

Convolution vs. Attention: A Comparative Study in Deepfake Detection

by

MD. Abu Hussain

21101242

Mst Nishat Tabassum Srabony

23341033

Noshin Anjum Malik Piasha

21201340

Subrina Nowrin

24341097

A thesis submitted to the Department of Computer Science and Engineering.
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
January 2025

© 2025. BRAC University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



MD. Abu Hussain
21101242



Mst Nishat Tabassum Srabony
23341033



Noshin Anjum Malik Piasha
21201340



Subrina Nowrin
24341097

Approval

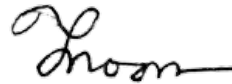
The thesis titled “Convolution vs. Attention: A Comparative Study in Deepfake Detection” submitted by

1. MD.Abu Hussain (21101242)
2. Mst Nishat Tabassum Srabony (23341033)
3. Noshin Anjum Malik Piasha (21201340)
4. Subrina Nowrin (24341097)

Of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 31, 2025.

Examining Committee:

Supervisor:
(Member)



Dr. Jannatun Noor Mukta
Associate Professor, CSE
Director, BSDS
United International University

Thesis Coordinator::
(Member)

Md. Golam Rabiul Alam, Ph.D.
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi
Associate Professor and Chairperson
Department of Computer Science and Engineering
BRAC University

Abstract

Deep Fakes are a kind of media which is made by using AI methodology to manipulate or generate realistic-looking content, often involving human faces. These techniques can have adverse effects across social, emotional, and political spheres. And hence, a demand for state-of-the-art neural network solutions for media authentication with precision has been risen to combat deep fakes. In this research, Convolution and Attention neural network architectures are compared and contrasted, thoroughly demonstrating the superiority of Convolutions in detecting deepfakes. Through rigorous experimental testing on diverse benchmark datasets, Convolution architectures consistently delivered over 15% higher accuracy, more robust feature extraction capability, and significantly fewer false-positive rates and fewer losses compared to Attention architectures. The research comparatively analyzes the architectural differences in these neural network approaches, revealing Convolutions' complex spatial pattern recognition mechanisms as a key differentiator for the detection of deep fakes. Experimental results highlight the importance of architectural design for deepfake detection technologies, offering valuable implications for researchers and practitioners who are involved in media authentication solutions. The findings not only reveal the present shortcomings of Attention models but also point toward a hopeful direction for further development of synthetic media detection methods. And to further prove this point, we've tested transformer models like CCT and ConvMixer, which utilize Convolutions in an attention-based transformer mechanism. We've also created a hybrid ViTXDensenet model which uses the DenseNet121's CNN mechanism for feature extraction, uses MultiHeadSelfAttention and FFN transformations of ViT along with Gelu activation, and passes the output through a Dense Layer after flattening. Among all the models we worked with, EfficientNetV2M performs best with 100% accuracy. Finding a suitable feature extraction method through different models, like Convolutions and Attentions, is essential for optimizing performance in tasks like image classification or deepfake detection. Convolutions excel at capturing local patterns, making them efficient for smaller tasks, while Attentions capture global relationships through self-attention, often performing better on large, complex datasets.

Keywords: DenseNet121, ResNet50V2, ViTXDesNet, Xception, EfficientNetV2M.

Table of Contents

Declaration	i
Approval	ii
Abstract	iii
Table of Contents	iv
List of Figures	vi
List of Tables	1
1 Introduction	2
1.1 Consideration and Significance of Deepfake	2
1.2 Background	3
1.3 Problem Statement	3
1.4 Research Objectives	4
1.5 Technical Foundations of Deepfake Detection	5
1.6 Current Detection Methods	5
1.7 Challenges and Limitations	6
1.8 Ethical and Legal Considerations	6
2 Precursors:	7
3 Literature Review	9
4 Methodology	26
5 Description of Model	28
5.1 Model Architecture	28
5.2 Configuration for Training:	36
6 Description of the Data	38
6.1 Features:	38
6.2 Data Preprocessing	39
7 Results and Analysis:	40
7.1 Evaluation Matrics:	40
7.1.1 Precision	40
7.1.2 Recall	40

7.1.3	F1 Score	41
7.1.4	ROC AUC Score	41
7.1.5	Average Precision Score	41
7.1.6	Accuracy	41
7.2	Convolution model Result and Analysis:	42
7.3	Attention model Result and Analysis:	42
8	Discussion	46
9	Conclusion	48
	Bibliography	54

List of Figures

1.1	Deepfake Examples Before and After	3
2.1	Architecture of Covolution	7
2.2	Architecture of Attention	8
4.1	Methodology Dataflow	27
5.1	Architecture of DenseNet121 [4]	28
5.2	Architecture of ResNet50V2 [42]	29
5.3	Architecture of MobileNetV3Large [65]	30
5.4	Architecture of InceptionResNetV2 [6]	30
5.5	Architecture of EfficientNetV2M [74]	31
5.6	Architecture of Xception [13]	31
5.7	Architecture of ConvMixer [50]	32
5.8	Architecture of ViT [70]	32
5.9	Architecture of DiT [59]	33
5.10	Architecture of EANet [53]	34
5.11	Architecture of CCT [53]	34
5.12	Architecture of ViTxDensenet	35
5.13	Architecture of Attention-based MIL method [64]	36
6.1	Real and fake faces	38
6.2	Total no of real and fake faces	39
7.1	Accuracy graph of EfficientNetV2M	42
7.2	Prediction graph of all Convolution-based Models	43
7.3	Prediction graph of all Attention-based Models	44
7.4	Prediction graph of custom hybrid model ViTxDenseNet	45

List of Tables

3.1	Summary of Research Papers on Deepfake Detection	22
5.1	Hyperparameters and other training details.	37
7.1	An overview of the architecture of the DenseNet121, ResNet50v2, MobileNetV3Large, InceptionResnetV2, EfficientNetV2M, Xception, ConvMixer, VitXDenseNet, VIT, CCT, DIT, Attention-based MIL, EANet models	45

Chapter 1

Introduction

1.1 Consideration and Significance of Deepfake

Deepfake, a kind of media in which the original media is manipulated into a fake, but convincing media, is becoming increasingly widespread in the world today, especially after the development and rapid advancement of AI technology. Although deep fakes in their infancy were too obvious and anyone could spot the altered content, nowadays, it has become harder than ever to detect deep fake content due to the advancement of deep fake technology. In fact, it has become so sophisticated that it has now become more challenging to differentiate deep fake content from original content, and it will continue to be more sophisticated in the future. As a result, people with malicious intent can use the technology to cause serious harm. For example, they can use it to spread misinformation on social networks, and these deep fakes can be used as the so-called proof to bolster their claims. Deep fakes also have other consequences, like privacy violations, and diminishing public trust in the media. A recent example of malicious actors creating a deep fake is that of Taylor Swift. In a recent CNN report by Tenbarge [73], some deepfake videos on X (formerly Twitter) of Taylor Swift supporting Donald Trump, a US Presidential candidate, were found, and one of them came from an account with a million followers. This shows how deepfakes can be used to impersonate celebrities for political gains.

Another example is that of Ukrainian President Zelensky. Pierson et al.[76] reported that a poorly edited deep fake of Zelensky surfaced where he publicly capitulated to Russia's demands during the Russo-Ukrainian war in 2022. Though it was a subject of ridicule, experts they interviewed predicted more sophisticated media might be brought to light. This shows how this technology can be used for dangerous situations like wars and conflicts. These are just two examples of how malicious deep fake use can be, and anyone who's not informed about this technology can be easily persuaded by it. And this is why it has become essential to devise deep fake detection technologies to counter such deep fakes. Effective detection technologies can help not only debunk the deep fakes that are so hard to differentiate from the original but also be used as evidence against the deep fakes that are used for malicious intent, proving they were fake. In this paper, we will take a look at deep fake detection technology in its current state, evaluate its effectiveness, and present enhancements to the current technology or propose new models to improve detection capabilities.

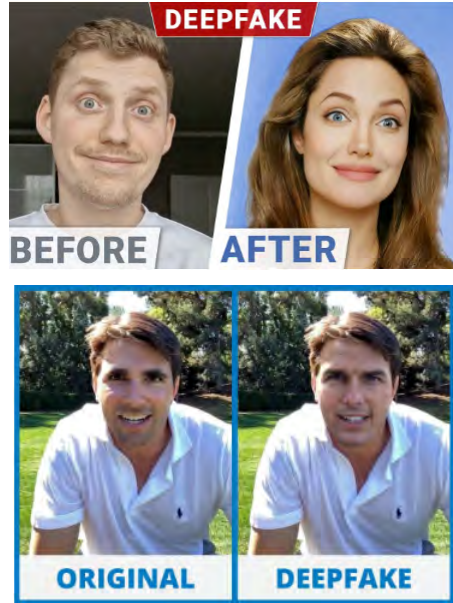


Figure 1.1: Deepfake Examples Before and After

1.2 Background

Before we dive deep into deepfake detection, we need to know what deepfake is, its history, and how it works. According to the Collins Dictionary [77], a deep fake is a technique where an image is superimposed on another image to make it look like it was not edited. Jye Sawtell-Rickson in his article “Builtin” [60] defines deepfake as “artificial media made with deep learning techniques where features of one image are replaced with another”. From the definitions, it is obvious that deepfakes, specifically image and video deepfakes work by face-swapping one image with another using AI. Though photo manipulation existed for a long time, deepfake technology started being developed around the 90s. Christoph et al. did research in 1997 [51], where they modified a video of a person speaking by altering his mouth to match with a different audio. This is regarded as the first system to do this kind of deepfake technique automatically using a machine learning method. Deepfake detection was used mainly for computer science research, until 2017 which according to Samantha from Vice [8], is when a Reddit community known as deepfakes started making it more public. She also mentions a user named ‘deepfake app’ who founded an app to create deepfakes, known as FakeApp, which allowed users to face-swap their faces with another. Eventually, deepfakes started gaining popularity, and with the advancement of AI, it has become mainstream. However, deepfakes did come with a lot of harm since deepfake videos and images were made without the consent of the person whose face or audio is being used. This brought the need to detect such fake, but convincing media, known as deepfake detection techniques.

1.3 Problem Statement

The rapid development of deepfake generating algorithms, spurred by deep architectures such as GANs and diffusion models, makes it ever challenging to discern

real and generated media. Convolutional neural networks (CNNs) have long been utilized for deepfake detection with regard to feature extraction in a spatial manner, but in recent times, approaches with an attention mechanism have become prevalent for long-term dependencies and contextual information modeling capabilities. Artifacts such as distortion, noise, blur, etc. in deepfake are concentrated in smaller areas, making them local features [39], [52], [68]. Finding the proper feature extraction mechanism will improve the future trend in deepfake detection as well. Nevertheless, little comparative evaluation can be noticed in current works in terms of effectiveness, efficiency, and robustness in deepfake detection when comparing these two techniques. Practitioners and researchers have no basis for choosing between a single best practice and a best practice in combination for deepfake detection in increasingly sophisticated deepfakes. Closing this research gap, this thesis seeks to conduct a comparative evaluation of CNNs and approaches with an attention mechanism with a view towards offering actionable recommendations for improving deepfake detection algorithms.

1.4 Research Objectives

As deepfake technology becomes increasingly sophisticated, it is essential to examine and compare the effectiveness of different AI-driven detection methods. This research aims to systematically analyze convolutional neural networks (CNNs) and attention-based models in deepfake detection, focusing on their accuracy, precision, recall, and overall reliability. Besides performance metrics, this study will assess how well these models generalize across diverse datasets, ensuring their robustness in real-world scenario.

The research objective is to explore and compare different feature extraction methods, such as Convolutions and Attentions, to determine the most effective approach for tasks like deepfake detection. By analyzing the strengths of each model, Convolutions for local pattern recognition and Attentions for capturing global relationships, the goal is to identify the optimal feature extraction technique that balances accuracy, efficiency, and performance across various datasets and tasks.

Computational efficiency is another critical aspect, as practical deployment requires models that balance accuracy with speed and resource consumption. This research will evaluate the computational demands of CNNs and attention mechanisms, determining their feasibility for real-time detection. Additionally, the robustness of these models against adversarial attacks and noise will be investigated, providing insights into their reliability in evolving threat landscapes. Recognizing that CNNs and attention mechanisms have unique strengths, this study will explore hybrid architectures that integrate both approaches, assessing whether they offer improved deepfake detection performance. Furthermore, interpretability remains a crucial factor in AI-driven detection systems. By analyzing the features learned by each method, this research will provide insights into their decision-making processes, enhancing transparency in automated deepfake detection. Through this comparative study, the research will contribute to the advancement of deepfake detection methodologies, addressing key technical challenges while also considering the broader implications

for digital security and prevention of misinformation.

1.5 Technical Foundations of Deepfake Detection

Deepfake detection employ numerous techniques, such as machine learning, computer vision, and signal processing. Detection methods often focus on identifying inconsistencies in physical properties such as facial movements, lighting, and audio-visual synchrony. This section will review foundational studies and key advancements in deepfake detection technologies.

Deepfake detection employ numerous techniques, such as machine learning, computer vision, and signal processing. Detection methods often focus on identifying inconsistencies in physical properties such as facial movements, lighting, and audio-visual synchrony. Since images were used as a medium for deepfake detection in this research, the focus will be on some of the weaknesses of deepfakes, which are basically the technical foundations of image deepfake detection.

- **Pixel-Level Anomalies:** Some imperfections like pixelated artifacts, blurring, and other imperfections in key places like eyes, nose, mouth, though subtle, exist in deepfakes. Texture inconsistencies like uneven skin texture also exist in deepfakes.
- **Facial Feature Inconsistencies:** Human face is naturally symmetrical and has consistent lighting in eyes and facial regions, but deepfakes are asymmetrical and the lighting in eyes and facial regions are different.
- **Biometric Inconsistencies:** Deepfakes sometimes make irregular placements of some facial landmarks like eyes and nose. Another common failure of deepfakes is that the human expression is unnatural and inconsistent.
- **Color & Lighting Inconsistencies:** Deepfakes usually struggle to keep the lighting, reflections, shading, and skin tones consistent.

By utilizing these weaknesses, deepfake detection methods can find out if a photo is real or fake.

1.6 Current Detection Methods

1. **Manual Inspection:** Initially, deepfake detection relied on expert analysis of visual and auditory cues. However, as deep fakes became more sophisticated, manual methods proved inadequate.
2. **Machine Learning-Based Approaches:** These approaches use supervised and unsupervised learning to train models on datasets of real and fake media. The most used methods are CNN and RNN.

3. **Forensic Techniques:** They analyze digital artifacts left by deepfake generation processes, such as inconsistencies in pixel patterns or compression artifacts.
4. **Hybrid Approaches:** They use a combination of multiple, such as using both forensic analysis and machine learning, to improve detection accuracy.

1.7 Challenges and Limitations

Despite advances, several challenges remain:

- **High Quality of Deepfakes:** As GANs and other AI models improve, distinguishing deepfakes from real media becomes increasingly difficult.
- **Dataset Limitations:** Effective detection requires large, diverse datasets of both real and fake media, which are often lacking.
- **Generalization:** Many detection models show great performance on some variants of deepfakes but unable to work generally across different manipulation techniques and media formats.
- **Adversarial Attacks:** Detection systems themselves can be targeted and fooled by adversarial attacks, complicating the detection process.

1.8 Ethical and Legal Considerations

Deepfake detection is essential for combating misinformation, but it raises ethical concerns around privacy, consent, and false positives. This study uses publicly available dataset that is used for deepfake detection research, which, like many others, includes images of public figures without explicit consent. While such datasets are crucial for training effective models, they highlight the ongoing debate about balancing technological advancement with individual rights. Additionally, the risk of

false positive where real content is mistakenly flagged as fake undervalues the need for careful and responsible application of these models. As deepfake detection technology progresses, establishing ethical guidelines and legal safeguards is critical to ensuring its fair and responsible use.

Chapter 2

Precursors:

Convolution [1] and attention [7] are two different techniques used in deep learning, especially for processing images or other structured data. Both have unique ways of working and are suitable for different types of tasks.

Convolution is the process of extracting features from input data by focusing on small local regions. Think of it as looking at an image through a small filter or kernel that runs over the image. This filter or kernel detects certain features like edges, corners, or textures for example. Convolution is great for capturing local features and spatial relationships, such as how pixels in a given area relate to each other. It builds from simple patterns in the first layer of the network to more complex patterns in the deeper layers. Here, Figure: 2.1 represents the architecture of convoluted networks.

