

A Deepfakes Detection Model for Face Forgeries Generated by Generic GAN in Social Media

A Thesis submitted to the

UPES

For the award of

Doctor Of Philosophy

In

Computer Science & Engineering

By

Preeti

October 2024

Under the guidance of

Supervisor

Dr. Hitesh Kumar Sharma

Dr. Manoj Kumar



**Department of Computer Science
School of Computer Science (SOCS)
UPES
Dehradun – 248007: Uttarakhand (India)**

A Deepfakes Detection Model for Face Forgeries Generated by Generic GAN in Social Media

A Thesis submitted to the

UPES

For the award of

Doctor Of Philosophy

In

Computer Science & Engineering

By

Preeti

(SAP ID: 500089519)

October 2024

Under the guidance of

Internal Supervisor:

Dr. Hitesh Kumar Sharma

Professor, UPES, Dehradun, India

External Supervisor:

Dr. Manoj Kumar

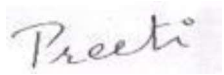
Associate Professor, University of Wollongong, Dubai, UAE



**Department of Computer Science
School of Computer Science (SOCS)
UPES
Dehradun – 248007: Uttarakhand (India)**

DECLARATION

I declare that the thesis entitled "**A Deepfakes Detection Model for Face Forgeries Generated by Generic GAN in Social Media**" has been prepared by me under the guidance of Dr. Hitesh Kumar Sharma, Professor, UPES, Dehradun and Dr. Manoj Kumar, Associate Professor, University of Wollongong, Dubai. No part of this thesis has formed the basis for the award of any degree or fellowship previously.



Preeti

School of Computer Science

UPES

Bidholi via Prem Nagar

Dehradun-248007, Uttarakhand, India

Date: 21.10.2024

CERTIFICATE

I certify that Preeti (SAP ID: 500089519) has prepared her thesis entitled “A Deepfakes Detection Model for Face Forgeries Generated by Generic GAN in Social Media” for the award of PhD degree of the UPES, under my guidance. She has carried out the work at the Department of Computer Science, UPES.

Internal Supervisor

Dr. Hitesh Kumar Sharma

Professor and Cluster Head
(CSO Cluster)
School of Computer Science
UPES,
Bidholi via Prem Nagar
Dehradun-248007, Uttarakhand, India
Date: 21.10.2024



UNIVERSITY
OF WOLLONGONG
IN DUBAI

To Whom It May Concern

I certify that Preeti (SAP ID: 500089519) has prepared her thesis entitled "A Deepfakes Detection Model for Face Forgeries Generated by Generic GAN in Social Media" for the award of PhD degree of the UPES, under my guidance. She has carried out the work at the Department of Computer Science, UPES.

External Supervisor

Dr. Manoj Kumar

Associate Professor

School of Computer Science, FEIS,

University of Wollongong in Dubai, Dubai Knowledge Park, Dubai, UAE

Date: 21.10.2024

A handwritten signature in blue ink that reads "Manoj Kumar".



Dr. Manoj Kumar

Associate Professor, Cyber Security

School of Computer Science, University of Wollongong in Dubai, Dubai Knowledge Park,
UAE Email: manojkumar@uowdubai.ac.ae

ABSTRACT

The rise of generative learning technology aided the generation of realistic fake images that are tricky to discriminate from real ones. DeepFakes involve altering facial appearances through deep generative techniques. Recent advancements in generative adversarial networks have enabled the generation of human facial images that even experts find hard to identify as fake. The proliferation of such images posted on the internet if utilized improperly, they might give rise to serious moral, legal, and ethical problems. Improving the capacity of forensic models to identify these newly created GAN-generated pictures becomes an essential field of study since new kinds of GANs emerge quickly. The forensics community continues to create new ways to detect these artificially manufactured fake images, especially face forgeries, and attempt to secure visual content's credibility. In this research, we investigate this crucial issue of forensics model generalization and propose a deepfake detection model that can be used for forensic investigation for complete face synthesis detection. This model achieves generalization to GANs by utilizing an ensemble-based discriminator architecture. The new GAN ensemble model aims to address common issues like Convergence, Instability, Mode Collapse, etc as knowledge is generated from a variety of samples. The suggested model is designed to minimize the differences associated with the generated and original distributions with employment of multiple CNN discriminators in a voting ensemble framework. This approach has proven to be reliable, delivering high accuracy in detecting forgeries. Key findings of the research show that this innovative technique outperforms existing GAN models on a limited dataset, as measured by metrics such as Fréchet Inception Distance, Structural Similarity Index Metric, the Inception Score, and Total Computational Time Function. Furthermore, the ensemble-based discriminator architecture not only enhances the model's robustness in identifying fake images but also demonstrates scalability in various practical applications. By leveraging the strengths of multiple discriminators, the model can adapt to new and evolving types of GAN-generated forgeries, maintaining high detection accuracy. This adaptability is crucial in an era where generative technologies are rapidly advancing. The research underscores the importance of continuous improvement and evaluation of forensic models to keep pace with the sophisticated methods used to create fake images, ensuring the integrity and reliability of digital media.

ACKNOWLEDGMENT

I am deeply grateful to my PhD supervisors, Dr. Hitesh Kumar Sharma and Dr. Manoj Kumar, for their unwavering constant support and guidance, which have been instrumental throughout my doctoral journey. Their expertise, patience, and encouragement have played a pivotal role in shaping my research and academic growth.

I extend my heartfelt thanks to Dr. Ram Sharma, Chancellor of UPES, Dr. D K Avasthi, Dean R&D, Dr. Vijaysekhar Chellaboina, Dean of the School of Computer Science at UPES, and Dr. Vinod Patidar, Associate Dean of the School of Computer Science at UPES, for providing me with the invaluable opportunity to pursue my doctoral studies under their esteemed institution. Their commitment to academic excellence has provided me with a conducive environment to thrive and excel in my research pursuits.

I am indebted to the entire organization of UPES for fostering a supportive research environment that has enabled me to explore and delve deeper into my research interests. The resources, facilities, and collaborative atmosphere have significantly contributed to the fruition of my thesis and academic endeavors. Additionally, I would like to convey my gratitude to my colleagues at UPES, whose camaraderie and intellectual exchange have enriched my educational experience. Their insights and discussions have broadened my perspectives and inspired me to push the boundaries of my research.

Last but certainly not least, I am immensely thankful to my family members, especially my husband, mother, and siblings, for their unwavering love, encouragement, and understanding throughout this journey. Special mention goes to my son, Vagesh Vitthal Sharma, whose contagious energy, innocent laughter, and unconditional support have been my ultimate source of energy. His self-sufficiency and resilience, even at the tender age of 10, have motivated me to persevere and complete my thesis.

Table of Contents

DECLARATION	iii
CERTIFICATE (Guide).....	iv
CERTIFICATE(Co-Guide).....	v
ABSTRACT	vi
ACKNOWLEDGMENT.....	vii
List of Contents	viii-xi
List of Figures	xii
List of Tables	xv
List of Abbreviations	xvii
CHAPTER- 1	1
Introduction	1
1.1 Overview	1
1.2 Background	5
1.3 Motivation	7
1.4 Research Gap	8
1.5 Research Questions.....	9
1.6 Objectives	9
1.7 Research Contribution	10
1.8 Thesis Outline	11
CHAPTER- 2	13
Theory and Related Work	13
2.1 Deepfake: Definition and Implications	13
2.2 Overview of Diffusion Models and CNN in Deepfake Detection.....	14
2.3 Loss Functions for Deepfake Detection	17
2.3.1 (BCE) Binary Cross-Entropy Loss	17
2.3.2 (MSE) Mean Squared Error	17
2.3.3 Hinge Loss	18
2.3.4 Focal Loss	18
2.4 Optimizer for Training Deepfake Detection Models	18
2.5 Applications and Implications of Data Generation Using GANs	20
2.7 GANs in Deepfake Generation and Social Media Propagation	23
2.8 Key Findings and Contributions of Research in Deepfake Detection	25
CHAPTER- 3	26

Literature Review	26
3.1 Literature Review	26
CHAPTER- 4	44
A Novel Generalized Detection Approach for Image Forgeries Using Generative Adversarial Network	44
4.1 Introduction	45
4.2 Related Research Work.....	47
4.2.1 Pix2Pix	47
4.2.2 Cycle GAN.....	49
4.2.3 DC-GAN.....	51
4.3 Proposed Methodology	53
4.3.1 Generative adversarial Network (GANs).....	53
4.3.2 Description of Proposed Ensemble Model	55
4.3.3 Max voting Approach	56
4.3.4 GAN Ensemble.....	57
4.3.5 Detailed Architecture	58
4.3.6 Dataset.....	61
4.3.6 Algorithm	62
4.3.7 Mathematical Construct	63
4.3. 8 Training.....	65
4.3.9 Training algorithm	65
4.3.10 Generator and discriminator training details.....	66
4.3.11 Evaluation Parameters	70
4.4 Experimentation	71
4.4.1 Hardware Requirements.....	71
4.4.2 Software Technologies Requirements.....	71
4.5 Results and Discussion.....	73
4.5.1. Plot image generated by the generator before training	73
4.5.2 Images Generated by the GAN (Through Iterations 1, 3, 5, 7,11, 15, 20....1000).....	74
4.5.3 Proposed GAN model Accuracy Plots.....	76
4.5.4 Proposed GAN model mode collapse plot.....	77
4.5.5 Proposed GAN model convergence plot.....	79
4.5.6 IS (Inspection Score) Curve	80
4.5.7 FID (Fréchet Inspection Distance) Curve	81
4.5.8 Total Computational time plot	82

4.5.9 SSIM Curve refers to the Structural Similarity Index Metric Curve.	84
4.6 Conclusion and Future Scope	89
CHAPTER- 5	91
Heterogeneous Discriminator Approach with Ensemble WGAN (EWG) model for Enhanced Image Synthesis and Deepfake Detection	91
5.1 Introduction	92
5.2 Key Innovations and Features of EWG	94
5.3 Related Research Work	95
5.3.1 Wasserstein GAN (WGAN)	96
5.3.2 Evolutionary Methods for Deepfake Detection	98
5.4 Proposed Methodology	98
5.4.1 Mathematical Notation of proposed GAN function	103
5.4.2 Training Algorithm	104
5.4.3 Dataset	105
5.4.4 Model Evaluation Parameters	106
5.5 Experimentation.....	107
5.5.1 Experimental Setup	107
5.5.2 Training	107
5.5.2.1 Input Parameters	108
5.5.2.2 Hyperparameters Scheme	108
5.5.3 Training Procedure	109
5.5.4 Layers Architecture	110
5.6 Results and Discussion	111
5.7 Conclusion and Future Scope	123
CHAPTER- 6	124
VOTSTACK: Ensemble Approach for classification of GAN-Generated Deepfake Images	124
6.1 Introduction	125
6.2 Related Research Work	129
6.3 Description of Methodology	133
6.3.1 Method	135
6.3.2 Algorithm	138
6.3.3 Dataset	140
6.3.4 Evaluation.....	141
6.3.4.2 Classification Accuracy.....	141
6.3.4.4 Recall	142

6.3.4.5 F1-score	142
6.4 Experimentation	142
6.4.1 Training	143
6.5 Results and Discussion.....	144
6.6 Conclusion and Future Scope.....	153
CHAPTER- 7	154
Conclusion and Future Scope	154
7.1 Conclusion.....	154
7.2 Future Scope.....	155
References.....	158
Publications	179

List of Figures

Figures	Title	Page No.
1.1	An Example of Deepfake on social media	16
1.2	Various Passive Forgery Detection Techniques	19
1.3	Review of advancement of GAN with different Architectures	20
4.1	The architecture of Pix 2 Pix GAN	48
4.2	Structure of Cycle GAN	50
4.3	DCGAN Generator Architecture	51
4.4	Visual representation of the Generative Adversarial Network (GAN)	54
4.5	Enhanced GAN Model Utilizing Ensemble Architecture	57
4.6	The proposed GAN Model uses a layered Architecture	58
4.7	Facemask Detection Dataset (FDD) dataset of real and fake images	59
4.8	Design of Generator for ensemble GAN Model	64
4.9	Design of Discriminator 1 for ensemble GAN Model	65
4.10	Design of Discriminator 2 for ensemble GAN Model	66
4.11	Design of Discriminator 3 for ensemble GAN Model	67
4.12	Production of images by GAN generator without prior training	70
4.13	Images generated with successive iterations (1,3,7,11....20)	71
4.14	Images generated with successive iterations (500)	71
4.15	Images generated with successive iterations (500)	72
4.16	Bar graph(a), Pie chart (b) illustrating the accuracy curve attained by the suggested Ensemble Model at its highest level	72-73
4.17	Mode Collapse line plot of suggested GAN model for fake detection	75
4.18	Charts showing generator performance in suggested ensemble GAN model with better convergence	76
4.19	Proposed GAN model Inspection Score (IS) curve under 2- scenarios(a): all standard and (b) with specified after outlier study	76-77
4.20	Proposed GAN model Fréchet Inspection Distance (FID) curve under 2- scenarios(a): all standard and (b) with specified after outlier study	77-78
4.21	Graph displays total amount of time taken by proposed model to	78

	complete computational tasks.	
4.22	SSIM curve of Proposed GAN model by plotting 2-set of variables (a): all values and (b) with specified after outlier study	80
5.1	Basic WGAN Model	90
5.2	Block Diagram of Ensemble Wasserstein (EWG) Model showing modified WGAN model using single generator with three different discriminators	92
5.3	Detailed Architecture of proposed EWG Model	93
5.4	Steps of Execution of proposed EWG Model	94
5.5	Input Image Size (64x64x3) used for execution of proposed model	96
5.6	Description of real images (a) Indian actor Images (b) 5- celebrity Faces Datasets used for execution of proposed Model	96
5.7	(a), (b) (c) and (d): Layered architecture of Generator, Discriminator 1, Discriminator 2, Discriminator 3	99
5.8	Quality of images generated by proposed EWG model over successive epochs (1-1000)	100
5.9	Accuracy graph generated by EWG Model for best chosen discriminator (D3) for (a) Indian actor Images (b) 5- celebrity Faces Datasets	101
5.10	Convergence graph generated by EWG Model for best chosen discriminator (D3) for (a) Indian actor Images (b) 5- celebrity Faces Datasets	101
5.11	Graph showing Mode collapse and Convergence values generated by proposed EWG Model for (a) Indian actor Images (b) 5- celebrity Faces Datasets	102
5.12	Gradient penalties values generated by proposed EWG Model for (a) Indian actor Images (b) 5- celebrity Faces Datasets	102
5.13	Graph showing computational Time (a) Bar Graph and (b) Pie Chart values generated by proposed EWG Model for Indian actor Images	103
5.14	Graph showing computational Time (a) Bar Graph and (b) Pie Chart values generated by proposed EWG Model for 5- celebrity Faces	104

	Datasets	
5.15	Graph showing SSIM values generated by proposed EWG Model for (a) Indian actor Images (b) 5- celebrity Faces Datasets	105
6.1	Example of Deepfake Image	110
6.2	GAN generated Images example	111
6.3	Architecture of proposed VOTSTACK ensemble model operates through a two-phase system	116
6.4	Integration of different components of proposed VOTSTACK ensemble approach	117
6.5	Description of Dataset	121
6.6	Example of Confusion Matrix	121
6.7	Description of Confusion Matrix of proposed VOTSTACK ensemble model achieved as per Real and Fake images	124
6.8	Accuracy Score of VOTSTACK Model with Base classifiers	125
6.9	Bar Graph showing comparative study of VOTSTACK ensemble model with other ensemble approaches utilizing different evaluation parameters (Accuracy, Precision, Recall, F1-score)	126
6.10	Description of Precision curve of VOTSTACK ensemble techniques achieved as per Real and Fake image dataset	127
6.11	Description of Recall curve of VOTSTACK ensemble techniques achieved as per Real and Fake image dataset	128
6.11	Description of F1-score curve of VOTSTACK ensemble techniques achieved as per Real and Fake image dataset	128

List of Tables

Table No.	Title	Page No.
1.1	Review of literature about domain from (2018-2023)	35-44
4.1	Proposed ensemble Model Function	56
4.2	Algorithm for ensemble-based GAN Model	59
4.3	Mathematical construct for ensemble-based GAN Model	60
4.4	Training Algorithm for ensemble-based GAN Model	62
4.5	Roles of the different Generator and Discriminator layers in an ensemble-based GAN model	63
4.6	Table demonstrating Accuracy attained by the suggested ensemble GAN Model throughout a series of epochs	73
4.7	Suggested that the ensemble GAN model's generator obtained mode values for different epochs.	75
4.8	Examination of suggested ensemble GAN model's IS values compared to existing models.	79
4.9	Suggested ensemble model's SSIM scores over time periods.	81
4.10	Review of current models and comparison of suggested GAN model's FID values.	82
4.11	Summarized comprehensiveness of IS and FID scores ensemble GAN model with other existing models with distinct respective datasets.	83
5.1	Mathematical Notation of proposed GAN function	94
5.2	Training Algorithm used for training of Proposed Ensemble WGAN model	95
5.3	Description of input parameter during training and generation	97-98
5.4	Showing comparative scores IS, FID, SSIM, and Total computational time parameter of EWG with existing WGAN model	106

5.5	Showing the Comparative Scores of Accuracy Parameter of EWG Model (*) with Existing GAN Model	106-107
5.6	Showing comparative scores IS and FID with the existing WGAN model	107
6.1	Table showing Algorithm for proposed VOTSTACK model	120
6.2	Accuracy score of VOTSTACK ensemble techniques with base classifier	125
6.3	Comparison study of variances ensemble techniques with proposed VOTSTACK ensemble approach	126

List of Abbreviations

GAN	Generative Adversarial Network
EUROPOL	European Union Agency for Law Enforcement Cooperation
UNICRI	United Nations Interregional Crime and Justice Research Institute
CGAN	Cycle GAN
PGGAN	Progressive Growing GAN
WGAN	Wasserstein GAN
VAEs	Variational Autoencoder
MMDA-VAE	Multi-Modal Domain Adaptative Variational Autoencoder
DBN	Deep Belief Network
TL-GDBN	Growing DBN with Transfer Learning
ARTN	Adversarial Residual Transform Network
DCT	Discrete Cosine Transform
DCGAN	Deep Convolutional Generative Adversarial Network
HOG	Histogram of Oriented Gradients
MRGAN	Manifold Regularized Generative Adversarial Network
DWT	Discrete Wavelet Transform
YLGAN	Yet Another Large-scale Generative Adversarial Network
SIFT	Scale-Invariant Feature Transform
PGGAN	Progressive Growing of Generative Adversarial Networks
LBP	Local Binary Patterns
SAGAN	Self-Attention Generative Adversarial Network
WGAN-ST	Wasserstein Generative Adversarial Network with Stochastic Gradient Descent
SS GAN	Semi-Supervised Generative Adversarial Network
CNN	Convolutional Neural Network
BIGGAN	Big Generative Adversarial Network
SN GAN	Spectral Normalization Generative Adversarial Network
RNN	Recurrent Neural Network
SVM	Support Vector Machine

RMSE	Root Mean Squared Error
UQI	Universal Quality Image Index
SSIM	Structural Similarity Index
MS-SSIM	Multiscale Structural Similarity Index
VIF	Visual Information Fidelity
BCE	Binary Cross Entropy Loss
SGD	Stochastic Gradient Descent
IS	Inception Score
FID	Fréchet Inception Distance
TCT	Total Computational Time

CHAPTER- 1

Introduction

1.1 Overview

Digital image forgery is a complex area involving the alteration and manipulation of digital images, presenting significant challenges for society. Among these challenges, "Deepfakes" is a particularly intricate form of forgery. Deepfakes are a type of fake image or video that are produced by replacing one person's face with another using sophisticated approaches in deep learning. Many ethical, societal, and security concerns—like the propagation of fraud and fake news—have been brought up by this phenomenon.

The prevalence of social networks has accelerated the dissemination of misleading content, particularly in the form of fake images and videos. This spread occurs at a faster pace on social media platforms compared to traditional mainstream media outlets. Developing new user-friendly technologies and advanced image processing tools or editing software have made it easy to create tampered images and make false propaganda for blackmail. The United Nations Interregional Crime and Justice Research Institute (UNICRI) and EUROPOL collaborated on a 2020 document that described the harmful applications of artificial intelligence and anticipated the adoption and exploitation of deepfake technology by cybercriminals. We are already witnessing attacks taking place in the wild, therefore it didn't take long for our forecast to come true.

Deepfake assaults are becoming more frequent, and this is drastically changing the danger picture for businesses, financial institutions, public figures, celebrities, and even regular individuals. We have already seen some users of underground forums looking for "deepfake specialists" for personal accounts and cryptocurrency exchanges in 2020 and the first part of 2021. As a matter of fact, there have long been internet resources for producing deepfakes, such as GitHub. As a result, people are beginning to respect photographs' authenticity more and

more. In deepfakes, face manipulation can be done using four techniques majorly, (I) complete face synthesis, (II) identity swap, (III) attribute manipulation, and (IV) expression swap [1]. Figure 1 shows an example of a manipulated or doctored image depicting Jeffrey Wong Su En while receiving the award from Queen Elizabeth II, published in Malaysian dailies, and the original picture of Ross Brawn receiving the award from the Queen.

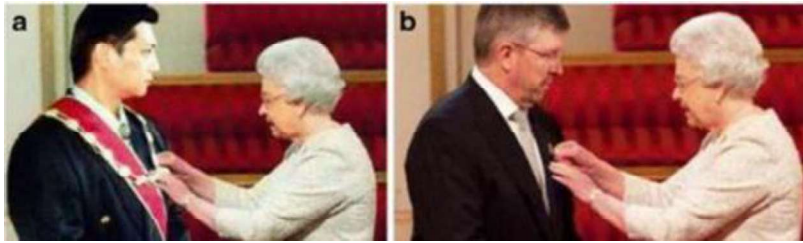


Figure 1.1: An example of —Deepfakes‖ on social media.

GAN (Generative Adversarial Networks) plays a critical role in the creation as well as detection of fake images. GAN-generated face images are becoming increasingly realistic and high-quality, making them difficult for human eyes to detect. New types of GANs emerge at a rapid pace, and the ability of forensics models to detect new types of GAN images is becoming increasingly difficult. This enquires into the ultimate requirement of forensics model generalization ability. GAN consists of two models‘ —Generator‖ and —Discriminator‖. The discriminator in this scenario is a deconvolution as its neural network, whereas the generator is a neural network based on convolution. The generator's mission is to provide outputs that the discriminator module may readily mistake for actual data. The issue lies in discriminators' strong performance in identifying phony faces for the generators they are used with. They perform worse, nevertheless, if they are forced to operate with unidentified sources or datasets. The requirement for a generic model for detecting face forgeries that is not reliant of generator type is highlighted by this issue. One generator is combined with an ensemble of discriminators in a generic GAN

model that is suggested as a solution to this issue. This approach outperforms as it can be exceptional to model the distribution of normal data and thus better face forgery detection [2].

GANs' increasing popularity may be ascribed to their varied architectures and exceptional capacity to adjust to many visual styles. High-resolution, photorealistic face pictures have been produced with considerable success by sophisticated models such as StyleGAN [4], CycleGAN (CGAN) [5], and Progressive Growing GAN (PGGAN) [6]. Furthermore, additional advancements in image synthesis have been made by autoregressive models such as transformer GANs [7],[8] and [9]. Additionally, Ji et al. [12] propose a deep transfer ensemble learning technique for SOH estimation with limited data, and Ohoto et al. [13] suggest an automatic technique for detecting COVID-19 utilizing pictures from X-rays. Their dataset includes 194 X-ray images of both COVID-19 patients and healthy individuals. With 98.5% accuracy and an F1 score, the study discovered that the architecture of Mobile Net combined with a Simple vector machine classifier with a kernel (linear) produced the best results.

An innovative method for learning shared common latent representations across domains for multimodal data is the multi-modal domain adaptive variational autoencoder which was introduced by Wang et al. [14]. To improve learning speed and model correctness, they also incorporate a growing DBN with transfer learning (TL-GDBN) [15]. A new adversarial domain adaptation technique called adversarial residual transform networks (ARTNs) [16]. Its goal is to increase generalization by transforming source features into the target feature space. With an emphasis on feature matching, adversarial adaptation, and model and parameter transfer, Yao et al. [17] organize a thorough review of transfer learning approaches in machinery problems—Adversarial adaptation, feature matching, and model and parameter transfer are utilized in machinery diagnostics and prognostics. Consequently, a variety of GAN algorithms have been created to overcome basic challenges with training of GAN and good-quality image generation. To improve system performance on tasks like classification or

prediction, machine learning practitioners frequently smartly combine models using ensemble learning. By precisely representing the distribution of normal data, an ensemble of GANs may carry out tasks like reconstruction or manipulation detection more efficiently.

This research highlights significant gaps in the application of GANs for face forgery detection on social media. GAN-generated images are typically specific to their respective architectures, leading to a degradation in performance when applied to unknown datasets encountered in wild scenarios. This specificity underscores the urgent need for generalized GAN models capable of robust performance across diverse datasets. Moreover, the rapid advancements in deepfake technology have exacerbated the difficulty of distinguishing between real and fake images, demanding substantial improvements in classifier precision. Additionally, persistent training issues such as non-convergence, mode collapse, and diminished gradients continue to plague GANs, requiring effective solutions to enhance the reliability and accuracy of fake detection systems.

To address these gaps, the research focuses on developing a deepfake detection model that utilizes a generalized GAN architecture integrated with ensemble-based discriminator techniques to boost overall performance. The main goal is to develop a model that exhibits strong cross-dataset generalization, which will enhance the discrimination accuracy and picture quality produced. On the discriminator side, an ensemble technique is constructed using several discriminators and a single generator to minimize the difference between the produced and original distributions. The paper also attempts to assess and confirm the improved detection capabilities of the ensemble method by examining the efficacy of the suggested ensemble GAN model in comparison to a parallel version that uses conventional machine learning classifiers.

The research aims to develop a robust system for detecting and countering fake images across various applications by tackling fundamental GAN training issues and employing an ensemble of discriminators. Forensic investigations, criminal investigations, insurance processing, surveillance systems, radiology,

and journalism are some of these uses. To improve GAN pathologies such as instability, convergence, etc., the suggested model, which uses several CNN discriminators employing a voting ensemble methodology, is intended to reduce the differences between generated and original distributions. Research shows that this new approach achieves good accuracy values as measured by parameters like the Structural Similarity Index Metric (SSIM), Fréchet Inception Distance (FID), Total Computational Time Function, and Inception Score (IS), outperforming existing GAN models on small datasets. By improving the resilience and effectiveness of GANs, this novel method seeks to provide a more potent instrument for identifying face forgeries and safeguarding the integrity of digital information.

1.2 Background

According to Official data stated by the National Crime Record Bureau (NCRB) on the date September 15, 2020—50,035 cybercrime events were reported, representing an 11.8 percent rise from the previous year—and 578 instances of "false news on social media" were also reported in India. In today's time, it is extremely tough to identify real and fake images with the naked eye, especially in social media. The process of producing phony pictures or films, known as "Deepfakes," involves replacing an individual's face with another people. The term originated from Redditor by the name of "deepfakes," who in 2017 claimed to devise a ML algorithm capable of superimposing celebrity faces onto pornographic videos.

Traditionally, tampering with the original image was the primary cause of creating fake faces. So, conventional fake images have been left with left traces, causing statistical data to change. With the help of these statistics, researchers have been able to detect fake faces [18]. DCT, HOG, DWT, SIFT, LBP etc. are some of the famous conventional forgery detection methods which produces very good result in this domain. Figure 2 represents the major passive forensic detection methods.

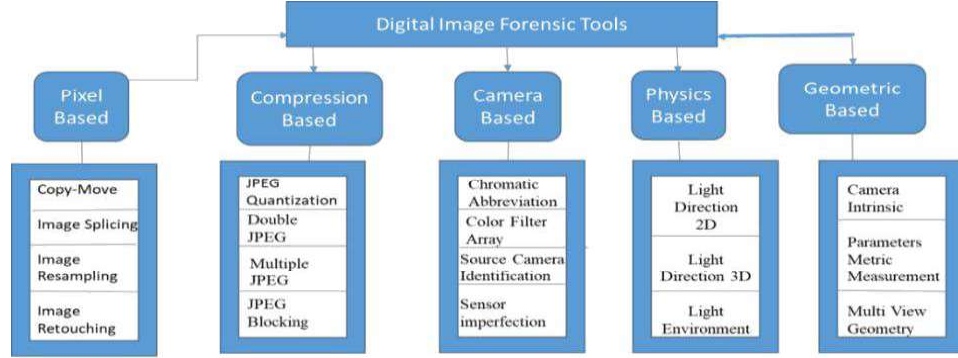


Figure 1.2: Various Passive forgery detection techniques

Next comes the era of Deep learning-based techniques for forgery elimination which automated the conventional methodologies to a very big extent. CNN, RNN, SVM, Random Forest, etc. are a few neural networks that are widely used. Nowadays fake faces in social media seem too real to be distinguished which are difficult to identify by these conventional methods. Since GAN creates pictures using a random vector, the produced faces are devoid of any manipulation information. GAN defines as generative model that composes two models of CNN networks to train it. The generator or generative real is one example of a model that learns to produce fresh believable examples. The discriminator or discriminative network model, on the other hand, learns to distinguish produced from actual samples. The two models compete or play a game in which the generator model attempts to deceive the discriminator model by providing both actual and produced samples to the discriminator [19]. The link between GAN evaluation and deep fakes is still being worked out, and there is room for improvement. GAN evaluation can help mitigate this issue, which is a major source of concern in the AI community today. Our work focuses on face alteration, namely on the synthesis of full facial pictures. The main contribution of

this study is to design a generic forensic model for detecting synthetic face images in social media.

The two primary types of GANs currently in use are: loss-variant and architecture-variant GANs. Within the second category, there are three subtypes: network architecture, application-focused, and latent space. Network architecture improvements, such as the progressive mechanism, bring upgradations to the overall GAN design. The category of latent space includes architectures that adjust based on different latent space representations; for example, in CGAN, the label information is integrated into both the discriminator and generator. Common evaluation metrics for assessing the performance of GANs include Root Mean Squared Error (RMSE), Visual Information Fidelity (VIF), Multi-scale Structural Similarity Index (MS-SSIM), and Universal Quality Image Index (UQI).

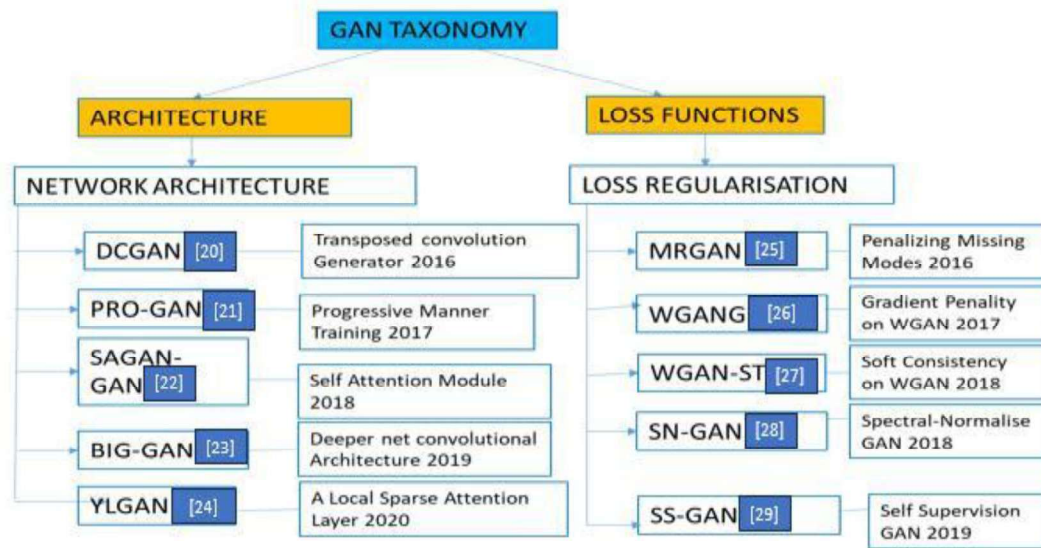


Figure 1.3: Review of advancement of GAN with different architectures

1.3 Motivation

In today's digital world, digital images serve as the fastest means of information dissemination across various platforms, including digital media, forensic applications, social media, medical sciences, and legal systems. However, the increasing prevalence of digital image forgery imposes a significant threat to the

integrity of these images. One prominent example is facing forgery, where parts of one image, such as a person's head, are composed onto another person's body. This practice is especially problematic in social networking environments, where the circulation of forged images can lead to misinformation and even serious crimes. Ensuring the trustworthiness of digital images is crucial for crime investigations, legal evidence, medical applications, and surveillance, highlighting the necessity for advanced solutions in this domain.

The urgent need for new, robust, and generic forgery detection techniques is evident as society grapples with the challenges posed by digital image forgery. Devising methods to accurately detect and prevent forgeries can significantly increase the credibility of digital images used in various critical sectors. Face forgeries, for example, can compromise the authenticity of evidence in criminal investigations, mislead medical diagnoses, and spread false information on social media, potentially leading to severe consequences. The ability to reliably detect such manipulations is not a societal imperative, building the establishment of forgery efficient detection methods as a top priority.

The motivation for this work is derived from the critical need to advance the reliability and security of digital media against forgery attacks. By developing robust and generalized forgery detection techniques, this research aims to address these pressing issues, providing a tool that can effectively combat digital forgeries and secure digital media. The drive to contribute to making digital platforms more secure and resistant to manipulation fuels the commitment to this research, to protect individuals and institutions from the adverse impacts of digital image forgery. Ensuring that digital images remain trustworthy when used in crime investigations, as legal evidence, in medical diagnostics, and for surveillance purposes is a key motivation behind this work.

1.4 Research Gap

- GAN-generated images are specific to their architecture. Performance degrades in wild scenarios for unknown datasets. So, generalization of GAN is an urgent need for face forgery detection in social media.

- Classification between real and fake face images is getting difficult with advancement of deepfake technology and so, classifiers require further improvement.
- Non convergence, Mode collapse, diminished gradients etc. are few training pathologies of GAN which require to be minimized or resolved for better detection of fakes.

1.5 Research Questions

Question 1: Do a GAN model offers an efficient method for detecting deep fakes in images shared on social media?

Question 2: Would the ensemble approach offer a way to overcome the significant constraint of working with challenging data set in deep learning models?

Question 3: Can a new GAN ensemble architecture be developed that substantially improves GAN training issues?

Question 4: Can the integrated ensemble technique act as a feasible solution to generate high-quality images for GAN?

1.6 Objectives

Main Objective:

The objective of this work is to design a deepfakes detection model for face forgeries generated by generic GAN.

Sub-Objectives:

- Develop a generalized Generative Adversarial Network (GAN) model through the integration of ensemble-based discriminator architectures to improve GAN performance.
- Boost the model's generalization to increase discrimination detection capability, which will enhance the model's general precision and the caliber of pictures it generates.

- Investigate the proposed ensemble GAN model against a parallel version using machine learning classifiers to evaluate the ensemble's detection capability.

1.7 Research Contribution

This research identifies a significant advancement in the domain of deep fake detection and image generation by introducing generalized GAN models adept at detecting deep fakes across a spectrum of datasets, encompassing street view and MR images. Through exhaustive training and meticulous evaluation across diverse image domains, the study demonstrates the potential of GANs to exhibit robust generalization, marking a pivotal progression in the endeavor to identify manipulated images disseminated across social media platforms. Furthermore, the exploration of ensemble techniques offers a promising avenue to surmount the challenges posed by intricate datasets, thereby bolstering the precision and robustness of models for detection of deepfakes in real-world.

Moreover, the study's focus on devising an enhanced GAN ensemble architecture signifies a critical stride forward in mitigating common training pathologies such as non-convergence and mode collapse. By amalgamating innovative architectural modifications and learning concepts, the research endeavors to furnish a more dependable and efficient solution for detecting forged images and producing high-fidelity synthetic images. This approach holds immense potential for enhancing the stability and efficacy of GAN training processes, laying a robust foundation for the advancement of more optimized deepfake detection and image generation methodologies.

Additionally, this study emphasizes how critical it is to point the urgency arises in the field of deepfake detection as well as picture production. By pushing the boundaries of GAN generalization and harnessing the power of ensemble techniques, the study offers practical strategies to fortify defenses against the proliferation of fake images, thereby enhancing and protecting the integrity of digital media platforms. The insights gleaned from this research

not only contribute to the advancement of technological solutions but also underscore the critical role of interdisciplinary collaboration in combatting digital misinformation and preserving trust in online content.

1.8 Thesis Outline

Chapter 1. **Introduction:** An outline of the research background and motivation, articulates the problem statement, delineates the specific research objectives, outlines the key contributions of the thesis, and presents a structured outline of the subsequent chapters.

Chapter 2. **Theory and Related Work:** This chapter comprehensively reviews topics like Deepfake: Definition and Implications, Overview of Diffusion Models and CNN in Deepfake Detection, Loss Functions for Deepfake Detection, Optimizer for Training Deepfake Detection Models, Applications and Implications of Data Generation Using GANs, Deepfake in social media propagation and other relevant theory that provides a good background to understand the idea of this research.

Chapter 3. **Literature Review:** This chapter presents systematic review of the literature was conducted, encompassing a detailed analysis of published articles from 2018 to 2023 on three critical domains: deepfake social media threats, GAN-based deepfake approaches, and current deepfake detection techniques. This review highlights the emergent evolution of deepfakes technology and growing significance on various sectors, including media, security, and personal privacy.

Chapter 4. **A Novel Generalized Detection Approach for Image Forgeries Using Generative Adversarial Network:** This chapter introduces a unique and robust image forgery detection method, elucidates its underlying methodology and architectural design, describes the

experimental setup and dataset utilized for evaluation, analyzes the results obtained using established evaluation metrics, and discusses the findings.

Chapter 5. Heterogeneous Discriminator Approach with Ensemble WGAN (EWG) model for Enhanced Image Synthesis and Deepfake Detection: This section introduces the Ensemble WGAN (EWG) framework, delineates its implementation details and architectural design, provides a comparative analysis with traditional GAN models, presents experimental evaluations on deepfake detection tasks, and discusses the efficacy of the heterogeneous discriminator approach employed.

