TRANSFORMER-BASED DEEPFAKE DETECTION METHODS

by

Gökhan Yıldırım

B.S., Electrical and Electronics Engineering, Boğaziçi University, 2019

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2025

ACKNOWLEDGEMENTS

I am deeply grateful to my supervisor Asist. Prof. Berk Gökberk, for his invaluable guidance and support throughout this research.

ABSTRACT

TRANSFORMER-BASED DEEPFAKE DETECTION METHODS

Deepfakes are synthetic media products generated by editing or swapping identities in videos or audio using artificial intelligence. Over the past few years, significant progress in deep learning has led to the development of advanced techniques to create deepfake images and videos. Although deepfake technology has legitimate uses in entertainment and education, it can be hazardous when misused. These tools, readily available to anyone, have the potential to manipulate societies, fuel hatred and hostility, violate privacy, and falsely accuse individuals or organizations of crimes. The inability of human perception to differentiate between real and fake content has spurred research into more sophisticated deepfake detection methods. Vision transformers have been increasingly used by state-of-the-art models for deepfake detection in recent studies. We propose a novel deepfake detection model, the Combinational Cross-Attention Vision Transformer (CCAViT), which integrates convolutional neural networks with Vision Transformers. CCAViT also utilizes the power of the cross-attention mechanism with combinational connections of visual transformers. This thesis aims to set a new benchmark in deepfake detection, offering robust performance against a wide range of manipulation techniques, leveraging the complementary strengths of convolutional neural networks and visual transformers. We developed an improved face detection technique for use in the data preprocessing stage. The challenges related to various stages of the deepfake detection process are explored to ensure more streamlined workflows. This thesis also provides guidance at every step in the implementation of deepfake detection methods, highlighting key focus areas to enhance detection performance.

ÖZET

GÖRÜ DÖNÜŞTÜRÜCÜ TEMELLİ DERİN SAHTE TESPİTİ YÖNTEMLERİ

Derin sahte, video veya seslerdeki kimliklerin yapay zeka ile değiştirilmesiyle üretilen sentetik medya ürünleridir. Son yıllarda derin öğrenmedeki önemli ilerlemeler, derin sahte görüntü ve videolar üretmek için ileri tekniklerin geliştirilmesini sağlamıştır. Derin sahte teknolojisinin eğlence ve eğitim gibi yasal kullanım alanları olsa da, kötüye kullanımı ciddi riskler taşır. Bu araçlar, toplumları manipüle etme, nefret ve düşmanlık körükleme, mahremiyeti ihlal etme ve bireyleri veya kurumları asılsız suçlama potansiyeline sahiptir. İnsanların gerçek ve sahte içerik arasındaki farkı algılayamaması, daha gelişmiş derin sahte tespit yöntemlerine yönelik araştırmaları teşvik etmiştir. Görü dönüştürücüler, son çalışmalarda ileri düzey modeller tarafından derin sahte tespiti için giderek daha fazla kullanılmaktadır. Bu çalışmada, evrişimsel sinir ağlarıyla görü dönüştürücüleri entegre eden Kombinasyonlu Çapraz Dikkat Görü Dönüştürücüleri (CCAViT) adlı yeni özgün bir derin sahte tespit modeli öneriyoruz. CCAViT, çapraz dikkat mekanizmalarıyla görü dönüştürücülerin kombinasyonel bağlantılarından yararlanır. Bu tez, evrişimsel sinir ağları ile görü dönüştürücülerin tamamlayıcı ve güçlü yönlerinden yararlanarak, geniş bir yelpazedeki manipülasyon tekniklerine karşı etkili öncü bir derin sahte tespit yöntemi sunmayı amaçlar. Ayrıca, veri ön işleme aşamasında daha verimli bir yüz tespit stratejisi öneriyoruz. Derin sahte tespitin her aşamasındaki zorluklar ele alınarak daha düzenli bir iş akışı hedeflenmiştir. Yaklaşımımız, tespit sürecinin her adımında uygulamalı rehberlik sağlayarak sonuçları iyileştirmek için odaklanılması gereken anahtar alanları vurgular.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZET v
LIST OF FIGURES viii
LIST OF TABLES x
LIST OF SYMBOLS xii
LIST OF ACRONYMS/ABBREVIATIONS xiii
1. INTRODUCTION 1
2. LITERATURE SURVEY 6
2.1. Deepfake Generation Methods 6
2.2. Deepfake Detection Methods 10
2.2.1. Identity-Based Approaches 14
2.2.2. Vision Transformer-Based Approaches 17
2.2.3. Other Approaches 18
3. PROPOSED METHOD 21
3.1. Combinational Cross-Attenntion Vision Transformer 21
3.2. Datasets 24
3.2.1. FF++ 25
3.2.2. DFD 28
3.2.3. CelebDF 29
3.2.4. DFDC 30
4. EXPERIMENTAL RESULTS 31
4.1. Combinational Cross-Attention ViT Results 33
4.1.1. Ablation Study 33
4.1.2. Implementation and Data Preproccessing Strategy 37
4.1.3. Performance Analysis of CCAViT 39
4.1.4. Hyperparameter Tuning 52
5. CONCLUSION 62

REFERENCES 64

LIST OF FIGURES

Figure 1.1. Example of deepfake generation using face swapping. 1

Figure 3.1. CCViT architecture. 22

Figure 3.2. Combinational Cross-Attention ViT (CCAViT) architecture. . . . 22

Figure 4.1. Dataset frequency in 35 Studies. 32

Figure 4.2. AUC for DFD test videos relative to frame number per video. . . . 39

Figure 4.3. AUC for CelebDF test videos relative to frame number per training video. 39

Figure 4.4. Confusion matrix of deepfake detection models. 42

Figure 4.5. Confusion matrix of CCAViT in DeepFakes. 43

Figure 4.6. Confusion matrix of CCAViT in FaceShifter. 44

Figure 4.7. Confusion matrix of CCAViT in Face2Face. 44

Figure 4.8. Confusion matrix of CCAViT in FaceSwap. 45

Figure 4.9. Confusion matrix of CCAViT in NeuralTextures. 45

Figure 4.10. Confusion matrix of CCAViT in CelebDF. 49

Figure 4.11. Confusion matrix of CCAViT in DFDC. 49

Figure 4.12. Confusion matrix of CCAViT in DFD. 50

LIST OF TABLES

Table 2.1. Performance of the methods trained on FF++ in terms of AUC(%). 16

Table 3.1. Example images from sub-datasets of FF++. 26

Table 3.2. Example images from DFD. 28

Table 3.3. Example images from CelebDF. 29

Table 3.4. Example images from DFDC. 30

Table 4.1. Performance comparison of the methods on DFDC in terms of AUC. 32

Table 4.2. Model size and training time of our implemented models. 35

Table 4.3. Test results of implemented models trained in FF++. 36

Table 4.4. Examples of fake images used in experiments with CCAViT. 40

Table 4.5. Images incorrectly predicted by CCAViT from NeuralTextures. . . 46

Table 4.6. Images incorrectly predicted by CCAViT Face2Face and FaceSwap. 47

Table 4.7. Examples of original images incorrectly predicted by CCAViT from FF++. 48

Table 4.8. AUC of CCAViT across sub-datasets of FF++ versus batch size. . 54

Table 4.9. AUC of CCAViT in DFDC, DFD and CelebDF versus batch size. . 54

Table 4.10. AUC of CCAViT across sub-datasets of FF++ versus dropout rate. 56

Table 4.11. AUC of CCAViT in DFDC, DFD and CelebDF versus dropout rate. 56

Table 4.12. AUC of CCAViT across sub-datasets of FF++ versus dropout rate of cross-attention module. 57

Table 4.13. AUC of CCAViT in DFDC, DFD and CelebDF versus dropout rate of cross-attention module. 57

Table 4.14. AUC of CCAViT across sub-datasets of FF++ versus middle branch patch size. 58

Table 4.15. AUC of CCAViT in DFDC, DFD and CelebDF versus middle branch patch size. 58

Table 4.16. AUC of CCAViT across sub-datasets of FF++ versus learning rate. 59

Table 4.17. AUC of CCAViT in DFDC, DFD and CelebDF versus learning rate. 59

Table 4.18. AUC of CCAViT across sub-datasets of FF++ versus middle branch coefficient 60

Table 4.19. AUC of CCAViT in DFDC, DFD and CelebDF versus middle branch coefficient. 60

LIST OF SYMBOLS

n	Number of branches

LIST OF ACRONYMS/ABBREVIATIONS

ADM	Artifact Detection Module
AI	Artificial Intelligence
AUC	Area Under Curve
AVC	Advanced Video Coding
CCAViT	Combinational Cross Vision Transformer
CCViT	Convolutional Cross Vision Transformer
CDA	Central Difference Attention
CDF	CelebDF
CNN	Convolutional Neural Network
CRF	Constant Rate Factors
CViT	Convolutional Vision Transformer
DCT	Discrete Cosine Transform
DF	DeepFakes
DFD	Deep Fake Detection
DFDC	Deepfake Detection Challenge
DNN	Deep Neural Network
F2F	Face2Face
FF++	Faceforensics++
FFT	Fast Fourier Transform
FN	False Negative
FP	False Positive
FS	FaceSwap
FSh	FaceShifter
GAN	Generative Adversarial Network
GB	Gigabytes
GD	Gradient Descent
GPU	Graphics Processing Unit
HCIL	Hierarchical Contrastive Inconsistency Learning

ICT	Identity Consistency Transformer
ICT-Ref	Identity Consistency Transformer with Reference
IID	Implicit Identity Driven
ISTVT	Interpretable Spatial-Temporal Video Transformer
LSDA	Latent Space Data Augmentation
LSTM	Long Short-Term Memory
MTCNN	Multi-Task Cascaded Convolutional Neural Networks
NT	NeuralTextures
PCCViT	Parallel Convolutional Cross Vision Transformer
PCViT	Parallel Convolutional Vision Transformer
PFGDFD	Preserving Fairness in Deepfake Detection
QAD-E	Quality-Agnostic Deepfake Detection EfficientNet
RNN	Recurrent Neural Network
SFDG	Spatial-Frequency Dynamic Graph
SIFT	Scale Invariant Feature Transform
SSTNet	Sliced Spatio-Temporal Network
SVM	Support Vector Machine
StyleDFD	Style Deepfake Detection
TALL	Temporal Aware Layered Learning
TN	True Negative
TP	True Positive
UCF	Uncovering Common Features for Generalizable Deepfake Detection
ViT	Vision Transformer

1. INTRODUCTION

The rise of computational power and the abundance of digital information available online have contributed significantly to the evolution of new advanced artificial intelligence technologies. In recent times, innovative approaches in deep learning have made it possible to create deepfake images and videos with remarkable precision. This process involves removing a face from one image and replacing it with a face from another while carefully replicating the original expressions, gestures, and movements. In Figure 1.1, two faces in the images are swapped using face swapping methods. To increase realism, the target individual's voice can also be synthesized to match the source audio. Since videos are composed of image frames and audio, stacking and blending frames through deepfake generation techniques can produce hyper-realistic media outputs.

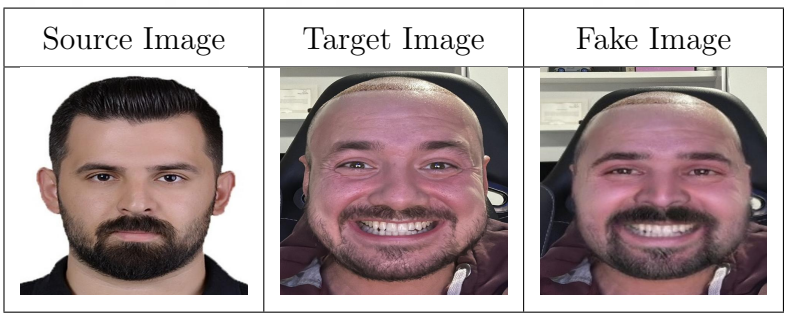


Figure 1.1. Example of deepfake generation using face swapping.

The difficulty humans face in discerning authentic content from manipulated media has led to extensive studies on detecting deepfake imagery. Although this technology serves legitimate purposes in areas such as education and entertainment, its misuse has serious consequences. Easily accessible tools can distort reality, provoke social discord, breach privacy, and falsely accuse people or organizations of wrongdoing. Therefore, improving methods for identifying deepfake content is essential. The spread of these deepfakes can also be curbed before reaching a wider audience.

If proper legal frameworks and preventative measures effectively minimize the misuse of deepfake technology, its beneficial applications could be utilized more freely, publicly, and even at no cost in certain fields. For instance, in filmmaking, deepfake technology enables the recreation of historical figures with stunning realism, making it appear as though the individuals themselves are performing. Similarly, in the gaming industry and digital art, it streamlines the development of diverse virtual characters and visual assets, significantly reducing costs and increasing production speed.

Deepfake detection models demonstrate outstanding performance when tested on familiar datasets containing deepfake images and videos. However, to ensure that these models are reliable and versatile, they must also exhibit strong predictive accuracy when confronted with previously unseen data. Key metrics for evaluating the effectiveness of such models include their performance in both intra-dataset and cross-dataset scenarios, their robustness against inputs with varying image resolutions, and the complexity of the model as measured by the number of parameters it utilizes.

A Reddit user with the alias "deepfakes" shared videos in 2017 that utilized face-swapping technology to superimpose celebrities' faces onto explicit content. This event marked the introduction and popularization of the term "deepfake". Shortly after, in 2018, FakeApp [1], one of the earliest and most widely used tools for generating deepfake videos, became available as open-source software on GitHub. This application leverages Generative Adversarial Networks (GANs) to enable users to exchange faces within video footage. The combination of Reddit's deepfake-related content and the accessibility of tools like FakeApp has intensified concerns surrounding privacy and security in the context of deepfakes.

Artificial intelligence is increasingly playing an important role in professions such as software engineering, retail operations, and healthcare analytics. Although this advancement improves the efficiency of existing industries, it also poses the risk of unemployment for many.

As with every major technological breakthrough, humanity is expected to adapt by turning this transformation into an opportunity, paving the way for new professions and reducing its potential drawbacks. For instance, software engineers who are proficient in Artificial Intelligence (AI) are now in higher demand, and a new profession, known as "prompt engineering", has already emerged, focusing on effectively utilizing AI tools.

Beyond the sector-specific risks posed by artificial intelligence, deep learning technologies such as deepfake can influence human perception, leading to issues such as fraud or defamation. Furthermore, conversational AI models can hinder social interaction by encouraging individuals to spend excessive time in virtual environments with artificial companions, potentially weakening their sense of reality and social skills. The psychological and sociological impacts of such applications must be thoroughly analyzed and necessary precautions should be implemented by authorities. Social security organizations and regulatory bodies have begun to take a more active role in addressing these challenges by introducing various legal frameworks and regulations.

In recent days, a new multimodal AI model is introduced named Grok [2]. With Grok, deepfakes can be created quickly and for free, delivering highly effective results. It has demonstrated success in challenging tasks such as converting readable text to images, transforming images into videos, and generating visuals from text. However, predicting the usage and impact of such rapidly advancing and publicly accessible technologies remains highly challenging.

For example, shortly after its beta release, a new visual generator of Grok, Aurora, known for producing highly realistic deepfakes, was discontinued and replaced by another visual generator. During its brief availability, numerous malicious deepfakes with high realism were widely shared. This scenario illustrates how fast technology is advancing, often outpacing humanity's ability to predict its consequences. Despite extensive analyses on the implications of these innovations, practical implementations remain scarce, underscoring the difficulty of anticipating real-world outcomes.

