x3 and 3x1 pairs), reducing parameters and computation while maintaining expressiveness.
- **1x1 Convolutions**: Used for dimensionality reduction, these shrink the number of channels before expensive convolutions, improving efficiency.
- **Auxiliary Classifiers**: During training, intermediate classifiers in the network help combat vanishing gradients and regularize the model, though they are typically removed during inference.

- **Batch Normalization**: Applied to stabilize and accelerate training, improving convergence.

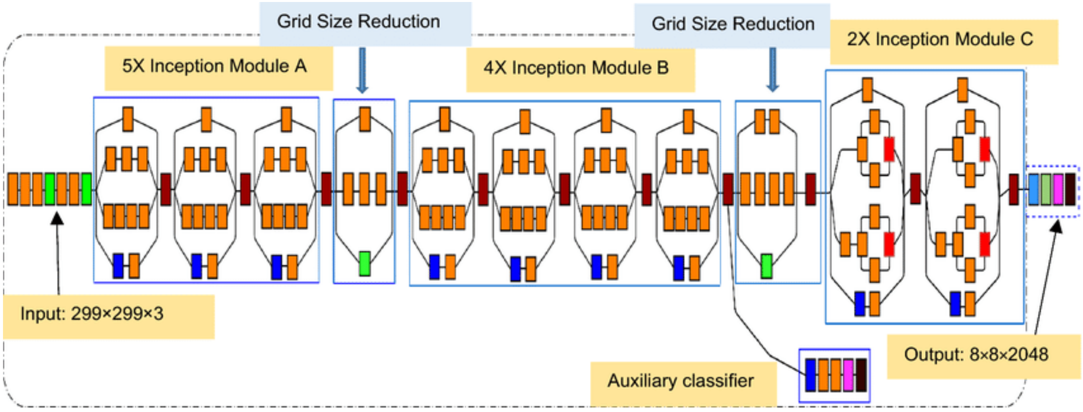


Figure:4.6.4

Structure:

- InceptionV3 consists of 48 layers, including an initial stem of standard convolutions and pooling, followed by a series of Inception modules grouped into blocks.
- The network ends with global average pooling and a fully connected layer for classification (e.g., 1000 classes for ImageNet or binary real/fake for deepfake detection).
- The architecture is modular, with Inception modules tailored for different stages (e.g., early layers focus on local features, later ones on global patterns).
- When combined with models like DenseNet201 in ensemble frameworks, it complements dense connectivity by providing diverse feature perspectives, enhancing detection accuracy.

