applied sciences

MDPI

Article

GenConViT: Deepfake Video Detection Using Generative Convolutional Vision Transformer

Deressa Wodajo Deressa[1,*], Hannes Mareen [1], Peter Lambert [1], Solomon Atnafu [2], Zahid Akhtar [3] and Glenn Van Wallendael [1]

[1] IDLab, Department of Electronics and Information Systems, Ghent University—imec, 9000 Gent, Belgium; hannes.mareen@ugent.be (H.M.); peter.lambert@ugent.be (P.L.); glenn.vanwallendael@ugent.be (G.V.W.)
[2] Department of Computer Science, Addis Ababa University, Addis Ababa P.O. Box 1176, Ethiopia; solomon.atnafu@aau.edu.et
[3] Department of Network and Computer Security, State University of New York Polytechnic Institute, C135, Kunsela Hall, Utica, NY 13502, USA; akhtarz@sunypoly.edu
* Correspondence: deressawodajo.deressa@ugent.be

Abstract: Deepfakes have raised significant concerns due to their potential to spread false information and compromise the integrity of digital media. Current deepfake detection models often struggle to generalize across a diverse range of deepfake generation techniques and video content. In this work, we propose a Generative Convolutional Vision Transformer (GenConViT) for deepfake video detection. Our model combines ConvNeXt and Swin Transformer models for feature extraction, and it utilizes an Autoencoder and Variational Autoencoder to learn from latent data distributions. By learning from the visual artifacts and latent data distribution, GenConViT achieves an improved performance in detecting a wide range of deepfake videos. The model is trained and evaluated on DFDC, FF++, TM, DeepfakeTIMIT, and Celeb-DF (v2) datasets. The proposed GenConViT model demonstrates strong performance in deepfake video detection, achieving high accuracy across the tested datasets. While our model shows promising results in deepfake video detection by leveraging visual and latent features, we demonstrate that further work is needed to improve its generalizability when encountering out-of-distribution data. Our model provides an effective solution for identifying a wide range of fake videos while preserving the integrity of media.

Keywords: deep learning; deepfakes; deepfake detection; vision transformer; generative models

Academic Editor: Lianwei Wu

Received: 9 May 2025
Revised: 6 June 2025
Accepted: 10 June 2025
Published: 12 June 2025

Citation: Deressa, D.W.; Mareen, H.; Lambert, P.; Akhtar, S.A.Z.; Van Wallendael, G. . GenConViT: Deepfake Video Detection Using Generative Convolutional Vision Transformer. *Appl. Sci.* **2025**, *15*, 6622. https://doi.org/10.3390/app15126622

1. Introduction

Deepfakes are hyper-realistic manipulated media generated using various advanced deep learning (DL) techniques. Deepfake videos are produced by superimposing one person's facial features onto another person's face through techniques such as face replacement, facial reenactment, face editing, and complete face synthesis [1,2], among others. Recently, deepfakes have become a public concern due to their potential for misuse and the spread of false information [3–5]. Various deepfake creation methods are readily available for use, and anyone interested can use them to easily modify existing media, presenting a false representation of reality where individuals appear to be speaking or performing actions that never actually took place. Additionally, the manipulation of these videos is carried out on a frame-by-frame basis, making deepfakes seem more realistic and believable. Consequently, deepfake creation techniques have been used to manipulate images and

videos, causing celebrities and politicians to be depicted as saying or doing things that are untrue.

The applications of deepfake videos can range from creative uses such as creating realistic special effects or replacing an actor in the film and entertainment industry to potentially harmful attacks, such as using deepfakes to create misleading videos for criminal purposes [6,7]. In the realm of politics, deepfakes have been utilized to create fake political videos, war propaganda, and videos intended to influence elections. This has led to widespread concern about the potential for malicious actors to use deepfakes to spread false information, erode the credibility of political candidates, or even pose a threat to national security. The dissemination of misleading and false information through deepfakes can have real-life consequences, making it imperative to address the need for accurate and reliable deepfake video detection techniques.

Deepfake video detection attempts to detect whether a given video has been tampered with or manipulated using deepfake techniques. The challenge of detecting deepfakes has inspired researchers and technology companies to develop various deep learning methods to identify tampered videos [1,3,8–11]. Currently, deepfake detection methods often rely heavily on visual features [12–15]. Visual features-based detection can be on a frame-by-frame basis [16] or use the temporal relationship in the differences between frames over time [17,18]. However, these methods mostly encounter challenges in detecting deepfakes that differ from their training data, thus failing to detect deepfakes sufficiently. In recent years, the rapid progress made in advanced generative models like Generative Adversarial Networks (GANs) [19–21], Variational Autoencoders (VAEs) [22,23], and Diffusion Models (DMs) [24] has made it even more challenging to detect deepfakes based on visual artifacts alone, as deepfakes leave little to no trace of visual clues that help to distinguish them from images of the real world. Several authors have proposed alternative detection methods, such as utilizing biological signals [25], geometric features [26], frequency information [27], spatial features with temporal information [28,29] and generative adversarial network fingerprints [30], as potential solutions to the difficulties in detecting deepfakes. Another approach to spotting deepfake videos involves examining the consistency of pixels or groups of similar pixels [31]. Current deepfake video detection methods generate competitive results when it comes to identifying manipulated videos, but they often fail to generalize to more diverse videos, particularly in environments with different facial poses, light angles, and movements [32].

In an effort to provide more generalization, we propose a novel architecture called the Generative Convolutional Vision Transformer (GenConViT), aimed at detecting a diverse set of fake videos. Our proposed architecture leverages both visual artifacts and the latent data distribution of the data in its detection process.

GenConViT has two main components: a generative part and a feature extraction part. The generative part utilizes two different models (an Autoencoder (AE) [33] and VAE), to learn the latent data distribution of the training data. The feature extraction component employs ConvNeXt [34] and the Swin Transformer [35] to extract relevant visual features. Our proposed model addresses the limitations of current detection methods by identifying both the visual artifacts and the latent data distribution, making it capable of detecting a wider range of fake videos. Extensive experimental results demonstrate that GenConViT produces competitive results compared to other state-of-the-art deepfake detection models.

In summary, our paper makes the following contributions to this area of research:

- GenConViT: We propose the Generative Convolutional Vision Transformer for deepfake video detection. Our method utilizes an AE and VAE to generate images, extracts the visual features of the original and generated image using ConvNeXt and the

Swin Transformer, and then uses the extracted features to detect whether a video is a deepfake video.
- By training our model on more than 1 million images extracted from five large deepfake datasets and detecting both their visual and latent features, our proposed method aims to enhance its deepfake detection performance compared to that of previously proposed methods.
- We thoroughly evaluated GenConViT on more than 3972 videos collected from five deepfake video datasets, demonstrating its robust performance under diverse experimental settings.
- We have open-sourced our code and weights at https://github.com/erprogs/GenConViT (accessed on 9 June 2025).
- While our approach achieves a strong performance, our comprehensive ablation study (a leave-one-dataset-out study) indicates a potential limitation in the generalizability of deepfake detection methods, particularly when faced with out-of-distribution data. This is an area for future work.

The remainder of this article is structured as follows: An overview of existing studies is presented in Section 2. The proposed Generative Convolutional Vision Transformer (GenConViT) deepfake detection framework and the datasets used are detailed in Section 3. Experiments are discussed in Section 4, and an ablation study is performed in Section 4.4.2. Finally, the conclusions are drawn in Section 6.

2. Related Work

In recent years, deepfakes have become more realistic as well as becoming harder to detect. In this section, we discuss the various deep learning methods used to create (Section 2.1) and detect (Section 2.2) deepfakes.

2.1. Deepfake Generation

Deepfake videos are created using advanced DL methods, such as GANs, VAEs, and DMs. The term deepfake was first introduced in 2017 on the social media platform Reddit by a user called Deepfakes to showcase a DL technique to generate deepfake videos [6]. The videos and the accompanying code garnered significant attention, and soon people started to explore other ways to create a hyper-realistic video. Some widely used DL techniques to create deepfakes include face synthesis, face reenactment, face replacement, identity swapping, attribute manipulation, and expression swapping, to name a few. GANs are a type of generative model (GM) that consists of two neural networks (NNs); a generator G, which takes in random noise and produces synthetic data; and a discriminator D, which determines whether the input sample is real or fake. Another NN used to create deepfakes is the VAE, which consists of an Encoder and a Decoder NN. The Encoder encodes the input image into a lower-dimensional latent space, and the Decoder reconstructs an image from that latent space. Convolutional Neural Networks (CNNs) [36,37] are also used for face synthesis as they can learn the mapping between a face image and its attributes, such as its facial expression, age, and gender.

Face synthesis [38] is a method for synthesizing desired non-existent face images based on a given input image. In face synthesis, a generator G generates a face image, and a discriminator D discriminates whether the sample is real or fake. Face reenactment [39–41] is a face synthesis method that transfers source face attributes to a target face while preserving the appearance and identity of the target face's features. Face2Face [39] is a real-time face reenactment technique that animates the facial expressions of the target video by using a source actor, generating a manipulated output video. Some other examples of face synthesis methods include frontal view synthesis, changing the facial pose of an input

image, altering facial attributes, and aging the face to create diverse and realistic results. Face swapping [42–44] is the process of replacing a person's face in an image or video with the face of another person to create a non-existent realistic face.