Chapter 6. VOTSTACK (Voting + Stacking) Ensemble: A Robust PCA-Based Technique for Classification of GAN-Generated Images: This chapter introduces the VOTSTACK ensemble technique, elucidates its methodology and algorithmic framework, explores the application of Principal Component Analysis (PCA) in image classification, presents performance evaluations and comparative analyses, and provides insights into the robustness and efficiency of VOTSTACK ensemble.

Chapter 7. Conclusion and Future Scope: In this concluding chapter, a brief of the research findings is provided, highlighting the contributions to the field, discussing limitations, and suggesting areas for future research. Practical implications for applications are discussed, and closing remarks along with recommendations for further study are provided.

CHAPTER- 2

Theory and Related Work

2.1 Deepfake: Definition and Implications

Deepfakes defined as a unique technique that blends "deep learning" and "fake," describes artificial media where artificial neural networks replace one individual's likeness with another in existing photos or videos. Leveraging efficient deep learning models, particularly Generative Adversarial Networks, these technologies can generate extraordinarily realistic and meticulously crafted counterfeit images and videos. GANs function by having distinguished networks, the generator and the discriminator, competing against each other. While deepfakes have beneficial applications in domains such as virtual reality, filmmaking, and privacy enhancement, they also present considerable dangers. They can be exploited for wicked purposes as disseminating manipulating public opinion, misinformation, and tarnishing reputations, thereby eroding trust in media. This dual-use nature of deepfakes necessitates robust technological and legal strategies to mitigate associated risks.

A key component of deepfakes, generative adversarial networks use training between distinguished neural networks, a generator and discriminator. Images or videos are produced by the generator, and their veracity is assessed by the discriminator. Iterations of the process continue until the discriminator is unable to differentiate the created media from genuine, legitimate ones. Deepfakes have important and varied repercussions too. Positively, deepfake technology can be applied to virtual reality, video games, filmmaking, and as a means of maintaining anonymity and privacy in delicate settings. Additionally, historical recreations and instructional settings may find use for it. On the other hand, the adverse effects are significant and concerning. Goodfellow et al. (2014) [30] proposed the concept of GANs, which are foundational to the development

of deepfake technology. Ajder et al. (2019) [31] present a review of the landscape, threats, and impact of deepfakes. It also has provided a comprehensive overview of the technological capabilities and the threats posed by deepfakes. Chesney & Citron (2019) [32] have discussed the legal and societal implications of deepfakes, and the potential geopolitical and social implications of deepfakes, including misinformation campaigns. Paris & Donovan (2020) [33] has analyzed the implications of deepfakes in the manipulation of audio and visual information. Tolosana et al. (2020) [34] have surveyed the techniques and challenges in detecting deepfake videos. Li & Lyu (2019) [35] have proposed methods for improving the detection of deepfakes through digital forensics analysis. Mirsky & Lee (2020) [36] have offered a detailed survey of both the creation techniques and detection strategies for deepfakes.

Hence, deepfakes allow for the generation of extremely convincing false information that may be exploited for manipulation, defamation, and misinformation, which puts the idea of truth in the media at considerable risk. It is a powerful instrument that may be used to disperse political misinformation, sway common people's opinions, and affect democratic processes and international relations because of these qualities. Deepfakes also bring up moral issues with permission and the ownership of one's image and likeness. An individual's reputation may be harmed, as well as personal and professional consequences when their likeness is used without permission. While there are creative and protective uses for deepfake technology, its capacity to produce convincingly fake content can erode public confidence in the media and information, making technological and legal precautions imperative.

2.2 Overview of Diffusion Models and CNN in Deepfake Detection

Diffusion models and Convolutional Neural Networks (CNNs) are pivotal in deepfake detection. Diffusion models work by learning to reverse a noise-adding diffusion process, highlighting subtle statistical differences between real and synthetic images. This method is effective in detecting the imperfections and

artifacts introduced by GAN-generated deepfakes. Researchers like Dhariwal and Nichol have shown that diffusion models can achieve high performance in image synthesis tasks, suggesting their potential in image validation and authentication. On the other hand, CNNs excel in analyzing visual input, identifying patterns and irregularities that may be invisible to the human eye. They are particularly adept at spotting inconsistencies in facial expressions and textures, making them effective in deepfake detection. Studies by Afchar et al. and Rossler et al. demonstrate CNNs' high accuracy in detecting manipulated videos, showcasing their robustness against complex deepfake techniques.

Diffusion models, originally developed for generating complex data distributions, have recently been explored as a promising approach to the detection of deepfakes. These models work by learning to reverse a diffusion process that gradually adds noise to an image until only random noise remains. In the context of deepfake detection, the model learns the characteristic patterns of real versus synthetic imagery by attempting to reverse the diffusion process and reconstruct the original, unaltered image from a potentially manipulated one.

The principle behind using diffusion models for deepfake detection lies in their sensitivity to the subtle statistical differences between genuine and manipulated images. Since deepfake generation, particularly those using GANs (Generative Adversarial Networks), often introduces slight imperfections and artifacts, diffusion models can potentially highlight these inconsistencies by focusing on how images degrade and recover through the diffusion process. When it comes to combating the more complex textural and temporal irregularities brought forth by cutting-edge deepfake technology, this sensitivity can be especially useful. Although the field is still in its infancy, research on the use of diffusion models for deepfake detection has great promise. It was shown by Dhariwal and Nichol (2021) [37] that diffusion models might attain cutting edge performance in image production tasks, indicating that their potential for use in inverse tasks such as image validation and authentication could be highly

encouraging. Likewise, like Saharia et al. (2022) [38] have emphasized the resilience and flexibility of the model, emphasizing how important it is to adjust to the changing landscape of digital image alteration. Therefore, diffusion models present a new and potentially effective deepfake detection technique. Their capacity to simulate and undo intricate data transformations makes them an invaluable tool in the fight against digital disinformation since it enables a thorough examination of the veracity of images. Nichol & Dhariwal (2021) [39] presents advancements in image processing diffusion model methods. It enhanced diffusion model training methods were introduced, possibly improving the models' ability to recognize altered images. Ho et al. (2020) [40] performed basic research on probabilistic diffusion models denoising. He created the diffusion model framework, which prepared the way for using it in challenging image analysis applications.

Because of their ability to analyze visual input, Convolutional Neural Networks (CNNs) have been used extensively in detecting deepfakes. In pictures and videos, these networks are excellent at picking up on patterns and irregularities that the human eye would miss. When spotting the tiny distortions and artifacts introduced by deepfake technologies, CNNs are very good at spotting irregular blinking patterns, boundary abnormalities, and inconsistent facial texturing, which are common in altered films. To understand and discern the subtle characteristics that separate authentic from modified content, CNNs are used in deepfake detection. To do this, the models are trained using a dataset including real and fake images. Studies show that CNNs can get high degrees of accuracy in this field. Afchar et al. (2018) [41] proposed the introduction of MesoNet, a CNN-based model for video forgery detection that demonstrated that CNNs could effectively detect facial forgeries by focusing on discrepancies in facial expressions and movements. Rossler et al. (2019) [42] proposed the development of the Face Forensics++ dataset and detection model. He has presented a CNN-based approach to identify manipulated facial images as well as videos, proving high accuracy with extensive datasets. Nguyen et al. (2019) [43]

highlighted the effectiveness of CNNs in identifying deepfake content across various social media formats. Li and Lyu (2018) [44] CNNs have researched and focused on CNN's ability to identify specific artifacts like unnatural blinking patterns that are often overlooked by other models.

2.3 Loss Functions for Deepfake Detection

Loss functions play a crucial role in training neural networks for deepfake detection by quantifying the discrepancy between predicted outcomes and actual truth values. Binary Cross-Entropy Loss, Mean Squared Error (MSE), Hinge Loss, and Focal Loss are commonly used in deepfake detection tasks to optimize model performance. These loss functions are tailored to address specific challenges in discerning real from synthetic images or videos, including imbalanced datasets, regression-based tasks, and class imbalance. In the field of deepfake detection, where the task is to discern real from synthetically altered images or videos, several specific loss functions are commonly employed to optimize model performance.

2.3.1 (BCE) Binary Cross-Entropy Loss

Binary Cross-Entropy Loss is widely used in 0 or 1-based classification binary tasks, including deepfake detection. It measures the performance of a model whose output is a probability value between 0 and 1. This loss function penalizes the model based on the difference between the predicted probability and the actual class label (0 for real, 1 for fake). It is particularly effective for datasets with imbalanced classes, which is often the case in deepfake detection [45].

2.3.2 (MSE) Mean Squared Error

Mean Squared Error is an additional function that calculates loss that is occasionally employed in the detection of deepfakes, particularly in situations where regression-based tasks are involved, like estimating the degree of image modification. To provide a reliable indicator of model

accuracy across continuous output spaces, MSE computes the square of the difference between the anticipated and actual values [46].

2.3.3 Hinge Loss

"Support vector machines" and certain classification-focused neural networks frequently use hinge loss. It is designed for "maximum-margin" classification, making it useful in deepfake detection for ensuring that predictions not only are correct but also are made with confidence. This loss function is particularly beneficial when there is a need to distinguish clearly between classes [47].

2.3.4 Focal Loss

Focal Loss is especially effective in addressing class imbalances, a frequent challenge in deepfake detection. Incorporating a modulation factor into the standard cross-entropy loss, it diminishes the relative contribution of well-classified examples, thereby emphasizing the harder, misclassified ones. This adjustment ensures that the model focuses more on the challenging cases, improving overall detection performance. This property makes it ideal for training on imbalanced datasets where fakes might be much less common than real examples [48]. Li & Lyu (2018) [49] have utilized CNN to detect subtle manipulations by focusing on inconsistencies in facial expressions, such as blinking. It contributed to understanding specific vulnerabilities in deepfake generation techniques, particularly around human-like behaviors. Successfully identified deepfakes by detecting unnatural blinking patterns.

2.4 Optimizer for Training Deepfake Detection Models

Optimizers play a crucial role in training deepfake detection models by updating network weights to minimize the loss function, thus improving the model's accuracy and efficiency. Stochastic Gradient Descent (SGD), Adam, RMSprop, and AMSGrad are prevalent optimizers in deep learning for tasks like deepfake detection. These optimizers offer unique

characteristics and advantages, such as adaptability to sparse gradients, efficient convergence, and stability in training. Studies by Kingma & Ba and Zeiler have demonstrated the effectiveness of these optimizers in enhancing model convergence and performance, underscoring their importance in developing robust deepfake detection systems.

Stochastic Gradient Descent (SGD) is one of the most traditional optimizers, known for its simplicity and effectiveness, particularly in large and complex datasets. SGD updates parameters in a direction that minimally decreases the loss, using a learning rate to control the step size. However, it can sometimes lead to slower convergence on flat surfaces or oscillate around minima [50]. Adam, which stands for Adaptive Moment Estimation, combines the benefits of two other extensions of SGD—AdaGrad and RMSprop. Adam computes adaptive learning rates for each parameter. It not only considers the average of the first moments of the gradients (like momentum) but also the second moments (the uncentered variances). This makes it particularly effective for training deep neural networks on complex tasks such as deepfake detection, as it can handle sparse gradients and different parameter scales [51].

RMSprop is another optimizer that is effective in handling the non-stationarity of objectives, as is common in detecting deepfakes. It adjusts the learning rate by dividing it by an exponentially decaying average of squared gradients, which helps in rapid convergence in the initial stages of training and stabilizes the training process in the later stages. In deepfake detection, the choice of optimizer can significantly impact the model's performance, especially in how quickly and effectively it learns to differentiate between real and manipulated content. For instance, in the training of convolutional neural networks (CNNs) for deepfake detection, Adam and RMSprop are often preferred for their efficiency in dealing with large and unbalanced datasets, as evidenced in several studies and implementations [52].

Zeiler (2012) [53] has proposed the development of RMSprop, which adjusts the learning rate based on a moving window of gradient updates for each weight. It showed effectiveness in accelerating convergence in non-stationary objectives compared to standard SGD. RMSprop's ability to stabilize training makes it suitable for deepfake detection models dealing with highly variable data. Diederik & Hinton (2012) [54] have utilized momentum to improve the convergence rate of SGD by considering past gradients in the update process. It has achieved faster convergence in training deep neural networks than using standard SGD alone. The incorporation of momentum is crucial for deep learning tasks, including deepfake detection, where escaping local minima is essential. Reddi et al. (2018) [55] have proposed AMSGrad, a variant of Adam, which corrects the converging issue by using the maximum of past squared gradients. It showed improved stability and performance over Adam in certain scenarios. AMSGrad addresses the shortcomings of Adam in deepfake detection by providing more reliable convergence in diverse datasets.

2.5 Applications and Implications of Data Generation Using GANs

GANs have become a powerful tool for data generation across various domains, including images, videos, music, and text. Their ability to generate new data from existing distributions has significant implications, such as enhancing creativity, augmenting datasets for machine learning tasks, and raising privacy and ethical concerns. Existing studies have demonstrated the potential of GANs in generating realistic images and videos, contributing to fields like digital media, entertainment, and medical imaging. However, the ethical implications of GAN-generated content, particularly in the context of deepfake technology, raise concerns about privacy, security, and information integrity, necessitating careful regulation and ethical considerations in their use.

The core technique of GANs for deepfake creation and detection is based on the generator learning to generate ever more realistic images or videos that emulate the desired data distribution (genuine faces, for example). The generator tries to trick the discriminator by providing data that the discriminator will recognize as real in a training process, which is how this is accomplished. The discriminator's capacity to identify fakes is enhanced by learning from both authentic data and the fakes generated by the generator. The generator learns to produce very convincing fakes through this iterative training process, which ends when the discriminator is unable to identify actual from generated fakes.

Deepfake generation using GAN-based approaches yields impressive results. It is now possible for GANs to generate artificial images and movies that are strikingly like the input data they are trained on, making them hard to distinguish from real-world sources. The degree of realism and quality of these deepfakes has increased to the point where they can readily trick human observers, traditional fake detection systems, and discriminators inside the GAN framework. Deepfakes produced by GANs have wide-ranging and significant consequences. Positively, this technology can be applied for legitimate applications where realistic environments and characters are advantageous, including in virtual reality, video games, and filmmaking. But one major worry with this technology is that it could be abused. It can be used to create harmful or false content, which can have an impact on political processes, public opinion, and individual reputations. As a result, it poses dangers to security, privacy, and information integrity.

Karras et al. (2018) [56] developed Progressive GANs, which enhance the quality of generated images by increasing the resolution progressively during training. It achieved unprecedented realism in generated images, including human faces, at high resolutions. Zakharov et al. (2019) [57] have created a GAN that can animate a single photo into a

talking model using few-shot learning techniques. Successfully animated faces using very few images, demonstrating how few data points are needed to create convincing deepfakes. It also raises significant implications for the ease of generating deepfakes with limited data, affecting privacy and security. Schwartz et al. (2020) [58] have explored the use of autoencoding GANs for deepfake generation by reconstructing facial images while maintaining identity. It has demonstrated high efficacy in preserving facial identity even under transformations, improving the realism of deepfakes. Amplifies concerns about the misuse of deepfake tech in impersonation and fraud, pressing the need for effective detection technologies. Regarding applications, GANs have been utilized in various fields for data generation:

- **Realistic Image and Video Creation:** GANs are widely employed to produce realistic images and movies, ranging from made-up human faces to imaginary landscapes.
- **Medical Imaging:** They aid in the creation of medical images for research and training, preventing concerns about patient data privacy.
- **Fashion and Design:** GANs help in creating new fashion designs and product models by learning from existing styles and trends.
- **Text Generation:** Although more challenging, GANs have also been adapted to generate coherent and contextually relevant text.

Implications:

- **Enhanced Creativity:** They can assist artists and designers by providing new creative pathways and inspirations.
- **Data Augmentation:** In machine learning, GANs are used to augment datasets, especially where data may be scarce or expensive to collect, thus improving the robustness of models.
- **Privacy and Ethics:** While GANs can reduce reliance on real data, potentially mitigating privacy issues, they also raise ethical concerns, especially related to the creation of deepfakes and misinformation

[59].Antoniou et al. (2017) [60] have also utilized GANs for augmentation in machine learning, specifically for training models with limited data. It enhanced model performance in tasks with limited data by generating new, varied, and realistic samples.Shrivastava et al. (2017) [61] have indulged GANs to improve the realism of synthetic medical images for training surgical simulators.Synthetic images refined by GANs were mistaken for real images by expert reviewers in clinical settings. It also highlighted the potential of GANs in medical training and research, reducing reliance on real patient data and enhancing privacy.

2.7 GANs in Deepfake Generation and Social Media Propagation

Generative Adversarial Networks are central to the implementation of deepfakes, involving a generator that creates or augments data and a discriminator that analyses its authenticity. The generator learns to produce increasingly realistic images and videos through iterative training, leading to highly convincing deepfakes that can deceive both humans and automated detection systems. The proliferation of deepfakes on social media is particularly concerning due to their potential for rapid and widespread dissemination. GAN-generated deepfakes can be tailored to reflect current events or popular opinion, making them more likely to go viral. This ease of propagation poses significant ethical, legal, and security challenges, including the potential for deepfakes to influence political outcomes, spread misinformation, and create social discord. The implications are vast, highlighting the urgent need for advanced detection technologies, legal reforms, and public awareness to combat the misuse of deepfakes.

Generative Adversarial Networks (GANs) have had a profound impact on the field of digital media, particularly in how they are used in social media for the creation and propagation of deepfakes. These technologies allow for the creation of phony content that is incredibly convincing and frequently hard to tell apart, such as edited photos and

movies and artificial voices and text. This ability has important ramifications for social media because of how quickly information may spread there and compound the negative impacts of misleading media. GANs are used in social media to provide visually appealing and captivating content for business, entertainment, and even instructional uses. The negative side, on the other hand, entails the production of deepfakes that can be used to deceive, injure, or mislead both private citizens and public leaders. Because GAN-generated content seems realistic, viewers may find it difficult to discern between actual and fake media, which is especially problematic when it comes to news and private correspondence.

Deepfakes spread easily on social media because of the viral nature of these sites. GANs help with this by making it possible to quickly create convincing deepfakes that can be adjusted to reflect current affairs or popular opinion, which helps them acquire popularity among users. Due to its ability to sway public opinion, affect political outcomes, and even spark social upheaval, this spreading potential presents both a technological and a social issue. The implications of GANs in social media and deep fake propagation are significant. They pose ethical, legal, and security challenges. Ethically, the use of GANs in creating and spreading false information breaches principles of honesty and integrity. Legally, there are questions about liability and rights to one's image and voice.

Westerlund (2019) [62] has reviewed the emergence and development of deepfake technology, focusing on its applications and societal impact. He has demonstrated that deepfakes could significantly disrupt industries and global politics. He has also discussed the dual-use nature of deepfake technology and the critical need for societal and regulatory preparedness. Paris & Donovan (2020) [63] has analyzed the use of deepfakes in public discourse, focusing on the

dangers arising from truth and credibility in media. He has found that deepfakes are used to create false narratives and impersonate public figures.

2.8 Key Findings and Contributions of Research in Deepfake Detection

Research in deepfake detection has yielded significant findings and contributions that have advanced the field. Works by Goodfellow et al., Afchar et al., and Rossler et al. have introduced foundational concepts and methodologies, such as GANs, CNN-based detection models, and loss functions tailored for deepfakes detection. These studies have provided insights into the capabilities and limitations of deepfake technology, highlighting the importance of developing robust detection techniques and raising awareness about the potential risks. Additionally, research on optimizers and data generation using GANs has expanded the toolkit for deepfake detection, offering new avenues for improving detection accuracy and efficiency. Overall, these contributions have paved the way for addressing the challenges posed by deepfake technology and safeguarding the integrity of digital media and information.

CHAPTER- 3

Literature Review

3.1 Literature Review

A diligent review encompassed a detailed analysis of published articles from 2018 to 2023 on three critical domains: deepfake social media threats, GAN-based deepfake approaches, and current deepfake detection techniques. This review analyses the steady evolution of deepfake technology, and its growing impact on various sectors, including media, security, and personal privacy. In particular, the literature documents the significant implication of social media in accelerating the dispersal of deepfakes, which can be utilized for malicious or illegal operations such as spreading fake news, conducting fraud, and executing blackmail. The critical risks highlight the crucial need for efficient detection and mitigation tactics to safeguard individuals and organizations from the adverse effects of deepfakes.

In examining GAN-based deepfake approaches, the review notes the substantial advancements made by models such as Cycle GAN, Progressive Growing GAN, and StyleGAN in generating high-resolution, photorealistic images. These models have significantly raised the bar in terms of the realism of deepfakes, complicating the task of detection. The literature also discusses the inherent challenges in training GANs, including issues like mode collapse, non-convergence, and diminished gradients, which impact the reliability and accuracy of these models. To counter these challenges, various studies propose innovative architectures and training methods, such as ensemble learning techniques and generalized GAN architectures. These approaches aim to enhance the robustness of detection systems by improving their ability to generalize across diverse datasets. This thorough analysis emphasizes how important it is to carry out ongoing research and progress in detection methods to stay up with the quickly

advancing deepfake-generating technology. Table 1 below presents a systematic review of literature, that is composed that includes the summarized tabulated scrutiny of published articles on deepfake social media threats, GAN deepfake approaches, and deepfake detection techniques that are currently being used and considered effective to develop insights about the topic.

Table 1: Review of literature about the domain from (2018-2023)

Deepfake social media Threats				
Author	Year	Research Objective(s)	Social media Threats	Conclusion/Possible Solutions
UNESCO [64]	2018	Impacts of misinformation spread through social messaging and social media	Loopholes of Privacy Policies and Copyright terms of individual online content publication networking platforms	Need to employ an authenticated legal framework as per UN standards that both states and online content-sharing companies should study and use to ensure safety while utilizing the rights to share, access and obtain information
Westerlund [65]	2019	threats and benefits of deepfakes technology; Types of Deepfakes	Responses collected from 84 Online News portals open for public access	Deepfakes arise significant threats to the political system, society, legal framework, and businesses. To control these threats, appropriate legislation and regulation should be enacted. Deepfakes can also drive the reform of legal regulations and stimulate education, training, and voluntary endeavors.
Botha and Pieterse [66]	2020	Rising social media False News Problems	Overview on the latest deep learning-based tools, such as GANS that are being used	The research motivates on the deepfake protection systems that according to the authors should be sufficiently equipped in identification and control of the illegal and security harming practice

			to generate false news in social media. The study indicates involvement of influential people in this practice for the sake of personal or collective motive driven gains	
Alattas and Bayoumi [67]	2020	Use of Artificial Intelligence in Deepfake applications	Generation of viral false news and identity threats affect large mass of people over the globe who are connected through social media raising security as well as cultural, democratic and industrial damage	The study encourages strong legal policies and awareness initiatives on deepfake corruptions to be enforced in governmental level. According to the authors, deepfake corruptions are factors to risk even national security
Jaiman [68]	2020	Ethical impacts of deepfakes	Argues that the rapid generation of AI generated deepfakes and false hate speeches increase distrust and promote factual relativism, boost for 'liar's	The article stresses on ethical regulation to be implemented in AI applications and legal control on online media sharing

			dividend' and wrongly impact national democracy and social stability	
Meskys et al. [69]	2020	Regulation frameworks of deepfakes	Discusses on the rising threats out of the open availability of deepfake generations apps in online platforms	The article throws insight on the judicial legal framework to detect harmful deepfakes and complement the positive side of image manipulation done for creative stimulation. Aso, the article boosts for public awareness on deepfakes and their impacts
Hodge et al. [70]	2021	Threat of Shallow fakes and Deepfakes in terms of misinformation and false news	The paper discusses on the lack of legal capacities to deal with the shortfalls that occur out of failure in identifying deepfakes and real media while mitigating for identify risk issues	The article recommends for deeper reading of legal theories on information sharing and publication to mitigate deepfake forgeries
Ebermann et al. [71]	2021	Implicationsof Synthetic Media andDeepfakes onCommunication Professionals	Based on the skills and awareness of Communicati on Professionals, namely, company CEO, public speaker, political personality, etc., deepfake	Periodic moderation on Synthetic Media data, evaluation of authenticity and suitable control policies are necessary to minimize the problems

			news can increase the possibility of distrust, loss of reputation, damage of public image, etc.	
Yadlin-Segal and Oppenheim [72]	2021	Global impacts of social media deepfake malpractices	Discussions done on deepfake procedures, namely, Polarization, exploitation, manipulation, and disinformation in the media ecosystem	This article illustrates more comprehensive theoretical and practical understandings of AI content control, ethics, and responsibility in digital culture.
Gregory et al. [73]	2023	Deepfake misinformation threat	Provides a discussion on human rights and safety risks caused due to deepfake information disorders spread in online media and news	The paper argues that extensive research on deepfake applications coupled with the legal frameworks that stop or permits deepfakes through available loopholes should be analysed simultaneously to mitigate the problem
Etienne et al. [74]	2021	Public trust on online content sharing platforms due to deepfake corruption	Discusses on public awareness towards the positive and negative aspects of AI that are currently used in online content sharing platforms and	Apart from public conflicts and damage of trust through proliferation of deepfake in online content sharing platforms, the article hints on the emergence of modern applications of technologies that are supportive for social rationality, usages and expansion of taking decision in digital ages

Van der Helm et al. [75]	2021	The problem of indistinguishability of AI generated deepfakes	maintain trust on the modern technological advances Discusses on the multi-dimensional forgeries that are generated by deepfakes, like, image manipulation, false video, identity theft, privacy damaging pornography corruption, etc.	The article prompts for legal improvements to deal such multi-segmented problems of deepfake forgery.
Karasavva and Noorbhai et al. [76]	2021	Deepfake pornography threat	Discusses on online generation of deepfake pornography with our legal permit of consent that poses social risks and damage of reputation	The article suggests for language specific laws applicable in zones with higher rate of criminal charges against deepfake pornographies and vigilances on the websites that publish such content
Feeney [77]	2021	Problems with laws against deepfakes	The generation of media in online platforms are multi-segmented and impact in various levels, from individual to national impacts	Seeks for specific legal scanning system on deepfake identification
Fernandez	2022	Legal Frameworks on	Discusses on	The article puts special

[78]		Deepfake Mitigation	digital transformation and open distribution of deepfakes online by using new intelligent technologies	significances on legal control and improvements of privacy and publication terms that can minimize objectionable publication of deepfake contents
Rancourt-Raymond and Smaili [79]	2022	Study on Unethical Uses of Deepfakes	Provides overview of deepfake technology, gathers answers to a several inquiries are necessary to more fully understand how these technologies affect businesses, such as the definition of deepfake. Who is the intended victim and who is the scammer? How and why should I utilize them? What comes next?	The study shows evidence that deepfake frauds can be threats to any organization. The study particularly stressed on email frauds other than identity forgeries and suggests for awareness and control measures
Hameleers et al. [80]	2022	Study on implicationsResearch on the effects of Vox Populi, or political deepfakes, on social media		Disinformation based deepfakes creates severe societal impacts added with democratic disruptions and consequently policy modifications and need of further mitigation systems. Deepfakes are not credible as

Pawelec et al. [81]	2022	Deepfakes and false hate speech and their impacts on democracy	Publication of deepfakes and false hate speech online under the advantages of loose legal system are in turn damaging political environment of a nation that may fuel disrespect and defiance	per the research. The article emphasizes of stronger vigilance and government initiatives to mitigate these problems that are ultimately degrading national political image of a country
Appel and Prietzel [82]	2022	Impacts of Political Deepfakes	Discusses on identified cases of political deepfakes leakages of political videos with manipulation online that significantly impacts of public expressions and national image	The article points the advances of deepfake identifications as better initiatives towards eradication this problem. On the other hand, the accuracy of deepfakes, according to the author is a risk in terms of authenticating deepfakes
Lussier [83]	2022	Privacy issues in terms of deepfake content publication and open availabilities	Discusses on the nonconsensual deepfake pornography that is damaging reputation and individual privacy	The article emphasizes on legal improvements on digital content creation and their online publication to minimize the problem
O'Connell [84]	2022	Issues of illegal online publication of intimate images online without	Discusses on government intervention	The article stresses on vigilances, extensive scanning and regulatory control to

		the consent of the victim	to stop online publication of deepfake generated intimate images without getting the consent of the victim	minimize the problem
Akhtar [85]	2023	Study on Deepfakes and Detection Methods	Identity face re-enactment,swapping, attribute manipulation, and entire facial rejuvenation are the four types of deepfake or face manipulations that are examined.	Even though there are advances in traditional and advanced computer vision deepfake detection techniques, there are challenging areas still exist in identification of real and fake media contents that need further improvements
Mustak et al. [86]	2023	Deepfake impacts, threats and current mitigation policy status	Studied the significances of deepfakes for firms and consumers, like, (a) harm to one's credibility, image, and reputation as well as the quick obsolescence of current technology (for industries); and (b)	The study provides factual evidence of positive uses of deepfakes. Also, the authors discuss on the current legal system active to control deepfake corruptions and suggested possible future developments

			revenge porn, identity theft, intimidation, blackmail, harassment, slander, and harassment.	
Pandey et al. [87]	2022	Current Deepfake threats identified and legal system flaws in India's context	The study acknowledges deepfake threats as global issues affecting users of social media	The study suggested reforms in IT laws in India and encouraged active global collaboration among countries to combat against deepfake problems effectively
Okolie et al. [88]	2023	Image-based sexual abuses and data corruption	Discusses on deepfake pornography and legal implementations to control the same	The article recommends for specific and stricter legal measures that should be urgently implemented to control and eradicate the problems of online sexual abuses

GAN based Deepfake Approaches

Odena et al. [89]	2017	Conditional Image Synthesis with Auxiliary Classifier GANs	Ensemble of 100 AC-GANs	MS-SSIM score	Identified the existing flaws of the model, such as, visual discriminability, inception accuracy, etc. that impact on model training and need to be addressed. The study encourages semi-supervised learning scheme
Shen et al. [90]	2018	GAN based Deepfakes	Cycle-GAN	Through perceptual studies, which is to let people distinguish real and	GAN can be used in different industrial sections, such as,

				fake images given a pair of input and output	image enhancements and replications. However, they need multiple data for training and the speed of training is slow
Parimala et al. [91]	2019	Study of Quality Aware GAN performance	WGAN-GP	A new quality-aware discriminator gradient penalty function that draws inspiration from the NIQE, a well-known picture rating of quality technique) and the Structural Similarity (SSIM) index	Regularizes GAN image generation performance.
Kancharla and Channappa [92]	2021	Quality Aware GAN Image Synthesis	WGAN-GP	Randomly generated SSIM based QAGAN and the NIQE based QAGAN images were evaluated with IS and FID schemes	The produced photos need to retain the distinct local structural and statistical characteristics of the original, unaltered natural photographs to improve their realistic impression
Maras & Alexandrou [93]	2019	Accuracy and Multi-dimensionality of GAN deepfakes	Stagewise GAN traditional variants	Standard evaluation metrics	The authors point the convenience and accuracy of GAN models used in generating deepfakes
Gu et al. [94]	2019	Mask-Guided Portrait Editing for deepfake generation	Mask-guided conditional GAN	FID, IS Helen Dataset used for deepfake region detection	GAN generated deepfakes are high in accuracy
Singh et al. [95]	2024	To generate GAN images of	StyleGAN2		By incorporating diff

		people with differential facial expressions to use to generate a deepfake			erentiability in facial expressions of synthetically generated training images using GAN.
Gong et al. [96]	2024	Study on application of GAN technologies across several domains to produce digital multidimensional information and their usability in identify deepfakes and benefits in deepfake detection	GANs centered around model application optimization; loss function optimization – From 100 published articles	Standard Image Assessment and Differentiation identification procedures	The study signifies relying on AI to detect fake contents. side-by-side, it poses questions for further investigations, such as, how to proficiently perform manual verification in technical testing; Should these technologies be somewhat open source before coming under government control? How can appropriate rules prevent the advancement of technology that is beneficial?
Santha et al. [97]	2020	Deepfakes Generation using LSTM based Generative Adversarial Networks	LSTM based GANs	Psychophysical study, where original and generated deepfake clips were submitted to Amazon mechanical Turk and the evaluation are done by humans.	The model is effective to produce images for deepfake detection
Guarnera et al. [98]	2020	App based image editing	ATTGAN, GDWCT,	Classifier based comparative	Precision based deepfake

		and translation	STARGAN, STYLEGAN and STYLEGAN2	evaluation in terms of Random square, size, rotation, scaling, gaussian blur, etc.	detection with special importance on organized image forensic dataset
Yang et al. [99]	2020	GAN based deepfake network architecture	ProGAN, InfoMaxGAN, MMDGAN, SNGAN	Multiple cross-test setups and a large-scale dataset	GAN model has varied efficiency of producing deepfakes
Paul et al. [100]	2021	Study on Deepfakes Generated by Generative Adversarial Networks	DCGAN	Accuracy test Done; Deepfake outcome analysed by Hypothesis test done using t-Test	Identified GAN failures in Deepfake production; highlighted GAN training problems, namely, vanishing gradients, mode collapse; the study recommends for the use of GAN deepfake analyses in human computer interaction as needed in deepfake detections
Cherenkov and Voynov et al. [101]	2021	Image-to-Image GAN based techniques	StyleGAN2	FID: Images compared using FFHQ, LSUN-Cars, LSUN-Horse and LSUN-Church datasets	GAN based smooth navigation tool is simple and effective
Gu et al. [102]	2022	Style-based 3d-aware generator for high-resolution image synthesis	StyleNeRF	Fréchet Inception Distance (FID) and Kernel Inception Distance (KID)	Several innovations, such as an improved-sampler and a new regularization loss, were

						proposed to lessen the discrepancies brought about by 2D up sampling. By using data-driven approach with minimal inductive bias, the study found that Smooth-Swap can achieve the best overall scores with fast convergence.
Kim et al.2022 [103]		Simple Enhancement for Face-Swapping with Smoothness	Smooth-Swap	Model performance compared with Face Shifter, MegaFS, HifiFace, SimSwap, and Neural Textures; Two earliest models: Deepfakes and Face swap. Quantitative evaluation done by using VGGFace2 and Arc Face embedders		
Deepfake Detection Techniques						
Xuan et al. [104]	2019	GAN image deepfake detection	DCGAN, WGAN-GP and PGGAN	Deep Learning with forced deep network capacity	Image Matching by using statistical processes; Confusion Matrix implemented	Deepfakes are evolving and accordingly detection schemes should align and locate the gaps
Nguyen et al. [105]	2019	Dynamic routing algorithm for deepfake detection	Traditional GANs	Convolution Capsules	Database match scheme	Deepfake detection needs
Matern et al. [106]	2019	Deepfake detection focused on Eye, teeth and facial tone values	Traditional GANs	Neural Networks	Feature Matching and Similarity Evaluation by using standard datasets	Based on the zone and feature selection scheme the detection model efficacy varies
Yang	2020	Defence	Transfor	Ensemble	Fast Fourier	Training a

et al. [107]		against GAN- based Deepfake attacks via Transfor mation- aware Adversari al Faces	mation- aware adversari al face generatio n using projected gradient descent (PGD)	transformation- aware adversarial face generation using PGD	Transform (FFT) comparison of loss functions	Deepfake model with adversarial perturbed face images can lead to a significant degradation in the characteristics of synthetic faces. Numerous measures make it simple to identify and visually recognize this decline. However, with advanced deepfake techniques, this defence method may need further improvements to enhance its efficacy. Diverse feature extractions need validation and availability of dataset and training schemes are needed
Güera et al. [108]	2018	Feature extraction based deepfake detection	Face translatio n GAN	Convolutional bidirectional recurrent LSTM network	Database match and performance score metrics	Deepfake evaluations and detections are dependent of feature identification and validation accuracy
Li et al. [109]	2020	Deepfake detection based on Face X- Ray	DC- GAN	CNN	Feature Matching and Correlation evaluation based on dataset	Detection accuracy is dependent on data availability and specificity of feature selection
Ciftci et al. [110]	2020	Synthetic Portrait Deepfake detection	Tradition al GANs	CNN	Database based resolution, compression, content, and context matching	

Hsu et al. [111]	2020	Image deepfake detection by using pairwise learning	DCGAN, WGAN, WGAN-GP, least squares GAN, PGGAN	CNN concatenated to CFFN	Feature Matching by using open-source standard database	Feature extraction and learning depends on specificity and training model selection
Malolan et al. [112]	2020	Deepfake detection using visual interpretability methods	Standard GANs	Deep Learning using CNN	Evaluation Metrics	In terms of diverse deepfake detection aspects, deepfake prediction efficacy of the model should be evolved
Shahed Shad et al. [113]	2021	Detection of deepfake images from real ones accurately	StyleGAN	Eight distinct CNN models were employed: three with DenseNet (121, 169, 201), two with VGGNet (16, 19), one with ResNet50, one with VGGFace, and one custom CNN.	Image Matching by using statistical processes; Confusion Matrix implemented	VGGFace outperformed the custom model used in the study in terms of recall, although VGGFace still performed best overall in a number of criteria. Impressive outcomes were also obtained using the bespoke models like ResNet50 DenseNet169 etc, Ultimately, gathered deepfake photos were examined to see if they were authentic or not, and the outcome was satisfactory.
Zhang et al. [114]	2022	Deep forgery discriminator via	Face swap adversarial	Novel discriminator-based framework	Comparative analysis	When compared to the original content, synthesized face

		image degradati on analysis	networks			photos do exhibit varying degrees of quality loss. To expose manipulated information and alteration patterns in the fake faces, a number of image quality-related handcraft characteristics, including deep features, their sharpness, texture, frequency domain features be extracted.
Juefei- Xu et al. [115]	2022	comprehe nsive overview and detailed analysis of the research work on the topic of the creation, identifica tion, and avoidanc e of deepfake informati on	GAN models as per their develop ment phases achieved	Available and Commonly applicable Deep Learning Models (DNNs), Classifiers, Ensembles and Artifact based systems	PSNR, SSIM, IS and FID (for generated image quality evaluation)	The study identified the existing lagging areas between deepfake generation systems and deepfake detection models
Hao et al. [116]	2022	Deepfake Detection Using Multiple Data Modalitie s	Face Swap Deepfak e schemes	Convolutional and Recurrent Neural Networks.	comparisons quantitatively between Xception, Conv-LSTM, Efficient Net, an upgraded	To ensure societal, organizational and government security, Deepfake detection and

					Conv-LSTM with input from the face areas identified with the integration of MTCNN	control is necessary and should be implemented robustly
Hooda et al. [117]	2024	Disjoint Deepfake Detection (D4) based on an ensemble-based technique	GAN based deepfakes	Ensembles of Shallow four-layer CNN binary classifier employed by the baseline Frank et al. (2020)	AUC-ROC score.	Study recommends for suitable deepfake mitigation policies to remediate the societal and individual security threats

CHAPTER- 4

A Novel Generalized Detection Approach for Image Forgeries Using Generative Adversarial Network

Aim

This chapter aims to enhance the detection of deepfake facial images by developing an advanced Generative Adversarial Network (GAN) model that addresses the common issues of mode collapse and convergence, thereby improving the overall accuracy and reliability of detection systems. It presents a novel methodology for deepfake detection using an ensemble-based Generative Adversarial Network model. The objectives include describing the proposed model, explaining the underlying techniques, and presenting preliminary results to optimize the reliability of deepfake detection systems.