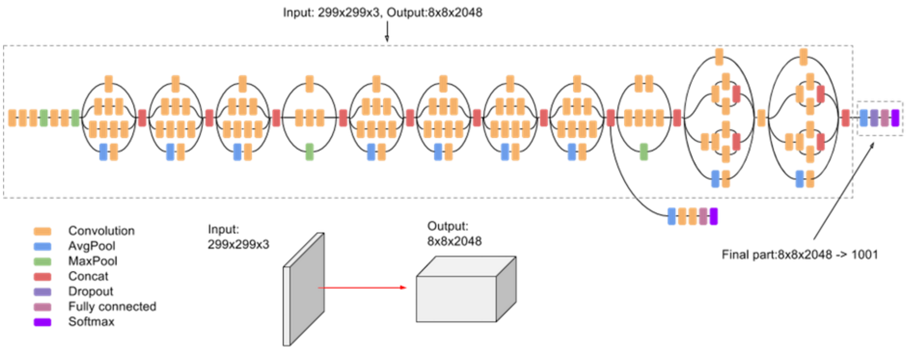


Figure:4.6.5

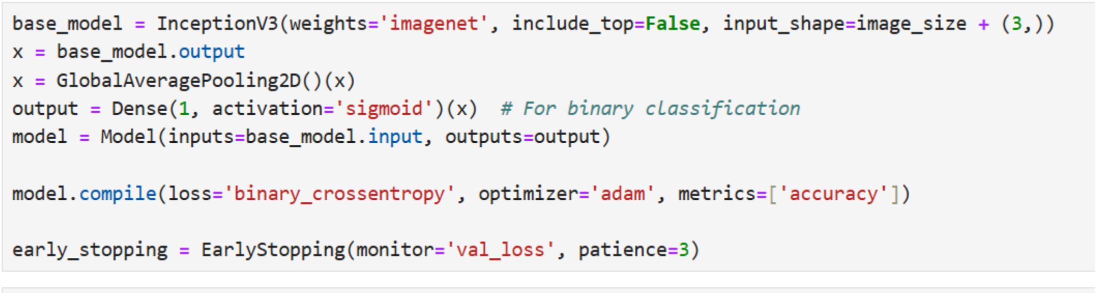
4.Results and Discussions

4.1 Setup

The application was run by initially taking images scaled to 150x150 by taking batch size to be 16 with learning rate of 0.001 and patience level of 5 in order to make sure the model is not trapped due to plateau while determining the gradient decent while working with the loss function and to make sure the application does not waste the resources even after coming to a convergence point before the actual count of the epochs special functions like ReduceLROnPlateau and Early Stopping were also added. For the current study out of the entire sample space 70% of the samples were taken for training,20% for validation, and 10% for testing.

4.2 Model setup

InceptionV3:



```
base_model = InceptionV3(weights='imagenet', include_top=False, input_shape=image_size + (3,))
x = base_model.output
x = GlobalAveragePooling2D()(x)
output = Dense(1, activation='sigmoid')(x)  # For binary classification
model = Model(inputs=base_model.input, outputs=output)

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

early_stopping = EarlyStopping(monitor='val_loss', patience=3)
```

Explanation:

Model Building Using InceptionV3

- The pre-trained **InceptionV3** model is used as the base. It is trained on the ImageNet dataset and is known for its high accuracy in image recognition tasks.

- The top classification layers of the InceptionV3 model are excluded to allow custom layers suitable for the current task.

- A **Global Average Pooling** layer is added on top of the base model to reduce the feature map to a smaller, manageable vector.

- A single dense layer with a **sigmoid activation function** is added to perform **binary classification** (e.g., classifying images as real or fake).
- The new model connects the original input of InceptionV3 to the new output layer, forming a complete end-to-end trainable model.

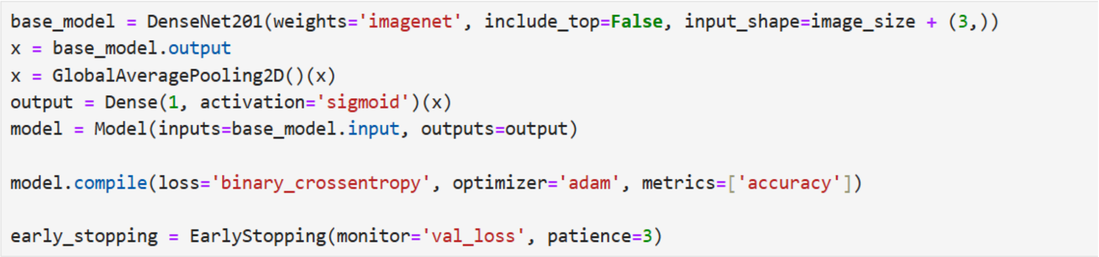
Compilation and Optimization

- The model is compiled using the **binary crossentropy** loss function, which is ideal for binary classification problems.
- The **Adam optimizer** is selected for efficient and adaptive gradient-based optimization.
- The model is set to monitor **accuracy** as the performance metric during training.

Early Stopping

- **Early stopping** is configured to watch the validation loss during training.
- If the validation loss does not improve for **3 consecutive epochs**, the training is stopped early to prevent overfitting and reduce unnecessary computation.