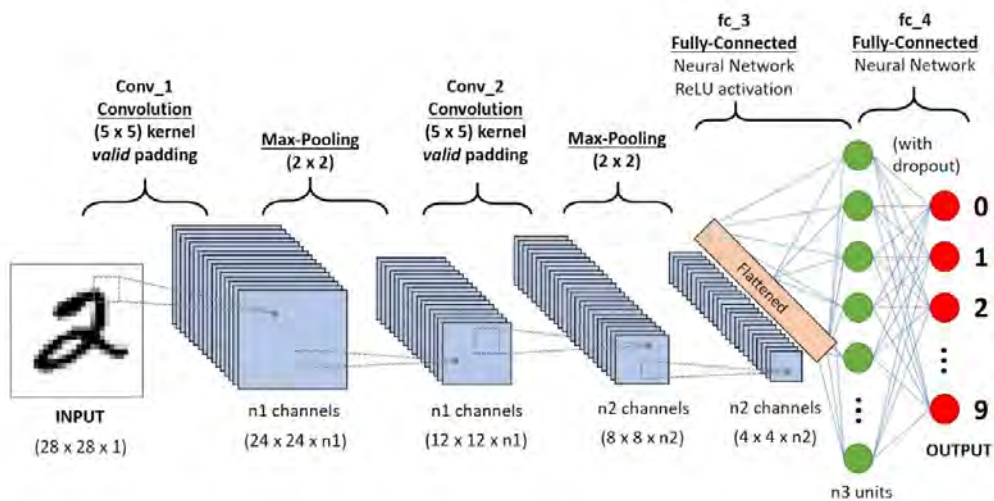


Figure 2.1: Architecture of Covolution

However, Attention works differently. Instead of focusing on just a local region, it looks at all parts of the input at once and decides which parts are most important

for the task. For example, in an image, attention might highlight regions that are more relevant for identifying an object. It calculates the relationships between each part of the input and every other part. This global view allows attention to connect distant elements, such as understanding the relationships between different parts of a sentence in text or different regions of an image. The basic idea of attention is to give “weights” to several segments of the input. These weights tell the model how much to focus on each part. For instance, when we’re to classify a cat’s image, attention might focus more on the face and legs than on the background. Here, figure:

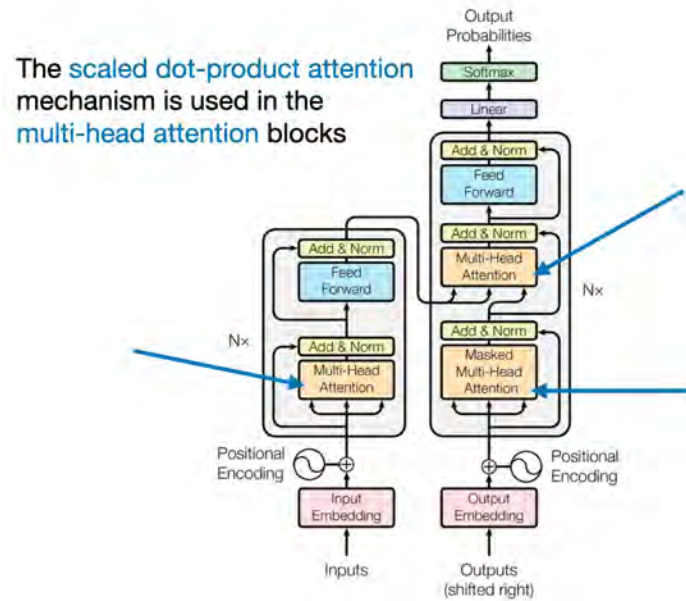


Figure 2.2: Architecture of Attention

Convolution is computationally efficient, especially for large inputs like high-resolution images, because it processes small segments of the input simultaneously. Attentions are usually computationally expensive, especially for large inputs, because it computes the relationship among each pair of elements in the input. Convolution is great for capturing local patterns but has difficulty connecting distant regions of an input. Attention, on the other hand, has a global perspective, which helps it understand the relationship between any part of the input.

Chapter 3

Literature Review

Looking at some research papers and publications, we found information regarding how some models work, how their dataset was collected, what detection methods they use, and what sort of algorithm was implemented. From our findings, data sets like VGGFace use sample images of celebrities, particularly taken from sites like IMDB. Although they might help in detecting deep fakes of certain top celebrities, they are not generalized, thus they might potentially fail to detect deep fakes on a broader scale. Some deep fake datasets use samples of people who voluntarily participated in deep fake research and consented to use their faces for deep fake and its detection. However, this fails to detect on a broader scale where images of countless individuals can be misused and because the detector is trained with datasets of celebrity images, it may have faults in deepfake detection. It may even show false positives.

This study, [31] used eight different CNN models and several architectures to perform DeepFake Image Detection. The model used the Kaggle dataset, which has 140,000 images from Flickr, and the styleGAN dataset. Around 8 models of CNN were trained by this. They used DenseNet, VGGNet, ResNet50, VGGFace, and a custom-made CNN architecture. The key to quickly and affordably detecting deepfake images and videos is machine learning. Additionally, a custom model for dropout and padding has been built to ascertain whether or not the other model is reflected on the object. For the result, they used five matrices: accuracy, precision, recall, F1 score, and area under the ROC curve. Among the models they tested, VGGFace had 99% accuracy, ResNet50 had 97%, DenseNet201 had 96%, DenseNet169 had 95%, VGG19 had 94%, VGG16 had 92%, DenseNet121 had 97%, and the custom model had 90% accuracy.

Using a multi-attentional deepfake detection network, in this research [33] proposes an advanced approach to the problem of detecting deepfakes. This method departs from conventional binary classification techniques. The objective is to improve the detection of instant local differences between authentic and altered images, which are frequently ignored by broader binary classifiers. Multiple well-known datasets, like FaceForensics++, Celeb-DF, and DFDC, were used for model testing. Multiple spatial attention heads are included in a new learning model that is presented in this research. With the help of these heads, the network can concentrate on various regions of a picture, improving its ability to identify subtle anomalies that suggest

manipulation. Textural Feature Enhancement, Bilinear Attention Pooling(BAP), Regional Independence Loss, and Attention Guided Data Augmentation are important parts of the model. The model makes use of some sophisticated mathematical techniques, including loss functions, attention mechanisms, normalization and pooling, and others. The model outperformed conventional techniques in terms of accuracy: The FaceForensics++ dataset achieved remarkable performance with an accuracy of up to 99.80% under high-quality conditions. Demonstrated notable gains over previous techniques in terms of robustness and transferability on the Celeb-DF and DFDC datasets.

This research [55] provides an advanced ML method for improvising the identification and classification of deepfake content using a modified model called ResNet-Swish-Dense54. Deepfakes are technologically modified media, usually audio or video, that are changed to portray content that is fictitious but sounds realistic. Robust detection approaches are required due to the growing frequency of deepfakes, which pose threats in the societal, political, and personal spheres. In the research, three important data sets DFDC, FaceForensec++ and CelebDF were used to evaluate the effectiveness of the suggested approach. The ResNet-Swish-Dense54 model forms the basis of the proposed solution. The ResNet architecture, which is well known for its deep residual learning framework, is integrated with additional dense layers to boost feature extraction and manipulation detection accuracy, as well as the Swish activation function for improved model training dynamics. Furthermore, the model ResNet-Swish-Dense54 achieves 99.99% accuracy. The model performed admirably on the FaceForensics++ model, outperforming current benchmarks in accuracy and obtaining almost flawless classification rates. This shows how highly successfully and dependably it can detect and categorize deepfake content.

This study [43] presents an ML method to efficiently identify deep-fake images. Deepfake technology, which creates and affects synthetic media, poses a serious risk since it can be used for many illegal purposes, such as identity theft, cyberextortion, and the spreading of false information. The goal is to prevent these harmful applications by developing a trustworthy technique to differentiate between real and fake media. Several dataset types, including FaceForensics++ (FF++), CelebDF, Deep-Fake Detection Challenge Dataset (DFDC), and others, were employed in this study. For deepfake detection, the DFP methodology showed 94% accuracy and 95% precision. To create a hybrid model specially for deepfake detection, the study combines convolutional neural network (CNN) components with VGG16 architecture. This method makes use of VGG16 depth for in-depth analysis as well as the power of conventional CNNs for feature extraction. Several mathematical techniques, including transfer learning, convolutional layers, dropout, and polling layers, were employed in this study. When compared to conventional models and other methods, the suggested model’s accuracy of 94% on the test dataset is noticeably greater. The goal of this study is to provide a more precise and effective technique to identify false images, giving cybersecurity experts the means to protect people and institutions from the possible dangers of altered media.

This study [44] tackles the problem of identifying image forgeries, which is important since it can distort public perception, harm people’s reputations and impede

legal proceedings. To improve the detection of manipulated images, the suggested solution makes use of a deep learning framework and a modified Xception network architecture combined with DenseNet features. 140000 images, divided equally into halves representing actual and synthetic faces, make up the main data set utilized to train and test the model. The false pictures, which show a range of image editing techniques, are created using StyleGAN, while the real images are taken from Flickr. The study involves the use of several ML models and algorithms, including DenseNet Integration and Modified Xception Network. Using several mathematical techniques such as density connections, depth-wise separable convolutions, and color space transformation. Detecting image forgeries, the model outperforms other models trained in different color spaces such as RGB, HSV, and XYZ, with an astounding accuracy of 99.41%. The goal of the project is to create a highly accurate technique for detecting forged images, which will be useful in cybersecurity applications and legal settings where digital image authenticity is important.

In a review paper [66], the authors review deepfake detection methods that use deep learning algorithms, which detect audio, video, and other multimedia sources. From their findings, they found that CNN was widely used for this purpose. They also found the usage of LBP, Auto-encoders, and LSTM-CNN to detect multimedia deepfakes. By reviewing the models, we found a lack of comparison and explanations of algorithms proposed in existing models. To conclude, Deep Learning and CNN methods are very important for detecting deepfakes.

This study [48] described the use of Vision Transformers for deepfake detection as accurate. This is a method of deepfake detection using Vision Transformers and SVM (Support Vector Machine) classifiers. Here, the latent and stable diffusion model methodology/data manipulation is used to for making datasets because there is a lack of open datasets. The celebA-HQ dataset is used for detecting facial tasks, and the FFHQ dataset is used for high-resolution image detection. Some part of the ImageNet-1K is used to generate an image-to-image method for the stability of the diffusion model. For open data sets, 'CIFAR-10, MNIST, FFHQ, and ImageNet' are used. To generate a deep fake including open datasets, GAN and VAE models have been used. For the fake detection model, CNN, optical flow, XGBoost, vision transformers, and different classifiers helped to improve accuracy. Here, SVM, KNN, MLP, Naive Bayes, and xDNN (samples, e.g. images of actual training data) classifiers have made the prophecy understandable. SVM is particularly useful for pre-trained features but others are useful for fine-tuned features. For identifying the model's prototypes the xDNN model generated a layer called megaClouds.

Novel D-CNN architecture is used in this study [56] because it has high accuracy for detecting real and deepfake pictures. Seven datasets are used as data preprocessing to increase generalization. In data preprocessing, the data enhancement technique was used. To enhance classification performance, a confusion matrix was implemented. D-CNN architecture follows batch normalization and average pooling layers. The evaluation process required an F1 score, precision, and recall values. ImageDataGenerator model is used for data enhancement, image resization, and assumption of the model. For feature extraction and classification, InceptionNet, DenseNet, and XceptionNet. CNN method requires different resolutions, like 1024

x 1023 to 178 x 218 for images. Data sets like CelebA, FFHQ, AttGAN, GDWCT, StyleGAN, StyleGAN2, and STARGAN are used here. The Keras preprocessing library is used for data manipulation like rotation, shape, zoom, and flips. The accuracy rate of the D-CNN model is 97.2% by using test datasets of real and deepfake images.

EfficientNets with varying sizes of pre-trained models are used in this study [30] because DFDC has become challenging because of the alarming growth of deep learning technologies. Here, the B4 and B5 models are used because they're efficiently accurate in video detection. The results showed that the size of the model and its efficiency did not consistently correlate, as B6 and B7 are larger models, so they have high computational complexities. That's why these two models show decreased accuracy. Larger models must be trained for a longer period in order to properly comprehend complicated patterns.

This study [22] presented a method called DeepVision where they employ the blinking of human eye method, which varies greatly depending on a number of pre-processing variables like gender, age, cognitive activity, and time of day. To differentiate between real from eye blinks, these operations are carried out at the frame level and compared with a database of real movements. This method allowed measuring the Eye Aspect Ratio(EAR) to detect the slightest blink of an eye. The EAR algorithm performs calculations based on the ratio of the left and right eye. Changes in EAR over time helped determine the frequency and duration of blinks to detect deepfakes. Through comparison and analysis of the blinking information which is measured in the database, integrity verification is performed. This algorithm uses machine learning and heuristic methods for verification. The accuracy of DeepVision detection through the blinking of the eyes is 87. 5%. Integrity verification methods faced obstacles due to continuous improvements in the GAN model. The pixel-based method helped to overcome the limitations of algorithms.

This study [45] used CNN and LSTM methods for deepfake detection since these models addressed the challenges of detecting fake videos altered by AI. RestNext and LSTM methods were used for fake video detection and CNN to extract features. These models help to prevent the spreading of false information through mass media. Image feature detectors, SVM regression, and GAN algorithms helped as detection methods. Celeb-Df dataset was used for training this model, and efficiency was analyzed by AUC score. Here, the constructed model is compared with cutting-edge techniques and indicates that the suggested model shows better performance than the previous model, which is shown by a higher AUC and accuracy level. So the accuracy level of the model is 91%, which is efficient.

The research paper [29] talks about deepfake where they focused on face manipulation in photos and videos. They used semantic segmentation for face identification networks and face context analysis for context recognition. Not only did they outperform Xception-Net, but they also got the best scores in AUC on all benchmarks like Celeb-DF-v2. They also outperformed the face identity difference baseline under attacks of laundering. However, due to limitations, their methods lack temporal coherence in video apps. There was also difficulty in reconstructing and manipulat-

ing, especially in temporal constraints in video, repeatedly. Overall, they proposed a novel detection model that had promising results.

The study [37] proposed a semi-supervised GAN-based deepfake detecting architecture in their research. They also did a comparative analysis of different deepfake detection models as well. For their model, they used the Flickr-Faces-HQ (FFHQ) dataset and the Deepfake Detection Challenge Dataset. A combination of these datasets produced 92.3% accuracy in their result, which has around 40,000 images. However, their model becomes less accurate when fewer datasets are used.

The authors did a research [12] where they used the eye blinking detection method to detect deepfake. They used CNN-based classifiers to LRCN, to detect temporal changes like eye blinking for example. They also evaluate based on the eye-aspect ratio (EAR). These resulted in promising results in several deepfake videos they tested this on. However, the EAR method uses the eye method, which is not always detectable, and CNN models are less effective against temporal consistency and small eye areas. Overall, their model seems like a simple way to approach deepfake detection, though it only relies on one aspect of the face, and not on others, which are also deepfakes.

This research [58] explores the detection of machine-generated tweets, particularly deepfake tweets. The authors propose a methodology that leverages a CNN architecture in combination with FastText word embeddings for classifying tweets. This research utilizes the TweepFake dataset containing 25572 tweets. The use of CNNs in this study gives the opportunity for the automatic learning of significant features from text inputs, facilitating the extraction of meaningful representations from the text. Their approach demonstrates impressive performance by achieving a 93% accuracy rate in the identification of fake tweets, surpassing other baseline methods and deep learning models' performance, such as LSTM and CNN-LSTM. For performance evaluation, the F1 score is employed, providing a comprehensive system combining recall and precision. The study also benchmarks the proposed method against state-of-the-art models like ROBERTA and BERT, further highlighting its effectiveness with a 93% accuracy rate in detecting deepfake text. The paper underscores the critical need for stronger techniques for deepfake text detection for combating misinformation on social media. It also suggests future research directions, including the exploration of quantum NLP and advanced methodologies for enhancing detection systems.

This study [54] delves into how GANs are employed for making deepfakes on social media, emphasizing the societal challenges and impacts of this technology. The celeb A dataset, which has over 200K images of public figures, is run in this study. It describes the implementation and detection of deepfakes using Deep Convolution-based GAN models and compares these with existing models by assessing Inception Score and Fréchet Inception Distance metrics. The study shows the proposed GAN model's effectiveness through performance graphs, showing improvements in accuracy, discriminator loss, and generator loss with continued training iterations. Additionally, the model proves to be efficient in deepfake detection, achieving high accuracy and addressing issues such as mode collapse and convergence. The paper

underscores the need for further research to improve GAN models' handling of small datasets and their generalization capabilities, aiming to advance deepfake detection on social media.

This research presents [20] presents a new method to combat Deepfake images by identifying Convolutional Traces(CT) left by Generative Adversarial Networks (GANs) during image creation. The study uses a dataset comprising Deepfakes from ten different GAN architectures, [CYCLEGAN], STARGAN, ATTGAN, GDWC, STYLEGAN, STYLEGAN, PRO-GAN], FACEFORENSICS++, IMLE, and SPADE, including those generated by FACEAPP, to assess the effectiveness of the CT extraction technique. The proposed model, built upon the Expectation-Maximization algorithm, is designed to extract the CT fingerprint from images, allowing for the identification of Deepfakes and the specific GAN architecture responsible for their creation. The results show that CT has a high discriminative power, achieving over 98% classification accuracy across different GAN architectures. Notably, when tested on FACEAPP-generated Deepfakes, the CT method achieved a 93% detection accuracy, demonstrating its robustness and effectiveness in practical scenarios.

