

T.C.
TED UNIVERSITY
GRADUATE SCHOOL
COMPUTER ENGINEERING

EXPLAINABLE DEEPFAKE DETECTION USING FRAME LEVEL CNN
MODELS: A COMPARATIVE STUDY OF AUGMENTATION AND CUTOUT
TECHNIQUES

MERT KAYA

ANKARA, 2025

EXPLAINABLE DEEPPFAKE DETECTION
USING FRAME LEVEL CNN MODELS:
A COMPARATIVE STUDY OF AUGMENTATION AND CUTOUT
TECHNIQUES

A Thesis Submitted To
The Graduate School
of
TED University

by

Mert Kaya

In Partial Fulfillment of The Requirements
For
Master of Science
in
Computer Engineering

ANKARA, 2025

Approval of the Graduate School

Prof. Dr. Ayça Tekin Kuru
Dean, Graduate School

Prof. Dr. Hakkı Gökhan İlk
Program Chair

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Venera Adanova
Supervisor

Examining Committee Members

Prof Dr. Tolga Kurtuluş Çapın (Committee Chair) _____
TED University, Department of Computer Engineering

Assoc. Prof. Dr. Yücel Çimtay _____
TOBB Economy and Technology University, Department of Computer
Engineering

Asst. Prof. Dr. Venera Adanova _____
Ted University, Department of Computer Engineering

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Mert Kaya

Signature :

ABSTRACT

EXPLAINABLE DEEPFAKE DETECTION USING FRAME LEVEL CNN MODELS: A COMPARATIVE STUDY OF AUGMENTATION AND CUTOUT TECHNIQUES

Mert Kaya

Master of Science, Computer Science

Supervisor: Assist. Prof. Dr. Venera Adanova

September, 2025

The progress in artificial intelligence has given rise to the generation of deepfake videos that have increasingly convincing details. While such technology is of immense value to entertainment and creative industries, it has, on the other hand, raised serious concerns regarding privacy, security, and public trust. With increasing sophistication in deepfake generation methods, interpretation and explainability of the detection methods have been identified as key research priorities, especially for judicial matters that require transparent evidence under the analysis of legal practitioners. This study examines the impact of preprocessing methods, visual augmentation, and targeted masking on the performance and explainability of CNN-based deepfake detection frameworks. For the experimental setup, a frame-based spatial domain deepfake detection approach is employed. Many variations and sets of combinations were systematically analyzed to test their effect on detection. Model interpretability was assessed using Grad-CAM visualization techniques. By observing decision-making patterns, it was possible to identify configurations that focus on the manipulated regions. Therefore, this indicates that certain preprocessing configurations enhance accuracy and explainability, paving the way for generating more credible and trustworthy detection pipelines.

These results highlight the importance of carefully considering preprocessing techniques in the development of detection methods that can remain effective against the ongoing evolution of deepfake technologies. This work contributes to the development of more explainable and reliable deepfake detection methods that can be implemented in the real world, especially where explainable AI is necessary to establish credibility and legal admissibility.

Keywords: Deepfake detection, explainable artificial intelligence, grad-gam, cnn, deepfakes



ÖZET

FRAME DÜZEYİNDE CNN MODELLERİ KULLANILARAK AÇIKLANABİLİR DEEPFAKE TESPİTİ: AUGMENTATION VE KESME TEKNİKLERİNİN KARŞILAŞTIRMALI BİR ÇALIŞMASI

Mert Kaya

Yüksek Lisans, Bilgisayar Mühendisliği

Tez Yöneticisi: Dr. Öğr. Üyesi Venera Adanova

Eylül, 2025

Yapay zekadaki ilerleme, giderek daha ikna edici ayrıntılara sahip deepfake videoların üretimini ortaya çıkarmıştır. Bu teknoloji eğlence ve yaratıcı endüstriler için büyük değer taşıırken, öte yandan mahremiyet, güvenlik ve kamu güveni konularında ciddi endişeler yaratmıştır. Deepfake üretim yöntemlerinin artan karmaşıklığı ile tespit yöntemlerinin yorumlanabilirliği ve açıklanabilirliği, özellikle hukuk uzmanlarının analizi altında şeffaf kanıt gerektiren adli konular için temel araştırma öncelikleri olarak belirlenmiştir. Bu çalışma, ön işleme yöntemlerinin, görsel artırımın ve hedefli maskeleyenin CNN tabanlı deepfake tespit çerçevelerinin performansı ve açıklanabilirliği üzerindeki etkisini incelemektedir. Deneysel düzenek için kare tabanlı uzamsal alan deepfake tespit yaklaşımı kullanılmıştır. Tespit üzerindeki etkilerini test etmek için birçok varyasyon ve kombinasyon seti sistematik olarak analiz edilmiştir. Model yorumlanabilirliği Grad-CAM görselleştirme teknikleri kullanılarak değerlendirilmiştir. Karar verme kalıpları gözlemlenerek, manipüle edilmiş bölgelere odaklanan konfigürasyonların belirlenmesi mümkün olmuştur. Bu nedenle, belirli ön işleme konfigürasyonlarının doğruluk ve açıklanabilirliği artırdığı, daha güvenilir ve etkili tespit hatlarının oluşturulması yolunu açtığı gösterilmektedir.

Bu sonuçlar, deepfake teknolojilerinin devam eden evrimi karşısında etkili kalabilecek tespit yöntemlerinin geliştirilmesinde ön işleme tekniklerinin dikkatli değerlendirilmesinin önemini vurgulamaktadır. Bu çalışma, özellikle açıklanabilir yapay zekanın güvenilirlik ve hukuki kabul edilebilirlik oluşturmak için gerekli olduğu durumlarda gerçek dünyada uygulanabilecek daha açıklanabilir ve güvenilir deepfake tespit yöntemlerinin geliştirilmesine katkıda bulunmaktadır.

Anahtar Kelimeler: Deepfake tespiti, açıklanabilir yapay zeka, grad-cam, cnn, deepfake



ACKNOWLEDGEMENTS

The author wishes to express his deepest gratitude to his supervisor Dr. Venera Adanova for their guidance, advice, criticism, encouragements and insight throughout the research.

I am profoundly grateful to my parents, Ali, Dilek and my Grandma Cemile, who have nurtured my intellectual curiosity since childhood and provided steadfast support for my educational goals and pursuits. Their belief in my potential has been the foundation upon which this work was built. Also I would like to thank my friends Berktug and Layla for their support.

To my partner, Betul, words cannot adequately express my gratitude for your infinite patience, understanding, and unwavering support, particularly during the most demanding phases of this research. You have been my anchor through every storm.

TABLE OF CONTENTS

ABSTRACT	iv
ÖZET	vi
ACKNOWLEDGMENTS	viii
TABLE OF CONTENTS	ix
LIST OF TABLES	xiii
LIST OF ABBREVIATIONS	xiv
LIST OF FIGURES	xv
CHAPTER 1	1
INTRODUCTION	1
1.1 Background	1
1.2 Societal and Ethical Implications of Deepfakes	2
1.3 Research Motivation	5
1.4 Role of Explainability in Deepfake Detection	7
1.5 Structure of the Thesis	8
CHAPTER 2	9
THEORETICAL FOUNDATIONS	9
2.1 Approaches in Machine Learning	9
2.1.1 Supervised Learning	10
2.1.2 Unsupervised Learning	11
2.2 Fundamental Concepts in Machine Learning	11
2.2.1 Generalization and Overfitting	11
2.2.2 Regularization	12
2.2.3 Transfer Learning and Fine Tuning	13
2.2.4 Robustness and Noise	14
2.2.5 Explainable AI (XAI)	14
2.3 Neural Networks	15
2.3.1 Fully Connected Networks	15

2.3.2 Backpropagation	18
2.4 Convolutional Neural Networks	19
2.4.1 Convolution	20
2.4.2 Pooling	21
2.4.3 Batch Normalization	21
2.4.4 Evolution of CNN Architectures	22
CHAPTER 3	24
DEEPPFAKE CREATION	24
3.1 Deepfake Creation Models	24
3.1.1 Encoder-Decoder Models	25
3.1.2 Reenacment Based Deepfake Methods	26
3.1.3 Second Generation Face Swapping Techniques.....	27
3.1.4 Generative Artificial Networks in Deepfake Generation.....	28
3.2 Types of Deepfakes.....	28
3.2.1 Entire Face Synthesis	29
3.2.2 Expression swap	31
3.2.3 Identity Swap	32
3.2.4 Attribute manipulation	32
CHAPTER 4	33
RELATED WORK	33
4.1 Feature-Based Deepfake Detection Methods	33
4.1.1 Spatial Feature-Based Detection Approaches.....	34
4.1.2 Temporal Feature-Based Detection Approaches.....	37
4.2 Learning Based Detection Approaches	38
4.2.1 Classification Approaches.....	38
4.2.2 Anomaly Detection	41
4.3 Deepfake Detection Challenge (DFDC)	42
4.4 Datasets	43

4.4.1 DFDC Dataset.....	43
4.4.2 FaceForensics++	44
4.4.3 CelebDF	44
CHAPTER 5	45
METHODOLOGY	45
5.1 Preprocessing and Training Pipeline	46
5.1.1 Preprocessing.....	47
5.1.2 Augmentations and Cutouts.....	47
5.1.3 Classification	49
5.2 Datasets	50
5.3 Evaluation Measures	50
5.3.1 Metrics	51
5.3.2 Diagnostic Curves and Explainability Analysis	53
CHAPTER 6	55
EXPERIMENTS	55
6.1 Baseline Model	55
6.2 Experimental Settings.....	56
6.3 Experimental Models	57
6.3.1 Only Augmentation Model.....	57
6.3.2 Only More Augmentation Model	58
6.3.3 Random Filled Cutouts with Augmentation.....	58
6.3.4 Black Filled Cutouts with Augmentation.....	58
6.3.5 White Filled Cutouts with Augmentation	59
6.3.6 Cutout Only Models	59
CHAPTER 7	60
RESULTS.....	60
7.1 Quantitative Performance Analysis	60
7.1.1 Area Under the Curve Results	61
7.1.2 F1Score Results	62

7.1.3 BrierLoss Results	64
7.1.4 LogLoss Results	65
7.2 Qualitative Interpretability Analysis.....	66
7.2.1 True Negative Case Analysis.....	70
7.2.2 True Positive Case Analysis	72
7.2.3 False Negative Case Analysis	74
7.2.4 False Positive Case Analysis.....	75
7.2.5 Stardart Deviation of Activation Analysis	76
CHAPTER 8	78
CONCLUSION AND DISCUSSION	78
8.1 Future Work.....	78
8.2 Conclusion.....	79
REFERENCES.....	80
APPENDICES	
A. Star Cutout Pseudo Code	87
B. Augmentations with lower probability	88
C. Augmentations with higher probability.....	89
D. Average Regional Activation for baseline model.....	90

LIST OF TABLES

TABLES

Table 1. Table containing the Test AUC results of different experimental models..... 61

Table 2. Table containing the Test F1Score results of different experimental models..... 62

Table 3. Table showcasing the brier loss of different experimental models.64

Table 4. Table showcasing the logloss of different experimental models. .. 65



LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
XAI	Explainable Artificial Intelligence
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
Grad-Cam	Gradient-weighted Class Activation Mapping
FCNN	Fully Connected Neural Network
BN	Batch Normalization
ICS	Internal Covariate Shift
FF++	FaceForensics++ Dataset
CelebDF	Celebrity Deepfakes Dataset
DFDC	Deepfake Detection Challenge

LIST OF FIGURES

Figure 1. Example of a deepfaked image using FaceForensics dataset, deep learning methods called DeepFakes and Faceswap	2
Figure 2. Illustration of a fully connected feed-forward neural network with multiple hidden layers(l1,l2,l3,l4). Each node in one layer is connected to all nodes in the subsequent layer	16
Figure 3. Illustration of a basic Convolutional Neural Network pipeline, consisting of convolution, activation, pooling, and fully connected layers leading to classification output	19
Figure 4. Visualization of a kernel filter calculation	20
Figure 5. Comparison of CNN architectures on ImageNet, showing Top-1 accuracy versus parameter count. EfficientNet achieves higher accuracy with fewer parameters through compound scaling	23
Figure 6. Deepfake Generation using an auto-encoder and a decoder	25
Figure 7. Expressions of the source actor are transferred onto the unmodified target face, resulting in a manipulated output in which the targets identity outside of the face scope is preserved but the facial features are altered	26
Figure 8. Face swapping using Faceshifter model results showing the source faces facial characteristics transfer onto the target subject. The algorithm maintains the target's non-facial attributes while replacing facial features with those of the source	27

Figure 9. In this illustration following deepfake methods are shown, entire face synthesis, attribute manipulation, identity swap and expression swap	29
Figure 10. A synthetic image that is generated and retrieved from thispersondoesnotexist website	30
Figure 11. Image showing the effect of Face X-ray revealing the blending boundaries in forged and real images. A real image(a) and its face X-ray, (b) fake images and their corresponding face X-rays	35
Figure 12. Detection Pipeline of the award winning solution of DeeperForensics Challenge 2020	39
Figure 13. Showcasing a deepfake detection pipeline that utilizes the OpticalFlow values from each frame to train a CNN+LSTM pipeline	40
Figure 14. Image of a comparison table showcasing EfficientNet models outperforming other architectures according in a study done by (Tan & Le, 2019)	40
Figure 15. Samples from DFDC Dataset	43
Figure 16. Methodology Pipeline can be seen in this figure	45
Figure 17. All of the 9 experimental models can be seen here. Each of the meanings of these model names are given next to them	46
Figure 18. The figure showcases fake frames subjected to augmentation and cutout variants, where Image 1 is generated with random fill, Image 2 with black fill, and Image 3 with white fill	47
Figure 19. Real Image (on the left), Fake image (on the right) and the ssim difference map in the middle can be seen	48

Figure 20. The figure presents a real frame with a star-shaped cutout applied on the cheek–eye region. The interior of the cutout is randomly filled with pixel intensities, illustrating the random-fill variant of the star cutout augmentation	49
Figure 21. Baseline EfficientNet-B4 schema with 12 input frames at 224×224 resolution, processed in a TimeDistributed manner	55
Figure 22. Illustration of Gradcam Heatmap(left) and Regional Activation Chart of this heatmap (right) of a True Negative test set sample can be seen	67
Figure 23. Presents the baseline models standard deviation analysis of GradCAM activation patterns across eight facial regions for different prediction categories	68
Figure 24. True Negative (TN) activation analysis across facial regions for nine different model configurations. The values represent the average activation ratios (%) in correctly classified real samples, separated into localized facial regions	70
Figure 25. True Positive (TP) activation analysis across facial regions for nine model configurations	72
Figure 26. False Negative (FN) activation ratios across facial regions for nine model configurations	74
Figure 27. False Negative (FN) Activation ratios across facial regions for nine model configurations	75

CHAPTER 1

INTRODUCTION

1.1 Background

Photograph, audio, and video manipulations have been carried out for centuries. In the past, a photograph was altered through manual, physical retouching, and videos were altered through frame-by-frame addition of visual effects. These processes required a lot of time and manual skill. The spreading of the personal computer and developments in digital technology all gave rise to editing softwares for the use by the common user. This historical evolution, technological emergence, and societal implications of media manipulation form the background issues addressed by this thesis.

With the recent improvements in Artificial Intelligence, creating these altered images became even easier.

These deepfakes (a widely used term in the public to represent images altered by AIs) represent the most recent installment in this progression, with media being altered or generated using deep learning (a type of artificial intelligence) technique into new instances.

One of the most widely discussed examples showed a deepfake of President Volodymyr Zelenskyy, falsely showing him asking his soldiers to surrender during the 2022 Russian invasion (BBC News, 2022). Another famous event included a deepfake of Mark Zuckerberg, in which he is said to present a speech implicating himself in the theft of user data (Hern, 2019). These examples show that deepfakes can appear authentic to their viewers and, as studies have shown, may even go unnoticed as genuine content under normal viewing circumstances (Nightingale & Farid, 2022).

While in a few cases, use of such technology can be creative and instructive, its unlawful usage makes this one of the major social problems.

Rising levels of altered digital media realism and easy access to these methods have consequently made it possible to release large-scale datasets like the DeepFake Detection Challenge (DFDC) datasets (Dolhansky et al., 2020) for building detection mechanisms.

However, as the past few decades have seen gains in this area, the need to address how manipulated media are detected and prevented has become more pressing. Some examples of these deepfaked images can be seen in below.



Figure 1

Example of a deepfaked image using FaceForensics dataset, deep learning methods called DeepFakes and Faceswap. This image is from (Rössler et al., 2019)

1.2 Societal and Ethical Implications of Deepfakes

With the rise of synthetic media generation, deepfakes have ceased to be a mere engineering curiosity and have become a significant social concern. Although manipulated images still present a great danger, video deepfakes pose an even greater threat due to the heightened realism they offer. Temporal consistency and natural facial expressions offer an immersive experience that static images cannot provide (Rössler et al., 2019). This newfound realism only

empowers deepfakes to sway public opinion and undermine trust in visual evidence.

Weaponizing deepfake technology against people seems to be on the rise, with one of the most grievous implementations being the non-consensual pornographic form. Women constitute the majority of victims in this regard, with image and video content created against their will. Stories and substantiations abound demonstrating how the violations resulting from these attacks have caused immense trauma and lifetime damages of a psychological and professional nature (Paris & Donovan, 2019). The realism of video deepfakes amplifies these effects due to their convincing pairing of motion and speech, rendering all but the most expert detectors powerless (Nightingale & Farid, 2022).

Legal remedies for these harms vary from one jurisdiction to another. Some jurisdictions have established specific criminal prohibitions, while others apply pre-existing laws, such as defamation or data protection legislation, which were not specifically designed with synthetic media in mind (Maras & Alexandrou, 2021).

Deepfakes, being a widespread technology, have therefore created a crisis of authenticity in communication within the digital domain. The concept of the "liar's dividend" has emerged, where the mere existence of deepfake technology acts as a tool to discredit genuine evidence by introducing the possibility of fabrication (Chesney & Citron, 2019). Social media platforms have inadvertently aggravated this issue, as the almost simultaneous release of content precedes verification efforts (Vaccari & Chadwick, 2020). By the time fact-checkers are able to issue a response, millions of individuals would have already been fed misinformation. Particularly, electoral processes find themselves under threat from deepfake videos. Strategic fabrication releases are a tool to manipulate voter choices toward suppressing voter turnout or at least growing social discord (Vaccari & Chadwick, 2020; Kreps et al., 2020).

Another challenge is the international character of these online platforms, as they can create and spread these videos across borders, thus evading any regulations intended for the national electoral process. Deepfake ads also pose a threat to financial markets and corporate reputations alike. A fake video of the CEO announcing company bankruptcy or revealing critical secrets could, in theory, cause an immediate market reaction that cannot be verified (Westerlund, 2019). Large-scale incidents have, however, been minimal up to now, but such technical capabilities exist.

In recent times, explainability in detection has been increasingly prioritized for fostering trust and supporting potential legal compliance. Contemporary deepfake identification approaches, employing state-of-the-art neural network architectures, can achieve high levels of accuracy; however, they often lack clear interpretability, particularly in borderline cases. In still image analysis, heat maps can help highlight potentially manipulated regions. For video analysis, more elaborate techniques are being explored, such as network dissection to relate learned features to specific facial regions, and dynamic prototype methods to capture and visualise temporal inconsistencies (Mansoor, 2025; Trinh et al., 2020). These explainable approaches will be increasingly required with the rise of AI governance frameworks that emphasize transparency and accountability.

