ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
(ICAI)

Máster en Big Data: Tecnología y Analítica Avanzada

Consulting a Group of Experts for Ensemble Deepfake Detection

Author
Juan Pablo Chávez Vélez

Directed by
Jose Luis Ruíz Gahete and Prasanna Kumar Ballepalli

Madrid
June 2025

Acknowledgements

I would like to express my deepest gratitude to all of the authors and research teams whose pioneering work laid the foundation for this thesis. In particular, my sincere thanks go to the creators of DeepFakeBench for providing an invaluable benchmarking toolkit that has been instrumental in evaluating and comparing detection methods. Special thanks to Eugenio, Alberto, and Ana for their expertise and passion in the vast field of deep learning, and to my Master's director, Carlos, for his guidance and mentorship throughout this journey. Finally, I owe a heartfelt thank you to my family for their unwavering support and encouragement during this process. I am also profoundly grateful to the researchers behind the seminal contributions to generative adversarial networks, autoencoders, diffusion models, and Vision Transformers—whose insights into model architectures, training strategies, and forensic applications have directly informed and inspired my own experiments. Without the tireless efforts of these scholars—whose names appear throughout my citations and whose code, data, and open-source releases made this work possible, the dedication of my instructors, and the love of my family, this project could never have achieved its current depth and rigor. Thank you to each and every one of you for advancing the field and for generously sharing your expertise and support with the community.

Resumen

Los deepfakes representan una grave amenaza para la integridad de los sistemas financieros, los procesos políticos y la percepción de la realidad. En este trabajo, realizo un estudio de los enfoques más comunes para la detección de deepfakes, sus puntos fuertes y sus deficiencias. Posteriormente, propongo un nuevo marco para la colaboración de varios modelos de detección de última generación como un conjunto. Se presenta una visión general de la arquitectura, así como un análisis más detallado de los modelos base seleccionados. El modelo ensemble se entrena y valida sobre el popular benchmark FaceForensics++ y se prueba en los benchmarks CelebDF–v1&v2 y DeepfakeDetection. Nuestros experimentos demuestran que, aunque no superan a los modelos individuales con mayor puntuación, tanto las estrategias de fusión de características como las de puntuación mejoran la generalización y la estabilidad en distintos dominios de conjuntos de datos.

Abstract

keywords - Deep-Learning, ensemble, Deepfake, Detection, forensic

Deepfakes present a serious threat to the integrity of financial systems, political processes, and one's sense of reality as a whole. In this work, I conduct a survey of the most common approaches toward Deepfake detection, their strengths, and shortcomings. Subsequently, I propose a novel framework for the collaboration of several state-of-the-art detection models as an ensemble. A general overview of the architecture is presented, as well as a more detailed analysis of the selected model backbones. The ensemble model is trained and validated on the popular Deepfake Detection benchmark FaceForensics++ and tested on CelebDF–v1&v2 and DeepfakeDetection benchmarks. Our experiments show that, while not surpassing the highest scoring individual models, both feature and score-level fusion strategies improve generalization and stability throughout distinct dataset domains.

List of Figures

List of Tables

ICAI	Instituto Católico de Artes e Industrias
TFM	Trabajo de Fin de Máster
CNN	Convolutional Neural Network
FCN	Fully Connected Network
FC	Final Classifier
ASV	Automatic Speaker Verification
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
ViT	Vision Transformer
LS	Latent Space
LR	Learning Rate
FF++	FaceForensics++
F2F	Face2Face
DFD	DeepFakeDetection
DFDC	DeepFake Detection Challenge
FS	FaceSwap
NT	NeuralTextures
GAN	Generative Adversarial Network
SPSL	Spatial-Phase Shallow Learning
UIA-ViT	Unsupervised Inconsistency-Aware Vision Transformer
UCF	Uncovering Common Features
STIL	Spatio-Temporal Inconsistency Learning
AE	Autoencoder
MAE	Masked Autoencoder
VAE	Variational Autoencoder
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
AUC	Area Under the Curve
ACC	Accuracy
EER	Equal-Error Rate
AP	Average Precision
SBI	Self-Blended Images

Contents

Chapter 1

Introduction

1.1 Background and Motivation

The rapid advancement of artificial intelligence has led to significant progress across the domains of computer graphics and computational science. In particular, the availability of high-performance consumer-grade graphics processing units (GPUs) has greatly facilitated the training and deployment of AI models. However, the increasing accessibility and democratization of AI technology have introduced substantial societal challenges. In an era characterized by the widespread dissemination of misinformation and so-called "fake news," the risks associated with media manipulation have intensified. These include threats such as mass deception, defamation, non-consensual explicit content [4], identity theft, electoral interference [5], and the amplification of extremist narratives—factors that collectively contribute to the erosion of public trust in digital media.

Given the growing sophistication of generative AI technologies, it is critical to develop effective methods for detecting manipulated media. Consider, for instance, a case where a big financial firm was scammed for 25 million US dollars with the use of generative AI [6]. One can only speculate on the potential ramifications, be it a fabricated press conference, to simulate an endorsement by a political figure [7], a head of state giving out sensitive strategic military information [8], a social media post inciting violence against a marginalized group or a bankruptcy announcement from a big corporation. The implications of such scenarios highlight the urgency of this research domain, as media manipulation now poses serious threats to the credibility of individuals, institutions, and international relations.

Although some may argue that concerns about fabricated media are not new, given the historical presence of computer-generated imagery (CGI), the current landscape is markedly different. The unprecedented speed, scale, and realism with which deepfakes and other AI-generated content can now be produced come to show the gravity of the problem. The widespread availability of open-source frameworks for high-fidelity deepfake generation further exacerbates the situation by facilitating access to powerful media counterfeiting tools to anyone with basic to no technical knowledge.

While state-of-the-art deepfake detection models demonstrate impressive performance on well established benchmarks, they often exhibit limited generalization when confronted with previously unseen manipulation techniques. This is largely due to the rapidly evolving nature of generative AI. In this work, we do not introduce a new detection model per se. Instead, we propose a framework in which multiple specialized detection models collaborate, leveraging their complementary feature extraction mechanisms to achieve improved robustness and generalization against diverse and novel forms of manipulated media.