Objectives

- **To introduce a novel generalized ensemble model based on Generative Adversarial Networks:** Develop a robust model that effectively addresses training challenges, including Mode Collapse and Convergence issues.
- **To implement a modified diversity loss function:** Employ a specialized loss function designed to minimize discrepancies between the generated and original distributions, enhancing the quality of synthetic data.
- **To utilize a voting ensemble approach with multiple discriminators:** Combine the strengths of multiple CNN discriminators to improve the model's decision-making accuracy using a max-voting technique.
- **To evaluate and compare the model's performance:** Assess the proposed model against existing GAN models using metrics such as Total Computational Time, Fréchet Inception Distance (FID), Inception Score (IS), and Structural Similarity Index Metric (SSIM).

- **To optimize the stability and improve picture quality of the GAN model:** Address mode collapse and convergence issues to ensure consistent and high-quality image generation.
- **To develop an optimal GAN model for effective deepfake detection:** Ensure the model is robust and reliable even with limited data availability, making it suitable for real-world applications in digital forensics and media integrity.

4.1 Introduction

Deepfake technology involves the replacement of one person's face with another's in images or videos, creating highly realistic but fabricated media. This is achieved using various advanced techniques. The manipulation tools include erasing, blending, multiplying, enhancing, and cropping [118]. Deepfakes can be created with remarkable ease using freely available tools online. They pose a significant threat as they can mislead viewers into believing fabricated content is real, which can have serious implications in various sectors, including media, politics, and personal privacy. A notable example of deepfake misuse includes the creation of pornographic videos. This not only severely damages the individual's reputation and societal image but also leads to significant emotional distress and privacy violations for the victims.

Detecting deepfakes is crucial for digital image forensics, particularly in the context of criminal prosecutions where digital images and videos often serve as critical evidence. The ability to manipulate photographs and videos using modern computer graphics and image processing technologies makes it easy to create convincing forgeries. These manipulations can range from simple edits like erasing or cropping to more sophisticated techniques like blending and enhancing. As a result, identifying whether a digital image or video has been altered is essential to ensure the integrity of evidence. Moreover, the rise of deepfakes brings up ethical and legal questions about the manipulation of digital media, necessitating robust detection methods to maintain trust in digital content. One

definition given by Merriam-Webster is "falsely and fraudulently changing a digital image," a concept with roots in the 1840s. The ability to replicate pictures with varying parameter settings has only become more sophisticated with modern technology. The global rise in serious instances of picture forgery has deeply concerned law enforcement agencies. The availability of many tools for editing, improving, correcting, modifying, and recreating images encourages illicit activity[119].

Generative Adversarial Networks play a significant role in both the generation and detection of deepfakes. A GAN consists of two neural networks, a generator, and a discriminator. The generator creates synthetic data, while the discriminator checks the authenticity of this data. Through an adversarial process, the discriminator learns to distinguish between real- and fake data, and the generator component improves its output to create more convincing fakes by classification between real-fake [120]. This procedure not only makes GANs effective at producing realistic images but also at enhancing the training of detection models by providing diverse and high-quality synthetic datasets. These datasets help in training robust models that can identify subtle discrepancies between real - and fake images.

To enhance the detection of deepfakes, the present research proposes a novel GAN ensemble model that utilizes multiple discriminator networks with a max-voting technique combined with a single generator. This approach addresses common challenges in GAN training, such as Mode Collapse and Convergence issues, by leveraging the strengths of an ensemble of discriminators. The max-voting technique ensures that the final decision on the authenticity of an image is more reliable, as it is based on the consensus of multiple discriminators. This model employs a tweaked diversified loss function to reduce the discrepancy between the produced and source data distributions, resulting in a highly accurate and reliable detection method. With an impressive 98.31% accuracy rate, it has shown to be a reliable method for detecting forgeries. On an insignificant dataset of "Face Mask Lite", the new approach was shown to outperform earlier GAN

models using Fréchet Inception distance, Total Computational Time Function, Inception score (IS), SSIM (Structural Similarity Index Metric), Inception score (IS) [121].

4.2 Related Research Work

We examine three distinct GAN architectures: Pix2pix, C-GAN, and DC-GAN to offer context for the GAN model's design and details.

4.2.1 Pix2Pix

A particular kind of Generative Adversarial Network called Pix2Pix GAN is intended for image-to-image translation applications. Unlike traditional GANs that generate entirely new images from random noise, the pix2pix GAN takes an input image and transforms it into a target image using a conditional adversary network. That example removes a generator of an output image that is in the same state as the input image. The framework is widely used in various- It is done in applications like manipulating sketches, increasing image resolution, and even translating aerial photos to maps. Its ability to recognize and translate visual information from one location to another with high fidelity makes it a powerful tool in the field for computer vision and image processing. The outcome of Pix2pix is heavily dependent on the information given (source image) by the user. Generator and Discriminator are neural network elements. Transformation is the game's name, which the generator uses to create this image. The discriminator module or classifier ultimate primary aim will be to decide whether the generator module generated an unfamiliar image through assessment. The correspondences in the input image and an unseen image can either be a reference image from a dataset or a generator module unknown image. Cut down the adversarial-predicted value of loss for the generated images by tuning the generator. To remedy the occurrence of overfitting whereby the Generator was not illuminated by the training data with intense light. On the contrary, in both training and prediction the cross-entropy loss function and the dropout level step in to handle the problem of disappearing neurons also.[122]

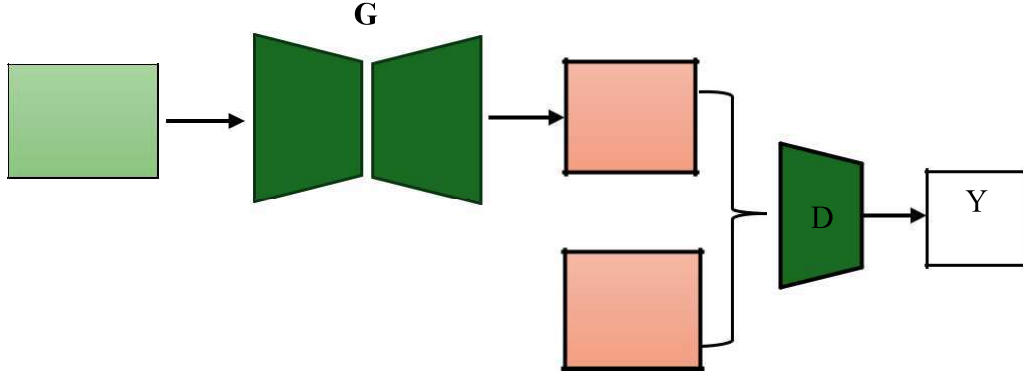


Figure 4.1: Architecture of the Pix2Pix Model [122]

In this study, **Christovam *et al.*, [2022]** proposed that the objective function incorporate the Multi-layer Perceptron loss function. The main objective was to provide the model with the ability to generate artificial pixels that had a value for each and function as a spectral response, which consequently let it increase the precision of the crop mapping. Additionally, it has been confirmed that the generative models can provide natural pixel values even for those images that have not been trained. In directive to measure the optimality of the proposed methods, an evaluation of the generator module images and the existing ones. The application is assessed using pixel value analysis, similarity metrics, visual analysis, and semantic segmentation. On average, two images of better quality were output from pix2pix cGAN compared to the first technique. It was more effective in terms of better noise reduction, visual quality upgrade, and crop type semantic division than the pix2pix algorithm classic version. With the addition of the proposed system, the synthesized image got an F1 score (44.2%) while the actual image achieved 44.7%. Granted the higher resolution for differentiation, the models produced better results with pixel results of superb quality than the image which was divided into two and made two separate images. Moreover, it was proven that models, which were trained with quarterly image pairs to embody the cyclone season have good results and, consequently, cloud-free areas are also not regarded.[123].

In this research, **Patil, Chaudhari, and Narawade, [2022]** proposed a method for remotely recognizing the disrupted area of a sensor image. The researchers are using the RSICD (Remote Sensing Image Capturing Dataset) which includes imagery from Google Maps, Google Earth, and Bing Maps. The synthetic dataset was created by manipulating the remote sensing photos with techniques such as copy-move and splicing forgery. The use of the two varieties of U-net was done to compare them and get the highest level of accuracy in the results. Initially, they utilized U-net cards for detecting the altered region. They have used a modified U-net architecture, adjusting the image size, and number of epochs to improve the identification of forgeries. Several criteria, such as F1 score, RMSE, Jaccard Index, Accuracy, and Peak SNR, are utilized when comparing the U-net version to the regular U-net model. It is evident from the data that both versions are significantly outperforming the present techniques [124].

4.2.2 Cycle GAN

Cycle GAN is a kind of Generative Adversarial Network designed for non-paired image-to-interpretation tasks, where there are no directly matching image pairs for training. Unlike traditional GANs that require paired datasets, Cycle GAN restores translated images from one domain to another. The framework of these two GANs consists of two generators and two discriminators, where each generator learns to translate images from one domain to another while each discriminator discriminates real images and generated in their fields. Turns out, this innovative approach allows flexible and effective image transformation on various unpaired image datasets without the need for integrated training samples, enabling applications such as style transfer, object transformation, and photo enhancement. If the two domains do not align pairwise, The Cycle-GAN algorithm is utilized to determine the correspondence between the image sets X and Y belonging to different domains. Both the forward and backward generators are represented by the letters G and F , respectively. The consistency of the cycle ensures that $(G(F(x))) \approx x$ and $(F(G(y))) \approx y$. The representations of the original

pictures can be learned in a manner like an auto-encoder using $\circ$ or $\circ$. Yi et al. also offered a comparable approach. Images sourced from many locations must contain common visual elements, for example, zebras and wild horses. When two domains are very different, normal image translation might be significantly impacted [125].

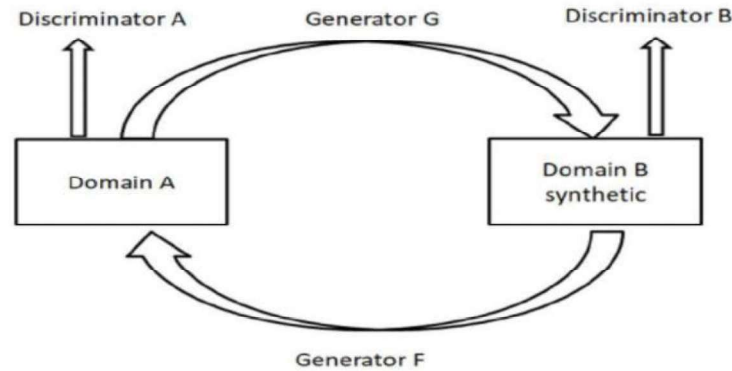


Figure 4.2:The structure of Cycle-GAN [4]

In this study **Fahn and Wu, [2022]** proposed that a DL-based technique for detecting faked images is introduced. Initially, discrete Fourier transform is performed on both genuine photographs and fake images that are generated using generative adversarial networks. Next, the Fourier spectrums that have been acquired are inputted into deep neural networks to train the model. For better quality of detection, they also introduce contrastive learning to the model that will allow the model to directly learn the distinguishing characteristics of both real and fake photographs. selected the following generative adversarial networks (GANs): Cycle GAN, Mixed GAN, DCGAN, And Auto GAN will provide fake images, for checking our suggested methodology. The experimental outcomes display that the introduced strategy reached a mean accuracy of 99.5% in discerning the four sort of GAN-generated pictures. The new method that we plan to adopt offers wider options in terms of detecting counterfeit images from different sources, unlike the current counter-forgery image detection method.[126].

In this research **Bi, Zhang and Xiao, [2021]** suggested that the process of retouching photoshopped pictures means lessening the traces of manipulation and

at the same time reading intact information. This editing technique is termed as the transformation of image style and a generator known as GT is presented for the change of fake to realistic images. A localizer generator GM is also indicated to pick the altered areas. This framework incorporates the use of a multi-decoder-single-task approach. The suggested α -learnable WCT block in GT stands for α -learnable whitening and colouring change can significantly reduce the artifacts in the forged photos by training two generators in an adversarial way. And by learning from the manipulated fraudulent pictures, GM will improve its identification and location skills. The findings of the experiments show that the two elements of the GAN, namely ones that imitate the correct imitation of the real and the artificial, can successfully interact together. Moreover, our splicing forgeries localization generator GM improved the results of previously proposed models and is now as close has any of them to perfect in the task of recognizing splicing manipulations across four publicly accessible datasets. [127].

4.2.3 DC-GAN

Deep convolutional generative adversarial network (DCGAN) is an improved GAN that uses deep convolutional networks to enhance the generation of high-quality synthetic images DCGAN architecture uses convolutional layers of both generator and discrimination networks, like those found in traditional GANs that are fully connected. This ultimate approach enables DCGANs to capture and render fine details in images through spatial hierarchies and inherent local dependencies. In DCGAN the generator uses transposed convolution to upscale random noise up to a realistic image, while the discriminator uses standard convolution to distinguish between real-world images. A proven success of the spectacular creation, making them popular for unsupervised image fusion, super-resolution, and feature learning tasks. Their ability to produce more realistic images has greatly enhanced the generative models. In contrast to vanilla GANs, this design employs deep convolutional networks with transposed layers (convolutional) as both the generator component and discriminator component,

making it capable to handle more complicated tasks. The original DCGAN study employed the following design, as presented in Figure 4.3.

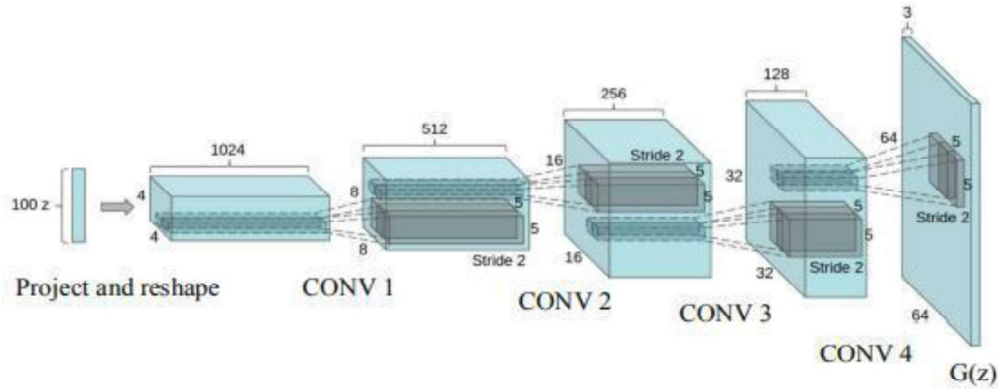


Figure 4.3: DCGAN generator architecture [128]

Dynamic convolutional neural networks (DCNN) use a noise vector z as its input and output, respectively. Novel visuals are produced by transposed convolutional layers. The design doesn't have any pooling or completely linked layers. Stridden convolutions and fractional-stride convolutions (generator) have supplanted pooling layers like max-pooling in the design.

In this study, **Mareen *et al.*, [2024]** aimed to enhance the ability to identify and locate forged items by integrating and utilizing multiple detection approaches that are mutually beneficial. Initially, the analysts examine the complementarity of these procedures by objectively measuring their degree of complementariness. They next calculate a goal performance value utilizing a theoretical oracle fusion. Next, they present an innovative fusion technique that integrates the outputs of the existing approaches. The fusion approach under consideration is taught with a Generative Adversarial Network architecture. The results of our trials show enhanced accuracy in detecting and pinpointing objects across different datasets. Despite the limitation of our fusion method in terms of generalization, this problem is frequently encountered in supervised learning, which serves as a driving force for future research. Ultimately, this study enhances our comprehension of the synergistic nature of forgery detection techniques and how to synchronize them. Therefore, they enhance the

safeguarding against picture alterations and aid in combating misinformation [129].

In this research **Zhang *et al.* [2024]** proposed a two-stage cascade system is proposed to identify instances of small region forgeries in medical images using GAN-based techniques, such as CT-GAN. To ensure that the detector is unaffected by interference data in real locations, the detector network is trained to employ tiny sub-images during the local detection step. By utilizing depth-wise separable convolution and residual networks—which also aid in preventing overfitting—the detector enhances its capacity for recognizing manufactured areas through the attention mechanism. A heatmap has been generated to arrange the finding results of all sub-images inside a similar image. The Gray-level concurrence matrix (GLCM) is a more efficient method for extracting features from the heatmap during the global classification phase. Hyperplanes are utilized for classification in infinite-dimensional spaces because they can effectively manage the complex shape and size of overlapping regions. Our technology possesses the capability to accurately classify if a CT image has been altered and precisely identify the location of the changed region. Comprehensive research indicates that our technique surpasses the existing cutting-edge detecting technologies [130].

4.3 Proposed Methodology

4.3.1 Generative adversarial Network (GANs)

A basic GAN model is made up of two parts: the generator and the discriminator. As the name implies, the generator uses training data sets to create a false output of unseen data, which then tricks the discriminator into thinking is real. In addition, the discriminator served as a police officer, sorting out the produced data from the training data, determining if the generator's samples were fake or real, and then classifying them accordingly. This loop, however, keeps going until the generator succeeds in fooling the discriminator with false data. The objective function for the discriminator loss is defined as:

$$= - \frac{1}{n} \sum_{i=1}^n [\log D(x_i) + \log(1 - D(G(x_i)))] \dots\dots\dots (4.1)$$

The objective function for generating loss is defined as:

$$= - \frac{1}{n} \sum_{i=1}^n \log D(G(x_i)) \dots\dots\dots (4.2)$$

The goal of training G is to maximize $\log D$, hence many mini batches containing real and fake images are used. The training consists of two components. Part 1 provides a comprehensive understanding of the discriminator, while Part 2 offers a thorough understanding of the generator. To optimize the discriminator, it is necessary to maximize the expression " $\log D(x) + \log(1 - D(G(x)))$ ".

To accomplish this, the procedure commences by generating a collection of authentic samples derived from the training data. The loss, denoted as $\mathcal{L}_D$, may then be determined by doing a forward pass with discriminator loss. Finally, perform a reverse pass to obtain the gradients. To enhance the quality of counterfeit items, the generator should strive to decrease the logarithm of the discrepancy between 1 and the output of the generator network, while simultaneously maximizing the logarithm of the output of the discriminator network. Figure 4.4 shows the basic GAN model below.

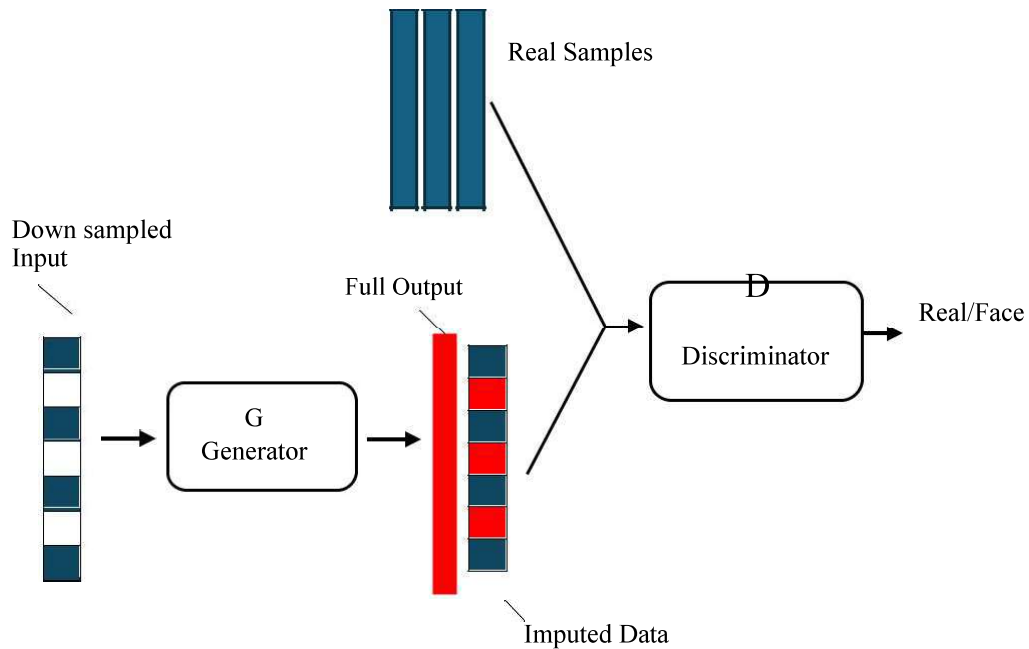


Figure 4.4: A visual representation of the Generative Adversarial Network (GAN)

4.3.2 Description of Proposed Ensemble Model

The proposed GAN model leverages the strengths of multiple discriminators to improve the accuracy and robustness of deepfake detection. Instead of relying on a single discriminator, which may not capture all the nuances of the data distribution, our model employs three distinct CNN-based discriminators. Each discriminator evaluates the generated images using a diversified loss function that combines the Inception Score, Fréchet Inception Distance, and conventional GAN loss. This multi-faceted evaluation ensures that each discriminator captures various aspects of image quality, such as realism and feature consistency. With each successive epoch, the value of loss of discriminators is computed using diverse loss functions defined as:

$$L = L_{\text{GAN}} + L_{\text{IS}} + L_{\text{FID}} \dots \dots \dots (4.3)$$

L_{GAN} stands for the BCE, and the hyperparameters α and β control each term's relative relevance in relation to the conventional GAN loss. The diverse loss function is a critical component of this ensemble approach shown in equation 4.3 above. It integrates three loss components: Binary Cross-Entropy (BCE) loss, which measures the probability, Inception Score (IS), which evaluates the clarity and diversity of generated images; and Fréchet Inception Distance (FID), which assesses the similarity between the distributions of real and generated images. By incorporating these diverse metrics, the loss function comprehensively evaluates each discriminator's performance. This approach ensures that the selected discriminator is not just the best at distinguishing real from fake images but also at promoting high-quality and varied outputs from the generator. The selected discriminator, having the minimum loss value, is then used to complete the GAN ensemble. This discriminator guides the generator's updates, ensuring that the generator produces high-quality images that are increasingly difficult to distinguish from real ones. The iterative process of selecting the best discriminator and updating the generator's weights continues until the generator achieves optimal performance. This ensemble approach not only enhances the stability of the GAN model but also ensures the generation of realistic and diverse images. By integrating the max voting mechanism and the diverse loss function, our model achieves superior performance in deepfake detection, making it a robust tool for applications in digital forensics, media integrity, and personal privacy protection.

4.3.3 Max voting Approach

Ensemble learning combines multiple models to improve overall performance. The key idea is to aggregate predictions from multiple discriminators to make a final decision. This approach helps in reducing variance, bias, and improving the model's generalization. The max voting ensemble approach selects the discriminator with the minimum diverse loss value at each epoch, ensuring the most accurate evaluation of generated images. This chosen discriminator updates the generator's weights, enhancing image quality and improving GAN training.

stability and convergence. At each epoch of the training process, the loss values for each discriminator module are calculated using the diverse or varied loss function. The discriminator module that exhibits the minimum loss value is considered the most effective for that epoch. This selection process is executed using a max voting approach, where the discriminator module with the minimum loss is chosen as the ideal representative for updating the generator's weights. This mechanism ensures that the generator receives consistent and reliable feedback, allowing it to improve progressively by learning from the most accurate and reliable discriminator at each step.

4.3.4 GAN Ensemble

In combination with a generator, the discriminator that shows the loss function's minimum value is selected as the optimum discriminator type and is used to modify the weights of the generator module finally. It is calculated utilizing a voting (max) approach that uses "The best discriminator function" to choose the minimum loss discriminator module type. The discriminator is constructed by utilizing the Adam optimizer in conjunction with binary cross-entropy loss. Once the ensemble model has been applied to a generator function, the generator is then explicitly defined. The usage of an optimizer (Adam) with BCE is employed to construct the GAN model. The structure of the discriminator is defined by the shape parameter, while the dimensionality of the generator's input is set by the latent dim parameter. The suggested ensemble model has capability to generate pictures that match the distribution of real samples, differ from real samples, and have resolutions like real images. Noise input is introduced to improve convergence and break the addressing mode. The proposed GAN Ensemble model Function is defined as follows:

Table 4.1: Proposed GAN Ensemble Model Function

```
# GAN Definition def GAN (shape, latent dim, best discriminator, generator
type):
# Configure the module for discrimination
```

```

Best discriminator(shape) = discriminator module
Discriminator_module.compile(optimizer_type=Adam
(learning_rate=0.0002, beta_1=0.5), loss="binary_crossentropy"))

# Describe the generator.
Generator type (latent dim) = generator module
# Combine the generator and discriminator into a Sequential
model Gan = Sequential ()
Gan.Add(generator)
Gan.Add(discriminator)

# Compile the GAN model
Gan.Compile(optimizer_type=Adam (learning_rate=0.0002,
beta_1=0.5), loss="binary_crossentropy"))
return

```

4.3.5 Detailed Architecture

Building the GAN function is the first step in defining the architecture of the model; subsequent iterations of the execution loop will define the switching of training of the generators and discriminators subsequently. We may train the discriminator on a mixture of real and fake pictures by employing a generator to produce a batch of counterfeit pictures. The discriminator is updated using artificial pictures produced by the GAN model once the generator has been trained with noise. A procedure is iterated for an initial number of epochs. The comprehensive architecture of the ensemble GAN model aiming to minimize the training pathologies, is illustrated in Figure 4.5.

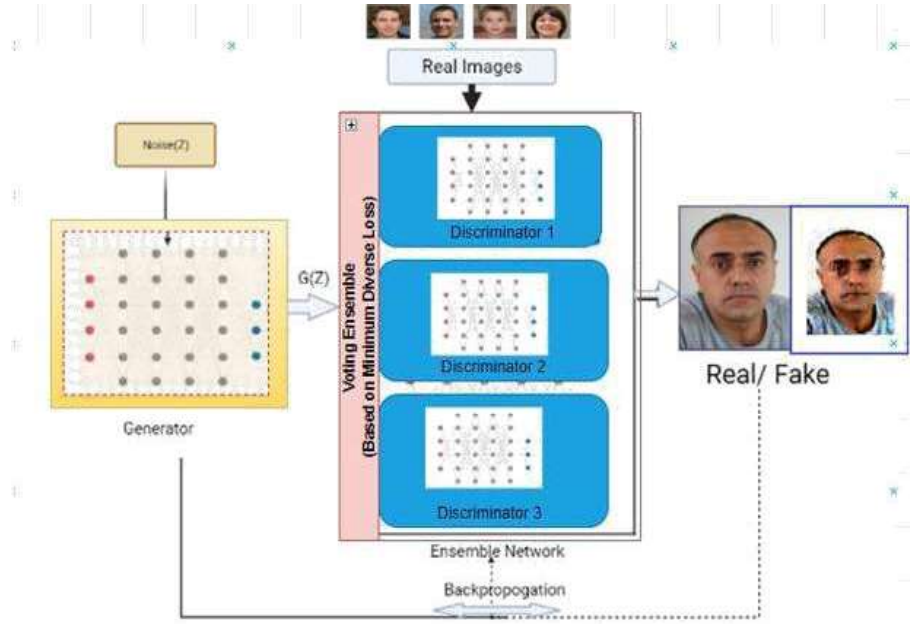


Figure 4.5:Enhanced GAN model utilizing Ensemble Architecture

For training, the enhanced GAN can generate images that exhibit an equivalent resolution to real ones, possess an identical distribution to authentic samples, and are distinct from real ones (i.e., there is no duplication). The synthesis network is designed such that each layer receives an instance of uncorrelated Gaussian noise, as shown in Figure 4.6. The criteria for evaluating generated images can be differentiated into the following categories: 1) assurance that no duplicates of real images are produced; 2) the generated images should maintain a consistent style, characterized by a distribution that closely mirrors that of authentic images. We provide a direct method for our generator to enhance convergence and mode collapse parameters by incorporating explicit noise inputs. Subsequently, the pre-feature scaling factors that were previously learned are utilized to merge noise images with their corresponding convolution outputs on all feature maps. Eight layers comprise each network, with one dedicated to processing, one to synthesis, and one to mapping. RGB is obtained by converting the output of the last layer using a unique 1×1 convolution. If R represents the set of real images, then R is equal to the set of imaginary images, denoted as " I_r ". The previously mentioned real image dataset was utilized for the purpose of

training of Generative Adversarial Network thereafter employed to produce a sequence of synthetic images. R is equivalent to the value of Ig. Grem comprises a generated image that remains after the removal of duplicates of authentic samples from G. The mode, expressed as the ratio of remaining photos to total in the creativity index (Grem to G), is calculated.

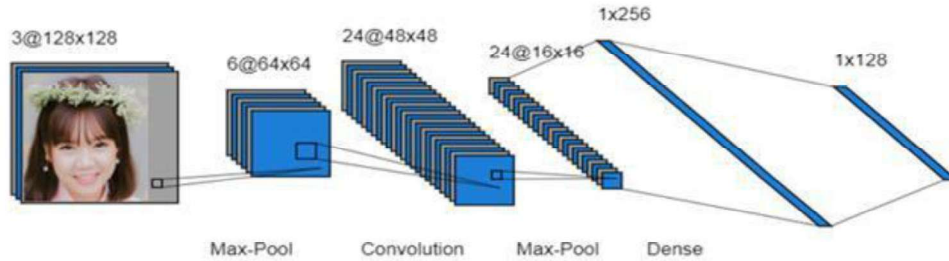


Figure 4.6:The proposed GAN model uses a Layered Architecture

$P(Y)$ defined as the discriminator's output defines the probability that the input images are members of a "true" class. A generator score is calculated by averaging a probability associated with an output of the discriminator for the images that are produced. $P(Y_{\text{Generated}})$ stores the probabilities associated with the generated images; therefore, the mean of $P(Y_{\text{Generated}})$ equals the score Generator. The likelihood that a picture is in the "generated" class ($1-P(Y)$) is replaced to compute the discriminator score, which is then determined by averaging the probabilities associated with input photographs that fall into the proper class. Equation 1 demonstrates the process of obtaining the discriminators score by taking the average of the probabilities of the correct class images inside the input image.

$$\frac{1}{2} (0 + \frac{1}{2}(1 - \dots)) = \dots (4.4)$$

The discriminator's actual image approximations are specified by the parameter $P(Y_{\text{Real}})$, given that the discriminator receives equal amounts of produced and real pictures. The cost function was determined by employing backpropagation to estimate the model parameters θ_D and θ_G of the discriminator D_c and generator G_n , which are denoted by the equation's third, fourth, and fifth terms, respectively.

$$J = -\log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) \quad (4.5)$$

$$= -\log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) \quad (4.6)$$

$$= -\log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) + \log P(Y_{\text{Real}}) \quad (4.7)$$

4.3.6 Dataset

A subset of the Face Mask Lite dataset¹, which we found on Kaggle (FDD), consisting of 193 unmasked images, was used in our study. Colored images with a resolution of 1024 by 1024 pixels initially made up this collection. 20,000 photos, divided 50/50 between produced and unmasked images, make up the FDD dataset. The 10,000 unmasked images served as a foundation for the 10,000 disguised images. Every picture has the same mask made. Every folder has 10,000 HD pictures created using the GAN-2 technique, both with and without a mask. This collection includes a 1024×1024 RGB picture. We investigated various batch sizes, epochs, and learning rate combinations to see which best suited our data. The dataset is accessible at the location and is freely available.

¹<https://www.kaggle.com/prasoonkottarathil/face-mask-lite-dataset>.



Figure 4.7: FDD dataset

4.3.6 Algorithm

Table 4.2: Algorithm for ensemble-based GAN model

Algorithm: Proposed ensemble-based GAN model

1. Initialize single generator and multiple discriminator networks, each with a distinct convolutional neural network architecture
 2. Import the set of training_data.
 3. Assign the hyperparameters by default:
 - a. A thousand iterations in total
 - b. Batch_size: eight
 - c. Kernel_size: four
 - d. Pool_size: (Three, Three)
 - e. Stride_length: Two per length
 4. For each epoch:
 - a. Make use of the generator to create a set of synthetic images.
 - b. Calculate the diverse function for calculating the loss value of each discriminator utilizing a validation set of images.
 - c. Use an ensemble method (voting), as shown by:
 $(_ = _ + _ + _)$, to choose the discriminator whose loss function value is the most consistent. The binary

-

cross-entropy loss in this case is denoted by L_{GAN} , and the hyperparameters α and β regulate how important each term is about the conventional GAN loss.

d. Using both genuine and false pictures for discriminator (referred to as D1) for a single iteration. Train the generator to produce one iteration using the binary cross-entropy loss function, using a batch of random noise as input and the selected discriminator as output.

5. Assess the performance of each GAN model using the FID and IS metrics at the end of each epoch.

6. Repeat step 4 for the specified number of epochs.

4.3.7 Mathematical Construct

Table 4.3:Mathematical Construct for ensemble-based GAN model

-
- Discriminators and Generator:
 - a. Our three discriminators, D1, D2, and D3, are based on CNNs.
 - b. One generator, G, is available.
-

- Loss Functions:

a. For discriminator D1, the GAN's loss function using Binary Cross-Entropy is:

$$L_{D1} = -\frac{1}{N} \sum_{i=1}^N (y_i \log p_i + (1-y_i) \log (1-p_i))$$

b. The discriminator loss (diverse) function that combines Binary Cross Entropy, IS, and Fréchet Inception Distance:

$$L_D = L_{D1} + \lambda_1 L_{IS} + \lambda_2 L_{FID}$$

- During the execution of each epoch:

a. Create fake samples using G.

b. Find the $L_D(x^{(1)}, x^{(2)}, x^{(3)})$.

c. Select the discriminator module with the smallest value of.

d. Training of the discriminator (step c) on a set of fake and real data using:

$$L_D(x^{(1)}, x^{(2)}, x^{(3)}) = -\frac{1}{N} \sum_{i=1}^N (y_i \log p_i + (1-y_i) \log (1-p_i))$$

For one iteration, incrementing or upgrading G using the output of the discriminator and loss using (step c & d).

a. Check for performance using metrics IS and FID.

- Optimizers: Train the generator and discriminators using the following parameters:

a. Rate of Learning(α): 0.0002.

b. β_1 : The initial instant estimate's exponential decay rate is usually 0.9.

c. β_2 : The second-moment estimate's exponential decay rate is usually 0.999.

d. ϵ : Avoid division by zero, a modest constant (usually 10^{-7}) is used.

e. t: A variable that keeps track of the execution step.

- Assign initial estimates:

a. At startup, the value of $m(\text{initial estimation})$ for every parameter set to 0.

b. Every parameter starts with a value of zero for second moment estimate v :

- For every iteration:

a. $\nabla \theta_j(\theta)$ is the gradient of the loss as it relates to the parameters.

b. Make a change to the first moment estimate m :

$$m \leftarrow \beta_1 m + (1 - \beta_1) \nabla \theta_j(\theta)$$

c. Update the second-moment estimate v :

$$v \leftarrow \beta_2 v + (1 - \beta_2) (\nabla \theta_j(\theta))^2$$

d. Correct the moment estimates for bias:

$$\hat{m} = m / (1 - \beta_1^t) \quad \hat{v} = v / (1 - \beta_2^t)$$

e. Update the parameters:

$$\theta \leftarrow \theta - \eta \hat{m}$$

4.3. 8 Training

With 128 images as batch size was used for training using mini-batch SGD. The pre-processing done to the scaling of training images was used to fit the tan activation function range of $[-1, 1]$. The standard deviation of the weight calculations was 0.02. The distribution used was normal. In every Leaky ReLU model, the leak slope was adjusted to 0.2. Layers for stride convolution and batch normalization make up the discriminator in this case, with Leaky ReLU serving as the activation function. The generator consists of batch normalization layers, ReLU activations, and convolutional-transpose layers. From 100 by 1 to 1024 by 4 by 4, the network grows! The 'Project and reshapes' label is applied to this layer. Table 2 details the functions of the several layers that were utilized. In Fig. 4.5, we can see the modified model's design.

4.3.9 Training algorithm

Table 4.4: Training Algorithm for ensemble-based GAN model

The number of training iterations and the hyperparameters α and β are modified to fit GAN training task as shown below:

Step 1: Do For total iterations

Step 2: Perform steps for "m".

From the noise before $pg(k)$, extract q noise samples (k_q) using a mini-batch sampling approach.

An instance of q -noise denoted by $\mathbf{r}_i$ is sampled using the distribution $p_{data}(\mathbf{r}_i)$.

Step 3: Increment the discriminator module's gradient (stochastic) to keep it up date.

$$\nabla_{\theta} D_c(\mathbf{r}_i) \leftarrow \frac{1}{q} \sum_{i=1}^q \nabla_{\theta} D_c(\mathbf{r}_i) + (1 - \frac{1}{q}) \nabla_{\theta} D_c(\mathbf{r}_{i-1})$$

Step 4: A batch of q samples $\{\mathbf{k}_i\}$ obtained from the noise samples collected from the previous iteration $pg(k)$ is used to make the selection.