Deepfake threats have been handled differently from country to country, with South Korea introducing explicit criminal penalties for the malicious creation and distribution of deepfakes. In the United States, several legislative instruments, such as the Deepfakes Accountability Act, have been proposed, with a focus on disclosure requirements to regulate the disclosure of images and videos. (Maras & Alexandrou, 2021). The EU's Digital Services Act will require synthetic content to be labeled, although it does not comprehensively address the issues surrounding real-time detection. These patchwork regulations provide an enforcement gap and might lead to uncertainty over jurisdiction for deepfake matters. Ethical considerations surrounding

deepfakes address areas of accountability for and prevention of harms (Pawelec, 2025).

Researchers, developers, and platform operators must negotiate between supporting creative uses and blocking malicious uses. The immersive quality of video deepfakes makes this balancing act particularly challenging.

Addressing the deepfake problem requires coordination across various sectors. Provenance tracking and watermarking systems would be the first steps in authenticating content. Next, media literacy must be sufficiently developed to strengthen the public's resistance to manipulation. Legal frameworks would also benefit from greater harmonization to address cross-border enforcement challenges.

Finally, advances in deepfake detection should incorporate explainable methods to improve trust and transparency. Without such integrated measures, deepfake technology may continue to undermine respect for individuals, institutional credibility, and democratic discourse.

1.3 Research Motivation

The growth of AI-based synthetic media generation technologies has led to various contexts in which deepfakes can be used for harmful purposes, including disinformation campaigns, fraudulent activities, identity theft, and damage to personal reputation. These evolving threats have rendered the development of reliable detection methodologies increasingly urgent. The ability to distinguish between genuine and manipulated content generated by automated means has been recognized as both a technical imperative and a societal requirement. This significance is amplified by the pace at which generative AI and AI-based media manipulation techniques continue to advance, and the necessity of detection systems being continuously refined to address emerging manipulation strategies.

Within the technical domain, deepfake detection is acknowledged as a particularly demanding and evolving research challenge. The complexity lies

in developing models capable of identifying subtle and frequently imperceptible artifacts that are introduced through the generation process. Although deepfake creation commonly employs approaches such as Generative Adversarial Networks or Encoder-Decoder frameworks, this thesis primarily focuses on the detection aspect of this phenomenon. Consequently, attention has been directed toward Convolutional Neural Network (CNN) architectures, which have been shown to excel in image and video classification contexts, especially when identifying spatial anomalies and nuanced visual indicators.

The pursuit of high detection accuracy alone, however, has been deemed insufficient. The matter of explainability within deepfake detection frameworks has attracted substantial academic interest. While black-box predictions may achieve considerable accuracy, their nature presents limitations in understanding decision-making processes, a critical shortcoming when such systems are deployed in sensitive domains, including journalism, legal proceedings, or content verification.

Thus, this research incorporates methodologies for rendering model decisions more transparent, employing techniques such as heatmap-based attribution to better understand which portions of an image are most strongly influencing classification outcomes. Such interpretative approaches are considered essential, not merely for identifying model limitations, but for establishing confidence when these systems are implemented in practical scenarios.

Given these considerations, a systematic examination is undertaken to investigate how various augmentation and preprocessing techniques, including cutout-based modifications, influence CNN classification accuracy. Secondly, interpretability mechanisms are embedded within the evaluation framework to facilitate comprehensive accuracy assessment and behavioral analysis. Through the integration of quantitative performance measurements with qualitative interpretability observations, this work aims to contribute to the search for a more transparent, resilient, and interpretable methodology for deepfake detection.

1.4 Role of Explainability in Deepfake Detection

Explainability has become an important aspect for deepfake detection models when automated outputs are subject to technical or societal implications. Researchers have noted that as models become increasingly complex, describing their decision-making process becomes more challenging; therefore, describing these systems as "black boxes" may limit transparency (Abir et al., 2023). This has sparked discussion regarding the relevance of transparent methods, especially when trust and reliability become important.

To enhance explainability, the methods usually try to track the patterns or regions that are seemingly influential in the model's decisions.

Visual attribution methods, including heatmapping ones, such as LIME and Grad-CAM, have been investigated for their ability to highlight image regions that may influence classification results. Some studies proposed that LIME is able to detect important regions; however, its performance can vary based on the particular setting and datasets used (Tsigos et al., 2024).

Visual explanation assessment helps researchers determine whether models distinguish between meaningful features and spurious patterns. Even if model explanations do not always align with human intuition, that divergence can mean that automated systems detect subtle patterns that may not be immediately observable to human observers. Hence, integrating explanation techniques with conventional performance indicators may reveal clues about model behavior, which in turn may contribute to system enhancement (Aghasanli et al., 2023). Since deepfake methodology is constantly being updated, incorporating explainability into detection systems has been proposed as a valuable approach. Such integration may help provide clearer insights into model decisions while potentially contributing to the development of more transparent detection methods.

1.5 Structure of the Thesis

The thesis firstly provides the theoretical foundations of machine learning, with particular emphasis on deep learning concepts such as neural networks and convolutional neural networks. Then discussion focuses on the social and ethical implications of deepfakes and introduces the role of explainability in deepfake detection. Building on these foundations, the main learning paradigms are outlined, followed by a presentation of explainable AI methods relevant to computer vision, with a focus on Grad-CAM. Afterwards, the creation of deepfakes is examined. Some of the commonly used generative models are introduced, different types of deepfakes are outlined, and generation pipelines used in the real world are described. Related work section reviews existing detection methods, beginning with directed approaches that exploit specific weaknesses in image and video forgeries and continuing with undirected approaches in which models autonomously identify discriminative artifacts.

This review covers datasets and explainability techniques that are relevant to the domain. The methodology of the study is then presented, including the baseline detection pipeline on which this research builds, the preprocessing strategy, augmentation and cutout techniques, the deepfake detection model architecture, and the evaluation metrics. The integration of an explainability framework based on Grad-CAM is also described in detail. The experimental setup follows, introducing the baseline results, selected augmentation–cutout configurations, and case studies that highlight interpretability through Grad-CAM visualizations. The results are then presented and discussed, combining quantitative performance metrics with qualitative analyses of explainability. Finally, the thesis concludes by summarizing the main contributions, addressing current limitations, and outlining directions for future research.

CHAPTER 2

THEORETICAL FOUNDATIONS

2.1 Approaches in Machine Learning

Typically, machine learning methods are categorized into three main categories: supervised learning, unsupervised learning, and reinforcement learning. In supervised learning, models are trained on datasets that contain a known label for every input, allowing the model to learn a mapping of inputs to outputs (Heidari et al., 2024). Unsupervised learning, in contrast, works with unlabeled data and aims to uncover hidden patterns or structures without being provided with knowledge of the output. This approach has been employed, for instance, in deepfake detection, particularly in cases where high-quality, labeled datasets are unavailable (Gupta et al., 2024).

Reinforcement learning is the third paradigm, in which agents interact with an environment generating a stream of rewards and penalties that shape the formation of the best strategies (Kaur et al., 2024).

In deepfake generation and detection, supervised and unsupervised approaches have been preferred. Supervised learning leads the deepfake detection with CNNs and, later, transformer-based models that are trained on very large datasets of labeled real and fake media (Khan & Dang-Nguyen, 2023). Unsupervised or semi-supervised models have gained interest in situations where there is very little labeled data, employing reconstruction-based anomaly detection or cross-modal inconsistency analysis as tactics to detect anomalous content.

Given that reinforcement learning has not been widely used as a standalone method for deepfake detection, hybrid methods are now emerging. Hybrid methods combine reinforcement learning with supervised models, for instance, to adaptively select data augmentations that improve performance across multiple datasets (Kaur et al., 2024).

With the limited scope of reinforcement learning in this particular area, this thesis focuses on supervised and unsupervised approaches, where most recent advances have been made.

2.1.1 Supervised Learning

In supervised learning, a predictive model is trained on a dataset consisting of input–output pairs, where each pair contains an input vector and its corresponding target label. The aim is to determine a function $g: X \rightarrow Y$ that maps feature vectors $x \in X$ to their associated targets $y \in Y$, based on a finite collection of training examples $(x_i, y_i)_{i=1}^m$ (Goodfellow et al., 2016).

During training, the model parameters θ are adjusted so that the predictions $g(x; \theta)$ minimize the difference, as measured by a predefined loss function L , between the predicted and actual values. The ability of the model to generalize is evaluated not only on the training set but also on an independent test set, providing an estimate of the generalization error. This error can be approximated by the empirical risk, given by:

$$R_{\text{emp}}(g; Z_n) = \frac{1}{n} \sum_{i=1}^n L(y_i, g(x_i; \theta)) \quad (2.1)$$

where Z_n denotes the test set containing n independently and identically distributed (i.i.d.) samples. Since any model is learned from a finite dataset, the structural risk of overfitting must also be considered. Overfitting occurs when the model fits noise or dataset-specific irregularities instead of underlying patterns that can be applied to unseen data.

2.1.2 Unsupervised Learning

In unsupervised machine learning, data samples are composed of feature vectors X , without the label Y accompanying them. Finding patterns in the data occurs without prior knowledge of the correct output values (Sarker, 2021). Typical objectives include clustering, wherein data points are grouped into clusters based on certain similarity measures; estimation of density, in which the probability distribution underlying the dataset within the feature space is modeled. These methods reveal new insights into previously unknown relationships and hidden structures.

Additional unsupervised techniques are used in dimensionality reduction, where higher-dimensional data is projected onto its lower-dimensional representation. Usually, this process serves for purposes of visualization, noise elimination, or accelerating further processes. Without any annotated sample, these algorithms produce representations that are informative and interpretable by learning explicitly from the data structure itself. These properties make unsupervised learning a useful candidate for exploratory data analysis and a building block for more complex paradigms such as semi-supervised and self-supervised learning.

2.2 Fundamental Concepts in Machine Learning

2.2.1 Generalization and Overfitting

In definition, generalization in machine learning is the ability of a trained model, extending its performance or functionality to samples that were not included in its training process. For an excellent generalization, the model can find the underlying statistical structure of the dataset rather than storing individual training examples. It means that when the model does not have sufficient capacity, it will not find relevant patterns in the data and is underfitting, which results in high error rates at both the training and testing levels.

However, when the model has too much capacity and is not adequately regularized, it will become overly close to the training data, leading to the learning of not only informative patterns but also random fluctuations and noise. This is typically referred to as overfitting, which enables very low training errors but poor prediction performance on new, unseen data samples from the training points.

Fitting or underfitting has always been considered the biggest of all challenges in research on machine learning; importance has been given to it through relevant researchers who have cited many works because good learning demands a model that is neither too simplistic nor too highly specialized for the set of learning examples (Goodfellow et al., 2016).

The importance of balancing model complexity control, the availability of sufficient and diverse data, and developing an appropriate layout of steps has been found in the literature, which offers a way toward stronger generalization. Thus, the generality of a model can be considered the result of a delicate balancing act among data, architecture, and training strategies, which achieves harmony to avoid overfitting.

2.2.2 Regularization

Regularization is known as a set of techniques applied to control the learning dynamics of machine learning models by imposing additional constraints on their parameters. Instead of allowing parameters to be optimized without any restrictions, regularization enforces models that prefer solutions that are more stable. Usually, these mechanisms are incorporated into the training process by adding penalty terms to the loss function or by modifying the optimization dynamics in a way that restricts the effective model capacity.

Parameter type penalties are among the most prominent of the regularization methods. L1 regularization imposes a penalty proportional to the absolute magnitude of the weights, causing the parameter space to become sparse. In

contrast, L2 regularization imposes a penalty proportional to the square of the magnitude of the weights, encouraging smaller parameters that are more evenly spread out (Goodfellow et al., 2016). Apart from such deterministic penalties, stochastic approaches are also in widespread use.

Dropout is a well-known example in which the program randomly drops an arbitrary set of activations during any given pass through the network, thereby forcing the network to create unnecessary and redundant internal representations. From weight constraints to stochastic changes, these are considered by (Goodfellow et al., 2016) as among those methods that must be used to encourage solutions to stay stable, reliable, and useful in general.

Augmentations in the training data are, as suggested by (Goodfellow et al., 2016) , one of the best solutions to generate greater amounts of data to increase the generalization. Imposing a Gaussian blur and adding Gaussian Noise, rotating the image, and cropping the image datasets are used to generalize Machine Learning models. While generalization methods in Neural Networks are similar, random noise is also used to increase the noise robustness of the model.

2.2.3 Transfer Learning and Fine Tuning

Transfer learning is essentially a methodological orientation where experience gained in executing task A is used to enhance performance on task B, which is often related. In particular, learned representations from the large source dataset are reused to accelerate learning in the target domain, where labeled data may be scarce or nonexistent. Fine-tuning constitutes a predominant method used in this context, where a model pre-trained on a source task is merely initialized with learned parameters and then adapted to the target dataset. Depending upon how similar the domains are, either just the final layers are retrained while the rest of the network stays frozen, or the entire network is adjusted. It has been stated that these techniques reduce training time, significantly reduce the massive amt of annotated datasets required, and

often deliver better results than when the model is trained from scratch (Iman et al., 2023). Particularly, fine-tuning allows for the generic features to be converted into domain-specific ones while still retaining the generalizable knowledge encoded during pre-training.

2.2.4 Robustness and Noise

Robustness denotes a model's inherent competence to deliver resilient performance with complexity in the input data. In assessment and training with images, it is classically brought forth by deliberate synthetic noise: blurring, occlusions via cutouts, or color adjustments that truly mimic any hard-to-predict variabilities that are encountered in real-world scenarios.

By injecting these distortions, the training will ensure that the model learns representations that are more persistent and dependent on the underlying structure rather than on superficial artifacts. Specifically, Lopes et al. (2019) demonstrated that input corruption with patch-wise Gaussian noise (Patch Gaussian) confers greater robustness to corruption without detriment to accuracy on clean data, thereby balancing sensitivity across the frequency domain.

2.2.5 Explainable AI (XAI)

In the context of deep learning, explainability has become essential for understanding the decision-making processes of convolutional neural networks (CNNs). Grad-CAM represents a widely-used approach that generates class activation heatmaps by calculating gradients of class output scores with respect to feature maps from convolutional layers. These heatmaps indicate spatial regions that contribute most significantly to model decisions (Selvaraju et al., 2017).

Gradient-weighted Class Activation Mapping (Grad-Cam) can be applied to frames extracted from video sequences to help identify facial regions that most influence classification outcomes. Heatmaps generated through this method

are overlaid onto original denormalized frames, enabling visual interpretation of model focus areas.

This visualization approach can be employed on both correctly classified and misclassified examples of real and fake samples, providing a direct method for examining whether networks rely on semantically meaningful regions or spurious artifacts. Such attribution maps enhance transparency in CNN decision-making processes, supporting model evaluation and interpretability.

2.3 Neural Networks

Neural networks are one of the fundamental frameworks in deep learning, with origins in the interconnected structures of biological neurons. They contain several layers of computational units, each providing a transformation of the input into higher levels of abstractions. The mechanism works by taking in information inputs at one layer and propagating the information through one or many hidden layers before arriving at an output layer, adjusting parameters so that it yields minimum errors in prediction.

The following subsections are devoted to fully connected networks, as the most elementary yet widely studied forms of neural networks. Their dense connectivity lies the foundation upon which to build an understanding of how features are propagated and combined to make predictions. Following this, backpropagation is presented, which is the standard method of training parameters. Now, these two constructs represent the conceptual prerequisite needed for diving into more elaborate architecture types such as convolutional neural networks, which are therefore presented in the next section.

2.3.1 Fully Connected Networks

Introduced by Rosenblatt in 1958, the perceptron is the simplest type of artificial neuron that calculates a weighted sum of inputs and applies a nonlinear activation function to the result. This basic formulation led to the

invention of multi-layer perceptrons (MLPs), also known as fully connected neural networks, which are perhaps one of the oldest types of architecture to have been intensively studied in deep learning (Bishop, 2006; Goodfellow, Bengio, & Courville, 2016).

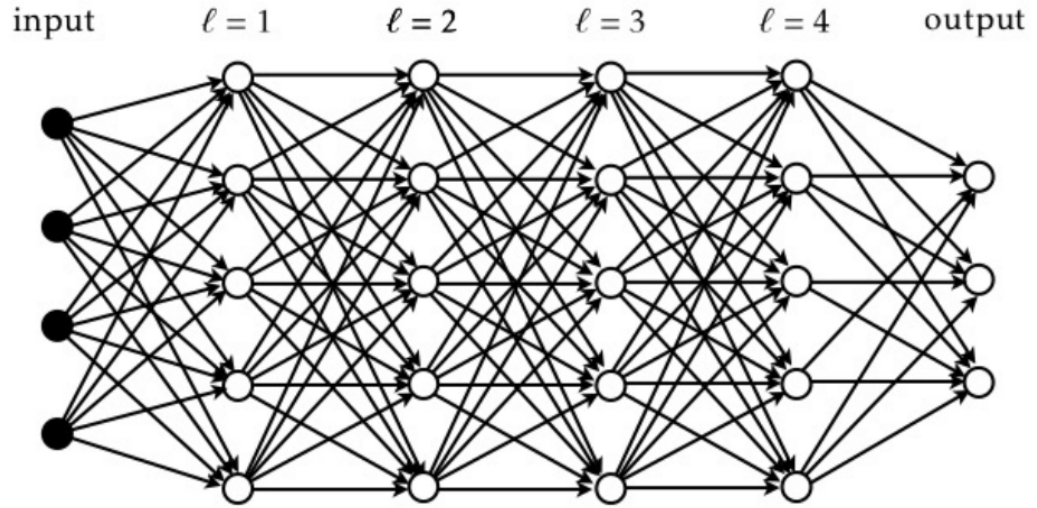


Figure 2

Illustration of a fully connected feed-forward neural network with multiple hidden layers ($\ell_1, \ell_2, \ell_3, \ell_4$). Each node in one layer is connected to all nodes in the subsequent layer. This figure is taken from (Mehlig, 2021).

In a fully connected neural network, each unit of a layer is connected to every unit of the subsequent layer as it can be seen on Figure 2. This dense connectivity allows the network to approximate a wide variety of nonlinear mappings between input and output spaces.

Such an arrangement is commonly illustrated as a feedforward architecture, in which information is transmitted in one direction from the input toward the output layer (Figure 2). The adjustment of parameters requires an additional mechanism in which error signals are propagated backwards through the network, a process that will be examined in detail in the following subsection. Mathematically, the activation of a unit i in the first hidden layer can be expressed as

$$a_i = \sum_{j=1}^n w_{ij}^{(1)} x_j + b_i^{(1)} \quad (2.2)$$

Where $w_{ij}^{(1)}$ are the weights that are learned, x_j represents the input features, and $b_i^{(1)}$ is the bias term. Resulting activation is transformed by a nonlinear function denoted as

$$z_i = \sigma(a_i) \quad (2.3)$$

where σ can be a sigmoid, hyperbolic tangent, or rectified linear unit (ReLU). Through successive layers, this transformation enables the network to capture hierarchical and increasingly abstract representations of the input data (Bishop, 2006).