Several types of GANs have been proposed for face synthesis, including Progressive GANs [45], Wasserstein GANs [46], and Style-Based GANs [20]. Progressive GANs allow for the generation of high-resolution images by gradually increasing the resolution of their generated images throughout the training process. Wasserstein GANs improve the stability of the GAN training process by using a different loss function. Style-Based GANs allow for the control of certain styled aspects of the generated faces, such as facial expression and hair style. Conditional GANs [21], also known as supervised GANs, use labeled data to generate facial semantics. Paired image-to-image translation GANs [47,48] are a type of conditional GAN that can translate an input image from one domain to another when given input–output pairs of images as training data. Pix2Pix is one example of a paired image-to-image translation GAN.

In conclusion, deepfake generation leverages a diverse range of deep learning methods, including GAN architectures (e.g., Progressive, Wasserstein, and Style-Based GANs), Transformers, VAEs, and CNNs. These methods are applied to create hyper-realistic images and videos through techniques such as face synthesis, reenactment, and face swapping. These rapid advancements and the diversity of approaches pose significant and evolving challenges for deepfake detection.

2.2. Deepfake Detection

The key point of the deepfake detection pipeline is determining whether a given video is fake or real. With the advancement of DL methods that can create hyper-realistic images, deepfake detection has become an increasingly challenging task. To this end, various authors have proposed deepfake detection techniques that use different approaches, including visual features, biological signals, and frequency information, to name a few [49,50].

Several deepfake detection methods rely on extracting visual features from manipulated videos. MesoNet [12] is a deepfake detection method that uses CNNs to extract mesoscopic properties and identify deepfakes created with techniques like Deepfake [51] and Face2Face [39]. Nguyen et al. [52] proposed a model that uses a combination of VGG-19 and capsule networks to learn complex hierarchical representations to detect various types of forgery, including FaceSwap [42], facial reenactment [39], replay attacks, and AI-generated videos. Yang et al. [13] proposed a model for comparing 3D head poses estimated from all facial landmarks. The method considers splicing synthesized face regions into original images to introduce errors in landmark locations. These landmark location errors can be detected by comparing the head poses estimated from the facial landmarks. Li and Lyu [53] proposed a CNN model for deepfake detection that identifies face-warping artifacts. Their method leverages the limitations of deepfake algorithms that generate face images of lower resolutions, which results in distinctive warping artifacts when the generated images are transformed to match the original faces in the deepfake creation pipeline. By comparing the deepfake face's region with surrounding pixels, resolution inconsistencies caused by face warping can be identified. Li et al. [54] proposed a technique called Face X-ray that detects the blending boundaries of images and reveals whether an input face image has been manipulated by blending two images from different sources. Sun et al. [55] proposed a virtual-anchor-based approach that robustly extracts the facial trajectory, capturing displacement information. They constructed a network utilizing dual-stream spatial–temporal graph attention and a gated recurrent unit backbone to expose manipulated videos. The proposed method achieves competitive results on the FaceForensics++ dataset, demonstrating its effectiveness in detecting manipulated videos.

Overall, these diverse methods of detecting deepfakes based on visual cues demonstrate that subtle visual inconsistencies and artifacts can be used to expose manipulated videos.

In addition to visual features, researchers have explored the use of biological signals and low-level features for deepfake detection. Y. Li [56] proposed a model that combines a CNN and a recurrent neural network (LRCN) to detect deepfake videos by tracking eye blinking using previous temporal knowledge. Chintha et al. [57] proposed a modified XceptionNet architecture, which uses visual frames, edge maps, and dense optical flow maps alongside RGB channel data to target low-level features. The architecture isolates deepfakes at the instance and video levels, making the technique effective in detecting deepfakes. Zhao et al. [58] proposed pair-wise self-consistency learning with an inconsistency image generator to train a ConvNet [34] that extracts local source features and measures their self-consistency to identify deepfakes.

D. Kim and K. Kim [29] proposed a facial forensic framework that uses pixel-level color features in the edge region of an image and a 3D-CNN classification model to interpret the extracted color features spatially and temporally for generalized and robust face manipulation detection. Sabir et al. [59] proposed a Recurrent convolutional model, whereas [60] proposed a Convolutional Vision Transformer (CViT) model. Y. Heo [16] introduced an improved Vision Transformer model with a vector-concatenated CNN feature and patch-based positioning, while [61] used a Vision Transformer with distillation and Coccomini et al. [62] combined EfficientNet with Vision Transformers (ViTs) to detect deepfakes. Hybrid CNN–Transformer approaches excel by extracting local and global features for robust and strong deepfake detection.

Chen et al. [63] proposed MCDM, a masked diffusion augmenter that reconstructs occluded facial regions to generate diverse, high-fidelity deepfakes. Their method's combined pixel and feature losses significantly improved deepfake detection. Bhattacharyya et al. [64] introduced two large diffusion-based deepfake datasets (DiffusionDB-Face and JourneyDB-Face) for deepfake detection. They showed that training on heterogeneous deepfake sources alone is insufficient and proposed a novel generic training strategy called momentum difficulty boosting to improve the performance of deepfake detection. Sun et al. [65] proposed DiffusionFake, which uses a frozen Stable Diffusion model to force detectors to disentangle source and target cues, achieving significantly better cross-domain deepfake detection with no extra inference cost. Xu et al. [66] proposed a Thumbnail Layout (TALL) method which transforms four consecutive video frames into 2x2 grid layout to capture spatio-temporal deepfake inconsistencies. By embedding temporal information into a single image input, TALL enables a standard vision backbone to learn spatio-temporal artifacts efficiently.

Le and Woo [67] proposed the quality-agnostic deepfake detection method (QAD) to handle instance-based intra-model collaborative learning. The authors used the Hilbert–Schmidt Independence Criterion (HSIC) to maximize the geometrical similarity between intermediate representations of high- and low-quality deepfakes, and Adversarial Weight Perturbation (AWP) to make the QAD model robust under varying input compressions. Li et al. [68] proposed FreqBlender to create pseudo-fakes by blending specific frequency knowledge. The authors used a novel Frequency Parsing Network to adaptively partition the frequency domain into three components corresponding to semantic information, structural information, and noise information, respectively.

Despite these promising advances, current detection methods struggle to generalize across diverse datasets. Table 1 lists the datasets used by existing deepfake detection models for training and evaluation. They often consider only a few datasets and hence generative methods are often used for training and evaluation. In the next section, we introduce our approach, which addresses this problem.

Table 1. Datasets used in selected deepfake detection methods.

Paper	Dataset(s) for Training and Evaluation
Image+Video Fusion [18]	DFDC
Selim EfficientNet B7 [69]	DFDC
CViT [60]	DFDC
Random cut-out [70]	DFDC (with face cut-out augmentation)
STDT [28]	FF++, Celeb-DF
ViT with distillation [61]	DFDC
Heo et al. [16]	DFDC
Coccomini et al. [62]	FF++, DFDC
Li et al. [54]	FF++, CelebA-HQ (FaceShifter GAN)
GenConViT (ours)	FF++, DFDC, TM, Celeb-DF (v2), TIMIT

3. Proposed Deepfake Detection Method

In this section, we propose a deepfake detection framework based on a Generative Convolutional Vision Transformer (GenConViT), which we introduce for the first time.

The proposed Generative Convolutional Vision Transformer model transforms the input facial images to latent spaces and extracts visual clues and hidden patterns from within them to determine whether a video is real or fake. The proposed GenConViT model is shown in Figure 1. It has two independently trained networks (A and B) and four main modules: an Autoencoder (AE), a Variational Autoencoder (VAE), a ConvNeXt layer, and a Swin Transformer. The first network (A) includes an AE, a ConvNeXt layer, and a Swin Transformer, while the second network (B) includes a VAE, a ConvNeXt layer, and a Swin Transformer. The first network uses an AE to transform images to a Latent Feature (LF) space, maximizing the model's class prediction probability, indicating the likelihood that a given input is a deepfake. The second network uses a VAE to maximize the probability of correct class prediction and minimize the reconstruction loss between the sample input image and the reconstructed image. Both AE and VAE models extract LFs from the input facial images (extracted from video frames), which capture hidden patterns and correlations present in the learned deepfake visual artifacts. The ConvNeXt and Swin Transformer models form a novel hybrid model: ConvNeXt-Swin. The ConvNeXt model acts as the backbone of the hybrid model, using a CNN to extract features from the input images. The Swin Transformer, with its hierarchical feature representation and attention mechanism, further extracts the global and local features of the input. The two networks each have two ConvNeXt-Swin models, which both take a 224×224 RGB image as their input, as well as an LF extracted by either the AE (I_A) or the VAE (I_B). The use of the ConvNeXt-Swin hybrid model enables the learning of relationships among the LFs extracted by the AE and VAE.

3.1. Autoencoder and Variational Autoencoder

An AE and a VAE consist of two networks: an Encoder and a Decoder. The Encoder of the AE maps an input image $X \in \mathbb{R}^{H \times W \times C}$ to a latent space $Z \in \mathbb{R}^{H' \times W' \times K}$, where K is the number of channels (features) in the output and H' and W' are the height and width of the output feature map, respectively. The Decoder of the AE maps the latent space $Z \in \mathbb{R}^{H' \times W' \times K}$ to an output image $X' \in \mathbb{R}^{H \times W \times C}$. The Encoder of the AE is composed of five convolutional layers with width starting from 3 and expanding up to 256, with kernels of size 3×3 and a stride of 2. Each convolutional layer is followed by ReLU non-linearity and Max pooling of kernel size 2×2 and stride 2. The output of the Encoder is a $256 \times 7 \times 7$ down-sampled LF. The Decoder is composed of five transposed convolutional layers with a width starting at 256 and ending at 3, with kernels of size 2×2 and stride of 2. Each transposed convolutional layer is followed by ReLU non-linearity. The output

of the Decoder, I_A, is a reconstructed feature space of the input image with dimensions $H \times W \times C$. In this case, I_A has dimensions $224 \times 224 \times 3$. The details of this configuration are shown in Table 2.