Step 5: Reduce the stochastic gradient of the generator to keep it updated:

$$\nabla_{\theta} D_c(\mathbf{r}_i) \leftarrow \frac{1}{q} \sum_{i=1}^q \nabla_{\theta} D_c(\mathbf{r}_i) + (1 - \frac{1}{q}) \nabla_{\theta} D_c(\mathbf{r}_{i-1})$$

Endloop

4.3.10 Generator and discriminator training details

Several small batches containing both legitimate and fake images are used to train G , whose only purpose is to maximize $\log D_c(G_n(z))$. The training consists of two sections. Get familiar with the Discriminator in Part 1, and the Generator in Part 2. "For the discriminant, we need to maximize or amplify the expression $\log(D_c(x)) + \log(1 - D_c(G_n(z)))$ ". The loss is computed using a forward pass-through D_c , the gradients are acquired using a backward pass, and the procedure continues until the training set contains genuine or true data. In Goodfellow's discussion, where G_n is a generator network and D_c is a discriminator network, it is said that to produce better

fakes, the Generator should minimize the $\log (1 - D_c (G_n(z)))$ as much as feasible while maximizing the $\log (D_c (G_n(z)))$. The setup of the generator and discriminators 1, 2 and 3 utilized for the suggested architecture are shown in Figures 4.8, 4.9, 4.10 and 4.11, respectively. The role of various layers used by the generator and discriminator are discussed in Table 4.5 below.

Table 4.5: Roles of the different Generator and Discriminator layers in an ensemble-based GAN model.

Discriminator Dense	This layer is known as the noise layer of the generator.
Conv2DTranspose:	The image is both upscaled and convolved by this layer. Using UpSampling2D and then Conv2D is a go analogy.
Leaky ReLU:	When compared to the ReLU functions, it works better since it does not use gradient descent.
Batch Normalization	The output of the convolution is normalized. The model's accuracy and convergence are both improved by this.
Reshape	A one-dimensional vector can be transformed into a three-dimensional array using its capabilities.

- **Generator configuration**

Model: "model"		
Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 100)]	0
dense (Dense)	(None, 32768)	3309568
leaky_re_lu (LeakyReLU)	(None, 32768)	0
reshape (Reshape)	(None, 16, 16, 128)	0
conv2d (Conv2D)	(None, 16, 16, 256)	819456
leaky_re_lu_1 (LeakyReLU)	(None, 16, 16, 256)	0
conv2d_transpose (Conv2DTranspose)	(None, 32, 32, 256)	1048832
leaky_re_lu_2 (LeakyReLU)	(None, 32, 32, 256)	0
conv2d_transpose_1 (Conv2DTranspose)	(None, 64, 64, 256)	1048832
leaky_re_lu_3 (LeakyReLU)	(None, 64, 64, 256)	0
conv2d_transpose_2 (Conv2DTranspose)	(None, 128, 128, 256)	1048832
leaky_re_lu_4 (LeakyReLU)	(None, 128, 128, 256)	0
conv2d_1 (Conv2D)	(None, 128, 128, 512)	2277312
leaky_re_lu_5 (LeakyReLU)	(None, 128, 128, 512)	0
conv2d_2 (Conv2D)	(None, 128, 128, 512)	6554112
leaky_re_lu_6 (LeakyReLU)	(None, 128, 128, 512)	0
conv2d_3 (Conv2D)	(None, 128, 128, 3)	75267
Total params: 17,182,211		
Trainable params: 17,182,211		
Non-trainable params: 0		
None		

Figure 4.8: Design of the generator for the ensemble GAN model

- **Discriminator 1 configuration**

Model: "sequential1"		
Layer (type)	Output Shape	Param #
conv2d_4 (Conv2D)	(None, 128, 128, 64)	1792
max_pooling2d (MaxPooling2D)	(None, 42, 42, 64)	0
conv2d_5 (Conv2D)	(None, 40, 40, 32)	18464
max_pooling2d_1 (MaxPooling2D)	(None, 13, 13, 32)	0
conv2d_6 (Conv2D)	(None, 11, 11, 32)	9248
max_pooling2d_2 (MaxPooling2D)	(None, 3, 3, 32)	0
flatten (Flatten)	(None, 288)	0
dense_1 (Dense)	(None, 64)	18496
dropout (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 1)	65
Total params: 48,065		
Trainable params: 48,065		
Non-trainable params: 0		
None		

Figure 4.9: Configuration of the first discriminator in the ensemble GAN model

- **Discriminator 2 configuration**

Model: "sequential_3"		
Layer (type)	Output Shape	Param #
batch_normalization (Batch Normalization)	(None, 128, 128, 3)	12
conv2d_7 (Conv2D)	(None, 128, 128, 6)	168
max_pooling2d_3 (MaxPooling2D)	(None, 42, 42, 6)	0
batch_normalization_1 (Batch Normalization)	(None, 42, 42, 6)	24
conv2d_8 (Conv2D)	(None, 42, 42, 32)	1760
max_pooling2d_4 (MaxPooling2D)	(None, 14, 14, 32)	0
batch_normalization_2 (Batch Normalization)	(None, 14, 14, 32)	128
dropout_1 (Dropout)	(None, 14, 14, 32)	0
conv2d_9 (Conv2D)	(None, 14, 14, 64)	18496
max_pooling2d_5 (MaxPooling2D)	(None, 4, 4, 64)	0
flatten_2 (Flatten)	(None, 1024)	0
dense_11 (Dense)	(None, 256)	262400
dense_12 (Dense)	(None, 128)	32896
dropout_2 (Dropout)	(None, 128)	0
dense_13 (Dense)	(None, 1)	129
Total params: 316,013		
Trainable params: 315,931		
Non-trainable params: 82		
None		

Figure 4.10: Ensemble GAN model's Discriminator 2 configuration

- **Discriminator 3 configuration See Figure 4.11**

Model: "sequential"		
Layer (type)	Output Shape	Param #
flatten_1 (Flatten)	(None, 12288)	0
dense (Dense)	(None, 512)	6291968
dense_1 (Dense)	(None, 256)	131328
dense_2 (Dense)	(None, 128)	32896
dense_3 (Dense)	(None, 64)	8256
dense_4 (Dense)	(None, 32)	2080
dense_5 (Dense)	(None, 1)	33
Total params: 6,466,561		
Trainable params: 6,466,561		
Non-trainable params: 0		
None		

Figure 4.11: Discriminator 3 configuration of proposed ensemble GAN model

4.3.11 Evaluation Parameters

The establishment of a reliable ML model relies heavily on performance evaluation. To make sure the prediction model works effectively with both known and unknown data, it must be evaluated. Assessing how well a model performs when applied to new, non-sample data is what performance evaluations are all about.

- **Accuracy:** Accuracy, which assesses the proportion of instances in the dataset that were correctly classified, is among the most fundamental performance metrics.
- **Inception score (IS):** Measures the quality and diversity of generated images.
- **Fréchet Inception Distance (FID):** Assesses the similarity between real and synthetic image distributions.
- **Structural Similarity Index Metric (SSIM):** Evaluates the perceived quality of generated images by comparing structural information to real images.

- **Total Computational Time Function:** Tracks the time taken for the model to complete its training and generation tasks, ensuring efficiency in the GAN's execution.

These metrics provide a detailed analysis of the generated data, ensuring the model will perform well in in-depth lie detection.

4.4 Experimentation

The necessary software and hardware for this project are listed below:

4.4.1 Hardware Requirements

Several modern operating systems were used for the study, such as:

- **Computer:** Training deep learning models efficiently requires a powerful computer, whether it's a workstation or server.
- **CPU:** A state-of-the-art multi-core CPU (such as an Intel Core i7 or later) to manage the processing demands.
- **GPU:** A robust graphics processing unit (GPU) that supports CUDA, such as the NVIDIA GeForce GTX or RTX series, to speed up DL model training and inference.
- **RAM:** Processor memory sufficient to handle massive datasets and deep learning models when training (16 GB or more is recommended).
- **Storage:** There should be enough room to save the dataset, model checkpoints, and interim outcomes.
- **Operating System:** The necessary software and libraries may be utilized on any major operating system, including Windows, macOS, and Linux.

4.4.2 Software Technologies Requirements

The simulations were executed utilizing the Python programming language, the Jupyter notebook, and the Google Collaboratory platform to guarantee accessibility and computing capabilities. Important libraries, such as Keras with the tensor flow library, were installed to make it easier to create machine learning methods.

4.4.2.1 Python Tool

Python is a practical programming language with a solid design. Python is another interpreter-based general-purpose programming language with a high level of dynamicity, object orientation, and a wide range of potential uses. The goal in developing Python was to make it as user-friendly as possible. Python is an excellent language for beginners. Thanks to its reputation as a language that even a novice can pick up and use, Python has surpassed Java in popularity and is now considered the best language for beginners. Python, being a dynamically typed language, is known for its exceptional flexibility. Additionally, Python is more error-tolerant, so your programme will continue to build and run until it reaches the faulty section. Python is an easy-to-learn programming language that is also very tidy and lightweight. Structural and object-oriented programming approaches are among those that this language can accommodate. Additional styles are also available. The ability to import pre-existing modules and components from other languages gives Python a great deal of flexibility. With Python's libraries and frameworks, you may easily apply image categorization algorithms. Some of them include:

- **TensorFlow:** a popular deep learning framework that provides several resources for creating and training neural networks to perform image categorization tasks.
- **Keras:** In addition to providing an interface for ANNs, it acts as a TensorFlow library interface. Keras has been fine-tuned to reduce cognitive load, allowing DL to run with minimal human input.

4.4.2.2. Jupyter Notebook

Jupyter Notebook is a browser-based tool that enables interactive computing. It can be used for various computing tasks such as test automation, documentation, and executing code. The results of the code execution are displayed on the screen. Jupyter Notebook supports around 40 web applications that use analytical programming languages. Jupyter Notebook starts by default using the —Menu mode and a user must input code and execute it when Jupyter is like the manual

user guide. Besides the results of code execution, will be displayed instantly. Directly on the block the user can edit the code and provide more information about the line of code being used in the same window.

4.5 Results and Discussion

- **Input and Output Images**

A 1024 x 1024 RGB image from this dataset is provided as an example of the output. Various combinations of learning rates, epochs, and sample sizes were tested to achieve the most optimal fit for our data (see Figure 4.12). The input images are extracted from a minor subsection of unmasked images in the Face Mask Lite dataset. The model's ability to identify images in a variety of contexts is ensured by training it with a small number of random images. As illustrated in Figure 4.12 —a, b, c, d, which depicts GAN-generated images with distinct iterations (1..., 1000), the image quality improves with each succeeding epoch from 1 to 1000.

4.5.1. Plot image generated by the generator before training

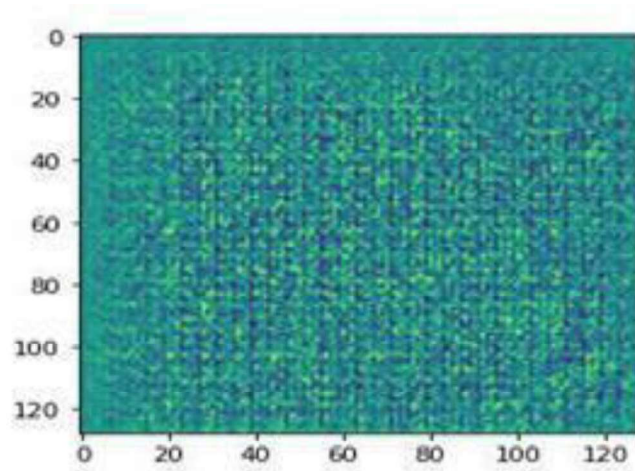


Figure 4.12: Production of the image by GAN generator without any prior training

Above Figure 4.12 shows the pixelated grid with various green and blue color shades. The interlaced multi-colored pixels give the frame an untidy appearance. It presents as a graph with horizontal and vertical axes between 0 to 120 and 0 to 100, respectively. It shows us an image that was produced with a GAN (Generative Adversarial Network) generator without training.

4.5.2 Images Generated by the GAN (Through Iterations 1, 3, 5, 7,11, 15, 20....1000)

Figure 4.13, Figure 4.14 and Figure 4.15 illustrates the process by which output images are generated from the input images at successive epochs ranging from 1 to 20. Figure 4.13 shows that the Generative Adversarial Network (GAN) takes to make images with successive iterations 1, 3, 5, 7... 11, 15, 20, Figure 4.14 depicts better image generation by iteration 20 and Figure 4.15 presents the generation of better images by iteration 1000. It has been observed that the quality of GAN-generated images with distinct iterations improves with each succeeding epoch.

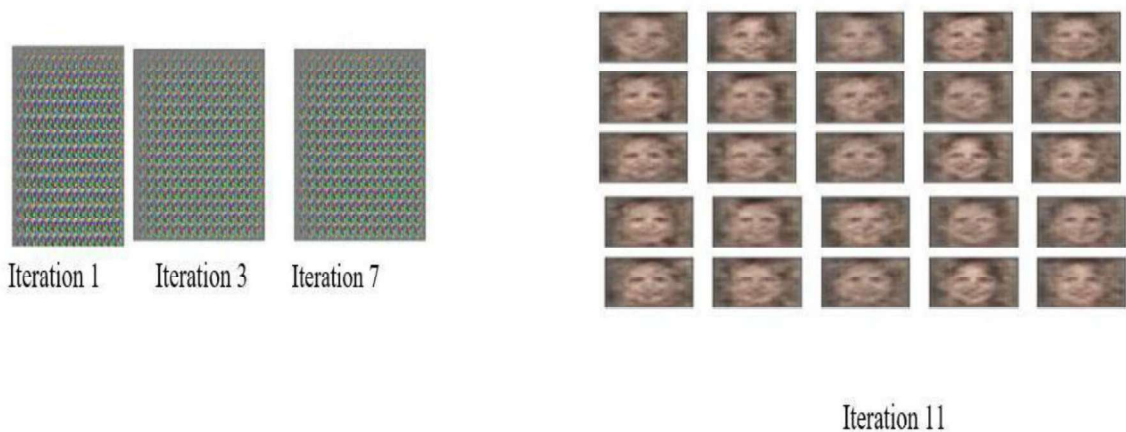


Figure 4.13: Image generated with successive iterations 1, 3, 7 and 11.



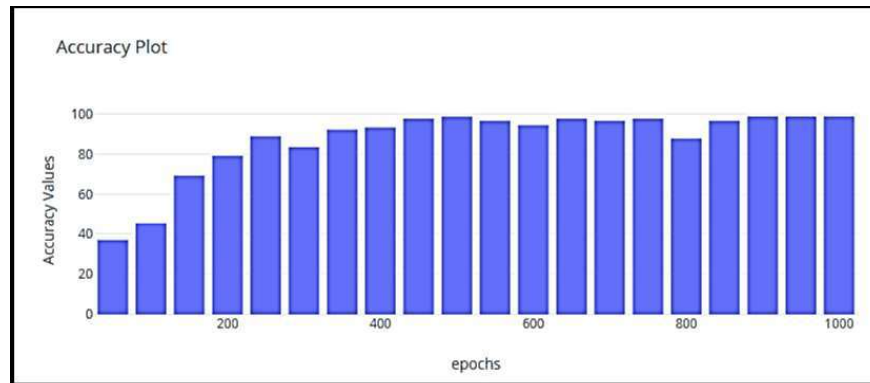
Figure 4.14: Image generated with successive iteration 500



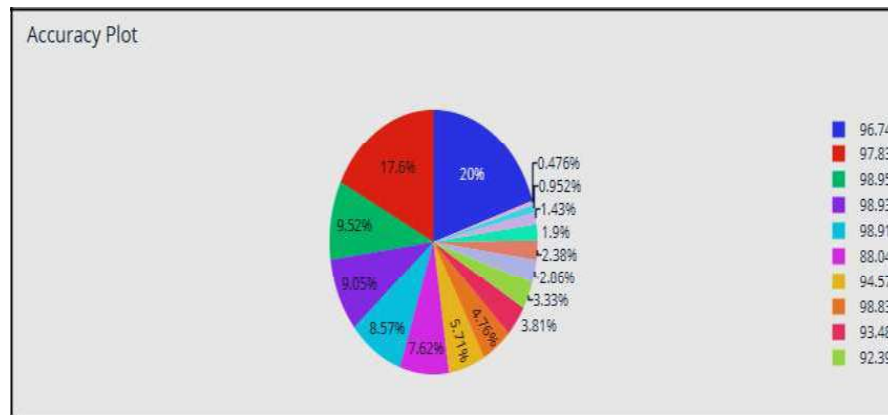
Figure 4.15: Image generated with successive Iteration 1000

Initially, throughout all the iterations, there are certain initial images that display extreme limitations and lack any obvious patterns or features. However, while the training progresses through epochs 3, 5, and 7, visible indications of development appear as the GAN images get more organized and begin showing the unique characteristics of the desired dataset. Across epochs 11, 15, and 20, the generated images have likely seen significant improvements, resulting in their extraordinary accuracy in portraying the original data. These images would exhibit exceptional levels of detail, textures, and other characteristics in comparison to real-world samples. The progressive enhancement in the GAN architecture has led to significant improvements in the quality and authenticity of synthetic images. This advancement is achieved through the asymmetrical training process between the generator and discriminator networks.

4.5.3 Proposed GAN model Accuracy Plots



(a)



(b)

Figure 4.16: Bar graph(a), Pie chart (b) illustrating the accuracy curve attained by the suggested Ensemble Model at its highest level

Given figure 4.16 (a) and (b) are the representations of the performance of the model in different epochs. On the x-axis, epochs are depicted with markers at every 200 intervals up to 1000, and on the y-axis, the accuracy values are shown, from 0% to 100%. The usage of blue bars is in an indication of the model's performance at each marked epoch. The accuracy trends indicate that at epoch 0, the accuracy begins barely above 40%. The accuracy improves from epoch 1 to epoch 200 and reaches around 80%. Since epoch 400 onwards, accuracy is maintained in the range of 90% and just below 100%, this means that the model is working consistently and performing at a high level. Table 4.6 displays the comparative accuracy of the proposed ensemble GAN model at various epochs.

Table 4.6:Table demonstrating Accuracy attained by the suggested ensemble GAN Model throughout a series of epochs

Iterations	Accuracy (%)
Epoch 50	45.63
Epoch 100	69.57
Epoch 200	79.35
Epoch 500	89.13
Epoch 700	96.74
Epoch 1000	98.31

The model achieves a 45.63 percent accuracy at epoch 50. At epoch 100, the accuracy increases to 69.57 percent, a remarkable enhancement, as evidenced by the average achievement of 69%. 57% correct predictions. At epoch 200, the model makes significant progress, reaching an accuracy of 79. The accuracy of the model was increased to 35% since it is still learning from the training data. At the end of epoch 500, the accuracy climbs to 89%. 13%, thus proving the model's ability to correctly interpret roughly 89% of the situations. 13% of predictions are correct. At era 700, the accuracy significantly increases to 96.74 percent, indicating a highly accurate model, with nearly 97% of the predictions being correct. Ultimately, the accuracy is 98% at epoch 1000.

4.5.4 Proposed GAN model mode collapse plot

Figure 4.17 further demonstrates that real data points expose the discriminator to local maxima of 0.74. In addition, since the produced sample is located within the data point's basin of attraction, bringing it closer to the actual datapoint is as simple as applying gradient changes to it directly. A variety of attractors, or local maxima, located in various parts of the data space led to a distribution of produced samples that is successful in reducing mode collapse. Over the course of several training epochs, Table 4.7 records the "Generator Mode values" of a GAN. These figures are

probably linked to some measure or quality of the generator's efficiency throughout training. At 50th Epoch, the "Generator Mode" value was 0.69, which means the generator is behaving in a particular way. Nevertheless, as training progresses, we observe fluctuations in this number. With a value of 0.68 by Epoch 100, we can see that the generator is consistently producing the same amount of power. The generator's behaviors show a slight change as there is a small drop to 0.67 at Epoch 200. One notable shift in the generator's output behaviors is the "Generator Mode" value, which climbs to 0.74 by Epoch 700. The generator is clearly generating unique patterns if this is the case. A further dramatic shift in the generator's output characteristics, maybe signaling a model changeover, is shown by the value's precipitous decline to 0.55 by Epoch 1000. Changing "Generator Mode values" illustrates how dynamic GAN training is. That the generator is always evolving and getting better at generating data is what they're getting at. When this metric shifts, it might be because the generator is producing different kinds of pictures or patterns. Monitoring and fine-tuning GAN models to get the required output and data quality requires such insights into the generator's behaviors.

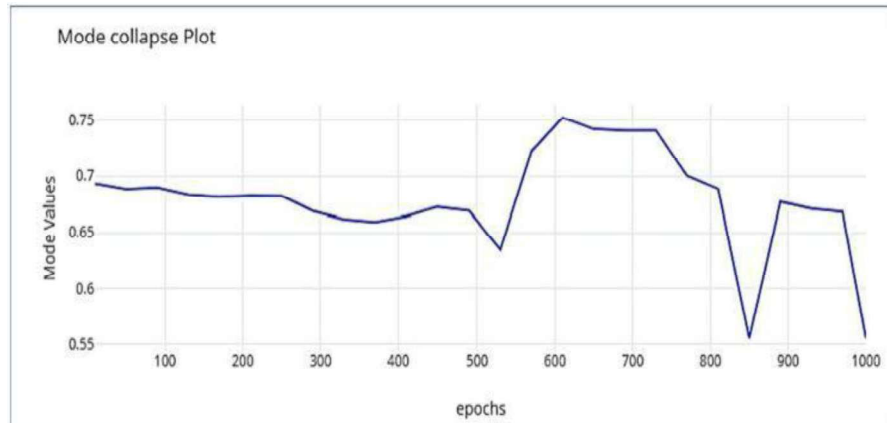


Figure 4.17: Mode collapse line plots of the suggested ensemble GAN model for fraud detection

Table 4.7: The suggested ensemble GAN model's generator obtained mode values for different epochs, as shown in the table.

Iterations	Generator
------------	-----------

	Mode values
Epoch 50	0.69
Epoch 100	0.68
Epoch 200	0.67
Epoch 500	0.69
Epoch 700	0.74
Epoch 1000	0.55

4.5.5 Proposed GAN model convergence plot

The generator produces a variety of data samples that span the whole target distribution, as seen in Figure 4.18. This proves that GAN is a well-converged generator that can span the data distribution and produce realistic data. A model of a GAN that has converged satisfactorily is shown in Figure 18. Epoch 50 saw the model's loss value at 1.259164, which was rather high and suggested it was still learning and improving its predictions. Nevertheless, the loss decreased dramatically with training, reaching 0.570837 by Epoch 100. The significant drop showed that the model was becoming better at making accurate predictions. The model was doing better by Epoch 500, when the loss value reduced to 0.128244. In the end, the loss dropped to 0.000143 at Epoch 1000, suggesting that the model's accuracy was probably close to ideal. Once this ensemble GAN model converges at Epoch 1000, it means it has learnt to generate data that is very similar to the underlying data distribution that was used for training. When the loss value is low, it means that the model's samples are close to the real thing and can't be told apart. Consequently, the GAN has successfully fulfilled its goal of generating synthetic data of exceptional quality, which may be applied in various ways such as generating images, enhancing existing data, and other possible applications. When we reach Epoch 1000, we can confidently say that the GAN's training was a huge success, and we can put our faith in the model to deliver realistic and accurate data samples.

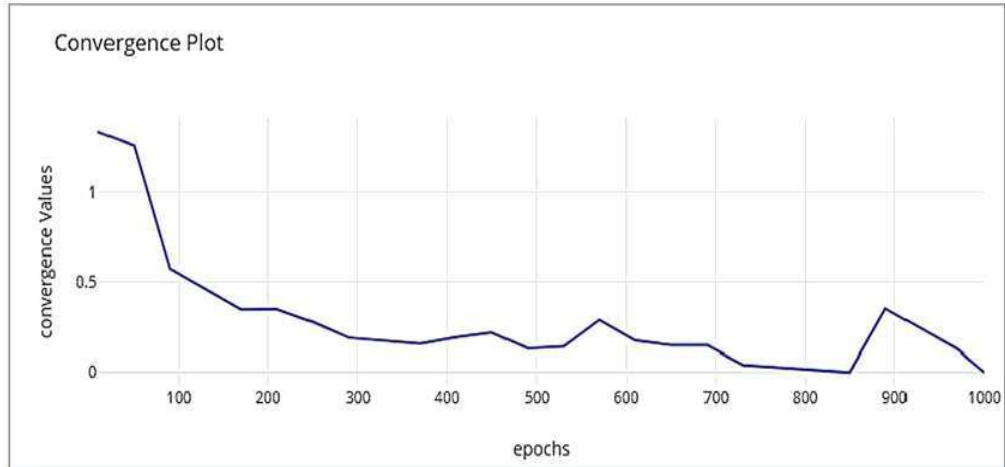
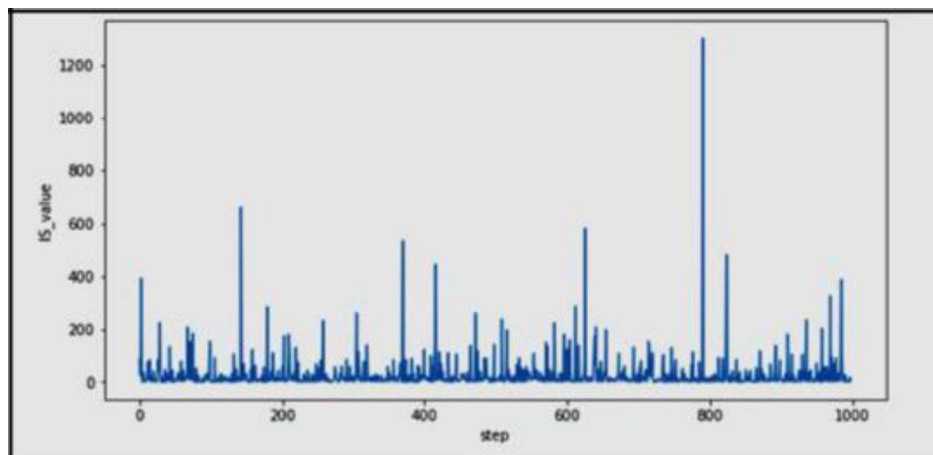
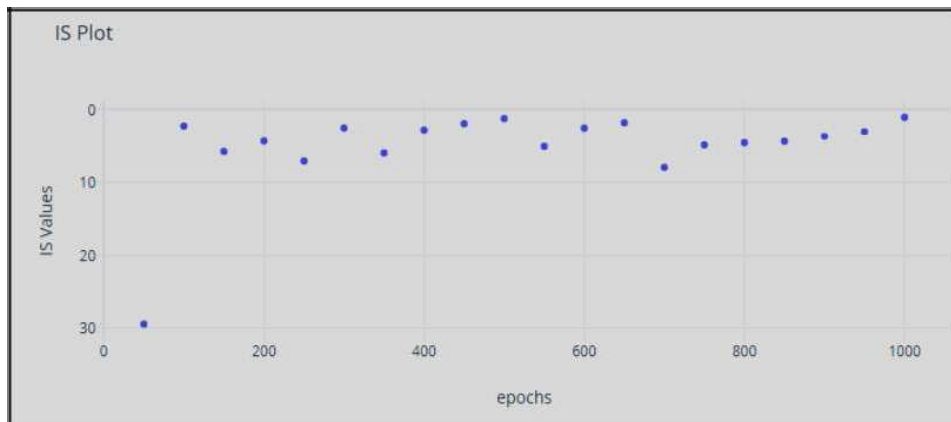


Figure 4.18: Charts showing the generator's performance in the suggested ensemble GAN model with better convergence

4.5.6 IS (Inspection Score) Curve



(a)

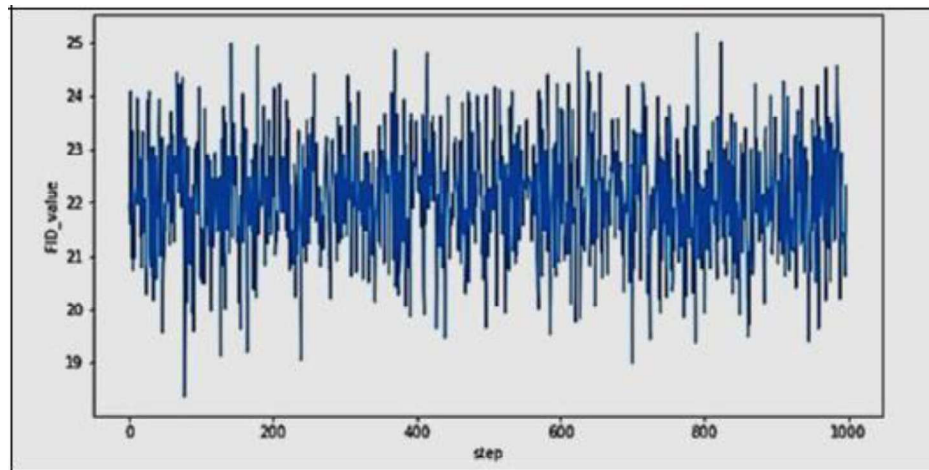


(b)

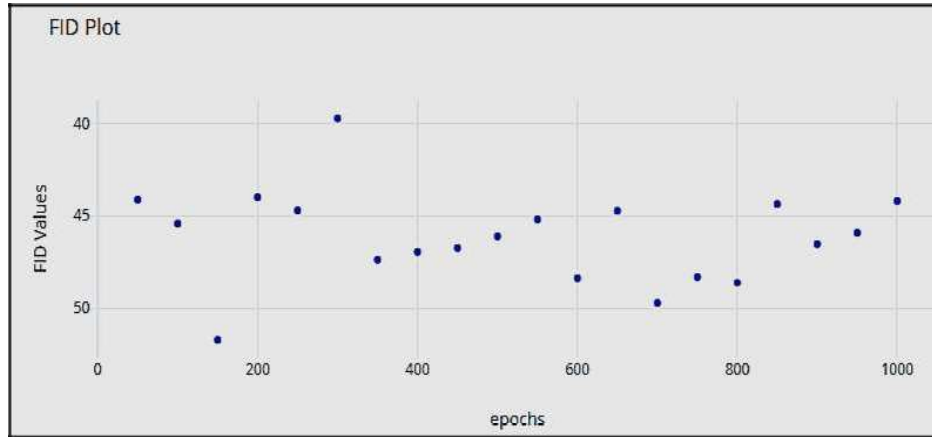
Figure 4.19: The proposed GAN ensemble model's Inspection Score (IS) Curve is presented in two scenarios: (a) utilizing all standards and (b) with specified standards after an outlier study.

The IS Curve of the proposed GAN ensemble model is illustrated in Figure 4.19. The x-axis labelled "step," spans from 0 to 1000, while the y-axis, denoted "fs's value," extends from 0 to 1200. (a) The IS curve is constructed utilizing all values, and (b) it exhibits specific values after outlier analysis. Blue dots are the markers of the values at each step, while the multiple small peaks are the ones which show the variations in the value of fs for each step. The most important step is the one where the value of it goes up to about 1200 fs at around step 800, thus indicating a considerable increase at that time.

4.5.7 FID (Fréchet Inspection Distance) Curve



(a)



(b)

Figure 4.20:The proposed GAN model's FID (Fréchet Inspection Distance) curve can be constructed by two methods: (a) utilizing all available data and (b) employing specified values after conducting an analysis of outliers.

The FID contours of a proposing GAN ensemble model are depicted in Figure 4.20, (a): utilizing all values and (b): the y-axis is designated "IS Values" and extends from 0 to approximately 40, while the x-axis is labelled "epochs" and contains specific values which have been analyzed for anomalies. Blue dots represent data points, with most clustering around an IS value of approximately 5 across various epochs. However, there is one outlier data point at an epoch less than 200 with an IS value of approximately 30, standing out distinctly from the rest of the data.

4.5.8 Total Computational time plot

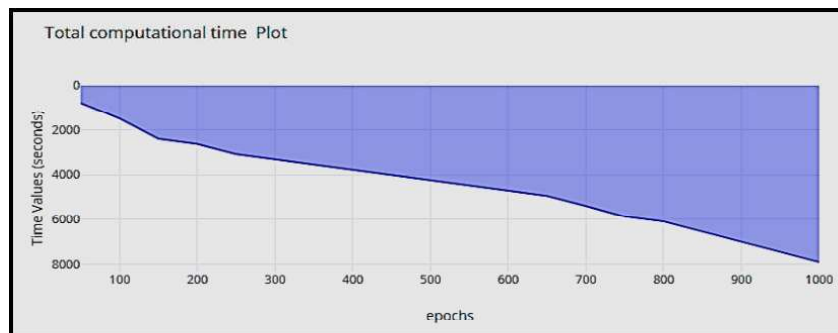


Figure 4.21:The graph displays the total amount of time (in seconds) taken by the proposed ensemble GAN to complete its computational tasks.

In the given Figure 4.21 shows the total computational time for ensemble GAN. This y-axis corresponds to —time values| (seconds), which range from zero to one thousand. The figure displays the y-axis with —Plot| as its label from 0 to 2000. The line graph itself is blue, and it progresses in a sloping direction as time passes. The graph features an evenly spaced hash mark on both the x- and y-axes. Completely, there is a 200-second interval on the x-axis and a 500-second interval on the y-axis. The line intersects the y-axis at 0, and it about ends in 1800 on the y-axis. Overall, it illustrates the positive relation of total computing time to seconds in time.

Time required to compute, in seconds, for different training epochs of a deep learning or ML model. The time necessary for the first phase of training is 784.312 seconds, as indicated by the TCT at Epoch 50. This includes tasks like updating parameters, doing optimization procedures, and forward and backward neural network passes. The TCT (Total Computing Time) grows to 1485.7316 seconds as training advances to Epoch 100, suggesting that the amount of time required for training during the first 100 epochs is larger. Since the model learns more complex patterns, computations become more demanding, and early training iterations are often faster, this is a common trend. Nevertheless, the computing demand increases throughout the training process. By Epoch 500, the TCT has jumped to 4252.4977 s, and by Epoch 1000, it reaches 7899.299 s. Due to the iterative nature of DL training, which involves complex computations, gradient changes, and model parameters at each epoch, the processing time increases. To develop better insights about good efficiency of the proposed model a comparative analysis is shown in Table 4.8. It was conducted to compare the values of Inspection score (IS) between the proposed ensemble GAN model and the specific values of current models after conducting an analysis to identify and remove outliers.

Table 4.8: Examination of the suggested ensemble GAN model's IS values in comparison to Existing models

Inspection score value (IS)			
Iteration	Proposed	DCGAN	Pix2Pix GAN CGAN
	ensemble mode		

Epoch 50	2.236	11.312	7.079	7.7747
Epoch 100	4.356	9.936	5.838	3.1131
Epoch500	1.19	5.853	3.987	5.7128
Epoch 1000	1.031	1.05477	1.721	2.67607

The proposed Ensemble Model stands out due to its continuous enhancement of image quality, as evidenced by its IS increasing from 2.236 at Epoch 50 to an impressive 4.356 at Epoch 1000. This growth signifies the model's effective training, making it an appealing option for tasks requiring high-quality image generation. In contrast, the "DCGAN" starts with a relatively high IS of 11.312 but shows slow improvement, reaching an IS of 1.05477 by Epoch 1000. The "Pix2Pix GAN" demonstrates a significant performance boost, with its IS decreasing from 7.079 at Epoch 50 to 1.721 at Epoch 1000, indicating its proficiency in enhancing image quality. Meanwhile, the "CGAN" exhibits inconsistent performance, with an IS of 7.7747 at Epoch 50 dropping to 2.67607 by Epoch 1000.

4.5.9 SSIM Curve refers to the Structural Similarity Index Metric Curve.

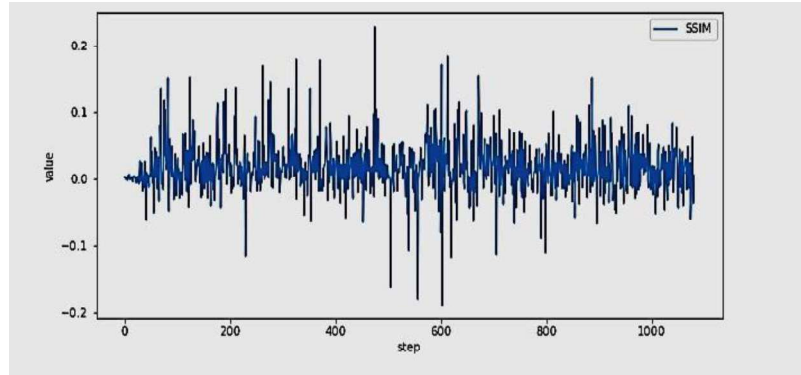


Figure 4.22:The SSIM Curve of the proposed GAN model generated by plotting two sets of variables: (a) all values and (b) values obtained after outlier analysis.

Given figure 4.22 shows the SSIM curve quantifies the structural similarity between two images. The x-axis is labelled as "step," ranging from 0 to 1000, while

the y-axis represents the "value," spanning from approximately -0.2 to 0.2. The curve consists of blue vertical lines, each representing an SSIM value at a specific step. Initially, there are significant fluctuations in SSIM values, especially around step 50. However, after this initial variation, the curve stabilizes, maintaining relatively low SSIM values for the remaining steps. Further analysis of specific values after outlier analysis may provide additional insights into the model's performance. Structural Similarity Index (SSIM) scores over several training epochs are shown in Table 4.9. When comparing the structural similarity of two images, the SSIM measure is employed; a higher SSIM score signifies a higher level of similarity. It evaluates the extent to which the computed images of the ensemble model resemble the real visuals here in this case. At Epoch 50, the value of the SSIM score is extremely small: 0.00284. This means that at this level of development, the Image Generation Module produces images that are structurally very distant from the genuine images. The graph of training dataset for SSIM reveals significant improvement when the training reaches Epoch 100; from the score of 0.03794. Thereby concluding that the true ensemble model is slightly improving the generation of images with structural resemblance to real images. The SSIM score rises to 0. The reason why the SSIM score is much lower compared to the PSNR and the SSIM is because the former measures the structure of the image by comparing the absolute difference between the peak pixel of the reconstructed image and the original image. 24023 by Epoch 500, which further proved the successful continuation of the model in producing more related structures of images. Finally, the SSIM score gets to 0 and as a result, the images are satisfactorily reconstructed by the proposed algorithm. It is seen from Figure 01397 at Epoch 1000 that there is much improvement in terms of the structural aspects of the images produced by the ensemble model. In summary, the pattern of the increase in SSIM scores presented in the previous section shows that the model under consideration continuously improves itself to construct images with a gradually enhanced structural similarity as the training phase grows. The given SSIM scores show that the ensemble model can learn to produce pictures with structural qualities that are quite like real data. The increasing SSIM scores

throughout training show that the model can improve its picture producing skills, which in turn produces images that are closer to the real dataset. This suggests that as the number of training epochs rises, the model improves in generating consistent and high-quality pictures. This is particularly beneficial for tasks such as picture synthesizing and data augmentation, that require the development of realistic and structurally similar data.