The architecture of a fully connected neural network (FCNN) can be explained in terms of its the number of layers (depth) and number of units per layer (width) . Networks with greater numbers of depth and suitable nonlinear activation functions have been proven to approximate any continuous function on compact subsets of $\mathbb{R}^n$, which has led to their characterization as universal approximators (Cybenko, 1989; Hornik, 1991).

This theoretical result forms a cornerstone of deep learning research, clearly stating that neural networks can, in theory, represent arbitrary, highly complex mappings, given enough capacity and training data (Goodfellow et al., 2016). Yet that expressive power is a counterbalanced against the practical capability of fully connected architectures. Because these architectures consider all units unrestrictively connected, they get computationally expensive and prone to overfitting if treated as classifiers on high-dimensional data such as images. Hence, they have rarely been used alone in large-scale vision subjects. The theoretical standpoint thus becomes the conceptual and mathematical base of

more advanced networks, including convolutional neural networks, which will be discussed in the following section (LeCun, Bengio, & Hinton, 2015).

2.3.2 Backpropagation

The adjustment of weights in a neural network is achieved through the backpropagation algorithm. After the forward computation, the error between the predicted output y' and the true label y is evaluated by means of a loss function $L(y', y)$. The gradient of this loss with respect to the network parameters is then obtained by applying the chain rule in a recursive manner, beginning at the output layer and propagating backwards through the hidden layers (Rumelhart, Hinton, & Williams, 1986; Bishop, 2006).

For a unit in layer l , the error signal can be expressed as

$$\delta^{(l)} = (W^{l+1})^T \delta^{l+1} \odot \sigma'(a^{(l)}) \quad (2.4)$$

where $\sigma'(a^{(l)})$ denotes the derivative of the activation function, and $\odot$ is the element-wise product.

Using this error term, the gradients with respect to their weights and biases are given by

$$\frac{\partial L}{\partial W^{(l)}} = \delta^{(l)} (h^{l-1})^T \quad (2.5)$$

$$\frac{\partial L}{\partial b^{(l)}} = \delta^{(l)} \quad (2.6)$$

These expressions are highly effective in computing the influence of each parameter on the total error. After gradient computation, an optimization method such as stochastic gradient descent is used for parameter updating, where the gradient is estimated on randomly selected subsets of the training data rather than on the entire dataset (Bottou, 2010). Through many iterations,

this process helps the network reduce its loss function and learn a suitable mapping from the input to the output (Goodfellow, Bengio, & Courville, 2016).

2.4 Convolutional Neural Networks

Convolutional neural networks are a certain form of artificial neural networks, especially created for handling data in spatial terms primarily with regards to the images. They operate with the assumption that proximal data points typically bear more important information than those offset by large distances, thereby indirectly incorporating a behavioral predisposition that greatly facilitates learning process optimization (Goodfellow et al., 2016).

The standard CNN architecture is composed of several factors convolutional layers, nonlinear activation functions, normalization techniques, and optional pooling operations. These elements work collectively to extract feature representations in a hierarchical manner. The basic CNN processing pipeline has been illustrated in Figure 3.

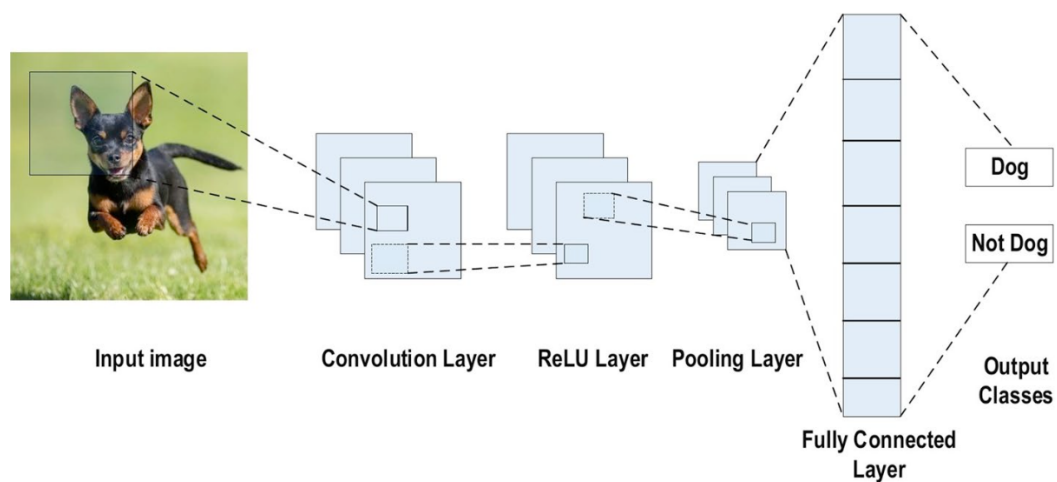


Figure 3

Illustration of a basic Convolutional Neural Network pipeline, consisting of convolution, activation, pooling, and fully connected layers leading to classification output. (Alzubaidi et al. (2021).

2.4.1 Convolution

Convolution is the main of operation CNNs, where the local regions of an image are processed by small kernels to extract features. A kernel (also known as filter) is a small matrix of weights that is learnable, applied across the input image to detect specific patterns such as edges, textures, or higher-level motifs (Goodfellow et al., 2016).

Mathematically, a two-dimensional convolution between an input I and a kernel K at position (i, j) can be expressed as:

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n) \quad (2.7)$$

This operation implements sparse local connections as well as weight sharing, which results in a low number of parameters compared to a fully connected network (LeCun, Bottou, and Orr, 1998). The convolution operation typically consists of using padding and stride for the purpose of regulating the size of the output. The sample calculation of a kernel filter is presented in Figure 4.

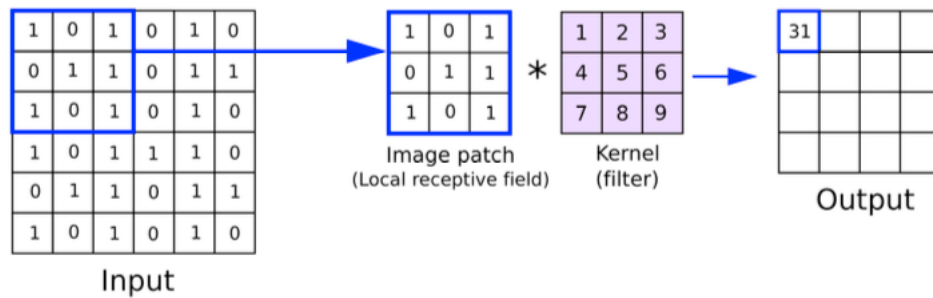


Figure 4

Visualization of a kernel filter calculation. This image is from (Reynolds, 2019)

2.4.2 Pooling

Pooling operations function to reduce the spatial resolution of feature maps while preserving critical information, consequently enhancing computational efficiency and introducing translation invariance properties

(Goodfellow et al., 2016). Maximum pooling extracts the highest activation value within a defined local region, while average pooling calculates the mean of activations. These pooling layers serve to aggregate activations and enhance the network's ability to generalize across minor input translations.

Global Average Pooling (GAP) has emerged as a popular replacement for fully connected layers in classification stages, effectively reducing each feature map to a single representative value.

2.4.3 Batch Normalization

Training very deep networks can be slowed down or even destabilized by the internal covariate shift (ICS) of the intermediate representations (or activations) of the neurons. ICS can be understood as the distributional shift of layer inputs that a layer must learn to model over the course of training. Batch Normalization (BN) was developed to counteract this problem by normalizing the activations of each layer in each mini-batch.

$$x'_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \quad (2.8)$$

$$y_i = \gamma x'_i + \beta \quad (2.9)$$

In this formulation, μ_B and σ_B represent the batch mean and variance respectively, while γ and β constitute learnable parameters for scaling and shifting. Batch normalization facilitates accelerated training processes, enables the utilization of higher learning rates, and diminishes sensitivity to

parameter initialization. This technique has been integrated as a fundamental component in contemporary CNN architectures, particularly in residual and mobile network designs. (Ioffe & Szegedy, 2015).

2.4.4 Evolution of CNN Architectures

The architecture of CNNs has seen major improvements in the last two decades. Considered a classical network architecture, LeNet-5 (LeCun et al., 1998) successfully demonstrated the suitability of convolutions and pooling layers for simple recognition tasks. However, as deep learning started shining more with larger image datasets, deeper architectures started emerging. VGGNet (Simonyan & Zisserman, 2015) set state-of-the-art results on such large, image-based datasets with the philosophy of 'deeper is better'. VGGNet uses even more convolutional layers than its predecessor, emphasizing this concept of simple 3x3 kernels that can translate into a structurally powerful deep neural network. Focus of the following innovations switched to efficiency improvements. Those enhancements were introduced in a way that methods with different sizes are parallelized inside the Inception modules (Szegedy et al., 2015); whereas the remaining developments just further simplified the design, with Xception (Chollet, 2017) and MobileNetV2 (Sandler et al., 2018) distilling the architecture into depthwise separable convolutions and inverted residual blocks for use in reducing computation cost.

Adding channel-wise attention with Squeeze and Excitation blocks (Hu et al., 2018) these improved performance even more across the board. In Figure 5 comparison of accuracy performance for these models can be seen.

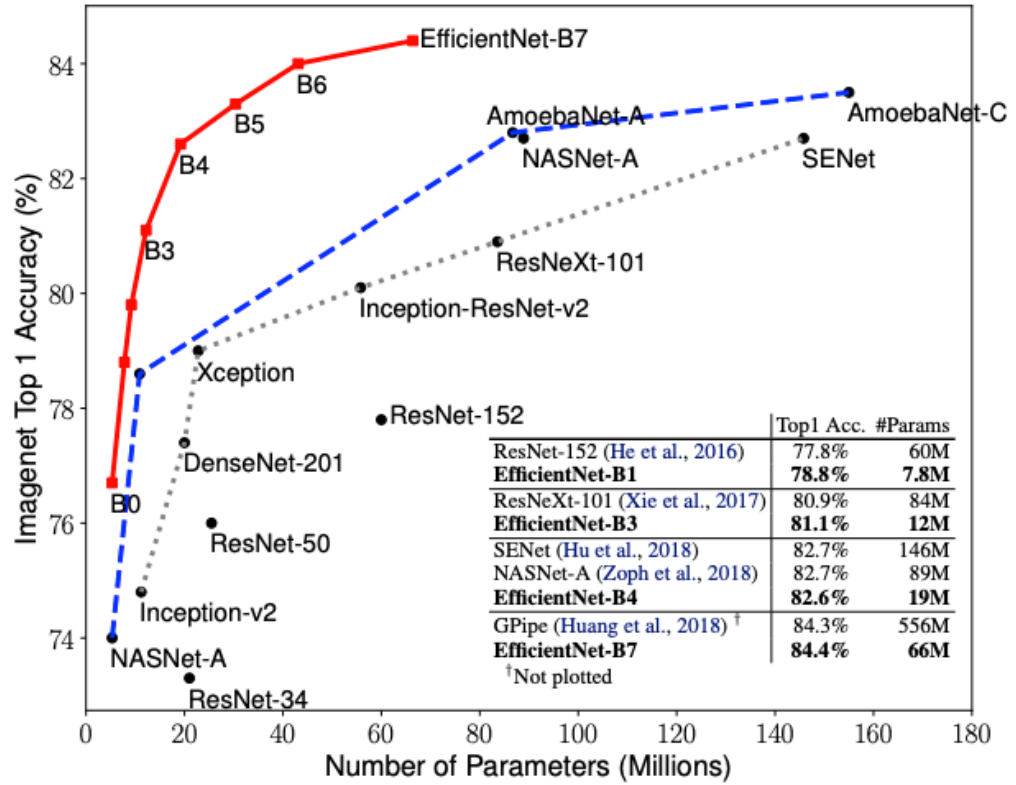


Figure 5

Comparison of CNN architectures on ImageNet, showing Top-1 accuracy versus parameter count. EfficientNet achieves higher accuracy with fewer parameters through compound scaling (Tan & Le, 2019).

The EfficientNets family (Tan & Le, 2019) unified previous proposals for efficient model design through the introduction of a simple compound scaling method to uniformly scale depth, width, and resolution of a network. Its main building blocks comprise inverted residual modules that distill depthwise separable convolutions, Batch Normalization, and Squeeze-and-Excitation mechanisms. The resulting backbone network is hence both fast and accurate, finding usage in a broad set of image and video understanding tasks.

CHAPTER 3

DEEPPFAKE CREATION

In this section deepfake creation methods are inspected. In order to find ways to improve the accuracy and explainability of the current state of the art models, it is crucial to have deep understanding on creation of deepfakes. Since the experiments of this thesis focuses on non-GAN (generative adversarial network) generated deepfakes, these models will be examined in greater detail.

3.1 Deepfake Creation Models

Deepfake creation models are a diverse set of generative approaches designed to manipulate facial appearances or behaviors in image or video content. While these models have differences in the way they encode, transfer, or completely regenerate facial features, they all share the common goal of producing photorealistic manipulations that are difficult to distinguish from authentic recordings. Depending on the method, these manipulations may involve identity swapping, reenacting expressions, or synthesizing entirely new appearances. The following subsections provides an overview of the main deepfake generation model families, such as encoder-decoder approaches, reenactment methods, advanced face-swapping architectures, and image-to-image translation frameworks. Their technical principles are briefly mentioned together with the artifacts they typically introduce, as these weaknesses create the basis of many deepfake detection strategies.

3.1.1 Encoder-Decoder Models

Encoder–decoder architectures have been traditionally used in early deepfake generation systems. An autoencoder processes an image of a face by compression inside a latent space via an encoder and reconstructs it through a decoder (Goodfellow et al., 2016).

The idea is now shared and adapted for deepfake generation with one encoder and multiple identity-specific decoders, each decoder learning to reconstruct the other's facial identity. (Tolosana et al., 2020).

This, allows expression and pose to be effectively mapped from one source person to the target person as it can be seen from Figure 6.

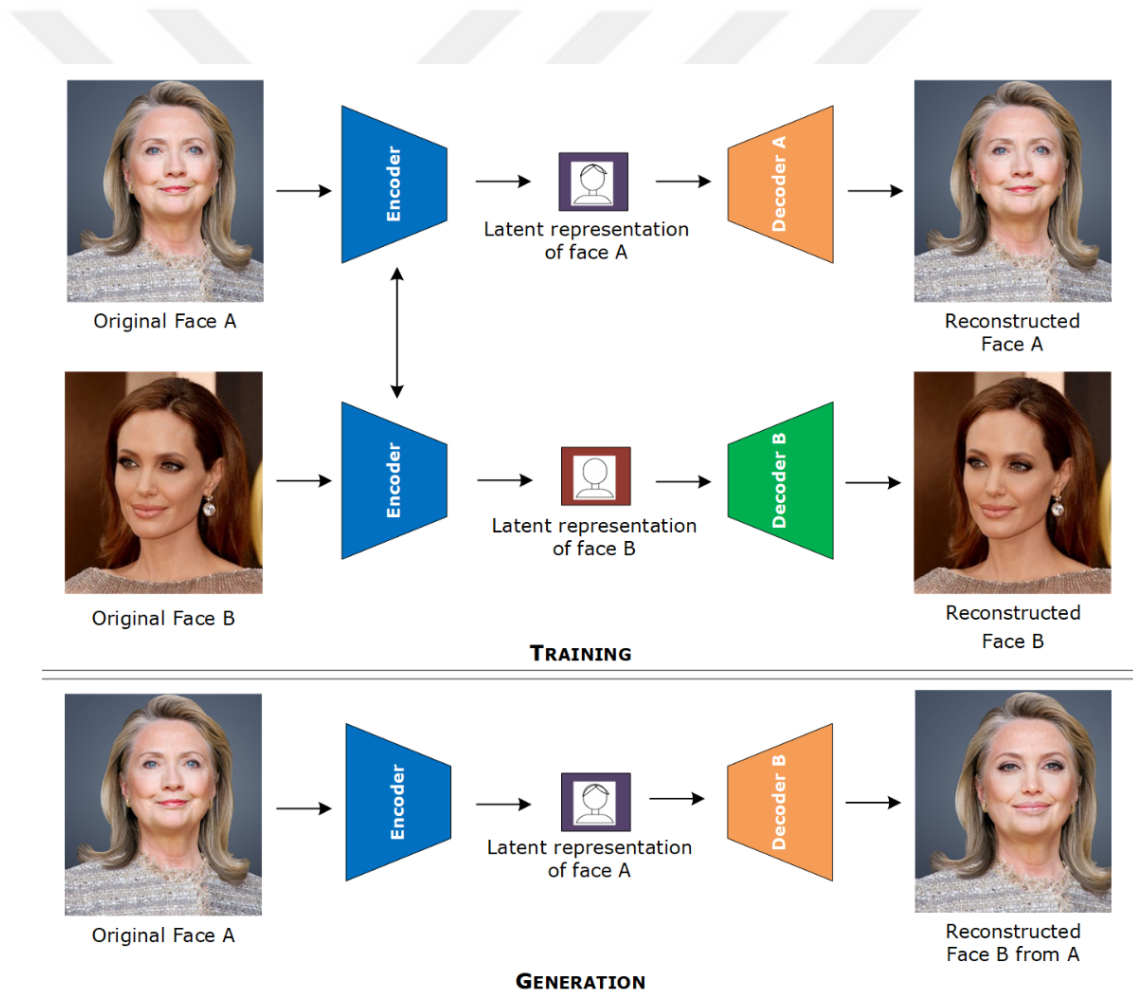


Figure 6

Deepfake Generation using an auto-encoder and a decoder (Image from (Masood et al., 2023))

While this models can generate convincing deep faked visuals, low-resolution synthesized faces may lead to inconsistencies and boundary artifacts while blending. In geometric alignment of faces forensic cues for detection might appear. (Li & Lyu, 2019).

3.1.2 Reenactment Based Deepfake Methods

In this deepfake method the aim is to transfer the facial expressions and movements of a source onto a target individual while preserving the target's identity. Unlike the approach of encoder–decoder models, which mainly focuses on identity replacement, these methods manipulate the dynamics of facial structures. The effect of this reenactment based deepfake method can be seen in Figure 7.



Figure 7

Expressions of the source actor are transferred onto the unmodified target face, resulting in a manipulated output in which the targets identity outside of the face scope is preserved but the facial features are altered. (Masood et al., 2023)

Face2Face demonstrates real-time expression transfer by directly controlling the geometry of the target face (Thies et al., 2016). Despite their sophisticated results, such techniques often introduce artifacts, most notably temporal jitter

in the mouth and around the jaw region, as well as inconsistencies in the texture of the face, which may be exploited by detection systems.

3.1.3 Second Generation Face Swapping Techniques

This generation of deepfake generation models, also known as second-generation models (developed subsequent to encoder-decoder based approaches), is also referred to as occlusion-aware methods. A state-of-the-art framework named Faceshifter has been introduced as a two-stage framework designed to improve the quality of face swaps. Firstly, the model combines the identity of the source face with their pose, facial expressions, and the lighting of the target. Next, a network focuses on refinement and corrects regions that are partly covered by objects such as glasses, hair, headdress or hands.



Figure 8

Face swapping using Faceshifter model results showing the source faces facial characteristics transfer onto the target subject. The algorithm maintains the target's non-facial attributes while replacing facial features with those of the source. (Image from : (Li, Bao, Yang, Chen, & Wen, 2019).