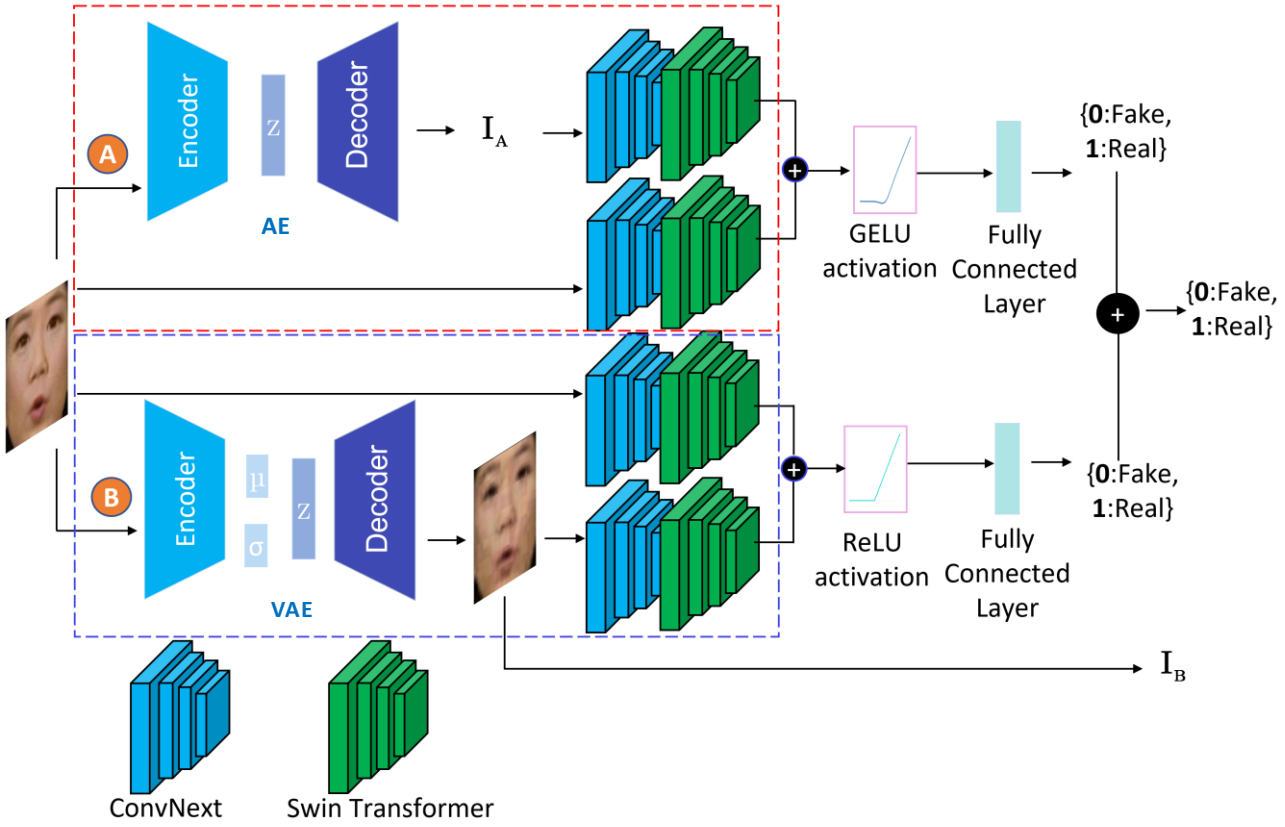


Figure 1. The proposed GenConViT deepfake detection framework.

The goal of the VAE is to learn a meaningful latent representation of the input image and reconstruct the input image by performing random sampling of the latent space while minimizing the reconstruction loss. The Encoder of the VAE maps an input image X to a probability distribution over a latent space $Z \in \mathbb{R}^K, Z \sim \mathcal{N}(\mu, \sigma^2)$, in which μ and σ^2 are the mean and variance of the learned distribution, respectively.

The Encoder of the VAE is composed of four convolutional layers with widths starting from 3 and expanding up to 128, with kernels of size 3×3 and a stride of 2. Each convolutional layer is followed by Batch Normalization (BN) and LeakyReLU non-linearity. The use of Batch Normalization also serves as a regularizer, mitigating the risk of overfitting during training. The Decoder is composed of four transposed convolutional layers with widths starting from 256 and reducing to 3, with kernels of size 2×2 and a stride of 2. Each transposed convolutional layer is followed by LeakyReLU non-linearity. The output of the Decoder, I_B, is an image reconstructed from the input image with the dimensions $H/2 \times W/2 \times C$. In this case, I_B has dimensions $112 \times 112 \times 3$. The details of this configuration are shown in Table 3. The choice of convolutional layers for both the AE and VAE is due to the computing power and memory we had, model accuracy, the extensive experiment, and the training time. To find the optimal hyperparameters for both the AE and VAE components, we experimented with latent sizes ranging from 8 to 1024 (i.e., 8, 64, 128, 256, 512, 1024) and various downsampling widths. We settled on 128 latent channels and 5 downsampling layers, since that setup minimized both the validation MSE and Cross-Entropy loss.

Table 2. Configuration of GenConViT model's Autoencoder.

Network	AE Configuration								
Encoder	Conv					Kernel	Stride	Activation	Normalization
	3-16	16-32	32-64	64-128	128-256	3	1	ReLU	-
Decoder	ConvTranspose								
	256-128	128-64	64-32	32-16	16-3	2	1	ReLU	-

Table 3. Configuration of GenConViT model's Variational Autoencoder.

Network	VAE Configuration							
Encoder	Conv				Kernel	Stride	Activation	Normalization
	3-16	16-32	32-64	64-128	3	2	LeakyReLU	BatchNorm2d
Decoder	ConvTranspose							
	256-64	64-32	32-16	16-3	2	2	LeakyReLU	-

3.2. ConvNeXt-Swin Hybrid

The ConvNeXt-Swin Transformer architecture is a hybrid CNN–Transformer model that combines the strengths of ConvNeXt [34] and Swin Transformer [35] architectures for deepfake detection tasks. The ConvNeXt model is a CNN architecture that has shown an impressive performance in image recognition tasks by extracting high-level features from images through a series of convolutional layers. The Swin Transformer is a transformer-based model that uses a self-attention mechanism to extract both local and global features.

The GenConViT model leverages the strengths of both architectures by using ConvNeXt as the backbone for feature extraction and the Swin Transformer for feature processing. In our proposed method, the ConvNeXt architecture extracts high-level features from images, which are then passed through a HybridEmbed module to embed the features into a compact and informative vector. The resulting vector is then passed to the Swin Transformer model. The ConvNeXt backbone consists of multiple convolutional layers that extract high-level features from the input images and the LFs from the AE or VAE. We used pre-trained ConvNeXt and Swin Transformer models, which were trained on an ImageNet dataset.

After extracting learnable features using the ConvNeXt backbone, we pass the feature maps through the HybridEmbed module. The HybridEmbed module is designed to extract feature maps from the ConvNeXt, flatten them, and project them to an embedding dimension of 768. It consists of a 1×1 convolutional layer which takes the feature maps from the backbone and reduces their channel dimension to the desired embedding dimension. The resulting feature maps are flattened and transposed to obtain a sequence of feature vectors, which are then further processed by the Swin Transformer.

The GenConViT's network A consists of two Hybrid ConvNeXt-Swin models that take in an LF of size $224 \times 224 \times 3$ generated by the AE (I_A) and an input image of the same size. The models output a feature space of size 1000, which is then concatenated. A linear mapping layer of 2 then transforms this combined feature vector into a class prediction, corresponding to the probabilities of the real and fake classes. Network B has the same configuration as A, but it uses the VAE and outputs both a class prediction probability and a reconstructed image of $224 \times 224 \times 3$ in size. Finally, the predictions of network A and network B are averaged to obtain the final real/fake prediction.

In summary, our proposed GenConViT method introduces a hybrid architecture for deepfake detection, combining the feature extraction capabilities of the pre-trained ConvNeXt-Swin architecture with generative components that use an AE and VAE. This setup helps our model to capture subtle inconsistencies in deepfake videos. GenConViT has

a total of 695M trainable parameters and requires approximately 6.855 GFLOPs for a single forward pass on a 224 × 224 px input image, and its average per-frame inference time is 0.10 s (NVIDIA RTX 4090 (Santa Clara, CA, USA)). The open-source code for GenConViT is available at https://github.com/erprogs/GenConViT (accessed on 9 June 2025).

4. Evaluation

We conducted extensive experiments on various configurations of the AE and VAE, as well as different variants of the CNN and Transformer models. Our findings suggest that a hybrid architecture using ConvNeXt and the Swin Transformer performs well. We first describe the experimental setup in Section 4.1. Then, the results are presented and discussed in Section 4.2, and a comparison with the state of the art is carried out in Section 4.3. Finally, the limitations of our method are analyzed in Section 4.4.