New technologies must undergo a thorough examination for potential misuse scenarios even before their release, enabling the implementation of more proactive measures. This highlights the need to educate people about deepfakes and equip them to critically evaluate the authenticity of the content they encounter. Individuals and governments have significant roles to play in addressing these challenges. Furthermore, collaboration with experts in fields such as security and social psychology is essential to develop effective strategies and protect society from the potential harms of these advances.

This thesis provides a comparative analysis of recent successful methods using various criteria. We aim to establish a foundation for future investigations by centering our research on studies from the past three years. Our experiments primarily evaluate the effectiveness of Vision Transformers (ViTs) [3] in this domain. Moreover, we implement and analyze the performance of the Convolutional Cross ViT (CCViT) model [4], which incorporates Vision Transformers for deepfake detection. As part of this work, we also introduce a new deepfake detection model, the Combinational Cross-Attention ViT (CCAViT).

The structure of this thesis is organized as follows. Section 2 provides a review of the literature on deepfake detection, covering key perspectives and classification approaches from prominent survey studies. This section classifies models based on their modality and, in its subsections, delves into recent state-of-the-art deepfake detection models, emphasizing their similarities and differences. The interrelations and dependencies among various detection methods are thoroughly explored to provide a cohesive understanding of their advantages and limitations. Section 2 also includes an introduction of the deepfake detection datasets that are used in our experiments. Section 3 details the architecture and implementation specifics of our proposed CCAViT model. Section 4 is divided into parts. We first present and analyze the performance results of the models discussed in the thesis. Then, there are the implementation details of the framework for the deepfake detection model. Benchmark datasets such as FF++ [5], DFDC [6], DFD [7], and CelebDF (CDF) [8] are used in our evaluations.

We also propose novel data preprocessing techniques. Optimal practices are investigated including the appropriate number of frames for training and testing. Furthermore, the performance of CCAViT is assessed under various conditions. Images are provided to enhance the visual understanding of the model's test results. The performance of the model was assessed using various performance metrics, such as accuracy and F1 score. To achieve optimal performance from the CCAViT, experiments were conducted for hyperparameter tuning. During fine-tuning of the hyperparameters, results are analyzed in a comparative manner to evaluate how each hyperparameter has influence on the model's effectiveness. The thesis concludes with Section 5, summarizing the findings and implications.

2. LITERATURE SURVEY

Survey-based studies on deepfake detection often classify detection methods using various criteria. For example, a notable comprehensive survey [9] on the deepfake organizes deepfake detection techniques into four primary categories: machine learning-based methods, deep learning-based methods, statistical measurement-based methods, and blockchain-based methods. As this study seeks to dig deeper into the most advanced and effective approaches, the focus is placed on deep learning-based methods, which have demonstrated superior performance compared to others. In general, deepfake detection techniques can also be categorized based on whether they rely on predefined features. Early detection models typically involve extracting predefined high-level features from images, followed by predictions made using machine learning methods such as Support Vector Machine (SVM), Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN).

In this thesis, we discuss the methods and techniques used in deepfake detection, in order to understand their mechanisms, evaluate their effectiveness, and explore their broader implications. To gain a better understanding of how these methods work and why they are so significant, it is essential to first explore deepfake generation techniques. We have categorized deepfake generation methods into three primary approaches: Facial Image Generation, Face Swapping, and Face Reenactment.

2.1. Deepfake Generation Methods

The first method, commonly referred to as "Face Synthesis" or "Facial Image Generation", involves generating facial images of identities that do not exist. In this approach, models are designed to produce distinct outputs corresponding to different random noise inputs. These models essentially learn the distribution of facial features and characteristics within their parameters, enabling them to generate realistic faces by modeling the underlying distribution of facial structures.

In the facial image generation process, larger and more diverse deep learning models capable of capturing complex distributions deliver significantly better performance in generating detailed and realistic faces. Among the methods used for face generation, Generative Adversarial Networks (GANs) [10] are the most prominent.

GANs introduce a dynamic learning process, unlike traditional neural networks, which rely primarily on labeled data and achieve their best performance when fitting training and validation sets. Proposed by Ian Goodfellow et al., GANs consist of two competing neural networks: the generator and the discriminator. The generator produces increasingly realistic images by iteratively refining its outputs based on the feedback from the discriminator. Meanwhile, the discriminator classifies images as real or fake and improves its ability to distinguish between the two.

This adversarial process ensures that both networks improve over time. The generator becomes more skilled at creating convincing images, while the discriminator refines its ability to identify fakes. Eventually, the system reaches equilibrium, where the generator produces highly realistic images and the discriminator achieves balanced performance. This framework has established GANs as a leading method for the facial image generation approach.

In recent years, as GANs have demonstrated high performance, their range of applications has expanded significantly. For example, the CycleGAN [11] model enables domain translation, allowing tasks such as aging young individuals in photos or altering a summer photograph to appear as if it was taken in a different season.

GANs have also become widely used in the deepfake field, proving to be highly promising by generating hyper-realistic media content that does not exist in reality. In the context of face generation for deepfakes, notable examples of GANs include StyleGAN [12] and StarGAN [13], both of which are commonly used to create highly realistic facial images.

Another face generation tool that does not require a prior image, source, or target image is the autoencoder. Autoencoders function as deep feature extractors. In the encoder phase, they compress the input data, capturing the essential features that make up the image. In the decoder phase, the model learns to reconstruct the original image with minimal difference between the generated output and the original input. This process relies on a loss function based on the similarity between the original and the reconstructed image.

Once trained, the encoder and decoder enable the decoder to generate new face images directly from latent space features. By providing random feature inputs to the decoder, it can produce entirely new images. Variational Autoencoders (VAEs) enhance this process by incorporating mean and variance calculations during encoding and decoding. This addition allows VAE to better capture the distribution of human faces, resulting in more realistic output. However, in recent years, the use of autoencoders and VAEs for deepfake generation has decreased significantly compared to GAN-based models, which have become the dominant approach in this field.

Moreover, instead of relying on a single approach, modern datasets increasingly feature deepfakes where face swapping and face reenactment techniques are applied together on the same image. This trend highlights the evolving complexity of deepfake generation, posing greater challenges for detection models to accurately identify manipulated content.

Face swapping can be defined as the process of exchanging facial regions between two different images. However, in the context of deepfake technology, face-swapping involves extracting the facial region from a source image and integrating it into a target image while preserving the background and non-facial parts of the target. This operation generates a new fake image or video. Face-swapping often overlaps with face reenactment, as it frequently retains the facial expressions of the target image, effectively reshaping the source face to match the target's structure.

Traditional methods used for face-swapping include Poisson image editing, histogram matching, and shape-based warping. Poisson image editing is based on the gradient matching between the source and target images. By solving the Poisson equation for gradients, it ensures consistency in local texture, lighting, color, and intensity, allowing for seamless integration of facial regions.

Histogram matching, on the contrary, operates at the pixel level rather than focusing on gradients. It aligns the distributions of pixel attributes, such as color, between the source and target images to achieve visual consistency. Shape-based warping, as its name suggests, concentrates on geometric transformations. This method uses facial landmarks to adjust the shape of facial features in the source image, ensuring alignment with those in the target image.

In addition to these traditional methods, GAN-based techniques are increasingly employed in face-swapping tasks. While GANs are well-known for facial image generation, they have also become widely adopted for creating highly realistic face-swapping applications due to their ability to model complex facial structures and attributes effectively.

Most of the benchmark datasets used by detection models consist mainly of content generated through face-swapping and face-reenactment methods. Recently introduced datasets have started to include a growing number of samples created using GAN-based techniques, reflecting advances in deepfake generation methods. High-quality face swapping can be achieved using CycleGAN. Furthermore, the FaceShifter sub-dataset in the FF++ dataset, which we utilized in our experiments, consists of data generated through face swapping using GANs. This underscores the significant role of GANs not only in facial image generation but also in face-swapping applications, contributing to the creation of highly realistic deepfake content.

Autoencoder-based approaches are widely used in face-swapping tasks. Typically, two separate encoder-decoder pairs are trained independently for source and target im-

ages. Once trained, the source and target images are passed through their respective encoders to obtain latent features. These features are then fed into each other's decoders, resulting in face-swapped outputs. In addition, there are models that utilize a single shared encoder during training, paired with two separate decoders for source and target images. In these variants, face swapping is achieved during the testing phase by exchanging the decoders, allowing the latent features from the shared encoder to be decoded into swapped outputs. The FaceSwapping sub-dataset of the FF++ dataset, which we utilized in our experiments, serves as an example of videos generated using this method.

2.2. Deepfake Detection Methods

Analyzing video frames involves capturing high-level visual and behavioral patterns to understand semantic information. Deep learning algorithms often face challenges in accurately modeling semantic features such as natural facial expressions, blinking, and head pose. Since many generative models operate frame-by-frame basis, the deepfake faces they produce frequently exhibit unnatural features, such as consistently open eyes. Moreover, these models typically do not capture temporal dependencies between consecutive frames. In [14], an effective technique for identifying fake videos was introduced by examining the absence or irregularities in eye-blinking patterns within AI-generated faces. This method employed a CNN architecture to determine whether the eyes were open or closed, while an LSTM utilized temporal data for predictions. Similarly, [15] focused on analyzing the speed of eye movements and coordination between both eyes by examining pupil positions and eye geometry, using these features for detection through an SVM classifier. In another study, Haliassos et al. [16] explored inconsistencies in the speech process to identify deepfakes. A different approach, proposed in [17], detects irregularities in three-dimensional head poses in the images. This technique utilizes a multilayer convolutional network to extract features, which are then embedded and processed by a temporal convolutional network for prediction.

Deepfake detection has also been approached with models that rely on large-scale standalone Deep Neural Networks (DNNs), bypassing the need for a separate feature extractor [18]. MesoNet, for instance, consists of sequential convolutional, batch normalization, and pooling layers, followed by a dense network. All components in the MesoNet are trained end-to-end. On the other hand, methods such as Scale Invariant Feature Transform (SIFT) [19] utilize deep neural networks as feature extractors to capture more intricate patterns from deeper layers of DNN. Similarly, [20] employs only CNNs, such as XceptionNet and MobileNet, for deepfake detection tasks. Another innovative approach is demonstrated by [21], where capsule networks address the positional sensitivity issues of CNNs by representing spatial features through vectors.

We propose a novel classification framework that emphasizes the shared characteristics of state-of-the-art methods in the literature review section. This approach aligns with our objective to analyze the similarities and differences between these advanced techniques. Deepfakes are classified into five dimensions based on the model's modality: spatial, temporal, frequency, audio, and multimodal. In this section, a modality-based classification is adopted to provide a coherent and systematic framework for examining detection methods from a historical perspective.

Deepfake detection methods, such as those in [15], [17, 18], [21], focus exclusively on spatial features, operating within a single modality framework. In contrast, approaches such as [14], [16] integrate spatial data with temporal dynamics to improve detection performance. Temporal models typically leverage spatial characteristics alongside temporal patterns through combined design strategies. Depending on their use of temporal information, deepfake generation and detection techniques can be categorized as frame-based or video-based. The high realism achieved by frame-based generation models in the spatial domain has motivated detection methods to place more emphasis on temporal features. Furthermore, low-resolution images and videos often lack sufficient visual detail for reliable spatial analysis. Another limitation arises when detectors, trained on deepfakes generated by specific methods, overfit unique spatial artifacts, hindering their ability to generalize across diverse datasets.

Temporal models, such as Recurrent Neural Networks (RNNs), are designed to process sequential data effectively. Among these, Long Short-Term Memory (LSTM) networks stand out for their ability to capture long-term dependencies, facilitated by an architecture that addresses the problem of vanishing gradients. In the context of deepfake detection, LSTM is commonly paired with a convolutional neural network (CNN) as feature extractors, forming a convolutional LSTM. For example, [22] achieved strong results using convolutional LSTM for videos at least two seconds long. Similarly, [23] demonstrated that models that employ three-dimensional convolutions are more effective in identifying motion-related anomalies due to their ability to analyze spatio-temporal data. Another notable approach, proposed by Cairong Zhao et al., is the Interpretable Spatial-Temporal Video Transformer (ISTVT) [24]. This architecture incorporates self-attention and a self-subtraction mechanism, enabling the visualization of temporal and spatial heatmaps while addressing the interpretability issues associated with large-scale models such as vision transformers.

An early example of a deepfake detection method leveraging frequency domain features is presented in [25], where the Discrete Cosine Transform (DCT) is preferred over the more commonly used Fast Fourier Transform (FFT) for frequency transformation. In [26], the proposed model incorporates a residual-guided spatial attention module and a high-frequency noise extraction module to address overfitting issues in spatial models caused by method-specific artifacts. Instead of focusing on artifacts unique to specific fake face generation techniques, this approach targets inconsistencies arising from face-blending operations such as color correction and blending, thereby improving generalization. Similarly, the model in [27] integrates a transformer architecture with two innovative modules utilizing information from the spatial and frequency domains. This design includes a Central Difference Attention (CDA) module that outperforms traditional approaches relying on DCT and FFT by identifying subtle fine-grained forgery artifacts. Additionally, the architecture employs a high-pass filter module to capture high-frequency details, further enhancing its capacity to generalize effectively across diverse datasets.

In [28], recent surveys on deepfake detection are reviewed, with particular criticism directed at their insufficient focus on the audio modality of deepfakes. The approach introduced in [29] addresses this gap by utilizing two independent detectors: one for the detection of audio spoof and the other for the analysis of video image. The performance of these detectors is assessed separately. However, a key limitation of this method is the lack of a unified decision-making mechanism that integrates information from multiple modalities when identifying deepfakes. To overcome this, [30] presents a synchronized joint audiovisual detection model, which effectively combines data from both modalities. Their findings reveal that this integrated approach surpasses separate stream models, where decisions are derived from independently computed probability outputs of modality-specific detectors.

The model proposed in [31] represents a multimodal approach by integrating spatial and temporal feature mapping within the frequency domain. The model identifies temporal inconsistencies in deepfake videos by detecting deviations from the expected random distribution in high- and low-frequency components. Similarly, [32] focuses on exposing artifacts at different levels, including low, medium, and high, that arise during image processing. Low-level features are extracted using steganalysis techniques that leverage abnormal statistical patterns in pixel distributions within the frequency domain. In contrast, middle- and high-level features are derived through a spatial-temporal network combining CNNs and LSTMs. Sliced Spatio-Temporal Network (SSTNet), the resulting architecture, operates in the spatial, temporal, and frequency domains, offering a comprehensive approach to deepfake detection.

This study explores various approaches across multiple modalities, emphasizing the benefits of ensemble techniques in deepfake detection. Combining multiple detection methods enhances prediction accuracy while promoting better generalization by leveraging the diversity of integrated techniques. EfficientNet [33], a widely recognized architecture for visual tasks, offers outstanding accuracy and efficiency in image recognition and classification. For example, [34] achieved impressive results by integrating a standard EfficientNet-B4 model with another EfficientNet-B4 model enhanced by an

attention mechanism. This attention mechanism improves detection performance by directing focus toward the most relevant areas of the input image. Another notable approach, DeepfakeStack [35], employs an ensemble of basic deep learning models using a greedy pairwise pretraining strategy. This method capitalizes on the strengths of individual models, enhancing overall performance and surpassing the capabilities of the standalone base models.

The evolution of deepfake detection has moved beyond simple visual methods to advanced deep learning approaches incorporating temporal dynamics, multimodal inputs, and GAN-based strategies. With deepfake generation techniques becoming increasingly complex, it is essential for detection models to remain robust against emerging manipulations and maintain their ability to generalize across various datasets.