1.2 Problem Statement

Some top-performing methods [9], [10] demonstrate strong performance on widely used deepfake detection benchmarks. However, their accuracy often deteriorates when applied to previously unseen manipulation techniques. Many of these detection approaches rely on identifying visual artifacts introduced by generative adversarial networks (GANs) or diffusion-based synthesis models. These artifacts, commonly referred to as ’fingerprints’, can manifest in a variety of ways, including inconsistencies at facial boundaries, unnatural distributions of pixel intensities, irregularities in the motion of the eyes or mouth, and anomalous patterns in the frequency domain. When such artifacts are systematically present in a given generation method, they can serve as effective cues for supervised classifiers. As a result, models trained to detect these specific signatures tend to perform well on the subset of manipulations from which they were derived. However, their generalizability across other types of manipulations is often limited, raising concerns about robustness in real-world scenarios.

1.3 Research Objectives

The primary aim of this study is to investigate the potential of deep learning ensemble architectures for enhancing generalizability in deepfake detection. Specifically, we explore a framework in which multiple base models, each optimized to detect distinct artifact types, are integrated to form a unified ensemble system. While ensemble learning is well-established in traditional machine learning domains, relatively little work has been conducted on its application to deepfake detection using deep neural networks. In this research, we evaluate the performance of several state-of-the-art detection models, both individually and as an ensemble, with the goal of understanding how their complementary strengths may contribute to more robust classification. The scope of this study is limited to visual-based deepfake detection (i.e., image and video analysis). Nevertheless, extending ensemble-based approaches to anti-audio spoofing represents a promising avenue for future research.

1.4 Key Contributions

In this work we make the following contributions:

- Provided a unified training and evaluation pipeline, available as open-source code, that loads pretrained branch weights, instantiates ensemble models, and produces per-dataset CSV prediction files for a more agile model evaluation. This reproducible framework eases comparisons among future detectors.

- Proposed and validated two complementary fusion paradigms, showing that score-level MLP fusion excels in stability across splits, while attentive feature fusion can yield even higher separation on highly heterogeneous data.

Chapter 2

Related Work

2.1 Deepfake Generation

The origins of modern deepfake technology can be traced back to 2014, when researchers introduced a groundbreaking machine learning framework known as Generative Adversarial Networks (GANs) [11], which enabled the estimation of complex generative models through adversarial training. This foundational development laid the groundwork for the sophisticated generative techniques that exist today. Since then, a wide range of methods have emerged to manipulate or synthesize visual content, particularly faces, in images and videos.

These techniques are typically categorized based on the nature of the manipulation being performed as [12] well describe. Common categories include face swapping, face reenactment, deep identity replacement (deepswap), facial attribute modification, and generation from scratch using latent generative models. They may also be differentiated by the type and amount of input data they require, ranging from image-driven and audio-driven methods to text-driven or fully multimodal systems.

Additionally, generative models are often classified by their data requirements in terms of identity representation: multi-shot, few-shot, one-shot, and zero-shot. This terminology reflects the amount of input data necessary for the model to learn and reproduce a given subject's identity. Notably, state-of-the-art generative architectures are increasingly capable of synthesizing highly realistic facial content from as little as a single reference image. When combined with zero-shot voice cloning technologies, these capabilities raise serious ethical and security concerns, including the potential for identity theft, fraud, and extortion.

2.1.1 Approaches to Image Forgery

In this next section, we describe the most common deepfake manipulations and some of their most representative algorithms.

Face Swapping

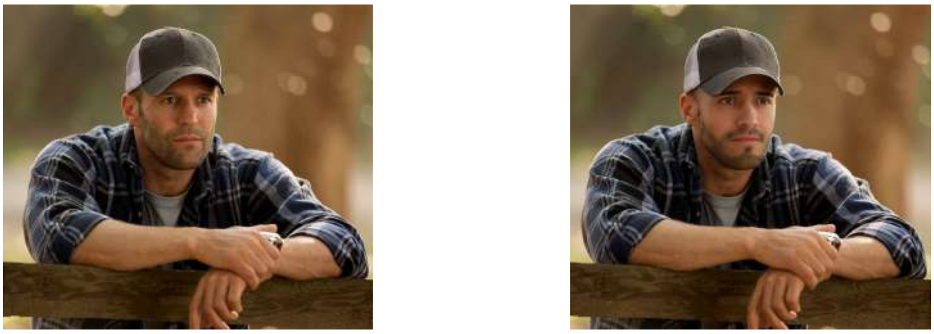
Face swapping refers to the process of replacing one individual's face with that of another in an image or video, such that the source's identity and features are transferred onto the target while maintaining the target's expressions, pose, and lighting conditions. Traditional approaches perform geometric alignment via facial landmarks, apply warping to match the source onto the target, and blend boundaries using color correction techniques or Poisson image editing to minimize visible seams. In contrast, contemporary methods leverage deep neural networks, typically autoencoder-based architectures trained to disentangle identity from appearance, to achieve more robust and seamless face swaps across varying poses and conditions.

One practical implementation is InSwapper128, an ONNX-exported face-swapping model developed by [13] and popularized through the FaceFusion pipeline [14]. This model operates at a resolution of 128×128 and integrates face detection, 3D landmark estimation, and identity transfer into a streamlined, real-time pipeline. It utilizes a pretrained face recognition network, similar to ArcFace, to extract identity embeddings, which are fused into the target image using adaptive normalization within a convolutional generator. Post-processing steps, such as restoration via [15], are applied to enhance realism and suppress artifacts. Due to its portability and speed, InSwapper128 is widely adopted for automated video face swapping.

Another notable method is GHOST (Generative High-fidelity One-Shot Transfer) [16], also referred to as GhostFace. It is a one-shot face-swapping pipeline capable of operating on both images and videos without requiring identity-specific fine-tuning. The method separates a source image into identity and appearance features, and merges these with the target's content using a cross-attention fusion module. A pretrained masked autoencoder (MAE) facilitates the extraction of high-quality facial representations, while a lightweight convolutional decoder reconstructs the final output. GHOST is designed for high fidelity even under complex head poses and lighting conditions, and is particularly effective in low-data or real-time applications due to its one-shot nature.