4.1. Experimental Setup

To assess GenConViT's performance, we used multiple evaluation metrics, including classification accuracy, F1 score, the Receiver Operating Characteristic (ROC) curve, and Area Under the Curve (AUC) values.

The implementation details, datasets, and preprocessing methodology are described in Sections 4.1.1, 4.1.2, and 4.1.3, respectively.

4.1.1. Implementation Details

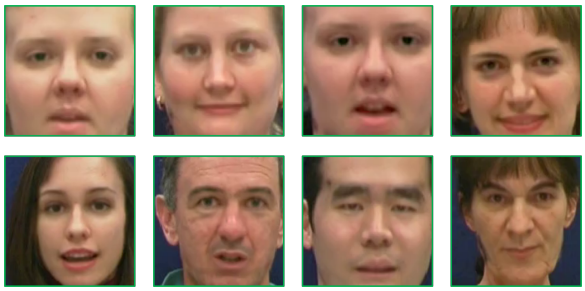
Networks *A* and *B* were trained to classify real and fake videos, while *B* was additionally trained to reconstruct the images, some examples of which are shown in Figure 2.

Therefore, network *A* was trained using the Cross-Entropy loss, while network *B* was trained using the Cross-Entropy and MSE losses. Network A is optimized for real vs. fake classification, so we used Cross-Entropy loss. Because we were focusing on classification objectives, we employed a larger batch size (32) for training. Network B is optimized for two goals: (i) real vs. fake classification and (ii) the reconstruction of each 224 × 224 input image. To enforce high reconstruction quality, we added an MSE term to the same Cross-Entropy objective for the reconstructed images. Since using MSE increases memory usage, we used a batch size of 16 for training. By using both losses, we encouraged the encoder to learn representations that both discriminate between real and fake images and minimize pixel-level reconstruction errors, as such by capturing better latent artifacts. We used the *timm* [71] library to load the class definitions and the weights of the pretrained ConvNeXt and Swin Transformer. Due to our limited resources and large training dataset, we implemented the "tiny" model versions of both architectures, namely `convnext_tiny` and `swin_tiny_patch4_window7_224`, both of which are trained on ImageNet-1k.

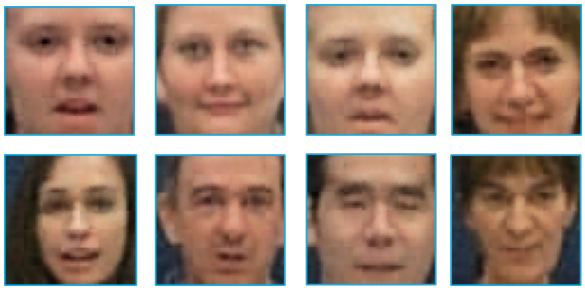
Both networks *A* and *B* were trained using the Adam optimizer with a learning rate of 0.0001 and weight decay of 0.0001. The Albumentation [72] library was used for data augmentation, and the following augmentation techniques were used with a strong augmentation rate of 90%: RandomRotate, Transpose, HorizontalFlip, VerticalFlip, GaussNoise, ShiftScaleRotate, CLAHE, Sharpen, IAAEmboss, RandomBrightnessContrast, and HueSaturationValue. The training data was normalized. The batch size for network *A* was set to 32 and for network *B* it was set to 16. Both networks were trained for 30 epochs.

4.1.2. Datasets

In our work, we utilized five datasets to train, validate, and test our model: DFDC [73,74], TrustedMedia (TM) [75], DeepfakeTIMIT (TIMIT) [76,77], Celeb-DF (v2) [78,79], and FaceForensics++ (FF++) [80]. DFDC and FF++ are well-known benchmark datasets for deepfake detection. TM is created using a diverse range of deepfake manipulation techniques.



(**a**) Input sample images.



(**b**) Corresponding images generated by network B.

Figure 2. Generated images (I_B): (**a**) input samples, (**b**) reconstruction from network B.

The DFDC dataset is the largest publicly available dataset and contains over 100,000 high-resolution real and fake videos. The dataset was created using 3426 volunteers, and the videos were captured in various natural settings, at different angles, and under different lighting conditions. The dataset was created using eight deepfake creation techniques.

The FF++ dataset comprises 1000 original videos collected from YouTube, which have been manipulated using four automated face manipulation methods: Deepfakes, Face2Face, FaceSwap, and NeuralTextures [81]. The dataset includes compressed videos with quantization parameters of $c23$ and $c40$ and various video resolutions. The TM dataset consists of 4380 fake and 2563 real videos, with multiple video and audio manipulation techniques used. The TM dataset is used only in the training phase. The Celeb-DF (v2) dataset consists of 890 real videos and 5639 deepfake videos.

We randomly extracted approximately 30 frames per video from each dataset to ensure diversity in our training data. To mitigate the different ratios between fake and real videos in the DFDC and TM datasets, we extracted a higher number of frames from their real videos. The DFDC dataset has a ratio of 6:1 for fake to real videos, and TM has approximately a 2:1 ratio.

By using a variety of datasets that utilize multiple deepfake creation techniques and videos captured in multiple settings, we aim to enable better generalization and robustness to varying environments. Notably, our model is trained (and evaluated) using significantly more datasets than those used for previous models (see Table 1).

4.1.3. Video Preprocessing

We preprocessed the frames of the videos in the datasets so that we could work with images that only contained information about faces and were correctly labeled. The preprocessing component in DL plays a critical role in preparing raw datasets for training, validation, and testing. The proposed model focuses on the face region, which is crucial in deepfake generation and synthesis mechanisms. We therefore preprocessed the videos using a series of image processing operations. These operations included the following steps:

1. Extracting the face region from each video using the OpenCV, face_recognition [82], and BlazeFace [83] face recognition deep learning libraries;
2. Resizing the input (facial) image to a 224 × 224 RGB format, where the dimensions of the input image are $H \times W \times C$, with $H = 224$ representing the height, $W = 224$ representing the width, and $C = 3$ representing the RGB channels;
3. Verifying the quality of extracted face-region images manually (only for the DFDC dataset).

After the face regions were extracted, we manually reviewed DFDC image frames to fix two problems. As noted in [74], (1) DFDC deepfake videos may contain pristine frames and (2) face regions may not always be accurately detected by the face recognition frameworks used to extract them, leading to some frames in the training dataset containing no faces [60]. To address this issue, we manually reviewed the images and excluded images that did not contain a face or were deemed to be a real image within the fake class. This approach allowed us to curate a fake class dataset comprising only relevant and potentially manipulated face images. In the future, we could opt for using automated methods rather than achieving this manually. For example, to filter real from fake frames within a deepfake video, an initial model could be trained on a small, manually filtered subset of images and could then classify the remaining subset into real and deepfake images. This classification could significantly speed up the manual verification process.

As a result of applying the preprocessing steps to the datasets, we collected a total of 1,004,810 images. To train, validate, and test our model, the images were divided into a ratio of approximately 80:15:5, resulting in 826,756 images for training, 130,948 images for validation, and 47,106 images for testing. Note that this preprocessed test set was only used for internal evaluation; the main experiments conducted in this paper were performed on videos (in contrast to individual frames). Namely, for our evaluation, we held back 3972 videos from the DFDC, DeepfakeTIMIT, Celeb-DF (v2), and FF++ datasets for testing. To create a single prediction per video, we extracted 15 frames from each video and averaged the resulting predictions.

4.2. Experimental Results and Discussion

In this section, we present the experimental results and discuss the performance of our proposed GenConViT model.

Table 4 summarizes the accuracy of the proposed GenConViT model, as well as its internal networks A and B, on the evaluation datasets. The table also shows its accuracy for the real and fake samples separately, to show potential discrepancies in its performance. Note that the TIMIT dataset does not contain any real videos.

The results demonstrate that GenConViT delivers strong performance across various datasets. The individual networks A and B also both demonstrate decent performances, but are outperformed by their combination. While networks A and B individually achieve 94% accuracy on average, the full GenConViT ensemble improves upon the average accuracy of network A by 2.2% and network B by 1.79%, indicating the benefit of using both models for detection. From Table 4, we can also see that the model has an average 5% increase in accuracy on the Celeb-DF (v2) dataset. Additionally, its measured performance is similar for both real and fake videos, except for Celeb-DF (v2), for which it has a worse accuracy in detecting real videos compared to fake videos.

For completeness, Tables 5 and 6 present the AUC value and F1 score, respectively. To further investigate GenConViT's performance, Figure 3 presents the resulting ROC curve. Both networks A and B lead to relatively similar results.

Overall, the proposed GenConViT model has an average accuracy of 95.8% and an AUC value of 99.3% across the tested datasets. These results highlight our model's

robust performance in detecting deepfake videos, demonstrating its potential for practical applications in the field.

Table 4. Comparison of accuracy values (%) of GenConViT and its internal networks across multiple evaluation datasets.