This research [41] delves into existing studies on identifying misinformation sources, estimating tweet retweet rates, and detecting deep-fakes on social media platforms. Various types of fake content, including text-based fakes and deepfakes, spread across social media, necessitating advanced detection methods. Researchers have proposed numerous techniques for detecting deepfakes, including dual CNN-based methods, deep neural networks, and analyzing eye-blinking patterns. To tackle these challenges, researchers have developed datasets and models. The authors introduced a dataset of deepfake texts from Twitter to test detection algorithms' generalizability. They created an mBERT-based architecture that excels at identifying fake COVID-19 tweets in Hindi and Bengali. Identifying the sources of fake news is crucial for mitigating the damage caused by misinformation, leading to algorithms designed to trace the origins of rumors on social networks. Retweet proneness has also been extensively studied, with researchers examining factors such as tweet content, user profiles, and the impact of various media types on a post's popularity. The paper proposes the DeSI algorithm to pinpoint the origins of deep fakes on Twitter, using both text and image data. This algorithm has been rigorously tested in various experimental setups. Encased within a user-friendly Web portal, the DeSI algorithm effectively identifies fake tweets and their sources, demonstrating high accuracy in controlled environments. This research contributes to the larger effort to fight against misinformation on social media platforms.

DeepFake detection is crucial for combating misinformation and defamation, but even advanced neural network models remain susceptible to adversarial attacks. These attacks were studied in a black-box setting, where attackers lack full knowledge of the classification models. The study [28] investigated the adaptability of adversarial disturbances among various models and suggested methods to improve this adaptability through the use of Universal Adversarial Perturbations, which produce easily shareable attacks among attackers and expose the weaknesses of the Deep Fake detectors in use today. Adversarial videos were created using the initial 100 Real and 100 Fake videos from the public DFDC validation set, comprising 30,300

frames under diverse conditions. Victim models approached DeepFake detection as a per-frame classification task, utilizing face tracking and CNN classification models similar to the top entries in the DFDC challenge. Various Deep Fake detection methods were evaluated, including those by Team Selim, Team WM, and NTech Lab, each employing different CNN architectures and face detection models. These methods achieved AUC scores between 0.717 and 0.724. This research underscores the importance of developing resilient defenses against adversarial attacks to maintain the effectiveness of Deep Fake detection systems in real-world scenarios.

In this paper [24], numerous studies have investigated methods for estimating PM2.5 and PM10 levels, using a variety of techniques ranging from physics-based models to automated ML and deep-learning methods. Initial attempts involved estimating the Ambient Scattering Coefficient from images and using linear regression to predict PM. ML algorithms like FEBM and SILM used image features for estimation, whereas deep learning models like ResNet18 and VGG19 used CNN to improve accuracy. The ResNet-LSTM model, which combines ResNet and LSTM networks, was designed to capture both spatial and temporal dependencies in sequential smartphone images for real-time PM estimation. This approach showed excellent outcomes, with a 6.74% and 6.56% reduction in SMAPE and MAE for PM2.5 calculation, and a 11.03% and 13.25% reduction in SMAPE and MAE for PM10 calculation, accordingly Other techniques looked into include the Haze-Image and Dark Channel models Deep Convolutional Neural Fields for depth estimation, and Parallel CNN models for analyzing different image aspects. Results indicate that deep learning models, particularly those incorporating temporal information and domain-specific features, show promise in overcoming the limitations of earlier methods and providing more accurate and robust PM2.5 and PM10 estimations.

The paper [18] addresses the limitations of DenseNet in capturing spatial correlations by introducing a new architecture called Multiple Feature Reweight DenseNet (MFR-DenseNet). This model enhances DenseNet through adaptive feature recalibration, using the Squeeze-and-Excitation Module for channel-wise feature recalibration and DSEM for modeling interdependencies among convolutional layers. By integrating CFR-DenseNet and ILFR-DenseNet, which utilize SEM and DSEM respectively, through ensemble learning, the MFR-DenseNet achieves significant performance improvements. Experiments on CIFAR-10 and CIFAR-100 datasets show a 3.57% error rate on CIFAR-10 and an 18.27% error rate on CIFAR-100, representing a 4.5% and 5.09% improvement, respectively, over traditional DenseNet models, with fewer parameters. These findings underscore the effectiveness of feature reweighting in boosting classification accuracy while maintaining computational efficiency. The authors suggest that increasing model’s depth and width could yield further performance gains, though scaling up may introduce computational challenges.

The paper [17] proposes a new deep-learning architecture called VCF for edge detection in images. The authors contend that conventional edge detection techniques struggle to capture intricate details, especially in complex scenes. Their approach leverages a modified VGG16 network, chosen for its strong feature extraction capabilities. They introduce three key improvements to the standard VGG16 architecture: reducing feature dimensions progressively with smaller convolutional ker-

nels, employing a custom loss function, and implementing a cross-fusion structure. These modifications aim to enhance the network’s ability to identify both high-level and low-level edge features accurately. The authors validate their method on the BSDS500 and NYUD V2 datasets. Their results demonstrate that VCF outperforms previous VGG16-based models like HED and RCF, achieving an ODS value of 0.808 and an FPS of 30+ on BSDS500. This suggests that VCF can locate edge pixels with higher precision and reduce the range of controversial pixels. However, the paper acknowledges limitations. While VCF shows promise, the authors highlight the need for further exploration to reduce reliance on large datasets and improve performance on images with low contrast or significant noise.

The paper [27] investigates the use of CNN, ResNet, and DenseNet for landslide detection mapping (LDM) in the Three Gorges Reservoir region, China, a landslide-prone area. High-resolution ZY-3 satellite images and 19 conditioning factors (CFs) such as elevation, slope, and geological data were used to train these models, with datasets divided into 70% training and 30% testing sets. The study shows that all models achieved high accuracy in landslide detection, with DenseNet outperforming CNN and ResNet. DenseNet, particularly when using both remote sensing images and CFs, achieved the highest accuracy (99.95%) and Kappa coefficient (0.9933), owing to its feature reuse ability and fewer parameters. Despite the success, limitations include the study’s focus on a specific geographic area, raising concerns about its generalizability to other regions. Furthermore, the authors suggest incorporating more advanced architectures like self-attention mechanisms and real-time data to enhance detection performance. DenseNet’s computational demands, although efficient, pose challenges when scaling for larger or more complex areas.

In this paper [67] looks into the use of transfer learning to classify two visually similar mackerel species, *R. kanagurta* (known as Indian mackerel) and *R. brachysoma* (known as short mackerel) using images captured on smartphones. This method uses pre-trained models like MobileNetV3Small, Xception, InceptionV3, VGG19, ResNet50 and VGG16 eliminating the need for controlled environments or specialized equipment. The dataset contains 517 images collected from five fishing zones in Thailand, and data augmentation methods were used to improve training. Xception outperformed the other models, with a training accuracy of 84.9%, validation accuracy of 75.4%, and testing accuracy of 83.3% after 30 epochs of fine tuning. This result marked an 11.508% improvement in predictive performance, surpassing other models in accuracy, precision, and recall. While the results demonstrate the viability of using smartphones for real-time fish species classification, the study acknowledges certain limitations, such as the need to validate the approach with a broader range of species and expand the dataset for greater robustness. The authors suggest developing a mobile app to help identify fish species in real-time for field use.

The paper [23] presents an enhanced deep-learning approach for detecting cracks in concrete pavement images. The study improves upon traditional methods by leveraging a modified VGG16 architecture optimized for crack feature extraction, coupled with a cross-entropy loss function for precise pixel-level classification. The proposed model was trained and evaluated using three diverse datasets—CFD, CrackTree200

and DeepCrack—which encompass various crack patterns and background complexities, thus enhancing the model’s robustness. The results show that the improved VGG16 significantly outperforms other techniques, such as U-Net and Percolation, with F1 scores of 0.896 on CFD, 0.892 on CrackTree200, and 0.901 on DeepCrack. These scores reflect the model’s effectiveness in accurately detecting and segmenting cracks, even in challenging images with noise, unclear boundaries, and complex backgrounds. However, limitations remain, particularly under difficult conditions such as uneven illumination and shadows. The authors propose that future studies should prioritize improving image preprocessing to reduce noise, exploring applications on other surfaces like glass and metal, and incorporating faster models like YOLO for real-time detection. They also recommend integrating the model with unmanned aerial vehicles for large-scale pavement inspections.

The study [61] evaluates five deep learning models for detecting rice blast disease, a significant risk to global food security. The research uses a dataset of approximately 2000 rice leaf images, both healthy and infected, to train models like Inception V3, VGG16, and ResNet50 through transfer learning with pre-trained ImageNet weights. ResNet50 achieved the highest accuracy at 99.75%, benefiting from its connection-skipping architecture, which prevents network degradation and enhances performance. While VGG16 demonstrated strong results with 98.47% accuracy, it was less effective than ResNet50 due to its higher computational demands and vulnerability to variations in new data. The findings highlight the promise of deep learning, especially ResNet50, for automated and accurate disease detection. However, the authors note limitations, such as the necessity for larger and more diverse datasets to improve generalizability across different environments and rice varieties. They also suggest integrating the approach with mobile or drone technologies for real-time field monitoring, improving practical application in agriculture.

This paper [21] explains how GAN-generated fake images can lead to major problems, particularly on social media where they could be misused. Because AI-generated fakes are made entirely from scratch rather than altered from real images, traditional approaches have difficulty detecting them. The authors suggest paired learning a deep learning-based approach that analyzes genuine and fake images in pairs. Their DenseNet and contrastive loss-based approach effectively determines if an image is real or fake. When compared to current techniques, the authors discovered that their suggested approach greatly improved the detection of fake images. Even if the fake image was created by a GAN the system had never seen before, their method was able to learn features that could differentiate between real and fake images by employing pairwise learning. Their solution performed better than other widely used techniques in experiments in terms of accuracy and recall rates. This shows that their approach is not only more effective but also better at generalizing to new types of fake images. They also highlighted that the system could identify specific areas in an image where the fake features were most visible making it easier to understand why an image was classified as fake.

The paper [63] discusses the issue of deepfakes which are manipulated images and videos that look real but are fake. They evaluated three AI models which are VGG16, VGG19, and ResNet50 on a dataset containing both real and fake images. After

testing, the authors found an accuracy of 94% and it performed well but was not the best among the three. Another model attained the highest accuracy of 98.5% making it the most effective model for detecting deep fake images in this study. Its additional layers helped capture more complex details. Lastly, this model achieved a slightly lower accuracy of 92%, which was slightly lower than the other two. This model has more complex features, but it did not perform as well, possibly due to the smaller dataset size. The study highlights VGG19's superior performance, showing its potential as a reliable tool for identifying fake images. The study aims to address this by comparing three AI models to find the most effective one for deep face detection.

The paper [34] likely focuses on using two well-known AI models Xception and ResNet50 to quickly classify deepfake. The study explores the effectiveness of these models in detecting deepfakes by applying image processing techniques specifically Local Binary Patterns. The main goal is to enhance the detection speed and accuracy of distinguishing real from manipulated ones. The study probably discusses the challenges of deepfake detection where AI creates realistic but fake visuals. Combining the power of pre-trained deep learning models which are Xception and ResNet50. The research highlights the importance of addressing deep fakes due to their potential for misuse in spreading misinformation. It is expected that Xception might perform better due to its architecture which has been effective in image processing tasks. ResNet50 might offer competitive results but could be slightly slower. The results would highlight which model is more effective for real-time applications and it shows accuracy rates above 90%.

This paper [35] focuses on detecting deep fake manipulated videos and images that use AI to create real but fake content. This study introduces a new model called ViXNet which combines two advanced techniques. One is Vision Transformers and another is Xception Network. The goal is to improve the accuracy of detecting deepfakes by effectively analyzing visual data. The growing concern over deep fakes highlights how they can be used to spread misinformation which has raised serious security and ethical issues. Accurately recognizing these forgeries is difficult particularly as they get increasingly complex. The study suggests ViXNet as a solution to this problem combining the advantages of both ViTs and Xception Network. Here ViTs manage picture data effectively and Xception Network works excellently at spotting minute modifications. The results showed that ViXNet performed well in detecting deepfakes and achieving high accuracy. Specifically, their model reached a performance rate of over 90% and demonstrated its effectiveness in identifying manipulated content. The paper concludes that ViXNet could be an effective solution in addressing the escalating challenges posed by deepfake technology.

The paper [69] focuses on improving the accuracy of detecting deepfakes AI-generated fake videos and images that can manipulate facial features. To overcome the shortcomings of the existing deepfake detection algorithms, the authors suggest a new approach that combines ensemble and transfer learning methods. Here, transfer learning helps by using pre-trained models that have already learned patterns from extensive datasets and making it easier to identify manipulated content. Ensemble techniques combine multiple models to improve the overall detection accuracy. To

improve the detection performance, this paper presents a novel strategy that makes use of transfer learning to make use of pre-existing knowledge from previous models and mixes it with ensemble approaches. Experimental results demonstrate that the proposed method achieves superior accuracy in detecting deepfakes surpassing many existing methods. Specifically, their model reached an accuracy rate of around 98%. Also, proving its effectiveness in identifying fake facial features. The authors come to the conclusion that deep fake detection systems can become much more reliable and adept at confirming authenticity by combining ensemble techniques with transfer learning.

This paper [49] concerns the proliferation of deepfake photos and videos, which can be exploited for fraudulent purposes. In the medical field, altering medical images may result in incorrect diagnoses and perhaps incurable outcomes. Therefore, the primary focus of this research is to tackle the challenge of detecting the handling of cancer and other disease samples. The study explores the application of deep learning algorithms to identify deepfake images, aiming to develop reliable methods for distinguishing between authentic and manipulated visual data. For the purpose of training a model that can distinguish between authentic and fraudulent medical images. This study suggests a framework that makes use of cutting-edge large datasets of authentic and deepfake medical images as well as deep convolutional neural networks. The paper specifically trained 6 CNN models: MobileNetV2, MobileNet, DensNet121, DenseNet201, ResNet101, and ResNet50. These models were evaluated against a number of cutting-edge deep fake detection models after being trained on 2000 data from three classes. Which is Untampered, FalseBenign, and False Malicious. By including more layers, the suggested model improved ResNet101 and achieved 99% training accuracy. The results of this investigation demonstrate almost flawless precision in identifying cases of tumor injections and excisions.

This paper's [15] significant advances in machine learning, computer vision, and image processing techniques that make it increasingly harder to detect images of phony or disguised faces. Furthermore, the proliferation of deep fakes has made it possible for hostile actors to employ phony images to target people and dissuade them from sharing accurate information. As a result, it is essential to develop a classifier that correctly separates a person from other or similar people. In this initial effort, they use the Disguised Faces in the Wild 2018 dataset to identify a target person's face among several similar persons. They assess the classification performance using popular normal face detection classifiers including ShallowNet, VGG16, and Xception. They use data augmentation to further enhance the detection performance. According to our initial findings, the Xception model has a 62% accuracy rate in classifying one person from another.

This paper [16] using the NUAA Photograph Impostor dataset, this paper introduces a deep learning method for face liveness detection that achieves 100% accuracy in classifying images as real or fake by combining texture analysis and a convolutional neural network.

The article [38] discusses the development of a robust face recognition system using a combination of Haar cascade, CNN, and Xception to achieve high accuracy in

face detection and recognition, overcoming issues with varying sizes and aspect ratios, noise, and illumination. The accuracy rate is 96.73% on the benchmark dataset FERET and 98.45% on the LFW dataset, exceeding existing methods. The AdaBoost algorithm is used to reduce the number of features, and Xception and DenseNet (CNN Model) used for the recognition process. In wild data sets face images are displayed through rigorous formalization techniques that consistently aim to capture the frontal aspect of a picture. HOG ICS LBP SHIFT and color-dominant descriptors for color texture and orientation are some of the multifocal face detection and function extractor techniques used in the study. To eliminate noisy elements from the photos the study also employs preprocessing techniques.

This study [47] uses a Convolutional Neural Network (CNN) with Xception architecture and transfer learning to classify facial images based on face shape, achieving an accuracy rate of 96.2% in training and 81.125% in validation. This study tests the developed model using fresh data and the results indicate an accuracy rate of 85 percent in facial image classification. Possible real-world uses for the study include creating facial recognition authentication systems that are quicker and more precise.

The study [46] aims to use deep learning algorithms to classify and detect biometric spoof attacks, which involve the use of fake photos, often sourced from social networks or social networks, to impersonate individuals and compromise their credentials or passwords, leading to cyberattacks. Accurate outcomes are obtained by extracting facial features using MobileNetV2 VGG16 ResNet50 and standard CNN models. The study of this work for detecting spoofing attacks Accepts a subset made available by Kaggle that includes 20,000 high-definition images of real faces and photoshopped ones that were fixed by scammers. The images are in the PNG format and the pixel resolution of each pixel is 1024 x 1024, with the fake ones being photographs edited with multiple Photoshop features. For the sake of the validation and training set the data augmentation is applied. The dataset is divided into training and validation sets in a 7:3 ratio. With the data augmentation function, the images are adjusted in scale and are also flipped in the horizontal and vertical directions. After achieving a trained MobileNetV2 model for 20 epochs, the classification report and accuracy are provided and these are 94% and 17% losses respectively. As for Vgg16 and ResNet50, the epochs were the same but we observed training accuracies of 85.16% and 66.20% respectively. The losses are 38.61% and 60.10%.