In the following sections, we analyze 13 studies selected for their notable success. The analysis groups these methods based on shared attributes and evaluates them under three overarching themes.

2.2.1. Identity-Based Approaches

Overfitting to patterns such as background details and identity-related features is a common issue in deepfake detection models. Identity leakage, a phenomenon in which a synthetic face retains identity traits from both the source and target images, exacerbates this problem. To mitigate identity leakage, researchers have introduced several strategies. Studies have highlighted that this leakage significantly reduces the generalizability of deep neural network models across diverse datasets.

Deepfake detection can be enhanced using the FST Matching (the matching fake, source, target images) approach [36], which enables models to differentiate the identity features of source and target images from fabricated attributes. Expanding on this idea, the Uncovering Common Features for Generalizable Deepfake Detection (UCF) method [37] goes further by incorporating method-specific features alongside the feature

space defined in FST Matching. Since datasets often represent a narrow range of deepfake generation techniques, models risk overfitting to textures unique to those methods. UCF demonstrates superior generalization capabilities across varied datasets and deepfake face generation approaches by utilizing fake attributes extracted from both content-based and method-specific features.

The Implicit Identity Driven (IID) framework introduced by Baojin Huang et al. [38] improve detection performance by utilizing face identity information, rather than isolating it from fake attributes in deepfakes with face swapping. The framework involves extracting feature vectors from the source and target images separately and calculating the distance between these vectors. After training on the FF++ dataset and evaluating on the DFDC dataset, their approach achieved the highest score in Table 2.1, with an Area Under Curve (AUC) of %81.23. Despite its effectiveness, the method's dependency on identity information requires datasets to be labeled according to identity features, which introduces additional constraints.

Shichao Dong et al. [39] designed a deepfake detection model that operates independently of identity awareness to address the issue of implicit identity leakage. Their approach features an Artifact Detection Module (ADM) that focuses on localized distortion areas within images. This ensures that the model does not learn general features, such as identity information. ADM shares conceptual similarities with models that use vision transformers by segmenting images into distinct regions and incorporating positional data.

Biases related to gender and race present a risk for models, as does their tendency to overfit identity information. Although fair loss functions can reduce bias within a specific domain, they generally do not enhance performance across different domains. Preserving Fairness in Deepfake Detection (PFGDFD) [40] method addresses this challenge by combining disentanglement learning with fair learning modules, achieving notable improvements in both generalization and fairness. This approach demonstrates exceptional performance on datasets previously unseen by the model.

Table 2.1. Performance of the methods trained on FF++ in terms of AUC(%).

Model	Year	Backbone	FF++	DFD	DFDC	CDF
FST-Matching [36]	2022	EfficientNet-B4	99.92	-	-	89.39
DFDT [41]	2022	ViT	99.94	-	76.10	88.30
HCIL [42]	2022	ResNet-50	99.01	-	69.21	79.00
UCF [37]	2023	Xception	99.60	94.50	80.50	82.40
IID [38]	2023	ResNet18	99.32	93.92	81.23	83.80
ADM [39]	2023	EfficientNet-B3	99.76	-	73.85	93.88
TALL-Swin [43]	2023	Swin-B	99.87	-	76.78	90.79
SFDG [44]	2023	EfficientNet-B4	99.53	88.00	73.64	75.83
QAD-E [45]	2023	EfficientNet-B1	98.47	-	-	-
PFGDFD [40]	2024	Xception	98.28	84.82	61.47	74.42
LSDA [46]	2024	EfficientNet-B4	-	88.00	73.60	83.00
StyleDFD [47]	2024	ResNet-50	98.40	96.10	-	89.00
CCViT [4]	2024	EfficientNet-B0	99.28	93.50	63.10	79.90
CCAViT (Ours)	2025	EfficientNet-B0	99.58	93.80	64.30	82.00

As an improved version of Identity Consistency Transformer (ICT), Identity Consistency Transformer with Regerence (ICT-Ref) [48], has a transformer-based architecture with an identity-based approach. ICT-Ref, categorized as a method utilizing identity information and vision transformers, operates exclusively with original videos due to its reliance on generating fake images during training. Since no training experiments are reported on the FF++, ICT is not included in Table 2.1. However, when evaluated in the FF++ dataset after training in original videos outside of FF++, it achieved an impressive AUC score of %98.56. A key limitation of this approach lies in its focus on deepfake face images and videos specifically created through face-swapping techniques. Deepfakes generated by ICT using original datasets may struggle to succeed with newly developed or future generation methods, as ICT currently produces fake content through a limited set of predefined approaches within its framework.

2.2.2. Vision Transformer-Based Approaches

The remarkable success of vision transformers in deep learning applications has motivated their adoption in the domain of deepfake face detection. Facebook's Deit [49] demonstrated the potential to integrate vision transformers with the distillation method. Based on this concept, a study focused on deepfake video detection [50] achieved an impressive AUC score of %97.8 in the DFDC dataset. However, a significant drawback of models that employ the distillation method is their large size and high parameter count, which can affect computational efficiency.

DFDT [41] incorporates a vision transformer architecture as its backbone. The images are divided into square regions, and the resulting image patches are transformed into embeddings. These embeddings are then combined on the basis of the positional information of the image patches. Then, DFDT processes image patches of two distinct sizes using separate vision transformer blocks, enabling it to extract fine-grained and high-level features effectively. This innovative design allowed DFDT to outperform all other models reviewed, achieving a remarkable %99.94 AUC on the FF++ dataset. Unlike conventional approaches, DFDT leverages the vision transformer for classification tasks and feature extraction, making it particularly effective in identifying subtle artifacts within manipulated videos. Instead of conventional self-attention mechanisms, a recurrent attention mechanism is utilized, further enhancing model performance. Furthermore, its dual-patch embedding mechanism enhances the model's ability to generalize across different deepfake generation techniques.

A novel approach to deepfake video detection introduced by Davide Coccomini et al. [4] combines Convolutional Neural Networks (CNNs) with vision transformers. In this method, CNNs extract features that are further processed within the vision transformer for accurate prediction. Convolutional Cross Vision Transformer (CCViT), similar to the DFDT model, uses two distinct image patch sizes to capture detailed and high-level patterns. However, it used patches of latent features not image itself. Vectors from two different branches are combined using a cross-attention mechanism.

This approach enables the model to effectively capture both local and global forgery patterns, demonstrating high generalization performance. Convolutional ViT shares similarities with the DFDT model in its use of two different image patch sizes, enhancing its ability to detect diverse patterns. Despite its lower parameter count, the model achieved results comparable to those of ensemble learning approaches and distillation-based methods, both known for their effectiveness in achieving high accuracy.

A state-of-the-art multimodal approach to deepfake face detection, Temporal Aware Layered Learning (TALL) [43], combines three-dimensional spatial patterns and temporal patterns to ensure robust and reliable identification. This method was integrated into the swin transformer [51], which utilizes a sliding window strategy with self-attention mechanism, resulting in the creation of the TALL-Swin model. In this method, four consecutive images are randomly chosen from densely sampled video frames. After being subject to masking and resizing processes, these images are merged into a single 2×2 grid image. Subsequently, convolution operations with sliding windows are performed. Achieving an exceptional AUC score of %99.87, TALL-Swin stands out as the most effective technique to detect deepfakes in videos compressed with the c23 method from the FF++ dataset. In terms of model size, TALL-Swin operates with 86 million parameters, requiring fewer parameters and computational resources compared to other vision transformer models.

2.2.3. Other Approaches

The Spatial-Frequency Dynamic Graph (SFDG) [44] method establishes a correlated integration of frequency-related and spatial features, enhancing their interaction through dynamic graph learning. Similarly, Liang Chen et al. [52] employ various forgery techniques to generate synthesized images within datasets and trained these images using a Generative Adversarial Network (GAN). Input diversity is enhanced during image generation by employing a range of variations in parameters, including fake regions, blending methods, and mix ratios. Tests conducted with images compressed at different rates prove the model's improved generalization capability.

Quality-Agnostic Deepfake Detection EfficientNet (QAD-E) [45] employs a universal intra-model collaborative learning framework to adapt to diverse distributions of real and fake images. The model improves its generalization by using videos with a range of image qualities instead of relying on compression levels. In contrast, Hierarchical Contrastive Inconsistency Learning (HCIL) [42] focuses on detecting motion inconsistencies in deepfake videos. This framework is based on contrastive learning and examines both local and global perspectives of movements. HCIL excels at identifying subtle irregularities in manipulated videos by analyzing spatial and temporal dimensions simultaneously.

Face forgery methods often create distinct clusters in the latent space feature distribution. Latent Space Data Augmentation (LSDA) [46] mitigates overfitting to method-specific features by augmenting data within the latent space. Within-domain augmentation is performed using operations such as addition and rotation of latent features, while interpolation is applied for cross-domain augmentation. This approach is effective for datasets with a limited variety of forgery methods. However, in more complex scenarios, overlapping distributions in the latent space make distinguishing real images from fake ones more challenging.

Style Deepfake Detection (StyleDFD) [47] represents another approach that utilizes latent space, but it emphasizes analyzing the temporal behavior of style latent vectors rather than altering latent space features. This innovative method provides insights into the temporal consistency of latent representations, which are crucial to understanding deepfake generation mechanisms. Furthermore, it bridges the gap between spatial and temporal analyses by leveraging both aspects effectively. The method incorporates a style attention module that integrates temporal inconsistency data from latent space vectors with spatial features derived from content. The study revealed that fake videos tend to have lower variance among latent space vectors. Unlike many methods that rely on pixel-level temporal information, this approach targets the temporal dynamics of high-level features within the latent space. The interpretability of deep learning models remains a critical area of interest in the AI community. The inclusion

of two latent-space-based models in our analysis highlights a promising direction for future research.

3. PROPOSED METHOD

3.1. Combinational Cross-Attenntion Vision Transformer

Many models listed in Table 2.1 employ EfficientNet as their backbone for deepfake detection tasks. EfficientNet [33], a widely recognized CNN architecture, highly regarded for its accuracy and scaling efficiency, which ensure robust performance across a wide range of datasets. Vision Transformer (ViT), on the other hand, processes images by dividing them into patches and treating each patch as a token for a transformer, a mechanism originally designed for natural language processing. Unlike CNNs, which focus on local features, ViT captures global relationships throughout the entire image, demonstrating superior performance in tasks such as image classification, particularly when trained on large-scale datasets.

Early explorations of attention mechanisms in conjunction with CNNs, such as those of [53] and [54], laid the groundwork for enhanced feature extraction and representation learning. Big-Little Net [55] introduces a multiscale feature representation strategy by using CNN branches of different sizes for visual and speech recognition. Inspired by this approach, CrossViT [56] adopts two Vision Transformer branches, each processing image patches of varying sizes. Through cross-attention, the model effectively combines information from these branches, enhancing its capacity to represent multiscale visual information. Similarly, [57] achieved notable results by integrating transformers with a convolutional network that extracted patches from faces detected in videos, showcasing the potential of hybrid architectures in visual tasks.

Following the approach of [57], Davide Coccomini et al. [4] integrate EfficientNet with CrossViT, employing the architecture depicted in Figure 3.1 for deepfake detection. Another approach, presented in [58], utilizes a novel deep convolutional transformer that combines convolutional pooling with re-attention mechanisms.

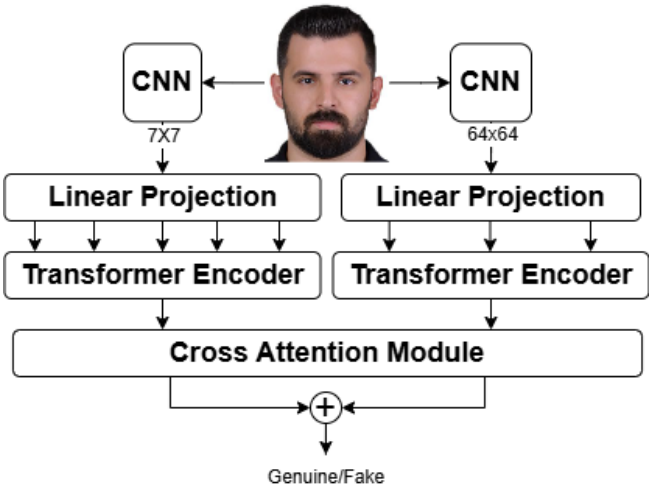


Figure 3.1. CCViT architecture.

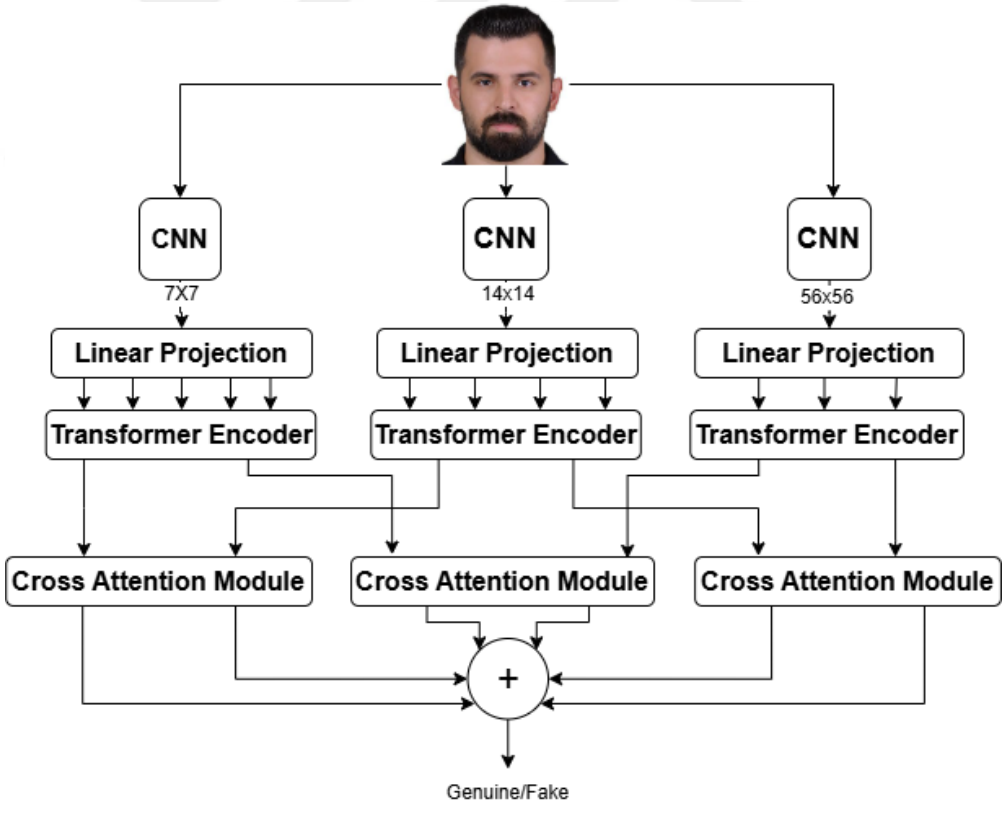


Figure 3.2. Combinational Cross-Attention ViT (CCAViT) architecture.

In our study, we implemented the Convolutional Cross ViT [4], a model designed for the detection of deepfake faces, and performed extensive tests. As illustrated in Figure 3.1, the Convolutional Cross ViT architecture generates visual features of two distinct sizes, 7×7 and 64×64, using EfficientNet-B0. This dual-patch design enhances the model's ability to capture both local details and global patterns effectively.