Dataset	GenConViT			GenConViT A			GenConViT B		
	ALL	REAL	FAKE	ALL	REAL	FAKE	ALL	REAL	FAKE
DFDC	98.50	98.70	98.45	97.50	98.70	97.20	98.45	98.70	95.52
FF++	97.00	95.58	98.50	95.57	94.12	95.56	96.80	95.58	98.02
TIMIT	98.28	-	98.28	97.65	-	97.50	97.81	-	97.80
Celeb-DF (v2)	90.94	83.00	98.80	85.42	70.22	93.38	83.97	55.00	99.38
Average	96.05	92.42	98.50	94.03	87.68	95.91	94.26	83.09	97.68

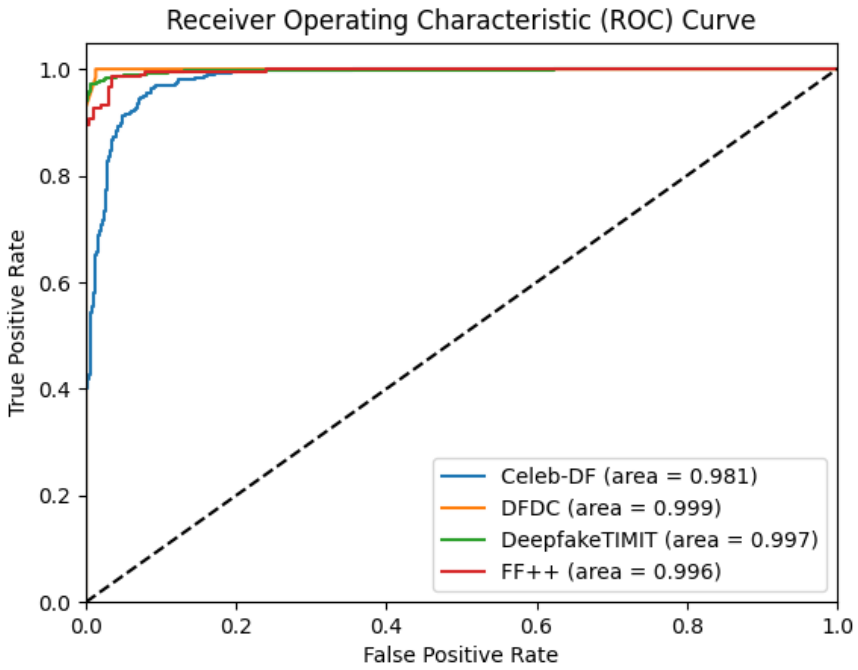


Figure 3. ROC curve illustrating the model's ability to discriminate between the real and fake classes.

Table 5. AUC values (%) for performance of GenConViT, GenConViT(*A*) and GenConViT(*B*) on each dataset.

Dataset	GenConViT	GenConViT(*A*)	GenConViT(*B*)
DFDC	99.9	99.9	99.9
FF++	99.6	99.1	99.6
Celeb-DF (v2)	98.1	97.8	94.1

Table 6. F1 scores (%) for performance of GenConViT, GenConViT(*A*) and GenConViT(*B*) on each dataset.

Dataset	GenConViT	GenConViT(*A*)	GenConViT(*B*)
DFDC	99.1	98.4	98.4
FF++	95.5	94.9	96.8
TIMIT	98.3	97.5	97.8
Celeb-DF (v2)	91.6	95.2	89.0

4.3. Comparison with State of the Art

We compared our GenConViT model with several other state-of-the-art methods on the DFDC dataset (Table 7), the FF++ dataset (Table 8), and the three subsets of FF++ (Table 9).

It is worth noting that the testing subdatasets of the DFDC and FF++ are usually not shared in the scientific literature. Hence, the exact datasets evaluated may differ, although comparing the reported values gives an idea of how their performances compare. Table 1 lists the datasets used by other deepfake detection models for training and evaluation. Note again that Table 1 shows that our dataset is trained and evaluated using the most datasets.

From Table 7 (DFDC dataset), we can observe that previous approaches achieved accuracy values ranging from 91.5% to 98.24% on the DFDC dataset. Notable models included Khan [18] (91.69%), Thing [84] (92.02%), and Seferbekov [69] (97.2%). Some models excelled in additional metrics, such as the STDT [28] model, with its 97.44% accuracy, 99.1% AUC, and 98.48% F1 score. In comparison to previous approaches, our proposed model demonstrated an excellent performance, achieving an average accuracy of 98.5%, an AUC of 99.9%, and an F1 score of 99.1%.

Table 7. GenConViT model's evaluation metrics on the **DFDC** dataset compared with those of other state-of-the-art models. (Bold is better).

Model	Accuracy (%)	AUC (%)	F1 Score (%)
Image+Video Fusion [18]	91.69	-	-
Selim EfficientNet B7 [69]	97.20	90.60	-
CViT [60]	91.50	91.00	-
Thing [84]	92.02	97.61	-
Random cut-out [70]	98.24	-	-
STDT [28]	97.44	99.10	98.48
ViT with distillation [61]	-	97.80	91.90
Heo et al. [16]	-	97.80	91.90
Coccomini et al. [62]	-	95.10	88.00
GenConViT (ours)	**98.50**	**99.90**	**99.10**

Table 8. GenConViT model's accuracy and AUC on the FF++ dataset compared with those of other state-of-the-art models.

Model	Accuracy (%)	AUC (%)	F1 Score (%)
Li et al. [54]	**97.73**	98.52	-
Image+Video Fusion [18]	99.52	99.64	**99.28**
Random cut-out [70]	97.00	99.28	-
MCDM [63]	-	99.31	-
MDB [64]	95.00	94.00	-
TALL-Swin [66]	98.65	99.97	-
GenConViT (ours)	97.00	**99.60**	97.1

Table 9. GenConViT model's accuracy (%) on the three subsets of the FF++ dataset compared with that of other state-of-the-art models.

Model	Deepfakes	Face2Face	NeuralTextures
Li et al. [54]	**99.17**	97.73	-
Coccomini [62]	87	-	69
Random cut-out [70]	**98.57**	98.57	90.71
QAD-E [67]	99.46	98.30	92.25
FreqBlender [68]	99.18	96.76	90.88
GenConViT (ours)	92.27	98.00	**97.00**

4.4. Limitation Analysis

This subsection analyzes two aspects of the proposed method: the consistency of its performance on the datasets (in Section 4.4.1) and its generalization through an out-of-distribution ablation study (in Section 4.4.2). The mitigation strategies used are subsequently described in Section 5.

4.4.1. Lower Performance on Celeb-DF v2: Feature Visualization

Table 4 demonstrates that our model's performance on the Celeb-DF (v2) dataset is lower than its performance on the other datasets. To further analyze this, Figure 4 shows the t-SNE visualization of our model's VAE's latent embeddings on three datasets (Celeb-DF (v2), DFDC, and FF++). We observe a clustering based on the dataset source (i.e., in green, red and blue). Within these clusters, we observe that the separation between real vs. fake clusters is not always strong and they heavily overlap. Specifically, fake and real Celeb-DF (v2) embeddings overlap more than the fake and real embeddings from DFDC or FF++. This may explain the model's lower performance on the Celeb-DF (v2) dataset. Section 5 discusses future strategies to improve its performance.

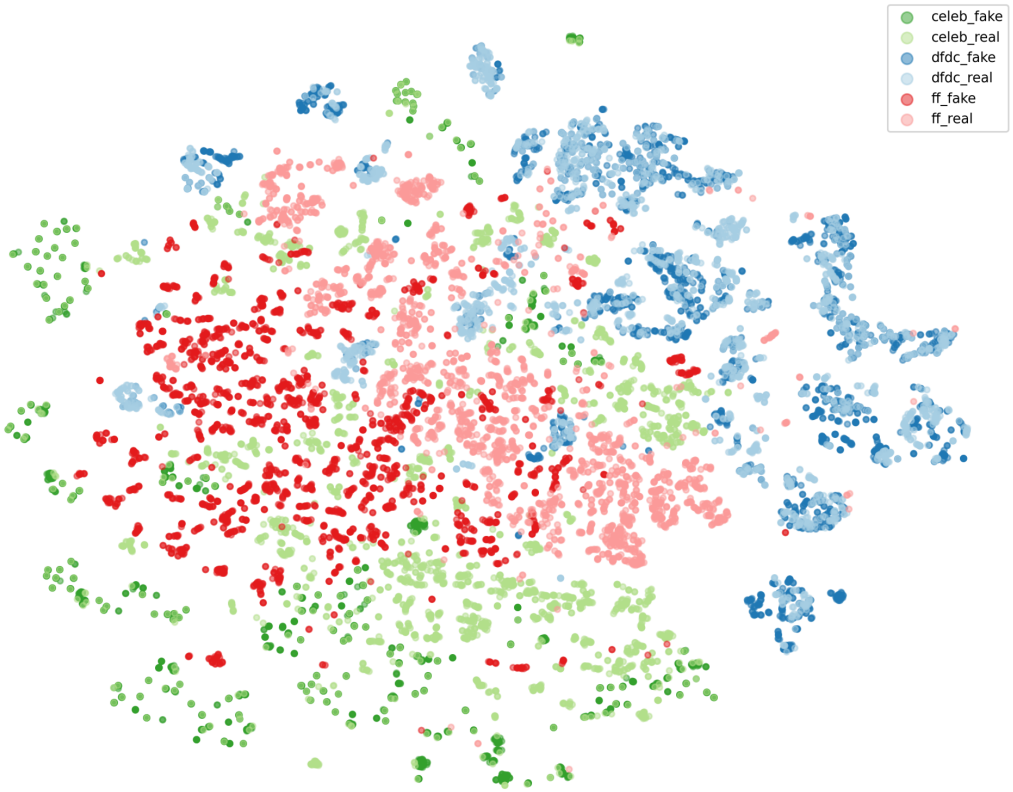


Figure 4. t-SNE visualization of samples from three deepfake datasets.