In this paper [26], the best way to add facial biometrics to the current username/password standard is determined by comparing the performance of eight deep learning-based facial recognition technologies. VGG16 and VGG19 performed best (more than 80% accuracy) among 8 CNN models in terms of F1 score and image classification accuracy. According to the study, most CNN models see improvements in classification report metrics and overall image accuracy when the number of testing images is increased. AlexNet and Xception show the biggest gains. With a smaller training set, VGG-16 and VGG-19 did however perform marginally worse. Here a balanced number of images and labels have been used from the Labeled Faces in the Wild (LFW) datasets. To assess how well the eight CNN models (AlexNet Xception VGG-16 VGG-19 ResNet50 ResNet101 Inception v2 and Inception v3) perform on

facial recognition tasks, the study uses different training and testing splits. According to the study, both recall and F1 scores significantly decreased as a result of the imbalanced dataset. In the future, the study intends to address these problems by incorporating more balanced datasets. The use of a single dataset (LFW) and CNN models default parameters which might not be ideal for the task are the study's limitations.

In this paper [72] the integration of an approximate multiplier, the Signed Carry Disregard Multiplier (SCDM8), useful for reducing energy while performing computations in CNN, is presented. In experiments based on VGG16, ResNet152 and ConvNeXt-T models, SCDM8 achieved a significant power consumption reduction is 35% for VGG16 and 42% for ResNet152 is 42%, with only a small drop in accuracy (1.5% and 1.8%, respectively). For ConvNeXt, 72% energy savings were measured with a 1.3% drop in Top-1 accuracy. The purpose of this study is to demonstrate the ACE-CNN method parameters as the most optimal for the qualitative and quantitative efficiency trade-off, relevant to low-power deep learning-oriented applications. Various studies demonstrated the benefits of such techniques, including significant reductions in energy consumption and hardware costs, while maintaining or slightly improving accuracy. Techniques such as replacing exact multipliers with approximate designs led to hardware cost reductions and up to a 0.63% accuracy improvement. Research on ResNet, Inception-v4, FPGA-based, and multiplier-less designs highlighted energy savings, reduced critical path delays, and power consumption, all contributing to more efficient neural network computations. they tested with VGG16, VGG19, ResNet101, ResNet152, MobileNetV2 and InceptionV3 for the image classification task to demonstrate the practicality of using their proposal approximate in real application instead of ConvNeXt-T (Tiny). This allowed them the freedom to use a range of pre-trained CNN models from the Python TensorFlow library. They used this possibility by using a few of the pre-trained CNN models from TensorFlow library in Python. More concretely, we implemented VGG16, V19ResNet101, ResNet152 and MobileNetV2 (which are well-known models in the literature for achieving good results across different computer vision tasks) alongside InceptionV3architecture using ConvNeXt-T since these models have strong performance in image classification

This paper [71] investigates ResNet-50 models for classifying artificial intelligence (AI) generated images versus natural images. It focuses on generators such as DALL-E Midjourney and Stable Diffusion and compares model accuracy training and inference complexity. With accuracy surpassing 99 percent in certain instances transfer learning considerably outperformed models trained from scratch. ResNet-50 achieved over 90% accuracy with unified training, and the gated network model achieved 92% accuracy across generators. The study can be used for content moderation and cybersecurity, but it has trouble generalizing to other datasets.

Table 3.1: Summary of Research Papers on Deepfake Detection

Reference	Methods Employed	Findings	Limitations
Zhao et al. (2021) [33]	AAC & AUC, Loss and attenuation guided data augmentation	Improved performance, Outperforms binary classifiers	Fails to detect subtle variations in images
Heidari et al. (2024) [66]	CNN, LBP, Autoencoders, LSTM-CNN	CNNs widely used for better performance	Lack of comparison and explanations of algorithms
Jung et al. (2021) [29]	Semantic segmentation, Face context analysis	Outperformed Xception-Net, Best AUC scores on all benchmarks	Lacks temporal coherence in video applications
John et al. (2022) [37]	FFHQ, Deepfake Detection Challenge Dataset	92.3% accuracy with 40,000 images	Lower accuracy with fewer images
Li et al. (2018) [12]	CNN-based classifiers to LRCN	Promising results on deepfake videos	Eye detection method not always effective
Shad et al. (2021) [31]	CNN, DenseNet121, VGGFace, ResNet50	VGGFace 99% accuracy, ResNet50 97%	Results not fully verifiable
Nawaz et al. (2023) [55]	ResNet-Swish-Dense54, OpenFace2 toolbox	99.26% accuracy on DFDC, consistent on CelebDF	Limited capacity to capture temporal changes
Raza et al. (2022) [43]	Using a deepfake predictor (DFP) technique, VGG16 and convolutional neural network (CNN) architectures are combined., Xception, NAS-Net, Mobile Net	The suggested DFP approach outperformed existing transfer learning strategies and cutting-edge research achieving a high accuracy of 94% and precision of 95% for deepfake detection.	The model has a limited capacity to capture temporal changes within frames.
Sahib et al. (2022)[44]	Xception, DenseNet, YCbCr color system	The proposed method outperformed other color systems like RGB, HSV and XYZ in detecting fraudulent images with an accuracy of 99.41%.	To obtain good accuracy additional training epochs were needed because of the reduced image dimensions.

Reference	Methods Employed	Findings	Limitations
Aghasanli et al. [48]	FFHQ, ImageNet-1K KNN, xDNN, CNN, XGBoost	Interpretability using XDNN and SVM for analyzing model predictions.	Insufficient amount of open datasets for detecting deepfakes.
Patel et al. (2023) [56]	FFHQ, GDWCT, AttGAN, StyleGAN, and StyleGAN2	MesoNet and MesoInception networks, achieved 77% accuracy. The CNN model detects images with 97.2% accuracy.	A model's accuracy decreases when it is trained on several data sources.
Pokroy et al. (2021) [30]	EfficientNets, B4, B5, B6, and B7 models.	B4 and B5 models show high accuracy (72.7% and 74.4%) in detecting videos.	B6 and B7 EfficientNet models show less accuracy due to their high computational complexity.
Sahib et al. (2022) [22]	GANs model and pixel-based method	DeepVision and Integrity verification through eye blinking is 87.5% accurate in detecting videos.	Continuous improvements of GAN models create obstacles to the integrity verification method.
Vamsi et al. (2022) [45]	CNN and LSTM methods are used and RestNext, SVM, GAN	The accuracy rate of the CelebDf dataset is 91%.	RNN model makes it more difficult to transfer information across lengthy sequences efficiently.
Sadiq et al. (2023) [58]	CNN, LSTM, CNN-LSTM	Outperforms all other models, achieving 93% accuracy, 92% precision, 95% recall, and a 93% F1 Score.	CNN might have trouble keeping up with modifying textual data and figuring out new ways to manipulate content on social media.
Kumar et al. (2023) [54]	GAN, CelebA dataset, DC-GAN, C-GAN, S-GAN	GAN model outperforms existing models and Fréchet Inception Distance values of 1.074 and 49.3 respectively.	Limitations like mode collapse and convergence issues in GAN training.
Guarnera et al. (2020) [20]	CYCLEGAN, STYLEGAN, STYLEGAN, FACEFORENSICS++, IMLE, and SPADE	Deepfakes from real images, achieving over 98% classification accuracy across GAN architectures.	The employed CNN showed limitations in robustness.
Narayan et al. (2022) [41]	MTCNN, LightCNN-29, XceptionNet	The algorithm accurately identified all deepfake tweets and their sources in a controlled experimental setup.	In unconstrained experimental setups, the algorithm's performance might be hindered by the diversity of deepfake content.
Neekhara et al. (2021) [28]	Iterative gradient sign method, universal adversarial perturbation	Adversarial attacks were successful in evading sophisticated DeepFake detectors.	Targeting the winning entries of the DFDC challenge may not fully capture the diversity of DeepFake detection.

Reference	Methods Employed	Findings	Limitations
N. Qazi et al. (2024) [69]	VGG16, DenseNet201, Xception	Learning has achieved 98% accuracy for models ConvNext-Tiny	If dataset lacks certain types of manipulated images, the model might not be able to detect them effectively.
J. Kim et al. (2019) [15]	ShallowNet, VGG16, Xception	Xception model can classify one from different individuals with a 62% accuracy.	If the dataset doesn't include certain types of disguises, then the model may fail to recognize them.
S. Ganguly et al. (2022) [35]	Vision Transformer, Xception Network	ViXNet model achieved an accuracy of over 90%.	If the dataset does not cover certain types of forgeries, then ViXNet may struggle to detect them accurately.
C. C. Hsu et al. (2020) [21]	DenseNet	The model using pairwise learning achieved an accuracy of 97.5%.	The model might not be generalized well to all types of deepfake forgeries.
Z. N. Ashani et al. (2024) [63]	VGG16, VGG19, ResNet	Accuracy rate of 95% while VGG16 and VGG19 had slightly lower accuracy rates.	The dataset does not contain diverse examples of deepfakes, and models might struggle to generalize manipulated images.
A. Arini et al. (2022) [34]	Xception, ResNet50	Xception with LBP achieved 98.7% accuracy, and ResNet50 with LBP 97.2% accuracy.	Deepfake techniques are evolving, and more sophisticated forgeries may not be detected as accurately.
A. Alsaheel et al. (2023) [49]	ResNet101, ResNet50, DenseNet201, MobileNet	Detection accuracy of 96% using the ensemble of CNN models.	Each type of medical image has unique characteristics, so the model might not perform equally well.
S. Song et al. (2020) [24]	ResNet LSTM	13.25% and 13.48% reduction for PM10 estimation compared to the best baseline.	Limited range of PM2.5 and PM10 values in their dataset, particularly lack of very low values in the training set.
K. Zhang et al. (2019) [18]	DenseNet, CIFAR-10, CIFAR-100	DenseNet achieves a 3.57% test error on CIFAR-10.	Scaling up may introduce computational challenges.
Z. Qu et al. (2019) [17]	VGG116, BSDS500, NYU Depth V2	Models like HED achieving an ODS value of 0.808 and an FPS of 30+.	On large datasets, models improve performance on images with low contrast or noise.

Reference	Methods Employed	Findings	Limitations
T. Liu et al. (2021) [27]	CNN, ResNet, DenseNet	The highest Kappa coefficient F1 score and generalization ability for LDM.	Data dependency, limited scope, computational demands.
R. Jongjaraunsuk et al. (2024) [67]	ResNet-50, Xception, VGG16	Xception model achieved the highest training accuracy of 84.9% and testing accuracy 84.3%.	Data augmentation trade-offs limited the range of data, specificity to two species.
Z. Qu et al. (2020) [23]	VGG116, CCD1500, CDD861	VGG16, U-Net, and Percolation algorithms achieved an F1 score of 90.1% on DeepCrack.	Dependence on image quality, computational cost, and different types of concrete explored.
Shah, S. R. et al. (2023) [61]	VGG16, VGG19, ResNet50	ResNet50 achieved the highest accuracy at 99.75%.	The need for more diverse datasets to improve generalizability and rice varieties.
Koshy R et al. (2019) [16]	Inception v4, ResNet50	Inception v4 produced the optimal accuracy of 100% in liveness detection.	No limitation.
Saskoro et al. (2024) [71]	Xception, DenseNet	Accuracy rate is 96.73% on the benchmark dataset FERET and 98.45% on the LFW dataset.	If certain preprocessing methods weren't used, then the picture would have gotten noisy.
Adityatama et al. (2023) [47]	Xception and Transfer learning.	The accuracy rate is 96.2% in training and 81.125% in validation of the Xception model.	Some forgeries may be difficult for Xception to identify if the dataset does not include them.
Gwyn et al. (2021) [26]	VGG16, VGG19, Xception	The model using VGG16 and VGG19 has an accuracy rate of more than 80%.	The CNN model performance was limited because only default parameters were used.
Verma et al. (2022) [46]	MobileNetV2, VGG16	MobileNetV2 model achieved 94% accuracy, while VGG16 and ResNet50 training accuracy were 85.16% and 66.20%, respectively.	The dataset (LFW) and CNN models use default parameters, which might not be ideal for the task.
Amirafshar et al. (2024) [72]	VGG16, InceptionV3, ResNet101	Model VGG16 performed accuracy of 61%, while ResNets and MobileNets achieved accuracies of 76% and 45%.	InceptionV3 architecture had to use only ConvNeXt-T since these models have strong performance in image classification.

The table shows the reviews of research papers on the detection of deep-fakes that used different kinds of methods, algorithms, and their accuracy rates as well as limitations.

Chapter 4

Methodology

The methodology begins with an initial research and literature review to establish a strong theoretical foundation. This phase involves studying the current image classification methods, focusing on architectures such as Vision Transformers (ViT), DenseNet121, and other relevant models like ResNet50. The goal is to understand the strengths and weaknesses of these models, identify gaps in existing approaches, and determine how a hybrid model combining ViT and DenseNet could address these gaps. This review helps in defining the problem statement and setting clear objectives for the project.

Next, the dataset preprocessing phase is conducted to ensure the dataset is preprocessed using TensorFlow tools like ImageDataGenerator to ensure consistency and suitability for model training. This includes setting batch size, setting class mode to binary for binary classification, and applying data augmentation techniques to enhance generalization. Since the dataset is already grouped into training, validation, and testing images, we set the dataset path for each of the groups separately.

In the next phase, multiple pretrained models are sourced from Keras Hub or other repositories. These models are selected to serve as baselines and to leverage their pretrained weights, which can significantly reduce training time and computational resources. The models are prepared for fine-tuning or direct evaluation on the preprocessed dataset, ensuring compatibility with TensorFlow and the established preprocessing pipeline.

The creation of a custom hybrid model combines the strengths of ViT and DenseNet. The Vision Transformer (ViT) is utilized for its ability to capture global contextual information through self-attention mechanisms, while DenseNet is integrated for its efficient local feature extraction and feature reuse via dense connections. The outputs of these two architectures are merged to form a single unified model. This hybrid model is implemented using TensorFlow and Keras, with DenseNet initialized using pretrained weights from imagenet to leverage transfer learning. The training setup includes defining a loss function (e.g., binary cross-entropy), selecting an optimizer (e.g., Adam), and setting up learning rate scheduling along with early stopping to minimize overfitting.

During the training and testing phase, all models, including the pretrained ones and the custom hybrid model, are trained and evaluated. Each pretrained model is fine-tuned on the training set, with performance monitored on the validation set.

The custom hybrid model is also trained on the training set, using the validation set for performance monitoring. After training, all models are evaluated on the test set to measure their generalization ability. This phase ensures a comprehensive comparison of the models' performance.

The evaluation of results involves analyzing the performance of all models using metrics such as accuracy, precision, recall, F1-score, and confusion matrices. The performance of each model is compared to identify scenarios where the convolution-based models provide advantages or face limitations over attention-based models. This analysis helps in understanding the effectiveness of combining ViT and DenseNet.

Finally, the conclusion phase summarizes the findings and outlines potential directions for further research figure: 4.1.

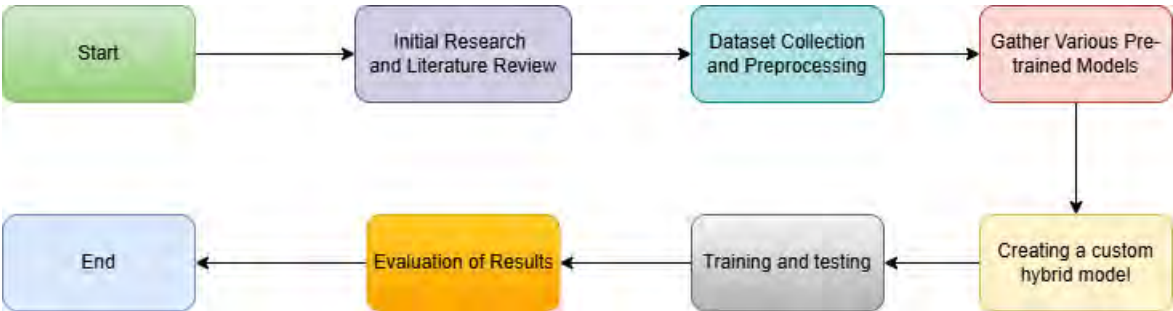


Figure 4.1: Methodology Dataflow

Chapter 5

Description of Model

5.1 Model Architecture

The first step involved gathering the dataset and extracting its features. Thirteen deep learning architectures were thus used, and they were assessed using five distinct evaluation metrics: recall, accuracy, precision, F1-score and average precision score.

DenseNet121: DenseNet121 (Dense convolutional network) [4] is a deep convolutional neural network for image recognition tasks with the characteristics of efficient parameters and reuse of features. To enable task-specific adjustments, we employed DenseNet121 as the base feature extractor in this case, excluding the top layers. DenseNet121, on the other hand, has dense connections, where every layer receives the input from all the previous layers. By removing the amount of input channels, computational efficiency increases for DenseNet121 with the help of bottleneck layers. Transition layers, such as batch normalization, 1x1 convolutions, and average pooling, are incorporated into DenseNet121 to control feature map sizes and minimize the number of parameters. To generate probabilities for binary classification (Fake vs. real) a sigmoid activation function was used to create a fully connected dense layer. Figure: 5.1 represents the architecture of the DenseNet121 model.