The Combinational Cross-Attention Vision Transformer (CCAViT), illustrated in Figure 3.2, is presented as a sophisticated solution for deepfake detection. This model is based on the convolutional ViT architecture, where each branch of the convolutional cross ViT structure originates from a straightforward implementation of convolutional ViT [57]. What distinguish CCAViT from CCViT are the inclusion of an additional branch with a mid-level patch size and the combinational connections to cross transformers. Inspired by MesoNet [18], which established the importance of mid-level mesoscopic features as a bridge between pixel-level details and global structures, this model is designed to capture low-, mid-, and high-level features comprehensively. Instead of using a two-branch architecture, the addition of a third branch enables a more detailed representation of facial features across multiple scales.

Face images are first processed through CNNs, using EfficientNet-B0 pretrained on ImageNet [59] for image classification. The model's final layers are removed via its delete block function, with the patch size determined by the specific layer where the truncation occurs. These two-dimensional visual patches are then converted into vector inputs through linear projection, making them compatible with transformer encoders. Vision transformers, combined with a cross-attention mechanism, serve as a classification module. The cross-attention mechanism enhances the focus on various local regions within the images, improving feature discrimination. Finally, the outputs from the three parallel streams are passed through a sigmoid function to generate the deepfake prediction.

The CCAViT architecture, shown in Figure 3.2, is specifically designed for a three-branch configuration, which is the default version discussed in this article un-

less otherwise stated. Each ViT branch in the model is paired with every other branch through cross-attention mechanisms. As the number of branches (n) increases, the number of cross-attention modules and the corresponding parameters grow at a quadratic rate, as they depend on all possible pairwise combinations of branches. Despite using EfficientNet-B0, a relatively lightweight backbone, CCAViT outperforms state-of-the-art models. Furthermore, we compare CCAViT with single-branch ViT, CCViT, and PCCViT (Parallel Convolutional Cross Vision Transformer), analyzing parameter counts and performance in both intra-dataset and cross-dataset evaluations. All models are based on EfficientNet-B0 and are trained with identical datasets, preprocessing methods, and voting strategies.

Deepfake detection datasets are typically large and require substantial storage capacity. Moreover, the models designed for these tasks are highly complex, with numerous parameters that demand significant GPU power and long training durations. To address these issues, a common practice is to compress videos using standards such as c23 and c40, which effectively reduce file sizes. However, this compression often reduces the resolution of images, potentially compromising the performance model. To avoid such drawbacks, we conducted our experiments using raw, uncompressed videos, preserving the original image quality.

3.2. Datasets

Many deep learning models are expected to demonstrate success in benchmark datasets before their performance in real world samples is widely accepted. If researchers did not showcase their models' results on similar datasets, it would be challenging to track progress across models, as each would claim success on different real-world datasets. However, the models that achieve the highest performance in benchmark datasets are not always the best in practice. In some cases, transitioning from well-established benchmark datasets to newer benchmarks can take considerable time. This delay can hinder the ability of rapidly evolving artificial intelligence technologies to address current real-world challenges effectively. Therefore, it is essential not to

focus solely on a single dataset. Models are expected to be evaluated not only on intra-dataset experiments but also through extensive cross-dataset studies to ensure broader applicability and robustness.

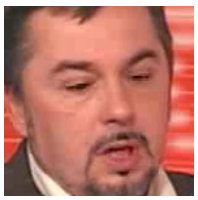
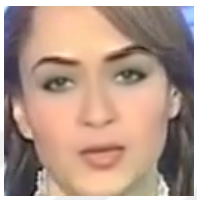
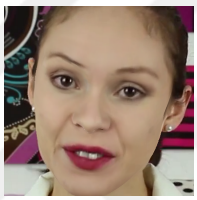
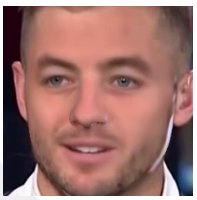
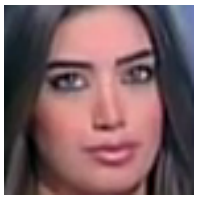
In the domain of deepfake detection, specific benchmark datasets have emerged as the standard for evaluating models, distinguishing themselves from others in terms of widespread adoption and utility. In this study, the most recent versions of these benchmarks are used to ensure the validity and applicability of the research findings. As illustrated in Figure 4.1, our review of the literature reveals that FF++ is the most widely used dataset for training and testing in deepfake detection studies. Prominent datasets in this field include FaceForensics++ (FF++), DeepFakeDetection (DFD), Deepfake Detection Challenge (DFDC), and CelebDF, all of which are incorporated into our experiments to ensure a comprehensive evaluation of the proposed models. This section provides a detailed analysis of these datasets, highlighting their characteristics and content.

3.2.1. FF++

The FF++ dataset comprises 1000 original videos, sourced from 997 distinct YouTube videos, and five sub-datasets containing deepfake videos generated using various manipulation techniques. The initially announced sub-datasets are Deepfakes, Face2Face, FaceSwap, and NeuralTextures, with FaceShifter added later. Each sub-dataset includes 1000 deepfake videos, generated by applying its respective deepfake generation method to all original videos. Table 3.1 demonstrates examples from all sub-datasets from FF++ [5].

To ensure consistency in experimental conditions and result comparisons, each sub-dataset is divided into 140 test videos, 140 validation videos, and 720 training videos. This setup results in a total of 700 manipulated videos and 140 original videos for each test set and validation set. For the training set, 3,600 manipulated videos and 720 original videos are used.

Table 3.1. Example images from sub-datasets of FF++.

Datasets	Images		
Original			
Deepfakes			
Face2Face			
FaceSwap			
FaceShifter			
NeuralTextures			

The raw version of the FF++ dataset requires approximately 700 gigabytes (GB) of storage. Compressed versions in two formats are available for FF++, providing smaller file sizes to accommodate varying storage constraints. Two compression rates are applied to the FF++ dataset using the H.264 video compression standard, com-

monly known as Advanced Video Coding (AVC). The compression levels, referred to as c23 and c40, are determined by constant rate factors (CRF), which range from 0 to 51. The c40 compressed FF++ dataset consists of highly compressed videos with a storage requirement of approximately two GB, while the c23 compression, considered the standard rate, requires around 10 GB.

Working with the 700 GB raw dataset poses significant challenges due to the substantial hardware requirements. Using all frames of videos can require storage capacities in the range of several terabytes. However, models trained on raw data often show superior performance during evaluations. In contrast, detecting deepfake content in compressed videos proves to be more challenging. One contributing factor to the performance drop in models tested on compressed data is the compression artifact. These artifacts can lead models to incorrectly classify non-deepfake videos as fake.

Training models on compressed videos can help them better adapt to scenarios involving compressed content. However, compression reduces video resolution, which adversely affects training performance by providing less detailed information and similarly impacts test set performance. State-of-the-art models typically achieve their highest performance metrics when trained and tested with raw videos. In our experiments, we opted to work with raw images to maximize model performance and ensure the best possible results across datasets. The sub-datasets and the deepfake generation methods used to create them are as follows:

(i) Deepfakes [60] is a specific manipulation method using two autoencoders with a shared encoder for face region swapping. The autoencoder output is then blended with the background of the image by Poisson image editing.
(ii) Face2Face [61] is a facial reenactment system designed to transfer the expression from a source video to a target video, while preserving the identity of the individual in the target video.
(iii) FaceSwap method [62] fits landmarks in swapped images and then blends with color correction.

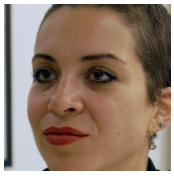
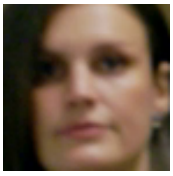
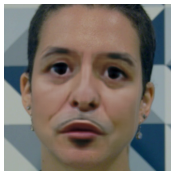
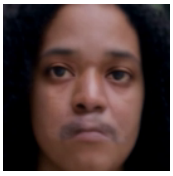
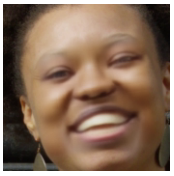
(iv) FaceShifter [63] capable of handling better facial occlusions. The method adaptively integrates the identity and the attributes for face synthesis. The model is design to maintain high visual fidelity in various conditions.

(v) The term "neural texture" refers to artificially generated textural details that are learned and stored by neural networks. FF++ utilizes the deferred neural rendering method [64] to create the NeuralTextures sub-dataset to create only the mouth part of the fake images. The deferred neural rendering method is based on neural rendering and neural textures. The NeuralTextures also utilizes GANs for manipulation of videos.

3.2.2. DFD

The DeepFakeDetection (DFD) dataset contains over 3,000 manipulated videos and 363 original video from 28 actors in 16 different scenes. The dataset has a similar file structure and is presented together with the FF++ regular dataset. DFD raw videos require a total capacity of 1.6 terabytes for manipulated videos and 200 gigabytes for the original videos. 387 manipulated videos are separated for test dataset among 3,000 manipulated videos. Cross-dataset evaluations also include 363 original videos. Table 3.2 presents sample images to provide a detailed perspective on the structure and content of the DFD [7] .

Table 3.2. Example images from DFD.

Datasets	DFD Images			
Original				
Fake				

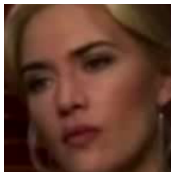
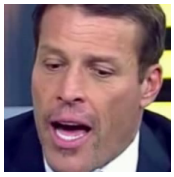
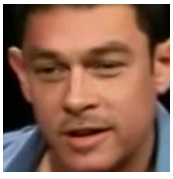
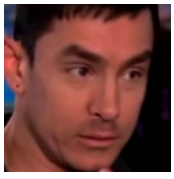
3.2.3. CelebDF

The Celeb-DF is a large-scale dataset which comprises 590 original videos collected from YouTube. 5,639 deepfake videos include videos with diverse ages, ethnic groups, and genders. The average length of all videos is approximately 13 seconds, with a standard frame rate of 30 frames per second. Original videos are sourced from publicly available YouTube content, primarily interviews of 59 celebrities.

The dataset employs Deep Neural Networks (DNNs) for color matching and applies video-level corrections using facial landmarks between frames. This approach incorporates visual and temporal features during the creation of deepfake content, resulting in highly realistic visual properties and temporal movements in facial regions. Table 3.3 showcases sample images, providing information on the composition and characteristics of the Celeb-DF dataset.

By enhancing the consistency of details such as natural facial expressions, mouth movements, and lighting conditions, Celeb-DF achieves a more realistic appearance compared to earlier datasets, where fake videos were of lower quality. Celeb-DF represents a significant improvement and is specifically designed to evaluate the generalizability of deepfake detection methods.

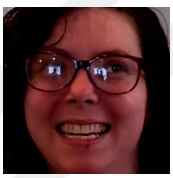
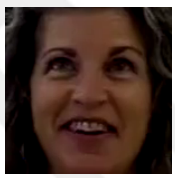
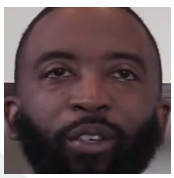
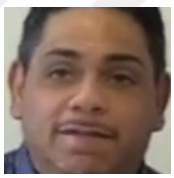
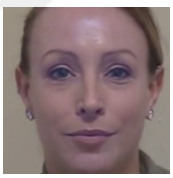
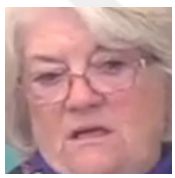
Table 3.3. Example images from CelebDF.

Datasets	CelebDF Images			
Original				
Fake				

3.2.4. DFDC

Deepfake Detection Challange dataset is a third generation deepfake detection dataset that is the largest among datasets used in our experiments. It consists of 124,000 videos including 3,426 paid actors. Featuring eight facial modification algorithms and eight modification algorithms. The test dataset has 5,000 videos, which is more than many datasets. Manipulation methods include face swapping using GANs. Table 3.4 shows the image frames randomly selected from DFDC [6].

Table 3.4. Example images from DFDC.

Datasets	DFDC Images			
Original				
Fake				

4. EXPERIMENTAL RESULTS

The recent deepfake detection methods included in this analysis represent some of the most remarkable advances in the field. Figure 4.1 illustrates the frequency of the datasets most commonly used in these studies, excluding survey-based research. Among the 35 works reviewed, FF++, CelebDF, DFDC, and DFD emerge as the datasets that are used most frequently. A few studies use datasets they created themselves, represented as 'self' in Figure 4.1. FF++ is particularly prominent, serving as the primary dataset for both training and testing, and is widely recognized for its contribution to high detection accuracy. In comparison, DFDC, DFD, and CelebDF are generally reserved for cross-dataset evaluations. These datasets are rarely used for training purposes and serve a critical role in evaluating the generalization performance of models trained on FF++.

The evaluation results for intra-dataset tests on the FF++ dataset and cross-dataset tests on the DFD, DFDC, and CDF datasets are summarized in Table 2.1. Among the models, DFDT achieved the highest performance in the intra-dataset evaluation. As a large-scale model, DFDT incorporates vision transformers directly into its backbone, contributing to its superior results. The AUC values listed in Table 2.1 represent the average scores obtained from the sub-datasets of FF++ that rely on face-swapping techniques for generating fake videos. The FF++ dataset itself consists of several sub-datasets corresponding to distinct deepfake generation methods, including Face2Face, FaceSwap, NeuralTextures, DeepFakes, and FaceShifter. When studies do not report average values, the table includes averages calculated from all sub-datasets to ensure comparability.

Table 4.1 presents the performance of various models evaluated in the DFDC dataset. The DFDC dataset is considered to be one of the most challenging benchmarks for deepfake face detection. Its synthetic videos are created using a variety of deepfake generation techniques, including Generative Adversarial Networks (GANs).

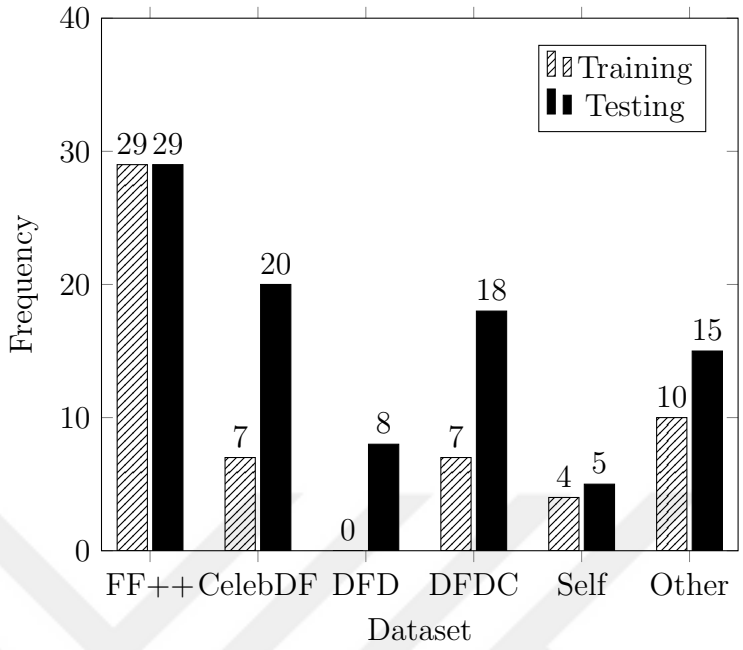


Figure 4.1. Dataset frequency in 35 Studies.

A few methods have achieved notable success in both intra-set and cross-dataset evaluations. Among these, the model proposed by [50] recorded the highest AUC score on the DFDC dataset. However, this model, which integrates a Vision Transformer (ViT) with a distillation mechanism, requires 462 million parameters, making it dependent on high-performance hardware and long training times. In comparison, the Convolutional Cross ViT model achieved nearly equivalent performance while utilizing approximately one-fourth the number of parameters with a compact backbone.

Table 4.1. Performance comparison of the methods on DFDC in terms of AUC.