Lip Syncing

Lip syncing in the context of deepfakes involves animating a still image or video of a person such that their mouth movements align convincingly with a given audio track or textual input. Earlier methods employed rule-based mappings from



(a) Pristine

(b) Manipulated

Figure 2.1: High resolution faceswapping with Deepswapper. [1]

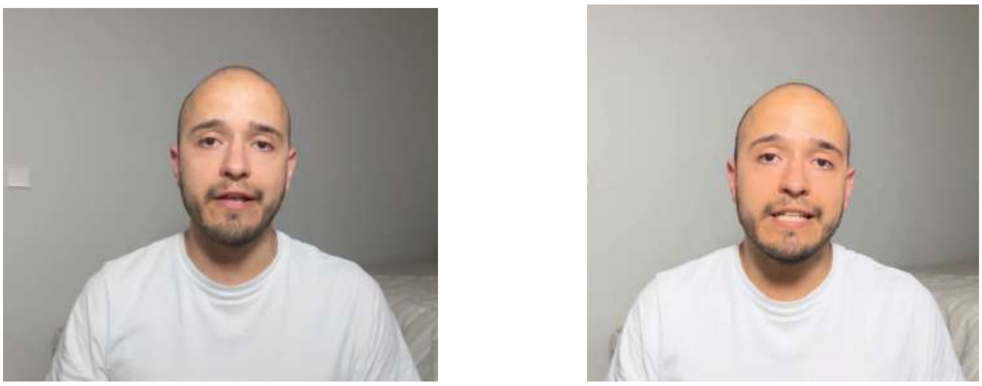
phonemes to visemes (mouth shapes), typically relying on facial landmarks and pre-defined motion templates. Recent advances, however, utilize deep learning models, particularly recurrent and transformer-based sequence models, to learn audio-to-lip motion mappings in an end-to-end manner.

Wav2Lip [17] represents a significant advancement in this domain. It is a speaker-independent model that achieves precise lip synchronization by employing a dual-stream encoder–decoder architecture: one stream processes video frames centered on the face, while the other processes Mel-spectrogram representations of the audio. A lip-sync discriminator evaluates the alignment between audio and video, providing feedback that guides the generator toward producing natural and temporally accurate lip movements. Unlike earlier approaches, Wav2Lip does not require identity-specific training and generalizes well across languages, lighting conditions, and speaker identities. Commercial systems [2] [18] have adopted similar principles, offering high-quality lip-sync results across diverse languages and voice profiles.

Face Reenactment

Face re-enactment involves transferring head pose, facial expressions, and movements of a source actor (the "driving" subject) onto the face of a target individual while preserving the target's identity. This technique is distinct from face swapping in that the identity remains fixed, and only dynamic facial characteristics are modified. Traditional pipelines achieve this through 3D facial landmark tracking or by fitting morphable face models to both subjects, which are then used to animate the target's face based on the source's motion parameters.

More recent methods incorporate deep neural networks to improve flexibility and visual realism. For instance, [19] [20] use keypoint-based transfer mechanisms or latent space transformations to map expressions from the source to the target.



(a) Pristine

(b) Manipulated

Figure 2.2: Lip sync AI avatar with HeyGen [2]

[21] extends this concept using a one-shot strategy based on StyleGAN2. [22] It first inverts the source (identity) and driving (expression) frames into latent codes, and then uses a lightweight hypernetwork to generate per-layer modulation weights for the StyleGAN2 synthesis network. This enables high-fidelity reenactment from a single image without fine-tuning and performs robustly even under challenging conditions such as extreme head rotations or nonfrontal angles.

Deep Swapping

Deep swapping is a specialized form of face swapping that relies on deep generative models, typically GAN-based encoder–decoder networks, to disentangle identity from expression and background. These systems are often trained on large-scale datasets to develop a generalized facial representation space. At inference, the identity of the source is encoded and transferred onto a target sequence while preserving the latter's motion and environmental cues. Popular open-source frameworks such as [23] and [24] embody this approach, often enhanced by perceptual or attention-based losses to improve visual coherence.

[25] is a notable GAN-based method designed for identity-agnostic face swapping at 256×256 resolution. Its key innovation, the Identity Injection Module (IIM), injects a source identity embedding into the target's feature representation, enabling a single model to generalize across many identities. Unlike explicit landmark-based blending, it learns to implicitly retain fine details such as gaze direction and micro-expressions via a weak feature-matching loss. During inference, source and target faces are aligned and passed through an encoder-IIM-decoder pipeline. The resulting output is blended into the original scene, supporting seamless, real-time identity swaps without requiring model retraining.

Facial Attribute Modification

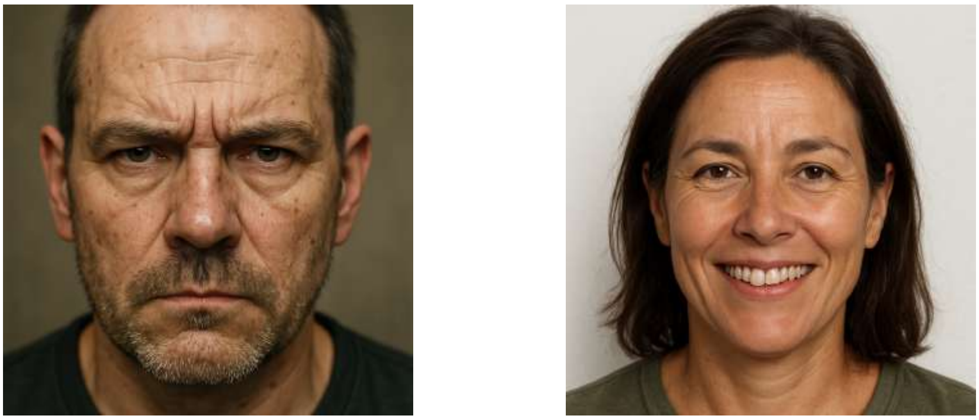
Facial attribute manipulation involves altering specific traits of a face, such as age, gender, hair color, skin tone, or the presence of accessories, while preserving the underlying identity and expression. A widely adopted approach employs conditional GANs [26], where the generator is conditioned on both the input image and the desired attribute label. Alternatively, latent space manipulation techniques in pretrained models like [27] allow fine control over features by shifting the latent vector along attribute-specific directions. Other architectures decouple content and attribute representations within encoder–decoder frameworks, enabling attribute editing through modification of the attribute code and subsequent decoding. These systems support multi-attribute editing and can maintain high fidelity and semantic consistency.