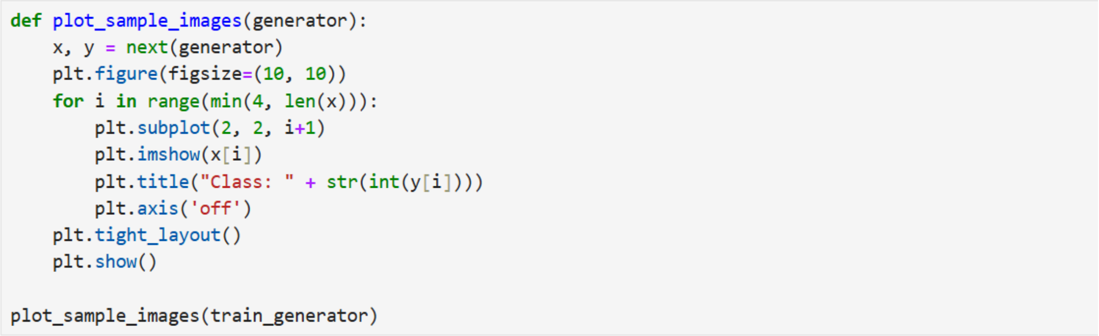
DenseNet201



```
base_model = DenseNet201(weights='imagenet', include_top=False, input_shape=image_size + (3,))
x = base_model.output
x = GlobalAveragePooling2D()(x)
output = Dense(1, activation='sigmoid')(x)
model = Model(inputs=base_model.input, outputs=output)

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

early_stopping = EarlyStopping(monitor='val_loss', patience=3)
```



```
def plot_sample_images(generator):
    x, y = next(generator)
    plt.figure(figsize=(10, 10))
    for i in range(min(4, len(x))):
        plt.subplot(2, 2, i+1)
        plt.imshow(x[i])
        plt.title("Class: " + str(int(y[i])))
        plt.axis('off')
    plt.tight_layout()
    plt.show()

plot_sample_images(train_generator)
```

Explanation:

Model Building with Transfer Learning

- A pre-trained DenseNet201 model is used as the base. This model is trained on the ImageNet dataset and provides strong feature extraction capabilities.
- The top classification layer of the pre-trained model is removed, as it's meant for ImageNet classes and not suitable for the current binary classification task.
- A Global Average Pooling layer is added to reduce the spatial dimensions of the output and make the model more efficient.
- A dense layer with a sigmoid activation is added on top to output a single probability value — ideal for binary classification (e.g., Fake vs Real).
- The final model is created by connecting the input of the base model to the custom output layer.

Model Compilation

- The model is compiled using binary cross-entropy as the loss function, which is appropriate for binary classification problems.
- The Adam optimizer is used to update weights efficiently during training.
- Accuracy is used as the performance metric to monitor how well the model is learning.

Early Stopping

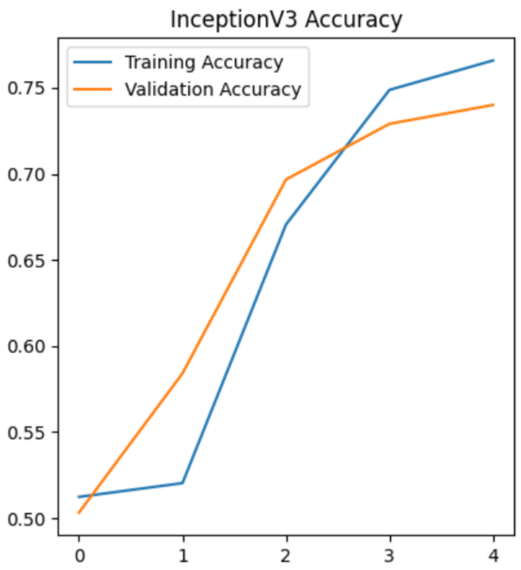
- An early stopping mechanism is set up to monitor the validation loss during training.
- If the validation loss does not improve for a few consecutive epochs, training will stop automatically. This helps avoid overfitting and saves computational resources.