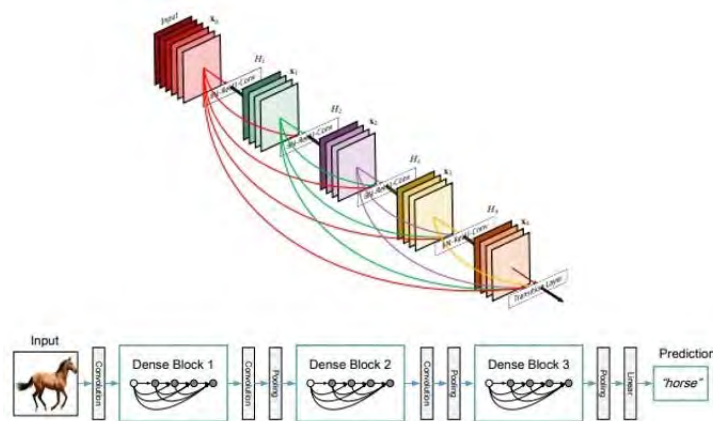


Figure 5.1: Architecture of DenseNet121 [4]

ResNet50V2: With 50 layers arranged into multiple residual blocks, ResNet50V2

[2] is a deep convolutional neural network intended for image recognition applications. The vanishing gradient issue is resolved, and very deep networks can be trained for the introduction of skip connections by residual blocks, which let the network avoid specific layers. The ResNet50V2 architecture pretrained on the ImageNet dataset is used in the deepfake detection model's transfer learning process. Transfer learning enables the model to apply features acquired from a sizable dataset to the particular deepfake detection task. The model classifies inputs as "Fake" or "Real" based on its binary classification structure. For the customization of binary classification, here we excluded the top layers of the ResNet50V2 model. $224 \times 224 \times 3$ was the input shape set to the resized image dimensions. The addition of a GlobalAveragePooling2D layer allowed the feature maps to be smaller in space while still preserving important data. Here a single neuron was added to a fully connected dense layer, which generated probabilities for the two classes (Fake and Real) using a sigmoid activation function. To simplify the design and preserve modularity, a sequential structure was used to implement the model. Through the integration of transfer learning with ResNet50V2 and well-crafted training techniques, the model demonstrated strong performance in the deepfake detection task. Figure: 5.2 represents the architecture of the ResNet50V2 mode.

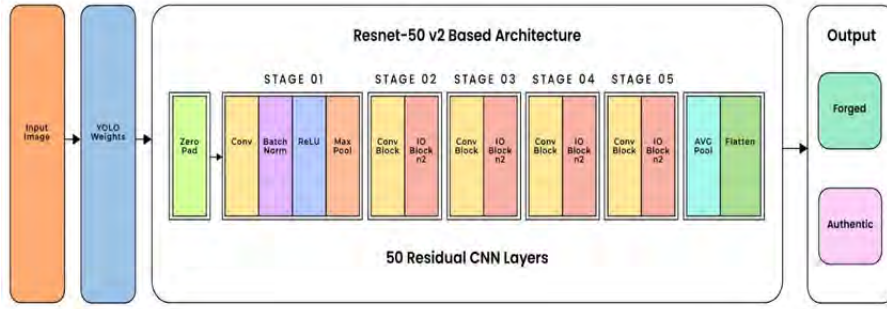


Figure 5.2: Architecture of ResNet50V2 [42]

MobileNetV3Large: MobileNetV3Large [14], a lightweight yet potent convolutional neural network optimized for effective feature extraction, is used in the deepfake detection model. Because of its optimized design, it is especially well-suited for computationally limited environments. Features are extracted with the least amount of computational overhead thanks to the architecture's use of effective building blocks like bottleneck layers and depthwise separable convolutions. By using weights that have been pretrained on the ImageNet dataset, the model offers a solid basis for feature extraction without the need for a large amount of training data. In order to customize it for a particular task, we also had to remove the top layers. A GlobalAveragePooling2D preserves the necessary spatial and contextual information while reducing the number of parameters. Because of MobileNetV3Large's lightweight architecture, the model operates effectively and can be deployed on devices with limited resources or used in real-time applications. Figure: 5.3 represents the architecture of the MobileNetV3Large model.

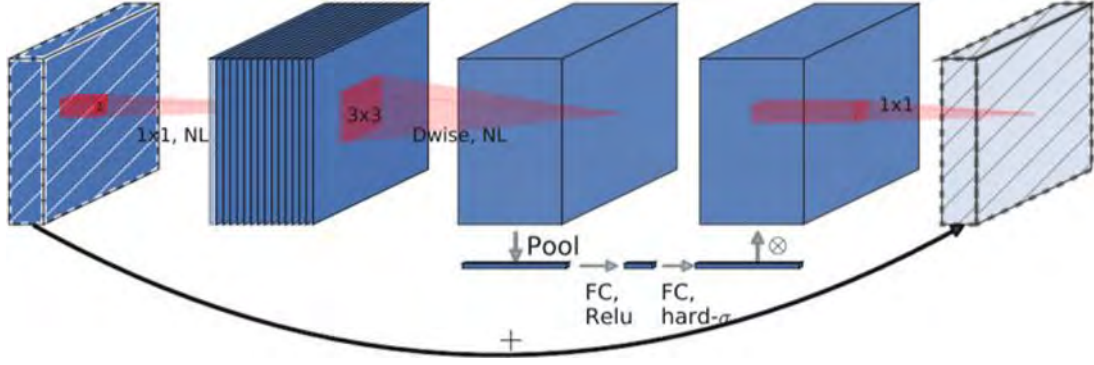


Figure 5.3: Architecture of MobileNetV3Large [65]

InceptionResNetV2: InceptionResNetV2 [6], a hybrid architecture that combines the advantages of residual connections and Inception modules, is used in the deep-fake detection model. Superior feature extraction and convergence capabilities are provided by a number of improvements introduced in this architecture. Adapted for binary classification, the model can distinguish between authentic and fraudulent content. Because of its depth and complexity, InceptionResNetV2 is able to capture hierarchical information and fine-grained features. The GlobalAveragePooling2D layer drastically lowers the number of parameters while keeping the most important features by condensing the spatial information into a single feature vector. Through the combination of transfer learning and personalized training techniques with the InceptionResNetV2 architecture, the model improved upon earlier architectures and demonstrated strong performance in identifying deepfake content. Figure: 5.4 represents the architecture of the InceptionResNetV2 model.

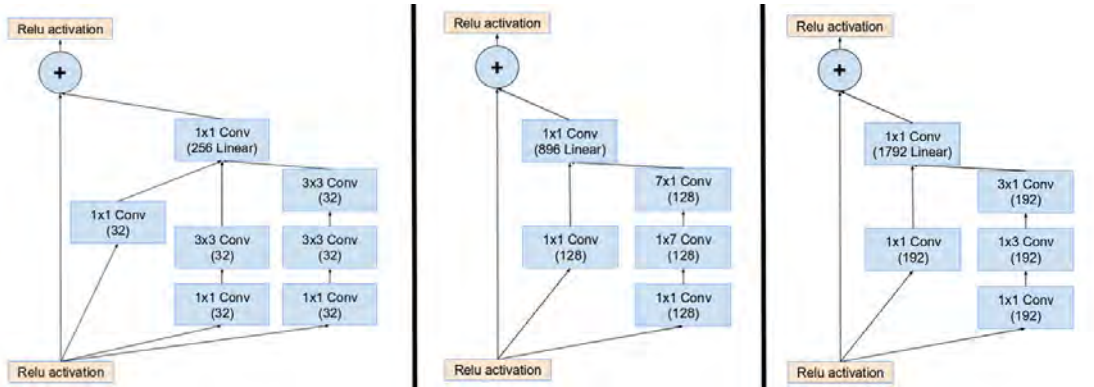


Figure 5.4: Architecture of InceptionResNetV2 [6]

EfficientNetV2M: The EfficientNetV2M [78] architecture, a powerful convolutional neural network tuned for increased speed and accuracy, is used by the deep-fake detection model. Binary classification is another method used in this model. By using a compound scaling technique, EfficientNetV2M ensures optimal performance across a range of resource levels by proportionally adjusting the model's depth, width, and resolution. By adding fused convolutions, EfficientNetV2M is an enhanced convolutional neural network that reduces overfitting and enhances convergence over the original EfficientNet. EfficientNetV2M outperforms larger models in environments with limited resources because it seamlessly adjusts to hardware

limitations. The model was able to balance high accuracy in identifying deepfake content with computational efficiency by combining the EfficientNetV2M architecture with transfer learning and customized training strategies. Figure: 5.5 represents the architecture of the EfficientNetV2M model.

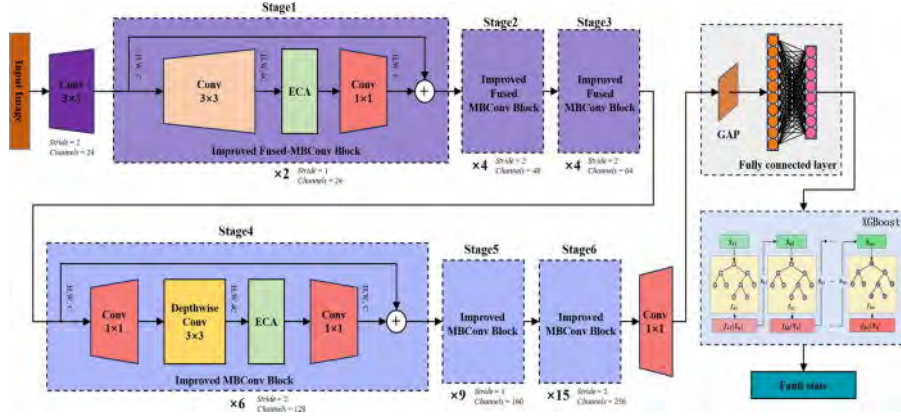


Figure 5.5: Architecture of EfficientNetV2M [74]

Xception: Extreme Inception is what the architecture is called. Building on the Inception architecture, Xception [3] uses depthwise separable convolutions in place of the typical Inception modules. Both feature extraction and computational efficiency are enhanced by this design. To differentiate between authentic and fraudulent content, the model is built for binary classification. By using weights that have been pretrained on the ImageNet dataset, the model offers a strong basis for feature extraction without requiring a significant amount of extra training. Top layers were eliminated for task-specific personalization. Here the global average pooling layer preserves important information while reducing the spatial dimensions of feature maps to a single vector. Because the dataset we used is sizable and intricate, Xception performs admirably. For stable feature training processes at multiple scales at once, Xception integrates linear residual connections, which improves handling of hierarchical features. Figure: 5.6 represents the architecture of the Xception model.

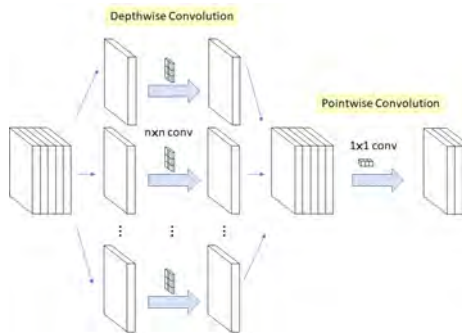


Figure 5.6: Architecture of Xception [13]

ConvMixer: ConvMixer [40] is a deep learning model that is used to extract important features from image or grid-structured data which can be used to perform tasks such as classification detection or segmentation. ConvMixer first detects local

patterns such as edges and textures of the image through small filters. Then, the computational cost is reduced by reducing the size of the image through a pooling layer. The layers gradually extract complex features such as shapes and objects. Finally, the input is classified into specific classes through a fully connected layer. Due to the efficiency of ConvMixer, it is very popular in image processing tasks. Figure: 5.7 represents the architecture of the ConvMixer model .

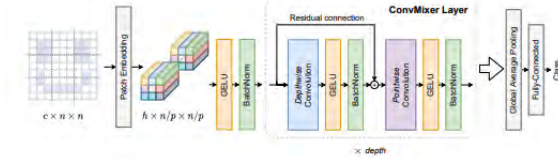


Figure 5.7: Architecture of ConvMixer [50]

ViT: ViT (Vision Transformer) [19] is a type of deep learning model that uses the Transformer architecture for image analysis. Instead of relying on small regions or filters like CNN, ViT divides the entire image into patches and processes each patch as a sequence. Each patch is converted into an embedding and sent to Transformer with positional encoding. Then using the Self-Attention Mechanism, ViT finds the relationship between different parts of the image and determines which part is important. This model can capture local and global relationships together, which is useful for complex image understanding. ViT works well on large datasets and is especially used in cases where it is important to understand the global context within the image. It provides much more flexibility than CNN and is robust for large datasets. Figure: 5.8 represents the architecture of ViT.

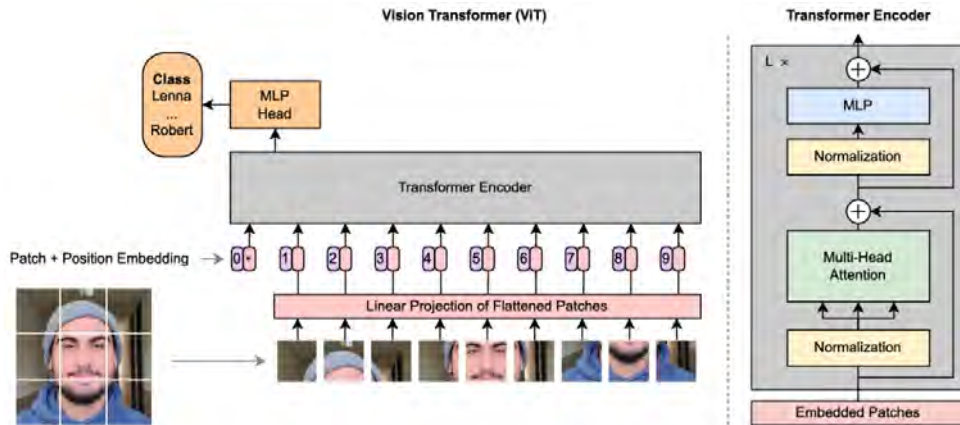


Figure 5.8: Architecture of ViT [70]

DiT: DiT (Document Image Transformer) [57] is a deep learning model specifically designed for document image understanding. It uses the Transformer architecture which is very effective for understanding the relationships and structure of different parts within an image. DiT divides a document image into small patches and converts each patch into an embedding, which represents different parts of the document

such as text, table, image block. These embeddings are added with positional encoding, which indicates the location of each patch within the document. The model then uses the Self-Attention Mechanism to determine the relationships between different parts of the image. It is able to understand the relationships between text blocks, table layouts, and images. DiT can analyze both the structure and content of the document from a global perspective. DiT is commonly used for tasks such as OCR (Optical Character Recognition) document classification, layout analysis, and table detection. It is better than traditional CNN because it is able to understand not only local patterns, but also the global context of the entire document. DiT is particularly useful where text, images, and layout need to be understood together. Its performance is even better on larger document datasets. Figure: 5.9 represents the architecture of DiT.

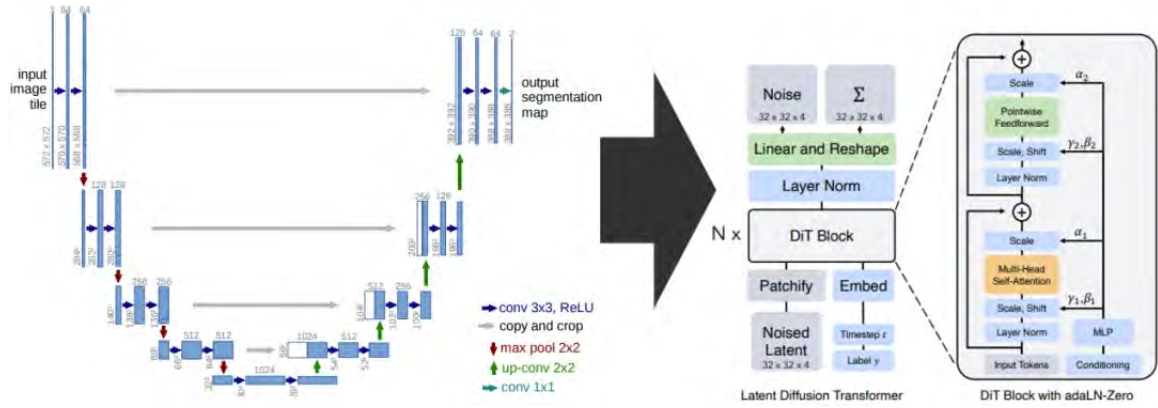


Figure 5.9: Architecture of DiT [59]

EANet: EANet (External Attention Network) [62] is a deep learning model that uses external attention during feature extraction. Traditional attention models use self-attention to determine the relationship between different parts of the input where EANet goes one step further and uses an external memory module. This external memory module stores the global features of the input and retrieves the necessary information during feature extraction. EANet starts by processing the input data, such as an image. First, the initial features of the image are extracted through a convolutional layer. Then these features are sent to the external attention module, where global context-aware information is retrieved from the external memory. With this retrieved information, the feature map is further refined and finally sent to a classifier or other module to produce the output. This model is computationally efficient compared to traditional self-attention, because it uses external memory to find important information instead of pairwise comparison of the entire input. EANet is very effective in understanding global and local features, especially for tasks like image classification, object detection, and segmentation. It works well on large input data and produces fast and accurate results using fewer computational resources, making it popular for complex image analysis and high-resolution image understanding. Figure: 5.10 represents the architecture of EANet model.