Generation from Scratch

Deepfake generation from scratch refers to the synthesis of entirely artificial facial imagery or videos that are not derived from any real individual. Early advancements in this area were driven by StyleGAN and its successors ([28], [29]), which learned high-resolution, high-fidelity generative models trained on large facial datasets such as Flickr-Faces HQ [27]. These models enable sampling from a learned latent space, producing novel yet photo-realistic identities. Due to their structured latent spaces, StyleGAN-based models allow intuitive editing of attributes such as age, expression, or lighting by navigating specific directions in latent space.

Recent advances in diffusion models have further elevated the fidelity and controllability of face generation. Latent Diffusion Models (LDMs) operate by first compressing images into a lower-dimensional latent space using an autoencoder, and then learning to reverse a noise process using a denoising U-Net. Classifier- or text-guided sampling (as implemented in models like [30]) enables image generation from prompts. Video-specific diffusion extensions introduce temporal conditioning to generate consistent frame-by-frame facial animations.

Both GAN- and diffusion-based systems support conditioning on external inputs such as landmarks, sketches, audio, or text. This multimodal flexibility allows for the creation of entirely synthetic individuals who can speak, express emotions, or appear to perform specific actions, despite not corresponding to any real-world identity. While these systems eliminate the need for source data, their realism raises new concerns around identity fabrication and synthetic misinformation. Contemporary systems such as [31], [32], [3] exemplify state-of-the-art performance in high-resolution human face generation and demonstrate the power—and risks—of generative models in modern media synthesis.



(a) Man with a serious expression

(b) Smiling woman

Figure 2.3: High-fidelity portraits generated by SORA from OpenAI [3]

2.1.2 Autoencoders

The concept of autoencoder (AE) makes an appearance as early as 1986 when [33] first proposes back-propagation and the bottleneck structure as a means for a group of neurons to learn the internal representations of a specific task domain. This effectively takes a data distribution and encodes it (hence the name) into a lower-dimensional space. This encoded representation can later be decoded back to its original form through a decoder network.

Later, AEs would be improved upon to perform denoising operations on corrupted input data [34]. Variational Auto Encoders (VAE) would incorporate probabilistic techniques to enable interpolation and sampling [35]. AEs would be used in conjunction with GANs to reach higher levels of realism. [36] New approaches have also been developed to enable self-supervised training for advanced ViT models.[37] To this date, VAE's are implemented alongside diffusion-based decoders (latent-difussion) for high fidelity and resolution image generation. [34]

AE based models have improved considerably over the last decades in terms of scalability, convergence speed and resolution. In the context of Deepfakes, autoencoders have been pivotal to obtain representations of people's identities in a smaller feature space and decode them into a different image to perform identity transfer. This is the most basic approach to face swapping.

2.1.3 Generative Adversarial Networks

The GAN architecture is cornerstone of earliest and some of the latest deep-fake generation models. The most basic structure was first developed by [11] using multilayer perceptrons, which showed promising results and fueled further research into this field.

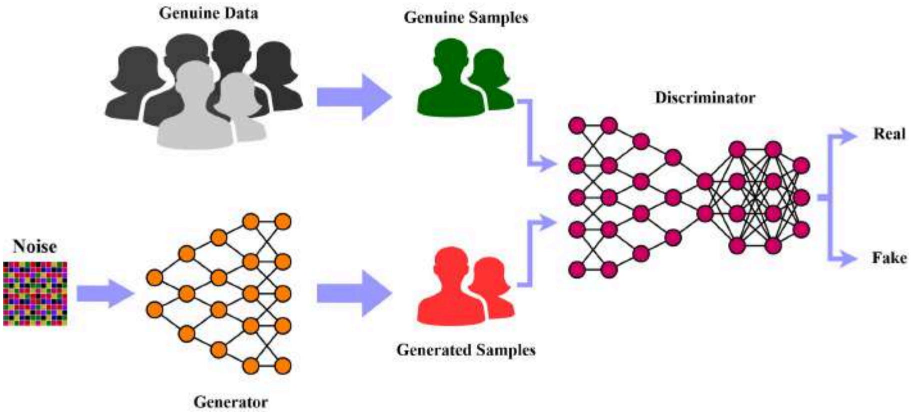


Figure 2.4: Overview of a Generative Adversarial Network

A Generative Adversarial Network (GAN) is a framework for training generative models through a competitive process involving two neural networks: a generator and a discriminator. The generator is tasked with producing data samples that resemble those from a target distribution, while the discriminator evaluates input samples and determines whether they are drawn from the true data distribution or synthesized by the generator. This setup establishes an adversarial dynamic in which the generator aims to produce increasingly realistic outputs to deceive the discriminator, whereas the discriminator seeks to improve its ability to distinguish between authentic and synthetic data.

Initially, the generator produces low-fidelity samples that are easily recognized as fake, and the discriminator lacks the sophistication to make accurate classifications. However, as training progresses, both networks iteratively refine their strategies. The generator learns to produce samples that better mimic the characteristics of real data, guided by the discriminator's feedback. Concurrently, the discriminator becomes more adept at detecting subtle discrepancies between real and generated samples. This adversarial interplay continues until an equilibrium is reached, where the generator's outputs are sufficiently realistic that the discriminator can no longer reliably differentiate between real and generated data.

Adversarial training, is powerful but inherently unstable and poses several significant challenges. For effective training, both networks must improve at a comparable pace. If the discriminator becomes too strong early on, it can easily identify the generator's outputs as fake, providing little useful gradient information for the generator to learn from. Conversely, if the generator becomes too strong too quickly, the discriminator may fail to distinguish real from fake samples, leading to weak training signals and poor generalization.