4.4.2. Generalization: Out-of-Distribution Ablation Study

To evaluate the generalization capability of our proposed model, we conducted an out-of-distribution (OOD) ablation study. We trained our model on four of the five available datasets, holding back the fifth dataset entirely (it was unseen during training and validation). We repeated this procedure using different held-back datasets and systematically varied the hyperparameters (such as the layers of the CNN depth, width, and learning rates) to select the best performing model variant. Moreover, to optimize computation, we performed training for ablation experiments on a randomly selected subset of approximately 5% of the images from the training datasets. Note that other datasets were not evaluated against out-of-domain deepfakes, hence their potential limitations may not have been actively exposed.

Table 10 summarizes the performance of the model across various scenarios. In each scenario, the column labeled "Held- out" denotes the dataset excluded from training, whereas the column labeled "Test set" denotes the dataset used for the evaluation. Then, the other columns present the model's performance on the test set when training used all test

datasets except for the held-back dataset. Cells highlighted in bold text represent significant drops in performance due to holding back the corresponding dataset during training.

We found that our model struggled to detect fake videos in the held-back dataset in an OOD setting, indicating that it still faces challenges in generalizing to unseen, more hyper-realistic deepfake images. Notably, when the model encounters a fully unseen hyper-realistic dataset, it struggles to detect fake videos, resulting in a substantial drop in accuracy in the fake class. Most notably, in the Celeb-DF (v2) hold-out scenario, the model's overall accuracy on Celeb-DF (v2) was as low as 55.67%, with an 11.56% accuracy on fake samples. We also note that the full model had the most difficulties with Celeb-DF (v2), as discussed in Section 4.4.1, which may explain its particularly low accuracy in the ablation study.

These results suggest that although our model performs well on in- domain data (TM, DeepfakeTIMIT, DFDC, and FF++), it faces challenges in generalizing to significantly different or higher-fidelity deepfakes. These experiments suggest that deeper or wider CNN architectures alone do not necessarily guarantee improved robustness to domain shifts. Despite experimenting with various hyperparameters (e.g., layer depth, width, and learning rates), the drop in fake detection accuracy remained significant when the held-back dataset was substantially different from the training sets.

Another noteworthy observation is that certain datasets, like DeepfakeTIMIT, appear easier to generalize to. This may be due to their relatively simpler manipulations. In contrast, datasets such as Celeb-DF (v2) or DFDC contain higher-quality forgeries and more varied manipulations, creating a more challenging OOD scenario. Consequently, our study underscores the importance of curating diverse, high-fidelity training sets when aiming to build robust deepfake detection models. Therefore, our proposed GenConViT model was trained on five datasets that represented a large variety of deepfakes and settings.

We have identified several possible reasons for the lower accuracy results in our OOD ablation study:

- Overfitting to Known Artifacts: Even with latent AE/VAE encoding, GenConViT's ConvNeXtSwin backbone may rely on domain-specific cues that do not transfer to unseen forgeries. As different datasets contain different generators, the artifacts from certain generators may not be representative of those from other generators.
- More Realistic Deepfake Generation: Datasets such as Celeb-DF (v2) use advanced deepfake generation methods which are difficult to detect. For example, Celeb-DF (v2), DFDC, and FF++ have subtler boundaries and lower noise-level residuals than the TIMIT dataset, as well as fewer warping artifacts. This makes it particularly hard for our leave-one-out model to spot these deepfake features when they were not seen during training.
- Ablation Sampling: Note that ablation is performed on randomly selected subsets of our training datasets without replacement (10k samples for each dataset). In contrast, the full model is trained on more than 1M images. Hence, although we deem the training on subsets to be representative enough to analyze the model's generalization performance, it must be carefully compared to the performance of the full model.
- Lack of Temporal Information: GenConViT relies on each frame independently and has no way of processing temporal information. Temporal artifacts may contain more generalizable features.

Overall, these findings confirm that while our model achieves a promising performance on in-domain data, it remains sensitive to domain shifts, particularly when encountering previously unseen or more realistic deepfake manipulations. This remains an area for future work to explore; several strategies for this are discussed further in Section 5.

Table 10. Ablation results across all training scenarios (40k samples).

Held-Out	Test Set	Acc. (%)	Real Acc. (%)	Fake Acc. (%)
Celeb-DF (v2)	TM	88.30	87.38	89.19
	TIMIT	99.96	99.91	100.00
	DFDC	89.11	89.11	89.11
	FF++	99.56	99.71	95.24
	Celeb-DF (v2)	**55.67**	99.78	**11.56**
DFDC	TM	88.30	87.38	89.19
	TIMIT	99.42	98.72	100.00
	FF++	99.53	99.64	96.53
	Celeb-DF (v2)	97.83	97.11	98.56
	DFDC	**62.08**	95.31	**28.84**
FF++	TM	88.88	89.76	88.04
	TIMIT	99.61	99.15	100.00
	Celeb-DF (v2)	96.92	95.67	98.18
	DFDC	88.53	91.29	85.78
	FF++	**64.86**	86.78	**44.20**
TIMIT	TM	88.38	88.81	87.97
	Celeb-DF (v2)	96.77	94.49	99.04
	DFDC	88.66	99.36	87.96
	FF++	99.51	99.68	94.59
	TIMIT	93.44	95.98	91.36
TM	TIMIT	99.88	99.74	100.00
	DFDC	88.76	91.93	85.58
	FF++	99.67	99.84	94.72
	Celeb-DF (v2)	97.73	98.53	96.93
	TM	**76.80**	97.78	**56.66**

5. Future Work

This section discusses strategies for future research tackling our proposed model's limitations (as discussed in Section 4.4). In essence, although our proposed deepfake detection model's performance is high for some datasets, the main limitation is that its performance is lower for some datasets that were seen during training (e.g., Celeb-DF (v2), as discussed in Section 4.4.1), and other datasets that were not seen during training (i.e., that are out of distribution, demonstrating its limited generalization, as discussed in Section 4.4.2).

To tackle these limitations, the deepfake classification model could incorporate additional features that allow it to generalize better. For example, a recent effective approach to synthetic image detection is using latent representations from pretrained foundation models such as CLIP [85].

Still, even with more generalizable features, deepfake detection models will continue to struggle with OOD samples. Therefore, another viable strategy would be to perform out-of-distribution detection prior to deepfake classification [86]. This way, we would know when the model's output cannot be trusted due to the input being OOD.

Finally, another strategy is to perform one-class training for deepfake detection. That is, by only considering real samples during training, we avoid overfitting to specific deepfake generators altogether. Such models can be trained in a multi-task manner, e.g., to jointly reconstruct disturbed real faces and perform deepfake detection [87]. Note that this strategy bears similarities to the face reconstruction used in the VAE of network B in our proposed GenConViT, and hence may be a particularly viable direction for future research.

6. Conclusions

In this work, we proposed a Generative Convolutional Vision Transformer (GenConViT) that extracts visual artifacts and latent data distributions to detect deepfake videos. GenConViT combines the ConvNeXt and Swin Transformer architectures to learn from the local and global image features of a video, as well as an AE and VAE to learn from a video's internal data representations. Our approach provides an effective solution for identifying fake videos while preserving the integrity of media. Through extensive experiments on a diverse range of datasets, including DFDC, FF++, DeepfakeTIMIT, and Celeb-DF (v2), our GenConViT model demonstrated an improved and robust performance with high classification accuracy, F1 scores, and AUC values. Our ablation study reveals challenges in generalizing to unseen or more complex manipulations, highlighting the need for further research to improve its domain adaptability. Overall, our proposed GenConViT model is a promising approach for accurate and reliable deepfake video detection. As our model is open-source, its practical use has already been demonstrated: it was used by TrueMedia.org, a non-profit organization, to detect deepfakes and support fact-checking efforts [88].

Author Contributions: Conceptualization, D.W.D., S.A. and Z.A.; Methodology, D.W.D., H.M., S.A. and Z.A.; Software, D.W.D.; Validation, D.W.D., H.M. and G.V.W.; Formal analysis, D.W.D. and H.M.; Investigation, D.W.D., H.M., P.L. and G.V.W.; Resources, D.W.D., P.L., S.A. and G.V.W.; Data curation, D.W.D. and H.M.; Writing—original draft, D.W.D.; Writing—review and editing, D.W.D., H.M. and G.V.W.; Visualization, D.W.D.; Supervision, H.M., P.L. and G.V.W.; Project administration, H.M., P.L., S.A. and G.V.W.; Funding acquisition, P.L., S.A. and G.V.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Addis Ababa University Research Grant for the Adaptive Problem-Solving Research. Reference number RD/PY-183/2021. Grant number AR/048/2021, and the Research Foundation—Flanders (FWO under project grant G0A2523N), the Flemish government (COM-PRESS project, within the relanceplan Vlaamse Veerkracht), IDLab (Ghent University—imec), Flanders Innovation and Entrepreneurship (VLAIO), and the European Union.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to third party copyrights.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Masood, M.; Nawaz, M.; Malik, K.M.; Javed, A.; Irtaza, A.; Malik, H. Deepfakes generation and detection: state-of-the-art, open challenges, countermeasures, and way forward. *Appl. Intell.* **2022**, *53*, 3974–4026. [CrossRef]
2. Tolosana, R.; Vera-Rodriguez, R.; Fierrez, J.; Morales, A.; Ortega-Garcia, J. An Introduction to Digital Face Manipulation. In *Handbook of Digital Face Manipulation and Detection: From DeepFakes to Morphing Attacks*; Rathgeb, C., Tolosana, R., Vera-Rodriguez, R., Busch, C., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 3–26. [CrossRef]
3. Dagar, D.; Vishwakarma, D.K. A literature review and perspectives in deepfakes: generation, detection, and applications. *Int. J. Multimed. Inf. Retr.* **2022**, *11*, 219–289. [CrossRef]
4. Juefei-Xu, F.; Wang, R.; Huang, Y.; Guo, Q.; Ma, L.; Liu, Y. Countering Malicious DeepFakes: Survey, Battleground, and Horizon. *Int. J. Comput. Vision* **2022**, *130*, 1678–1734. [CrossRef]
5. Zachary, G.P. Digital Manipulation and the Future of Electoral Democracy in the U.S. *IEEE Trans. Technol. Soc.* **2020**, *1*, 104–112. [CrossRef]
6. Nguyen, T.T.; Nguyen, Q.V.H.; Nguyen, D.T.; Nguyen, D.T.; Huynh-The, T.; Nahavandi, S.; Nguyen, T.T.; Pham, Q.V.; Nguyen, C.M. Deep learning for deepfakes creation and detection: A survey. *Comput. Vis. Image Underst.* **2022**, *223*, 103525. [CrossRef]