Model	Year	Backbone	DFDC
ADM [39]	2023	EfficientNet-B3	95.67
ViT with distillation [50]	2021	EfficientNet-B7	97.80
CCViT [4]	2022	EfficientNet B0	95.10
Efficient ViT [4]	2022	EfficientNet-B0	91.90

The implemented models are tested and evaluated using the most widely utilized datasets in deepfake face detection research, including FF++, DFDC, DFD, and CelebDF. Furthermore, Table 2.1 highlights the best cross-dataset performance achieved by models trained on the FF++ dataset. Among these, the IID method, which adopts an identity-based approach, recorded the highest AUC score on the DFDC dataset. This outcome suggests that identity-based methods are particularly effective in enhancing generalization capabilities.

The literature review part of the thesis reveals that the most successful deepfake detection models in recent years predominantly utilize vision transformers or adopt strategies based on face identity information. We also figure out that FF++ is the most frequently used dataset for evaluating the performance of these models. Building on this trend, we develop a convolutional cross ViT structure [4] inspired by the vision transformer approach, introducing several enhancements to improve its effectiveness. The CCAViT model is tested under various experimental conditions. In our experiments, the model is trained using the FF++ dataset, and the results are evaluated through the FF++, DFD, DFDC, and CelebDF datasets, as detailed in Table 2.1. All experiments were performed with a GeForce RTX 4070 TI Super graphics card with 16 gigabytes (GB) of memory.

4.1. Combinational Cross-Attention ViT Results

4.1.1. Ablation Study

The ablation study presented in this thesis focuses on the primary components of the CCAViT architecture. Initially, we analyze the model performance when the combinational connections are excluded. Next, we investigate the impact of omitting additional middle-level features from the design. The study also evaluates the performance of CCAViT in the absence of the cross-attention module. In addition, we examine single-branch models that do not combine low- and high-level features.

The effect of excluding supplementary output from the convolutional neural network to the vision transformers is also assessed. This configuration represents a simplified convolutional vision transformer design, serving as a baseline for comparison. For each ablation condition, we performed a performance analysis using models named according to their architectural structure or, if applicable, based on existing terminology in the literature. Comparisons were made between models missing some specific component and those missing an additional component. For example, cross-attention cannot be applied to single-branch models. This approach allows for a more detailed evaluation of how each architectural element contributes to the model's overall performance.

A novel convolutional ViT architecture is proposed, as depicted in Figure 3.2, featuring cross-attention modules applied to all possible 2-way combinations of token outputs from n transformer branches. In this design, each of the n transformers represents an individual branch, with their outputs connected pairwise through cross-attention modules, resulting in $\binom{n}{2}$ such modules. Our experiments specifically focus on the case of a combinational architecture where n is three, as illustrated in Figure 3.2. Additionally, we implemented two baseline models: CCViT, shown in Figure 3.1, and a simple single-branch Convolutional ViT (CViT), for comparison.

We define Parallel Convolutional Cross ViT (PCCViT) as an extension by simply combining n CCViTs in parallel. In PCCViT, the cross-attention outputs of n CCViTs are aggregated before generating the binary prediction. PCCViT is referred to as P2CCViT or P3CCViT, depending on whether it consists of two or three CCViTs. Additionally, we implemented Parallel CViT (PCViT), two CViTs in parallel, without a cross-attention mechanism.

Table 4.2 provides details on the sizes and training times of CViT, PCViT, CCViT, P2CCViT, P3CCViT, and CCAViT with three and four branches. In addition to model size and architecture, training time is influenced by hardware components such as CPU, RAM, and GPU.

Due to hardware constraints, we trained and tested all models listed in Table 4.2 except for the four-branch CCAViT and P3CCViT, as these represent large-scale models. Since we use mini batch gradient descent optimizer, batch size also has an impact on training time. All training time values are calculated by training with the batch size of 16. The experimental results for intra- and cross-dataset evaluations of our CViT, PCViT, CCViT, CCAViT, P2CCViT implementations are summarized in Table 4.3.

Table 4.2. Model size and training time of our implemented models.

Model	Parameters	Average Training Time (min/epoch)
CViT	62M	3.71
P2CViT	89M	4.64
CCViT	101M	6.48
CCAViT (n=3)	152M	8.32
CCAViT (n=4)	213M	13.03
P2CCViT	159M	11.10
P3CCViT	216M	13.55

We conducted a comparison of various convolutional vision transformers with distinct designs to evaluate the architectural advantages of our CCAViT model. Although its performance does not match that of more sophisticated models, CViT, with a single branch consisting of 56×56 large patches, demonstrated satisfactory performance in intra-dataset evaluations, as presented in Table 4.3. However, in cross-dataset evaluations, its performance was limited, showing strong results only on the DFD dataset while failing to generalize effectively to CelebDF and DFDC test images. By increasing the number of branches to two in the P2CViT architecture, the model was able to combine small and large patches, enabling it to simultaneously capture low- and high-level features. This dual-branch structure enhances the model's generalization capability, resulting in substantial performance improvements on the CelebDF and DFDC datasets.

The CCViT architecture improved performance with the integration of the cross-attention module, particularly in cross-dataset evaluations. In contrast, P2CCViT did not perform as well in cross-dataset scenarios. A comparison between CCAViT and the more complex PCCViT architecture demonstrates that the combinational approach with three CNN branches produces better results than the integration of two CCViTs in parallel. CCAViT surpasses CViT, PViT, CCViT, and PCCViT in both intra-dataset and cross-dataset evaluations. This superiority is first attributed to the inclusion of an additional branch in CCAViT that captures middle-level features, enhancing its representational capacity compared to CCViT. Second, the combinational connections amplify the effectiveness of the cross-attention mechanism in convolutional vision transformers, leading to further performance gains.

Table 4.3. Test results of implemented models trained in FF++.

Model	FF++	DFD	DFDC	CDF
CViT	99.16	91.20	57.50	53.60
P2CViT	99.32	92.90	63.10	76.70
CCViT	99.38	93.50	63.10	79.70
P2CCViT	99.36	92.00	64.30	77.20
CCAViT	99.58	93.80	64.30	82.00

CViT, P2ViT, CCViT, CCAViT, and the parallel variants of CCViT were trained using the StepLR scheduler, binary cross-entropy loss, and mini batch Gradient Descent (GD) with weight decay, following the original methodology outlined in the study [4] that CCViT is proposed. To accommodate the unique combinational architecture of our proposed CCAViT model, we fine-tuned its training process. Each model was trained three times with the same set of variables to reduce the influence of the initial parameter values. Training was carried out over 300 epochs, with an early stopping criterion applied if no improvement in the validation score was observed for five consecutive iterations.

4.1.2. Implementation and Data Preproccessing Strategy

The best performing models were identified based on their highest validation accuracy among the top five models across all datasets. Since training accuracy consistently converged during our tests, we selected models based on epochs where training accuracy had stabilized, typically after 40 epochs. Importantly, the results of the top five models showed minimal variation in terms of accuracy, AUC, and F1 score, underscoring the reliability of the training process.

The process of predicting deepfake detection in videos begins with extracting images from video frames. Similarly to other methods for deepfake detection, our approach depends on analyzing images to determine video authenticity. We utilized the Multi-Task Cascaded Convolutional Neural Network (MTCNN) method [65] to extract frames that contain only facial regions. To ensure effective face detection across all videos, the threshold parameter of MTCNN was adjusted.

MTCNN uses three parameters, each with values between zero and one, to balance the weighting across the three stages of its cascaded deep convolutional network. Increasing a parameter value closer to one improves the accuracy of face detection predictions. However, this adjustment also decreases the probability of detecting a face in the frame, highlighting the trade-off between detection precision and coverage.

We introduced a technique called the cascaded MTCNN strategy, which iteratively applies MTCNN by starting with parameter values close to one and gradually reducing them until each video contains at least one frame available for deepfake detection. This strategy proves especially effective for datasets where extracting face frames is challenging. For instance, while MTCNN with high parameter values performs reliably for FF++, the cascaded strategy yields slight improvements in face detection accuracy for cross-dataset evaluations, such as DFDC. Using a 10-stage cascaded MTCNN approach, we improved AUC by %0.04 on DFDC.

Face detection plays a critical role in the accuracy of deepfake detection. Although MTCNN is among the most effective face detection methods for deepfake datasets, it struggles to accurately extract thousands of face frames from videos. Although the cascaded approach enhances model performance by improving face frame capture, integrating multiple face detection methods could further refine performance through cross-validation mechanisms, ensuring better reliability.

Deepfake datasets often contain a large number of frames in each video, however, it is unnecessary to use all of them for training or testing. Many frames provide almost identical information to the ones immediately preceding or following them. To optimize efficiency, we explore the most effective number of frames to use. Figure 4.2 illustrates the evaluation of a model trained in the FF++ dataset with 45 frames per video and tested on the DFD dataset with varying numbers of frames. Our experiments determine that using 30 frames is optimal for test data. For training, we selected 45 frames, as the performance began to stabilize at this number, as shown in Figure 4.3.

We reduce both storage requirements and testing times by decreasing the number of frames used. Furthermore, the time required for face detection and extraction decreased, as these times depend on the complexity of the algorithms employed to obtain images from videos. Using the FF++ dataset, we validated that 45 frames are sufficient for training by comparing the results obtained with 16, 30, 45, and 60 frames. This approach establishes threshold values that can be applied to real-world scenarios, guiding models trained not only on a single dataset but across diverse datasets to ensure that only the necessary frames are processed for each video.

There are several voting strategies that can be utilized for video-based predictions. In our approach, the prediction scores of images are derived from actor-based classification labels generated through MTCNN. For each actor, the final score is determined by averaging the scores of all images in which the actor is detected. A strict voting method is then applied, predicting the video as fake if any actor within the video is classified as fake based on their actor-based score.

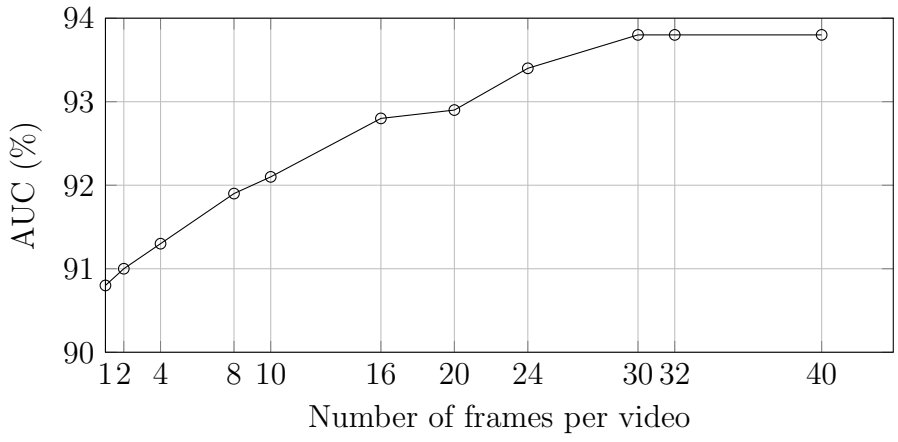


Figure 4.2. AUC for DFD test videos relative to frame number per video.

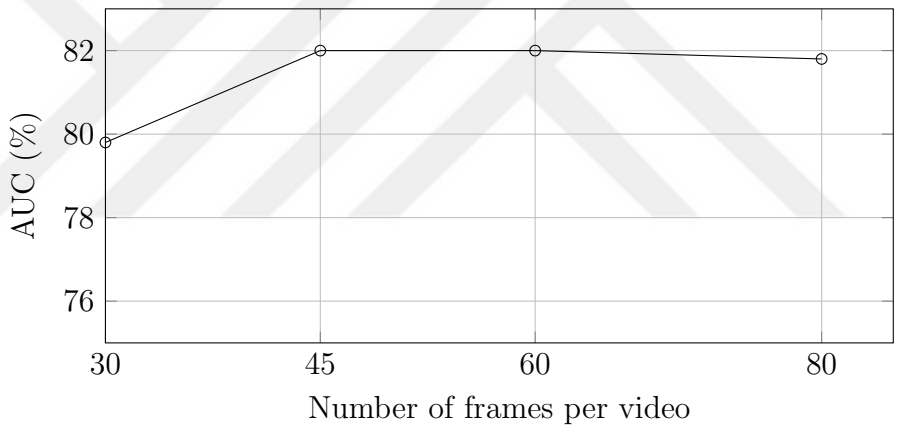
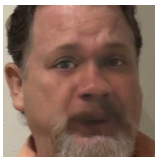
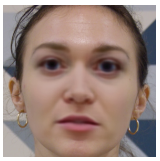
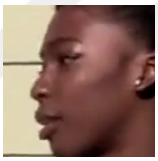
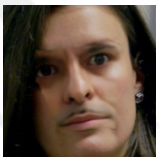
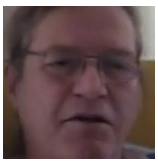
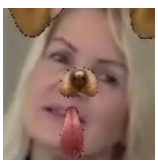
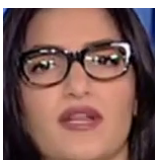
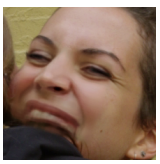


Figure 4.3. AUC for CelebDF test videos relative to frame number per training video.

4.1.3. Performance Analysis of CCAViT

Table 4.4 provides randomly chosen examples of fake images for correctly and incorrectly classified videos from FF++ [5], DFDC [6] and DFD [7] in our experiments. The first three rows in the correct prediction part of the table highlight instances where the CCAViT model successfully identified fake images, showcasing its strong detection capabilities. In contrast, the second part of the last three rows in the incorrect part of the table illustrates cases where the model misclassified fake images as real.

Table 4.4. Examples of fake images used in experiments with CCAViT.

Model Prediction	DFDC	FF++	DFD
Correct			
			
			
Incorrect			
			
			

A detailed analysis of benchmark datasets such as DFD and DFDC reveals that, as noted in [8], many fake samples are not cumbersome to detect for humans. However, certain fake samples remain highly challenging even for state-of-the-art models. For the FF++ dataset, the CCAViT model exhibited a notably low false positive rate, with only four incorrect predictions on fake image samples, as depicted in Table 4.4. This highlights the reliability of the model in detecting fake images with minimal errors.

The 18 examples in Table 4.4 emphasize the difficulty of distinguishing deepfake images solely through visual analysis, underscoring the advanced nature of modern deepfake generation techniques. To achieve robust performance on unseen images in real-world scenarios, models need to be trained on advanced datasets that include high-resolution original videos produced using the latest deepfake creation methods.

We conducted a detailed analysis of the correct and incorrect intra-dataset predictions made by CCAViT on the FF++ dataset, where the model exhibited exceptionally high success rates. Focusing on a small number of incorrect predictions allows for more precise generalizations. Our evaluation includes 840 test videos from FF++, consisting of 140 videos from each of the five sub-datasets and 140 original videos. In particular, the model did not make incorrect predictions among the 140 original videos. For the 700 AI-generated videos, only four incorrect predictions are observed, one of these being the same video pair manipulated using different methods. This highlights the model's strong performance and its reliability in detecting deepfakes.

The correct and incorrect predictions summarized in Table 4.4 are based on the classification of the prediction scores for images from benchmark datasets. The predictions depend on whether the scores from the last layer of the model exceed or fall below a predetermined threshold. Accuracy calculations are totally the result of calculated binary predictions. Such predictions not only reflect the model's ability to distinguish fake from real content but also highlight the importance of choosing an optimal threshold value to balance false positives and false negatives effectively. We choose 0.55 as the threshold value in our test calculations.