In Figure 8 it is demonstrated that facial swapping can be accomplished by Faceshifter even when faces are partially occluded by objects. Caused by this

ability to handle partially hidden areas, these methods are described as occlusion-aware.

While it reduces many of the obvious artifacts, smaller inconsistencies remain and can still be exploited by detection systems (Li, Bao, Yang, Chen, & Wen, 2019).

3.1.4 Generative Artificial Networks in Deepfake Generation

Generative Adversarial Networks (GANs) are formed by two neural networks trained in opposition, one generator creates synthetic samples while the other discriminator establishes real versus fake inputs. Subsequently, adversarial training is the gradual improvement of the generator's capability to produce outputs. GANs differ from autoencoders in that they learn to generate new samples from the data distribution, rather than reconstructing existing data. GANs may still yield unrealistic images, thereby causing the appearance of artifacts such as texture inconsistencies or domain shifts. This researches interest will be primarily directed toward non-GAN methods, which show clearly different traits and pose separate challenges for their detection, notwithstanding the marvelous achievement of the GAN-based deepfake generation.

3.2 Types of Deepfakes

According to Tolosana et al. (2020), deepfake techniques can be classified into four distinct categories depending on the degree of manipulation and the goal of the operation. The categories in Figure 9 are visually illustrated, with a brief description of each being provided in the following subsections.

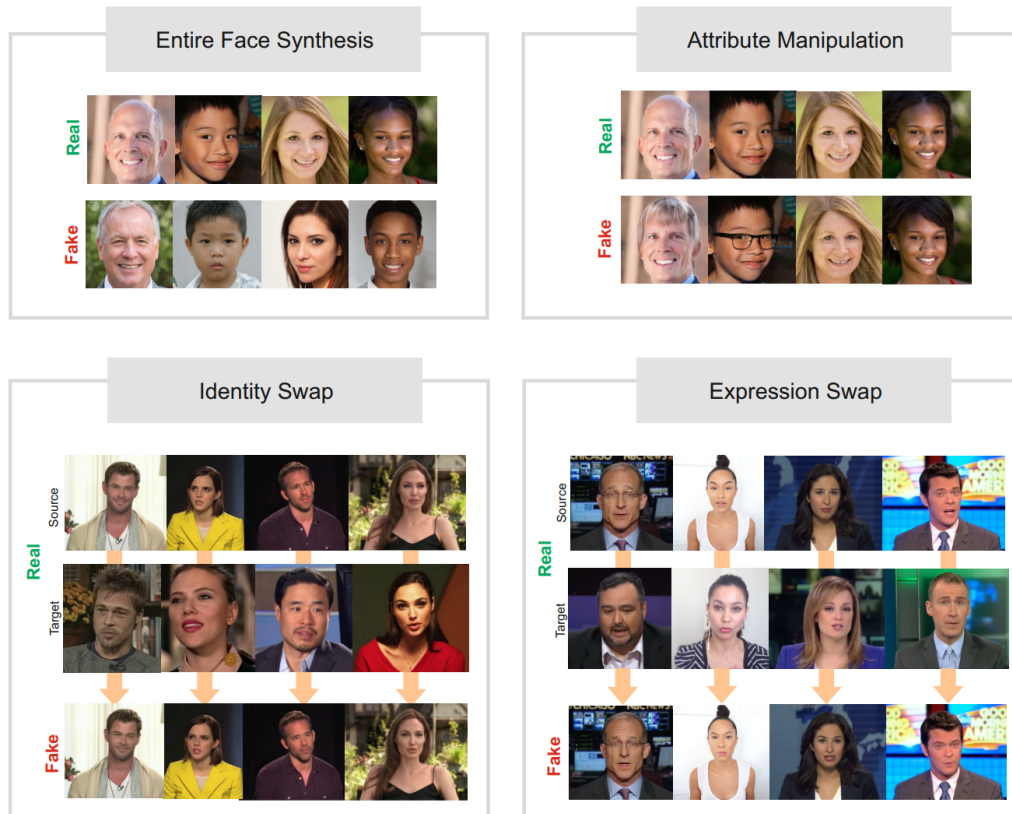


Figure 9

In this illustration following deepfake methods are shown, entire face synthesis, attribute manipulation, identity swap and expression swap. (Tolosana et al., 2020).

3.2.1 Entire Face Synthesis

Generation of a complete synthesis of a face involves procedures that create or process fully artificial face images, often to the extent of being declared almost indistinguishable from real photographs. Previously, such facial synthesis was largely accomplished using generative adversarial networks (GANs). A major breakthrough in the domain of entire face synthesis was StyleGAN, which introduced a style-based generator architecture for producing highly realistic face images (Karras et al., 2019). The popularity of the method was well illustrated by the public website “This Person Does Not Exist,” which showcased the capacity of GANs to fabricate believable fake identities.



Figure 10

A synthetic image that is generated and retrieved from thispersondoesnotexist website. (Tero Karras, 2019)

Further enhancements were made in StyleGAN2 by reducing visual artifacts and increasing image fidelity (Karras et al., 2020). In Figure 10, a synthetic image that is generated using StyleGAN2 of a baby's photograph can be seen. With StyleGAN3, alias-free generation has been introduced for improved realism. (Karras et al., 2021). Following that, face synthesis has experienced a resurgence with diffusion-based methodologies emerging as worthy competitors to GANs. In Early 2025, with MorphFace 3D morphable models that can guide diffusion to generate novel identities with realistic pose, expression, and style variations (Zhang et al., 2025). HyperFace, in the meantime, put forth a hypersphere-based embedding scheme to generate synthetic datasets with better inter and intra-class diversity for recognition tasks (Lin et al., 2025). These developments set the scene for a shift from an absolutely GAN-centered synthesis to hybrid and diffusion-based architectures, which meant a considerable step forward in both realism and utility.

3.2.2 Expression swap

The transfer of the facial expression and pose of the source individual to the target face while retaining identity-specific attributes of the target is known as Expression Swapping. Conventionally, these types of methods were mostly GAN architecture approaches that suffered from irregular training and the introduction of artifacts. However, more recent research has focused on diffusion-based and hybrid frameworks, which promise even more stability and realism.

For example, the Pose and Facial Expression Transfer model manipulates StyleGAN2 latent representations through a dual encoder and mapping network to transfer expressions and poses in an unsupervised manner at nearly real-time speeds (Zhou et al., 2025). Another groundbreaking model, called the DynamicFace, is an expression-driven diffusion framework that utilizes adaptive attention layers to balance expression fidelity and identity preservation, yielding highly realistic outcomes for both static images and even video sequences (Li et al., 2025).

Likewise, in DiffFace, an identity-conditioned denoising diffusion probabilistic model substitutes for GAN-based training, yielding more consistent and robust expression transfer across widely differing scenarios (Chen et al., 2025). This provides a clear illustration of the migration from the formerly GAN-dominated paradigm to diffusion-guided architectures, which offer better expression transfer accuracy while preserving the visual integrity of the synthesized identity.

3.2.3 Identity Swap

The identity swap is the most recognized type and a commonly applied form of deepfake manipulation. Here, the facial identity of the source is applied to that of a designated person in images or video sequences, and this is done in such a way that it appears the target individual is performing the actions of the source. While previous attempts were primarily focused on computer graphics techniques, more recent research has found that deep learning-based approaches have taken center stage, as they can generally achieve much more realistic and temporally consistent results (Tolosana et al., 2020). Among the available tools in the public domain, DeepFaceLab is often referred to as one of the best and most flexible tools for performing identity swaps (Perov et al., 2020).

3.2.4 Attribute manipulation

The modification of specifically selected attributes while preserving the associated identity of the person is known as attribute manipulation. Common facial attributes changes might involve modifications to hair color, age, facial hair, or expressions to enhance or alter the perceived appearance of the person. Generally, these changes are made using generative models that learn to disentangle facial attributes in a latent space, allowing for direct editing without replacing the entire face. In some cases, however, these methods may generate somewhat inconsistent results, especially around fine details in texturing or illumination, which can be exploited by detection approaches.(Tolosana et al., 2020).

CHAPTER 4

RELATED WORK

In this section, related work on approaches that detect artifacts in deepfake-generated content is examined. These approaches can be divided into two main categories; Feature-Based Deepfake Detection Approaches and Learning-Based Deepfake Detection Approaches. Feature-based methods detect deepfakes by identifying residual artifacts that are left by the generative model used to create the examined image or video content. Learning-based methods are primarily classification-based approaches that are specifically trained to address this detection task.

4.1 Feature-Based Deepfake Detection Methods

Feature-based deepfake detection methods operate by recognizing and analyzing residual artifacts that emerge during the synthesis process, targeting specific inconsistencies that generative models may inadvertently introduce during content manipulation. In the literature, investigations in this area have been separated into two major analytical perspectives; spatial and temporal artifact detection methods. Spatial detection methods attempt to trace and capture deepfake-related artifacts appearing in individual frames, which include boundary blurring, color inconsistencies, resolution discrepancies, and compression artifacts coming into existence through the manipulation process. Spatial analysis methods have been effectively employed in detecting blending boundaries in fake images (Li et al., 2020) or frequency-domain anomalies caused by up-sampling artifacts (Liu et al., 2021) and geometric distortions from imperfectly synthesized faces. Temporal detection aims to identify inconsistencies between consecutive frames, assuming that deepfake videos are often rendered frame-by-frame,

resulting in inconsistencies within frame sequences. Recent studies have identified novel, under-exploited temporal forgery artifacts, such as Facial Feature Drift (FFD), whereby subtle, inconsistent artifacts between frames coexist in the position and shape of facial organs as a result of frame-by-frame face-swapping operations (Song et al., 2024). Contemporary temporal detection systems can also be augmented with advanced motion analysis methods, such as the Anchor-Mesh Motion (AMM) algorithm, which models the precise geometric motion of facial micro-expressions (Wu et al., 2022).

Some state-of-the-art methods have turned to pixel-wise temporal-frequency analysis, which involves performing Fourier transforms along the temporal axis to derive features that are highly sensitive to temporal inconsistencies within regions of interest, correlated with unnatural motions (Yoon et al., 2025). These feature-based methods form the backbone for modern detection systems by capitalizing on the inherent technical constraints of current generation technologies, hence providing robust detection capabilities that can adapt to evolving synthesis methods.

4.1.1 Spatial Feature-Based Detection Approaches

Blending artifacts constitute one of the primary detection mechanisms, wherein inconsistencies are detected at the boundaries of where synthesized facial regions are joined into the original background or neck regions (Li et al., 2020; Shiohara & Yamasaki, 2022). These artifacts may manifest as unnatural color transitions, illumination mismatches, or resolution discrepancies, where generative models struggle to blend the swapped face smoothly with its surrounding context. Blending artifact detection techniques typically include edge detection and gradient analysis, whereby sudden changes in pixel intensity or color distribution that would not normally occur in authentic footage are detected. This method works well against face-swapping techniques that show clear boundary irregularities during the manipulation stage.

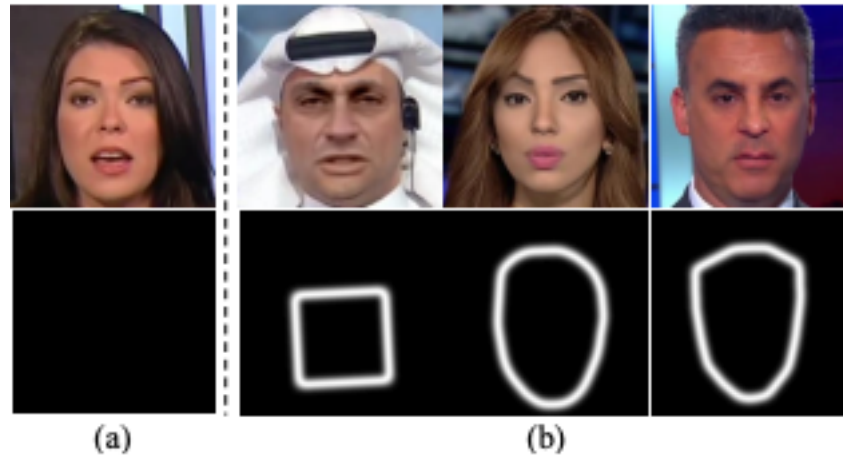


Figure 11

Image showing the effect of Face X-ray revealing the blending boundaries in forged and real images. A real image (a) and its face X-ray, (b) fake images and their corresponding face X-rays. (Li et al., 2020).

Face X-ray method is a high-level detection technique that detects internal structural inconsistencies in the facial regions by looking at blending borderlines that disclose image decomposition from different sources. (Li et al., 2020). FaceX-ray operates on these boundaries that are generated in training, these boundaries can be seen in Figure 11. This technique works under the assumption that nearly all existing face manipulation methods share a common step: blending the altered face into an existing background image, a process that leaves behind artifacts in the underlying image statistics. The X-ray method outputs a grayscale image which depicts the blending boundary for forged images while also indicating no blending for authentic content, thereby resulting in robust detection capabilities that remain valid even when tested with forgeries generated through face manipulation techniques that were never seen before.

Frequency domain anomaly detection involves spectrally analyzing signals and primarily using the FFT or DCT to examine irregular patterns in the frequency spectrum that are characteristic of artificially generated content (Qian et al., 2020). These anomalies typically emerge as a result of the upsampling, downsampling, or compression upon generative model pipelines,

thereby instilling periodic artifacts or unnatural frequency distribution that would usually not be present in a natural setting. The comparison via this approach analyzes frequency-aware decomposed image components, coupled with local frequency statistics, to identify suspicious patterns of synthetic manipulation. It generally performs well against deepfakes that may exhibit compression artifacts or subtle forensic traces that the spatial domain might miss.

In Early 2019, Generative model fingerprinting approaches observed that different GAN architectures leave characteristic noise patterns, or "fingerprints," on the generated outputs during the synthesis process. These fingerprints present themselves as subtle yet recognizable artifacts that accompany the training and architectural attributes of the generative model. (Yu et al., 2019) Early works demonstrated that such a presence of fingerprints could be used reliably for detecting synthetic content and for attributing it to some specific generation methods. Later works demonstrated that such fingerprints could be removed or masked effectively through a range of post-processing techniques, adversarial means, or even through arbitrary noise injection, thus presenting an obstacle to the long-term viability of such fingerprint-based detection as a sole approach. (Neves et al, 2020)

Local texture features utilize traditional computer vision(CV) methods, such as Local Binary Patterns (LBP) and Gray Level Co-occurrence Matrix (GLCM), to detect synthetic facial content by analyzing small-scale texture patterns (Ojala et al., 2002). These techniques examine how pixels are arranged in local areas to identify unnatural patterns in skin texture and facial details that occur when content is artificially generated. Local texture analysis is computationally simple, works reliably under different lighting conditions, and can detect basic textural problems that appear across different generation methods, providing a practical foundation for detection systems.

4.1.2 Temporal Feature-Based Detection Approaches

Methods based on temporal features analyze inconsistencies across video frames that could be linked to deepfake artifacts, deriving from the assumption that temporal inconsistencies exist in deepfake videos since processing is done on a frame-by-frame basis.

Frame-to-frame inconsistencies are centered on detecting any positional or morphological variation of facial features from one frame to the next. The very basis of this approach lies in the observation that deepfake generation procedures will often manipulate each individual frame independently and, thus, engender some variations that can be detected with respect to the position and shape of facial features. Recent studies have identified temporal forgery artifacts such as subtle inconsistencies in facial organ positions that emerge as a consequence of frame-by-frame face-swapping operations. These inconsistencies arise because the generative models have difficulty maintaining perfect geometric consistency over temporal sequences.

Optical flow analysis inspects motion vectors and any abnormalities arising within a video sequence, concentrating on motion boundary inconsistencies and unnatural motion patterns. The Anchor-Mesh Motion (AMM) algorithm, for example, has been introduced to accurately model the geometric motion of facial micro-expressions to detect even the most subtle temporal-facial artifacts (Wu et al., 2022). These approaches thus have the strongest ability to detect irregularities in facial dynamics and micro-expression dynamics that come into existence when synthetic faces are injected into real video sequences.

Assessment of temporal coherence considers the overall consistency of a video and the smoothness of frame transitions to locate synthetic content. This method analyses global temporal dependencies and captures video-level artifacts such as flickering effects and abrupt changes on faces. Contemporary implementations include temporal-frequency analysis methods that perform

transformations along the temporal axis to extract features that are sensitive to temporal inconsistencies with respect to unnatural motions.

4.2 Learning Based Detection Approaches

Deepfake detection methods based on learning (also known as undirected approach) differ fundamentally from feature-based ones in that they require the supervised training of discriminating patterns directly from labeled datasets composed of both genuine and manipulated content. The core principle of learning-based detection approaches is to use a computer program, mainly deep neural networks, to extract features automatically and build a classification model discriminating between real and fake media. These learning-based approaches are highly dependent on the completeness of their training datasets and how well they generalize upon encountering different manipulations and compression levels (Rossler et al., 2019).

Feature-based models typically explicitly look for artifacts or inconsistencies, whereas learning-based methods learn implicit representations from vast amounts of training data and thus can detect often very complex patterns that are not easily identifiable through classical analysis methods.

4.2.1 Classification Approaches

The methods of classification-based deepfake detection utilize a supervised learning model to directly achieve either binary or multi-class classification, thereby distinguishing between genuine and fabricated content. Generally, these deepfake detection approaches utilize convolutional neural networks (CNNs) that are pre-trained on large-scale datasets, such as ImageNet, and then fine-tuned for deepfake detection tasks.

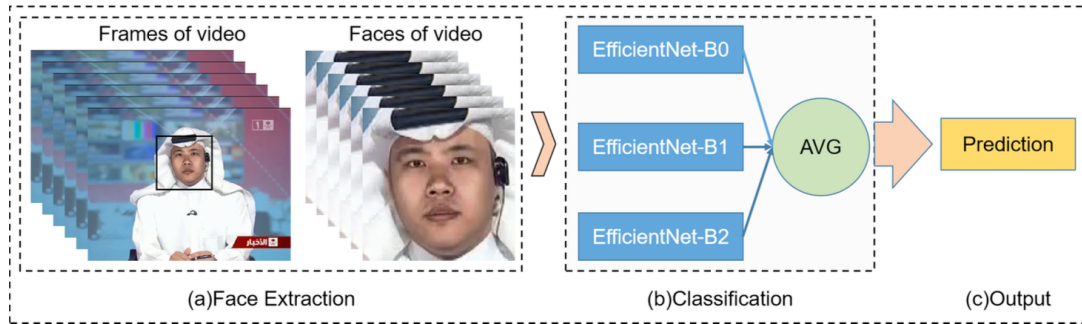


Figure 12

Detection Pipeline of the award winning solution of DeeperForensics Challenge 2020. (Jiang et al., 2021)

Classification pipelines in deepfake video detection are generally implemented through two primary methodologies. The first approach employs individual CNN (or RNN) architectures such as EfficientNet or ResNet that process extracted and cropped facial regions from video frames, utilizing binary labels for supervised training. The second method uses temporal modeling through LSTM networks that complement the spatial feature extraction capabilities of CNN architectures, which aids the of capturing sequential knowledge across frames. Implementation of the first pipeline, as illustrated in Figure 11, demonstrates the process where, videos are segmented into individual frames, facial regions are extracted and cropped, and then fed into classification networks with corresponding labels. In the specific solution presented, multiple models are trained independently from each other. Their predictions are combined in an ensemble learning technique to produce the final binary classification of pristine versus manipulated content.