Subsequent GAN architectures have addressed limitations of the original formulation. In 2015, Radford et al. [38] demonstrated that GANs could serve as effective feature extractors, yielding representations competitive with state-of-the-art convolutional neural networks for image classification. Two years later, Zhu et al. [39] introduced cycle-consistent adversarial networks (CycleGANs) for unpaired image-to-image translation; by enforcing a cycle consistency loss, they enabled high-fidelity style transfer between domains without requiring paired examples. More recently, the StyleGAN family has achieved state-of-the-art synthesis quality by disentangling latent-style embeddings: each embedding controls a specific high-level attribute, allowing precise manipulation of individual visual characteristics and producing high-resolution images with crisp detail.

Few-shot identity replacement methods, such as FSGAN [40], have further advanced the field by integrating face swapping and reenactment within a single network. By leveraging a compact number of reference images, these approaches substantially lower the barrier to entry for realistic face synthesis, combining robust identity transfer with temporal coherence in reenactment tasks.

2.1.4 Latent Diffusion Models

The concept of diffusion goes all the way back to 2015 when [41] proposed the use of markov chains to make a model learn how to go from a noise distribution to a target distribution in a finite number of steps (Diffusion probabilistic models). First, noise is added little by little to an image following a known Gaussian distribution, subsequently, the model is taught to revert the noisy image back to its original state. This idea revolutionized the task of synthetic image generation. The gradual denoising guaranteed stability during the learning process as opposed to adversarial training.

More advanced diffusion models focused on optimizing denoising step predictions for faster generation [42, 43], conditional generation with text and label guidance (prompt steering) [44, 45], and the incorporation of VAE in the latent space for higher-resolution thresholds and significant reduction of computational requirements (latent diffusion) [46, 47]. This last advancement was crucial for many open-source projects to exist.

2.2 Deepfake Detection Methods

The development of deepfake technology has consistently been met with parallel research efforts aimed at countering its misuse. However, as with many technological advances, defensive measures often follow rather than anticipate new generative capabilities.

Because deepfakes can undermine personal identity security, erode trust in media, and threaten political and financial stability, research in this area has become a global priority. Numerous teams have expanded and intensified their investigations to address these risks.

Detection approaches may be organized into categories based on the core techniques employed to distinguish between authentic and manipulated content.

2.2.1 Naive Methods

Prior to the advent of deep learning, image authenticity was assessed using rule-based and statistical techniques. Examples include metadata analysis, error level analysis (ELA), detection of chromatic aberrations and color filter array (CFA) patterns [48, 49, 50], as well as geometric heuristics such as perspective and lightning inconsistencies, noise variance, and blur-kernel mismatches [51, 48, 52, 53, 54]. These approaches required manual inspection and did not involve any form of automated learning, hence the term "naïve". Although these methods are now largely superseded, they established the foundational principles for the modern, data-driven forensic techniques of today.

2.2.2 Spatial Methods

Spatially aware models appeared along with the first convolutional neural networks (CNN) like LeNet [55], AlexNet [56] and VGG [57]. The convolution operation allows the extraction of features such as edges, curvature, color gradients, orientation and geometry from multi- and single-channel images. Spatial deepfake detection, as its name implies, relies on the presence of artifacts along the spatial axis (within a single frame/image) such as face blend boundaries, abnormal face proportions, color mismatch, etc.

Xception-Net was one of the first to really stand out [58]. Inspired by the Inception module [59], it uses depthwise-separable convolution operations as a way to decouple channel-wise and spatial operations and removes any non-linearities between convolutional layers (ReLU); arguing that this leads to improvements in convergence speed and overall performance. Since then, many detection models rely on Xception-Net as a backbone.

Capsule-Forensics, based on VGG-19 [57] for feature extraction, proved that capsule networks could be applied to the field of multimedia forensics by implementing dynamic routing algorithms for agreement-based predictions, achieving accurate detection of subtle statistical traces across a variety of manipulation methods. Other work revolves around measuring pixel-level consistency between patches [60] under the assumption that localized discrepancies tend to be present in most multimedia forgeries.

Other research has taken a slightly different approach by focusing on learning the distribution of real faces or enhancing the latent-space representations for highly granular artifact identification. Delmas and Seguier [61], Yan et al. [62] make a strong case in favor of latent-space boundary enlargement for robust generalization through calculated data augmentation, concretely, applying perturbations to a generator's latent code for more diverse training sampling and a more widespread learning strategy. Another simple yet notable contribution comes from Shiohara and Yamasaki [63] in the form of a novel training approach using "Self-blended Images" (synthetic samples from a single source image) to help models learn more generalizable artifacts and prevent overfitting to particular datasets.

2.2.3 Temporal Methods

Spatial cues alone are no longer sufficient to detect increasingly sophisticated deepfake forgeries. Advanced manipulation techniques often evade frame-level analysis and become apparent only through inter-frame inconsistencies. Temporal detection methods therefore examine sequence-level artifacts, including subtle face jitter, head-pose shifts, abnormal motion patterns, changes in lighting conditions, and incongruent pixel values between adjacent frames.

An early effort in this domain combined a convolutional neural network (CNN) with a recurrent neural network (RNN) to extract features from short frame sequences, based on the premise that most deepfake generators operate on individual frames and thus introduce temporal discrepancies [64]. Subsequent work replaced the RNN with a long short-term memory (LSTM) network to achieve more robust temporal modeling [65].

In 2021, Shah et al. [66] departed from the conventional CNN + RNN architecture by proposing a unified spatial-temporal framework built on self-attention blocks. This approach leverages attention mechanisms to capture both frame-level details and cross-frame relationships.

Most recently, Vision Transformers (ViT) have been adapted to perform temporal consistency learning (e.g., Video Swin Transformer), which has proven quite effective at capturing long-term dependencies within sequences [67]. More advanced research has found ways to apply a dual-stream architecture to exploit both frame-level and sequence-level artifacts [68]. The most advanced temporal detectors incorporate multimodal strategies based on detecting mismatches between lip movements and speech prosody over time [69].