We analyze the performance of our CCAViT model, optimized with the best parameters, using confusion matrices based on accuracy values. The accuracy values presented in the confusion matrices correspond to video-level predictions rather than frame-level predictions. This approach ensures that the performance evaluation reflects the model's ability to classify entire videos accurately, rather than individual frames. To further investigate the performance of the model on the FF++ dataset,

confusion matrices were individually examined for each sub-dataset, including DeepFakes, Face2Face, FaceShifter, FaceSwap, and NeuralTextures. This detailed analysis provides insight into the effectiveness of the model in different video manipulation techniques.

Since deepfake detection is a binary classification problem, the confusion matrices for all datasets consist of four cells. False predictions are highlighted in red to indicate errors, whereas true predictions are marked in green to represent correct classifications. This color-coded representation helps to visually distinguish between correct and incorrect predictions. Confusion matrices provide a comprehensive view of the model's classification performance across both original and fake samples. Since the primary objective of a deepfake detection model is to identify fake videos rather than original ones, a positive prediction corresponds to the classification of a video as fake. The confusion matrice presented in Figure 4.4 includes the meaning of each cell.

- True Positive (TP): Fake videos correctly identified by the model as fake.
- False Positive (FP): Original videos incorrectly classified by the model as fake.
- False Negative (FN): Fake videos incorrectly classified by the model as original.
- True Negative (TN): Original videos correctly identified by the model as original.

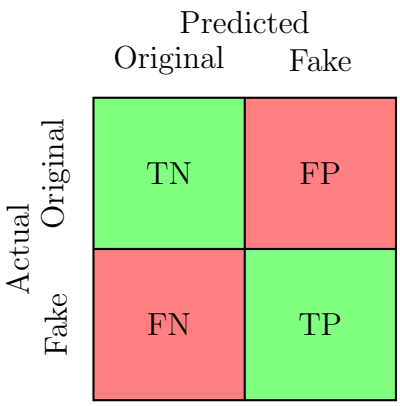


Figure 4.4. Confusion matrix of deepfake detection models.

The confusion matrices presented in Figures 4.5 and 4.6 illustrate the classification performance of CCAViT in the DeepFakes and FaceShifter sub-datasets in FF++, with high values for true positives (140) and true negatives (125), indicating the robustness of the model. All fake videos in the DeepFakes and FaceShifter datasets are correctly classified by the model.

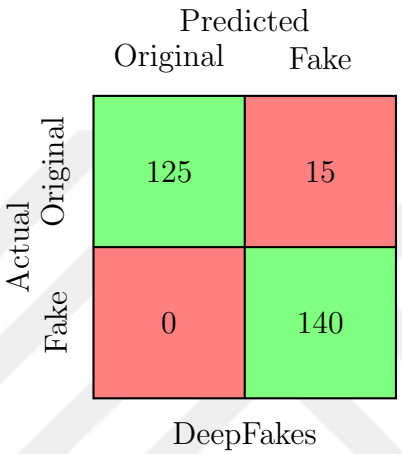


Figure 4.5. Confusion matrix of CCAViT in DeepFakes.

The results in Figures 4.7, 4.8, and 4.9 highlight the effectiveness of the CCAViT model. Since the original videos remain consistent across all sub-datasets, the 15 incorrect predictions for original videos are consistently reflected in the confusion matrices. The total accuracy for the FF++ is calculated by combining the incorrect predictions for fake videos in each sub-dataset with those for original videos. The CCAViT incorrectly classified five fake videos in FF++, resulting in a total of 20 incorrect predictions out of 840 videos. This corresponds to an overall accuracy of %97.61.

NeuralTextures stands out as the most challenging sub-dataset within FF++. Unlike the other sub-datasets, NeuralTextures and Face2Face are not based on face-swapping techniques; instead, they employ face reenactment. As a result, achieving a high success rate in the NeuralTextures sub-dataset is more difficult compared to those involving traditional face-swapping methods.

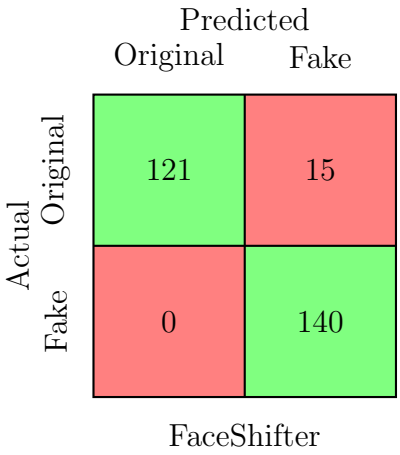


Figure 4.6. Confusion matrix of CCAViT in FaceShifter.

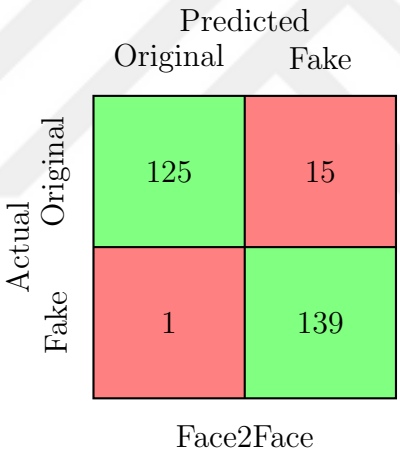


Figure 4.7. Confusion matrix of CCAViT in Face2Face.

The confusion matrices for FF++ reveal that the model exhibits imbalanced performance in the favor of fake videos. During the training phase, 720 videos are used for each sub-dataset, including 720 videos from the original dataset. This created a five to one ratio of fake to original videos. To maintain the generalization ability model, particularly due to the variety of fake generation methods, we chose not to adjust this ratio to an even one to one balance. However, to address the imbalance between original and fake videos, a rebalancing factor of 0.3 is applied during training. This factor controls the probability of selecting fake images.

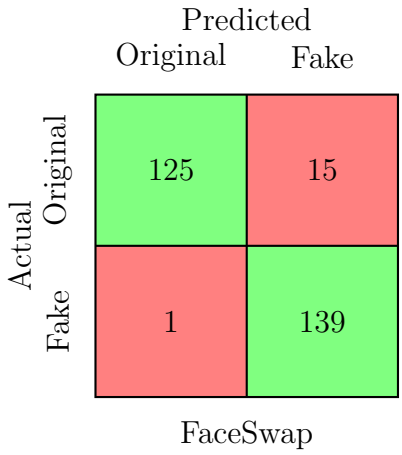


Figure 4.8. Confusion matrix of CCAViT in FaceSwap.

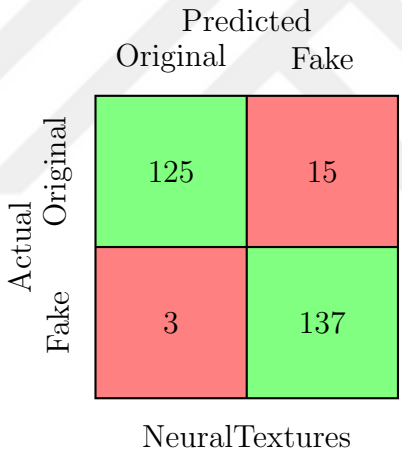


Figure 4.9. Confusion matrix of CCAViT in NeuralTextures.

In Table 4.5, the manipulated videos retain the identity and background details of the source image due to the manipulation approach used in the NeuralTextures dataset. The images in the table illustrate that partial mimicry and expression swapping are challenging to detect, even from a human perspective. Among the three images, the difference between the real and fake versions is noticeable only in the teeth of the rightmost image. However, no trace of manipulation or anomaly is evident in the presented images.

Table 4.5. Images incorrectly predicted by CCAViT from NeuralTextures.

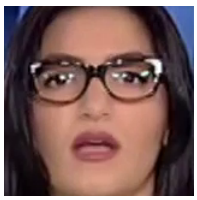
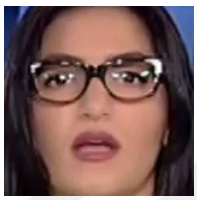
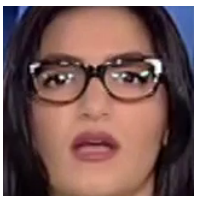
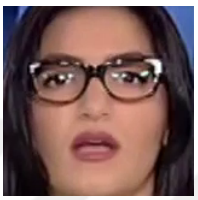
Image label	Neraltextures		
Original			
Deepfake			

Table 4.5 presents three misclassified images from the NeuralTextures [5] dataset along with their original counterparts. The CCAViT model successfully identified all three original images as genuine. However, for the corresponding fake images from the NeuralTextures dataset, the model did not make correct predictions. Upon detailed examination, it is evident that detecting the fake nature of these images is a highly challenging task, even for human perception. This difficulty arises from the fact that NeuralTextures is based on face reenactment rather than face swapping. Unlike face-swapping techniques, which alter the face itself, face reenactment modifies facial expressions and gestures between source and target images. Consequently, identifying these images as fake becomes significantly more complex.

In Table 4.5 and Table 4.6, the first frames are extracted from videos to allow a clear comparison between the same images in the manipulated and original forms. Table 4.6 presents the fake images from Face2Face [5] and FaceSwap [5] incorrectly predicted by CCAViT. The manipulated image of Face2Face closely resembles its original counterpart, making it difficult to distinguish. Similarly, the fake image from FaceSwap appears nearly identical to the original but differs slightly in the camera's gaze direction. Both images from these datasets present significant challenges in predicting their authenticity as either fake or real.

Table 4.6. Images incorrectly predicted by CCAViT Face2Face and FaceSwap.

Image label	Face2Face	FaceSwap
Original		
Deepfake		

We present four examples of original images from the FF++ dataset that were incorrectly predicted by CCAViT. Among the 15 original videos misclassified by the model, 13 contain background faces that are not related to the main characters in the videos. The deepfake detection model struggled to correctly classify these background faces, accounting for the majority of errors. This proportion, 13 out of 15, is significant given that the FF++ dataset consists primarily of videos without background faces. A detailed examination of the images in Table 4.7 reveals that all frames include two background faces, which misled the model.

The images in Table 4.7 show the full video frames of FF++ [5] before face extraction was applied. Green squares in the images indicate the faces detected by MTCNN. It is evident that even smaller faces relative to the foreground faces are detected, while very small individuals in the background remain undetected because of the appropriate parameter tuning of the face detector. This tuning ensures a balance between detecting relevant faces and avoiding excessively small or irrelevant facial frames.

Background faces pose a significant challenge to obtain the perfect deepfake detector, as highlighted in our experiments. Advanced face detection algorithms used

in the deepfake detection phase capture faces, even those with very small sizes. Consequently, the detection model can classify these low-resolution faces as fake images. Although this issue can be mitigated wisely using a face detection strategy, it is important to note that overly restrictive settings could lead to missed foreground faces, further impacting performance. Balancing sensitivity and precision in face detection is crucial to effectively address this problem.

Table 4.7. Examples of original images incorrectly predicted by CCAViT from FF++.

The intra-dataset confusion matrices are presented in Figures 4.10, 4.11, and 4.12. From the confusion matrices for FF++, it becomes evident that our training balancing approach does not fully address the performance issues with original images in the intra-dataset evaluations. Similarly, in DFD cross-dataset evaluations, the same challenge persists.

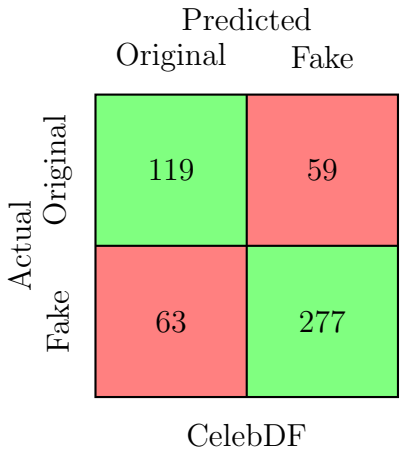


Figure 4.10. Confusion matrix of CCAViT in CelebDF.

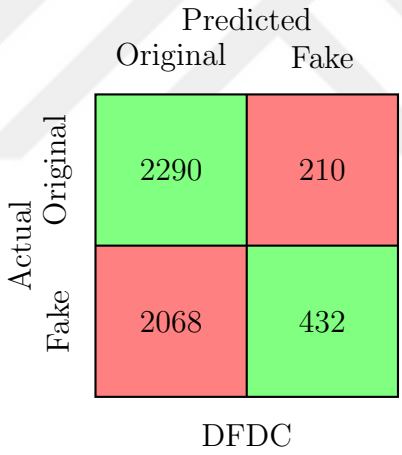


Figure 4.11. Confusion matrix of CCAViT in DFDC.

However, the model struggled more to predict fake videos in CelebDF and DFDC datasets. This difficulty is attributed to the diversity and the large number of generation methods present in these datasets. In real-world scenarios, the variety of generation methods is even greater and continuously evolving, making the development of robust real-world deepfake detection applications more challenging. The performance of deepfake detection methods is directly proportional to variety of datasets used during training.

CelebDF contains 178 real and 340 fake test videos, as illustrated in Figure 4.10. The CCAViT model demonstrates a more balanced performance in predicting both original and fake images on this dataset. In contrast, DFDC, a much larger dataset with 2,500 manipulated and 2,500 original videos, presents a greater challenge. As shown in the confusion matrix in Figure 4.11, CCAViT demonstrates robust performance on original videos, while it encounters considerable difficulty in accurately predicting fake images. This aligns with the widely recognized difficulty of DFDC, which includes a diverse range of deepfake generation methods, making it one of the most challenging datasets for deepfake detection.

The DFD dataset consists of 363 original videos and 387 manipulated videos. Figure 4.12 presents the confusion matrix for the DFD dataset using the CCAViT model. The results of the cross-dataset evaluation indicate that the model faces challenges in accurately detecting the original images. However, it performs exceptionally well in identifying fake images and correctly classifies the vast majority of them as fake. Since DFD shares more similarities with FF++, both published in the same study, we place greater emphasis on the CelebDF and DFDC results to evaluate the model's performance on truly unseen data. Consequently, our balancing strategy aligns well with the trends observed in the confusion matrices.

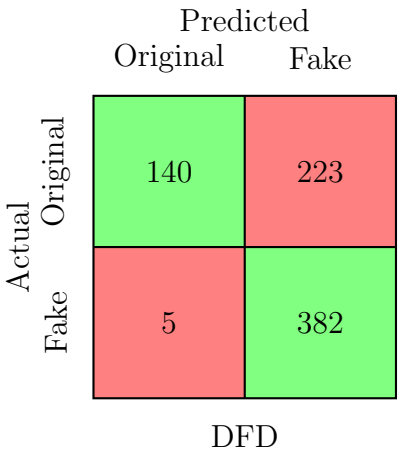


Figure 4.12. Confusion matrix of CCAViT in DFD.

The study in [20] trains four separate models, each focusing on one of the face generation methods within the FF++ dataset, and combines their outputs using an ensemble approach. In contrast, our method relies on a single model trained across the entire FF++ dataset, with the aim of creating a generalized solution capable of handling unseen examples. The vast range of generation techniques in real-world scenarios makes it impractical for models to specialize in individual methods. Instead, training on datasets with diverse generation techniques allows the model to develop strong generalization capabilities, enabling it to detect manipulations from previously unseen methods. To further evaluate its effectiveness, we test our model on each FF++ subset to measure its performance against specific manipulation techniques.

Videos with higher loss function errors during predictions were found to share a common characteristic. The FF++ dataset includes both single-face and multiface videos. Although the proportion of multiface videos is relatively small, seven out of eight original videos containing multiple faces in our analysis exhibited higher error values. Despite this, only one incorrect prediction occurred when applying the optimal threshold value. For videos predicted with high accuracy, it is evident that multiple individuals are clearly visible on the screen. In cases where a person appears in the background, either distant from the camera or displayed on a background monitor, the model's performance slightly declines because of reduced resolution and smaller face sizes in the frame.