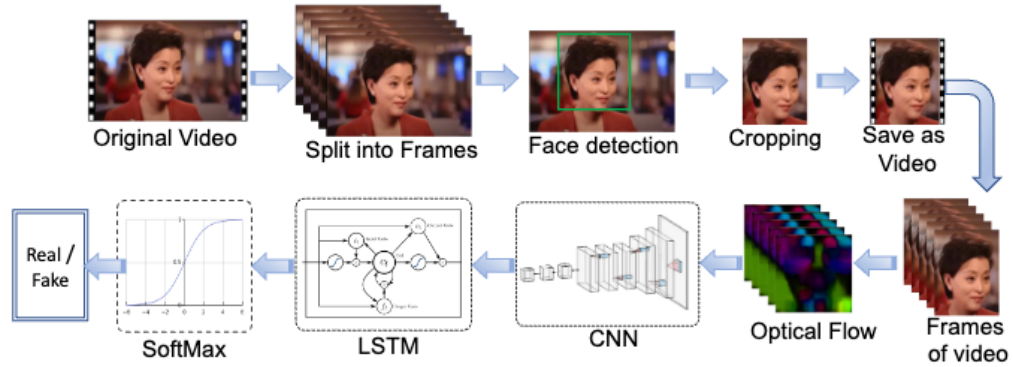


Figure 13

Showcasing a deepfake detection pipeline that utilizes the OpticalFlow values from each frame to train a CNN+LSTM pipeline. (Tiwari et.al 2023)

The second pipeline, as illustrated in Figure 12, follows a similar pre-processing where videos are segmented into individual frames and facial regions are extracted and cropped. This approach additionally utilizes Long Short-Term Memory (LSTM) networks in order to capture temporal dependencies between consecutive frames. In this methodology, the extracted facial features are processed through both spatial CNN architectures and temporal LSTM networks to preserve inter-frame sequential information.

	Model	Comparison to best public-available results				
		Acc.	#Param	Our Model	Acc.	#Param(ratio)
CIFAR-10	NASNet-A	98.0%	85M	EfficientNet-B0	98.1%	4M (21x)
CIFAR-100	NASNet-A	87.5%	85M	EfficientNet-B0	88.1%	4M (21x)
Birdsnap	Inception-v4	81.8%	41M	EfficientNet-B5	82.0%	28M (1.5x)
Stanford Cars	Inception-v4	93.4%	41M	EfficientNet-B3	93.6%	10M (4.1x)
Flowers	Inception-v4	98.5%	41M	EfficientNet-B5	98.5%	28M (1.5x)
FGVC Aircraft	Inception-v4	90.9%	41M	EfficientNet-B3	90.7%	10M (4.1x)
Oxford-IIIT Pets	ResNet-152	94.5%	58M	EfficientNet-B4	94.8%	17M (5.6x)
Food-101	Inception-v4	90.8%	41M	EfficientNet-B4	91.5%	17M (2.4x)

Figure 14

Image of a comparison table showcasing EfficientNet models outperforming other architectures according in a study done by (Tan & Le, 2019).

Convolutional Neural Network Architectures are the framework of most classification-based methods.

Different architectures have been applied for feature extraction and representation learning, including ResNet, XceptionNet, and also EfficientNet, each with its own properties. XceptionNet implements depthwise separable convolutions to extract features in facial areas of video frames. ResNet architectures, especially ResNeXt variants, utilize residual connections to perform spatial feature extraction across single video frames. In some image tasks EfficientNets outperform other structures as it can be seen by the Figure 13.

EfficientNet is a CNN architecture that uses a compound scaling method to scale network depth, width, and resolution with fixed coefficients (attributed to Tan & Le, 2019). Such a compound scaling controls trade-offs between accuracy and efficiency by balancing these three dimensions systematically, rather than scaling them arbitrarily. EfficientNet has outperformed conventional CNN architectures in image classification and transfer learning tasks due to its compound scaling method, in which network depth, width, and resolution are jointly optimized (Tan & Le, 2019). In the context of deepfake detection, this scalable architecture allows more efficient and robust extraction of fine-grained facial features, thereby improving generalization across diverse manipulation techniques. Temporal classification methods extend spatial analysis by relying on recurrence, especially in recurrent LSTM networks, to model the temporal dependencies of frames in a video sequence. CNN and RNN architectures perform spatial feature extraction using CNNs on video frames, and RNNs analyze the sequences formed by these features (Guera & Delp, 2018). These models first extract features from all video frames using a CNN backbone and then feed these sequences of frame-level features into LSTM networks to capture temporal patterns in the video content.

4.2.2 Anomaly Detection

Anomaly detection is a machine paradigm; the characteristics of normal data are learned by such systems, which then declare any input that falls outside

this distribution as an outlier or anomaly. In deepfake detection, these methods have been designed to train solely on the data of genuine faces so that during inference, the manipulated content is regarded as a statistical anomaly, thereby helping ensure the robustness of detection against any unseen generation technique. Some representative approaches are VAE-based methods wherein real faces are treated as the normal class to set detection thresholds based on the reconstruction error (Khalid & Woo, 2020). The intuition behind this is that experimentally, the training dataset of legitimate faces can reconstruct faces with low error, whereas deepfakes with a distribution shift introduced by the synthesis process yield higher reconstruction errors (Mirsky & Lee, 2021).

4.3 Deepfake Detection Challenge (DFDC)

In 2020, the Deepfake Detection Challenge was hosted on Kaggle and became one of the largest collaborative efforts against deepfake-generated synthetic media, supported by Facebook, Microsoft, and Google, with a one-million-dollar prize pool to maximize participation.

The winning solution of Selim Seferbekov used a preprocessing method based on Multitask Cascade CNN (MTCNN), a face-finding and cropping framework that detects bounding boxes and landmarks, allowing the faces to be cropped with an additional margin to capture subtle cues in the surrounding regions. Features were extracted using EfficientNet-B7 after applying heavy data augmentation, including blur, Gaussian noise, geometric transforms, and occlusions, to achieve maximum robustness. Seven models were ensembled in video-level classification, averaging frame predictions for the entire sequence. Although approximately 82% detection accuracy was achieved on the public test set, this performance dropped to 65% on the more challenging private test set of hand-picked unseen manipulations, demonstrating the intricacies of generalization in deepfake detection, whereas the challenge set a benchmark-level dataset and showed that joint industry effort could solve the problem.(Seferbekov, 2020)

4.4 Datasets

In this section, publicly available datasets created with the focus of generating deepfake detection models are explained. The publicly available DFDC, FF++, and CelebDF datasets are the most widely used in this research topic.

4.4.1 DFDC Dataset

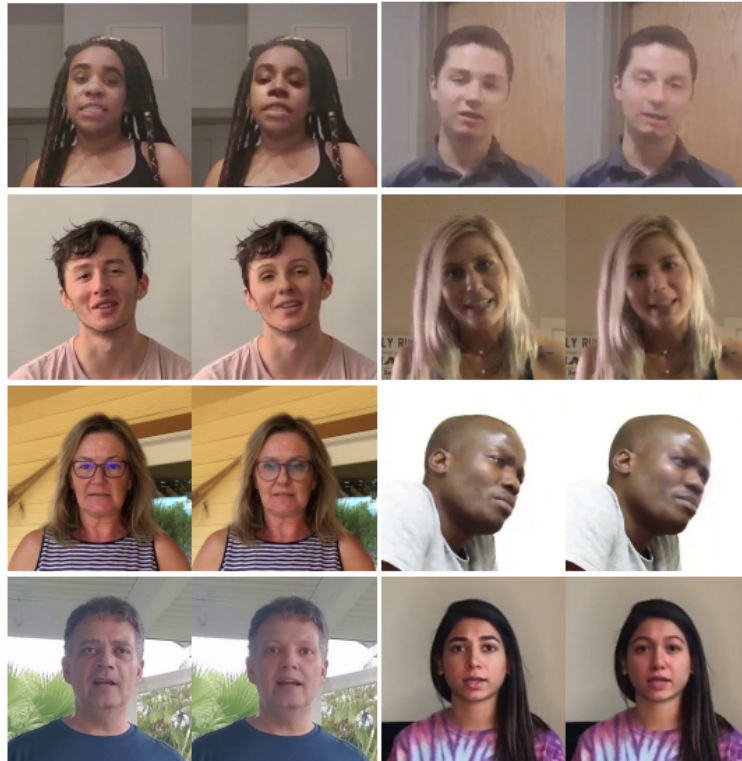


Figure 15

Samples from DFDC Dataset (Dolhansky et al., 2019).

Initiated as a Kaggle competition, the Deepfake Detection Challenge (DFDC) was organized in late 2019 with big industry powerhouses supporting it and with a prize pool of one million USD (Kaggle, 2020). Prior to the competition, Facebook's research team released about a 5,000-video preview set to guarantee that participants at least had some basic data to work with (Dolhansky et al., 2019). The final dataset, published in 2020, was far larger,

comprising of 23,654 real videos and 104,500 manipulated ones with the help of 3,426 paid actors (Dolhansky et al., 2020).

4.4.2 FaceForensics++

Developed by Rössler et al. (2019), the FaceForensics dataset was introduced in 2018 and later expanded in 2019 to become FaceForensics++. Together, 1,371 pristine videos and 8,087 manipulated ones are present in the combined version. Whereas the original primarily generated forgeries through Face2Face, the new update adds four more manipulation types: FaceSwap, FaceShifter, Deepfakes, and NeuralTextures. In parallel with this dataset, the authors maintain the DFDC dataset (Nick & Gully, 2019).

4.4.3 CelebDF

Celeb-DF was created by Li et al. (2020), containing 590 real clips and 5639 synthetically modified videos. The real ones were collected from YouTube, where the goals of gender, age, and ethnicity-based diversity would have been followed. The fake examples were generated using an improved pipeline for deepfake creation and featured deeper architectures and a higher-dimensional space, resulting in improved image quality and resolution.

CHAPTER 5

METHODOLOGY

This chapter describes the methodology of the deepfake detection pipeline and explainability techniques studied in this thesis. The scheme is based on Seferbekov's award-winning baseline, configured into a flexible and parametric framework. Specifically, different kinds of settings, such as the configuration of augmentation, the use of cutout strategies, and alternative types of fill, were implemented for systematic testing under varying conditions. While the pipeline provides frame-level detection, Grad-CAM visualizations were incorporated into its methodology to explain what lies behind the model's decisions. Hence, the matter of detection accuracy and explainability of experimental models in this thesis is addressed.

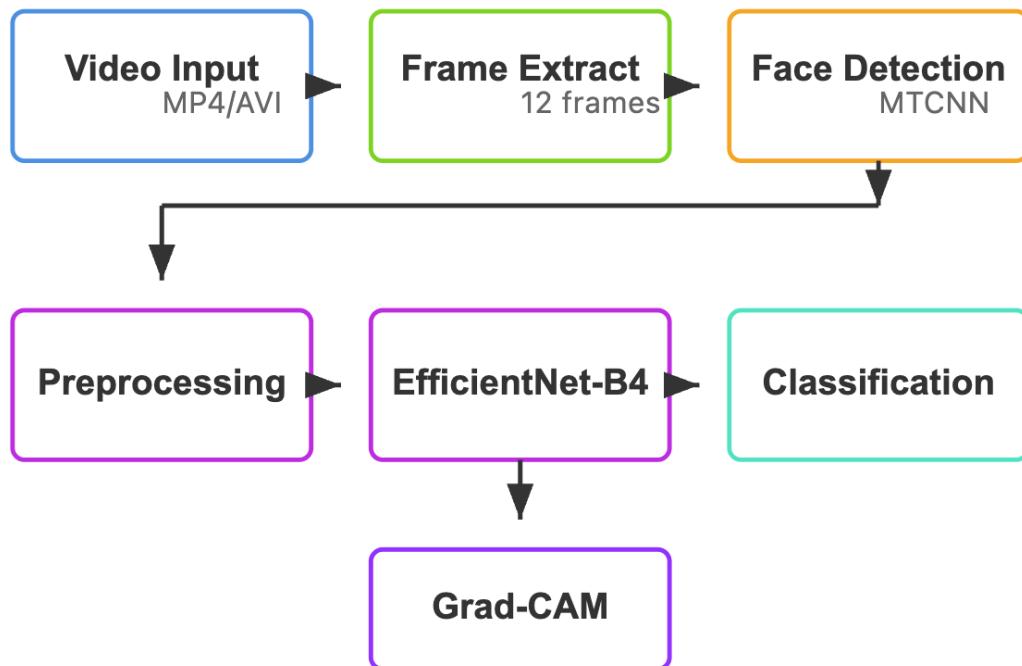


Figure 16

Methodology Pipeline can be seen in this figure.

5.1 Preprocessing and Training Pipeline

The pipeline was implemented as a flexible framework for evaluating diverse experimental settings without altering the core pipeline's concept. The overlay of it can be seen in detail in Figure 15. Since deepfake detection is strongly dependent on proper preprocessing and augmentation, the adopted design is based on Seferbekov's (2020) winning solution from the DFDC competition, which was adapted into a framework for the FaceForensics++ dataset. To save time, frame extraction and face cropping were performed beforehand, while augmentation was done on the fly during training. The network follows a time-distributed architecture, where EfficientNetB4 is deployed at the frame level, and the features are pooled in the classification layer to make judgments on whether the video is genuine or forged as it can be seen in Figure 15.

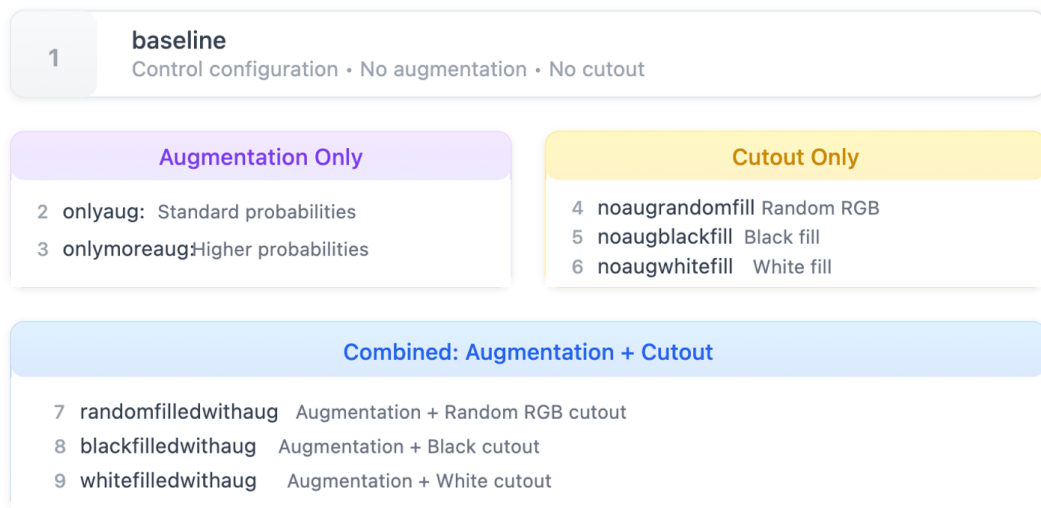


Figure 17

All of the 9 experimental models can be seen here. Each of the meanings of these model names are given next to them.

5.1.1 Preprocessing

In preprocessing, bounding boxes of faces and five-point landmarks were extracted from all frames using the MTCNN detector (Zhang et al., 2016), implemented with a TensorFlow-based library (Ipazc, 2019). Detected faces were aligned according to the eye positions, cropped with a fixed margin, and resized to 224×224 resolution. From each of the videos in our dataset, 32 frames with equal intervals were picked to construct consistent temporal sequences. These frames are used to recreate face-cropped versions of original videos. Additionally, 68 facial landmarks (eyes, nose, mouth) were extracted using the dlib library (King, 2009) to serve as references for subsequent processing steps.

5.1.2 Augmentations and Cutouts

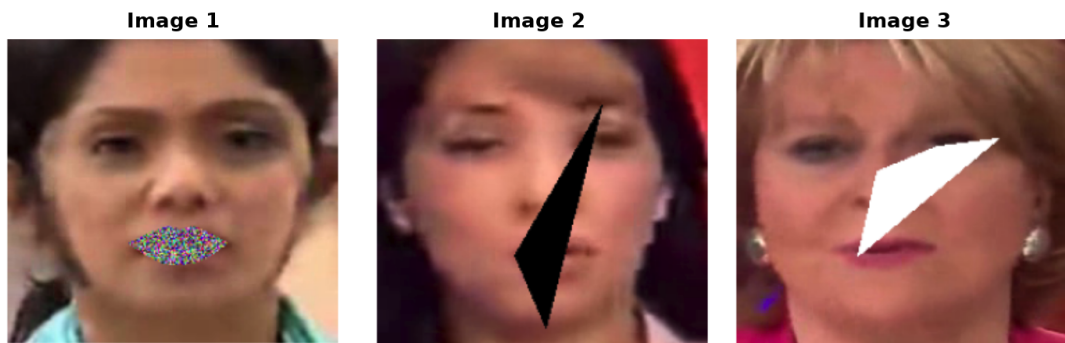


Figure 18

The figure showcases fake frames subjected to augmentation and cutout variants, where Image 1 is generated with random fill, Image 2 with black fill, and Image 3 with white fill.

These cutout differences that can be seen in Figure 18 result in different experimental model bases as it is explained in Seferbekov's(2020) solution. The Albumentations library [Buslaev et al., 2020] was utilized for data augmentation to enhance the model's generalization ability and to facilitate experiments on explainability. Some of the augmentation techniques used include operations such as flipping, scaling, and rotating the image, as well as

operations like noise addition, blurring, and slight adjustments in brightness and color. After augmentation, the frames were rescaled to the range of $[0, 1]$ and normalized with ImageNet's average and standard deviation, as is conventional in training convolutional neural networks.

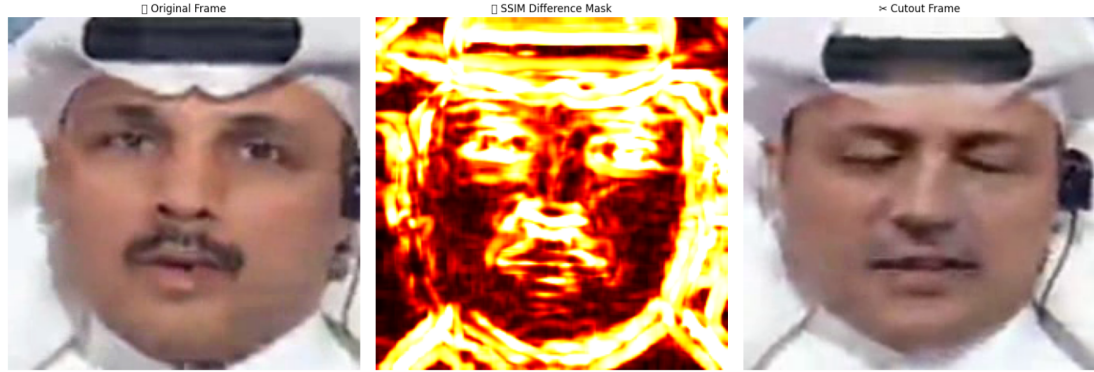


Figure 19

Real Image (on the left) , Fake image (on the right) and the ssim difference map in the middle can be seen.