7. Korshunov, P.; Marcel, S., The Threat of Deepfakes to Computer and Human Visions. In *Handbook of Digital Face Manipulation and Detection: From DeepFakes to Morphing Attacks*; Rathgeb, C.; Tolosana, R.; Vera-Rodriguez, R., Busch, C., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 97–115. [CrossRef]
8. Vasist, P.N.; Krishnan, S. Engaging with deepfakes: A meta-synthesis from the perspective of social shaping of technology theory. *Internet Res.* **2022**, *33*, 1670–1726. [CrossRef]
9. Kingra, S.; Aggarwal, N.; Kaur, N. Emergence of deepfakes and video tampering detection approaches: A survey. *Multimed. Tools Appl.* **2022**, *82*, 10165–10209. [CrossRef]
10. Liu, P.; Lin, Y.; He, Y.; Wei, Y.; Zhen, L.; Zhou, J.T.; Goh, R.S.M.; Liu, J. Automated deepfake detection. *arXiv* **2021**, arXiv:2106.10705.
11. Ur Rehman Ahmed, N.; Badshah, A.; Adeel, H.; Tajammul, A.; Daud, A.; Alsahfi, T. Visual Deepfake Detection: Review of Techniques, Tools, Limitations, and Future Prospects. *IEEE Access* **2025**, *13*, 1923–1961. [CrossRef]
12. Afchar, D.; Nozick, V.; Yamagishi, J.; Echizen, I. Mesonet: a compact facial video forgery detection network. In Proceedings of the 2018 IEEE international workshop on information forensics and security (WIFS), Hong Kong, 11–13 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–7.
13. Yang, X.; Li, Y.; Lyu, S. Exposing Deep Fakes Using Inconsistent Head Poses. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 8261–8265. [CrossRef]
14. Trabelsi, A.; Pic, M.M.; Dugelay, J.L. Improving deepfake detection by mixing top solutions of the DFDC. In Proceedings of the 2022 30th European Signal Processing Conference (EUSIPCO), Belgrade, Serbia, 29 August–2 September 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 643–647.
15. Matern, F.; Riess, C.; Stamminger, M. Exploiting visual artifacts to expose deepfakes and face manipulations. In Proceedings of the 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW), Waikoloa Village, HI, USA, 7–11 January 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 83–92.
16. Heo, Y.J.; Yeo, W.H.; Kim, B.G. DeepFake detection algorithm based on improved vision transformer. *Appl. Intell.* **2022**, *53*, 7512–7527. [CrossRef]
17. Gu, Z.; Chen, Y.; Yao, T.; Ding, S.; Li, J.; Huang, F.; Ma, L. Spatiotemporal inconsistency learning for deepfake video detection. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, 20–24 October 2021; pp. 3473–3481.
18. Khan, S.A.; Dai, H. Video transformer for deepfake detection with incremental learning. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, 20–24 October 2021; pp. 1821–1828.
19. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. ACM* **2020**, *63*, 139–144. [CrossRef]
20. Karras, T.; Laine, S.; Aila, T. A Style-Based Generator Architecture for Generative Adversarial Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 4217–4228. [CrossRef] [PubMed]
21. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
22. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2022**, arXiv:1312.6114.
23. Kingma, D.P.; Welling, M. An introduction to variational autoencoders. *Found. Trends® Mach. Learn.* **2019**, *12*, 307–392. [CrossRef]
24. Dhariwal, P.; Nichol, A. Diffusion models beat gans on image synthesis. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 8780–8794.
25. Ciftci, U.A.; Demir, I.; Yin, L. Fakecatcher: Detection of synthetic portrait videos using biological signals. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *Early Access*. [CrossRef]
26. Yan, Z.; Sun, P.; Lang, Y.; Du, S.; Zhang, S.; Wang, W.; Liu, L. Multimodal graph learning for deepfake detection. *arXiv* **2022**, arXiv:2209.05419.
27. Wesselkamp, V.; Rieck, K.; Arp, D.; Quiring, E. Misleading deep-fake detection with gan fingerprints. In Proceedings of the 2022 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 22–26 May 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 59–65.
28. Zhang, D.; Lin, F.; Hua, Y.; Wang, P.; Zeng, D.; Ge, S. Deepfake video detection with spatiotemporal dropout transformer. In Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 10–14 October 2022; pp. 5833–5841.
29. Kim, D.K.; Kim, K. Generalized Facial Manipulation Detection with Edge Region Feature Extraction. In Proceedings of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2022; pp. 2784–2794.
30. Yu, N.; Skripniuk, V.; Abdelnabi, S.; Fritz, M. Artificial fingerprinting for generative models: Rooting deepfake attribution in training data. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 14448–14457.
31. Guillaro, F.; Cozzolino, D.; Sud, A.; Dufour, N.; Verdoliva, L. Trufor: Leveraging all-round clues for trustworthy image forgery detection and localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 20606–20615.
32. Li, B.; Sun, J.; Poskitt, C.M.; Wang, X. How Generalizable are Deepfake Image Detectors? An Empirical Study. *arXiv* **2024**, arXiv:2308.04177.

33. Bank, D.; Koenigstein, N.; Giryes, R., Autoencoders. In *Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook*; Rokach, L., Maimon, O., Shmueli, E., Eds.; Springer International Publishing: Cham, Switzerland, 2023; pp. 353–374. [CrossRef]
34. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11976–11986.
35. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 10012–10022.
36. O'shea, K.; Nash, R. An introduction to convolutional neural networks. *arXiv* **2015**, arXiv:1511.08458.
37. Haridas, R.; Jyothi, R. Convolutional neural networks: A comprehensive survey. *Int. J. Appl. Eng. Res.* **2019**, *14*, 780–789. [CrossRef]
38. Kammoun, A.; Slama, R.; Tabia, H.; Ouni, T.; Abid, M. Generative Adversarial Networks for Face Generation: A Survey. *ACM Comput. Surv.* **2022**, *55*, 1–37. [CrossRef]
39. Thies, J.; Zollhofer, M.; Stamminger, M.; Theobalt, C.; Nießner, M. Face2face: Real-time face capture and reenactment of rgb videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2387–2395.
40. Hsu, G.S.; Tsai, C.H.; Wu, H.Y. Dual-generator face reenactment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 642–650.
41. Sun, J.; Deng, Q.; Li, Q.; Sun, M.; Ren, M.; Sun, Z. AnyFace: Free-Style Text-To-Face Synthesis and Manipulation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 18687–18696.
42. Korshunova, I.; Shi, W.; Dambre, J.; Theis, L. Fast face-swap using convolutional neural networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 3697–3705.
43. Li, L.; Bao, J.; Yang, H.; Chen, D.; Wen, F. FaceShifter: Towards High Fidelity And Occlusion Aware Face Swapping. *arXiv* **2020**, arXiv:1912.13457.
44. Zhu, Y.; Li, Q.; Wang, J.; Xu, C.Z.; Sun, Z. One Shot Face Swapping on Megapixels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 4834–4844.
45. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive Growing of GANs for Improved Quality, Stability, and Variation. *arXiv* **2018**, arXiv:1710.10196.
46. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein Generative Adversarial Networks. In Proceedings of the the 34th International Conference on Machine Learning (ICML), Sydney, NSW, Australia, 6–11 August 2017; pp. 214–223.
47. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-To-Image Translation With Conditional Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
48. Pang, Y.; Lin, J.; Qin, T.; Chen, Z. Image-to-Image Translation: Methods and Applications. *IEEE Trans. Multimed.* **2022**, *24*, 3859–3881. [CrossRef]
49. Gragnaniello, D.; Marra, F.; Verdoliva, L. Detection of AI-Generated Synthetic Faces. In *Handbook of Digital Face Manipulation and Detection: From DeepFakes to Morphing Attacks*; Rathgeb, C., Tolosana, R., Vera-Rodriguez, R., Busch, C., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 191–212. [CrossRef]
50. Mareen, H.; D'haeseleer, S.; Van Damme, K.; Evens, T.; Lambert, P.; Van Wallendael, G. COM-PRESS: Dashboard to Detect (AI-Based) Image Manipulations. In Proceedings of the 2024 IEEE Gaming, Entertainment, and Media Conference (GEM), Turin, Italy, 5–7 June 2024; p. 1. [CrossRef]
51. Nguyen, H.; Tieu, T.N.D.T.; Nguyen-Son, H.Q.; Nozick, V.; Yamagishi, J.; Echizen, I. Modular Convolutional Neural Network for Discriminating between Computer-Generated Images and Photographic Images. In Proceedings of the ARES 2018 Proceedings of the 13th International Conference on Availability, Reliability and Security, Hambourg, Germany, 27–30 August 2018; Volume 10, pp. 1–10. [CrossRef]
52. Nguyen, H.H.; Yamagishi, J.; Echizen, I. Capsule-forensics: Using Capsule Networks to Detect Forged Images and Videos. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 2307–2311. [CrossRef]
53. Li, Y.; Lyu, S. Exposing DeepFake Videos By Detecting Face Warping Artifacts. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–20 June 2019.
54. Li, L.; Bao, J.; Zhang, T.; Yang, H.; Chen, D.; Wen, F.; Guo, B. Face X-Ray for More General Face Forgery Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 5000–5009. [CrossRef]