Employing more exponential loss functions could potentially enhance the model's AUC score. However, this approach could negatively impact the model's cross-dataset performance by increasing the risk of overfitting. Therefore, our main objective is to increase performance not only by tracing the AUC score but also other indicators such as the accuracy of the results.

Improving predictions in scenarios involving undesirable background faces can be achieved by ignoring detected face frames that are considerably smaller than foreground faces or filtering out images below a specific resolution threshold. A more advanced so-

lution involves designing deep models capable of addressing multiface scenarios directly within their architecture, eliminating the reliance on heuristic-based methods.

Another approach is to train models for three-way classification, distinguishing between genuine faces, fake faces, and undefined categories for low-resolution face frames, although this would require new labels and datasets. Enhancing face detection models with specialized modules for detecting fake faces is becoming increasingly vital to improve overall performance.

4.1.4. Hyperparameter Tuning

The performance of the CCAViT model, similar to other deep learning models, is influenced by the choice of hyperparameters. Since CCAViT shares numerous similarities with CCViT as its foundation, we based our approach on the parameters used for CCViT to identify the most optimal values. With 152 million parameters, CCAViT requires significant training time and substantial hardware resources. Despite extensive fine-tuning tests, the potential for even better performance remains. Since the model incorporates both convolutional neural networks and vision transformers, it offers a wide range of architectural parameters that can be further optimized. This flexibility allows for the exploration of enhancements that could lead to improved outcomes in future studies.

All experiments in this section are performed using the optimal hyperparameters determined by prior analysis. For each experiment, all hyperparameter values are kept at their optimal settings except batch size, with only one parameter adjusted at a time to isolate its impact on the model's performance. This controlled approach ensures a clear understanding of how each individual hyperparameter influences the overall results while minimizing the effect of interactions between multiple parameters. The batch size value is not currently optimal in experiments, as it was the last parameter to be optimized. As new optimal values are identified, other hyperparameters can also be updated to further enhance the model's performance.

Throughout the hyperparameter tuning experiments, we use batch size of 16, 0.2 for dropout ratio in transformers, 0.2 for dropout ratio in cross transformers, 7,14 and 56 patch sizes, 0.01 for learning rate and 1.0 for middle branch coefficient. All hyperparameter tuning experiments are conducted with a training duration of 100 epochs. For models demonstrating performance close to the best performing ones, the number of epochs is increased to 300 to enable more precise fine-tuning to obtain the best model. For CCAViT, a remarkably higher number of epochs is used in hyperparameter tuning because even small performance improvements could solidify its position among state-of-the-art models.

This section of the thesis examines how hyperparameters such as batch size, the number of frames per video, drop rate, patch sizes, learning rate, and the mid-level coefficient influence the model's performance. Each table detailing the hyperparameters presents the optimal value as well as the values that are one step below and one step above the optimal. Some extreme hyperparameter values are excluded from the tables as their performance results were worse and irrelevant.

The CCAViT model utilizes Mini-batch Gradient Descent (GD) as its optimization algorithm. While Batch Gradient Descent processes the entire dataset and calls the loss function once per epoch, Stochastic Gradient Descent operates on a single data instance at a time per epoch. In contrast to these methods, the mini-batch gradient descent algorithm divides the dataset into smaller batches for processing. Gradients for all images within a batch are calculated and propagated simultaneously, enabling more efficient optimization.

We compared the performance of CCAViT using batch sizes of 8, 16 and 30, as shown in Table 4.8 and Table 4.9. Larger batch sizes demand higher GPU capacity, and batch sizes exceeding 30 were beyond the limits of our GPU hardware. Thus, 30 was selected as the upper limit. The results indicate that a batch size of 8 surprisingly outperforms 16 and 32. For deep learning models, larger batch sizes are often preferred because they tend to deliver better performance. However, the primary disadvantage

of using a large batch size is the increase in the amount of input data processed per epoch, which consequently extends the training time. Some of the large models are implemented using batch sizes of 100 or even 1,000. However, increasing the batch size does not yield better results for our model.

The performance of the CCAViT model with different hyperparameter values is evaluated on each sub-dataset in FF++. The FF++ dataset comprises DeepFakes (DF), Face2Face (F2F), FaceSwap (FS), NeuralTextures (NT), FaceShifter (FSh), and pristine images labeled real. For cross-dataset evaluations, the model is tested on the CelebDF (CDF), DFD, and DFDC datasets to assess its generalization capabilities. Each sub-dataset calculation also includes original videos. The average value is calculated by taking the direct mean of the AUC values across all sub-datasets. In fine-tuning experiments, all versions of the CCAViT model are trained using whole FF++ training set videos.

Table 4.8. AUC of CCAViT across sub-datasets of FF++ versus batch size.

Batch size	DF	F2F	FS	FSh	NT	AVG
4	99.80	99.30	99.60	99.50	99.00	99.44
8	100.00	99.30	99.90	99.70	99.00	99.58
16	100.00	99.30	99.90	99.70	99.00	99.58
30	99.90	99.10	99.80	99.50	98.60	99.38

Table 4.9. AUC of CCAViT in DFDC, DFD and CelebDF versus batch size.

Batch size	DFD	CelebDF	DFDC
4	93.10	76.40	64.00
8	93.80	82.00	64.30
16	93.80	80.20	64.00
30	93.50	75.40	63.10

Dropout is a regularization technique used in deep learning models, where some nodes are randomly "frozen" (set to inactive) during training. This process helps prevent overfitting by reducing the dependency on specific nodes and encouraging the model to generalize better. During testing, all nodes are utilized, including the previously frozen ones. Since the model uses fewer parameters during training compared to the full network, a compensation mechanism is applied during testing to account for this reduced parameter usage, ensuring consistent performance.

The dropout rates that we used in fine-tuning, such as 0.10, 0.25 and 0.20, correspond to percentage values, representing the proportion of nodes frozen during training. For example, a rate of 0.10 indicates that %1 of the nodes are frozen, while a ratio of 0.20 means %2 of the nodes are inactive during training. We obtain the best results using the dropout rate of 0.20 by fine-tuning the model.

Table 4.10 and Table 4.11 present the performance variations observed in experiments based on different dropout rate values. CCAViT demonstrates improved performance in both intra-dataset and cross-dataset evaluations by dropping out %20 of the nodes in the vision transformers during training. The optimal dropout rate for CCViT is reported as 0.15. Considering that the likelihood of overfitting the training set increases with larger models, CCAViT achieving better performance with a higher dropout rate compared to CCViT. Since the architectural designs of the two models are not identical, it is not appropriate to assume a linear relationship for all hyperparameters.

The cross-attention module of CCAViT incorporates cross-transformers with dropout strategy. We define the dropout rate used in the cross-transformers in CCAViT as the cross-dropout rate. The performance in the intra-dataset with cross-dropout rates of 0.15 and 0.20 is comparable as shown in Table 4.12, with 0.15 performing slightly better. In contrast, a cross-dropout rate of 0.20 improves the performance in the DFDC dataset by %1.7, as shown in Table 4.13, which is a notable increase. Furthermore, the dropout ratio of 0.15 yields slightly improved results in the DFD and CelebDF.

Table 4.10. AUC of CCAViT across sub-datasets of FF++ versus dropout rate.

Dropout rate	DF	F2F	FS	FSh	NT	AVG
0.15	99.90	99.30	99.80	99.50	98.90	99.48
0.20	100.00	99.30	99.90	99.70	99.00	99.58
0.25	99.90	99.20	99.70	99.50	98.80	99.42

Table 4.11. AUC of CCAViT in DFDC, DFD and CelebDF versus dropout rate.

Dropout rate	DFD	CelebDF	DFDC
0.15	92.50	73.70	62.70
0.20	93.80	80.20	64.00
0.25	92.70	79.00	63.10

Since the preferred dropout ratio for cross transformers in the CCViT model is reported to be 0.15, we used this value as our starting point. Given that CCAViT is a larger and more complex model compared to CCViT, it is expected to have a greater tendency to overfit under similar conditions. Increasing the cross-dropout ratio is typically effective in preventing overfitting. However, our experiments indicate that the high dropout rate applied to the cross-transformers in CCAViT did not produce the same effect. This outcome underscores the significant architectural differences in CCAViT, demonstrating that the impact of dropout can vary according to the model structure. Since the model comprises various components and architectures, including convolutional neural networks, vision transformers, and cross-attention modules, applying different dropout strategies to specific components can influence performance in unexpected ways. A dropout rate of 0.15 is determined to be the most optimal choice for cross-dropout. The model performs exceptionally well on certain datasets, and any decrease in performance on these would not be justified by a slight improvement in poorer-performing datasets. Since DFDC performance remains low with cross-dropout rate of 0.15, the improvement is insufficient to warrant prioritizing it over the performance on other datasets.

Table 4.12. AUC of CCAViT across sub-datasets of FF++ versus dropout rate of cross-attention module.

Cross dropout rate	DF	F2F	FS	FSh	NT	AVG
0.15	100.00	99.30	99.90	99.70	99.00	99.58
0.20	99.90	99.30	99.80	99.70	99.00	99.54

Table 4.13. AUC of CCAViT in DFDC, DFD and CelebDF versus dropout rate of cross-attention module.

Cross dropout ratio	DFD	CelebDF	DFDC
0.15	93.80	80.20	64.00
0.20	93.40	80.00	65.70

The CCAViT structure consists of three branches, each corresponding to a different patch size. For these branches, three convolutional networks (EfficientNet-B0s) are truncated at different specific depths, and the remaining blocks are discarded. Using 224×224 input images, EfficientNet-B0 can generate latent space features only with dimensions of 7×7, 14×14, 28×28, and 56×56. Patch sizes are determined on the basis of these feature sizes at specific layers of the CNN at backbone.

Models employing patching and a two-branch architecture, such as CCViT and DFDT, typically extract features from the first and last layers of the backbone. Thus, we selected patches of 7×7 for low-level features and 56×56 for high-level features. To include mid-level features, we introduced an additional branch in the CCAViT model. Experiments are conducted to identify the optimal mid-level patch size.

Table 4.14 and Table 4.15 present a comparison of the model performance using middle-branch patch sizes of 14×14 and 28×28. The model with a 28×28 patch size shows slightly better results in the DFDC dataset, while the 14×14 patch size outperforms in both intra-dataset and cross-dataset evaluations.

Table 4.14. AUC of CCAViT across sub-datasets of FF++ versus middle branch patch size.

Middle branch patch size	DF	F2F	FS	FSh	NT	AVG
14×14	100.00	99.30	99.90	99.70	99.00	99.54
28×28	99.90	99.20	99.80	99.60	98.80	99.46

Table 4.15. AUC of CCAViT in DFDC, DFD and CelebDF versus middle branch patch size.

Middle branch patch size	DFD	CelebDF	DFDC
14×14	93.80	80.20	64.00
28×28	92.80	79.80	64.20

The learning rate is a hyperparameter that controls the extent to which gradients are updated during each backward calculation. A higher learning rate allows the model to adjust the parameters more quickly, but excessively high values can hinder fine-tuning and may prevent convergence. Conversely, a very small learning rate enables the model to learn slowly and risks getting stuck in local minima, but it can achieve a highly precise solution. Even when using a high learning rate, the application of weight decay in our training process helps the model converge to better results toward the end of the training phase.

We selected a learning rate of 0.1 as the starting point, consistent with the recommended value for the CCViT model. Since our proposed model has a larger number of parameters, we demonstrated whether it could perform successfully with a high learning rate. As shown in Table 4.16 and Table 4.17, increasing or decreasing the learning rate resulted in reduced performance. Despite the significant difference in model sizes, the CCAViT model achieved optimal results with learning rates in a range similar to that of CCViT, highlighting the compatibility of the proposed structures.

Table 4.16. AUC of CCAViT across sub-datasets of FF++ versus learning rate.

Learning rate	DF	F2F	FS	FSh	NT	AVG
0.005	99.90	99.20	99.70	99.30	98.60	99.40
0.01	100.00	99.30	99.90	99.70	99.00	99.58
0.02	99.90	99.20	99.90	99.60	98.90	99.50

Table 4.17. AUC of CCAViT in DFDC, DFD and CelebDF versus learning rate.

Learning rate	DFD	CelebDF	DFDC
0.005	92.80	76.10	63.60
0.01	93.80	80.20	64.00
0.02	93.80	80.20	63.00

We evaluated the impact of mid-level features by assigning different coefficients during the aggregation of cross-transformer outputs in the final layer With the introduction of the three-branch mechanism in the CCAViT model. Although the influence of mid-level features was already demonstrated when comparing all implemented models, this experiment specifically focuses on analyzing the effect of mid-level contributions during the aggregation process. To explore potential performance improvements, the middle-branch coefficient has been examined in greater detail, serving as an example of the numerous tests performed related to architecture design.

Table 4.18 and Table 4.19 demonstrate that a middle branch coefficient of 1.0 yields the best performance. Similarly, the other two branches also use a coefficient of 1.0 during aggregation, effectively acting as if no additional multiplier were applied. To avoid the problem of exploding gradients in training, all values are normalized when the middle branch coefficient is adjusted. Consequently, applying an extra coefficient multiplier other than one for the CCAViT architecture does not result in superior performance.

Table 4.18. AUC of CCAViT across sub-datasets of FF++ versus middle branch coefficient

Middle branch coefficient	DF	F2F	FS	FSh	NT	AVG
0.5	99.90	99.20	99.70	99.40	98.90	99.42
1.0	100.00	99.30	99.90	99.70	99.00	99.58
1.5	99.80	99.20	99.80	99.60	98.90	99.46

Table 4.19. AUC of CCAViT in DFDC, DFD and CelebDF versus middle branch coefficient.

Middle branch coefficient	DFD	CelebDF	DFDC
0.5	93.30	78.10	57.50
1.0	93.80	80.20	64.00
1.5	92.70	79.90	63.80

All hyperparameters are fine-tuned to the best values using an approach similar to grid search. However, testing all possible combinations of multiple variables is impractical for large models such as CCAViT. Instead, we adjusted one value at a time, starting from the optimal values determined for CCViT model. Once all experiments were completed, we focused on refining parameters of the model around these optimal values.

Although hyperparameter tuning experiments yielded consistent results, the step size of parameter adjustments could be reduced, and wider intervals could be explored. This suggests that further optimization may still be possible for CCAViT. However, such experiments require significantly more time and some tests might result in diminishing returns. The complexity of the optimization process also highlights the need for more efficient methods or automated approaches to explore hyperparameter spaces. Furthermore, leveraging distributed computing resources could mitigate time and hardware constraints, enabling more extensive experimentation. Moreover, like batch size,

our experimentation is also constrained by hardware limitations. However, the results indicate that we have achieved near-optimal hyperparameter values for CCAViT model.

5. CONCLUSION

As digital media usage continues to expand in today's society, safeguarding the security and reliability of shared information has become critically important. Rapid progress in deepfake image and video generation technologies has resulted in the production of fake content that is nearly impossible for humans to distinguish. To avoid the potential for misuse, the development of highly accurate and robust deepfake detection methods becomes crucially necessary.

The results of our experiments reveal that current deepfake detection methods perform well on existing datasets. However, detecting fake content in real-world images requires methods with stronger generalization capabilities. A review of recent studies highlights the increasing prominence of vision transformers in deepfake detection. Furthermore, models addressing overfitting to identity information have demonstrated notable improvements in their ability to generalize. Despite these advancements, the effectiveness of such models remains largely confined to synthetic videos created using face-swapping techniques, underscoring the need for further innovation.