Finally, SSIM maps [Wang et al., 2004] were computed between real and manipulated frames to locate important regions of high similarity. With these maps, polygon regions around parts of the face were created, and cutout operations were applied using the award winning FaceCrop algorithm of Seferbekov (2020). Differences of this augmentation and cutout techniques are applied in the pipeline to achieve the models in Figure 17.

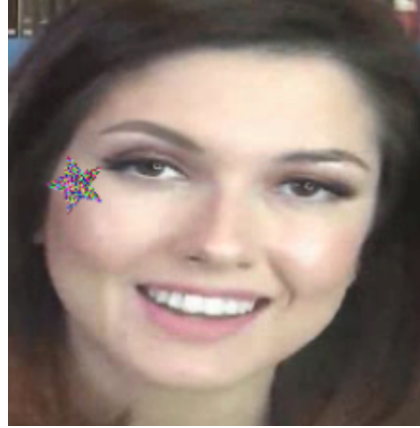


Figure 20

The figure presents a real frame with a star-shaped cutout applied on the cheek – eye region. The interior of the cutout is randomly filled with pixel intensities, illustrating the random-fill variant of the star cutout augmentation.

As the pre-determined polygon for the cutout on real images, its outer radius is between 8 and 16 pixels. Generation algorithm can be examined on Appendix A, and an example can be seen in Figure 19 and example that is effected by star cutout that is only applied to random real images can be seen in Figure 20. The asymmetric cutout strategy applies different perturbations to real and fake samples. FaceCrop polygons target fake images because they simulate detection-resistant artifacts, while star cutouts on real images prevent overfitting to pristine facial features.

5.1.3 Classification

In this research, a sequential model using Tensorflow was designed for the binary classification of deepfakes at the video level, where the decision was based on frame-wise evidence. Face-cropped sequences were passed through a TimeDistributed convolutional backbone (EfficientNet-B4), which extracted spatial features from individual frames independently. These features were subjected to global average pooling, yielding one score per frame through a sigmoid layer that indicated the frame's likelihood of being real or fake. Video-level prediction was obtained by temporally averaging the

frame-wise scores to suppress artifacts and outlier frames, while highlighting consistent cues. In training, the binary cross-entropy was applied to the temporally aggregated prediction to ensure consistency. The backbone was initialized using fine-tuned ImageNet weights, allowing its representational capacity to adapt to manipulation. The 12-frame sequence provides optimal temporal coverage while maintaining computational efficiency. Batch size of 4 balances GPU memory constraints with stable gradient computation.

5.2 Datasets

FaceForensics++ was adopted as the sole dataset because it presents identity-preserving real-fake pairs alongside clearly defined manipulation families for controlled experimentation in artifact localization and explainability. From the roughly 5,000 candidates, a corpus of 1,000 paired samples was created by randomly selecting unique real videos and matching each with respective fake versions generated through one of the four manipulations: FaceSwap, Face2Face, FaceShifter, and DeepFakes.

To balance the manipulation types, an equal number from each strategy was applied, cycling among the four methods, resulting in a balanced set (exactly 250 pairs per manipulation). The final selection consisted of 1,000 unique real videos and 1,000 unique fake videos. The pairs were split into training, validation, and test splits on a pair basis (70/15/15) so that a real-fake pair and identities stayed together in one split.

All analyses were conducted on the face-cropped versions of these videos; the preprocessing details are previously described. Additional datasets (e.g., Celeb-DF, DFDC) were excluded due to their size and computational requirements.

5.3 Evaluation Measures

To evaluate, compare, and test the trained models, various metrics are used.

In this subsection, the reasoning behind these metrics and the calculation of them are explained.

5.3.1 Metrics

Area Under the Curve (*AUC*) is computed from the Receiver Operating Characteristic (*ROC*) curve, which plots the true positive rate (*TPR*) against the false positive rate (*FPR*) at different classification thresholds. The true positive rate is defined as

$$TPR = \frac{TP}{TP + FN}, \quad (5.1)$$

where *TP* denotes true positives and *FN* false negatives. The false positive rate is given by

$$FPR = \frac{FP}{FP + TN}, \quad (5.2)$$

with *FP* denoting false positives and *TN* true negatives. The *ROC AUC* corresponds to the integral under this curve,

$$AUC = \int_0^1 TPR(FPR) dFPR, \quad (5.3)$$

Unlike accuracy *AUC* provides a threshold-independent measure of discrimination between classes. Depending on a decision threshold, *AUC* evaluates the ranking quality of predictions across all thresholds, making it more robust in imbalanced classification tasks such as deepfake detection.

Logarithmic Loss (*LogLoss*) evaluates the quality of probabilistic predictions by penalizing deviations from the true labels. For binary classification, it is defined as

$$LogLoss = -\frac{1}{N} \sum_{i=1}^N [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)], \quad (5.4)$$

where $y_i \in \{0,1\}$ is the true label and p_i the predicted probability of the positive class. Overconfident predictions that are wrong get high penalties, while correct but uncertain predictions are penalized less. This property makes LogLoss suitable for evaluating both classification correctness and calibration of probability estimates at the video level.

The F1-score combines precision and recall into a single harmonic mean. Precision is defined as

$$Precision = \frac{TP}{TP + FP}, \quad (5.5)$$

while recall corresponds to the true positive rate,

$$Recall = \frac{TP}{TP + FN}, \quad (5.6)$$

The F1-score is then computed as

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}, \quad (5.7)$$

F1-score provides a balanced assessment of performance in cases where false positives and false negatives carry different costs. Compared to accuracy, it is more informative in imbalanced classification scenarios, since it

emphasizes both the ability to correctly detect manipulated content and the need to avoid excessive false alarms.

The *Brier* score measures the mean squared error between predicted probabilities and ground-truth labels, capturing both accuracy and calibration of predictions. It is expressed as

$$Brier = \frac{1}{N} \sum_{i=1}^N (p_i - y_i)^2, \quad (5.8)$$

where p_i is the predicted probability and y_i the true label. Unlike metrics that only evaluate discrimination, the *Brier* score accounts for the confidence of predictions and penalizes both overestimation and underestimation of probabilities. This makes it particularly crucial in applications where calibrated probability outputs are critical, such as ensuring reliability in deepfake detection systems.

5.3.2 Diagnostic Curves and Explainability Analysis

Evaluation of models was not just based on scalar metrics but also with the support of diagnostic curves and visualizations. Receiver Operating Characteristic (ROC) curves and Precision–Recall curves were utilized to provide threshold-independent classification performances that offer insights into the TP, FP and TN, FN trade-offs.

Calibration curves (or reliability diagrams) were used to evaluate the degree to which the predicted probabilities matched the observed frequencies, while assessing the trustworthiness of the probabilistic outputs.

Furthermore, histograms of output predictions provided a high-level view of the confidence distributions, thereby helping to reveal overconfidence or underconfidence in the model's predictions. To complement these analyses,

optional threshold-sweeping plots were used, which included F1-score or precision–recall values as a function of decision thresholds, allowing for the determination of optimal operational points for practical use.

In addition to classical evaluations, explainability was studied using Grad-CAM visualizations. Frame-level activation maps were combined into video-level summaries to capture consistent patterns of model attention. Region-based statistics were computed by registering Grad-CAM activations with predefined facial landmarks, thereby providing an average activation score for regions such as the eyes, mouth, nose, and chin.

Activation ratios were studied to quantify relative importance across regions, while temporal consistency was measured by the inverse variance of the activations across frames.

Overlap measures, such as the intersection of activations within SSIM-defined polygons and inspections on thresholded heatmaps and selected regions, were used to describe spatial correspondence with manipulated areas. Additional descriptors included localization of peak activations, inside versus outside face comparisons, class-conditional summarizations for true and false predictions, correlation of activations with model confidence, as well as the sparsity or entropy of the heatmaps. Taken together, these analyses provide a multidisciplinary perspective on both detection performance and interpretability.

CHAPTER 6

EXPERIMENTS

This chapter describes the experiments conducted to investigate the model's performance, regional focus, and explainability. The scientific experiments were conducted using predetermined parameters on an adapted model based on Seferbekov's baseline. To maintain experimental consistency, only the specified parameters were modified while all other variables remained constant.

6.1 Baseline Model

Baseline EfficientNet-B4 (224×224 Input)



Figure 21

Baseline EfficientNet-B4 schema with 12 input frames at 224×224 resolution, processed in a TimeDistributed manner.

Baseline model is an EfficientNet-B4 that is adapted from Seferbekovs FF++ model. Input sequences consisted of 12 RGB frames at 224×224 resolution can be seen in Figure 21, normalized using ImageNet mean and standard deviation. A TimeDistributed EfficientNet-B4 backbone initialized with ImageNet weights was employed; frame-level features were reduced by global average pooling and passed to a minimalist classifier comprising dropout ($p = 0.55$) and a single sigmoid-activated unit with L2 kernel regularization (10^{-3}). As an output the model returns a value from 0 to 1 where the value denotes realness

Video-level predictions were obtained by averaging the frame-wise probabilities across time, and the learning objective was defined as binary cross-entropy on this temporally averaged output to match the inference rule. Optimization was carried out with Adam under a cosine learning-rate schedule with gradient clipping (clipvalue = 1.0), while early stopping monitored validation loss with a patience of five epochs and restored the best weights. A batch size of four was used throughout. No data augmentation or cutout operations were applied in this baseline configuration

6.2 Experimental Settings

All hyperparameters and architectural choices were maintained at baseline values to guarantee experimental validity. Only two key factors of variation were considered: whether or not the augmentation is applied, the intensity of label-preserving data augmentations, and the application of fake-only polygonal cutout procedures with different fill processes.

The polygonal cutout procedure entailed an SSIM-guided, landmark-derived selection process to identify the most similar facial regions. When enabled, the FaceCrop cutout procedure was applied only to synthetic frames, using a single polygon per frame. In the case of the cutout procedure, the star cutout is applied onto the real images with a probability of fifty percent (Appendix A and Appendix B). All other experimental parameters, such as the structure of the model, temporal strategy, loss function, optimizers, and evaluation procedure, were kept constant across all experiments.

The experimental setting consisted of eight distinct design configurations, arranged along two main axes of variation. The baseline setting was the control setup, where no augmentations or cutouts were used. Two configurations are selected for augmentation: the first relies on standard geometric, blur, noise, and photometric transformations, whereas the other performs the same transformations but at a higher intensity. In both cases, no cutouts to real and fake images were applied.

Three other configurations were checked to independently verify the effects of the polygonal-cutout procedure. As isolated cutout variants, they applied the SSIM-guided polygonal selection only to synthetic content, one with a zero-value (black) fill, one with a white fill, and the other with a random-value fill, without any additional augmentation other than cutouts.

Three further configurations combine the two experimental factors. The integrated approaches used standard augmentations, along with fake-only polygonal FaceCrop cutout procedures with stat cutout on real images, while systematically varying the fill methodology across three alternatives: random-value fill, zero-value fill, and white-value fill.

These experiments enabled the systematic isolation and evaluation of each factor's contribution to model performance and interpretability measures, allowing for direct comparison with the other models of this experiment.

All experimental models are trained 3 times to calculate the average results and their standard deviation values.

6.3 Experimental Models

The models that are briefly explained in the Section 6.2 are introduced with detail and their corresponding naming in this section.

6.3.1 Only Augmentation Model

This configuration uses the same architecture and training settings as the base model described in Section 6.1, differing only in the use of data augmentation. Following the methodology of Seferbekov (2020), augmentations are applied to duplicated copies of both real and fake frames during training, thereby doubling the size of the training set. The augmentation operators are listed in Appendix C. This configuration is referred to as “*onlyaugmodel*” throughout the remainder of this thesis. In all of the following models the training size is

doubled, as explained this is caused by the creation of copies in the process of augmentation or cutout application.

6.3.2 Only More Augmentation Model

This configuration is architecturally identical to “*onlyaug*” and retains all hyperparameters from Section 6.1. It differs solely in the per-operator application probabilities of the augmentations.

The baseline probabilities are provided in Appendix C, whereas the modified schedule used here is detailed in Appendix D. All other settings remain unchanged. This model is called “*onlymoreaug*”

6.3.3 Random Filled Cutouts with Augmentation

This configuration, denoted “*randomfilledwithaug*,” employs cutout masks whose locations are selected by the FaceCrop procedure proposed by Seferbekov (2020). The masked facial regions are removed and in-painted with uniformly random RGB values. After the FaceCrop region is determined, the augmentation suite listed in Appendix C is applied to duplicated copies of both real and fake images. All remaining training settings are the same as in Section 6.1. In addition, the “star” cutout variant described and illustrated in Appendices A and B is applied to real images.

6.3.4 Black Filled Cutouts with Augmentation

Named “*blackfilledwithaug*,” this configuration is identical to “*randomfilledwithaug*” except for the fill value used inside the cutout: removed regions are set to zero (black). The “star” cutout is likewise applied to real images.

6.3.5 White Filled Cutouts with Augmentation

This configuration, referred to as “*whitefilledwithaug*,” matches “*randomfilledwithaug*” in every aspect except for the cutout fill, which is set to white. The “star” cutout is again applied to real images, and all other training settings with cutout.

6.3.6 Cutout Only Models

No data augmentation was applied the following models that employ only FaceCrop with the star cutout on real images. The corresponding namings are by their fill types. For random fill the name is “*noaugrandomfill*,” for the model with black fill is “*noaugblackfill*,” and the one with white fill is called “*noaugwhitefill*.”

CHAPTER 7

RESULTS

In this chapter, the results obtained across the nine model configurations, including the baseline and eight experimental variants with various combinations of augmentation and cutout techniques, are presented. The results are divided mainly into two parts: quantitative performance analysis using classical classification metrics (AUC, LogLoss, F1-score, and Brier score) and qualitative interpretability analysis using Grad-CAM visualizations.

The performance section compares all configurations in terms of detection accuracy to identify the best pre-processing strategies. Meanwhile, the explainability analysis examines which areas attract attention during classification, therefore exposing whether models learn to focus on meaningful facial regions or get distracted by artifacts. Together, the findings provide an insight into the efficacy and reliability of various pre-processing approaches for deepfake detection.

7.1 Quantitative Performance Analysis

This subsection presents the performance metrics obtained from all nine model configurations, with results summarized in comparison tables and diagnostic plots to identify the effects of preprocessing strategies .

7.1.1 Area Under the Curve Results

Table 1

Table containing the Test AUC results of different experimental models.

Models	Test AUC
<i>Blackfilledwithaug</i>	0.8971 ± 0.0064
<i>Randomfilledwithaug</i>	0.8837 ± 0.0072
<i>whitefilledwithaug</i>	0.8734 ± 0.0061
<i>noaugrandomfill</i>	0.8711 ± 0.0051
<i>onlymoreaug</i>	0.8718 ± 0.0043
<i>baseline</i>	0.8678 ± 0.0044
<i>noaugblackfill</i>	0.8666 ± 0.0043
<i>noaugwhitefill</i>	0.8639 ± 0.0047
<i>onlyaug</i>	0.8610 ± 0.0051

From the AUC results in Table 7.1, several key findings may be concluded regarding the power of various preprocessing strategies. First, the combined augmentation with the cutout technique gave rise to the most-performing model, wherein the black-filled cutout model (blackfilledwithaug) achieved the highest AUC score, 0.8971 ± 0.0064 -with roughly a 3% increase from the baseline (0.8678 ± 0.0044). Interestingly, the fill strategy of the cutout had a sizeable impact in that the random-filled (0.8837 ± 0.0072) and white-filled (0.8734 ± 0.0061) versions also beat baseline, with diminishing strength.

Augmentation-only strategy performed slightly below the baseline (onlyaug: 0.8610 ± 0.0051), indicating that augmentation by itself, free of any cutout strategy, could be inducing noise into the pipeline and hence lowering detection performance.

The cutout-only models showed mixed results compared to the baseline. While the random-filled cutout (noaugrandomfill: 0.8711 ± 0.0051) achieved a slight improvement over the baseline, both black-filled (0.8666 ± 0.0043) and white-filled (0.8639 ± 0.0047) cutout variants performed worse than the baseline, suggesting that the choice of fill strategy impacts the effectiveness of cutout procedures when applied without augmentation.

Interestingly, this is not in accordance with Seferbekov's original results, wherein random fill went best. Such a disparity may have to do with the dataset scale itself, wherein a much more limited subset of 1,000 video pairs was employed in this study versus the full-scale DFDC dataset of the original work, which could affect what is the optimal regularization signature for cutout operations.

7.1.2 F1Score Results

Table 2

Table containing the Test F1Score results of different experimental models.

Models	Test F1Score
<i>Blackfilledwithaug</i>	0.8429 ± 0.0101
<i>onlyaug</i>	0.7955 ± 0.0084
<i>Randomfilledwithaug</i>	0.7950 ± 0.0098
<i>whitefilledwithaug</i>	0.7951 ± 0.0095
<i>onlymoreaug</i>	0.7932 ± 0.0077
<i>noaugblackfill</i>	0.7912 ± 0.0077
<i>baseline</i>	0.7780 ± 0.0065
<i>noaugrandomfill</i>	0.7769 ± 0.0069
<i>noaugwhitefill</i>	0.7704 ± 0.0074

The F1-score results presented in Table 7.2 reveal some differences from the AUC rankings, highlighting the distinct nature of these performance metrics. While the black-filled cutout with augmentation model (blackfilledwithaug) maintained its superior position with an F1-score of 0.8429 ± 0.0101 , representing an 8.3% improvement over the baseline (0.7780 ± 0.0065), the most striking difference emerged in the performance of the augmentation-only model (onlyaug). This configuration, which ranked lowest in AUC evaluation, achieved the second-highest F1-score (0.7955 ± 0.0084), indicating that while augmentation alone may compromise the model's probability ranking ability, it significantly enhances the precision-recall balance at the optimal threshold.

In the contrary, the random-filled cutout with augmentation (randomfilledwithaug) dropped from second place in AUC to a mid-tier position in F1-score (0.7950 ± 0.0098), suggesting potential issues with threshold-dependent classification performance despite strong ranking capabilities. These divergent patterns between AUC and F1-score metrics underscore the importance of evaluating multiple performance measures, as threshold-independent ranking ability (AUC) and precision-recall optimization (F1) can respond differently to various preprocessing strategies, particularly regarding the trade-offs between probability calibration and decision boundary effectiveness.

7.1.3 BrierLoss Results

Table 3

Table showcasing the brier loss of different experimental models.

Models	Test Brier Score
blackfilledwithaug	0.1242 ± 0.0009
whitefilledwithaug	0.1455 ± 0.0012
onlymorehaug	0.1431 ± 0.0008
randomfilledwithaug	0.1450 ± 0.0011
noaugwhitefill	0.1462 ± 0.0011
baseline	0.1524 ± 0.0013
noaugrandomfill	0.1524 ± 0.0012
noaugblackfill	0.1537 ± 0.0009
onlyaug	0.1572 ± 0.0010

The Brier Score results demonstrate consistent superiority of the black-filled cutout with augmentation model (blackfilledwithaug: 0.1242 ± 0.0009), achieving the best probability calibration performance. Notably, all augmentation-cutout combinations and the enhanced augmentation model significantly outperformed the baseline (0.1524 ± 0.0013), with improvements ranging from 4.5% to 18.5%. The augmentation-only model (onlyaug: 0.1572 ± 0.0010) exhibited the poorest calibration quality, reinforcing the importance of strategic cutout integration for reliable probability estimates in deepfake detection systems. The Brier Score specifically measures how well the model's predicted probabilities align with actual outcomes, with lower values indicating better calibrated and more trustworthy confidence estimates.