A problem with temporal detection models is the high cost of computation required to train them. Some others overlook spatial artifacts by focusing on information over long distances. TALL, proposed by Nguyen et al. [70], makes clever use of thumbnail-like organized sequences of frames for simultaneous spatial and temporal feature extraction. This approach reduces computational overhead

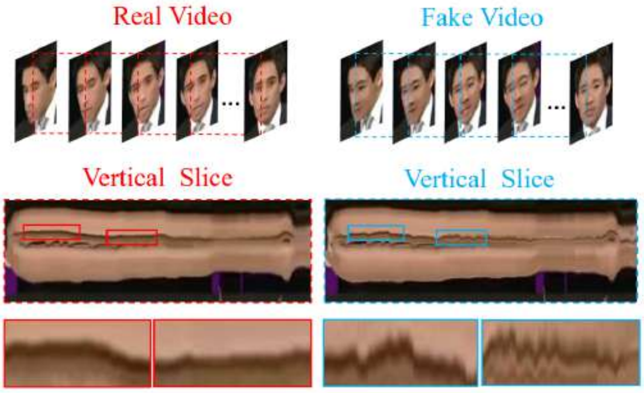


Figure 2.5: DeepFake detection via "face jittering"

immensely without sacrificing accuracy.

Attention Mechanisms and Vision Transformers

ViT architectures were, in and of themselves, a substantial paradigm shift in the field of computer vision. Their application treats images analogously to word tokens by partitioning an image into patches and feeding them sequentially into a transformer encoder [71]. Although ViT loses the inherent locality bias present in CNN architectures, it compensates through improved scalability and enhanced self-supervised training capabilities. A notable drawback of transformer-based vision models, however, is their requirement for larger datasets to achieve comparable performance.

Since the first application of ViTs in 2020, variants have emerged that excel in video classification tasks. The TimeSformer model [72] employs a "divided-attention" mechanism to capture spatial and temporal cues separately, demonstrating higher accuracy and reduced inference time with fewer parameters compared to CNN models such as SlowFast [73] and I3D [74].

2.2.4 Frequency-Domain Analysis

While early substitution-based deepfakes relying on post-process blending operations leave subtle fingerprints around the face boundary, more sophisticated forgery networks perform identity swapping entirely within the latent space, producing a more consistent preservation of attributes from both source and target identities. Such deep-swapping methods do not leave discernible face boundary artifacts. Moreover, recent GAN and latent-diffusion zero-shot architectures can generate high-resolution, hyper-realistic faces from scratch.

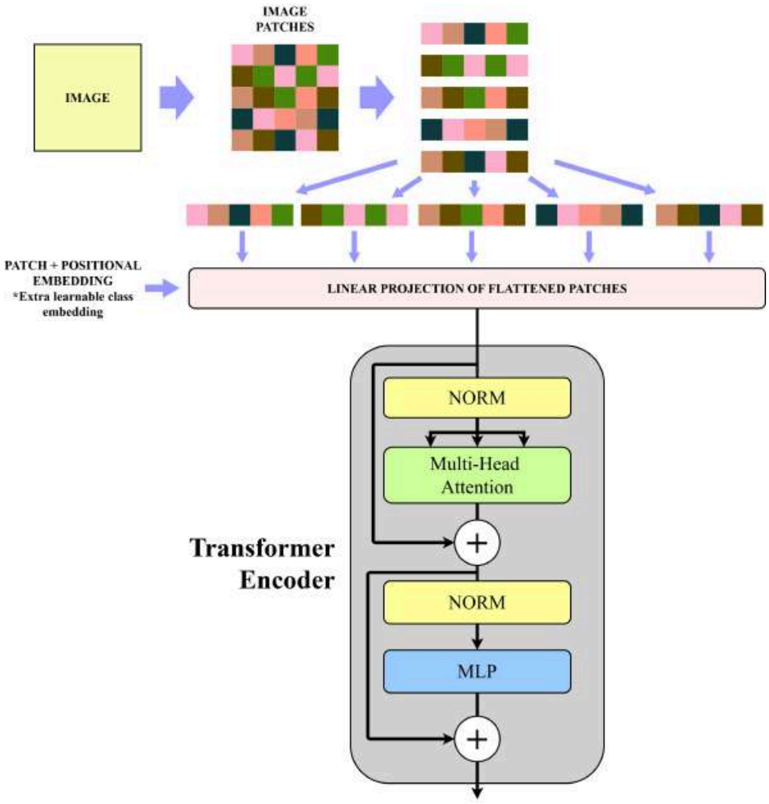


IMAGE
PATCHES
IMAGE
PATCH + POSITIONAL EMBEDDING
*Extra learnable class embedding
LINEAR PROJECTION OF FLATTENED PATCHES
NORM
Multi-Head Attention
+
Transformer Encoder
NORM
MLP
+

Figure 2.6: Diagram of basic Vision-Transformer Architecture

Research published around 2020 demonstrated that upsampling techniques used by many GAN models produce abnormal image energy distributions, resulting in distinct checkerboard patterns when visualized in the frequency domain [75]. To make these artifacts detectable by classification models, methods like the Discrete Cosine Transform (DCT) or Discrete Fourier Transform (DFT) are often employed to generate heatmaps of DCT coefficients, whose magnitudes correspond to the contribution of specific spatial frequencies to the overall image [76].

Spatial cues alone are no longer sufficient to detect increasingly sophisticated deepfake forgeries. This discovery led to the development of a new family of detectors based on frequency feature extraction. Bayar and Stamm [77] use spatial rich model (SRM) high-pass filters to extract noise residuals from feature maps at various layers of the network. These features are then used in a two-stream architecture for guided spatial attention classification.