Table 4.9: The suggested ensemble model's SSIM scores over time periods

Comparative SSIM Values	
Iteration	Proposed ensemble model
Epoch 50	0.00284
Epoch 100	0.03794
Epoch 500	0.02995
Epoch 1000	0.01397

Table 4.10 compares the FID scores of several generating models at various training epochs. This set of models comprises the "Proposed Ensemble GAN," "DCGAN," "Pix2Pix GAN," and "CGAN." One way to compare the distribution of produced pictures to that of actual pictures is with the FID score; a lower number suggests an improved image.

Table 4.10 Review of current models and comparison of suggested ensemble GAN model's FID values

Comparative Fréchet Inception Distance (FID)				
Iteration	Proposed ensemble GAN	DCGAN	Pix2Pix GAN	CGAN

Epoch 50	47.14	53.953	119.85	39.083
Epoch 100	45.43	41.274	116.44	41.871
Epoch 500	46.11	59.331	88.35	38.028
Epoch 1000	44.21	54.775	79.76	37.914

The "Proposed Ensemble GAN" starts with a lower FID score (47.14 at Epoch 50) compared to the other models, suggesting that its initial picture distribution is closer to real data. Consistent image quality is indicated by the fact that the "Proposed Ensemble GAN" keeps competitive FID scores while training progresses. The other models, on the other hand, perform to varying degrees, with "DCGAN" eventually receiving lower FID ratings. The "Proposed Ensemble GAN" has a lower FID score of 47.14 than other models, including "DCGAN" 53.953, "Pix 2Pix GAN" 119.85, and "CGAN" 39.083, starting at Epoch 50. This implies that the "Proposed Ensemble GAN" commences by producing image distributions that are more closely aligned with the actual data. With a FID score of 45.43 at Epoch 100, "Proposed Ensemble GAN" has continued to dominate, with "DCGAN" (41.274), "Pix2Pix GAN" (116.44), and "CGAN" (41.871) continuing in the lead."

A comparison table highlighting other research in this area that are similar is included in Table 4.11 below. Thus, this work analyzes GAN based on factors such different GAN versions and training datasets, and it uses two assessment metrics (FID and IS). When trained using "Face Mask Lite dataset," the proposed ensemble GAN performs much better than the others, achieving an IS of 1.031, an FID score of 44, and an FID score of 47, respectively. 21, demonstrating the model's ability to generate finely detailed images that closely mimic real set photos. This means that it is applicable where there is need for image production and where strict structural similarity is required and quality of that image. The other GAN variations, however, present different IS scores in different generality while at the same time also resulting in lower FID values hinting at the possible disparities in performance.

Table 4.11: The summarized comprehensiveness of the IS, and FID scores utilizing other current approaches with the suggested ensemble model.

Models	Datasets	IS	FID
Our ensemble Model	Face mask	1.031	44.21
DCGAN	lite dataset	6.69	42.5
SA-GAN [131]	CIFAR10	18.65	52.52
LS-GAN [132]	ImageNet	6.76	29.5
SP-GAN [133]	CIFAR10	74	166.5
SN-GAN [134]	ImageNet	36.80	27.621

In the —Face Mask Lite dataset, —Proposed Ensemble GAN can be said to have a good work capability since it can generate images that possess good structural similarity as well as high image quality. ACGAN also applies to well-maintained DCGAN and LS-GAN with reasonable structural similarity and picture quality. In the case of SA-GAN and SP-GAN, there is less of a resemblance to real image structure and, hence, higher FID implying a difference in the distribution of images. In terms of the FID score, SN-GAN produced low scores meaning that it is properly diversified in the real image distribution, and the structural similarity yields moderate results. These insights help academics and practitioners to decide about the best GAN versions on different datasets that are locally and globally available for image-generating applications. In conclusion, the suggested GAN model presents a novel and fascinating approach to generative modeling. It involves combining many discriminator networks into a single generator by utilizing a variable loss function along with characteristics such as FID, IS, and Binary Cross-Entropy. Using the talents of several discriminators, the model may produce more high-quality, structurally connected pictures by implementing this ensemble approach. For the suggested ensemble model, the optimal IS and FID parameter values are 1.031 and 44.21,

respectively, beating the competition. Moreover, the approach's "Dropout" in the ensemble design helps provide an explicit ensemble of networks for testing and fixes the mode collapse issue that frequently arises during GAN training of shared weight models. Its use is restricted to the discriminator architecture that it chooses, and training a network dropout takes twice as long as it does for a conventional neural network. In a later study, the following methods work well for increasing the values of the assessment parameters.

- **Grouping the classes:**Organizing the data based on its several classes is a common strategy for combating the Mode Collapse problem. Consequently, sub-batches and validated batches can be distinguished using the discriminator.
- **Anticipating Counteractions:** This strategy trains the generator to outdo the discriminator while still accounting for the discriminator's counteractions; the goal is to stop the discriminator from "chasing" the generator. Longer training time and more complicated gradient computation are two of the technique's drawbacks.
- **Learning from Experience:** Using a predetermined number of iterations, the discriminator is trained using previously created false samples.

4.6 Conclusion and Future Scope

This study contributes to the advancement of GAN research by introducing an innovative ensemble model that effectively addresses common training challenges and generates high-quality, diverse samples. It introduces a generalized novel image forgery detection model based on an ensemble of Generative Adversarial Networks (GANs). This model leveraged multiple discriminators to enhance the accuracy and robustness of deepfake detection. Through a unique voting ensemble approach, we dynamically selected the most effective discriminator determined by using a variety of loss functions, including adversarial loss that included adversarial loss and additional metrics like SSIM. This method not only

improved the detection of forgeries but also mitigated common GAN training challenges such as mode collapse and convergence issues.

The model was thoroughly evaluated employing several key metrics, including FID, SSIM, TCT, and Inception Score (IS). The results demonstrated the model's effectiveness in generating high-quality images and accurately identifying deepfakes. However, despite these advancements, the model faced limitations in generating consistently high-quality images and achieving optimal deepfake detection under all conditions.

Future work will focus on enhancing the generalized novel image forgery detection model through the development of the Ensemble WGAN (EWG). This advanced model will integrate the Wasserstein Loss to further improve image generation quality and stability. By leveraging this novel approach, we aim to address the remaining challenges of mode collapse and convergence more effectively.

Additionally, future research will explore the integration of more diverse and sophisticated loss functions to further enhance the robustness and accuracy of deepfake detection. Expanding the dataset to include a wider variety of image types and forgeries will also be a key focus, ensuring that the model is applicable to a broader range of real-world scenarios. Furthermore, optimizing the computational efficiency of the model will be prioritized to make it more practical for real-time applications in digital forensics and media integrity. This continued development aims to solidify the model's position as a state-of-the-art solution in the fight against image forgery and deepfakes.

CHAPTER-5

Heterogeneous Discriminator Approach with Ensemble WGAN (EWG) model for Enhanced Image Synthesis and Deepfake Detection

Aim

The Ensemble WGAN (EWG) is an advanced Generative Adversarial Network (GAN) model that incorporates Wasserstein Loss to improve image generation quality and stability. Employing an ensemble approach with multiple discriminators enhances both image generation and deepfake detection accuracy. The EWG addresses common GAN challenges such as mode collapse and convergence issues, providing a robust solution for generating realistic images and identifying deepfakes. It aims to surpass the previously proposed ensemble model discussed in the previous chapter by achieving superior performance in both image quality and deepfake detection accuracy.

Objectives

- **To introduce a novel ensemble model based on Wasserstein Generative Adversarial Networks (WGANs):** Develop a robust model that effectively addresses training challenges, including mode collapse and convergence issues. This is achieved by integrating the Wasserstein distance, which offers superior image generation quality and stability, ultimately improving the performance of the ensemble GAN.
- **To implement a modified diversity loss function:** Employ a specialized loss function designed to minimize discrepancies between the generated and original distributions, enhancing the quality of synthetic data.

- **To utilize a voting ensemble approach with multiple discriminators:** Combine the strengths of multiple CNN discriminators to improve the model's decision-making accuracy using a max-voting technique.
- **To evaluate and compare the model's performance:** Assess the proposed model against existing GAN models using metrics such as Total Computational Time, Structural Similarity Index Metric (SSIM), Fréchet Inception Distance (FID), and Inception Score (IS).
- **To enhance the stability and picture quality of the GAN model:** Address mode collapse and convergence issues to ensure consistent and high-quality image generation by leveraging the Wasserstein loss in WGANs.
- **To develop an optimal GAN model for effective deepfake detection:** Ensure the model is robust and reliable even with limited data availability, making it suitable for real-world applications in digital forensics and media integrity.

5.1 Introduction

The Ensemble WGAN (EWG) represents a significant advancement in the field of GANs, offering a robust and efficient solution for generating high-quality images and detecting deepfakes. By leveraging Wasserstein Loss, an ensemble of discriminators, majority voting mechanisms, and SSIM integration, the EWG model addresses and overcomes many of the limitations faced by traditional GANs. It sets a new standard for performance and reliability in image generation and deepfake detection, making it a valuable tool for various applications in deep learning and computer vision. It enhances existing methodologies by integrating Wasserstein Loss instead of the traditional adversarial loss. This integration aims to improve image generation quality and stability, addressing persistent challenges like mode collapse and convergence issues commonly found in standard GANs.

The previous ensemble GAN model discussed in Chapter 4 utilized the same ensemble technique but relied on the standard GAN loss function. While this approach showed promise, it faced several limitations. Standard GAN loss often leads to issues such as mode collapse, where the generator produces a limited variety of images, and training instability, which can result in the generator and discriminator failing to converge. These pathologies hinder the model's ability to produce consistently high-quality images and negatively impact its performance in deepfake detection.

In contrast, the Ensemble WGAN (EWG) model leverages the Wasserstein Loss, which offers significant improvements over the adversarial loss used in the previous model. Wasserstein Loss provides a more stable and reliable training process by maintaining a smooth gradient flow, thereby mitigating issues like mode collapse and ensuring better convergence. As a result, EWG generates more realistic images with fewer epochs, enhancing the efficiency and effectiveness of the training process. This improvement not only leads to superior image quality but also achieves a higher accuracy rate in deepfake detection, as evidenced by the model's performance metrics on challenging datasets. By addressing the training pathologies more effectively, EWG represents a substantial advancement over the Chapter 4 ensemble GAN model.

Moreover, the incorporation of Structural Similarity Index Measure (SSIM) into the EWG model further distinguishes it from the previous ensemble GAN. SSIM evaluates the structural features, contrast, and brightness of generated images, promoting the creation of visually and structurally similar images to real ones. This integration helps penalize deviations in both pixel and structural information, resulting in more convincing adversarial images. The previous model, lacking this sophisticated evaluation metric, struggled to achieve the same level of image authenticity and structural coherence. Consequently, the EWG model not only surpasses the previous ensemble GAN in generating high-quality images but also demonstrates superior performance in deepfake detection,

making it a more robust and reliable solution for practical applications in image generation and verification.

5.2 Key Innovations and Features of EWG

1. **Wasserstein Loss Integration:**

- Wasserstein Loss, known for its stability and convergence properties, replaces the adversarial loss, making the training process more robust and reducing the risk of mode collapse. This change allows for more reliable and higher-quality image generation.

2. **Ensemble Approach with Multiple Discriminators:**

- EWG employs an ensemble of three distinct Convolutional Neural Networks (CNNs) as discriminators. This ensemble approach improves the model's ability to differentiate between real and generated images, thereby enhancing both image generation and deepfake detection performance.

3. **Majority Voting for Deepfake Detection:**

- The ensemble of discriminators utilizes majority voting. This method increases the robustness of deepfake detection, leading to higher accuracy and reliability in identifying manipulated images.

4. **Superior Performance Metrics:**

- The EWG model demonstrates superior performance on challenging datasets, such as Indian Actor Images and 5 Celebrity Faces, achieving high accuracy rates (98.480% and 96.417%) and optimized evaluation metrics (10.4, 32.71) and (11.5, 37.52), respectively. These results outperform previous ensemble models, showcasing EWG's effectiveness.

5. **Structural Similarity Index Measure (SSIM):**

- The model integrates SSIM into its loss function, which analyses the validity and quality of generated images by considering structural features, contrast, and brightness. This helps ensure that

the generated images are not only visually appealing but also structurally like real images.

- The diverse loss function is formulated as $\mathcal{L}_{div} = \mathcal{L}_{adv} + \lambda \mathcal{L}_{div}$ which is required to be minimized to promote the generated images to be visually and structurally similar, and adversarial convincing. It further helps to classify deepfake by penalizing images that deviate from the original one in both pixel and structural information. This loss function promotes the generation of high-quality, adversarial convincing images and aids in the detection of deepfakes by penalizing deviations from original images in terms of both pixel and structural information.

6. Optimized Evaluation Parameters:

- EWG excels in key performance metrics, including SSIM, Fréchet Inception Distance (FID), and Inception Score (IS), as well as overall computation time. These optimized values reflect the model's efficiency and effectiveness in generating realistic images and detecting deepfakes, setting a new benchmark in GAN performance.

5.3 Related Research Work

The initial GAN model, introduced by Ian J. Goodfellow et al. in 2014, revolutionized unsupervised generative modelling by automatically finding data distributions. GANs comprise two primary models: the Generator module, typically a convolutional-based network, and the Discriminator module, often a deconvolutional-based network. These models work together using noisy contrastive estimates and loss functions to generate synthetic data. Since 2017, GANs have been extensively used for image enhancement techniques, particularly with human faces, producing high-quality graphics. While semi-supervised and

reinforced learning is possible, most GAN implementations rely on unsupervised machine learning algorithms. Effective applications include image-to-image translation, frame-by-frame prediction, and text-to-image synthesis. Despite trade-offs in PSNR/SSIM metrics, models have shown perceptual improvements, reconstructing more detailed images.

GAN models inherently face several challenges, particularly in visual data reconstruction, database and training stability issues, and visual enhancement and prediction applications. Comparative studies and performance evaluations have highlighted these challenges, leading to advancements in ensemble architecture for GAN reconstruction models. The literature and our proposed model in the previous chapter identify four main challenges in GAN models:

1. Visual data reconstruction difficulties.
2. Database and training stability issues.
3. Comparative studies and performance evaluations.
4. Advances in ensemble architecture for GAN models.

Deepfakes, a significant application of GANs, utilize machine learning, vocal synthesis, and facial recognition to produce authentic-looking audio, video, or image content. While deepfake technology offers legitimate applications, it also raises serious security and societal concerns due to its potential for misuse in altering video content.

5.3.1 Wasserstein GAN (WGAN)

A particular kind of GAN, called the Wasserstein GAN (WGAN), addresses common issues like mode collapse and training instability in traditional GANs by employing the Earth Mover's Distance (also known as Wasserstein Distance) as the loss function. The Wasserstein distance has a rigorous mathematical definition given by Arjovsky et al.[135] . This shift in loss function enhances training stability, improves convergence, and results in higher-quality samples[136].

When the generator generates a small subset, mode collapse is a severe problem, failing to accurately reflect the data distribution. The WGAN approach mitigates this by incentivizing the generator to learn the dataset distribution directly from optimal transport theory, avoiding the pitfalls of a zero-sum game with samples. When using GANs for anomaly detection, preventing mode collapse is crucial. Mode collapse can lead to missing modes being mistaken for anomalies, increasing false positives. The WGAN model addresses these challenges by implementing several modifications:

1. Replacing the sigmoid output layer in the critic model with a linear activation function.
2. Labelling real photographs with a -1 and phony images with a 1.
3. Using the W_loss to upgrade the generator and critic or discriminator's models.
4. Compared to the generator, a higher number of critic model modifications are made per cycle.
5. Gradient descent with a sluggish learning rate and no momentum using the RMS Prop method.

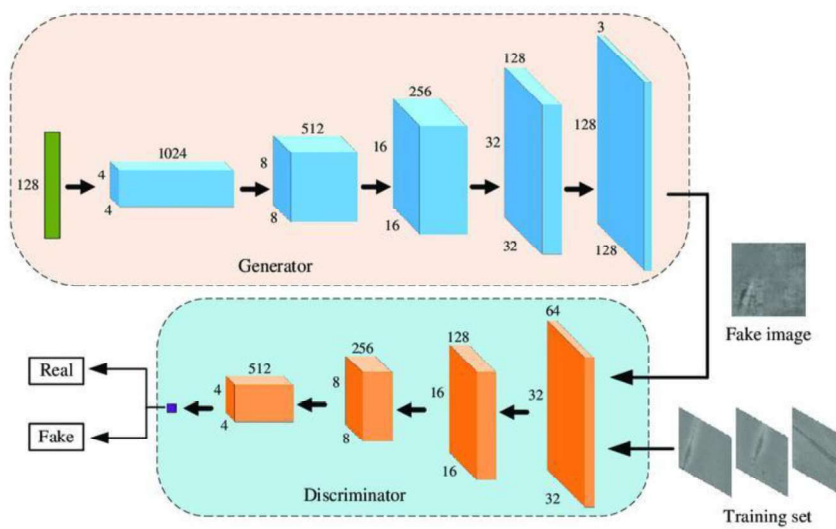


Figure 5.1: The Basic WGAN model[137]

5.3.2 Evolutionary Methods for Deepfake Detection

This paper presents an evolutionary method for detecting deepfakes using a Wasserstein GAN-based ensemble discriminator architecture with enhanced SSIM, Inception Score, and Fréchet Inception Distance approaches. These improvements support various fields, including mechanics, finance, and medical applications, addressing the growing problem of fraudulent video generation and detection. GAN models, designed for image processing, use basic convolutional filters to identify deepfake issues. However, deeper networks may require consolidated data and more processing units. By integrating SSIM, which includes luminance, contrast, and structure components, the proposed method enhances the detection and quality assessment of generated images. This approach significantly improves the reliability and performance of GAN models in detecting and generating high-quality, realistic images, thereby providing a robust solution for combating deepfakes. In nutshell, the Ensemble WGAN (EWG) model represents a significant advancement in GAN technology by addressing the limitations of previous ensemble GAN models. By integrating Wasserstein Loss and employing a CNN-based ensemble discriminator architecture with enhanced evaluation metrics, the EWG model achieves superior image quality and higher accuracy in deepfake detection, setting a new benchmark in the field.

5.4 Proposed Methodology

The proposed model is an ensemble-based extension of the Wasserstein GAN (WGAN), which recreates the WGAN using a voting ensemble technique involving three discriminators and a single generator. The generator's weights for a given epoch are adjusted by selecting the discriminator with the lowest Diverse loss value, as shown in Equation 5.1. By decreasing the distance between the created and original distributions, the model uses the Wasserstein distance

(W_loss) to improve training consistency. Utilizing a majority vote ensemble technique and multiple CNN models, the discriminator component identifies deepfakes. The primary objective is to enhance the generator's gradients by using the weights updated by a specific discriminator. To optimize learning and yield optimal outcomes, the method combines concurrent training of the generator and discriminator models. This strategy establishes the following criteria:

- a. Identifying a more accurate discriminator D (better approximating $\min W(\cdot, \cdot)$).
- b. Creating a discriminator ensemble better suited to the generator's G capabilities.
- c. Overcoming common GAN training issues, also including local feature ambiguity and global structure failure.

In this work, generative models are repurposed to produce samples by an objective function $\mathcal{L} = \prod_{i=1}^3 \{ \cdot \}$ to perform the pairwise evaluation of separately trained networks mentioned in equations 5.1, 5.2, and 5.3. To improve training consistency and eliminate pixel manipulations using Wasserstein distance and clipping techniques, the researchers created a new dataset. Diverse loss calculates the minimum W-loss among three discriminators using the SSIM score, with the importance of the SSIM term controlled by a coefficient called α .

$$\mathcal{L} = \sum_{i=1}^3 \left(\frac{1}{2} \left(\mathcal{L}_{\text{Wasserstein}} + \mathcal{L}_{\text{SSIM}} \right) \right) \quad (5.1)$$

$$\mathcal{L} = \sum_{i=1}^n \left(\frac{1}{2} \left(\mathcal{L}_{\text{Wasserstein}} + \mathcal{L}_{\text{SSIM}} \right) \right) \quad (5.2)$$

$$\mathcal{L} = \sum_{i=1}^3 \left(\frac{1}{2} \left(\mathcal{L}_{\text{Wasserstein}} + \mathcal{L}_{\text{SSIM}} \right) \right) \quad \text{for } i=1 \text{ to } 3. \quad (5.3)$$

The aim is to improve the generator's gradients by using the weights updated by a specific discriminator. To optimize learning and achieve the best results, this approach integrates and simultaneously trains the generator and discriminator models. The block diagram of the proposed ensemble WGAN model is shown in Figure 5.1 below.

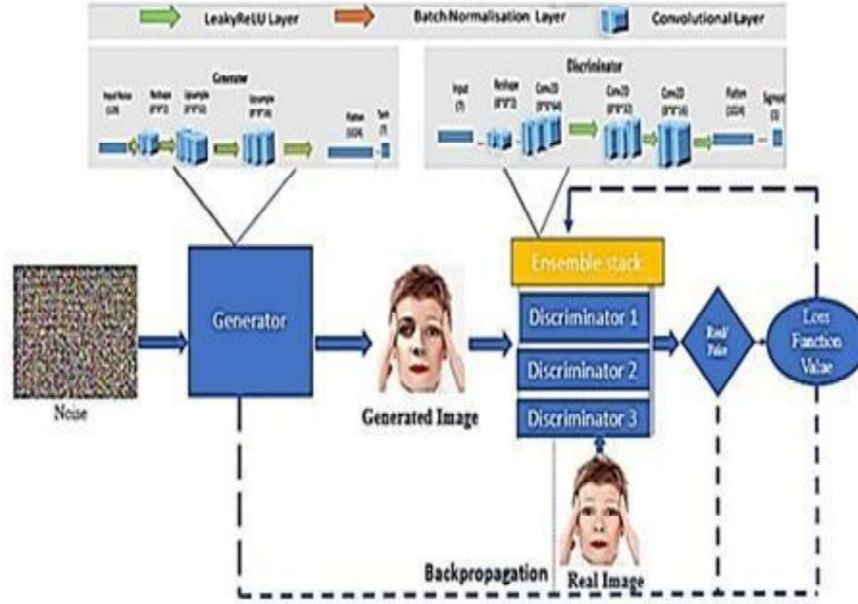


Figure 5.2: Block diagram of Ensemble Wasserstein(EWG) Model showing modified WGAN with

The architecture of the EWG model, comprising one generator and three discriminators, is depicted in Figure 5.2. Feedback from various discriminators is collected for the generator's training. GGG is trained against the best discriminator if $E^* = \arg \min_{g \in G} \max_{d \in D} W(g, d)$ where E^* represents the optimized ensemble, e_i represents the loss value calculated for each instance $\{g, d_i\}$ within the ensemble. Each instance consists of a generator g and a discriminator d_i , and W_i represents the Wasserstein-1 or W-loss operator, which calculates the minimum value among the given set of loss values. P_r represents the actual data distribution. The noise distribution is shown by P_z . A sample of the actual data distribution P_r is represented by the variable x . Z represents a sample taken from the noise distribution. P_z . $G(z)$

represents the generated sample obtained by running noise z through the generator G .
When given a real data sample x , the discriminator D 's output is represented by $D(x)$.
Given a generated sample $G(z)$, $D(G(z))$ is the discriminator's output.

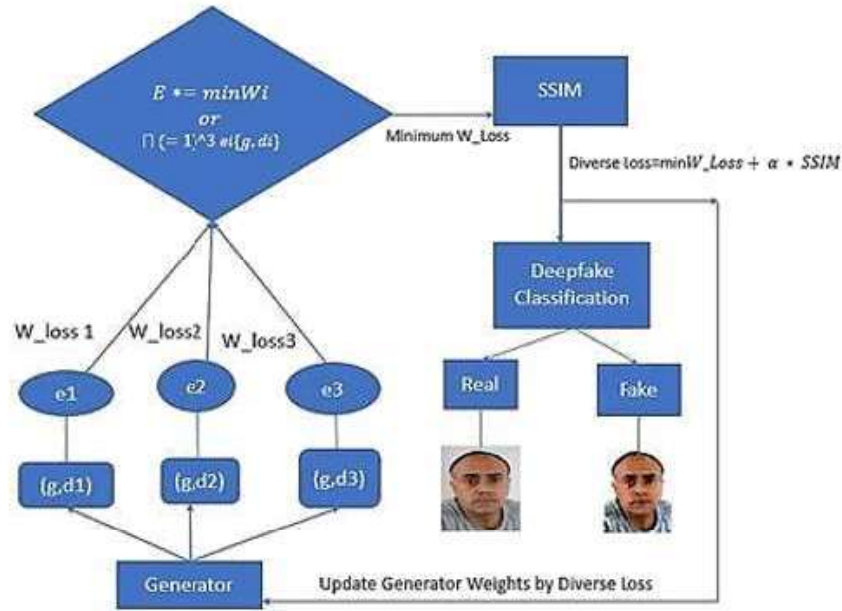


Figure 5.3: The detailed Architecture of the proposed Ensemble Wasserstein (EWG) Model

5.4.1 Simplified Loss Function (Wasserstein Loss)

To convert the discriminator from a classifier to a critic, the WGAN model presents a novel loss function that seeks to maximize the difference in scores for true cases and minimize it for erroneous ones. By doing this, the loss of authentic and fake photos is reduced. Mathematically, the Wasserstein separation between the generated data distribution P_g and the real data distribution P_r is shown in Equation 5.4:

$$W(P_r, P_g) = \inf_{\gamma \in (\Pi(P_r, P_g))} \int \sum_{(x,y)} \|x - y\| d\gamma(x,y) \quad \dots\dots\dots (5.4)$$

Where, the set of all joint distributions (x, y) with marginals that are respectively P_r and P_g is denoted by the notation $\Pi(P_r, P_g)$. If P_r represents the real or current distribution and P_g represents the desired or generated distribution, the Wasserstein distance measures the cheapest transportation cost. Infimum, or the greatest value that is less than all the components in a set, is another name for this minimum loss. To define the execution steps figure 5.4 mentions the step of

execution below.

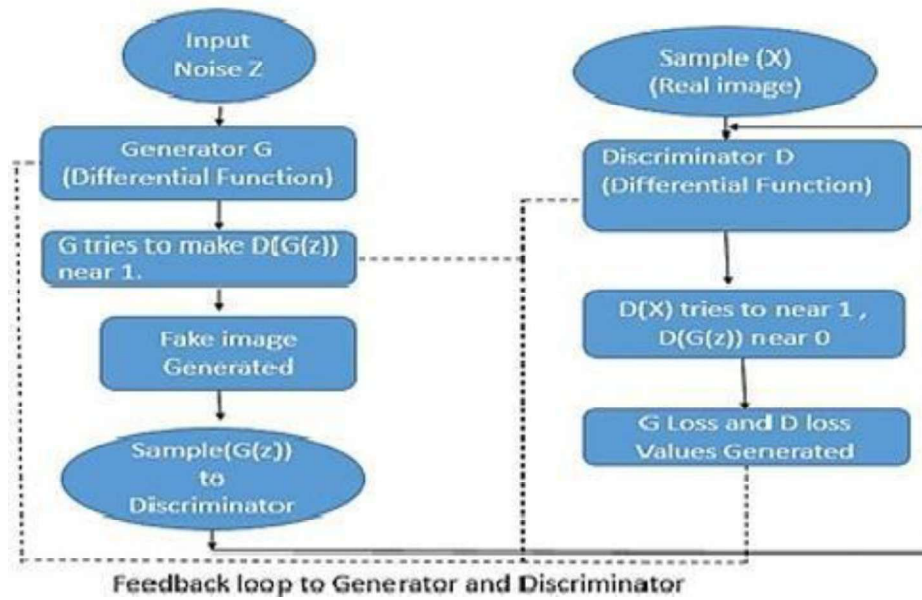


Figure 5.4: The steps of execution of the proposed GAN model

5.4.1 Mathematical Notation of proposed GAN function

Table 5.1: Mathematical Notation of proposed GAN function

Function: Proposed Ensemble GAN function.

Ensemble GAN(Best_Discriminator, Generator, shaped input, latent_dimension):

1. # discriminator module:

discriminator = optimal discriminator (shaped input)

discriminator. Compile (optimizer = "Adam (lr=0.0002, beta_1=0.5," loss = "binary_crossentropy")

2. # Generator modules:

generator = generator (latent dimension)

3.# Ensemble_GAN

gan = Sequential Function ()

gan. utilize (generator)

```

gan. utilize (discriminator)

gan. Build (optimizer=Adam, loss='binary_crossentropy', beta_1=0.5, lr=0.0002)

Return gan.

```

5.4.2 Training Algorithm

Table 5.2: Training Algorithm used for training of Proposed Ensemble WGAN model

Algorithm 1: EWG: Ensemble WGAN Model.

Input: Real samples and generated samples are entered.

Output: Discriminator Loss and Generator Loss.

Step 1: Set the Generator, D1, D2, and D3 to their initial values.

2. Set batch size = 128 and epochs=1000. For each batch: For every epoch:

3. Sample a batch of real samples.

4. Use the Generator to create a batch of fake samples.

5. Determine each discriminator's Wasserstein loss: -Determine the values of $W1 = \text{loss}$

(D1, real samples) and $W1_fake = \text{loss}$ (D1, generated samples). - Determine the values

of $W2 = \text{loss}$ (D2, real samples) and $W2_fake = \text{loss}$ (D2, generated samples). – Determine

the values of $W3 = \text{loss}$ (D3, real samples) and $W3_fake = \text{loss}$ (D3, generated samples).

6. Pick the discriminator that has the smallest Wasserstein loss: -If $W1 > W2$, and $W1 >$

$W3$, use D1 for additional calculations. If $W2 > W1$ and $W2 > W3$, use D2 for additional

calculations. -If $W3 > W1$ and $W3 > W2$, D3 for additional calculations.

7. Calculate SSIM, IS, FID and Total Computational Time.

8. Calculate diverse loss $= (W1, W2, W3) + \dots$

9. Update Generator weights using diverse loss: - Calculate generator loss= loss

(Diverse_loss, generated samples) - Adjust the weights in the Generator based on G_loss.

10. Update the Selected Discriminator: - Using the chosen Discriminator, compute the

Wasserstein loss for real and fake samples. - Adjust the weights of the Discriminator to

account for the Wasserstein loss.

11. Repetition of steps 3 to 10 for the batch's allotted number of iterations.

12. Repeat steps 2 to 11 for all specified epochs.

End: Training Process completed.

5.4.3 Dataset

To demonstrate the model's interoperability in various circumstances, two datasets were built using the Indian Actor images and the 5-celebrity images dataset, respectively. The Indian celebrity dataset proved to be a more effective for proposed model. A 64x64x3-pixel RGB three channel picture is commonly used as the input image. Indian Actor photos and 5-Celebrity Faces databases are the sources of the photos. The images have been acquired from Indian Actor Images and 5-Celebrity Faces datasets. The input image is frequently a three-channel RGB with size (64x64x3) -pixel shown in Fig.5.5. The array of input images of both the datasets are mentioned below in Fig.5.6. It is possible to obtain the data set for free at:

<https://www.kaggle.com/dansbecker/5-celebrity-faces-dataset> [138]

<https://www.kaggle.com/iamsouravbanerjee/indian-actor-images-dataset> [139].

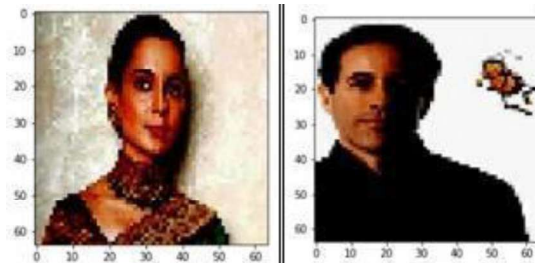


Fig.5.5: Input Images Size (64x64x3) used for execution of Proposed model.

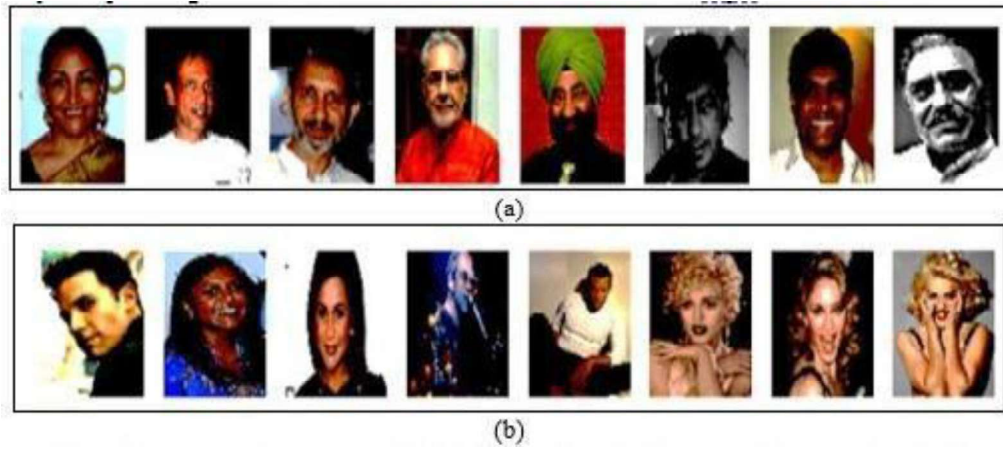


Fig. 5.6: (a) Real images from Indian Actor Images and (b) 5-Celebrity Faces datasets used for execution of Proposed model.

5.4.4 Model Evaluation Parameters

The performance of the ensemble GAN model is check out using the following metrics.

- **Accuracy:** Accuracy, which assesses the proportion of instances in the dataset that were correctly classified, is among the most fundamental performance metrics.
- **Inception score (IS):** Measures the quality and diversity of generated images.
- **Fréchet Inception Distance (FID):** Assesses the similarity between real and synthetic image distributions.
- **Structural Similarity Index Metric (SSIM):** Evaluates the perceived quality of generated images by comparing structural information to real images.
- **Total Computational Time Function:** Tracks the time taken for the model to complete its training and generation tasks, ensuring efficiency in the GAN's execution.

5.5 Experimentation

The proposed model is implemented using Jupyter (Anaconda framework) and is specifically designed to manage parameter updates between the generator and discriminator networks, unlike the standard parallel framework. The input image is typically a three-channel RGB image with a size of 64x64x3 pixels. These images are sourced from the 5-Celebrity Faces dataset and Indian Actor Images dataset, providing a diverse set of facial features and expressions necessary for training robust deepfake detection models.

5.5.1 Experimental Setup

Implementation Environment:

- **Platform:** Jupyter Notebook (Anaconda framework)
- **Languages and Libraries:** Python, TensorFlow, Keras, NumPy, SciPy, scikit-learn, Matplotlib, PyTorch
- **Development Tools:** Google Collaboratory for additional computational resources

Hardware Requirements:

- **Computer:** A high-performance workstation or server
- **CPU:** Multi-core CPU (e.g., Intel Core i7 or above)
- **GPU:** CUDA-compatible GPU (e.g., NVIDIA GeForce GTX or RTX series)
- **RAM:** Minimum of 16 GB
- **Storage:** Sufficient space for datasets, model checkpoints, and interim results
- **Operating System:** Compatible with Linux, Windows, macOS

5.5.2 Training

The generator is trained using a Gaussian random number sample, inverted labels

and loss functions, and batch norm statistics. Stride in convolutional layers is used to downsample the discriminator model, while deconvolutional layers are used for up sampling. The generator creates its own spatial up sampling, and the final convolution layer is flattened and fed into a single sigmoid output. The discriminator uses low random noise and label smoothing to enhance image quality. Training employs a 50% dropout rate, with activations standardized by batch normalization. Inputs are normalized to the range $[-1,1]$, and tanh is used in the generator output. All models include batch norm layers, except for the discriminator's input and output. GANs are trained using Binary Cross Entropy (BCE) for classification tasks.

5.5.2.1 Input Parameters

The architecture and behaviour of the GAN model are defined by the following input parameters during training and generation is shown in Table 5.3 below.

Table 5.3: Description of input parameters during training and generation

Parameters	Value
Batch Size	128 images
Learning Rate	0.0002
Momentum (Beta)	0.5
Optimizer	Adam's Optimizer
Epochs	1000
Loss Function	Wasserstein Loss
Dataset	Indian Actor Images and 5-Celebrity Faces datasets
Evaluation Metrics	IS, SSIM, FID, and Total Computational Time

5.5.2.2 Hyperparameters Scheme

1. **Hp drop rate:** The difficulty, user involvement, and overall generator-

discriminator minimax game experience are all considered throughout the design and balancing of the game. To create balance and dynamics, the HP loss rate is adjusted through iterative testing and player feedback. The float variable hp drop rate, which has a range of 0-0.9, affects the dropout layer drop rate of the discriminator.

2. **Batch norm:** A popular method for ensuring training stability and convergence in GAN models, especially in complicated architectures, is batch normalization. The issue complexity, dataset characteristics, and intended GAN model performance all influence the batch normalization option. The Boolean variable batch norm controls the layers of batch normalization.
3. **Activation function:** In GAN models, the activation function is essential since it establishes the nonlinearity of each neuron's output range. ReLU uses the activation function specified by string variables, which minimizes the vanishing gradient problem and speeds up the convergence of the model.