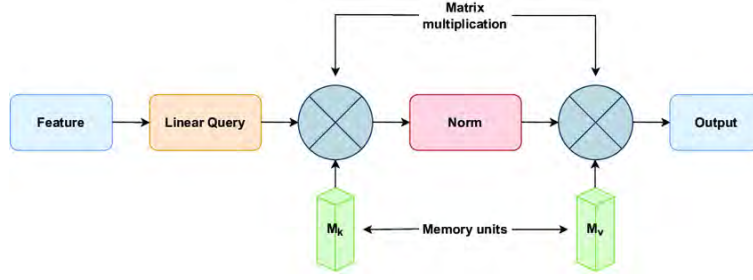


Figure 5.10: Architecture of EANet [53]

CCT: CCT (Compact Convolutional Transformer) [36] is a deep learning model, which is a combination of CNN and Transformer. It is specially designed in such a way that the local feature extraction capabilities of the convolutional layer and the global attention capturing capabilities of the transformer can be used together. CCT provides high performance in image understanding, because it has the ability to understand both local and global relationships of the image. The work of CCT starts by processing the image through the convolutional layer. Here the convolutional layer extracts local patterns such as edges, textures of the image and creates a feature map. This feature map is then divided into small tokens and sent to the Transformer encoder. The Transformer encoder uses the Self-Attention Mechanism to find relationships between different parts of the image. In this way, CCT can understand the global context of the image and focus on important regions. Figure: 5.11 represents the architecture of CCT.

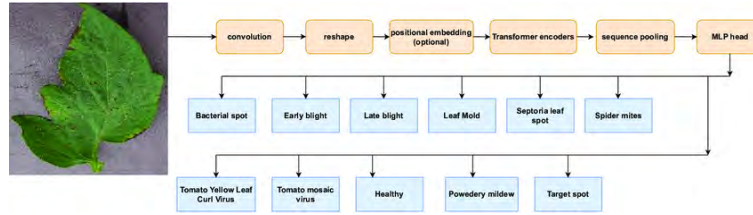


Figure 5.11: Architecture of CCT [53]

VitXDenseNet: VitXDenseNet is a hybrid model that combines the strengths of DenseNet (Convolutional Neural Network) [4] and ViT (Vision Transformer)[change this-18] for image understanding. This model works in a way that combines the efficient feature extraction capabilities of DenseNet with the global context understanding capabilities of ViT. As a result, the model can analyze both local and global features of the image. First, DenseNet works on the input image. It extracts basic local features such as edges, textures from the pixel-level information of the image and creates a feature map. The dense connectivity structure of DenseNet results in feature reuse, making the model computationally efficient. The extracted features are then reshaped into a sequence format, compatible with ViT. Next, this feature map is sent to ViT, where Multi-Head Self-Attention is applied to capture long-range dependencies in the data, followed by a feedforward network to further refine the features. Finally, the output is passed through a dense layer with a sigmoid activation function for binary classification. With the help of ViT Transformer, the Self-Attention Mechanism is used to extract relationships between these patches and

help understand the global context. This allows for the analysis of complex relationships between layouts, blocks, tables, or other visual components. DenseNet retains low-level details, while ViT ensures global understanding. Thus, the VitXDenseNet model is very effective for tasks such as image classification, layout analysis, and OCR. It creates the best combination of local and global feature extraction. Figure: 5.12 represents the architecture of VitXDensenet.

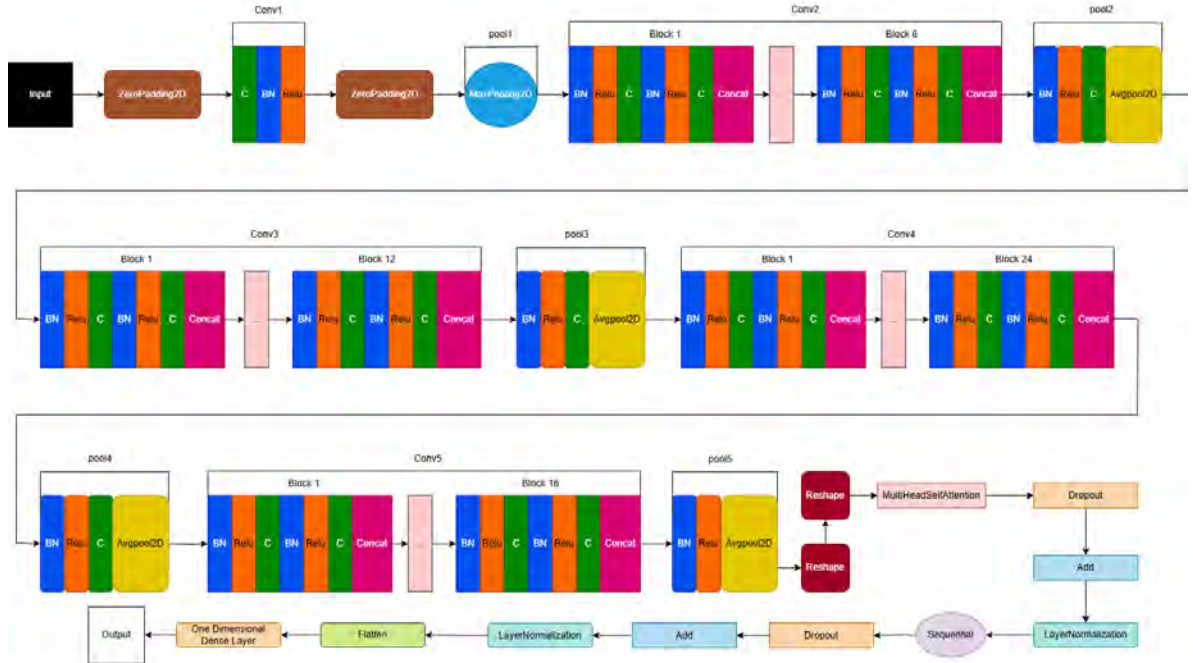


Figure 5.12: Architecture of VitXDensenet

Attention-based MIL: Attention-based MIL (Multiple Instance Learning) [10] is a deep learning method that is specifically designed to work with weakly labeled data. In MIL, a dataset does not assign specific labels to individual instances or data points, but rather the entire data is labeled as a "bag" or collection. Attention-based MIL uses the Attention Mechanism to find the most important instances within this collection. This model first extracts features from the data instances in the entire "bag", which is usually done using a CNN or other feature extractor. The Attention Mechanism then assigns a weight to each instance, which indicates how important that instance is. These weights are used to perform a weighted aggregation of all instances in the bag, with the important instances having the most impact. As a result, the model predicts the label of the entire bag. This model is very useful for tasks such as document classification, medical imaging, and object detection. The Attention Mechanism increases interpretability, because it can be understood which parts the model has given more importance to. It is more accurate and flexible than traditional MIL, as it can automatically focus on important regions or instances. Figure: 5.13 represents Attention-based MIL method.

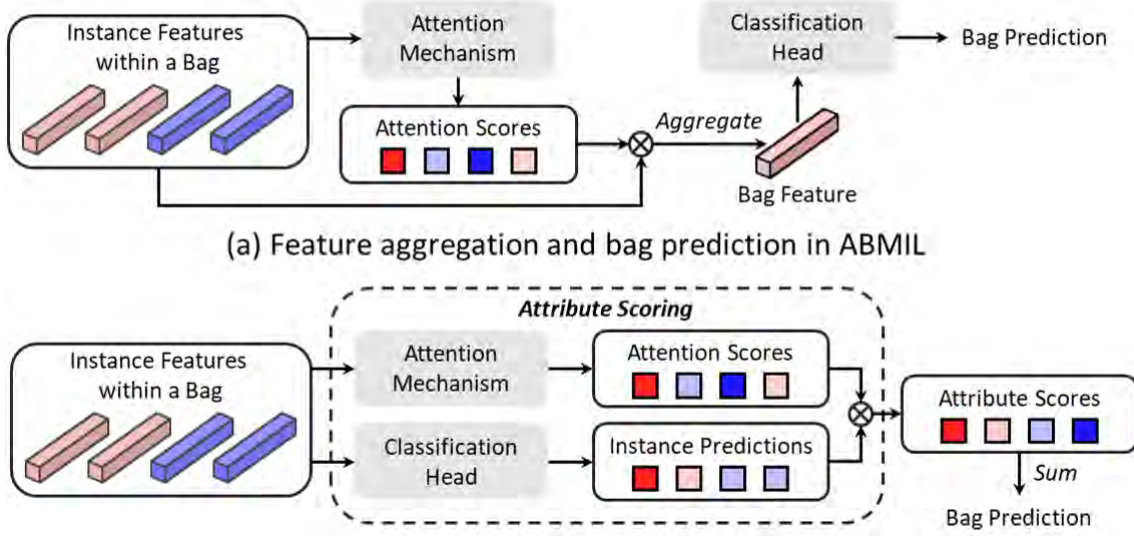


Figure 5.13: Architecture of Attention-based MIL method [64]

5.2 Configuration for Training:

Given table 5.1 shows the training configuration of different models. Some hyper-parameters and training details are mentioned for each model. The table provides information for a total of 13 models. The description of each column in the table is given below:

The 'Model Name' column gives the names of different models. For example: DenseNet121, ResNet50v2, MobileNetV3Large, InceptionResnetV2, EfficientNetV2M, Xception, ConvMixer, ViTxDenseNet, ViT, CCT, DIT, Attention-based MIL, and EANet. The 'Epochs' column specifies the number of training epochs for each model. For example: 16 epochs for the DenseNet121 model, 9 epochs for the ResNet50v2 model, etc. The 'Batch Size' specifies the batch size for each model. The batch size for all models is 100. The 'Image Size' column specifies the image size for each model. The image size for all models is 224x224 pixels. The 'Learning Rate' column specifies the learning rate for each model. The learning rate for all models is 1e-3 (0.001).

Table 5.1: Hyperparameters and other training details.

Model Name	Epochs	Batch size	Image size	Larning Rate
DenseNet121	16	100	224x224	1e-3
ResNet50v2	9	100	224x224	1e-3
MobileNetV3Large	16	100	224x224	1e-3
InceptionResnetV2	11	100	224x224	1e-3
EfficientNetV2M	7	100	224x224	1e-3
Xception	20	100	224x224	1e-3
ConvMixer	10	100	224x224	1e-3
VitXDenseNet	8	100	224x224	1e-3
VIT	16	100	224x224	1e-3
CCT	14	100	224x224	1e-3
DIT	20	100	224x224	1e-3
Attention-based MIL	11	100	224x224	1e-3
EANet	10	100	224x224	1e-3

This table helps to compare the training parameters of different models and displays the configuration details for each model.

Chapter 6

Description of the Data

6.1 Features:

A carefully selected collection of 140000 photos was created from the data set used to help create and assess models that can discriminate between real and artificial faces. The data set is divided into two main subsets. The first subset consists of 70000 REAL faces that were originally gathered by Nvidia from the Flickr dataset. These pictures depict genuine human faces taken in natural settings. The 70000 FAKE faces in the second subset are taken from a total of one million FAKE faces created with StyleGAN, a sophisticated generative adversarial network (GAN) renowned for its capacity to create incredibly lifelike artificial images. By merging these two subsets, the dataset offers a diverse and well-balanced collection of photos, facilitating the creation of models that can successfully distinguish between faces produced by GANs and those that are real.

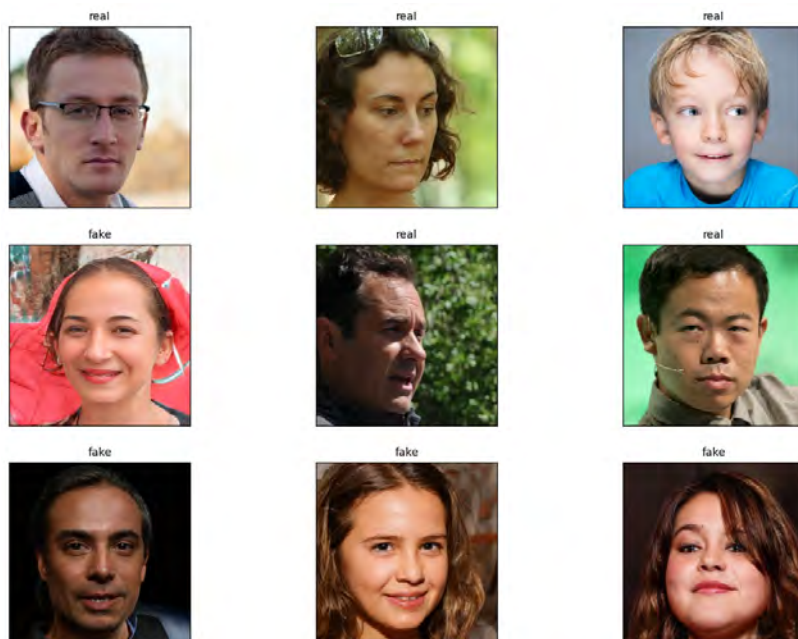


Figure 6.1: Real and fake faces

To facilitate training and assessment, the data set is divided into three different subsets. The 100000 photos in the training set are equally divided between 50000 real and 50000 fake faces. The model can learn the characteristics that differentiate real images from artificial ones by using this subset for training. During the training phase, the validation set—which comprises 20000 photos with 10,000 real and 10,000 fake faces—is used to fine-tune the model and make sure it performs well when applied to new data. Lastly, there are 20000 photos in the test set, 10000 of which are of real faces and 10000 of which are of fake faces. This subset is reserved for the performance evaluation of the model, offering an objective assessment of its accuracy in classifying synthetic and real faces. The data sets balance and separate the training validation and test sets to guarantee solid and reliable model development and assessment. This data set is a useful tool for furthering image forensic research, especially in the area of GAN-generated synthetic image detection.

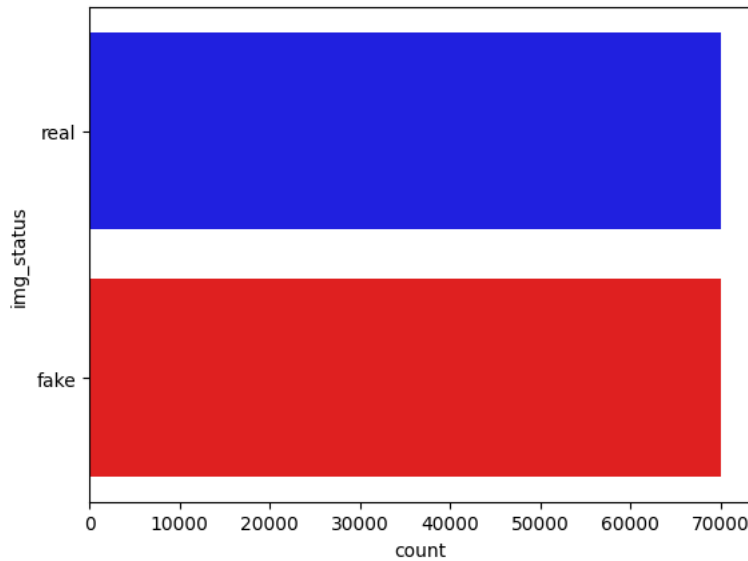


Figure 6.2: Total no of real and fake faces

6.2 Data Preprocessing

In this study, we utilized the `ImageDataGenerator` class from TensorFlow/Keras for preprocessing, ensuring efficient data handling and model training. The dataset was structured into three subsets: training, validation, and test sets. All images were kept the original **224×224 pixels** size, which already matches the input requirements of the deep learning models. The training and validation batches were set to 100 images per batch, while the test set was processed one image at a time to allow for precise evaluation. Given that our deepfake detection task involves binary classification, the labels were assigned in **binary mode**. This preprocessing pipeline focused on standardized image preparation without additional transformations to ensure model performance reflects real-world scenarios. The test set was not shuffled to ensure consistent evaluation and maintain the correct order of images for accurate result comparisons.

Chapter 7

Results and Analysis:

7.1 Evaluation Metrics:

For evaluating all the models, we're using accuracy, precision, recall, F1 score, ROC AUC, and mean accuracy score. Precision (Equation (7.1)) measures the proportion of true positive cases out of all predicted positives. This determines whether the model has higher or lower false positives. Recall (Equation (7.2)) measures the ability of the model to detect all true positive instances. This determines whether the model has higher or lower false negatives. The F1 score (Equation (7.3)) is the harmonic average of precision and recall, balancing both metrics. The Receiver Operating Characteristic - Area Under the Curve (ROC AUC) (Equation (7.4)) evaluates a model's ability to differentiate among positive and negative classes. It combines the True Positive Rate (TPR) and the False Positive Rate (FPR) to measure classification performance. Average Precision Score (Equation (7.7)) calculates the area under the Precision-Recall (PR) curve. It considers precision at different recall levels to evaluate performance, especially for unbalanced datasets. These evaluation metrics ensure a comprehensive analysis of model performance, helping to optimize predictive accuracy and reliability.