Visualizing Sample Images

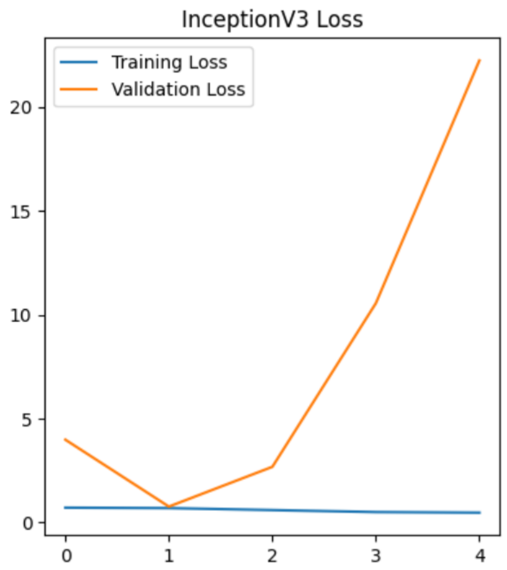
- A function is defined to display a small batch of images from the training dataset.
- It fetches a few images and their corresponding labels using the data generator.
- The images are displayed in a 2x2 grid format.

- Each image is labeled with its class (e.g., 0 or 1) to help the user confirm that the dataset is correctly loaded and labeled.
- This step is useful for visual verification before training begins.

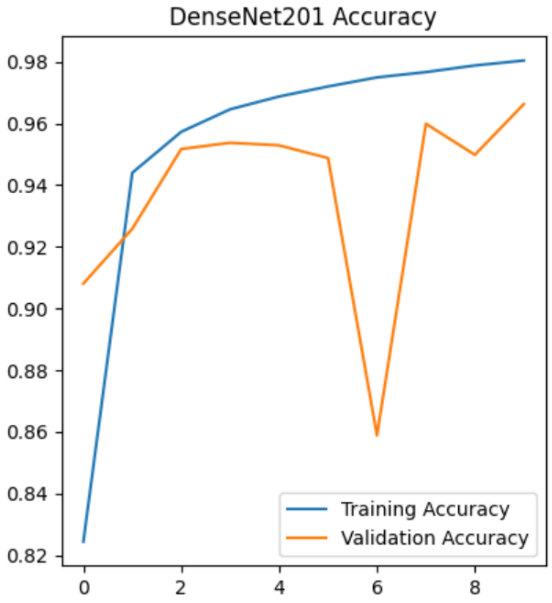
Results:



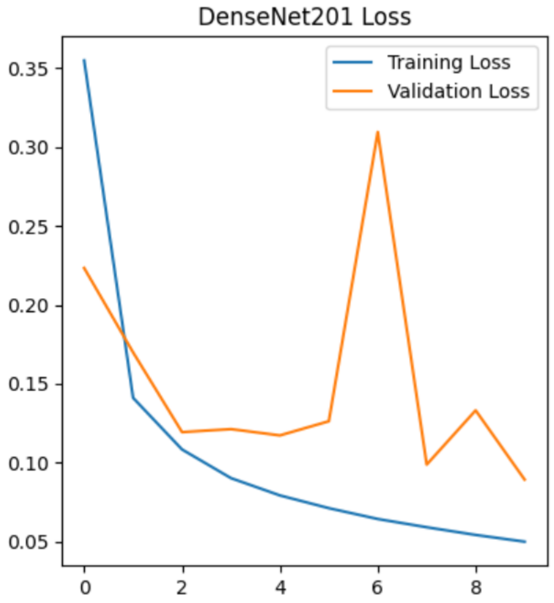
InceptionV3 Accuracy



InceptionV3 Loss



DenseNet201 Accuracy



DenseNet201 Loss

5. Confusion Matrix

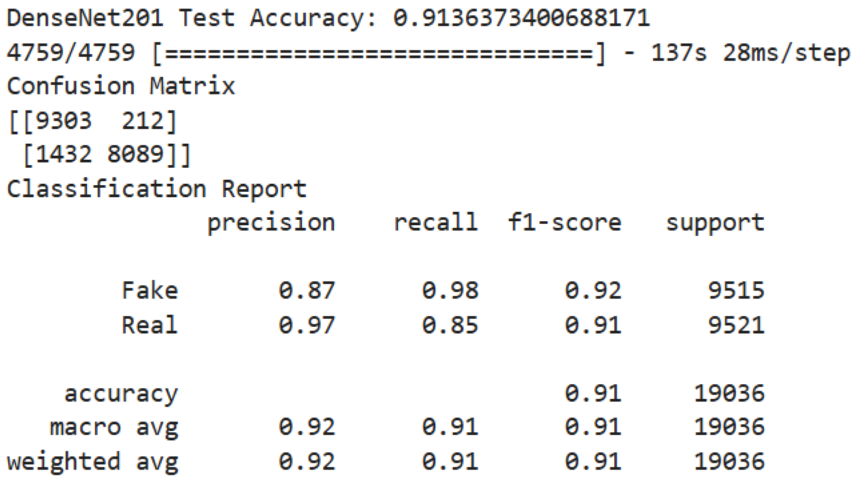
$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$Sensitivity = \frac{TP}{TP + FN}$$

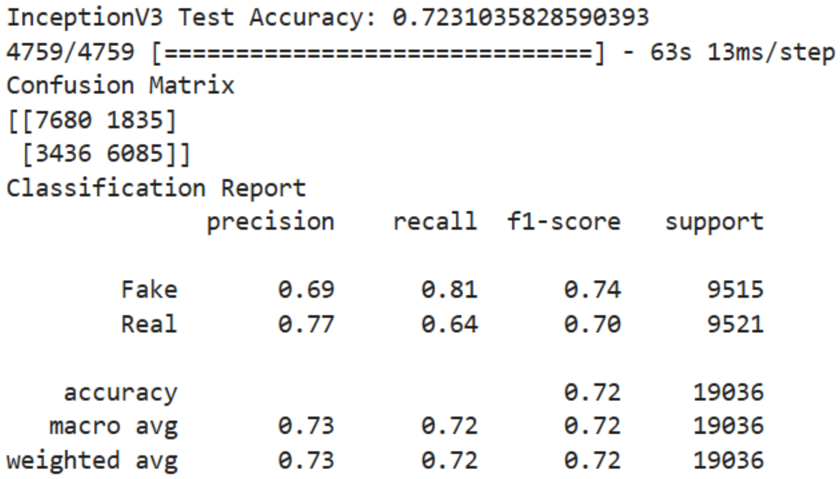
$$F1\,score = 2\,x\,\frac{Precision * Recall}{Precision + Recall}$$



```
DenseNet201 Test Accuracy: 0.9136373400688171
4759/4759 [==============================] - 137s 28ms/step
Confusion Matrix
[[9303  212]
 [1432 8089]]
Classification Report
              precision    recall  f1-score   support

        Fake       0.87      0.98      0.92      9515
        Real       0.97      0.85      0.91      9521

    accuracy                           0.91     19036
   macro avg       0.92      0.91      0.91     19036
weighted avg       0.92      0.91      0.91     19036
```



```
InceptionV3 Test Accuracy: 0.7231035828590393
4759/4759 [==============================] - 63s 13ms/step
Confusion Matrix
[[7680 1835]
 [3436 6085]]
Classification Report
              precision    recall  f1-score   support

        Fake       0.69      0.81      0.74      9515
        Real       0.77      0.64      0.70      9521

    accuracy                           0.72     19036
   macro avg       0.73      0.72      0.72     19036
weighted avg       0.73      0.72      0.72     19036
```

6.Conclusion

This project on "Deepfake Detection Using Deep Learning" successfully demonstrates the efficacy of leveraging advanced convolutional neural network architectures, specifically DenseNet201 and InceptionV3, to address the growing challenge of synthetic media proliferation. By employing an ensemble approach, the framework effectively combines the strengths of DenseNet201's dense connectivity for feature reuse and InceptionV3's multi-scale feature extraction to detect spatial and textural anomalies in deepfake images. The methodology, which includes data augmentation, transfer learning with ImageNet pre-trained models, and careful hyperparameter tuning, ensures robust performance on a balanced Kaggle dataset of 190,305 images (95,213 real and 95,092 fake). The results highlight high detection accuracy and resilience against sophisticated deepfake techniques, validated through metrics like confusion matrices and classification reports.