5.5.3 Training Procedure

The training process involves several key steps to ensure the generator and discriminator networks learn effectively from the data:

1. **Data Preparation:** Split the data into training and validation sets and normalize the input data to $[-1,1]$ range.
2. **Generator Training:**
 - Use a Gaussian random number sample as input.
 - Apply batch normalization and ReLU activation functions.
 - Generate images through a series of deconvolutional layers.
3. **Discriminator Training:**
 - Use stride in convolutional layers for down sampling.
 - Apply deterministic spatial pooling functions.
 - Flatten the convolutional layers and feed into the output layer.

4. **Optimization:**

- Use LR (learning rate):0.0002,optimizer: Adam, and momentum (beta) :0.5.
- Apply a 50% dropout rate during training to prevent overfitting.
- Use label smoothing and low random noise to enhance image quality.

5. **Loss Functions:**

- Train the GAN using the Wasserstein loss function to ensure stability and improve training consistency.

6. **Evaluation Metrics:**

- Measure performance using Inception Score (IS), Structural Similarity Index Measure (SSIM), Fréchet Inception Distance (FID), and total computational time.

5.5.4Layers Architecture

The generator and discriminator architectures of the GAN frequently employ convolutional neural networks (CNN) and deconvolutional networks, respectively. The layered architectures define the structure of the networks' discriminator and generator, with each having a slightly unique design to enhance the ensemble's performance. Figures 5.7 show the layered architecture for the generator and each of the three discriminators used in the EWG model below the specific architecture details for the generator and each of the three discriminators are as follows:

- **Generator Architecture:** Includes layers for convolution, batch normalization, ReLU activation, and tanh activation for the output layer.
- **Discriminator Architectures:** Each discriminator has a slightly different configuration but generally includes layers for convolution, batch normalization, and Leaky ReLU activation.

Generator		
Layer (type)	Output Shape	Param #
Input_1 (InputLayer)	[None, 1, 1, 1281]	0
upsample_block_1 (UpsampleBlock)	[None, 4, 4, 512]	1090912
upsample_block_2 (UpsampleBlock)	[None, 8, 8, 256]	509760
upsample_block_3 (UpsampleBlock)	[None, 16, 16, 128]	54400
upsample_block_4 (UpsampleBlock)	[None, 32, 32, 64]	13568
conv2d_transpose_1 (Conv2DTranspose)	[None, 64, 64, 3]	960
activation (Activation)	[None, 64, 64, 3]	0
Total params: 1,869,984 Trainable params: 1,869,984 non-trainable params: 0		

(a)

Discriminator_1		
Layer (type)	Output Shape	Param #
Input_1 (InputLayer)	[None, 64, 64, 3]	0
conv_layer (ConvLayer)	[None, 32, 32, 64]	3200
conv_layer_1 (ConvLayer)	[None, 16, 16, 128]	131328
conv_layer_2 (ConvLayer)	[None, 8, 8, 256]	524800
conv_layer_3 (ConvLayer)	[None, 4, 4, 512]	209856
conv2d_5 (Conv2D)	[None, 1, 1, 1]	810
Flatten (Flatten)	[None, 5]	0
Total params: 2,765,600 Trainable params: 2,765,600 non-trainable params: 0		

(b)

Discriminator_2		
Layer (type)	Output Shape	Param #
Flatten_1 (Flatten)	[None, 12288]	0
dense (Dense)	[None, 1024]	1259648
dense_1 (Dense)	[None, 256]	262144
dense_2 (Dense)	[None, 128]	32896
dense_3 (Dense)	[None, 64]	8192
dense_4 (Dense)	[None, 32]	2048
dense_5 (Dense)	[None, 1]	32
Total params: 1,586,464 Trainable params: 1,586,464 non-trainable params: 0		

(c)

Discriminator_3		
Layer (type)	Output Shape	Param #
Flatten_1 (Flatten)	[None, 12288]	0
dense (Dense)	[None, 1024]	1259648
dense_1 (Dense)	[None, 256]	262144
dense_2 (Dense)	[None, 128]	32896
dense_3 (Dense)	[None, 64]	8192
dense_4 (Dense)	[None, 32]	2048
dense_5 (Dense)	[None, 1]	32
Total params: 1,586,464 Trainable params: 1,586,464 non-trainable params: 0		

(d)

Figure 5.7(a),(b), (c), and (d): Layered architecture of the Generator, Discriminator 1, Discriminator 2, and Discriminator 3.

By meticulously designing the experimental setup and training procedure, we aim to increment the ability of the proposed model to accurately detect deepfakes, leveraging robust training methodologies and state-of-the-art hardware and software technologies.

5.6 Results and Discussion

The research aims to generate high-quality images by addressing non-convergence, vanishing gradients, and mode collapse during training. The model enhances image diversity to optimize its output. Key evaluation parameters include the Fréchet Inception Distance (FID), Structural Similarity Index Measure (SSIM), Inception Score (IS), and total computational time. The model achieves high accuracy (98.480% and 96.417%) and demonstrates efficient processing times of 1813.251 seconds and 2197.011 seconds for the two datasets, outperforming previous methods. Additionally, the SSIM measure is improved to

compute image similarity, aiding in detecting deepfakes by identifying distortions, artifacts, or inconsistencies produced by the generator. SSIM is crucial for evaluating the resemblance between real and GAN-generated images, with a score of 0.25 indicating a high probability of a deepfake due to noticeable differences. Figure 5.8 illustrates the quality of images over successive epochs, showing that the image quality improves with each epoch, resulting in more recognizable faces.



Figure 5.8: Quality of images generated by EWG model over successive epochs.

In the Indian celebrity picture dataset and the five celebrity image datasets, discriminator D3 outperforms the other two discriminators in terms of accuracy. It operates with three times the efficiency of the others, boosting accuracy from 4% in the initial epochs to 32%. As the dataset size increases, D3's accuracy improves significantly, reaching 98.48% for the Indian celebrity photographs and 96.417% for the five celebrity image datasets as shown in Figure 5.9.

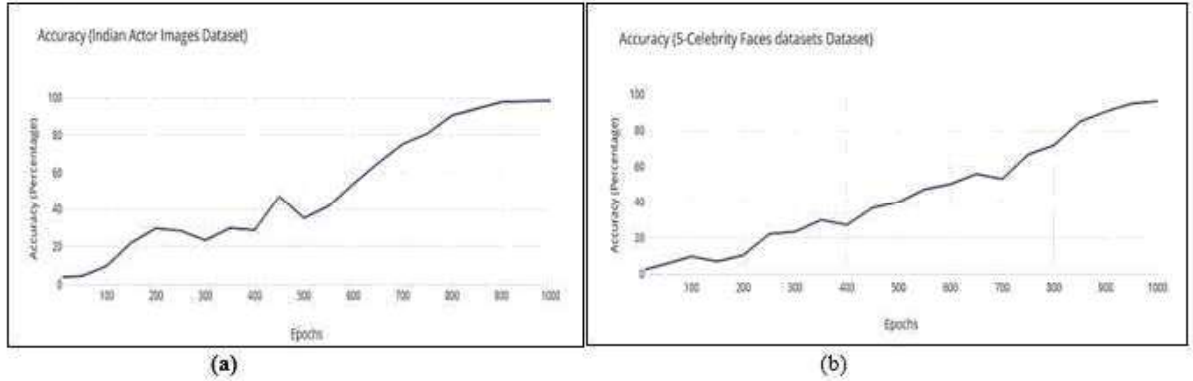


Figure 5.9: Accuracy graphs generated by EWG model for best chosen discriminator (D3) for (a) Indian Actor Images and (b) 5-Celebrity Faces datasets.

The convergence range increases to an upper bound value of 0.4, starting from -0.25, as illustrated in Fig. 5.10. The observed increase in the convergence range to an upper bound value of 0.4, starting from -0.25, as illustrated in Fig. 5.11, indicates a significant improvement in the training stability and performance of the model. This shift suggests that the model is effectively learning and adjusting its parameters, leading to a more stable and consistent convergence. The upper bound value of 0.4 is a positive indicator, as it demonstrates the model's ability to achieve a higher level of accuracy and reliability in its predictions. This value is particularly noteworthy given the initial negative starting point, highlighting the model's capability to overcome initial challenges and progressively refine its performance. Therefore, the convergence range reaching an upper bound of 0.4 can be considered a good value, reflecting the robustness and effectiveness of the model in handling complex datasets and improving the quality of generated images.

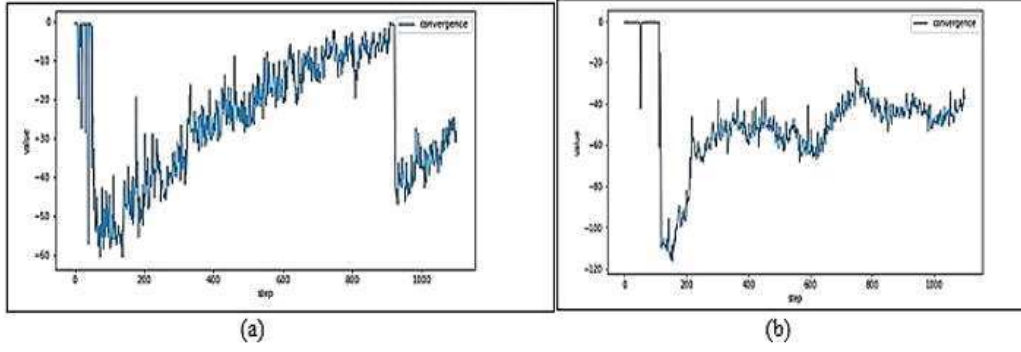


Figure 5.10: Convergence Graphs Generated by Proposed EWG Model for (a) Indian Actor Images and (b) 5-Celebrity Faces Datasets

The Gradient Penalty decreases from 12.5 to 0.5, eventually stabilizing at the final range. This improvement effectively addresses the mode collapse issue, with values dropping from 120 to 0.7 for the Indian celebrity images dataset and from 380 to 0.87 for the 5 Celebrity Image dataset, reaching 0.9, as depicted in Figs. 5.11 and 5.12. These significant values were achieved by the model for both datasets using discriminator three. The substantial decrease in Gradient Penalty from 12.5 to 0.5, which eventually stabilizes, marks a notable advancement in the model's performance.

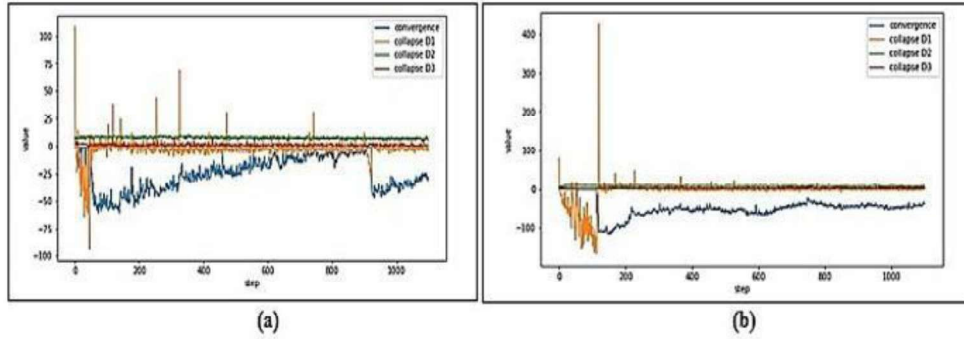


Fig.5.11: Graph showing Mode collapse and Convergence values generated by Proposed EWG model for (a) Indian Actor Images and (b) 5-Celebrity Faces datasets.

Gradient Penalty ensures the stability of Generative Adversarial Networks models during training, particularly in addressing the issue of mode collapse. Mode collapse, a common problem in GANs, occurs when the generator fails to capture the diversity of the data distribution and produces limited variations of samples. The observed reduction in Gradient Penalty indicates that the model is effectively regularizing the gradients, leading to more stable training dynamics and mitigating the risk of mode collapse. This improvement is further underscored by the values achieved for the Indian celebrity images dataset, where the penalty dropped dramatically from 120 to 0.7, showcasing the model's enhanced ability to maintain a diverse set of generated images.

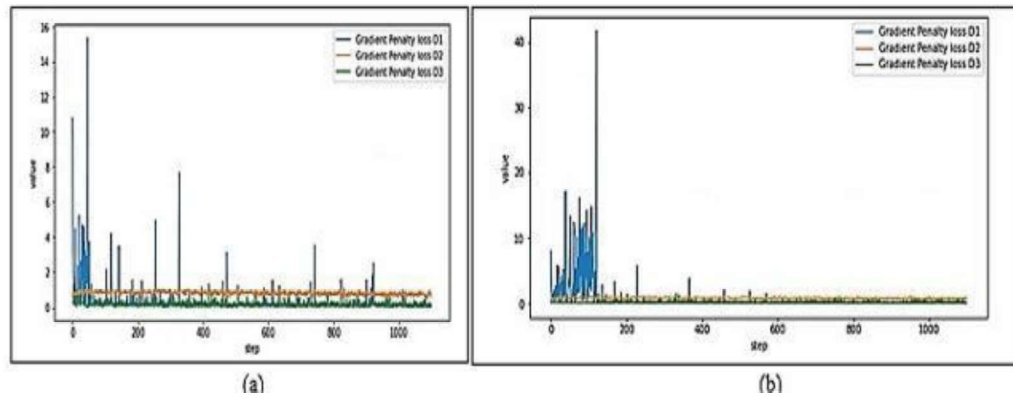


Fig.5.12: Gradient Penalties generated by Proposed EWG model for (a) Indian Actor Images and (b) 5-Celebrity Faces datasets.

For the 5 Celebrity Image dataset, the Gradient Penalty reduction from 380 to 0.87, eventually stabilizing at 0.9, further exemplifies the model's robustness. This decrease signifies a significant enhancement in the model's capability to handle complex and varied data inputs, leading to improved image quality and diversity. The consistency in the lower Gradient Penalty values across both datasets suggests that the model, particularly with the use of discriminator 3, is effectively learning and adapting to different data distributions. This adaptability is crucial for generating high-quality, diverse images and ensuring the GAN's overall effectiveness. The achievement of such significant values in

Gradient Penalty reduction demonstrates the model's superior training mechanism and highlights its potential for broader applications in image generation tasks.

In brief, the vanishing gradient problem in GAN networks, a significant challenge during training, is effectively addressed by our proposed model. It has been achieved using activation functions like ReLU, which promote sparse activations. Leaky ReLU, a variant of ReLU that allows for negative values, is particularly beneficial for the discriminator as it helps maintain the flow of gradients during backpropagation, preventing the gradients from becoming too small. This choice has led to substantial improvements in our model's performance, with significant reductions in Gradient Penalty values, as evidenced by the drop from 12.5 to 0.5 and stabilization. Additionally, mode collapse issues were effectively addressed, with values dropping from 120 to 0.7 for the Indian celebrity images dataset and from 380 to 0.87, stabilizing at 0.9, for the 5 Celebrity Image dataset. These results underscore the efficacy of using Leaky ReLU in the discriminator to enhance training stability and overall model performance.

The graphical analysis of the computational performance, as presented in Figures 5.12 and 5.1, The graphs depict the execution time per epoch across the 1000 epochs, illustrating the model's consistent performance. The slight fluctuations in execution time are within an acceptable range, reinforcing the model's reliability. This analysis confirms that the proposed model not only achieves high accuracy and quality in image generation but also excels in computational efficiency. This balance between performance and efficiency makes the model highly suitable for practical applications where time and resource management are critical.

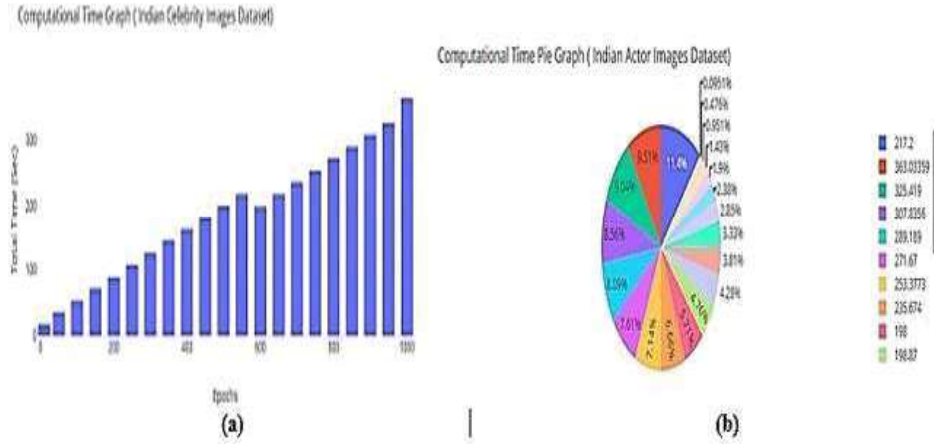


Fig.5.13: Graph showing Computational time Bar Graph (a) and Pie chart (b) generated by the Proposed EWG model using Indian Actor Images dataset.

The computational efficiency of the proposed model was evaluated by measuring the execution time per epoch for both datasets. The findings indicate that the model performs remarkably well in terms of speed. For the Indian celebrity dataset, each epoch executes in approximately 17.274 seconds, demonstrating the model's ability to handle complex computations efficiently. Similarly, for the 5 Celebrity dataset, the initial execution time per epoch is around 18.331 seconds. These times reflect the model's streamlined architecture and optimized processes, which contribute to its rapid execution.

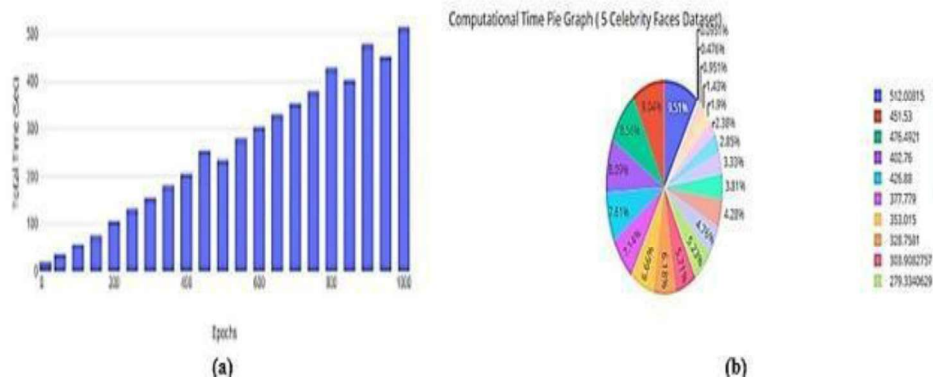


Fig.5.14: Computational time graphs Bar Graph (a) and Pie chart (b) generated by Proposed EWG model using 5-Celebrity Faces.

Over the course of 1000 epochs, the total execution time for the Indian celebrity dataset was recorded at approximately 1813.251 seconds. This showcases the model's ability to maintain consistent performance across many iterations, highlighting its robustness and stability. For the 5 Celebrity dataset, the total execution time was slightly higher, at around 2197.011 seconds. This minor increase in time can be attributed to the additional computational complexity and the larger size of the 5 Celebrity dataset. Despite this, the model's performance remains commendably efficient, ensuring that even extensive datasets can be processed in a reasonable timeframe. The Structural Similarity Index (SSIM) serves as a vital visual metric in assessing the quality of images, particularly in the realm of image processing where resolution loss is a common issue. SSIM measures the perceived changes in structural information, making it highly effective in comparing real images against those generated by a GAN. The SSIM values achieved for the Indian celebrity dataset and the 5 Celebrity dataset were 0.25 and 0.3, respectively. These values suggest that the model employs a highly effective reconstruction technique, ensuring that the generated images maintain a high degree of fidelity to the original ones. Figure 5.15 clearly illustrates this optimal reconstruction quality, demonstrating the model's ability to produce visually compelling images.

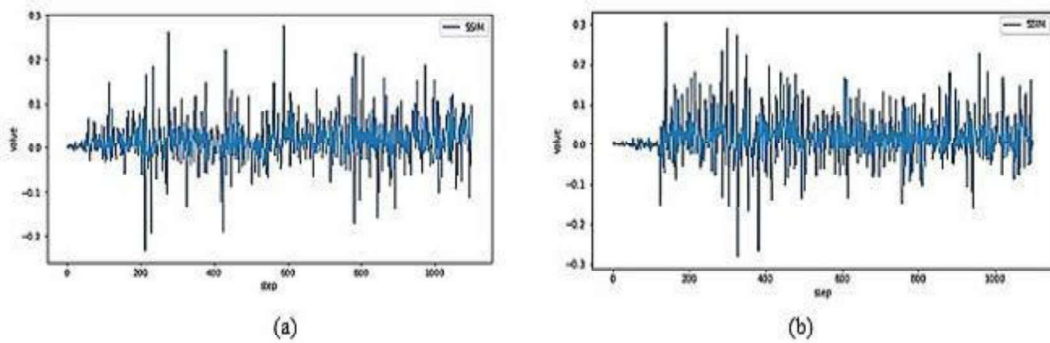


Fig.5.15: Graphs showing SSIM values generated by Proposed EWG model for (a) Indian Actor Images and (b) 5-Celebrity Faces datasets.

In addition to its role in image quality assessment, SSIM proves to be a powerful tool for detecting Deepfake images. By providing a quantitative measure of similarity, SSIM helps in distinguishing between authentic and artificially generated images. The model's ability to achieve an SSIM score of 0.25 for the Indian celebrity dataset indicates a high level of similarity between the generated images and the real ones. This high SSIM score is crucial for applications where the authenticity of images is paramount, such as in security and media. Figure 5.16 shows the comparative analysis, highlighting the efficacy of SSIM in detecting discrepancies and ensuring that the generated images closely resemble the genuine ones.

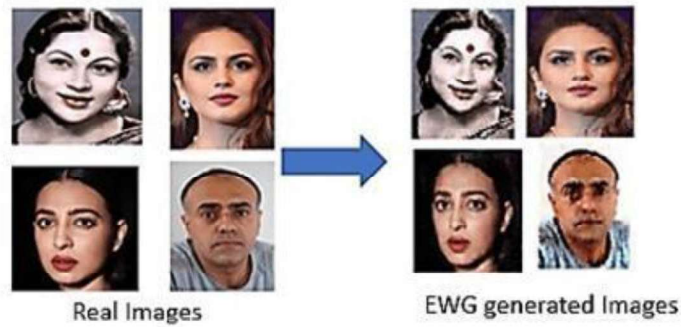


Fig. 5.16: Graph showing (a) Real Image used for training EWG Model(b) Generated Images by the EWG model.

Furthermore, the use of SSIM as a state-of-the-art parameter underscores its importance in modern GAN research. A decent score of 0.25, as achieved in this study, not only reflects the high quality of the generated images but also showcases the model's robustness in maintaining image integrity. This quantitative metric is invaluable for automatic evaluation, offering a reliable measure of perceived quality without requiring human intervention. The integration of SSIM into the evaluation framework enhances the model's

credibility and ensures that the generated images can be used confidently in real-world applications. As demonstrated in the graphical representations, SSIM serves as a benchmark for assessing and improving GAN performance, paving the way for further advancements in the field of image generation.

For better insights, Table 5.4 shows, the findings for Fréchet Inception Distance (FID), Structural Similarity Index Measure (SSIM), Inception Score (IS), and total computational parameters are compared between the proposed model and an existing WGAN model. The evaluation is conducted on two datasets: Indian Actor Images and 5 Famous Celebrities. In the case of the EWG model on the Indian Actor Images dataset, provides the IS score.10.4, FID at 32.71, SSIM at 0.25 total computational time of 1813.251 seconds. By contrast, on the 5 Celebrity Faces dataset the EWG model ranked IS with 11.5, FID at 37.52, SSIM at 0.3, 2197.011 being the computational time. In contrast, the evaluated Basic WGAN Model shows the results on the Indians Actor Images and 5 Celebrity Faces datasets with lower scores in all parameters, as well as significantly more time for total computations.

Table 5.4: Showing the Comparative Scores of IS, FID, SSIM and Total Computational Time Parameters of EWG Model with Existing WGAN Model

Evaluation Parameters	Proposed Model * (Indian Actor Images dataset)	Proposed Model * (5 Celebrity Faces datasets)	WGAN Model (Indian Actor Images dataset)	WGAN Model (5 Celebrity Faces dataset)
IS	10.4	11.5	7.06	7.18
FID	32.71	37.52	54.46	55.90
SSIM	0.25	0.3	0.0313	0.00454

Total	1813.251	2197.011	4970.271	4950.913
Computational time (Seconds)				

Table 5.5 shows the EWG model score on accuracy in comparison with the other GAN models through different research datasets. The achieved accuracy level for EWG* model on both dataset is 98.480% and 96.417% respectively. Consistently, with the CIFAR10 and MNIST datasets, the NFO SCC-GAN attains accuracy scores of 69.51% and 98.90% respectively. In addition, the exact results for the CELEB A dataset show the EWG model gives accuracy scores of 93.27% for Bald, 99.88% for Eyeglasses, 95.68% for Mustache, 94.62% for Wearing Hat, 98.62% for Wearing Necktie. Besides the fact that CIFAR10 and CIFAR100 datasets demonstrate an accuracy score of 65.0/58.2% and 3.5/2.4% respectively.

Table 5.5: Showing the Comparative Scores of Accuracy Parameter of EWG Model (*) with Existing GAN Model

Research	Dataset	Accuracy Score
EWG* Model	Indian celebrity Faces and 5-celebrity Faces	98.480%(Indian celebrity Faces) 96.417% (5-celebrity Faces)
Benny et al. [140]	CIFAR10 MNIST	69.51% 98.90%
Kinakh et al. [141] (NFO SCC-GAN)	CELEB A	93.27% (Bald) 99.88% (Eyeglasses) 95.68% (Mustache) 94.62% (Wearing Hat) 98.62% (Wearing Necktie)
Castro et al. [142]	CIFAR10 CIFAR100	65.0/58.2 (in %) 3.5/2.4 (in %)

Table 5.6 represents evaluation of and Fréchet Inception Distance (FID), Inception Score (IS) values for the ESG model and different GAN models on

different datasets. The IS Scores at 10.4 and 11.5 are presented while applying the EWG* Model for the Indian Actor Images and 5 Celebrity Faces datasets, respectively, where the corresponding FID Scores are 32.71 and 37.52. While the Deep Convolution GAN reports an IS score of 1.074 and FID of 49.3 for the Celeb A data set, the DC-GAN's IS score is 7.06 and 6.87, with the FID scores of 42.23 and 49.18 for CIFAR 10 and CIFAR 100. FC-GAN gave an IS score of 6.41 and FID of 42.6 to CIFAR 10, whereas S-GAN has an IS score of 25.99 ± 4.26 for the CUB-200-2011 dataset. Last of all, the BIG GAN is evaluated with the IS score of 1.94 and the FID of 50.88 on the JFT-300M dataset.

Table 5.6: Showing the Comparative Scores of IS and FID Parameters of the suggested model

GAN Variants	Dataset	IS	FID
EWG Model	Indian Actor Images dataset 5 Celebrity Faces dataset	10.4 11.5	32.71 37.52
Deep Convolution GAN [143]	Celeb A	1.074	49.3
DC-GAN [144]	CIFAR 10 CIFAR 100	7.06 6.87	42.23 44.18
FC-GAN [145]	CIFAR 10	6.41	42.6
S-GAN [146]	CUB-200-2011	-	25.99 ± 4.26
BIG GAN [147]	JFT-300M	1.94	50.88

5.7 Conclusion and Future Scope

In conclusion, this research offers the Ensemble WGAN (EWG), a new model of WGAN to enhance deepfake detection and solve the training problems in the GANs. Through the merging of Wasserstein GAN (WGAN) with three discriminators and one generator, EWG, automates the choice of the best discriminator by using a variety of loss functions, hence, it increases the performance and the deepfake detection accuracy. The EWG model beats the present GAN models by creating better and sharpen images and at the same time, the GAN pathologies are suppressed.

In the case of the future work, research should be directed at the improvement of the diversity in the generated images to be able to notice the new forgery techniques. The development of real-time deepfake detection techniques is necessary to have the possibility of a fast response to the threats that are appearing on social networks. The EWG model needs to be tested on its robustness to check its resistance to adversarial attacks and the new deepfake techniques. Besides, the extension of the EWG and models like the one to other domains than image and video processing, e. g. text generation and voice synthesis, turns out to be a very interesting field of research for the future.

CHAPTER- 6

VOTSTACK: Ensemble Approach for classification of GAN-Generated Deepfake Images

Aim:

This chapter aims to develop a sophisticated deepfake detection model leveraging ensemble machine learning techniques. The proposed model, VOTSTACK, employs a combination of stacking and voting ensemble approaches, utilizing classifiers such as Support Vector Machine (SVM), Logistic Regression, and Decision Tree within a Principal Component Analysis (PCA) framework for dimensionality reduction. This model seeks to enhance the accuracy and reliability of deepfake detection systems, surpassing the performance of existing models discussed in the previous chapters. Additionally, it aims to implement a parallel version of the ensemble GAN model proposed in Chapter 4 and Chapter 5, which utilized multiple discriminators and max voting techniques. This parallel version aims to provide a more versatile and comprehensive approach to deep fake detection by incorporating machine learning classifiers and is used for making a comparison of approaches between CNN-based and conventional machine learning-based classifiers.

Objectives:

- **To introduce a novel ensemble model integrating machine learning classifiers:** Develop a robust detection model using a combination of SVM, Logistic Regression, and Decision Tree classifiers to improve the precision and reliability of deepfake detection.

- **To implement a dual-phase architecture with stacking and voting ensembles:** Utilize a two-phase system where the first phase employs a voting ensemble approach, and the second phase aggregates results using a stacking technique for final classification.
- **To leverage Principal Component Analysis (PCA) for feature selection:** Employ PCA as a dimensionality reduction method to streamline data preprocessing, enhance classifier performance, and reduce computational complexity.
- **To evaluate and compare the model's performance:** Assess VOTSTACK against baseline models using metrics such as accuracy, precision, recall, F1-Score, and total computational time to demonstrate its effectiveness.
- **To enhance the stability and reliability of deepfake detection:** Ensure the model's robustness and applicability in real-world scenarios by addressing common issues in deepfake detection, such as false positives and computational efficiency.
- **To contribute to digital trust and security:** Develop a state-of-the-art detection system that strengthens digital security measures, helping mitigate the risks associated with malicious deepfake usage on social media and other platforms.

By achieving these objectives, Chapter 6 aims to present a comprehensive and innovative approach to deepfake detection, building on the foundations laid in previous chapters and pushing the boundaries of what is possible with current technologies.

6.1 Introduction

The technology in which a person's image or video has been digitally altered and replaced with that of another person is referred to as "Deepfake." This technology produces both positive and negative applications for our society. On the positive

side, it can be used for artistic and entertaining purposes, such as creating 3D-game environments. On the negative side, it can be exploited for malicious activities like spreading false information, fabricating evidence, and influencing public opinion. As deepfake technology evolves, robust detection methods are needed to mitigate risks. Recent corruption has raised concerns about the spread of false information and identity theft using deepfakes as shown in Figure 6.1.



Figure 6.1: Example of Deepfake Image [148]

In 2018, a manipulated video of Barack Obama making statements he never said further showcased the detrimental applications of deepfakes[149]. These harmful uses of deepfake technology have the potential to negatively affect our society by spreading false information, particularly on social media. The advent of Generative Adversarial Networks (GANs) has made it increasingly difficult to combat deepfakes due to their powerful media recreation capabilities, which generate fake images that look real and indistinguishable from genuine ones. Determining the source and reliability of digital media has thus become an urgent need to fight against the misuse of deepfake technology shown in Figure 6.2 below.

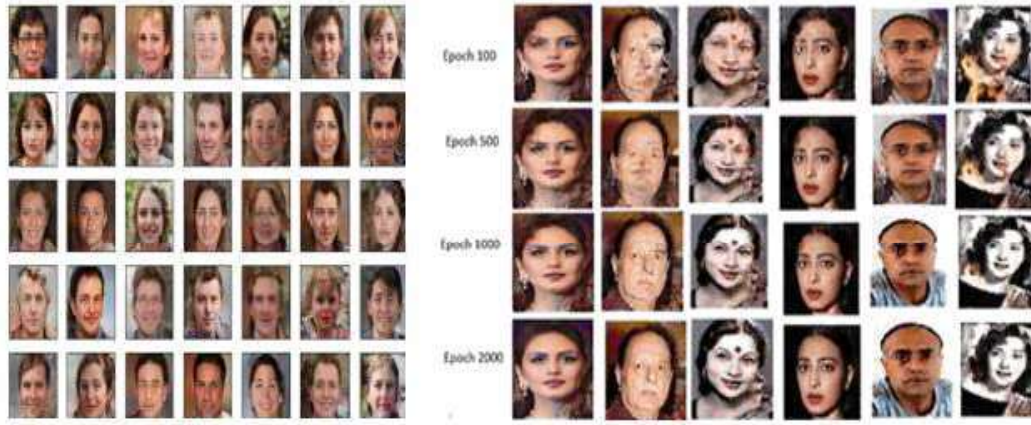


Figure 6.2:GAN-generated Images Example [150]

Researchers have developed several strategies in recent years to identify deepfakes using machine learning algorithms. One promising approach to address this issue is the use of ensemble learning techniques, which integrate the output of various detection models to increase accuracy. The goal of ensemble learning (EL) is to solve a particular problem by combining several approaches rather than relying on a single model. This tactic aims to produce a variety of unique solutions to achieve more accurate predictions. Ensemble learning has been effectively used in several fields, including computer vision, speech recognition, and natural language processing.

Leveraging the power of ensembles and combining machine and deep learning approaches, this chapter presents VOTSTACK, an innovative ensemble model designed to combat the proliferation of deepfake images on social media. VOTSTACK utilizes a blended approach that combines Voting and Stacking ensemble techniques. It leverages the collective intelligence of three different classifiers—Decision Tree, Logistic Regression, and SVM—executing a hybrid feature selection method with Principal Component Analysis (PCA) conditioning as the preprocessing framework. This model operates through a two-phase architecture, with the first phase consolidating results using a voting ensemble and the second phase aggregating collective knowledge into a final decision using a stacking ensemble.

In Chapters 4 and 5, we introduced advanced GAN models to enhance deepfake detection and generation. Chapter 4 detailed the creation of an ensemble GAN model employing three discriminators and one generator using a max-voting technique. This approach aimed to address common GAN challenges such as mode collapse and convergence issues, thereby improving the overall accuracy and reliability of detection systems. Chapter 5 further built on this by integrating Wasserstein GANs (WGANs) and incorporating Wasserstein loss to enhance image generation quality and stability. The ensemble WGAN (EWGAN) model demonstrated superior performance in generating realistic images and detecting deepfakes, surpassing the initial ensemble GAN model.

While promising, the credibility of these models necessitates further validation through comparative analysis using alternative methodologies. In this chapter, we propose a parallel version of the models discussed in the previous chapters, leveraging ensemble machine-learning techniques instead of deep-learning CNNs. By implementing the VOTSTACK model, we aim to test the reliability and accuracy of the previously proposed models under different conditions and methodologies. This parallel approach not only serves as a validation tool but also explores the potential of machine learning classifiers in deepfake detection, using the same max-voting technique employed in the models discussed in Chapters 4 and 5.

It has been observed that VOTSTACK's multi-level ensemble design maximizes deepfake detection accuracy over conventional methods. Empirical testing on a Real and Fake Images dataset demonstrates VOTSTACK's superior performance, achieving an impressive 91.6% accuracy, 90.3% precision, 89.8% recall, and an exceptional F1-Score of 90%. Furthermore, VOTSTACK serves as a benchmark for evaluating deepfake detection, outperforming conventional methods. This chapter aims to enhance the understanding and application of ensemble machine learning models in the context of deepfake detection, providing a robust solution for safeguarding digital trust and security.

VOTSTACK proves to be a state-of-the-art technique against individual baseline machine-learning models, establishing itself as a reliable approach to combating the malicious use of deepfake technology.

6.2 Related Research Work

In this research **Naskar *et al.*, [2024]** solved the issue of deepfake identification by spotting video sequences with deepfake forgeries. By combining characteristics from two well-known deep learning models—Xception and EfficientNet-B7—in a stacking-based ensemble technique, this study presents a solution to the challenge. Afterwards, a meta-learner known as a multi-layer perceptron is utilized to classify authentic and fabricated videos by selecting a subset of features that is nearly optimal through a ranking-based approach. As demonstrated by the outcomes of our experiments, the meta-learning model achieved a higher level of accuracy (96.33 percent on the Celeb-DF (V2) dataset and 98 percent on the Face Forensics dataset) than the individual base models. To verify a robustness of the present methodology, a multitude of experiments have been performed [151].

In this study **Rafique *et al.*, [2023]** a method was suggested for the automated classification of deep fake images through the utilization of ML and DL methodologies. Traditional ML systems that rely on handmade feature extraction are inadequate for capturing intricate patterns that are either inadequately comprehended or uncomplicated to represent with simple features. These systems have limited ability to generalize to unknown data. Additionally, these systems are susceptible to performance degradation due to their sensitivity to noise or fluctuations in the data. Thus, their utility in real-world applications where the data is continuously evolving may be constrained by these issues. Error Level Analysis is performed initially on the image by the proposed framework to ascertain whether the image has been altered. The resulting image is subsequently fed into CNNs after extracting deep features. The feature vectors that are obtained after optimizing the hyperparameters are categorized utilizing SVM and KNN. A

procedure under consideration attained its maximum accuracy of 89.5% through the utilization of Residual Network and KNN. The outcomes validate the effectiveness and resilience of the suggested methodology; consequently, it can be employed to identify profoundly fabricated images and mitigate the potential peril posed by slander and propaganda. [152]

In this study **Kosarkar, Sarkarkar and Gedam, [2023]** created a customized CNN algorithm to detect deepfake images from a collection of videos, then evaluated it against two other approaches to see which worked better. Our model was both trained and evaluated using the Kaggle dataset. This study employed CNNs to distinguish between authentic and deepfake images through the training of three distinct CNN models. In addition, a customized CNN model comprising numerous additional layers, including a dropout layer and MaxPooling-enhanced dense layer, has been devised and implemented. By utilizing a methodology that encompasses face feature extraction, frame extraction, data preprocessing, and classification, it is possible to ascertain whether video images accurately represent their intended content. The performance metrics were employed through the determination of loss, accuracy, and the area under ROC curve. As regards to all other models, the customized CNN obtained an accuracy of 91.4%, a loss value decrease of 0.342, and achieves an AUC of 0.92. Additionally, the CNN gets accuracy tested at 85.2% as against 95.5% for the MLP-CNN model[153].