7.1.1 Precision

$$\text{Precision} = \frac{\sum TP}{\sum (TP + FP) + \epsilon} \quad (7.1)$$

Where:

- **TP** = True Positives
- **FP** = False Positives
- ϵ is a small value added for numerical stability.

7.1.2 Recall

$$\text{Recall} = \frac{\sum TP}{\sum (TP + FN) + \epsilon} \quad (7.2)$$

Where:

- **FN** = False Negatives

7.1.3 F1 Score

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall} + \epsilon} \quad (7.3)$$

7.1.4 ROC AUC Score

The **Receiver Operating Characteristic - Area Under Curve (ROC-AUC)** score is computed as:

$$\text{ROC AUC} = \int_{-\infty}^{\infty} \text{TPR}(t) \cdot d\text{FPR}(t) \quad (7.4)$$

Where:

- **TPR (True Positive Rate)** = Sensitivity (Recall)

$$\text{TPR} = \frac{TP}{TP + FN} \quad (7.5)$$

- **FPR (False Positive Rate)**

$$\text{FPR} = \frac{FP}{FP + TN} \quad (7.6)$$

7.1.5 Average Precision Score

$$AP = \sum_n (R_n - R_{n-1}) \cdot P_n \quad (7.7)$$

Where:

- P_n is the precision at threshold n .
- R_n is the recall at threshold n .

7.1.6 Accuracy

For binary classification:

$$\text{Accuracy} = \frac{\sum 1(\hat{y} = y)}{N} \quad (7.8)$$

Where:

- $\hat{y}$ is the predicted label.
- y is the ground truth.
- N is the total number of samples.

7.2 Convolution model Result and Analysis:

The convolution models exhibited strong performance across training and validation datasets. Accuracy ranged from 89.76-99.63%, with F1 scores consistently above 0.8, indicating strong classification capabilities. Precision and recall metrics showed balanced performance, with precision above 0.79 and recall above 0.97, suggesting reliable deepfake detection. The sensitivity at specificity for all convolution-based models tested are above 98%. The ROC-AUC scores of 0.9703-0.9999 highlighted the models excellent discriminative power, while Average Precision (AP) scores above 0.96 confirmed their ability to rank potential deepfake instances effectively. For fake images, the precision were around 0.90 and recall around 0.92, leading to an F1-score of around 0.91, indicating a strong balance between correctly identifying positives and minimizing false positives. For real images, precision is around 0.92 and recall is above 0.89, with an F1-score also above 0.91, reflecting similar performance. The macro and weighted averages for precision, recall, and F1-score were all 0.91, indicating consistent performance across both cases. Prediction graphs of several convolution-based models below revealed clear decision boundaries, with very minimal misclassification zones, further validating the convolution models' effectiveness in distinguishing authentic from manipulated media figure: 7.1 and figure: 7.4.

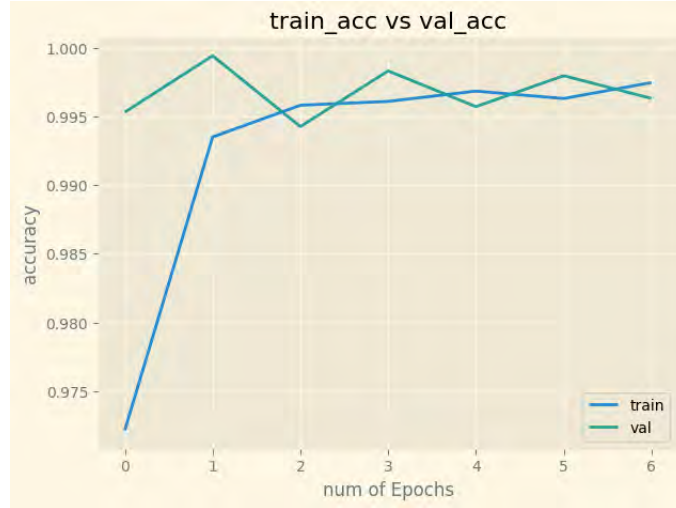


Figure 7.1: Accuracy graph of EfficientNetV2M

7.3 Attention model Result and Analysis:

Attention-based models demonstrated notably lower performance compared to convolutional models. Accuracy fluctuated between 72-78%, with F1 scores around 0.70, indicating less reliable classification. Precision and recall metrics showed inconsistent results, typically ranging from 0.6-0.75. ROC-AUC scores of 0.80-0.87 and Average Precision scores around 0.66-0.86 suggested limited discriminative capabilities in deepfake detection. For fake images, the precision were around 0.64 and recall around 0.63, leading to an F1-score of around 0.66, indicating a weaker balance between correctly identifying positives and minimizing false positives. For real images, precision is around 0.64 and recall is around 0.66, with an F1-score also around 0.62, reflecting similar performance. The macro and weighted averages

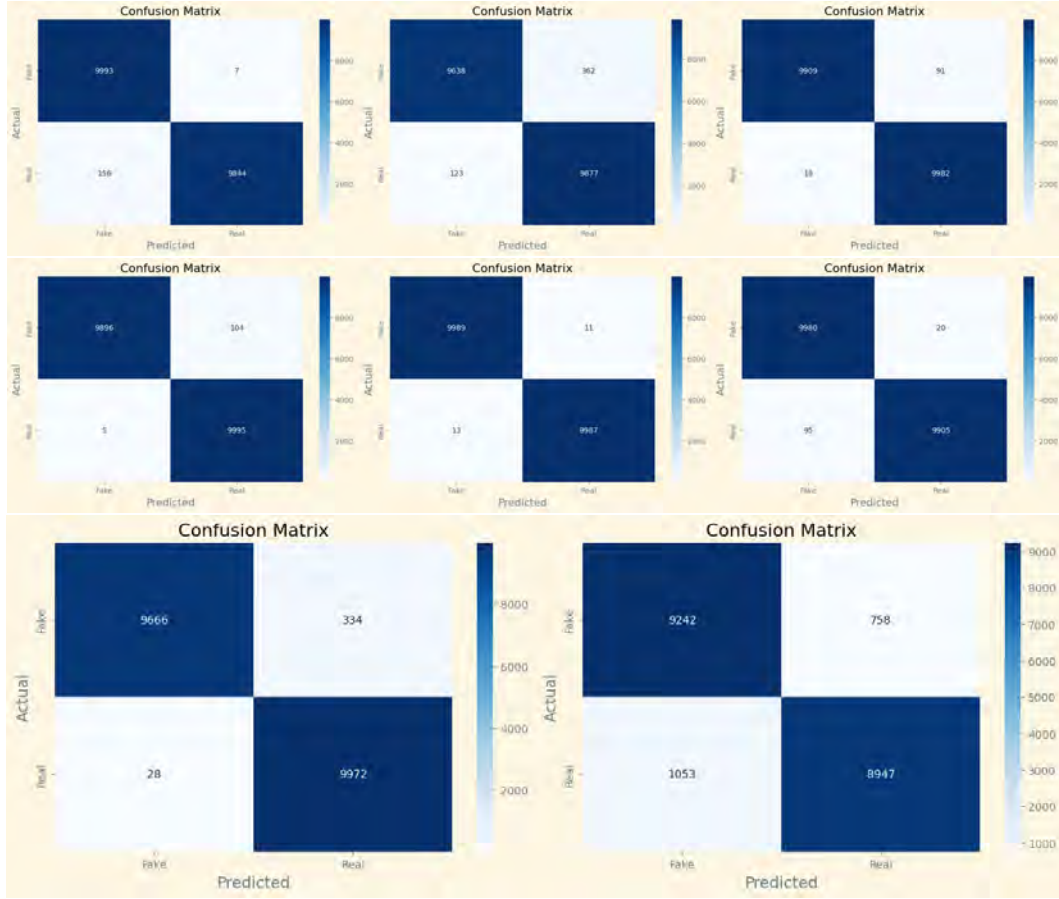


Figure 7.2: Prediction graph of all Convolution-based Models

for precision, recall, and F1-score were all 0.65, indicating consistent performance across both cases. Prediction graphs revealed more ambiguous decision boundaries and increased misclassification zones, highlighting the challenges of attention-based architectures in effectively distinguishing authentic from manipulated media figure: 7.3.

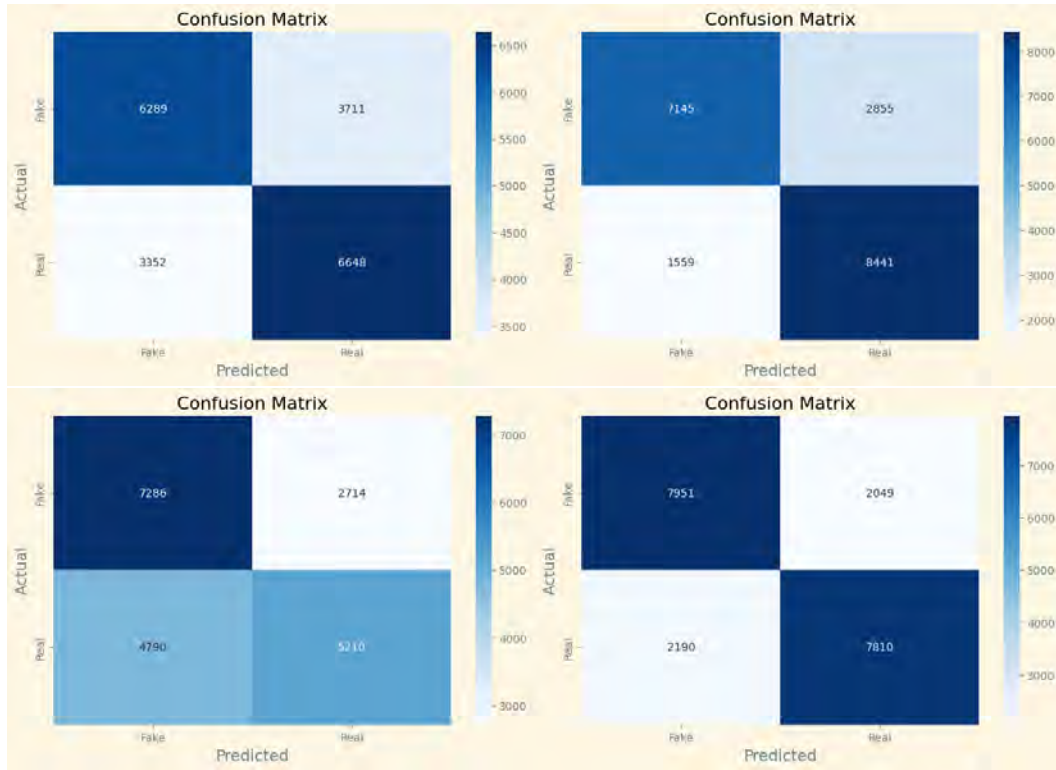


Figure 7.3: Prediction graph of all Attention-based Models

Convolution improves the performance of models because convolution can capture local features of the data very effectively. Convolutional layers develop feature maps and extract intricate local patterns in transformer models such as ConvMixer, ViTX-Densenet, CCT. Here, ViTDenseNet has reduced noise and simplified the raw input data. It is transformed into tokens that attention-based models can use to process the data further. It enables models to effectively capture long-range dependencies or relationships and analyze the data's global context. For instance, DenseNet uses convolution to extract fine-grained local features from input data, while ViT's attention mechanism concentrates on comprehending the image's overall structure and global patterns. Convolutional layers generate tokens in models such as ConvMixer and CCT which are then processed by the Transformer architecture to examine both local and global features. Convolution and attention mechanisms work together to enhance the model's capacity to handle challenging tasks, which is used by ViTDensenet. This approach guarantees that attention mechanisms examine global features to make informed decisions while convolution swiftly and precisely detects local details. This design improves model performance in challenging tasks like image classification, document layout analysis, and medical imaging by bridging the gap between local and global understanding. For better accuracy and efficiency, convolution ultimately makes it easier to extract and interpret both localized and overall relationships by allowing attention-based models to concentrate on meaningful data representations.



Figure 7.4: Prediction graph of custom hybrid model ViTXDenseNet

Table 7.1: An overview of the architecture of the DenseNet121, ResNet50v2, MobileNetV3Large, InceptionResnetV2, EfficientNetV2M, Xception, ConvMixer, VitXDenseNet, VIT, CCT, DIT, Attention-based MIL, EANet models

Model name	Accuracy	Recall	Precision	F1	ROC-AUC	AP Score
DenseNet121	0.99	0.99	0.99	0.99	0.999891859	0.9998907819
ResNet50v2	0.98	0.98	0.98	0.98	0.99742805	0.997303133
MobileNetV3Large	0.99	0.99	0.99	0.99	0.99992028	0.999920566
InceptionResnetV2	0.99	0.99	0.99	0.99	0.99989726	0.99989589
EfficientNetV2M	0.9990	0.9912	0.9937	0.9	0.99999188	0.9999919
Xception	0.99	0.99	0.99	0.99	0.999955745	0.999955696
ConvMixer	0.98	0.98	0.98	0.98	0.99852133	0.99803980546
VitXDenseNet	0.9985	0.9895	0.9965	0.99	0.99995588	0.99995473
VIT	0.78	0.78	0.78	0.78	0.853961655	0.82864927979
CCT	0.91	0.91	0.91	0.91	0.9703575	0.96928858206
DIT	0.65	0.65	0.65	0.65	0.70730992	0.7077668126
Attention-based MIL	0.79	0.79	0.79	0.79	0.8673506849	0.86179107086
EANet	0.62	0.62	0.62	0.62	0.67283586	0.66777460325

Chapter 8

Discussion

Deepfakes are artificial intelligence-generated images or videos that purport to show someone saying or doing something they haven't. Though it is not flawless, the AI leaves behind minor errors known as subtle artifacts when it blends the faces of two people. Small distortions like uneven skin tone, strange shadows, or misaligned mouth movements are examples of these [68]. Skin pores, wrinkles, and light reflections are examples of fine-grained details that are challenging for AI to replicate precisely. This may cause facial features such as the eyes or hair to appear overly smooth or artificial. In deepfakes, local patterns like skin texture or shadow placement might also appear erratic because the AI combines two images [75]. The mistakes help humans and AI to detect deep fakes, despite their difficulty in identifying deep fakes at first glance [25].

Convolution is excellent at identifying minute local characteristics in pictures. In order to identify patterns like edges, corners, or textures, it uses tiny filters called kernels to scan specific areas of the image [11]. Every filter applies mathematical operations to discrete areas of the picture, producing a single value that indicates the pattern's location. For instance, regions with abrupt changes in color or brightness are highlighted by an edge detection filter. The filter is more accurate than examining the entire image at once because its receptive field or the portion of the image it sees at once is small, allowing it to concentrate on local details [5]. Additionally, filters employ weight sharing, which applies the same filter to every part of the image. As a result, computation is more efficient, and patterns are always detectable. Convolution preserves only significant local features such as edges or textures while eliminating extraneous information and noise [9]. Because of this, it is effective at image classification tasks like deepfake detection.

For deepfake detection, convolutional models outperform attention-based models since they are better at identifying patterns and local context in the data. Strong attention to fine-grained localized details is necessary for deepfake classification in order to detect subtle image manipulations, a task that convolutional models are naturally equipped to perform. Edge textures and pixel relationships are examples of local patterns that our convolutional models detect and extract using convolutional layers. The detection of minor yet significant alterations made to phony images depends on these characteristics. Our convolutional layers simplify the image raw data, lower noise, and highlight significant local structures by producing comprehensive feature maps. This enables the models to reliably and precisely differentiate

real images from ones that have been altered.

The superior benefits of hybrid models, such as Compact Convolutional Transformers (CCT) which incorporate convolutional layers to efficiently capture local context and patterns before feeding the extracted feature maps into attention mechanisms for global understanding, are demonstrated by our work. Together they produce a well-balanced architecture that makes use of the advantages of both attention-based and convolutional techniques. By employing convolutional layers, CCT streamlines unprocessed input data, minimizes noise, and produces superior tokens that attention mechanisms can use to examine global connections. CCT models show improved optimization stability, faster convergence, and robust generalization across datasets and complexities, in addition to improving classification accuracy and reliability [32]. Also we have another hybrid model, which is ConvMixer, and ViTXDensenet. ConvMixer and ViTXDensenet both emphasize how crucial it is to combine convolutional operations with sophisticated architectures in order to enhance model performance. With its purely convolutional architecture ConvMixer is excellent at capturing local spatial details whereas ViTXDensenet uses the complementary effects of convolution and attention to analyze both local and global features. These models show how convolutional layers can improve attention-based models processing of meaningful data representations, improving accuracy and efficiency in difficult tasks. The advancement of contemporary deep learning models in computer vision and other fields depends on this fusion of local and global knowledge. DenseNet is mainly strong on image-related tasks, while DiT performs well on text-related tasks. If these two models are combined, better results can be obtained on multi-modal tasks where both images and text are present. Due to the use of convolution in DenseNet, it is more effective in capturing spatial or local features, which gives good results in image-related tasks. Therefore, we got better results in convolution for local features. However, we get better results from attention models using Desnet + DiT but get worse results from convolution models and in the context of attention mechanisms, the best results are often achieved with CCT.