7.1.4 LogLoss Results

Table 4

LogLoss values in test of each model showing great differences in positions from the F1Score and AUC.

Models	Test LogLoss
<i>Randomfilledwithaug</i>	0.4656 $\pm$ 0.0067
<i>Blackfilledwithaug</i>	0.4710 $\pm$ 0.0063
<i>whitefilledwithaug</i>	0.4761 $\pm$ 0.0071
<i>baseline</i>	0.4827 $\pm$ 0.0069
<i>noaugblackfill</i>	0.4989 $\pm$ 0.0074
<i>noaugrandomfill</i>	0.5241 $\pm$ 0.0078
<i>noaugwhitefill</i>	0.5244 $\pm$ 0.0076
<i>onlymoreaug</i>	0.5288 $\pm$ 0.0081
<i>onlyaug</i>	0.5719 $\pm$ 0.0084

According to the LogLoss values of Table 7.2, a clear hierarchy of performance was revealed, very much different from the hierarchy according to the AUC, as well as the F1-score. That shows the uniqueness of this Metric in terms of its sensitivity to the quality of calibration of Probability.

The random-filled cutout with augmentation model (*randomfilledwithaug*: 0.4656 $\pm$ 0.0067) had the best LogLoss; hence it can be said to perform better than the black-filled version that ruled the AUC evaluations in the first place. What is more striking about the model using only augmentation, having a LogLoss of 0.5719 $\pm$ 0.0084, is that it came in last: even though it nevertheless achieved competitive F1-scores, these results strongly indicate the presence of glaring issues with probability calibration when augmentation is applied without the concurrent application of cutout procedures.

These divergent patterns across metrics underscore the different facets of model performance each measure addresses. While AUC looks at threshold independent ranking ability and F1-score concentrates on precision-recall balance at the best thresholds, LogLoss, on the other hand, penalizes an overconfident but incorrect prediction and rewards proper calibration of probability estimates.

The logarithmic loss superiority of augmentation-cutout combinations resonates with Seferbekov's initial discoveries-an almost similar preprocessing approach that emphasize the probability calibration aspect in deepfake detection systems. This metric convergence validates the usefulness of the combination of augmentations and cutout for producing probability estimates that are both very accurate and trustworthy enough for deployment in real-world scenarios.

7.2 Qualitative Interpretability Analysis

In this subsection explainability analysis obtained from all nine model configurations using Grad-CAM techniques are presented, with results summarized through attention heatmap comparisons and regional focus patterns to identify how preprocessing strategies influence model decision-making and spatial attention mechanisms.

The following subsections present comprehensive Grad-CAM analyses wherein the spatial attention patterns and regional activation distributions are examined for each model configuration to explain the decision-making mechanisms and identify the most influential facial regions contributing to classification outcomes.

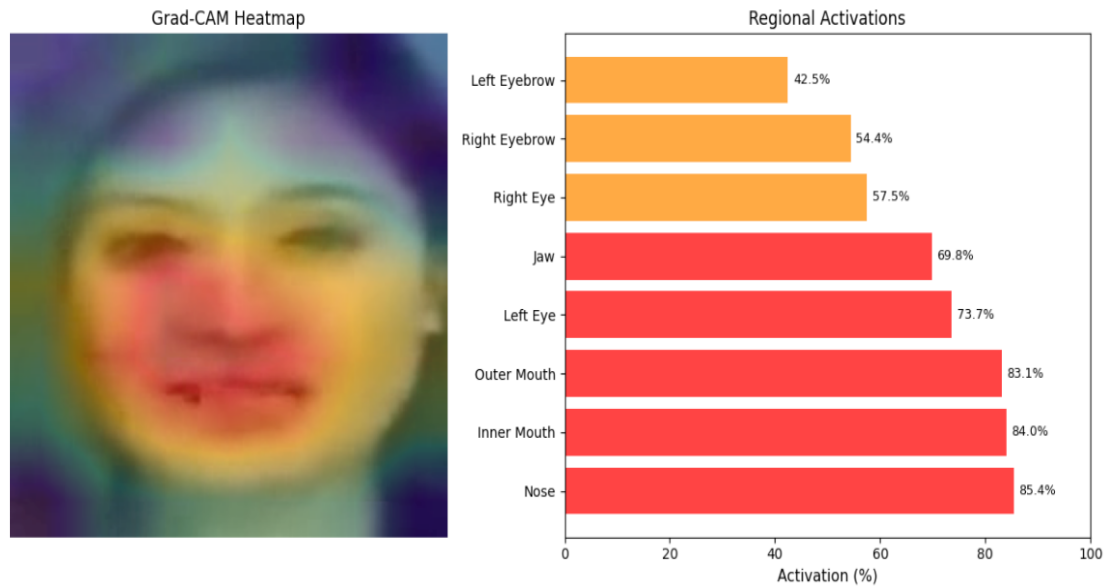


Figure 22

Illustration of Gradcam Heatmap(left) and Regional Activation Chart of this heatmap (right) of a True Negative test set sample can be seen.

Most influential facial region analysis and average regional activation charts for each model are calculated using heatmap the average of each frame in their respected test dataset frames. These heapmaps and calculations are generated seperately as it can be seen from Figure 22 and later averaged for True Positive, True Negative, False Positive and False Negative cases which can be seen on Appendix D for the baseline model.

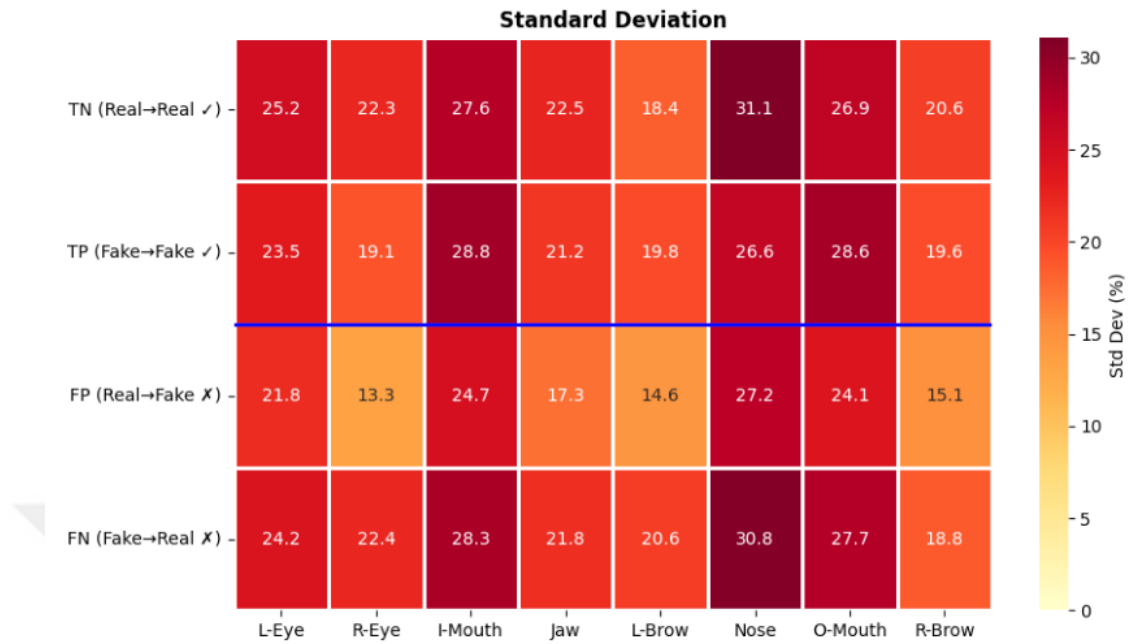


Figure 23

Presents the baseline models standard deviation analysis of GradCAM activation patterns across eight facial regions for different prediction categories

Standard deviation heatmaps were plotted for each of the nine models to quantify the variance in GradCAM activation patterns across the eight different facial regions (left eye, right eye, mouth, jaw, left eyebrow, nose, outer mouth, and right eyebrow) for all permutations of prediction categories (True Positive, True Negative, False Positive, and False Negative).

The analysis that is made from data underlying graph in Figure 23 revealed that models with considerate attention to lower values of standard deviation exhibited uniform attention patterns in specific facial regions, which contributed paradoxically to higher instances of misclassification, especially under False Positive scenarios where genuine content was falsely identified as manipulated. The findings suggested that excessive consistency in focusing attention, instead of flexible interpretation of local regions, may induce overfitting in the model and hurt its ability to generalize. Regions like the nose

were noted for having come up consistently different in all models and in each prediction category, whereas the eyes showed asymmetric attention patterns between well-classified and misclassified samples.

The prominence of the nose region in deepfake detection has been linked to its geometric centrality and lower variability compared to other facial areas. Because the nose acts as a structural anchor, even subtle blending artifacts or shading inconsistencies introduced during synthesis can be reliably exploited by CNN-based detectors (Matern et al., 2019).

Prior analyses of deepfake artifacts similarly emphasized that stable mid-facial regions tend to reveal manipulation traces more consistently than dynamic areas such as the eyes or mouth (Li et al., 2020).

The implication is that, to do well, a model needs a trade-off between consistent feature extraction and attention flexibility because an excessively consistent attention mechanism might stand in the way of the model being able to separate real facial content from manipulated content across various datasets.

7.2.1 True Negative Case Analysis

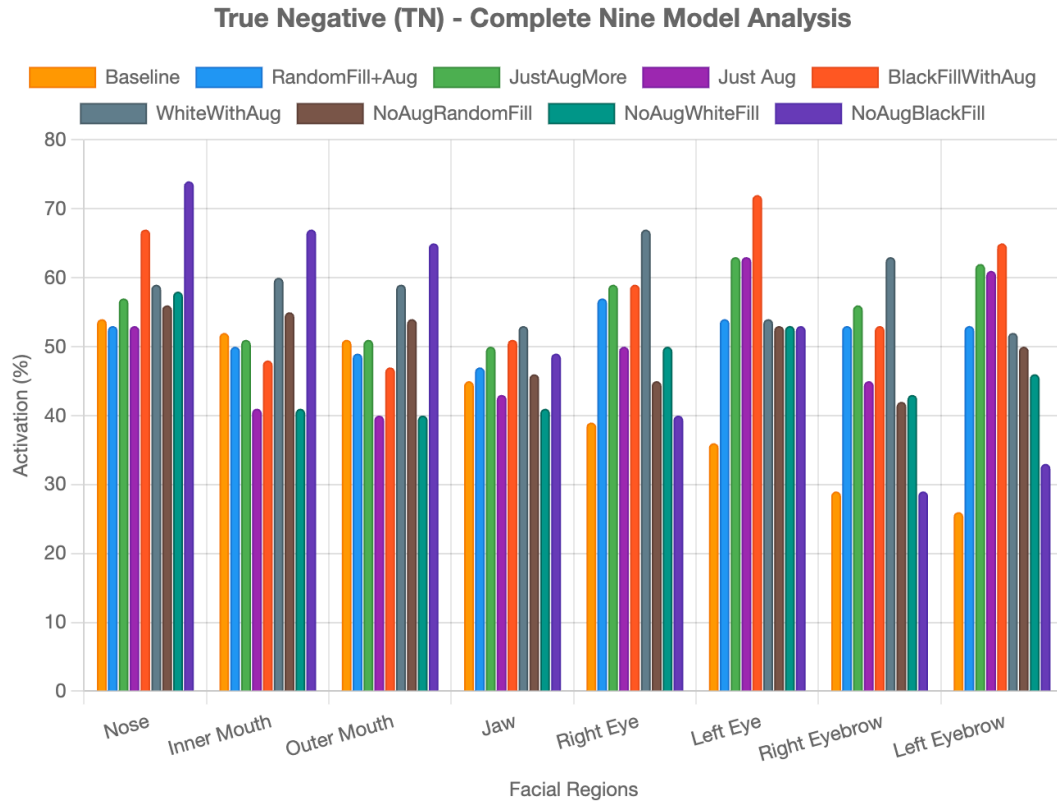


Figure 24

True Negative (TN) activation analysis across facial regions for nine different model configurations. The values represent the average activation ratios (%) in correctly classified real samples, separated into localized facial regions.

Figure 24 presents a comparison of the relative activation strengths observed across nine model variants when authentic samples were correctly classified as true negatives. It was found that all models exhibited non-uniform activation distributions across different facial regions, indicating that the various augmentation and cutout strategies influenced which areas of the face were given emphasis during the recognition of genuine content.

Notably, the *NoAugBlackFill* model was observed to demonstrate consistently higher activations across multiple regions, including the nose, mouth, jaw, and eyes, whereas the baseline model was found to display relatively lower activations, particularly in the eye and eyebrow areas. These findings suggest

that when black-fill perturbations were introduced without augmentation, a stronger reliance on broader facial cues was developed for recognizing authentic samples.

When individual regions were examined more closely, further distinctions between strategies were revealed. For instance, it was observed that in the left eye region, *BlackFillWithAug* produced the highest activation levels (exceeding 70%), while the baseline model showed considerably weaker performance (approximately 35%). This highlighted the sensitivity of eye regions to the design of augmentation and cutout techniques. Similarly, eyebrow activations were found to remain generally low in the baseline model but showed notable improvement under augmentation-heavy configurations, suggesting that these subtle regions were better utilized when perturbation-driven training was employed. Collectively, these results indicate that models trained with targeted fill or augmentation strategies not only demonstrate improved general robustness but also develop more diversified attention patterns across facial landmarks, while the baseline model was observed to remain comparatively less adaptive in its attentional focus.

7.2.2 True Positive Case Analysis

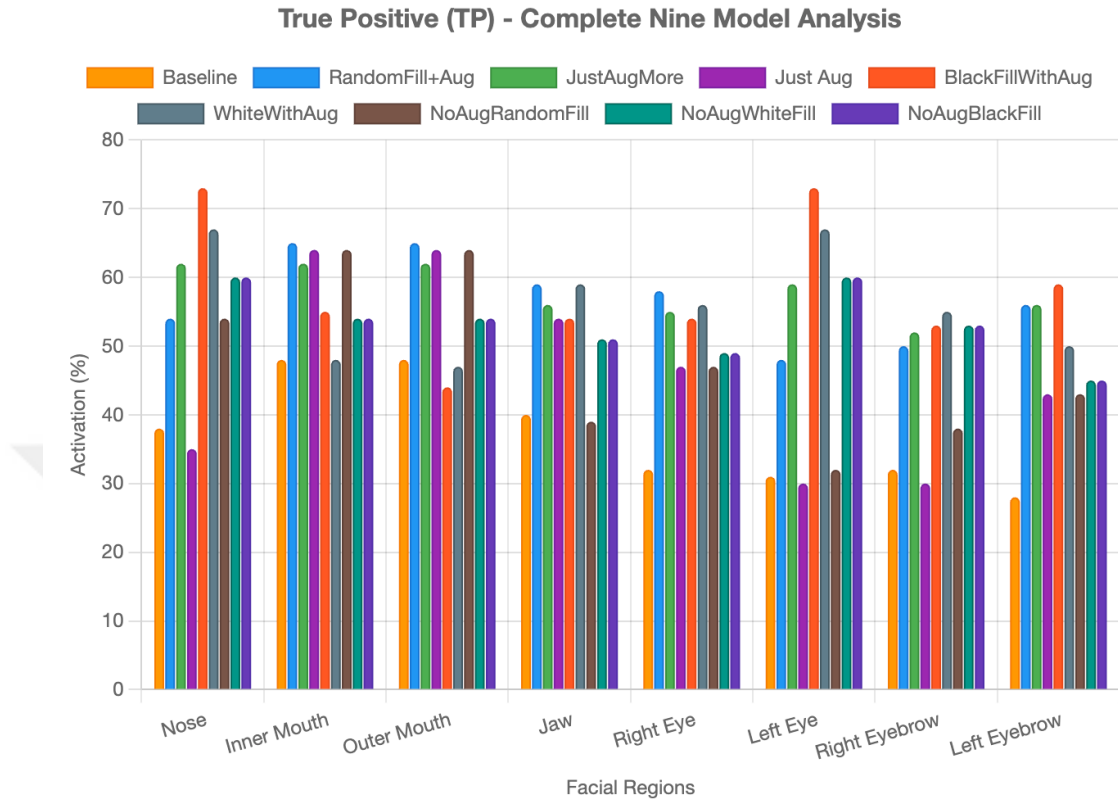


Figure 25

True Positive (TP) activation analysis across facial regions for nine model configurations.

When compared to the true negative case, a notable shift in the spatial distribution of activations was observed in the true positive results. This activation values on TP cases can be further examined in Figure 25. While *NoAugBlackFill* had been found to dominate in the nose region during TN analysis, the strongest activations under TP conditions were observed for *BlackFillWithAug*, with both the nose and left eye regions reaching approximately 73%. Similarly, *WhiteWithAug* was observed to shift from its peak performance in the right eye under TN conditions (67%) to a left eye emphasis in TP at a comparable magnitude, suggesting that the model's visual reliance was adapted according to the classification context.

The baseline model, in contrast, was found to remain weak across all regions in TP detection (ranging from 28% to 48%), indicating limited adaptability in localizing discriminative cues when manipulated content was being detected.

Additionally, the mouth regions were identified as critical indicators in TP detection. Configurations such as *JustAugMore*, *RandomFill+Aug*, and *Just Aug* were observed to display consistently elevated activations in both inner and outer mouth regions, reflecting the fact that mouth dynamics often contain strong manipulation artifacts.

When compared to TN conditions, where eye regions were found to be more balanced across methods, TP analysis revealed a redistribution of attention towards the mouth and selective eye regions, depending on the augmentation strategy employed. This shift suggests that augmentation and cutout policies not only determine the magnitude of activation but also govern the allocation of model attention between facial landmarks in authentic versus manipulated classification scenarios.

7.2.3 False Negative Case Analysis

False Negative (FN) Failed to detect fake content								
	Nose	I.Mouth	O.Mouth	Jaw	R.Eye	L.Eye	R.Brow	L.Brow
Baseline	49%	49%	47%	43%	39%	34%	34%	27%
RandomFill+Aug	61%	60%	59%	58%	58%	58%	56%	55%
JustAugMore	62%	59%	59%	58%	58%	58%	54%	53%
JustAug	62%	59%	59%	58%	58%	58%	54%	53%
BlackFill+Aug	78%	75%	71%	67%	61%	58%	57%	56%
White+Aug	69%	66%	62%	62%	60%	59%	55%	54%
NoAugRandFill	52%	52%	51%	44%	43%	42%	39%	38%
NoAugWhite	70%	63%	61%	57%	55%	55%	53%	51%
NoAugBlack	70%	63%	61%	57%	55%	55%	53%	51%

Figure 26

False Negative (FN) activation ratios across facial regions for nine model configurations.