55. Sun, Y.; Zhang, Z.; Echizen, I.; Nguyen, H.H.; Qiu, C.; Sun, L. Face Forgery Detection Based on Facial Region Displacement Trajectory Series. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops, Waikoloa, HI, USA, 3–7 January 2023.
56. Li, Y.; Chang, M.C.; Lyu, S. In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking. In Proceedings of the 2018 IEEE International Workshop on Information Forensics and Security (WIFS), Hong Kong, 11–13 December 2018; pp. 1–7. [CrossRef]
57. Chintha, A.; Rao, A.; Sohrawardi, S.; Bhatt, K.; Wright, M.; Ptucha, R. Leveraging edges and optical flow on faces for deepfake detection. In Proceedings of the 2020 IEEE International Joint Conference on Biometrics (IJCB), Houston, TX, USA, 28 September–1 October 2020.
58. Zhao, T.; Xu, X.; Xu, M.; Ding, H.; Xiong, Y.; Xia, W. Learning Self-Consistency for Deepfake Detection. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 11–17 October 2021.
59. Sabir, E.; Cheng, J.; Jaiswal, A.; AbdAlmageed, W.; Masi, I.; Natarajan, P. Recurrent Convolutional Strategies for Face Manipulation Detection in Videos. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Long Beach, CA, USA, 15–20 June 2019.
60. Deressa, D.W.; Lambert, P.; Van Wallendael, G.; Atnafu, S.; Mareen, H. Improved Deepfake Video Detection Using Convolutional Vision Transformer. In Proceedings of the 2024 IEEE Gaming, Entertainment, and Media Conference (GEM), Turin, Italy, 5–7 June 2024; pp. 1–6. [CrossRef]
61. Heo, Y.J.; Choi, Y.J.; Lee, Y.W.; Kim, B.G. Deepfake Detection Scheme Based on Vision Transformer and Distillation. *arXiv* **2021**, arXiv:2104.01353.
62. Coccomini, D.A.; Messina, N.; Gennaro, C.; Falchi, F. Combining EfficientNet andVision Transformers for Video Deepfake Detection. In *Image Analysis and Processing—ICIAP 2022: 21st International Conference, Lecce, Italy, 23–27 May 2022, Proceedings, Part III*; Springer: Cham, Switzerland, 2022; pp. 219–229. [CrossRef]
63. Chen, T.; Yang, S.; Hu, S.; Fang, Z.; Fu, Y.; Wu, X.; Wang, X. Masked Conditional Diffusion Model for Enhancing Deepfake Detection. In Proceedings of the 2024 International Joint Conference on Neural Networks (IJCNN), Yokohama, Japan, 30 June–5 July 2024; pp. 1–7. [CrossRef]
64. Bhattacharyya, C.; Wang, H.; Zhang, F.; Kim, S.; Zhu, X. Diffusion Deepfake. *arXiv* **2024**, arXiv:2404.01579.
65. Sun, K.; Chen, S.; Yao, T.; Liu, H.; Sun, X.; Ding, S.; Ji, R. DiffusionFake: Enhancing Generalization in Deepfake Detection via Guided Stable Diffusion. In *Advances in Neural Information Processing Systems*; Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C., Eds.; Curran Associates, Inc.: Red Hook, NY, USA; 2024; Volume 37, pp. 101474–101497.
66. Xu, Y.; Liang, J.; Jia, G.; Yang, Z.; Zhang, Y.; He, R. TALL: Thumbnail Layout for Deepfake Video Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 22658–22668.
67. Le, B.M.; Woo, S.S. Quality-Agnostic Deepfake Detection with Intra-model Collaborative Learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 22378–22389.
68. Li, H.; Zhou, J.; Li, Y.; Wu, B.; Li, B.; Dong, J. FreqBlender: Enhancing DeepFake Detection by Blending Frequency Knowledge. In *Advances in Neural Information Processing Systems*; Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C., Eds.; Curran Associates, Inc.: Red Hook, NY, USA; 2024; Volume 37, pp. 44965–44988.
69. Seferbekov, S. A Prize Winning Solution for DFDC Challenge. Github, 2020. Available online: https://github.com/selimsef/dfdc_deepfake_challenge (accessed on 9 June 2025).
70. Khan, S.A.; Dang-Nguyen, D.T. Hybrid Transformer Network for Deepfake Detection. *arXiv* **2022**, arXiv:2208.05820.
71. Wightman, R. PyTorch Image Models, G. Github, 2025. Available online: https://github.com/huggingface/pytorch-image-models (accessed on 9 June 2025).
72. Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and Flexible Image Augmentations. *Information* **2020**, *11*, 125. [CrossRef]
73. Dolhansky, B.; Howes, R.; Pflaum, B.; Baram, N.; Ferrer, C.C. The Deepfake Detection Challenge (DFDC) Preview Dataset. *arXiv* **2019**, arXiv:1910.08854.
74. Dolhansky, B.; Bitton, J.; Pflaum, B.; Lu, J.; Howes, R.; Wang, M.; Ferrer, C.C. The DeepFake Detection Challenge (DFDC) Dataset. *arXiv* **2020**, arXiv:2006.07397.
75. Chen, W.; Chua, S.L.B.; Winkler, S.; Ng, S.K. Trusted Media Challenge Dataset and User Study. In Proceedings of the 31st ACM International Conference on Information and Knowledge Management, Atlanta, GA, USA, 17–21 October 2022.
76. Korshunov, P.; Marcel, S. DeepFakes: A New Threat to Face Recognition? Assessment and Detection. *arXiv* **2018**, arXiv:1812.08685.
77. Sanderson, C.; Lovell, B.C. Multi-Region Probabilistic Histograms for Robust and Scalable Identity Inference. In *Advances in Biometrics*; Springer: Berlin/Heidelberg, Germany, 2009.
78. Li, Y.; Yang, X.; Sun, P.; Qi, H.; Lyu, S. Celeb-DF: A Large-Scale Challenging Dataset for DeepFake Forensics. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020.

79. Li, Y.; Sun, P.; Qi, H.; Lyu, S. Toward the Creation and Obstruction of DeepFakes. In *Handbook of Digital Face Manipulation and Detection: From DeepFakes to Morphing Attacks*; Rathgeb, C., Tolosana, R., Vera-Rodriguez, R., Busch, C., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 71–96. [CrossRef]
80. Rössler, A.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J.; Nießner, M. FaceForensics++: Learning to Detect Manipulated Facial Images. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 February 2020; pp. 1–11.
81. Thies, J.; Zollhöfer, M.; Nießner, M. Deferred neural rendering: image synthesis using neural textures. *ACM Trans. Graph.* **2019**, *38*, 1–12. [CrossRef]
82. Geitgey, A. face_recognition, 2017. GitHub Repository. Available online: https://github.com/ageitgey/face_recognition (accessed on 9 June 2025).
83. Bazarevsky, V.; Kartynnik, Y.; Vakunov, A.; Raveendran, K.; Grundmann, M. BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs. *arXiv* **2019**, arXiv:1907.05047.
84. Thing, V.L.L. Deepfake Detection with Deep Learning: Convolutional Neural Networks versus Transformers. *arXiv* **2023**, arXiv:2304.03698.
85. Ojha, U.; Li, Y.; Lee, Y.J. Towards universal fake image detectors that generalize across generative models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 24480–24489.
86. Maiano, L.; Casadei, F.; Amerini, I. Enhancing Abnormality Identification: Robust Out-of-Distribution Strategies for Deepfake Detection. *arXiv* **2025**, arXiv:2506.02857.
87. Tian, J.; Yu, C.; Wang, X.; Chen, P.; Xiao, Z.; Dai, J.; Han, J.; Chai, Y. Real appearance modeling for more general deepfake detection. In Proceedings of the European Conference on Computer Vision, Milan, Italy, 29 September–4 October 2024; Springer: Berlin/Heidelberg, Germany, 2024; pp. 402–419.
88. Kim, A. Open-Source Deepfake Detection Models from TrueMedia.org, 2025. Available online: https://automata88.medium.com/open-source-deepfake-detection-models-from-truemedia-org-29f9ce59882d (accessed on 3 March 2025).