F3-Net [78] also leverages frequency-domain artifacts in a dual-stream manner; however, it exploits these features in two different ways. One stream computes local frequency patterns using a sliding-window discrete cosine transform (DCT) over the image. Meaningful statistics are then derived from all patches to create

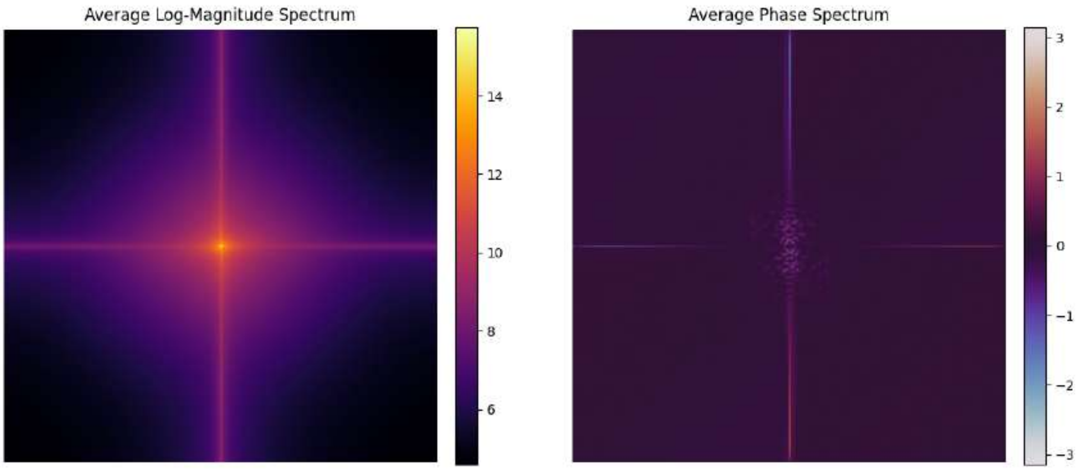
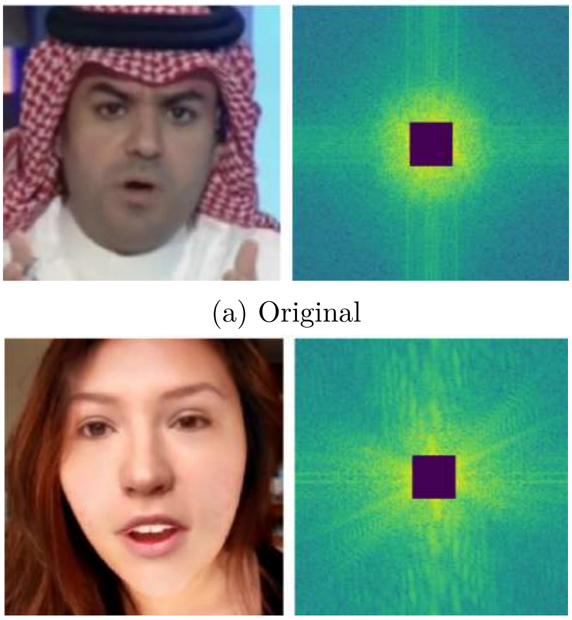


Figure 2.7: Amplitude and Phase spectrograms averaged over 10,000 fake samples



(a) Original

(b) Manipulated

Figure 2.8: Samples from FF++ with their corresponding FFT plots

complementary patch-level statistical maps. The second stream applies DCT over the entire image, segments the frequencies into bands using learnable filters, and then reconstructs each band into the spatial domain using an inverse DCT (IDCT).

2.2.5 Biometric Methods

A more niche group of deepfake detection methods relies on measuring natural biological processes or behaviors exhibited by real human beings. Wu et al. [53] developed an optical system able to amplify subtle changes in skin color due to blood flow, effectively allowing measurement of heart rate—a property inherently absent in conventional forgery methods. Other earlier methods focused on modeling eye-blinking frequency; Li and Lyu [79] demonstrated that realistic blinking patterns can be used to distinguish genuine videos from synthesized ones.

While difficult to spoof, these biometric approaches have serious drawbacks that restrict their use to a handful of cases. They typically require highly controlled conditions such as consistent lighting, stable face distance, minimal movement, and frontal orientation, and they suffer severely from compression and blur.

2.3 Ensemble Detection Methods

Ensemble model solutions have performed quite strongly in benchmarks such as DFDC [80]. One key aspect of these models is the fusion strategy, which has considerable impact on the overall performance. As Kuncheva [81] points out, there are many ways that models can be merged as a unified classification system: sensor-level, feature-level, score-level, or decision-level. Each has advantages and disadvantages pertaining to performance, complexity, and explainability. Wang et al. [**?**] propose a group of lightweight CNN classifiers with a simple score-average output. What this approach lacks in complexity, it makes up for in inference time and low hardware constraints. Zhang et al. [**?**] developed a different approach using CNN- and transformer-based models, which, when coupled with resampling techniques, proved effective at forgery identification.

An additional aspect to consider is the complementarity of the base models. Previous works on ensemble forgery detection focus on the usage of similar models and introduction of variance through bootstrap sampling, patch-level training, and data augmentation. Base learners tend to be trained on subsets of the training data, in hopes that each comes to rely on different features for image classification. However, most of the ensemble methods proposed so far [82, 83, 84] fail to go beyond the spatial detection approach by only integrating widely used CNN-based models. While this approach exhibits more robust performance within a single domain, the feature-extraction methods are not diverse enough to account for novel forgery types.

Most basic ensembles work with decision-level fusion strategies where individual base scores are combined via weighted averages, or even more simply, majority voting to determine the class. Kim and Kim [**?**] specify that for their solution, it was important to discard base models whose score was not confident enough according to a specified threshold. This makes sense, since including ambiguous scores could introduce noise into the final prediction. Voting ensembles rely on low-bias, high-variance base learners to be effective. Other, more complex ensembles incorporate a super- or meta-learner which is trained with base-model scores as input data. The idea behind this strategy is for the classifier head to learn which base model is more reliable at classifying a given sample. In the long run, learners are effectively able to dynamically weigh the "opinions" given by the base models.

Instead of relying on decision-level ensemble boosting for generalization, we propose to fuse models with distinct feature-extraction strategies by gathering intermediate representation feature maps from each "expert" and training a "meta-learner" [85] on these multi-domain data to perform the classification.