The results that can be seen from the Figure 26, revealed that the nose region was identified as the most problematic area for missed detections, with FN rates observed to range from 27% in the baseline model to as high as 78% in *BlackFillWithAug*. Similarly, *NoAugWhite* and *NoAugBlack* were found to show high FN ratios in the nose region (70%), while *WhiteWithAug* was observed to follow closely at 69%. In contrast, eyebrow regions were found to show comparatively lower FN values, suggesting that manipulations in these areas were less frequently overlooked. Among the model configurations examined, the baseline and *NoAugRandFill* were found to yield the lowest FN levels overall (27–49% and 38–52%, respectively), indicating that stronger reliability

in detecting manipulated content was achieved. Mouth regions were observed to consistently occupy a middle ground, with FN ratios typically found to be between 47% and 75%. When taken together, these findings suggest that augmentation strategies emphasizing black or white fill operations tend to increase the likelihood that manipulations in central facial regions, particularly the nose, will be missed, while simpler configurations were found to maintain better balance in sensitivity.

7.2.4 False Positive Case Analysis

 False Positive (FP) Wrongly flagged real as fake								
	Nose	I.Mouth	O.Mouth	Jaw	R.Eye	L.Eye	R.Brow	L.Brow
Baseline	53%	52%	51%	46%	39%	37%	36%	32%
RandomFill+Aug	66%	65%	65%	64%	64%	64%	60%	59%
JustAugMore	65%	63%	62%	62%	62%	60%	59%	59%
JustAug	67%	66%	56%	53%	48%	45%	44%	44%
BlackFill+Aug	69%	69%	58%	55%	51%	50%	50%	48%
White+Aug	65%	61%	61%	60%	58%	57%	55%	55%
NoAugRandFill	53%	53%	49%	44%	42%	40%	35%	32%
NoAugWhite	70%	68%	74%	73%	73%	61%	61%	60%
NoAugBlack	77%	76%	74%	73%	73%	61%	61%	60%

Figure 27

False Negative (FN) Activation ratios across facial regions for nine model configurations

The analysis from Figure 27 demonstrates that false alarms (False Positives) occurred most frequently in the *NoAugBlack* configuration, where nose region performance reached 77% and consistently exceeded 70% across multiple facial areas. A similar pattern emerged with *NoAugWhite*, which produced values ranging from 70% to 74% in several regions. Models utilizing augmentation strategies, particularly *JustAug*, also generated elevated FP ratios, with nose region performance reaching 67%. In contrast, both baseline and *NoAugRandFill* models maintained substantially lower FP levels (32–53% across all regions), demonstrating more conservative classification behavior when evaluating authentic content.

Comparison between FP and FN outcomes reveals a distinct trade-off relationship. *BlackFillWithAug* performed poorly in detecting manipulated content (FN 78%), whereas *NoAugBlack* demonstrated excessive sensitivity in classifying genuine faces as manipulated (FP 77%). Regional error patterns remained consistent across both error categories, with the nose area representing the most problematic region for both FN and FP classifications. Conservative approaches, including baseline and *NoAugRandFill* models, achieved balanced detection capabilities by maintaining reduced error rates across both categories. Conversely, intensive augmentation or fill strategies compromised this balance, either elevating missed detection rates (FN) or increasing false alarm frequencies (FP). These findings indicate that effective robustness depends on establishing an optimal equilibrium between augmentation intensity and false alarm tolerance.

7.2.5 Standard Deviation of Activation Analysis

In order to examine the standard deviation/face regions graphs across the 9 models, distinct consistency differences emerge between classification cases. True negative (TN) predictions demonstrate the most stable performance in well-performing models, whereas true positive (TP) predictions exhibit the highest variability in poorly performing models.

False positive (FP) systematic patterns show significant elimination in the best models, whereas false negative (FN) predictions maintain moderate consistency across most models.

In the regional analysis, jaw, left brow, and outer mouth regions consistently emerge as the most reliable areas across all models. In contrast, the nose, inner mouth, and left eye regions are identified as the most inconsistent areas. Data augmentation techniques play a critical role in model consistency. Models that use augmentation achieve consistency within the 11-27 range, whereas models without augmentation demonstrate more inconsistency, with a 25-37 range. This 40-60% improvement clearly establishes the fundamental importance of augmentation for model reliability.

When compared with the baseline model, significant improvements are observed in systematic error patterns. The systematic error previously observed in the right eye region during false-positive cases has been successfully eliminated in the improved models. This advancement in regional attention consistency holds critical importance for model interpretability and represents a substantial step forward in understanding how these models process facial features.

CHAPTER 8

CONCLUSION AND DISCUSSION

8.1 Future Work

This research opens several important directions for future investigation. First, the current study used only 1,000 video pairs, which may explain why the black-filled cutout performed better than random fill in our experiments, whereas Seferbekov's original work found random fill to be superior on larger datasets. Testing these methods on larger datasets, such as the full DFDC collection or other datasets like CelebDF, would help determine which findings apply more broadly. Also, cross-dataset examinations are a future research opportunity.

The current approach focused only on individual video frames, but deepfakes often have artifacts that appear across multiple frames over time. Future work should explore methods that analyze how faces change from frame to frame, utilizing techniques such as LSTM networks or attention mechanisms that can identify temporal inconsistencies that single-frame analysis may miss.

While this study focused on non-GAN deepfake methods, research opportunities exist for GAN-based deepfake detection. Future investigations should examine how the proposed augmentation and cutout strategies perform against GAN-generated content, particularly StyleGAN variants and emerging diffusion models. Cross-generational evaluation testing models trained on non-GAN-generated deepfakes against GAN-generated content would reveal important generalization patterns.

Grad-CAM analysis revealed that the nose region causes problems for detection systems. This suggests that different facial regions may require different processing approaches.

Future research could develop region-specific methods or use more advanced explanation techniques to better understand model decisions.

Finally, practical deployment considerations include real-time processing, adversarial robustness testing, and ensemble methods that combine multiple approaches for reliable and explainable deepfake detection in forensic applications.

8.2 Conclusion

The results of this research confirmed that preprocessing has a significant impact on deep learning systems based on CNNs for deepfake detection, affecting both performance and interpretability. The study conducted a systematic assessment of nine configurations and revealed that augmentation alone is detrimental to detection performance, while strategic combinations applied with cutout methods, particularly black-filled cutouts, were found to be most influential in improving accuracy and explainability.

This comprehensive analysis, utilizing multiple metrics, exposed crucial trade-offs in model behavior, with black-filled cutout performing the best in the majority of metrics. Grad-CAM visualizations provided significant insights into the decision-making process of models, revealing increased consistency in attention while also highlighting the nose region as a notable weakness in both incorrect positive and incorrect negative classifications.

These findings underscore the importance of carefully balancing preprocessing techniques and evaluation procedures when developing robust and interpretable deepfake detection methods. It lays important foundations toward building reliable tools suitable in forensic and legal settings, where explainable AI becomes necessary for credibility and admissibility.

REFERENCES

- Abir, W. H., Uddin, M. F., Khanam, F., & Tazin, T. (2023). Detecting deepfake images using deep learning techniques and explainable AI methods. *Intelligent Automation & Soft Computing*, 35(2), 2151–2169.
- Aghasanli, A., Kangin, D., & Angelov, P. (2023). Interpretable-through-prototypes deepfake detection for diffusion models. In *2023 IEEE/CVF International Conference on Computer Vision Workshops* (pp. 467–474).
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1), 53.
- Baliah, S., Lin, Q., Liao, S., Liang, X., & Khan, M. (2025). DiffFace: Diffusion-based face swapping with facial guidance. *arXiv preprint arXiv:2505*.
- BBC News. (2022, March 16). Ukraine war: Zelensky deepfake video used to spread surrender claim. *BBC*.
- Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. *Proceedings of COMPSTAT'2010* (pp. 177–186). Springer.
- Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., & Kalinin, A. A. (2020). Albumentations: Fast and flexible image augmentations. *Information*, 11(2), 125.
- Chesney, R., & Citron, D. K. (2019). Deep fakes: A looming challenge for privacy, democracy, and national security. *California Law Review*, 107(6), 1753–1819.
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1251–1258.
- Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4), 303–314.
- Dolhansky, B., Howes, R., Pflaum, B., Baram, N., & Ferrer, C. C. (2019). The Deepfake Detection Challenge (DFDC) Preview Dataset. *arXiv*. <https://arxiv.org/abs/1910.08854>
- Dolhansky, B., Howes, R., Pflaum, B., Baram, N., & Ferrer, C. C. (2020). The DeepFake Detection Challenge (DFDC) dataset. *arXiv preprint arXiv:2006.07397*.

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
- Gupta, G., Raja, K., Gupta, M., Jan, T., Whiteside, S. T., & Prasad, M. (2024). A comprehensive review of deepfake detection using advanced machine learning and fusion methods. *Electronics*, 13(1), 95.
- Heidari, A., Jafari Navimipour, N., Dag, H., & Unal, M. (2024). Deepfake detection using deep learning methods: A systematic and comprehensive review. *WIREs Data Mining and Knowledge Discovery*, 14(2), e1520.
- Hern, A. (2019, June 11). Deepfake of Facebook's Zuckerberg posted on Instagram. *The Guardian*.
- Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2), 251–257.
- Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7132–7141.
- Iman, M., Arabnia, H. R., & Rasheed, K. (2023). A review of deep transfer learning and recent advancements. *Technologies*, 11(2), 40.
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In F. Bach & D. Blei (Eds.), *Proceedings of the 32nd International Conference on Machine Learning* (Vol. 37, pp. 448–456). PMLR.
- Ipazc. (2019). MTCNN – Multi-task cascaded convolutional networks for face detection (TensorFlow implementation) [Computer software]. *GitHub*. <https://github.com/ipazc/mtcnn>
- Jahoda, P., & Cech, J. (2025). Pose and facial expression transfer by using StyleGAN. *arXiv preprint arXiv:2504.13021*.
- Jiang, L., Guo, Z., Wu, W., Liu, Z., Liu, Z., Loy, C. C., Yang, S., Xiong, Y., Xia, W., Chen, B., Zhuang, P., Li, S., Chen, S., Yao, T., Ding, S., Li, J., Huang, F., Cao, L., Ji, R., Lu, C., & Tan, G. (2021). DeeperForensics challenge 2020 on real-world face forgery detection: Methods and results. *arXiv preprint arXiv:2102.09471*.
- Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2019). A style-based generator architecture for generative adversarial networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4401–4410. <https://doi.org/10.1109/CVPR.2019.00453>
- Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., & Aila, T. (2021). Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 34, 852–863.

- Karras, T., Laine, S., & Aila, T. (2020). Analyzing and improving the image quality of StyleGAN. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8110–8119.
- Kaur, A., Hoshyar, A. N., Saikrishna, V., et al. (2024). Deepfake video detection: Challenges and opportunities. *Artificial Intelligence Review*, 57, 159.
- Khan, S. A., & Dang-Nguyen, D. T. (2023). Deepfake detection: A comparative analysis. *arXiv*.
- King, D. E. (2009). Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10, 1755–1758.
- Kreps, S. E., McCain, R. M., & Brundage, M. (2020). All the news that's fit to fabricate: AI-generated text as a tool of media misinformation. *Journal of Experimental Political Science*, 7(3), 1–14.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Li, L., Bao, J., Yang, H., Chen, D., & Wen, F. (2019). FaceShifter: Towards high fidelity and occlusion aware face swapping. *arXiv preprint arXiv:1912.13457*.
- Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F., & Guo, B. (2020). Face x-ray for more general face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 5001–5010).
- Li, Y., & Lyu, S. (2019). Exposing DeepFake videos by detecting face warping artifacts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 1–7). IEEE.
- Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. (2020). Celeb-DF: A large-scale challenging dataset for DeepFake forensics. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3207–3216. <https://doi.org/10.1109/CVPR42600.2020.00328>
- Mansoor, N., & Iliev, A. I. (2025). Explainable AI for DeepFake Detection. *Applied Sciences*, 15(2), 725.
- Maras, M.-H., & Alexandrou, A. (2021). Determining authenticity of video evidence in the age of artificial intelligence and in the wake of deepfake videos. *International Journal of Evidence & Proof*, 25(3), 255–275.

- Matern, F., Riess, C., & Stamminger, M. (2019). Exploiting visual artifacts to expose deepfakes and face manipulations. *IEEE Winter Applications of Computer Vision Workshops (WACVW)*, 83–92.
- Mehlig, B. (2021). Machine learning with neural networks. *arXiv:901.05639*.
- Mirsky, Y., & Lee, W. (2020). The creation and detection of deepfakes: A survey. *ACM Computing Surveys*, 53(5), 1–41.
- Neves, J. C., Tolosana, R., Vera-Rodriguez, R., Lopes, V., Proença, H., & Fierrez, J. (2020). GANprintR: Improved fakes and evaluation of the state of the art in face manipulation detection. *IEEE Journal of Selected Topics in Signal Processing*, 14(5), 1038–1048.
- Nick, D., & Gully, A. (2019). Contributing data to deepfake detection research. *Google AI Blog*. <https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html>
- Nightingale, S. J., & Farid, H. (2022). AI-synthesized faces are indistinguishable from real faces and more trustworthy. *Proceedings of the National Academy of Sciences*, 119(8), e2120481119.
- Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), 971–987.
- Paris, B., & Donovan, J. (2019). Deepfakes and cheap fakes: The manipulation of audio and visual evidence. *Data & Society Research Institute*.
- Pawelec, M. (2025). Values guiding professional deepfake development. *AI and Ethics*.
- Perov, I., Gao, D., Chervoniy, N., Liu, K., Marangonda, S., Umé, C., Facenheim, D., Jiang, Y., Zhang, H., Wu, W., Zhou, K., & Zhang, W. (2020). DeepFaceLab: Integrated, flexible and extensible face-swapping framework. *arXiv preprint arXiv:2005.05535*.
- Qian, Y., Yin, G., Sheng, L., Chen, Z., & Shao, J. (2020). Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *European Conference on Computer Vision* (pp. 86–103). Springer.
- Reynolds, A. H. (2019). Convolutional neural networks (CNNs).
- Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019, October). FaceForensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. *arXiv preprint arXiv:1901.08971*.

- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4510–4520.
- Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science*, 2(3), 160.
- Seferbekov, S. (2020). Towards solving the deepfake problem. *arXiv*.
<https://doi.org/10.48550/arXiv.2001.06578>
- Shiohara, K., & Yamasaki, T. (2022). Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 18720–18729).
- Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*.
- Song, L., Wu, X., Qian, Y., He, X., & Lyu, S. (2024). Facial feature drift detection for deepfake video analysis. *IEEE Transactions on Information Forensics and Security*, 19, 2847–2862.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9.
- Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning (ICML)*, 6105–6114.
- Tero Karras. (2019). This Person Does Not Exist [Website].
<https://thispersondoesnotexist.com>
- Thies, J., Zollhöfer, M., Stamminger, M., Theobalt, C., & Nießner, M. (2016). Face2Face: Real-time face capture and reenactment of RGB videos. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2387–2395.
- Tian, M., Khayatkhoei, M., Mathai, J., & AbdAlmageed, W. (2023, November 28). Unsupervised multimodal deepfake detection using intra- and cross-modal inconsistencies. *arXiv*.
- Tiwari, A., Dave, R., & Vanamala, M. (2023). Leveraging deep learning approaches for deepfake detection: A review. In *Proceedings of the 2023 7th International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence (ISMSI '23)* (pp. 12–19). Association for Computing Machinery.
<https://doi.org/10.1145/3596947.3596959>

- Tiwari, S. (2020). Towards solving the deepfake problem. *arXiv*.
<https://doi.org/10.48550/arXiv.2001.06578>
- Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). DeepFakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64, 131–148.
- Trinh, L., Tsang, M., Rambhatla, S., & Liu, Y. (2020/2021). Interpretable and trustworthy deepfake detection via dynamic prototypes. *WACV/ArXiv*.
- Tsigos, K., Cornelis, B., Raptopoulos, A., Efthymiou, I., & Mezaris, V. (2024). Towards quantitative evaluation of explainable AI methods for deepfake detection. In *Proceedings of the 3rd ACM International Workshop on Multimedia AI against Disinformation* (pp. 30–38). ACM.
- Vaccari, C., & Chadwick, A. (2020). Deepfakes and disinformation: Exploring the impact of synthetic political video on deception, uncertainty, and trust in news. *Social Media + Society*, 6(1), 2056305120903408.
- Wang, R., Xu, S., He, T., Chen, Y., Zhu, W., Song, D., Chen, N., Tang, X., & Hu, Y. (2025). DynamicFace: High-quality and consistent video face swapping using composable 3D facial priors. *arXiv preprint arXiv:2501.08553*.
- Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612.
- Westerlund, M. (2019). The emergence of deepfake technology: A review. *Technology Innovation Management Review*, 9(11), 39–52.
- Wu, H., Zhou, J., Li, J., Dong, W., & Shi, G. (2022). Anchor-mesh motion based temporal inconsistency detection for deepfake videos. *Computer Vision and Image Understanding*, 220, 103–116.
- Wu, H., Zhou, J., Zhang, S., & Tian, J. (2022). Exploring spatial-temporal features for deepfake detection and localization. *arXiv preprint arXiv:2210.15872*.
- Yoon, S., Park, J., Kim, H., & Lee, S. (2025). Pixel-wise temporal-frequency analysis for deepfake detection in compressed videos. *Pattern Recognition*, 138, 109–124.
- Yu, N., Davis, L. S., & Fritz, M. (2019). Attributing fake images to GANs: Learning and analyzing GAN fingerprints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 7556–7566).
- Zhang, K., Zhang, T., Wu, C., Chen, J., Liu, J., Lin, C., ... Sun, Z. (2025). MorphFace: Face synthesis with 3D morphable model guided diffusion. *arXiv preprint arXiv:2504.00430*.

Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10), 1499–1503.



APPENDICES

Appendix A

```
def _random_star_polygon(h, w, min_r=STAR_MIN, max_r=STAR_MAX,
points=5):
    R = np.random.randint(min_r, max_r)    # outer radius
    r = max(3, int(R * 0.5))              # inner radius
    cx = np.random.randint(R, w - R)
    cy = np.random.randint(R, h - R)
    pts = []
    for i in range(points * 2):
        ang = np.pi * i / points
        rad = R if (i % 2 == 0) else r
        x = int(cx + rad * np.cos(ang))
        y = int(cy + rad * np.sin(ang))
        pts.append((x, y))
    return pts
```

Appendix B

```
GaussNoise(p=0.05),
GaussianBlur(blur_limit=(3, 7), p=0.05),
HorizontalFlip(),
OneOf([
    RandomBrightnessContrast(brightness_limit=0.05,
                             contrast_limit=0.05),
    FancyPCA(alpha=0.05),
    HueSaturationValue(hue_shift_limit=5,                sat_shift_limit=5,
                       val_shift_limit=5)
], p=0.1),
ShiftScaleRotate(shift_limit=0.05, scale_limit=0.1, rotate_limit=5,
                  border_mode=cv2.BORDER_CONSTANT, p=0.1),]
```



Appendix C

```
GaussNoise(p=0.2),
GaussianBlur(blur_limit=(3, 7), p=0.2),
HorizontalFlip(),
OneOf([
    RandomBrightnessContrast(brightness_limit=0.05,
                             contrast_limit=0.05),
    FancyPCA(alpha=0.05), # artık uint8 üzerinde çalışacak
    HueSaturationValue(hue_shift_limit=5,          sat_shift_limit=5,
                       val_shift_limit=5)], p=0.5),
ShiftScaleRotate(shift_limit=0.05, scale_limit=0.1, rotate_limit=5,
border_mode=cv2.BORDER_CONSTANT, p=0.5),]
```



Appendix D