In this study, a new deepfake detection model is introduced, the Combinational Cross-Attention Vision Transformer (CCAViT). The model integrates the strengths of Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs). Additionally, we implement Convolutional Vision Transformer (CViT), Convolutional Cross Vision Transformer (CCViT), and parallel variants of convolutional vision transformer models. Various versions of CViT, CCViT, and CCAViT are trained and evaluated.

In the experiments, CCAViT demonstrates superior performance compared to the other implemented models in both intra-dataset and cross-dataset evaluations. This model leverages multiscale features extracted from different layers of EfficientNet using its multibranch structure. Unlike CCViT, CCAViT incorporates mid-level features, further enhancing its representational capacity. The combinational connections

in CCAViT enable the use of additional cross-attention modules per branch, contributing to its improved effectiveness.

This thesis offers researchers a comprehensive guideline for implementing convolutional vision transformer architectures in deepfake detection, addressing key challenges and obstacles encountered in the detection process. Through our experimental results, we provide insight into determining the optimal number of frames to extract from videos for both training and testing. Additionally, we introduce an efficient data preprocessing approach for the face detection phase, termed cascaded MTCNN.

Our comparative analysis of recent high-performing models demonstrates that CCAViT, even when using the smallest version of EfficientNet (EfficientNet-B0), performs competitively with state-of-the-art approaches. However, the substantial number of parameters in these models necessitates significant hardware resources and extended training durations.

Enhanced hardware capabilities would enable the proposed model to undergo additional training with minor architectural adjustments and more precise fine-tuning, potentially achieving even greater performance. The intra-dataset analysis of experimental results on FF++ indicates that most incorrect predictions are linked to videos where multiple faces appear in the frames. Consequently, future research could focus on several areas to enhance deepfake detection models. These include integrating a more advanced face detection system, refining data preprocessing strategies, or creating novel techniques specifically tailored for multiface scenarios.

REFERENCES

1. FakeApp, "FakeApp", 2019, https://www.malavida.com/en/soft/fakeapp/, accessed on Nov. 11, 2024.

2. X Corp., "Grok", 2023, https://x.ai/grok, accessed on Jan. 7, 2025.

3. Dosovitskiy, A., L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", *ArXiv*, Vol. abs/2010.11929, 2020.

4. Coccomini, D. A., N. Messina, C. Gennaro and F. Falchi, "Combining EfficientNet and Vision Transformers for Video Deepfake Detection", *International Conference on Image Analysis and Processing (ICIAP)*, pp. 219–229, Lecce, Italy, 2022.

5. Rossler, A., D. Cozzolino, L. Verdoliva, C. Riess, J. Thies and M. Nießner, "FaceForensics++: Learning to Detect Manipulated Facial Images", *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–11, Seoul, Korea, 2019.

6. Dolhansky, B., J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang and C. C. Ferrer, "The Deepfake Detection Challenge (DFDC) Dataset", *ArXiv preprint arXiv:2006.07397*, 2020.

7. Google, "Google Research Nick Dufour and Jigsaw Andrew Gully. Deep Fake Detection Dataset", 2019, https://ai.googleblog.com/2019/09/contributing-data-to-deepfake-detection.html, accessed on Nov. 11, 2024.

8. Li, Y., X. Yang, P. Sun, H. Qi and S. Lyu, "Celeb-df: A Large-Scale Challenging Dataset for Deepfake Forensics", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3207–3216, Seattle,

Washington, USA, 2020.

9. Rana, M. S., M. N. Nobi, B. Murali and A. H. Sung, "Deepfake Detection: A Systematic Literature Review", *IEEE Access*, Vol. 10, pp. 25494–25513, 2022.

10. Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, "Generative Adversarial Nets", *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems*, Vol. 27, pp. 2672–2680, Montreal, Quebec, Canada, 2014.

11. Zhu, J.-Y., T. Park, P. Isola and A. A. Efros, "Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks", *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, Venice, Italy, 2017.

12. Karras, T., S. Laine and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4396–4405, Long Beach, California, USA, 2019.

13. Choi, Y., M. Choi, M. Kim, J.-W. Ha, S. Kim and J. Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8789–8797, Salt Lake City, Utah, USA, 2018.

14. Li, Y., M.-C. Chang and S. Lyu, "In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking", *IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–7, Hong Kong, 2018.

15. Li, M., B. Liu, Y. Hu and Y. Wang, "Exposing Deepfake Videos by Tracking Eye Movements", *International Conference on Pattern Recognition (ICPR)*, pp. 5184–5189, Milan, Italy, 2021.

16. Haliassos, A., K. Vougioukas, S. Petridis and M. Pantic, "Lips Don't Lie: A Generalisable and Robust Approach To Face Forgery Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5039–5049, Nashville, Tennessee, USA, 2021.

17. Yang, X., Y. Li and S. Lyu, "Exposing Deep Fakes Using Inconsistent Head Poses", *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8261–8265, Brighton, United Kingdom, 2019.

18. Afchar, D., V. Nozick, J. Yamagishi and I. Echizen, "MesoNet: a Compact Facial Video Forgery Detection Network", *IEEE International Workshop on Information Forensics and Security (WIFS)*, p. 1–7, Hong Kong, 2018.

19. Lowe, D., "Object Recognition from Local Scale-Invariant Features", *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Vol. 2, pp. 1150–1157 vol.2, Corfu, Greece, 1999.

20. Pan, D., L. Sun, R. Wang, X. Zhang and R. O. Sinnott, "Deepfake Detection through Deep Learning", *IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT)*, pp. 134–143, Leicester, United Kingdom, 2020.

21. Nguyen, H. H., J. Yamagishi and I. Echizen, "Capsule-Forensics: Using Capsule Networks to Detect Forged Images and Videos", *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2307–2311, Calgary, Alberta, Canada, 2018.

22. Güera, D. and E. J. Delp, "Deepfake Video Detection Using Recurrent Neural Networks", *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6, Auckland, New Zealand, 2018.

23. Wang, Y. and A. Dantcheva, "A video is worth more than 1000 lies. Comparing

3DCNN approaches for detecting deepfakes", *IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 515–519, Buenos Aires, Argentina, 2020.

24. Zhao, C., C. Wang, G. Hu, H. Chen, C. Liu and J. Tang, "ISTVT: Interpretable Spatial-Temporal Video Transformer for Deepfake Detection", *IEEE Transactions on Information Forensics and Security*, Vol. 18, pp. 1335–1348, 2023.

25. Qian, Y., G. Yin, L. Sheng, Z. Chen and J. Shao, "Thinking in Frequency: Face Forgery Detection by Mining Frequency-Aware Clues", *European Conference on Computer Vision*, pp. 86–103, Glasgow, United Kingdom, 2020.

26. Luo, Y., Y. Zhang, J. Yan and W. Liu, "Generalizing Face Forgery Detection With High-Frequency Features", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16317–16326, Nashville, Tennessee, USA, 2021.

27. Miao, C., Z. Tan, Q. Chu, H. Liu, H. Hu and N. Yu, "F2Trans: High-Frequency Fine-Grained Transformer for Face Forgery Detection", *IEEE Transactions on Information Forensics and Security*, Vol. 18, pp. 1039–1051, 2023.

28. Patel, Y., S. Tanwar, R. Gupta, P. Bhattacharya, I. E. Davidson, R. Nyameko, S. Aluvala and V. Vimal, "Deepfake Generation and Detection: Case Study and Challenges", *IEEE Access*, Vol. 11, pp. 143296–143323, 2023.

29. Chintha, A., B. Thai, S. J. Sohrawardi, K. Bhatt, A. Hickerson, M. Wright and R. Ptucha, "Recurrent Convolutional Structures for Audio Spoof and Video Deepfake Detection", *IEEE Journal of Selected Topics in Signal Processing*, Vol. 14, No. 5, pp. 1024–1037, 2020.

30. Zhou, Y. and S.-N. Lim, "Joint Audio-Visual Deepfake Detection", *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14780–14789, Montreal,

Qerbec, Canada, 2021.

31. Wang, Y., C. Peng, D. Liu, N. Wang and X. Gao, "Spatial-Temporal Frequency Forgery Clue for Video Forgery Detection in VIS and NIR Scenario", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 33, No. 12, pp. 7943–7956, 2023.

32. Wu, X., Z. Xie, Y. Gao and Y. Xiao, "SSTNet: Detecting Manipulated Faces Through Spatial, Steganalysis and Temporal Features", *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2952–2956, Barcelona, Spain, 2020.

33. Tan, M. and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks", K. Chaudhuri and R. Salakhutdinov (Editors), *Proceedings of the International Conference on Machine Learning (ICML)*, Vol. 97, pp. 6105–6114, Long Beach, California, USA, 2019.

34. Bonettini, N., E. D. Cannas, S. Mandelli, L. Bondi, P. Bestagini and S. Tubaro, "Video Face Manipulation Detection Through Ensemble of CNNs", *International Conference on Pattern Recognition (ICPR)*, pp. 5012–5019, Milan, Italy, 2021.

35. Rana, M. S. and A. H. Sung, "DeepfakeStack: A Deep Ensemble-based Learning Technique for Deepfake Detection", *IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 70–75, New York City, New York, USA, 2020.

36. Dong, S., J. Wang, J. Liang, H. Fan and R. Ji, "Explaining Deepfake Detection by Analysing Image Matching", *European Conference on Computer Vision (ECCV)*, pp. 18–35, Tel Aviv, Israel, 2022.

37. Yan, Z., Y. Zhang, Y. Fan and B. Wu, "UCF: Uncovering Common Features for

Generalizable Deepfake Detection", *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22412–22423, Paris, France, 2023.

38. Huang, B., Z. Wang, J. Yang, J. Ai, Q. Zou, Q. Wang and D. Ye, "Implicit Identity Driven Deepfake Face Swapping Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4490–4499, Vancouver, Canada, 2023.

39. Dong, S., J. Wang, R. Ji, J. Liang, H. Fan and Z. Ge, "Implicit Identity Leakage: The Stumbling Block to Improving Deepfake Detection Generalization", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3994–4004, Vancouver, Canada, 2023.

40. Lin, L., X. He, Y. Ju, X. Wang, F. Ding and S. Hu, "Preserving Fairness Generalization in Deepfake Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16815–16825, Seattle, Washington, USA, 2024.

41. Khormali, A. and J.-S. Yuan, "DFDT: An End-to-End Deepfake Detection Framework Using Vision Transformer", *Applied Sciences*, Vol. 12, No. 6, p. 2953, 2022.

42. Gu, Z., T. Yao, Y. Chen, S. Ding and L. Ma, "Hierarchical Contrastive Inconsistency Learning for Deepfake Video Detection", *European Conference on Computer Vision (ECCV)*, pp. 596–613, Tel Aviv, Israel, 2022.

43. Xu, Y., J. Liang, G. Jia, Z. Yang, Y. Zhang and R. He, "TALL: Thumbnail Layout for Deepfake Video Detection", *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22658–22668, Paris, France, 2023.

44. Wang, Y., K. Yu, C. Chen, X. Hu and S. Peng, "Dynamic Graph Learning With Content-Guided Spatial-Frequency Relation Reasoning for Deepfake Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recog-*

nition (CVPR), pp. 7278–7287, Vancouver, Canada, 2023.

45. Le, B. M. and S. S. Woo, "Quality-Agnostic Deepfake Detection with Intra-model Collaborative Learning", *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22378–22389, Paris, France, 2023.

46. Yan, Z., Y. Luo, S. Lyu, Q. Liu and B. Wu, "Transcending Forgery Specificity with Latent Space Augmentation for Generalizable Deepfake Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8984–8994, Seattle, Washington, USA, 2024.

47. Choi, J., T. Kim, Y. Jeong, S. Baek and J. Choi, "Exploiting Style Latent Flows for Generalizing Deepfake Video Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1133–1143, Seattle, Washington, USA, 2024.

48. Dong, X., J. Bao, D. Chen, T. Zhang, W. Zhang, N. Yu, D. Chen, F. Wen and B. Guo, "Protecting Celebrities from Deepfake with Identity Consistency Transformer", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9468–9478, New Orleans, Louisiana, USA, 2022.

49. Touvron, H., M. Cord, M. Douze, F. Massa, A. Sablayrolles and H. Jegou, "Training Data-Efficient Image Transformers Distillation through Attention", *International Conference on Machine Learning (ICML)*, Vol. 139, pp. 10347–10357, Virtual Only, 2021.

50. Heo, Y.-J., Y.-J. Choi, Y.-W. Lee and B.-G. Kim, "Deepfake Detection Scheme Based on Vision Transformer and Distillation", *ArXiv preprint arXiv:2104.01353*, 2021.

51. Liu, Z., Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin and B. Guo, "Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows", *Proceed-*

ings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10012–10022, Montreal, Qerbec, Canada, 2021.

52. Chen, L., Y. Zhang, Y. Song, L. Liu and J. Wang, "Self-Supervised Learning of Adversarial Example: Towards Good Generalizations for Deepfake Detection", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18710–18719, New Orleans, Louisiana, USA, 2022.

53. Bello, I., B. Zoph, Q. Le, A. Vaswani and J. Shlens, "Attention Augmented Convolutional Networks", *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3285–3294, Seoul, Korea, 2019.

54. Zhao, H., J. Jia and V. Koltun, "Exploring Self-Attention for Image Recognition", *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10076–10085, Seattle, Washington, USA, 2020.

55. Chen, C.-F., Q. Fan, N. R. Mallinar, T. Sercu and R. S. Feris, "Big-Little Net: An Efficient Multi-Scale Feature Representation for Visual and Speech Recognition", *ArXiv*, Vol. abs/1807.03848, 2019.

56. Chen, C.-F. R., Q. Fan and R. Panda, "CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification", *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 347–356, Montreal, Qerbec, Canada, 2021.

57. Wodajo, D. and S. Atnafu, "Deepfake Video Detection Using Convolutional Vision Transformer", *ArXiv*, Vol. abs/2102.11126, 2021.

58. Wang, T., H. Cheng, K. P. Chow and L. Nie, "Deep Convolutional Pooling Transformer for Deepfake Detection", *ACM Transactions on Multimedia Computing, Communications and Applications*, Vol. 19, No. 6, 2023.

59. Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database", *IEEE Conference on Computer Vision and*

Pattern Recognition (CVPR), pp. 248–255, Miami, Florida, USA, 2009.

60. Deepfakes, "Deepfakes GitHub Repository: FaceSwap", https://github.com/deepfakes/faceswap, accessed on Jan. 7, 2025.

61. Thies, J., M. Zollhöfer, M. Stamminger, C. Theobalt and M. Nießner, "Face2Face: Real-Time Face Capture and Reenactment of RGB Videos", *Commun. ACM*, Vol. 62, No. 1, p. 96–104, 2018.

62. Kowalski, M., "FaceSwap GitHub Repository", 2021, https://github.com/MarekKowalski/FaceSwap/, accessed on on Nov. 11, 2024.

63. Li, L., J. Bao, H. Yang, D. Chen and F. Wen, "Advancing High Fidelity Identity Swapping for Forgery Detection", *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5073–5082, Seattle, Washington, USA, 2020.

64. Thies, J., M. Zollhöfer and M. Nießner, "Deferred Neural Rendering: Image Synthesis Using Neural Textures", *ACM Trans. Graph.*, Vol. 38, No. 4, 2019.

65. Zhang, K., Z. Zhang, Z. Li and Y. Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks", *IEEE Signal Processing Letters*, Vol. 23, No. 10, pp. 1499–1503, 2016.