In this study **kareem and Altaei, [2023]** When a SVM is implemented as a classifier to understand artificial human-like features, the proposed method is better than existing techniques. The preprocessing stage starts with the conversion of the images in RGB to YCbCr followed by the provision of gamma correction. Furthermore, the findings demonstrate that the Canny filter points effectively distinguished facial features from other components. In this course of action, "SVM" with "PCA" and "SVM" without —PCA" were utilized as classifiers for two different classification methods. From the outcomes, it is evident that an accuracy

of the SVM had improved by 96.8% when it was combined with PCA rather than being used alone at a rate of 72.2%.[154].

In this study **Padmashree and Karunkar, [2023]** their deep learning ML and DL systems programmed to spot creation of deepfakes within videos. Detectors using the YOLO-V3 algorithm are utilized to identify people in video and the features that are extracted from the faces are represented with respect to the efficient Net DL model. An ensemble of ML classifiers, including K-NN, SVM, DT, and NB, is employed to detect deepfakes. This ensemble is selected using the max voting method, which offers improved performance on datasets of diverse sizes and resolutions. The proposed method surpasses current modern approaches by achieving an accuracy of 99.64% in experiments involving the integration of the Face Forensics++ (FF++) and Celeb-DF(v2) datasets [155].

In this study **Bray, Johnson and Kleinberg, [2023]** objective was to evaluate the capacity of humans to distinguish deepfake images of human faces (generated at random from the FFHQ dataset using the StyleGAN2 algorithm and consisting of uncensored output) from a collection of non-deepfake images (generated at random from the FFHQ dataset). Additionally, the study aimed to determine the efficacy of a few straightforward interventions designed to enhance detection accuracy. There was a control group and three support interventions administered to the 280 participants via an online survey. There was a pool of 100 photos of human faces, 50 of which were deepfakes and 50 of which were genuine. From this pool, each participant saw 20 photographs at random. Every image was either created by humans or by AI, and participants were asked to score their confidence in each claim and explain their thought process. The detection accuracy remained marginally above chance and was not significantly improved by any of the interventions. Considerable as well was the fact that participants' confidence in the accuracy of their answers was high but uncorrelated with it. Analyzing the outcomes per image demonstrates that participants consistently encountered difficulty accurately labelling specific images while finding others simple to do so. Nevertheless, irrespective of the

image, they reported the same level of high confidence. As a result, while the overall accuracy of the participants was 62%, the accuracy varied considerably between 85% and 30% across the images, with one in every five images achieving an accuracy below 50%. According to their interpretation of the findings, an immediate call to action is required to counter this peril. [156].

In this study **Mitra *et al.*, [2021]** introduced a novel technique for detecting fraudulent videos using neural networks. To reduce the computational load associated with identifying deepfake videos, they implemented a crucial video frame extraction method. The proposed algorithm is accompanied by a model that comprises a CNN and a classifier network. By virtue of its compatibility with our classifier, the Xception network has been selected in comparison to InceptionV3 and Resnet50. A visual artifact-based detection approach is our model. The next classifier network takes the feature vectors produced by the CNN module and uses them to categorize the video signals. To find the most effective model, they used the datasets from Face Forensics++ and Deepfake Detection Challenge. At reduced computational cost and with extremely high accuracy, our algorithm identifies deepfake videos on social media that have been heavily compressed. The accuracy rates they reached were 92.33% when using the Face Forensics++ dataset alone and 98.5% when using both the Face Forensics++ and Deepfake Detection Challenge datasets together. Our model can identify videos that were made by autoencoders. Almost every fake video containing more than one essential video frame has been identified by our method. With a single crucial video frame, the accuracy outlined here pertains to the detection of a fake video. Individuals will be able to verify the veracity of a video due to the method's convenience. Although not exclusively, our work focuses on resolving the social and economic problems brought on by fake videos on social media. Without using a massive quantity of data to train the model, they can attain the high accuracy. When comparing the key video frame extraction approach to previous efforts, the calculations are greatly reduced. [157].

6.3 Description of Methodology

The VOTSTACK ensemble model methodology encompasses several critical components: Dataset, Hybrid Feature Selection Method, Architecture, and Algorithm. This ensemble approach targets the proliferation of deepfake images on social media. It begins with a hybrid feature selection method followed by Principal Component Analysis (PCA) conditioning, which is used for data preprocessing. The hybrid feature selection method ensures that the most discriminative features are utilized by combining filtering and wrapper methods. Initially, feature filtering techniques rank features based on their relevance using the mutual information method. These selected features are then applied in a wrapper method, iterative feature resolution with cross-validation (RFECV), which evaluates various model performance sub-components and eliminates the least important components. PCA further streamlines data, reduces noise, and aids in dimensionality reduction by converting large 1-D vectors into compact principal components. This prepared data is then utilized by Voting ensemble techniques and stacking ensemble methods for training three base classifiers: Decision Tree, Logistic Regression, and SVM. The goal is to leverage the individual strengths and mitigate the weaknesses of each classifier, thereby enhancing overall model accuracy.

The architecture of the proposed VOTSTACK ensemble model operates through a two-phase system as shown in Figure 6.3. In the first phase, a voting ensemble combines predictions from the three base models (Decision Tree, Logistic Regression, and SVM) using majority voting. This means that each model is individually trained on the training data, and their predictions are aggregated. If two or more models predict a certain class, that class is the final prediction. In the case of ties, the model with the highest confidence score determines the final prediction. This method capitalizes on the advantages of each model, improving overall accuracy and robustness. For implementation, the Real and Fake image dataset, consisting of images generated by GAN models, is used and divided into training and testing sets in a 70:30 ratio.

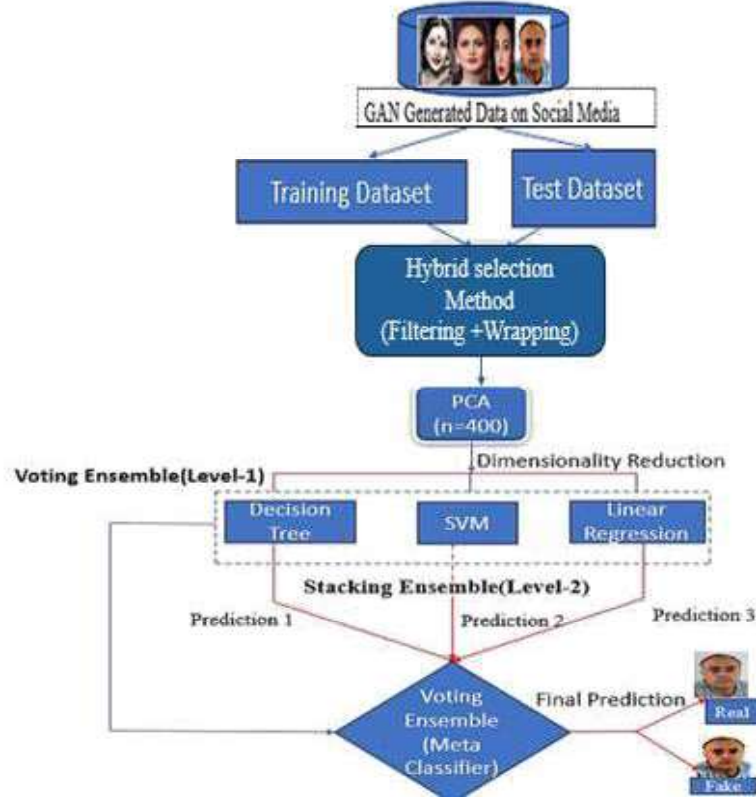


Figure 6.3: Architecture of the proposed VOTSTACK ensemble model operates through a two-phase system.

In the second phase, a stacking ensemble uses the outputs of the voting ensemble from the first phase as input to a meta-classifier. This meta-classifier, or meta-model, is trained on the results obtained from the base classifiers to forecast the ultimate output. The meta-model assigns the final prediction—real or fake—based on the maximum vote received by the three different conventional machine learning models. This multi-level ensemble design enhances the detection accuracy by considering the individual strengths of each model. The detailed architecture of the proposed ensemble approach ensures a robust solution for identifying deepfake images, leveraging the combination of voting and stacking ensembles to provide high accuracy and reliability.

6.3.1 Method

The execution of the VOTSTACK approach is based on different phases of execution based on the collective intelligence Decision Tree, Logistic Regression, and SVM using voting and stacking ensemble – executing within the preprocessing framework comprising the hybrid selection method and Principal Component Analysis (PCA). The comprehensive methodology and the integration of different components are shown in Figure 6.4. Each of these methods is explained in detail in subsequent subsections below:

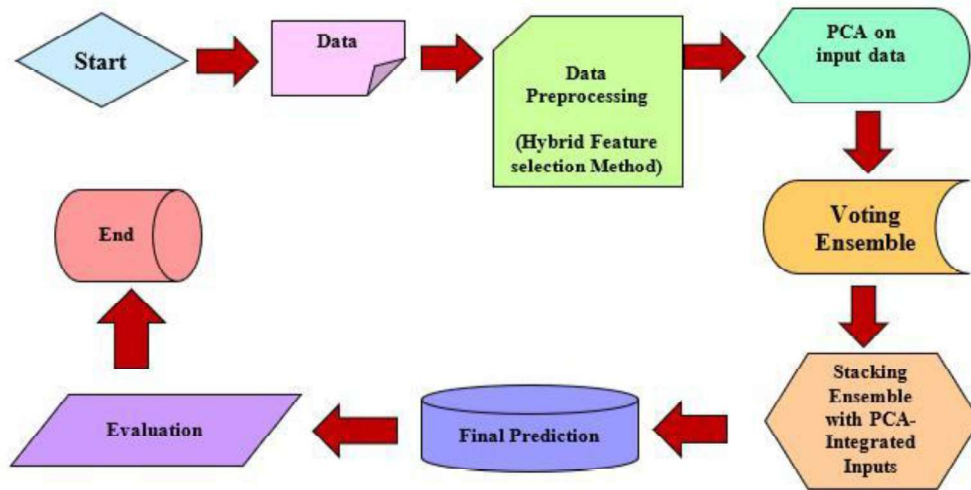


Figure 6.4. The integration of different components of the proposed VOTSTACK ensemble Approach

6.3.1.1 Hybrid Feature Selection Method

A hybrid selection method ensures that the most discriminating features are used. It combines filtering and wrapper methods:

- Filtering methods: Initially, feature filtering techniques such as reciprocity are used to classify features based on their relevance to the target. Rank features using the mutual information method.
- Wrapper method: The filter features are then applied in a wrapper method named iterative feature resolution with cross-validation (RFECV). This

method evaluates various model performance sub-components and eliminates the least important components.

6.3.1.2 Pixel Feature Processing using Principal Component Analysis (PCA) Pixel features, which represent the intensity and color information of each pixel in the image, are first extracted from the resized 256x256 images. This gives us a dataset with many dimensions, where a 65,536-dimensional vector represents each image. Since there's a high number of dimensions and potential redundancy in the pixel data, we use Principal Component Analysis (PCA). It helps to reduce the number of features while keeping the most important information. A multivariate technique, PCA examines a data table containing descriptions of observations by several interdependent variables that are quantitative and intercorrelated. The objective of this process is to derive significant insights from statistical data, which are then represented as a collection of orthogonal variables known as principal components. Spot maps are utilized to illustrate the pattern of similarity among the variables and the observations.

The fundamental concept underlying PCA is to minimize the dimensionality of a dataset comprising numerous interconnected variables while preserving the maximum amount of variation that is inherent in the dataset. This is accomplished through the transformation of the original variables into a new set known as the PCs. The uncorrelated PCs are arranged in such a way that the initial few retain most of the variation that was present in all the original variables. It helps for lowering the dimensionality of huge datasets and boosting interpretability while minimizing information loss. It accomplishes this by generating new uncorrelated variables that successively maximize variance. Finding such new variables, the main components reduce to solving an eigenvalue/eigenvector problem, and the new variables are defined by the dataset at hand, not a priori, making PCA an adaptive data processing technique. It is also adaptive in another way because variations of the approach have been developed that are adapted to diverse data kinds and architectures. It is based on the eigen-

decomposition of positive semi-definite matrices and the singular value decomposition (SVD) of rectangular matrices.

- Mean Subtraction: We adjust each pixel value by subtracting the average pixel value of the dataset.
- Covariance Matrix Calculation: We calculate the covariance matrix of the adjusted dataset to understand the variation and relationships between different pixels.
- Eigen Decomposition: The covariance matrix is broken down to get eigenvalues and eigenvectors.
- Principal Component Selection: Based on the eigenvalues, we select the top principal components (around 400) that capture most of the variation in the data.
- Data Transformation: Using the selected principal components, we transform the original high-dimensional pixel data into a lower-dimensional space.

6.3.1.3 Voting Ensemble (First Level Ensemble)

In the first step, predictions from three basis classifiers (decision tree, SVM, and logistic regression) are electively combined using a majority voting approach. Each model is trained on the training data and generates predictions on unseen data. These individual predictions are then aggregated using the max voting approach (hard voting) as a meta classifier to produce the prediction for the next level called stacking Ensemble. If two or more models predict a certain class, then the final prediction is that class. If there is a tie between the predictions, the prediction of the model with the highest confidence score is taken as the final prediction. This approach is known to improve the overall accuracy of the models by taking advantage of the strengths of each model. The second level employs a stacking ensemble by employing a voting ensemble defined in the first level as a meta-classifier or meta-model.

- Majority vote: Each classifier independently predicts a class label for a given input. Final class scores are determined by the majority opinion of

the classifiers. If two or more classifiers agreed on a class label, that label was chosen as the final prediction. If equal, the prediction with the highest confidence score is selected.

6.3.1.4 Stacking Ensemble (Second Level Ensemble)

In the second phase, the stacking team further adjusts the classification by receiving predictions from the voting ensemble (First level). There are specific features of these classifiers that make them suitable for the final classification task. Decision trees use hyper-rectangles derived in the input space and SVM uses the kernel method to address non-linear problems. When dealing with collinearity, decision trees are better suited for categorical data. Logistic regression increases the posterior class probability, whereas SVM tries to maximize the distance between the closest support vectors. SVM is deterministic, and LR uses probabilistic approach (however we can use the Platts model for probability score). SVM is more rapid in the kernel space. Also, Decision Trees, divide the space into progressively smaller sections and Logistic Regression fits a single line to precisely divide the space into two. These lines would logically generalize to planes and hyperplanes for data with higher dimensions.

- Meta-model training: The results of the voting process are used as input to the meta-model. This meta-model learns to combine these predictions to improve the final accuracy.
- Final Prediction: The meta-model aggregates predictions from the base classifications, and considers their strengths and weaknesses, to derive a final classification. The meta-model is trained to predict the final output. It assigned the final prediction of the image as real, or fake based on the maximum vote received by three different conventional machine learning models.

6.3.2 Algorithm

Table 6.1: Algorithm used for VOTSTACK ensemble method

Algorithm: VOTSTACK ensemble method for hybrid selection

Input: dataset X with n observations and m attributes, target variable Y , test dataset X_{test}

Step 1: Hybrid Feature selection technique

- 1.1 Filter Method: Rank features using mutually related information.
- 1.2 Identify top k features.
- 1.3 Wrapper Method: Use RFECV to the top k features.
- 1.4 Form X_{selected} with selected features.

Step 2: Principal component analysis (PCA).

- 2.1 Standardize X_{selected}
- 2.2 Calculate Covariance Matrix Statistics C
- 2.3 Perform the eigen decomposition
- 2.4 Select the top 400 eigenvectors
- 2.5 Change X_{selected} to X' .

Step 3: Train the base classifiers

- 3.1 Define basic classifiers: SVM, Linear Regression, Decision Tree
- 3.2 Train each base classifier on X' .

Step 4: Voting Ensemble (Phase 1).

- 4.1 Initialize voting classifier with base classifiers
- 4.2 Determine the voting classifier on X' .
- 4.3 Make predictions with a voting classifier for test samples using a majority voting approach.

Step 5: Generate meta feature.

- 5.1 Initialize the meta-feature matrix for training.
- 5.2 Initialize the meta-feature matrix for the test data.

Step 6: Stacking Ensemble (Phase 2)

- 6.1 Define meta-classifiers (Voting Classifier from Phase 1).
- 6.2 Train meta-classifiers on meta-features

Step 7: Prediction with Stacking Ensemble

- 7.1 Final prediction using meta-classification of meta-features of test
-

data.

Output: Final prediction of test data as real or fake image with Accuracy, Precision, Recall and F1-score.

6.3.3 Dataset

For this study, we also used the Real and Fake Face Detection dataset from Kaggle [158]. The dataset contains 1,001 real face images and 1,001 fake face images generated by a Generative Adversarial Network (GAN). The fake face images are generated by training the GAN on the real face images, and then using the trained GAN to generate new fake face images. The dataset is meticulously organized into training, validation, and test sets with all images resized to 256 pixels to facilitate convenient processing and analysis. The real face images are obtained from the Labeled Faces in the Wild (LFW) dataset, which is a benchmark dataset for face recognition. The fake face images are generated using the StyleGAN2 model, which is a state-of-the-art GAN for generating high-quality images. Real and Fake images of human faces found in this dataset are shown in **Figure 6.5**. This dataset serves as a reliable source offering a balanced combination for evaluation of deepfake detection models. It reflects the complexity and diversity of real-world scenarios which is considered very important for advancing research in deepfake domain.



Figure 6.5:Description of Dataset

6.3.4 Evaluation

The establishment of a reliable ML model relies heavily on performance evaluation. To make sure the prediction model works effectively with both known and unknown data, it must be evaluated. Assessing how well a model performs when applied to new, non-sample data is what performance evaluations are all about.

6.3.4.1 Confusion Matrix

The results of predictions produced by a binary classifier are represented in the form of a confusion matrix in tabular format shown in Figure 6.6. The primary objective of this evaluation is to assess the performance of the classification model using a predetermined set of test data, in which the actual values exist. The columns of the matrix represent the outcomes, while the rows of the matrix correspond to the labels utilized in the training dataset.

		Condition	
		Condition positive	Condition negative
Test Outcome	Test Positive	Ture Positive (TP)	False Negative (FN)
	Test negative	False Positive (FP)	Ture Negative (TN)

Figure 6.6: Example of confusion matrix [159]

6.3.4.2 Classification Accuracy

The accuracy measure, which is one of the easiest Classification metrics to use, is defined as the ratio of accurate predictions to total predictions.

$$= \frac{TP + TN}{TP + TN + FN + FP} \dots (6.1)$$

$$TP + TN$$

6.3.4.3 Precision

The precision metric is used to compensate for the accuracy constraint. The precision of a positive forecast indicates how many of them were right. It may be determined by comparing the True Positive or true forecasts to the overall positive predictions.

$$= \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad \text{.....(6.2)}$$

6.3.4.4 Recall

Its objective is to determine the proportion of true positives that were erroneously identified, which is comparable to the Precision metric. True Positives, which represent the proportion of accurate positive or false negative predictions in relation to the overall number of positive predictions, can be computed.

$$= \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad \text{.....(6.3)}$$

6.3.4.5 F1-score

The metric known as the F-score, or F1 Score is utilized to assess the performance of a binary classification model by considering the predictions generated for the positive class. It is computed utilizing the Precision and Recall functions. This form of singular score serves as an indicator for both Precision and Recall. The F1 Score can be determined by taking the harmonic mean of precision and recall, with each component being assigned an equal weight.

$$= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad \text{.....(6.4)}$$

6.4 Experimentation

Implementation Environment:

- **Platform:** Jupyter Notebook (Anaconda framework)
- **Languages and Libraries:** Python, TensorFlow, Keras, NumPy, SciPy, scikit-learn, Matplotlib, PyTorch

- **Development Tools:** Google Collaboratory for additional computational resources

Hardware Requirements:

- **Computer:** A high-performance workstation or server
- **CPU:** Modern multi-core CPU (e.g., Intel Core i7 or above)
- **GPU:** CUDA-compatible GPU (e.g., NVIDIA GeForce GTX or RTX series)
- **RAM:** Minimum of 16 GB
- **Storage:** Sufficient space for datasets, model checkpoints, and interim results
- **Operating System:** Compatible with Linux, Windows, macOS

6.4.1 Training

6.4.1.1 Decision Tree (DT)

- **Tree structure:** In the decision tree model, the data is divided into parts based on the most relevant features. Every node represents a decision point, where the node is split into sub-nodes using a feature for the best information gain / Gini impurity reduction.
- **Training:** The decision tree trains by repeatedly splitting the data into two smaller subsets until each of the leaves is pure (contains only one class), or up to a maximum tree depth.
- **Prediction:** Given a new data point, the model moves down a tree from the root node to a leaf node by assessing the value of the data at each node it passes and returns the class label for the final terminal leaf that is reached.

6.4.1.2 Support Vector Machine (SVM)

- **Hyperplane:** The SVM algorithm tries to find the best hyperplane that maximizes the difference between classes in the transformed feature space.
- **Training:** The SVM uses kernel functions (e.g., linear, polynomial, RBF), transforms the data into a high-dimensional space where linear separation is possible and then determines the support vectors, which are the data points closest to the hyperplane.
- **Prediction:** For a new data point, the model determines which side of the hyperplane the point is on, thus classifying it into one of the classes.

6.4.1.3 Logistic Regression (LR)

- **Sigmoid Function:** Logistic regression uses the sigmoid function to model the probability of a binary outcome. The output is probability from 0 to 1 from this function.
- **Training:** Next, the model learns how to predict probabilities, or the predicted labels, by adjusting weights so that the predicted probabilities are as close as possible to the ground truth probabilities.
- **Prediction:** New data is prepared, for which the calculated sum weights of input weights

$$z = \sum_{i=1}^n w_i x_i$$
(), but the sigmoid function is applied to return the probability of belonging to a class or classes or for a classical logistic regression model. This is then fed into a threshold (e.g. 0.5) to decide upon the final class label for the sample.

6.5 Results and Discussion

As per the results shown in the confusion matrix in Figure 6.7 the True Negative (TN) counts 17,950, False Positive (FP) counts 1,936, False Negatives (FN) counts 2,050, and True positives (TP) count 18,064. These values analyze the outcomes of the proposed model using PCA integrated Voting and Stacking Ensemble Approach.

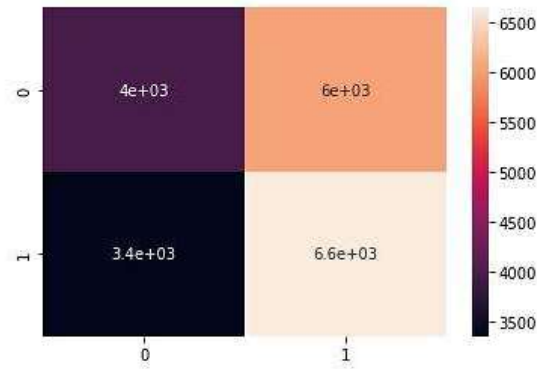


Figure 6.7:Description of Confusion matrix of the VOTSTACK ensemble techniquesachieved as per Real and Fake Images Dataset.

The accuracy defines the overall correctness of the predictions and is defined as, $(TP + TN) / (TP + TN + FP + FN)$. Accuracy in this instance is determined as; $Accuracy = (18,064 + 17,950) / (18,064 + 17,950 + 1,936 + 2,050) = 0.916$. So, the voting stacking ensemble method has a 91.6% accuracy, or about 0.916, accuracy. The second important parameter is Precision: It is the percentage of positive instances that were correctly predicted, out of all the positive instances that were predicted. It is equal to $TP / (TP + FP)$. Calculating the precision with the proposed approach is calculated as; $18,064 / (18,064 + 1,936) = 0.903$. The voting stacking ensemble method has a precision of about 0.903, or 90.3%. The next sensitive parameter that defines the behaviour of the model is Recall (Sensitivity). It quantifies the percentage of positive instances that were correctly predicted out of all positive instances overall.

Mathematically, $Recall = TP / (TP + FN)$ and is calculated in this instance as; $18,064 / (18,064 + 2,050) = 0.898$. So, the proposed approach's recall (sensitivity) is 0.898, or 89.8%. F1-Score signifies the harmonic mean of recall and precision; it provides a fair comparison of the two. F1-Score is equal to $2 * (Precision * Recall) / (Precision + Recall)$. In this situation, the F1-score can be calculated as follows: $F1-Score = 2 * (0.903 * 0.898) / (0.903 + 0.898) = 0.900$. The proposed approach's F1-score is approximately 0.900, or 90.0%. So, finally based on these findings, the stacking approach had an approximate accuracy of

91.6%, precision of 90.3%, recall (sensitivity), of approximately 89.8%, and an approximate F1-score of 90.0%. These metrics show how well the VOTSTACK (Voting + Stacking) ensemble technique performed at classifying the data. The comparison of the proposed techniques with its base classifiers is presented in Table 3 and in Figure 6.8 below.

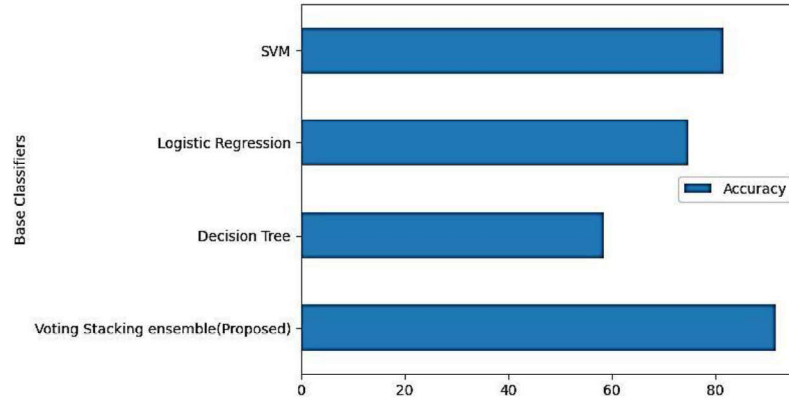


Figure 6.8: Accuracy score of the VOTSTACK ensemble techniques with base classifiers

The VOTSTACK ensemble strategy uses the final prediction to combine the results of three separate models (Decision Tree, Logistic Regression, and SVM). Voting involves aggregating the predictions of these multiple models and deciding based on the majority vote or consensus. Then, stacking employs voting as a meta-model and trains it to learn the optimal combination of the individual models' predictions. It has been observed that individual decision tree marks 58.4% accuracy, Logistic regression achieves 73.7% accuracy, and SVM archives 81.5% accuracy. Most of all our proposed VOTSTACK approach outperforms with 91.6% accuracy. The description of the Accuracy score of the VOTSTACK ensemble techniques with base classifier approaches is presented in Table 6.2.

Table 6.2: Accuracy score of the VOTSTACK ensemble techniques with base classifiers.

Model	Percentage (%)
VOTSTACK	91.6

Logistic regression	73.7
SVM	81.5
Decision Tree	58.4

The comparison of the approach with other existing ensemble approaches is shown in **Figure 6.9** below. Table 6.3 illustrates the values achieved by the different models using different evaluation parameters like Accuracy, precision, Recall and F1-score. The method using the Voting Stacking Ensemble achieves the highest accuracy of 91.6%. This shows that the predictive performance is enhanced when multiple base models are combined using both voting and stacking methods. The accuracy of the voting approach is 91.17%, which is slightly less in comparison with our approach. As a result, it seems possible that the Voting Stacking Ensemble approach would perform slightly better than using just the voting technique without stacking.

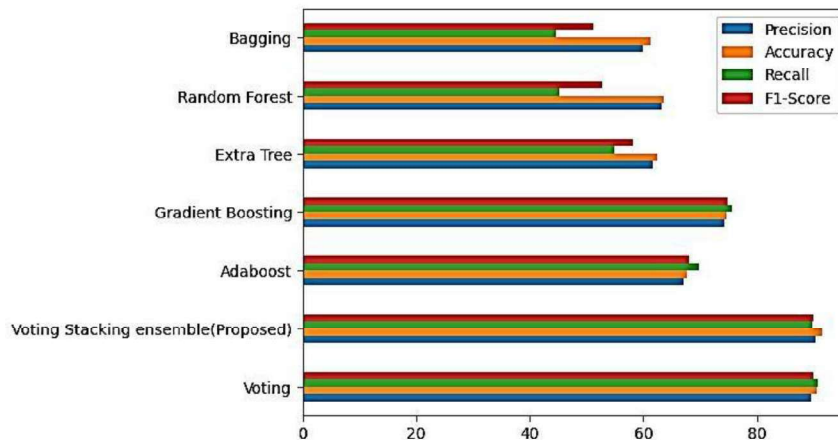


Figure 6.9:Bar graph showing comparative study of VOTSTACK ensemble techniques with different ensemble approaches usingdifferent evaluation parameters (Accuracy, Precision, Recall andF1-score).

An accuracy of 74.72% is attained by the Gradient Boosting model. This shows that applying boosting techniques can enhance performance compared to standalone base models, but it is still inferior to ensemble methods. A 67.6% accuracy rate is attained using the boosting approach. This suggests that using

boosting techniques on their own, independent of ensemble methods, may lead to a modest improvement over using individual base models, but it is still inferior to using ensemble methods. The accuracy rate for the bagging method is 61.31%. Using bagging techniques, like Extra Tree classifier, can offer some improvement over individual models, but it is less than ensemble approaches, according to this. The accuracy of the Extra Tree classifier is 62.42%. Compared to the Bagging Approach, it performs marginally better, but it still falls short of the ensemble approaches. 63.71% accuracy is achieved by the Random Forest. Although it performs marginally better than the Bagging Approach and the Extra Tree classifier, it still falls short of ensemble methods. So, it has been found that in terms of accuracy, the voting stacking ensemble and voting approach models perform better than the individual base models gradient boosting, boosting approach, bagging approach, extra tree classifier, and random forest. This suggests that combining multiple models with ensemble techniques, like voting and stacking, enhances the performance of prediction as shown in Table 4 below.

Table 6.3: Comparison study of the various ensemble techniques with proposed VOTSTACKEnsemble approach.

Ensemble Models	Accuracy	Precision	Recall	F1-Score
VOTSTACK Ensemble (Voting + Stacking) approach	91.6	90.3	89.8	90
Voting Approach	91.17	89.6	90.7	89.9
Gradient Boosting	74.72	74.21	75.76	74.9
Boosting Approach	67.6	67.15	69.89	68.04

Bagging Approach	61.31	60.01	44.78	51.29
Extra Tree classifier	62.42	61.7	54.9	58.14
Random Forest	63.71	63.22	45.13	52.66

The highest level of precision, 90.3%, is achieved by the Voting Stacking Ensemble method as shown in **Figure 6.10**. This shows that a high proportion of correctly predicted positive instances compared to the total number of positively predicted instances are obtained when combining multiple base models using both voting and stacking techniques. The Voting Approach achieves a precision of 89.6%, which is slightly less. This suggests that compared to the Voting Stacking Ensemble approach, using the voting technique alone without stacking might yield a slightly lower precision. The precision of the Gradient Boosting model is 74.21%. This suggests that while using boosting techniques can increase precision when compared to individual base models, ensemble approaches are still more precise. The precision of the Boosting Approach is 67.15%. As a result, it seems possible that using boosting techniques on their own, independent of ensemble methods, could lead to a modest increase in precision compared to using individual base models, though it would still be less than using ensemble methods. The precision of the Bagging Approach is 60.01%. This shows that while bagging techniques, like the Random Forest and Extra Tree classifier, can improve precision over individual models to a certain extent, they do so at a lower rate than ensemble methods.

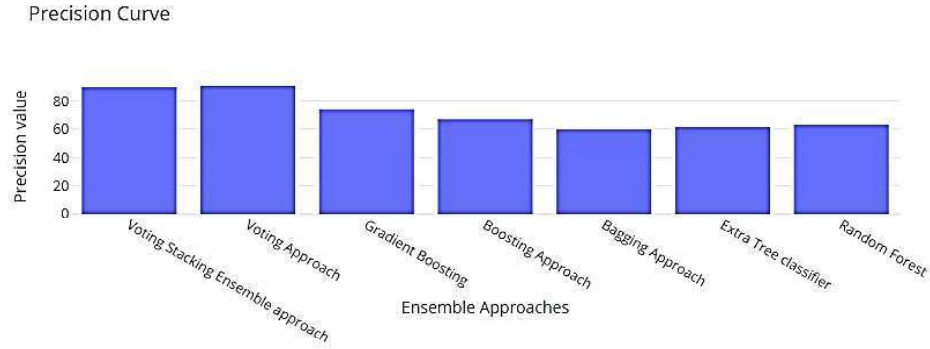


Figure 6.10: Description of Precision curve of VOTSTACK ensemble techniques achieved as per Real and Fake Images Dataset.

Higher values indicate better performance in capturing positive situations. Recall measures the proportion of true positive instances correctly identified, while F1-score provides a balanced measure of precision and recall. Figure 6.11 and Figure 6.12 indicate the respective scores of recalls and F1-score achieved by different models. The Voting Approach successfully recognized 90.7% of the positive instances in this instance, making it a trustworthy model for identifying positive cases. The second-highest recall value was achieved by the Voting Stacking Ensemble method, at 89.8%. It shows that 89.8% of the positive events were accurately detected by this ensemble model. The ensemble method mixes the results of various models, thereby using their advantages and enhancing performance. A recall of 69.89% was attained using the boosting approach. This means that of the positive events, the model accurately predicted 69.89% of them. In comparison to voting-based approaches, it has a lesser recall, yet it still performs well. The Bagging Approach had a recall value of only 44.78%. It suggests that the model only accurately detected 44.78% of the positive events. Despite having a somewhat inferior recall performance, it may still be beneficial in some situations especially when used in conjunction with other models.

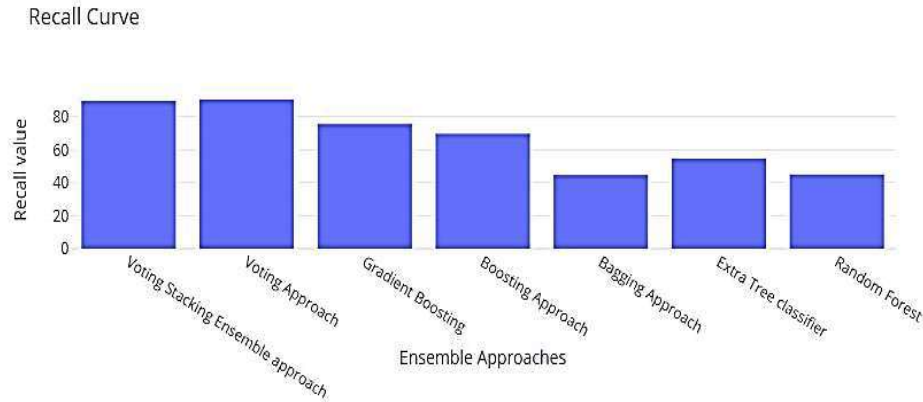


Figure 6.11:Description of Recall curve of VOTSTACK ensemble techniquesachieved as per Real and Fake Images Dataset.

With a recall of 75.76%, the Gradient Boosting model was able to correctly identify 75.76% of the positive examples. It captures a sizable fraction of the positive cases even though its recall is lower than that of voting-based methods. A recall of 63.71% was attained by the Random Forest model. It suggests that 63.71% of the positive events were accurately recognized by the model. It can nevertheless help collect a sizeable fraction of positive cases despite having a lower recall than the prior models. Recall for the Extra Tree classifier was 54.9%. This shows that it accurately recognized 54.9% of the affirmative instances.

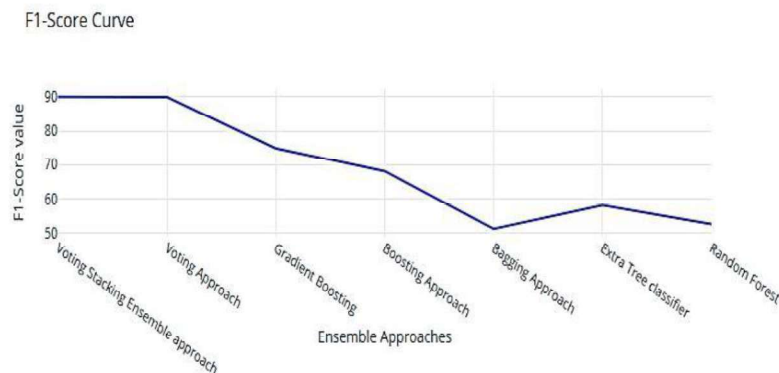


Figure 6.12:Description of F1-Score curve of VOTSTACK ensemble techniques achieved as per Real and Fake Images Dataset.

The Voting Approach obtained the maximum recall of 90.7% and an F1-score of 89.9% based on the provided values. The Voting Stacking Ensemble method also did well, with an F1-score of 90% and a recall of 89.8%. Comparatively, the F1-score and recall for the Gradient Boosting model were both 74.9%. A lower recall of 69.89% and an F1-score of 68.04% were achieved with the boosting approach. The F1-score for the Bagging Approach was 51.29%, with a recall of 44.78%. An F1-score of 58.14% and a recall of 54.9% were achieved using the Extra Tree classifier. The recall and F1 scores for the Random Forest model were both 63.71% and 63.22%, respectively. So, based on the available metrics, we can conclude that the Voting Stacking Ensemble strategy achieved the best accuracy (91.6%), high precision (90.3%), relatively high recall (89.8%), and F1-Score (90%) of all the approaches. The Voting Stacking Ensemble technique, closely followed by the Voting technique, is the best-performing ensemble model. The performance of the Boosting Approach was the lowest in this scenario. It's vital to remember that the ideal model selection also depends on the requirements and situational factors of the current challenge. The comparison of the proposed approach with existing ensemble approaches is discussed in Table 6.3 below.

Table 6.3: A comparison study of the VOTSTACK ensemble approach with various existing ensemble techniques.