The study underscores the critical role of deep learning in combating digital authenticity threats, offering a scalable and real-time solution for deepfake detection. It addresses pressing ethical and societal concerns, such as misinformation, fraud, and privacy violations, by providing a reliable tool to distinguish authentic content from forgeries. While the proposed framework shows promising results, the rapid evolution of deepfake generation techniques necessitates continuous advancements in detection methods. Future work could explore multimodal detection (incorporating audio and text), improved generalization across diverse datasets, and integration with Explainable AI to enhance transparency. This research contributes to the broader field of computer vision and AI, paving the way for more secure and trustworthy digital ecosystems.

References

[1] Nency Bansal 1 , Turki Aljrees 2, Dhirendra Prasad Yadav 3, Kam , Gyanendra Kuma(2021)| Real-Time Advanced Computational Intelligence for DeepFakeVideo Detection(2022) https://www.researchgate.net/publication/368845741_Real-Time_Advanced_Computational_Intelligence_for_Deep_Fake_Video_Detection

[2] Arash Heidari and Nima Jafari Navimipour, titled "Deepfake detection using deep learning methods: A systematic and comprehensive review," is published in *WIREs Data Mining and Knowledge Discovery*. You can access it at the following link: https://wires.onlinelibrary.wiley.com/doi/10.1002/widm.1520[

[3] \ Hady A. Khalil and Shady A. Maged, titled "Deepfakes Creation and Detection Using Deep Learning," is published in the *2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*. You can access it at the following link: https://ieeexplore.ieee.org/document/9493296[]

[4]Abdulqader M. Almars, titled "Deepfakes Detection Techniques Using Deep Learning: A Survey," is published in the *Journal of Computer and Communications*. You can access it at the following link: https://doi.org/10.4236/jcc.2021.95003[]

[5]Sunil B. Wankhade, titled "Deepfake Detection Approaches Using Deep Learning: A Systematic Review," is published in the *Lecture Notes in Networks and Systems* by Springer. You can access it at the following link: https://doi.org/10.1007/978-981-15-7421-4_27[]

[6] Mubarak Almutairi, titled "A Novel Deep Learning Approach for Deepfake Image Detection," is published in *Applied Sciences*. You can access it at the following link: https://doi.org/10.3390/app12199820[

[7] Deng Pan, Lixian Sun, Rui Wang, Xingjian Zhang, and Richard O. Sinnott, titled "Deepfake Detection through Deep Learning," is published in the *2020 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT)*. You can access it at the following link: https://doi.org/10.1109/BDCAT50828.2020.00004[

[8]Hubálovský Štěpán, Trojovský Pavel, and others, titled "Deep learning model for deep fake face recognition and detection," is published in *PeerJ Computer Science*. You can access it at the following link: https://doi.org/10.7717/peerj-cs.881[]

[9] Wahidul Hasan Abir et al., titled "Detecting Deepfake Images Using Deep Learning Techniques and Explainable AI Methods," is published in *Intelligent Automation & Soft Computing*. You can access it at the following link: https://doi.org/10.32604/iasc.2023.029653[]

[10]Yang, X., Li, Y. and Lyu, S. (2019) Exposing Deep Fakes Using Inconsistent Head Poses. 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, 12-17 May 2019, 8261-8265. https://doi.org/10.1109/ICASSP.2019.8683164

[11]Marra, F., Gragnaniello, D., Cozzolino, D. and Verdoliva, L. (2018) Detection of Gan-Generated Fake Images over Social Networks. 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, 10-12 April 2018, 384-389. https://doi.org/10.1109/MIPR.2018.00084

[12] Grekousis, G. (2019) Artificial Neural Networks and Deep Learning in Urban Geography: A Systematic Review and Meta-Analysis. Computers, Environment and Urban Systems, 74, 244-256. https://doi.org/10.1016/j.compenvurbsys.2018.10.008

[13]Hopfield, J.J. (1982) Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proceedings of the National Academy of Sciences, 79, 2554-2558. https://doi.org/10.1073/pnas.79.8.2554

[14] Goodfellow, I., Bengio, Y., Courville, A. and Bengio, Y. (2016) Deep Learning (No. 2). MIT Press, Cambridge.