Chapter 9

Conclusion

In this day and age, Deepfake Detection Technologies are much needed to combat the foul use of deepfake technology. In addition to preventing widespread misunderstandings and false information, this can be used to combat deepfakes that are intended to damage a person's or an organization's reputation or to further any other malevolent intent that might endanger a country's safety and security. The purpose of this thesis is to make a comparative analysis of deepfake detection using convolutions and attention mechanism, with a view to determining which, and possibly a combination of, techniques is most effective at identifying deepfake content. In this work, by comparing them in terms of accuracy, generalizability, computational expense, and robustness, an attempt will be made to shed light on the respective strengths and weaknesses of each technique. In addition, potential hybrid approaches combining convolutions' and attention mechanism's complementary strengths will be examined. What is hoped to be derived through such an investigation is a contribution towards guiding future work in developing and improving current deepfake detection approaches, and developing even more reliable and efficient tools for counteracting deepfake proliferation in media. Robust detection methods are required due to the significant harm these images offer to the privacy, security, and authenticity of the information. Using a dataset of 100,000 training images and 20,000 validation or testing images, the 13 tested models have given promising results for both training and validation images. And of the 13 models, EfficientNetV2M has given the best result so far in all testing criteria, followed by our custom model ViTXDenseNet. So, from our experiment, we can safely say that EfficientNetV2M is the best model when it comes to deepfake detection.

Bibliography

- [1] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” *arXiv preprint arXiv:1603.07285*, 2016.
- [2] K. He, X. Zhang, S. Ren, and J. Sun, “Identity mappings in deep residual networks,” in *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, Springer, 2016, pp. 630–645.
- [3] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.
- [4] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [5] V. Pappas, J. Sulam, and M. Elad, “Working locally thinking globally: Theoretical guarantees for convolutional sparse coding,” *IEEE Transactions on Signal Processing*, vol. 65, no. 21, pp. 5687–5701, 2017.
- [6] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.
- [7] A. Vaswani, “Attention is all you need,” *Advances in Neural Information Processing Systems*, 2017.
- [8] S. Cole, *We are truly fucked: Everyone is making AI-generated fake porn now*, en, <https://www.vice.com/en/article/bjye8a/reddit-fake-porn-app-daisy-ridley>, Accessed: 2024-6-5, Jan. 2018.
- [9] I. Hadji and R. P. Wildes, “What do we understand about convolutional networks?” *arXiv preprint arXiv:1803.08834*, 2018.
- [10] M. Ilse, J. Tomczak, and M. Welling, “Attention-based deep multiple instance learning,” in *International conference on machine learning*, PMLR, 2018, pp. 2127–2136.
- [11] M. Jogin, M. Madhulika, G. Divya, R. Meghana, S. Apoorva, *et al.*, “Feature extraction using convolution neural networks (cnn) and deep learning,” in *2018 3rd IEEE international conference on recent trends in electronics, information & communication technology (RTEICT)*, IEEE, 2018, pp. 2319–2323.
- [12] Y. Li, M.-C. Chang, and S. Lyu, “In ictu oculi: Exposing ai created fake videos by detecting eye blinking,” in *2018 IEEE International workshop on information forensics and security (WIFS)*, IEEE, 2018, pp. 1–7.

- [13] S.-H. Tsang, “Review: Xception—with depthwise separable convolution, better than inception-v3 (image classification),” *Towards Data Science*, vol. 25, 2018.
- [14] A. Howard, M. Sandler, G. Chu, *et al.*, “Searching for mobilenetv3,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1314–1324.
- [15] J. Kim, S. Han, and S. S. Woo, “Classifying genuine face images from disguised face images,” in *2019 IEEE International Conference on Big Data (Big Data)*, IEEE, 2019, pp. 6248–6250.
- [16] R. Koshy and A. Mahmood, “Optimizing deep cnn architectures for face liveness detection,” *Entropy*, vol. 21, no. 4, p. 423, 2019.
- [17] Z. Qu, S.-Y. Wang, L. Liu, and D.-Y. Zhou, “Visual cross-image fusion using deep neural networks for image edge detection,” *IEEE Access*, vol. 7, pp. 57 604–57 615, 2019.
- [18] K. Zhang, Y. Guo, X. Wang, J. Yuan, and Q. Ding, “Multiple feature reweight densenet for image classification,” *IEEE access*, vol. 7, pp. 9872–9880, 2019.
- [19] D. Alexey, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv: 2010.11929*, 2020.
- [20] L. Guarnera, O. Giudice, and S. Battiato, “Fighting deepfake by exposing the convolutional traces on images,” *IEEE Access*, vol. 8, pp. 165 085–165 098, 2020.
- [21] C.-C. Hsu, Y.-X. Zhuang, and C.-Y. Lee, “Deep fake image detection based on pairwise learning,” *Applied Sciences*, vol. 10, no. 1, p. 370, 2020.
- [22] T. Jung, S. Kim, and K. Kim, “Deepvision: Deepfakes detection using human eye blinking pattern,” *IEEE Access*, vol. 8, pp. 83 144–83 154, 2020.
- [23] Z. Qu, J. Mei, L. Liu, and D.-Y. Zhou, “Crack detection of concrete pavement with cross-entropy loss function and improved vgg16 network model,” *Ieee Access*, vol. 8, pp. 54 564–54 573, 2020.
- [24] S. Song, J. C. Lam, Y. Han, and V. O. Li, “Resnet-lstm for real-time pm_{2.5} and pm₁₀ estimation using sequential smartphone images,” *IEEE Access*, vol. 8, pp. 220 069–220 082, 2020.
- [25] S. Chen, T. Yao, Y. Chen, S. Ding, J. Li, and R. Ji, “Local relation learning for face forgery detection,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, 2021, pp. 1081–1088.
- [26] T. Gwyn, K. Roy, and M. Atay, *Face recognition using popular deep net architectures: A brief comparative study. futur internet 13 (7): 1–15*, 2021.
- [27] T. Liu, T. Chen, R. Niu, and A. Plaza, “Landslide detection mapping employing cnn, resnet, and densenet in the three gorges reservoir, china,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 11 417–11 428, 2021.
- [28] P. Neekhara, B. Dolhansky, J. Bitton, and C. C. Ferrer, “Adversarial threats to deepfake detection: A practical perspective,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 923–932.

- [29] Y. Nirkin, L. Wolf, Y. Keller, and T. Hassner, “Deepfake detection based on discrepancies between faces and their context,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6111–6121, 2021.
- [30] A. A. Pokroy and A. D. Egorov, “Efficientnets for deepfake detection: Comparison of pretrained models,” in *2021 IEEE conference of russian young researchers in electrical and electronic engineering (ElConRus)*, IEEE, 2021, pp. 598–600.
- [31] H. S. Shad, M. M. Rizvee, N. T. Roza, *et al.*, “Comparative analysis of deepfake image detection method using convolutional neural network,” *Computational intelligence and neuroscience*, vol. 2021, 2021.
- [32] T. Xiao, M. Singh, E. Mintun, T. Darrell, P. Dollár, and R. Girshick, “Early convolutions help transformers see better,” *Advances in neural information processing systems*, vol. 34, pp. 30 392–30 400, 2021.
- [33] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, and N. Yu, “Multi-attentional deepfake detection,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2185–2194.
- [34] A. Arini, R. B. Bahaweres, and J. Al Haq, “Quick classification of xception and resnet-50 models on deepfake video using local binary pattern,” in *2021 International Seminar on Machine Learning, Optimization, and Data Science (ISMODE)*, IEEE, 2022, pp. 254–259.
- [35] S. Ganguly, A. Ganguly, S. Mohiuddin, S. Malakar, and R. Sarkar, “Vixnet: Vision transformer with xception network for deepfakes based video and image forgery detection,” *Expert Systems with Applications*, vol. 210, p. 118 423, 2022.
- [36] A. I. Jajja, A. Abbas, H. A. Khattak, *et al.*, “Compact convolutional transformer (cct)-based approach for whitefly attack detection in cotton crops,” *Agriculture*, vol. 12, no. 10, p. 1529, 2022.
- [37] J. John and B. V. Sherif, “Comparative analysis on different deepfake detection methods and semi supervised gan architecture for deepfake detection,” in *2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, IEEE, 2022, pp. 516–521.
- [38] P. V. Lal, U. Srilakshmi, and D. Venkateswarlu, “Face recognition using deep learning xception cnn method,” *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 2, p. 2023, 2022.
- [39] X. Li, R. Ni, P. Yang, Z. Fu, and Y. Zhao, “Artifacts-disentangled adversarial learning for deepfake detection,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 4, pp. 1658–1670, 2022.
- [40] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A convnet for the 2020s,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 11 976–11 986.
- [41] K. Narayan, H. Agarwal, S. Mittal, *et al.*, “Desi: Deepfake source identifier for social media,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2858–2867.

- [42] E. U. H. Qazi, T. Zia, and A. Almorjan, "Deep learning-based digital image forgery detection system," *Applied Sciences*, vol. 12, no. 6, p. 2851, 2022.
- [43] A. Raza, K. Munir, and M. Almutairi, "A novel deep learning approach for deepfake image detection," *Applied Sciences*, vol. 12, no. 19, p. 9820, 2022.
- [44] I. Sahib and T. A. A. AlAsady, "Deep learning for image forgery classification based on modified xception net and dense net," in *AIP Conference Proceedings*, AIP Publishing, vol. 2547, 2022.
- [45] V. V. V. N. S. Vamsi, S. S. Shet, S. S. M. Reddy, *et al.*, "Deepfake detection in digital media forensics," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 74–79, 2022.
- [46] S. Verma, "Classification of spoofing attack detection using deep learning algorithms," Ph.D. dissertation, Dublin, National College of Ireland, 2022.
- [47] R. Adityatama and A. T. Putra, "Image classification of human face shapes using convolutional neural network xception architecture with transfer learning," *Recursive Journal of Informatics*, vol. 1, no. 2, pp. 102–109, 2023.
- [48] A. Aghasanli, D. Kangin, and P. Angelov, "Interpretable-through-prototypes deepfake detection for diffusion models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 467–474.
- [49] A. Alsaheel, R. Alhassoun, R. Alrashed, N. Almatrafi, N. Almallouhi, and S. Albahli, "Deep fakes in healthcare: How deep learning can help to detect forgeries," *Computers, Materials & Continua*, vol. 76, no. 2, 2023.
- [50] R. Aso, T. Chuman, and H. Kiya, "A privacy preserving method with a random orthogonal matrix for convmixer models," *arXiv preprint arXiv:2301.03843*, 2023.
- [51] C. Bregler, M. Covell, and M. Slaney, "Video rewrite: Driving visual speech with audio," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 715–722.
- [52] H. Chen, Y. Li, D. Lin, B. Li, and J. Wu, "Watching the big artifacts: Exposing deepfake videos via bi-granularity artifacts," *Pattern Recognition*, vol. 135, p. 109179, 2023.
- [53] S. Hossain, M. Tanzim Reza, A. Chakrabarty, and Y. J. Jung, "Aggregating different scales of attention on feature variants for tomato leaf disease diagnosis from image data: A transformer driven study," *Sensors*, vol. 23, no. 7, p. 3751, 2023.
- [54] M. Kumar, H. K. Sharma, *et al.*, "A gan-based model of deepfake detection in social media," *Procedia Computer Science*, vol. 218, pp. 2153–2162, 2023.
- [55] M. Nawaz, A. Javed, and A. Irtaza, "Resnet-swish-dense54: A deep learning approach for deepfakes detection," *The Visual Computer*, vol. 39, no. 12, pp. 6323–6344, 2023.
- [56] Y. Patel, S. Tanwar, P. Bhattacharya, *et al.*, "An improved dense cnn architecture for deepfake image detection," *IEEE Access*, vol. 11, pp. 22081–22095, 2023.

- [57] W. Peebles and S. Xie, “Scalable diffusion models with transformers,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4195–4205.
- [58] S. Sadiq, T. Aljrees, and S. Ullah, “Deepfake detection on social media: Leveraging deep learning and fasttext embeddings for identifying machine-generated tweets,” *IEEE Access*, 2023.
- [59] T. Saleena, P. Ilyas, V. Sajna, and A. Haque, “Deep learning techniques for quantification of tumour necrosis in post-neoadjuvant chemotherapy osteosarcoma resection specimens for effective treatment planning,” *Acta Informatica Pragensia*, vol. 12, no. 1, pp. 87–103, 2023.
- [60] J. Sawtell-Rickson, *What is a deepfake?* en, <https://builtin.com/machine-learning/deepfake>, Accessed: 2024-6-5, Jan. 2023.
- [61] S. R. Shah, S. Qadri, H. Bibi, S. M. W. Shah, M. I. Sharif, and F. Marinello, “Comparing inception v3, vgg 16, vgg 19, cnn, and resnet 50: A case study on early detection of a rice disease,” *Agronomy*, vol. 13, no. 6, p. 1633, 2023.
- [62] P. Yao, T. Mao, M. Shi, J. Sun, and Z. Wang, “Eanet: Expert attention network for online trajectory prediction,” *arXiv preprint arXiv:2309.05683*, 2023.
- [63] Z. N. Ashani, I. S. C. Ilias, K. Y. Ng, M. R. K. Ariffin, A. D. Jarno, and N. Z. Zamri, “Comparative analysis of deepfake image detection method using vgg16, vgg19 and resnet50,” *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 47, no. 1, pp. 16–28, 2024.
- [64] L. Cai, S. Huang, Y. Zhang, J. Lu, and Y. Zhang, “Rethinking attention-based multiple instance learning for whole-slide pathological image classification: An instance attribute viewpoint,” *arXiv preprint arXiv:2404.00351*, 2024.
- [65] D. Gerdan Koc, C. Koc, H. E. Polat, and A. Koc, “Artificial intelligence-based camel face identification system for sustainable livestock farming,” *Neural Computing and Applications*, vol. 36, no. 6, pp. 3107–3124, 2024.
- [66] A. Heidari, N. Jafari Navimipour, H. Dag, and M. Unal, “Deepfake detection using deep learning methods: A systematic and comprehensive review,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 14, no. 2, e1520, 2024.
- [67] R. Jongjaraunsuk, W. Taparhudee, S. Sirisuay, M. Kaewnern, V. Dulyapurk, and S. Janekitkarn, “Transfer learning model application for rastrelliger brachysoma and r. kanagurta image classification using smartphone-captured images,” *Fishes*, vol. 9, no. 3, p. 103, 2024.
- [68] D. Nguyen, N. Mejri, I. P. Singh, *et al.*, “Laa-net: Localized artifact attention network for quality-agnostic and generalizable deepfake detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 17 395–17 405.
- [69] N. Qazi and I. Ahmed, “Enhancing authenticity verification with transfer learning and ensemble techniques in facial feature-based deepfake detection,” in *2024 14th International Conference on Pattern Recognition Systems (ICPRS)*, IEEE, 2024, pp. 1–6.

- [70] M. Rodrigo, C. Cuevas, and N. García, “Comprehensive comparison between vision transformers and convolutional neural networks for face recognition tasks,” *Scientific reports*, vol. 14, no. 1, p. 21 392, 2024.
- [71] R. A. F. Saskoro, N. Yudistira, and T. N. Fatyanosa, “Detection of ai-generated images from various generators using gated expert convolutional neural network,” *IEEE Access*, 2024.
- [72] S. Shakibhamedan, N. Amirafshar, A. S. Baroughi, H. S. Shahhoseini, and N. Taherinejad, “Ace-cnn: Approximate carry disregard multipliers for energy-efficient cnn-based image classification,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2024.
- [73] K. Tenbarga, *Taylor swift deepfakes on X falsely depict her supporting trump*, en, <https://www.nbcnews.com/tech/internet/taylor-swift-deepfake-x-falsely-depict-supporting-trump-grammys-flag-rcna137620>, Accessed: 2024-6-5, Feb. 2024.
- [74] X. Yang, Y. Li, L. Yang, Y. Zhang, X. Wang, and Q. Zhang, “High-noise solar panel defect identification method based on the improved efficientnet-v2,” *Journal of Renewable and Sustainable Energy*, vol. 16, no. 5, 2024.
- [75] W. Zhou, X. Luo, Z. Zhang, J. He, and X. Wu, “Capture artifacts via progressive disentangling and purifying blended identities for deepfake detection,” *arXiv preprint arXiv:2410.10244*, 2024.
- [76] <https://www.reuters.com/world/europe/deepfake-footage-purports-show-ukrainian-president-capitulating-2022-03-16/>, Accessed: 2024-6-5.
- [77] <https://www.collinsdictionary.com/dictionary/english/deepfake>, Accessed: 2024-6-5.
- [78] M. Tan and Q. Le, “Efficientnetv2: Smaller models and faster training. arxiv 2021,” *arXiv preprint arXiv:2104.00298*,