Author	Dataset	Approach	Accuracy
Ours	Real and Fake Face Dataset	(Voting + Stacking) approach	91.6%
Mehra et al. [160]	DFDC dataset	LSTM+Xception Net Approach	83.42%
Trabelsi et al.[161]	DFDC dataset	Deep Ensemble Approach	82%
Wang et al. [162]	CELEB DF V2	Ensemble Approach Using Random Forest	70.60%

So, compared with other existing ensemble approaches proposed VOTSTACK ensemble approach achieved the best 91.6% accuracy rate using Real and Fake Dataset. The LSTM + Xception Net method created by Mehra et al. had an accuracy of 83.42% on the DFDC dataset. This shows that the model has an accuracy rate of 83.42% in classifying the videos or images in the DFDC dataset. Wang et al. used an ensemble strategy with Random Forest classifiers on the CELEB DF V2 dataset, attaining an accuracy of 70.60%. So, finally, it is worth saying that the combined voting and stacking methods help to boost prediction accuracy and make the proposed ensemble strategy robust and reliable.

6.6 Conclusion and Future Scope

VOTSTACK Ensemble has been proven as a cutting-edge technique for deepfake identification of fake images. The method is one of a kind utilizing hybrid feature selection based PCA-integrated (Voting +stacking) ensemble techniques that are powerful to deliver accurate categorization of real and false images. Three different baseline classifiers (Decision Tree, Logistic Regression, and SVM), provide a stable foundation for the proposed ensemble's two-phase implementation. The stacking ensemble utilizes the voting ensemble as a meta-classifier and successfully offers accurate prediction scores. The model outperforms the previous ensemble approaches in terms of accuracy (91.6%), precision (90.3%), relatively good recall (89.8%), and F1-Score (90%). In terms of future work, the proposed method can be improved for a higher number of baseline models to gain deep insights into execution. It would be promising to consider other evaluation parameters to gain a more comprehensive understanding of the ensemble models' performance.

CHAPTER- 7

Conclusion and Future Scope

7.1 Conclusion

The research titled "A Deepfakes Detection Model for Face Forgeries Generated by Generic GAN in Social Media" has made significant strides in enhancing the detection of deepfake images by leveraging advanced ensemble-based methodologies. The initial phase of this research involved the development of an ensemble GAN model designed to address prevalent issues such as mode collapse and convergence, which are common in GAN training. In **Chapter 4**, a novel ensemble GAN model was introduced, utilizing multiple discriminators with a max voting technique. This approach effectively addressed critical issues such as mode collapse and convergence, leading to improved detection accuracy and reliability. By integrating these techniques, the model demonstrated substantial improvements in identifying deepfake images, setting a robust foundation for further advancements in the field. This model employed multiple discriminators combined through a max-voting technique, resulting in improved accuracy and reliability in detecting deepfake images. The success of this approach underscored the potential of ensemble methods in enhancing deepfake detection capabilities.

Further advancing this foundation, the research incorporated the Wasserstein loss function within the GAN framework to improve the quality and stability of generated images. Building on this foundation, **Chapter 5** introduced the Ensemble WGAN (EWG) model, which incorporated the Wasserstein loss function to enhance image generation quality and stability. The EWG model's novel architecture, combining multiple discriminators in a voting ensemble, significantly outperformed traditional GAN models in both image synthesis and deepfake detection. This chapter underscored the importance of leveraging

advanced loss functions and ensemble techniques to achieve higher precision and robustness in generative models, highlighting the EWG model's superiority in generating high-quality images and effectively detecting deepfakes.

The culmination of this research was the development of the VOTSTACK model, which represents a significant innovation in the field of deepfake detection in **Chapter 6**. This model was presented, which employed a hybrid feature selection method and Principal Component Analysis (PCA) for preprocessing. This model utilized traditional machine learning classifiers in a two-phase ensemble approach, combining voting and stacking techniques. The VOTSTACK model achieved remarkable performance metrics, demonstrating its efficacy in detecting GAN-generated deepfake images with impressive accuracy, precision, recall, and F1-Score. This comprehensive approach not only validated the effectiveness of ensemble methods in deepfake detection but also provided a scalable and robust solution for real-world applications. Future research should focus on enhancing real-time detection capabilities, cross-domain adaptation, explainable AI, robustness against adversarial attacks, and scalability to further strengthen deepfake detection systems.

Overall, the research findings underscore the critical role of ensemble methods in addressing the challenges posed by deepfake technology. By integrating advanced loss functions, hybrid feature selection, and ensemble techniques, the developed models have shown significant improvements in the detection of GAN-generated deepfake images. These advancements contribute valuable insights and practical solutions to the field of digital forensics and media integrity, offering robust tools for mitigating the risks associated with deepfake technology on social media platforms.

7.2 Future Scope

Looking ahead, future research can build upon these findings by exploring several promising directions. One potential avenue is the further refinement of ensemble

models by incorporating additional machine learning and deep learning techniques, such as neural architecture search and meta-learning, to enhance model performance and adaptability. Additionally, expanding the dataset to include a wider variety of deepfake scenarios and sources can help improve the generalization and robustness of detection models. Finally, developing real-time detection capabilities and integrating these models into social media platforms and digital forensics tools can provide immediate and practical applications, helping to safeguard digital content and maintain the integrity of online information.

Future research should prioritize the enhancement of real-time detection capabilities to address the increasing volume and sophistication of deepfake content. Developing models that can quickly and accurately detect deepfakes in streaming media will be crucial for mitigating their impact in various domains, including social media, news dissemination, and digital security. Additionally, exploring cross-domain adaptation will be essential for creating models that can generalize across distinct types of deepfakes and varying levels of quality, ensuring robustness in diverse real-world scenarios.

Furthermore, scalability remains a key challenge as the volume of digital media content continues to grow. Future research should explore scalable solutions that can manage large-scale data efficiently while maintaining high detection accuracy. Collaboration with social media platforms and digital forensics agencies will be crucial in implementing these models in real-world applications. By focusing on these areas, future research can continue to advance the field of deepfake detection, providing robust, dependable, and scalable solutions to protect digital integrity and security.

Another critical area for future investigation is the development of explainable AI techniques to provide transparency in deepfake detection. Understanding the decision-making process of these models will enhance trust and reliability,

particularly in sensitive applications such as digital forensics and legal proceedings. Moreover, improving robustness against adversarial attacks will be vital to prevent the manipulation of detection systems. Lastly, focusing on the scalability of these models to manage large datasets and integrate seamlessly with existing digital infrastructure will ensure their practical applicability and widespread adoption in combating the threat of deepfakes. In short, the future research could focus on the following areas to build on these findings:

1. **Real-Time Detection:** Developing models capable of real-time deepfake detection to enhance practical applicability in social media monitoring and digital forensics.
2. **Cross-Domain Adaptation:** Enhancing the models' ability to generalize across different types of deepfakes and various media formats.
3. **Explainable AI:** Integrating explainability into deepfake detection models to better understand the decision-making process and improve trust in automated systems.
4. **Robustness Against Adversarial Attacks:** Strengthening models to resist adversarial attacks aimed at bypassing deepfake detection systems.
5. **Scalability:** Ensuring that the proposed models can scale effectively with increasing data volumes and diverse datasets in real-world scenarios.

These directions will not only enhance the robustness and accuracy of deepfake detection systems but also contribute to the broader field of digital media integrity and security.

References

- [1] Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). Deepfakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64, 131-148.
- [2] Han, X., Chen, X., & Liu, L. P. (2021, May). Gan ensemble for anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 5, pp. 4090-4097).
- [3] Feng, D., Lu, X., & Lin, X. (2020). Deep detection for face manipulation. In *Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 18–22, 2020, Proceedings, Part V* 27 (pp. 316-323). Springer International Publishing.
- [4] Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4401-4410).
- [5] Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2223-2232).
- [6] Wang, Z., Wang, J., Kourakos, M., Hoang, N., Lee, H. H., Mathieson, I., & Mathieson, S. (2021). Automatic inference of demographic parameters using generative adversarial networks. *Molecular ecology resources*, 21(8), 2689-2705.
- [7] Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., & Graves, A. (2016). Conditional image generation with pixelcnn decoders. *Advances in neural information processing systems*, 29.
- [8] Welling, M., & Kingma, D. P. (2014). Auto-encoding variational bayes. *CoRR*.

- [9] Hudson, D. A., & Zitnick, L. (2021, July). Generative adversarial transformers. In *International conference on machine learning* (pp. 4487-4499). PMLR.
- [10] Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*.
- [11] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8110-8119).
- [12] Ji, D., Wei, Z., Tian, C., Cai, H., & Zhao, J. (2022). Deep transfer ensemble learning-based diagnostic of lithium-ion battery. *IEEE/CAA Journal of Automatica Sinica*.
- [13] Ohata, E. F., Bezerra, G. M., das Chagas, J. V. S., Neto, A. V. L., Albuquerque, A. B., De Albuquerque, V. H. C., & Reboucas Filho, P. P. (2020). Automatic detection of COVID-19 infection using chest X-ray images through transfer learning. *IEEE/CAA Journal of Automatica Sinica*, 8(1), 239-248.
- [14] Wang, Y., Qiu, S., Li, D., Du, C., Lu, B. L., & He, H. (2022). Multi-modal domain adaptation variational autoencoder for EEG-based emotion recognition. *IEEE/CAA Journal of Automatica Sinica*, 9(9), 1612-1626.
- [15] Wang, G., Qiao, J., Bi, J., Li, W., & Zhou, M. (2018). TL-GDBN: Growing deep belief network with transfer learning. *IEEE Transactions on Automation Science and Engineering*, 16(2), 874-885.

- [16] Cai, G., Wang, Y., He, L., & Zhou, M. (2019). Unsupervised domain adaptation with adversarial residual transform networks. *IEEE transactions on neural networks and learning systems*, 31(8), 3073-3086.
- [17] Yao, S., Kang, Q., Zhou, M., Rawa, M. J., & Abusorrah, A. (2023). A survey of transfer learning for machinery diagnostics and prognostics. *Artificial Intelligence Review*, 56(4), 2871-2922.
- [18] Liu, X., & Chen, X. (2020). A survey of GAN-generated fake faces detection method based on deep learning. *Journal of Information Hiding and Privacy Protection*, 2(2), 87.
- [19] Borji, A. (2022). Pros and cons of GAN evaluation measures: New developments. *Computer Vision and Image Understanding*, 215, 103329.
- [20] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.
- [21] Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.
- [22] Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2019, May). Self-attention generative adversarial networks. In *International conference on machine learning* (pp. 7354-7363). PMLR.
- [23] Han, S., Srivastava, A., Hurwitz, C., Sattigeri, P., & Cox, D. D. (2020). not-so-biggan: Generating high-fidelity images on a small compute budget. *arXiv preprint arXiv:2009.04433*, 2.

- [24] Daras, G., Odena, A., Zhang, H., & Dimakis, A. G. (2020). Your local GAN: Designing two-dimensional local attention mechanisms for generative models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 14531-14539).
- [25] Park, D. K., Yoo, S., Bahng, H., Choo, J., & Park, N. (2018). Megan: Mixture of experts of generative adversarial networks for multimodal image generation. *arXiv preprint arXiv:1805.02481*.
- [26] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.
- [27] Li, D. C., Chen, S. C., Lin, Y. S., & Huang, K. C. (2021). A generative adversarial network structure for learning with small numerical data sets. *Applied Sciences*, 11(22), 10823.
- [28] Miyato, T., Kataoka, T., Koyama, M., & Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*.
- [29] Chen, T., Zhai, X., Ritter, M., Lucic, M., & Houlsby, N. (2019). Self-supervised gans via auxiliary rotation loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12154-12163).
- [30] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.

- [31] Ajder, H., Patrini, G., Cavalli, F., & Cullen, L. (2019). The state of deepfakes: Landscape, threats, and impact. *Amsterdam: Deepttrace*, 27.
- [32] Chesney, R., & Citron, D. (2019). Deepfakes and the new disinformation war: The coming age of post-truth geopolitics. *Foreign Aff.*, 98, 147.
- [33] Paris, B., & Donovan, J. (2019). Deepfakes and cheap fakes.
- [34] Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). Deepfakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64, 131-148.
- [35] Li, Y., & Lyu, S. (2018). Exposing deepfake videos by detecting face warping artifacts. *arXiv preprint arXiv:1811.00656*.
- [36] Mirsky, Y., & Lee, W. (2021). The creation and detection of deepfakes: A survey. *ACM computing surveys (CSUR)*, 54(1), 1-41.
- [37] Dhariwal, P., & Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34, 8780-8794.
- [38] Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., ... & Norouzi, M. (2022). Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35, 36479-36494.
- [39] Nichol, A. Q., & Dhariwal, P. (2021, July). Improved denoising diffusion probabilistic models. In *International conference on machine learning* (pp. 8162-8171). PMLR.

- [40] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33, 6840-6851.
- [41] Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I. (2018, December). Mesonet: a compact facial video forgery detection network. In *2018 IEEE international workshop on information forensics and security (WIFS)* (pp. 1-7). IEEE.
- [42] Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1-11).
- [43] Nguyen, H. H., Yamagishi, J., & Echizen, I. (2019, May). Capsule-forensics: Using capsule networks to detect forged images and videos. In *ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 2307-2311). IEEE.
- [44] Li, Y., Chang, M. C., & Lyu, S. (2018, December). In actu oculi: Exposing ai created fake videos by detecting eye blinking. In *2018 IEEE International workshop on information forensics and security (WIFS)* (pp. 1-7). IEEE.
- [45] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep feedforward networks. *Deep learning*, (1).
- [46] Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, No. 4, p. 738). New York: springer.
- [47] Rosasco, L., De Vito, E., Caponnetto, A., Piana, M., & Verri, A. (2004). Are loss functions all the same? *Neural computation*, 16(5), 1063-1076.

- [48] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980-2988).
- [49] Li, Y., & Lyu, S. (2018). Exposing deepfake videos by detecting face warping artifacts. *arXiv preprint arXiv:1811.00656*.
- [50] Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010: 19th International Conference on Computational Statistics Paris France, August 22-27, 2010, Keynote, Invited and Contributed Papers* (pp. 177-186). Physica-Verlag HD.
- [51] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- [52] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.
- [53] Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.
- [54] Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013, May). On the importance of initialization and momentum in deep learning. In *International conference on machine learning* (pp. 1139-1147). PMLR.
- [55] Reddi, S. J., Kale, S., & Kumar, S. (2019). On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*.

- [56] Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.
- [57] Zakharov, E., Shysheya, A., Burkov, E., & Lempitsky, V. (2019). Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9459-9468).
- [58] Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). Deepfakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64, 131-148.
- [59] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1), 53-65.
- [60] Antoniou, A., Storkey, A., & Edwards, H. (2017). Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*.
- [61] Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., & Webb, R. (2017). Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2107-2116).
- [62] Westerlund, M. (2019). The emergence of deepfake technology: A review. *Technology innovation management review*, 9(11).
- [63] Paris, B., & Donovan, J. (2020). Deepfakes and Cheap Fakes: The Manipulation of Audio and Visual Evidence. Permanent Record.

- [64] UNESCO (2018). Impacts of misinformation spread through social messaging and social media.
- [65] Westerlund, M. (2019). The emergence of deepfake technology: A review. *Technology innovation management review*, 9(11).
- [66] Botha, J., & Pieterse, H. (2020, March). Fake news and deepfakes: A dangerous threat for 21st century information security. In *ICCWS 2020 15th International Conference on Cyber Warfare and Security. Academic Conferences and publishing limited* (p. 57).
- [67] Alattas, K. A., & Bayoumi, M. (2020). Reviewing the Existing Methodologies and Tools of Cloud Forensics: Challenges and Solutions. *International Journal of Cyber-Security and Digital Forensics*, 9(4), 147-155.
- [68] Jaiman, A. (2020). Debating the ethics of deepfakes. *Tackling Insurgent Ideologies in a Pandemic World, ORF and Global Policy Journal, New Delhi*, 75-79.
- [69] Meskys, E., Kalpokiene, J., Jurcys, P., & Liaudanskas, A. (2020). Regulating deep fakes: legal and ethical considerations. *Journal of Intellectual Property Law & Practice*, 15(1), 24-31.
- [70] Hodge Jr, S. D. (2021). Don't Always Believe What You See: Shallowfake and Deepfake Media Has Altered the Perception of Reality. *Hofstra L. Rev.*, 50, 51.
- [71] Ebermann, A. (2021). The effects of deepfakes and synthetic media on communication professionals.

- [72] Yadlin-Segal, A., & Oppenheim, Y. (2021). Whose dystopia is it anyway? Deepfakes and social media regulation. *Convergence*, 27(1), 36-51.
- [73] Gregory, S. (2023). Fortify the truth: How to defend human rights in an age of deepfakes and generative AI. *Journal of Human Rights Practice*, 15(3), 702-714.
- [74] Etienne, H. (2021). The future of online trust (and why Deepfake is advancing it). *AI and Ethics*, 1(4), 553-562.
- [75] van der Helm, M. (2021). *Harmful deepfakes and the GDPR* (Doctoral dissertation, Yüksek Lisans Tezi). <http://arno.uvt.nl/show.cgi>.
- [76] Karasavva, V., & Noorbhai, A. (2021). The real threat of deepfake pornography: A review of Canadian policy. *Cyberpsychology, Behavior, and Social Networking*, 24(3), 203-209.
- [77] Feeney, M. (2021). Deepfake Laws Risk Creating More Problems Than They Solve. *Regulatory Transparency Project*.
- [78] Fernandez, A. (2022). —Deep fakes!/: disentangling terms in the proposed EU Artificial Intelligence Act. *UFITA Archiv für Medienrecht und Medienwissenschaft*, 85(2), 392-433.
- [79] de Rancourt-Raymond, A., & Smaili, N. (2023). The unethical use of deepfakes. *Journal of Financial Crime*, 30(4), 1066-1077.

[80] Hameleers, M., van der Meer, T. G., & Dobber, T. (2022). You won't believe what they just said! The effects of political deepfakes embedded as vox populi on social media. *Social Media+ Society*, 8(3), 20563051221116346.

[81] Pawelec, M., & Sievi, L. (2023). Falschinformationen in den sozialen Medien als Herausforderung für deutsche Sicherheitsbehörden und-organisationen. *Kriminologie-Das Online-Journal| Criminology-The Online Journal*, (5), 316-347.

[82] Appel, M., & Prietzel, F. (2022). The detection of political deepfakes. *Journal of Computer-Mediated Communication*, 27(4), zmac008.

[83] Lussier, N. (2022). Nonconsensual deepfakes: detecting and regulating this rising threat to privacy. *Idaho L. Rev.*, 58, 353.

[84] O'Connell, A. (2024). Image-based Sexual Abuse and Gendered Conceptions of Harm in Tort. In *Diverse Voices in Tort Law* (pp. 221-244). Bristol University Press.

[85] Akhtar, Z. (2023). Deepfakes generation and detection: A short survey. *Journal of Imaging*, 9(1), 18.

[86] Mustak, M., Salminen, J., Mäntymäki, M., Rahman, A., & Dwivedi, Y. K. (2023). Deepfakes: Deceptions, mitigations, and opportunities. *Journal of Business Research*, 154, 113368.

[87] Pandey, M., & Singh, S. (2022, October). A Review on Various Deepfakes' Detection Methods. In *International Conference on Computing, Communications, and Cyber-Security* (pp. 179-191). Singapore: Springer Nature Singapore.

- [88] Okolie, C. (2023). Artificial intelligence-altered videos (deepfakes), image-based sexual abuse, and data privacy concerns. *Journal of International Women's Studies*, 25(2), 11.
- [89] Odena, A., Olah, C., & Shlens, J. (2017, July). Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning* (pp. 2642-2651). PMLR.
- [90] Shen, T., Liu, R., Bai, J., & Li, Z. (2018). ‘Deep fakes’ using generative adversarial networks (gan). *Noiselab, University of California, San Diego*.
- [91] Parimala, K., & Channappayya, S. (2019). Quality aware generative adversarial networks. *Advances in neural information processing systems*, 32.
- [92] Kancharla, P., & Channappayya, S. S. (2021). Improving the visual quality of video frame prediction models using the perceptual straightening hypothesis. *IEEE Signal Processing Letters*, 28, 2167-2171.
- [93] Maras, M. H., & Alexandrou, A. (2019). Determining authenticity of video evidence in the age of artificial intelligence and in the wake of Deepfake videos. *The International Journal of Evidence & Proof*, 23(3), 255-262.
- [94] Gu, S., Bao, J., Yang, H., Chen, D., Wen, F., & Yuan, L. (2019). Mask-guided portrait editing with conditional gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3436-3445).
- [95] Singh, R., Sethi, A., Saini, K., Saurav, S., Tiwari, A., & Singh, S. (2024). VALD-GAN: video anomaly detection using latent discriminator augmented GAN. *Signal, Image and Video Processing*, 18(1), 821-831.

- [96] Gong, L. Y., & Li, X. J. (2024). A Contemporary Survey on Deepfake Detection: Datasets, Algorithms, and Challenges. *Electronics*, 13(3), 585.
- [97] Santha, A. (2020). *Deepfakes generation using LSTM based generative adversarial networks*. Rochester Institute of Technology.
- [98] Guarnera, L., Giudice, O., & Battiato, S. (2020). Fighting deepfake by exposing the convolutional traces on images. *IEEE Access*, 8, 165085-165098.
- [99] Yang, T., Huang, Z., Cao, J., Li, L., & Li, X. (2022, June). Deepfake network architecture attribution. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 36, No. 4, pp. 4662-4670).
- [100] Paul, O. A. (2021). Deepfakes Generated by Generative Adversarial Networks.
- [101] Cherepkov, A., Voynov, A., & Babenko, A. (2021). Navigating the gan parameter space for semantic image editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3671-3680).
- [102] Gu, J., Liu, L., Wang, P., & Theobalt, C. (2021). Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*.
- [103] Kim, J., Lee, J., & Zhang, B. T. (2022). Smooth-swap: A simple enhancement for face-swapping with smoothness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10779-10788).

- [104] Xuan, X., Peng, B., Wang, W., & Dong, J. (2019, October). On the generalization of GAN image forensics. In *Chinese conference on biometric recognition* (pp. 134-141). Cham: Springer International Publishing.
- [105] Nguyen, H. H., Yamagishi, J., & Echizen, I. (2019). Use of a capsule network to detect fake images and videos. *arXiv preprint arXiv:1910.12467*.
- [106] Matern, F., Riess, C., & Stamminger, M. (2019, January). Exploiting visual artifacts to expose deepfakes and face manipulations. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)* (pp. 83-92). IEEE.
- [107] Yang, C., Ding, L., Chen, Y., & Li, H. (2021, July). Defending against gan-based deepfake attacks via transformation-aware adversarial faces. In *2021 international joint conference on neural networks (IJCNN)* (pp. 1-8). IEEE.
- [108] Güera, D., & Delp, E. J. (2018, November). Deepfake video detection using recurrent neural networks. In *2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS)* (pp. 1-6). IEEE.
- [109] Li, Y., Chang, M. C., & Lyu, S. (2018, December). In icu oculi: Exposing ai created fake videos by detecting eye blinking. In *2018 IEEE International workshop on information forensics and security (WIFS)* (pp. 1-7). IEEE.
- [110] Ciftci, U. A., Demir, I., & Yin, L. (2020). Fakecatcher: Detection of synthetic portrait videos using biological signals. *IEEE transactions on pattern analysis and machine intelligence*.
- [111] Hsu, C. C., Zhuang, Y. X., & Lee, C. Y. (2020). Deep fake image detection based on pairwise learning. *Applied Sciences*, 10(1), 370.

- [112] Malolan, B., Parekh, A., & Kazi, F. (2020, March). Explainable deep-fake detection using visual interpretability methods. In *2020 3rd International conference on Information and Computer Technologies (ICICT)* (pp. 289-293). IEEE.
- [113] Shad, H. S., Rizvee, M. M., Roza, N. T., Hoq, S. A., Monirujjaman Khan, M., Singh, A., ... & Bourouis, S. (2021). Comparative Analysis of Deepfake Image Detection Method Using Convolutional Neural Network. *Computational intelligence and neuroscience*, 2021(1), 3111676.
- [114] Zhang, H., Chen, B., Wang, J., & Zhao, G. (2022). A local perturbation generation method for gan-generated face anti-forensics. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(2), 661-676.
- [115] Juefei-Xu, F., Wang, R., Huang, Y., Guo, Q., Ma, L., & Liu, Y. (2022). Countering malicious deepfakes: Survey, battleground, and horizon. *International journal of computer vision*, 130(7), 1678-1734.
- [116] Hao, H., Bartusiak, E. R., Güera, D., Mas Montserrat, D., Baireddy, S., Xiang, Z., ... & Delp, E. J. (2022). Deepfake detection using multiple data modalities. In *Handbook of digital face manipulation and Detection: From DeepFakes to morphing attacks* (pp. 235-254). Cham: Springer International Publishing.
- [117] Hooda, A., Mangaokar, N., Feng, R., Fawaz, K., Jha, S., & Prakash, A. (2024). D4: Detection of adversarial diffusion deepfakes using disjoint ensembles. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 3812-3822).

- [118] Rai, A. K., & Srivastava, S. (2023). A Thorough Investigation on Image Forgery Detection. *CMES-Computer Modeling in Engineering & Sciences*, 134(3).
- [119] Sharma, P., Kumar, M., & Sharma, H. (2023). Comprehensive analyses of image forgery detection methods from traditional to deep learning approaches: an evaluation. *Multimedia Tools and Applications*, 82(12), 18117-18150.
- [120] Chakraborty, T., KS, U. R., Naik, S. M., Panja, M., & Manvitha, B. (2024). Ten years of generative adversarial nets (GANs): a survey of the state-of-the-art. *Machine Learning: Science and Technology*, 5(1), 011001.
- [121] Sharma, P., Kumar, M., & Sharma, H. K. (2023). A generalized novel image forgery detection method using generative adversarial network. *Multimedia Tools and Applications*, 1-32.
- [122] Abdelmotaal, H., Abdou, A. A., Omar, A. F., El-Sebaity, D. M., & Abdelazeem, K. (2021). Pix2pix conditional generative adversarial networks for scheimpflug camera color-coded corneal tomography image generation. *Translational Vision Science & Technology*, 10(7), 21-21.
- [123] Christovam, L. E., Shimabukuro, M. H., Galo, M. D. L. B., & Honkavaara, E. (2021). Pix2pix conditional generative adversarial network with MLP loss function for cloud removal in a cropland time series. *Remote Sensing*, 14(1), 144.
- [124] H. Patil, S. Chaudhari, and V. Narawade, —Remote Sensing Image Forgery Detection Using Modified U-net, 43rd Asian Conf. Remote Sensing, ACRS 2022, 2022.

- [125] Lan, L., You, L., Zhang, Z., Fan, Z., Zhao, W., Zeng, N., ... & Zhou, X. (2020). Generative adversarial networks and its applications in biomedical informatics. *Frontiers in public health*, 8, 164.
- [126] Fahh, C. S., & Wu, T. C. (2022, September). A Deep-Neural-Network-Based Approach To Detecting Forgery Images Generated From Various Generative Adversarial Networks. In *2022 International Conference on Machine Learning and Cybernetics (ICMLC)* (pp. 115-123). IEEE.
- [127] Bi, X., Zhang, Z., & Xiao, B. (2021). Reality transform adversarial generators for image splicing forgery detection and localization. In *proceedings of the IEEE/CVF international conference on computer vision* (pp. 14294-14303).
- [128] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.
- [129] Mareen, H., De Neve, L., Lambert, P., & Van Wallendael, G. (2023). Harmonizing Image Forgery Detection & Localization: Fusion of Complementary Approaches. *Journal of Imaging*, 10(1), 4.
- [130] Zhang, J., Huang, X., Liu, Y., Han, Y., & Xiang, Z. (2024). GAN-based medical image small region forgery detection via a two-stage cascade framework. *Plos one*, 19(1), e0290303.
- [131] Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2019, May). Self-attention generative adversarial networks. In *International conference on machine learning* (pp. 7354-7363). PMLR.

- [132] Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., & Paul Smolley, S. (2017). Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2794-2802).
- [133] Li, R., Li, X., Hui, K. H., & Fu, C. W. (2021). SP-GAN: Sphere-guided 3D shape generation and manipulation. *ACM Transactions on Graphics (TOG)*, 40(4), 1-12.
- [134] Miyato, T., Kataoka, T., Koyama, M., & Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*.
- [135] Arjovsky, M., Chintala, S., & Bottou, L. (2017, July). Wasserstein generative adversarial networks. In *International conference on machine learning* (pp. 214-223). PMLR.
- [136] Haloui, I., Gupta, J. S., & Feuillard, V. (2018). Anomaly detection with Wasserstein GAN. *arXiv preprint arXiv:1812.02463*.
- [137] Hao, Z., Li, Z., Ren, F., Lv, S., & Ni, H. (2022). Strip steel surface defects classification based on generative adversarial network and attention mechanism. *Metals*, 12(2), 311.
- [138] <https://www.kaggle.com/dansbecker/5-celebrity-faces-dataset>
- [139] <https://www.kaggle.com/iamsouravbanerjee/indian-actor-images->
- [140] Benny, Y., Galanti, T., Benaim, S., & Wolf, L. (2021). Evaluation metrics for conditional image generation. *International Journal of Computer Vision*, 129, 1712-1731.

- [141] Kinakh, V., Drozdova, M., Quétant, G., Golling, T., & Voloshynovskiy, S. (2021). Information-theoretic stochastic contrastive conditional GAN: InfoSCC-GAN. *arXiv preprint arXiv:2112.09653*.
- [142] Castro, F. M., Marín-Jiménez, M. J., Guil, N., Schmid, C., & Alahari, K. (2018). End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 233-248).
- [143] Kumar, M., & Sharma, H. K. (2023). A GAN-based model of deepfake detection in social media. *Procedia Computer Science*, 218, 2153-2162.
- [144] Shmelkov, K., Schmid, C., & Alahari, K. (2018). How good is my GAN? In *Proceedings of the European conference on computer vision (ECCV)* (pp. 213-229).
- [145] Li, C., Wang, Z., & Qi, H. (2018, October). Fast-converging conditional generative adversarial networks for image synthesis. In *2018 25th IEEE international conference on image processing (ICIP)* (pp. 2132-2136). IEEE.
- [146] Chavdarova, T., & Fleuret, F. (2018). Sgan: An alternative training of generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 9407-9415).
- [147] Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*.
- [148] Lee, G., & Kim, M. (2021). Deepfake detection using the rate of change between frames based on computer vision. *Sensors*, 21(21), 7367.
- [149] Almars, A. M. (2021). Deepfakes detection techniques using deep learning: a survey. *Journal of Computer and Communications*, 9(05), 20-35.

- [150] Sharma, P. , Kumar, M., & Sharma, H.K. (2023). Ensemble WGAN (EWG): Advancing Image Synthesis and Deepfake Detection with Heterogeneous Discriminator Approach. *Computology: Journal of Applied Computer Science and Intelligent Technologies*, 3 (2), 69-97.

- [151] Naskar, G., Mohiuddin, S., Malakar, S., Cuevas, E., & Sarkar, R. (2024). Deepfake Detection using Deep Feature Stacking and Meta-learning. *Heliyon*.

- [152] Rafique, R., Gantassi, R., Amin, R., Frnda, J., Mustapha, A., & Alshehri, A. H. (2023). Deep fake detection and classification using error-level analysis and deep learning. *Scientific Reports*, 13(1), 7422.

- [153] Kosarkar, U., Sarkarkar, G., & Gedam, S. (2023). Revealing and Classification of Deepfakes Video's Images using a Customize Convolution Neural Network Model. *Procedia Computer Science*, 218, 2636-2652.

- [154] Altaei, M. S. M. (2023). Detection of Deep Fake in Face Images Based Machine Learning. *Al-Salam Journal for Engineering and Technology*, 2(2), 1-12.

- [155] AK, K. (2023). Ensemble of Machine Learning Classifiers for Detecting Deepfake Videos using Deep Feature. *IAENG International Journal of Computer Science*, 50(4).

- [156] Bray, S. D., Johnson, S. D., & Kleinberg, B. (2023). Testing human ability to detect _deepfake‘images of human faces. *Journal of Cybersecurity*, 9(1), tyad011.

- [157] Mitra, A., Mohanty, S. P., Corcoran, P., & Kougianos, E. (2021). A machine learning based approach for deepfake detection in social media through key video frame extraction. *SN Computer Science*, 2(2), 98.

- [158] Bernhard Mathe. "Real and Fake Face Detection." Kaggle, 2020. Real and Fake Face Detection Dataset. Accessed May 27, 2024.

- [159] Liu, Y., & Yang, S. (2022). Application of Decision Tree-Based Classification Algorithm on Content Marketing. *Journal of Mathematics*, 2022(1), 6469054.

- [160] Mehra, A. (2020). *Deepfake detection using capsule networks with long short-term memory networks* (Master's thesis, University of Twente).

- [161] Trabelsi, A., Pic, M. M., & Dugelay, J. L. (2022, August). Improving Deepfake Detection by Mixing Top Solutions of the DFDC. In 2022 30th European Signal Processing Conference (EUSIPCO) (pp. 643-647). IEEE.

- [162] Wang, G., Jiang, Q., Jin, X., & Cui, X. (2022). FFR_FD: Effective and fast detection of DeepFakes via feature point defects. *Information Sciences*, 596, 472-488.

List of Publications

- Sharma, Preeti, Manoj Kumar, and Hitesh Kumar Sharma. "GAN-CNN Ensemble: A Robust Deepfake Detection Model of Social Media Images Using Minimized Catastrophic Forgetting and Generative Replay Technique." *Procedia Computer Science* 235 (2024): 948-960. **DOI:** [10.1016/j.procs.2024.04.090](https://doi.org/10.1016/j.procs.2024.04.090)
- Sharma, P., Kumar, M., Sharma, H.K. et al. Generative adversarial networks (GANs): Introduction, Taxonomy, Variants, Limitations, and Applications. *Multimed Tools Appl* (2024). **DOI:** <https://doi.org/10.1007/s11042-024-18767-y>.
- P. Sharma, M. Kumar, and H. K. Sharma, —Robust GAN-Based CNN Model as Generative AI Application for Deepfake Detection, *EAI Endorsed Trans IoT*, vol. 10, Apr. 2024. **DOI:** <https://doi.org/10.4108/eetiot.5637>
- P. Sharma, M. Kumar, H. K. Sharma and M. Memoria, "Implementation of Attention Mechanism in GAN for the Purpose of Quality Detection," *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)*, Greater Noida, India, 2024, pp. 1081-1087, **DOI:** [10.1109/IC2PCT60090.2024.10486643](https://doi.org/10.1109/IC2PCT60090.2024.10486643)
- Sharma, Preeti, Manoj Kumar, and Hitesh Kumar Sharma. "A generalized novel image forgery detection method using generative adversarial network." *Multimedia Tools and Applications* (2023): 1-32. **DOI:** [10.1007/s11042-023-17588-9](https://doi.org/10.1007/s11042-023-17588-9)

- Sharma, Preeti, Kumar, M., & Sharma, H. K. (2023). A GAN-based model of deepfake detection in social media. *Procedia Computer Science*, 218, 2153-2162.) **DOI:** [10.1016/j.procs.2023.01.191](https://doi.org/10.1016/j.procs.2023.01.191)
- Sharma, Preeti, Manoj Kumar, and Hitesh Sharma. "Comprehensive analyses of image forgery detection methods from traditional to deep learning approaches: an evaluation." *Multimedia Tools and Applications* 82.12 (2023): 18117-18150. **DOI:** [10.1007/s11042-022-13808-w](https://doi.org/10.1007/s11042-022-13808-w)
- A research paper entitled —Ensemble WGAN(EWG): Advancing image synthesis and deep fake detection with heterogeneous discriminator Approach is submitted in the journal —Computology: Journal of Applied Computer Science and Intelligent Technologies. **DOI:** [10.17492/computology.v3i2.2304](https://doi.org/10.17492/computology.v3i2.2304)
- Sharma, Preeti, Manoj Kumar, and Hitesh Kumar Sharma. "Unsupervised Multi-Scale Image Enhancement Using Generative Deep Learning Approach." *CS & IT Conference Proceedings*. Vol. 13. No. 24. CS & IT Conference Proceedings, 2023. **DOI:** [10.5121/csit.2023.132415](https://doi.org/10.5121/csit.2023.132415)

PLAGIARISM CERTIFICATE

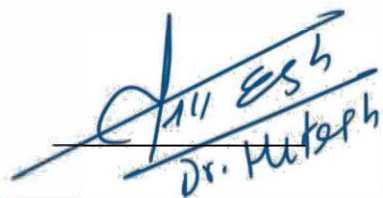
1. We _Dr. Hitesh kumar sharma_ (Internal Guide), Dr. Manoj kumar_ (Co Guide/ External Guide) certify that the Thesis titled "A Deepfakes Detection Model for Face Forgeries Generated by Generic GAN in Social Media"

submitted by Scholar Mr/ Ms _____ Preeti _____ having SAP ID

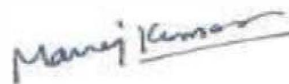
_____500089519_____ has been run through a Plagiarism Check Software and the Plagiarism

Percentage is reported to be __9%_____.

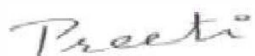
2. Plagiarism Report generated by the Plagiarism Software is attached .



Signature of the Internal Guide



Signature of External Guide/ Co Guide



Signature of the Scholar

ORIGINALITY REPORT

9%

SIMILARITY INDEX

6%

INTERNET SOURCES

7%

PUBLICATIONS

2%

STUDENT PAPERS

PRIMARY SOURCES

1

dokumen.pub

Internet Source

<1%

2

Preeti Sharma, Manoj Kumar, Hitesh Kumar Sharma. "A generalized novel image forgery detection method using generative adversarial network", Multimedia Tools and Applications, 2023

Publication

<1%

3

Submitted to University of Petroleum and Energy Studies

Student Paper

<1%

4

Submitted to Carnegie Mellon University

Student Paper

<1%

5

Submitted to Liverpool John Moores University

Student Paper

<1%

6

Preeti Sharma, Manoj Kumar, Hitesh Kumar Sharma. "GAN-CNN Ensemble: A Robust Deepfake Detection Model of Social Media Images Using Minimized Catastrophic Forgetting and Generative Replay Technique",

<1%