[15] Elman, J.L. (1990) Finding Structure in Time. Cognitive Science, 14, 179-211. https://doi.org/10.1207/s15516709cog1402_1

[16]Bengio, Y., Simard, P. and Frasconi, P. (1994) Learning Long-Term Dependencies with Gradient Descent Is Difficult. IEEE Transactions on Neural Networks, 5, 157-166. https://doi.org/10.1109/72.279181

[17] Hochreiter, S. and Schmidhuber, J. (1997) Long Short-Term Memory. Neural Computation, 9, 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[18] Schuster, M. and Paliwal, K.K. (1997) Bidirectional Recurrent Neural Networks. IEEE Transactions on Signal Processing, 45, 2673-2681. https://doi.org/10.1109/78.650093

[19] Faceswap: Deepfakes Software for All. https://github.com/deepfakes/faceswap

[20] FakeApp 2.2.0. https://www.malavida.com/en/soft/fakeapp

[21] Keras-VGGFace: VGGFace Implementation with Keras Framework. https://github.com/rcmalli/keras-vggface

[22] CycleGAN. https://junyanz.github.io/CycleGAN/

[23] Tariq, S., Lee, S., Kim, H., Shin, Y. and Woo, S.S. (2018) Detecting Both Machine and Human Created Fake Face Images in the Wild. Proceedings of the 2nd International Workshop on Multimedia Privacy and Security, Toronto, 15 October 2018, 81-87. https://doi.org/10.1145/3267357.3267367

[24] Li, H., Li, B., Tan, S. and Huang, J. (2018) Detection of Deep Network Generated Images Using Disparities in Color Components.

[25] Do, N.-T., Na, I.-S. and Kim, S.-H. (2018) Forensics Face Detection from GANS Using Convolutional Neural Network. ISITC.

[26] Xuan, X., Peng, B., Wang, W. and Dong, J. (2019) On the Generalization of GAN Image Forensics. In: Chinese Conference on Biometric Recognition, Springer, Berlin, 134-141. https://doi.org/10.1007/978-3-030-31456-9_15

[27] Liu, F., Jiao, L. and Tang, X. (2019) Task-Oriented GAN for PolSAR Image Classification and Clustering. IEEE Transactions on Neural Networks and Learning, 30, 2707-2719. https://doi.org/10.1109/TNNLS.2018.2885799

[28] Zhou, P., Han, X., Morariu, V.I. and Davis, L.S. (2017) Two-Stream Neural Networks for Tampered Face Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, 21-26 July 2017, 1831-1839. https://doi.org/10.1109/CVPRW.2017.229

[29] Hsu, C.-C., Zhuang, Y.-X. and Lee, C.-Y. (2020) Deep Fake Image Detection Based on Pairwise Learning. Applied Sciences, 10, 370. https://doi.org/10.3390/app10010370

[30] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z. (2016) Rethinking the Inception Architecture for Computer Vision. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 27-30 June 2016, 2818-2826. https://doi.org/10.1109/CVPR.2016.308

[31] Aman Mishra (W1600017) ,Kevin Lan(W1628780), "Deepfake Detection" ML_deepfakeDetection_21m.pdf

[32] https://images.app.goo.gl/PnNgqayxSTLJ4Tgq9

[33]https://www.edureka.co/blog/wp-content/uploads/2017/05/Deep-Neural-Network-What-is-Deep- Learning-Edureka.png

[34] https://images.app.goo.gl/TCD6LFWb6HiaUo9n7