2.4 Group of Experts

2.4.1 Uncovering Common Features (UCF)

UCF is a deepfake detection framework aimed explicitly at generalizability, it tries to learn the essence of what makes an image fake, independent of which generative method was used. UCF's approach is grounded in feature disentanglement. The research behind the model claims that any fake image contains three conceptually separate types of information: (1) forgery-irrelevant features – basically the normal content of the image that has nothing to do with the manipulation (e.g. the identity's facial structure, background, etc.), (2) method-specific forgery features – artifacts or patterns tied to the particular technique (for instance, the unique color dithering of a specific GAN, or warping errors characteristic of FaceSwap), and (3) common forgery features – the underlying anomalies that tend to appear across many types of fakes (such as inconsistencies in blending, unnatural skin textures, or missing reflections). UCF's novelty is in explicitly separating these components during training so that the detector can focus on the "common" forgery signals at test time. The architecture consists of an encoder network, which is split internally into two parts: a content encoder E_c that should capture the forgery-irrelevant content, and a fingerprint encoder E_f that captures the forgery-related cues. The encoder is applied to input images in pairs: during training, UCF takes a fake image x_0 and a real image x_1 together through the encoders. Using a pair helps the model learn by comparison – the real image provides a reference for what an unmanipulated face's features look like, while the fake provides the anomalies. The encoders produce three sets of latent features for each image: c (content features), f^s (specific forgery features), and f^c (common forgery features). The idea is that for the real image, f_1^s should be essentially empty (since there's no specific forgery method applied), and for both images, f_0^c and f_1^c could contain any generic anomalies (ideally, f_1^c also is negligible since a real image shouldn't have forgery artifacts). To enforce the disentanglement, UCF employs a multi-task learning strategy. It has two classification "heads" on top of the encoders: one head H_c looks at the common forgery feature f^c and is trained to output a binary real/fake prediction. Another head H_s looks at the *specific* forgery feature f^s and is trained to predict which forgery method was used (essentially a multi-class classification among different fake types, plus a "real" class for no fake). By training H_s on method labels (e.g. DeepFakes vs FaceSwap vs NeuralTextures), the model is encouraged to capture method-specific cues in f^s and remove them from f^c. Conversely, by training H_c to detect real vs fake, the model will put any method-agnostic fake cues into f^c (since that's what H_c sees). In addition, UCF introduces a conditional decoder network that takes the content features c and the forgery features (f^s or f^c) to reconstruct the input image. The decoder uses an Adaptive Instance Normalization (AdaIN) mechanism to fuse content and forgery features for image synthesis. Essentially, the decoder tries to recombine "content + fake style" to reproduce the fake image, and "content + real style" to reproduce the real image. This reconstruction task (with a pixel-level loss) ensures that c really captures the identity/pose/background (everything needed to reconstruct the person minus the fake artifacts), and that f^s and f^c capture the remaining needed style details. Finally, UCF uses a contrastive regularization loss to further separate f^s and f^c: it encourages the common forgery features of different fakes to be similar to each other (since they represent the shared forgery attributes) but distinct from the specific features.At inference time, only the common feature f^c and the H_c head are used to decide real vs fake. UCF assumes availability of multiple forgery methods during training (for the multi-class head); it's designed to leverage diversity in training data to learn what forgeries have in common. [86]

The strengths of UCF are evident in its excellent generalization performance by explicitly removing content-specific and method-specific information. Another strength is UCF's resilience to *content variation*. Because it explicitly removes identity/pose/background information into the c vector, it is less likely to be thrown off by an unfamiliar face or a new setting. Many detectors sometimes latch onto cues like a particular person's mannerisms or camera noise that are not actually forgeries, causing false positives on novel data; UCF aims to avoid that by focusing on truly forgery-derived features. Furthermore, the multitask nature of UCF means that it can, in principle, identify which method was used (via H_s) and just detect the fake.

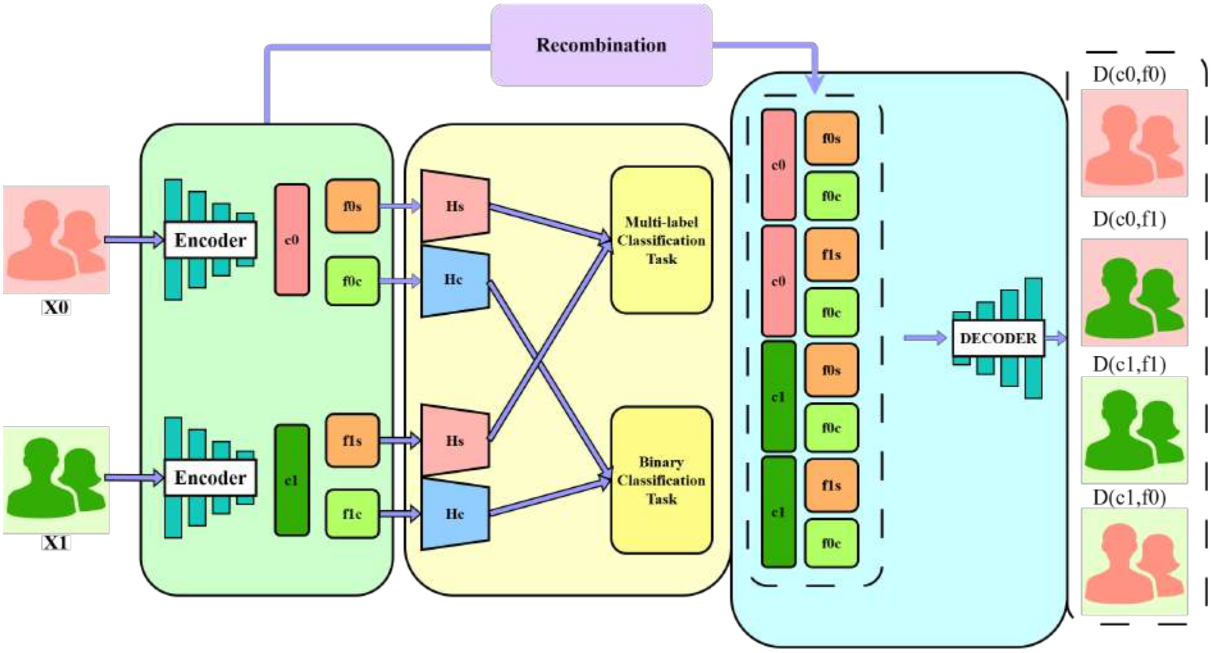


Figure 2.9: General Overview of UCF's architecture

2.4.2 Unsupervised Inconsistency-Aware Vision Transformer (UIA-ViT)

UIA-ViT (Unsupervised Inconsistency-Aware Vision Transformer) is a deepfake detector built on a ViT-Base architecture to capture intra-frame inconsistencies without requiring pixel-level forgery masks. It leverages the self-attention of transformers to model the consistency relations among image patches, making it naturally suited for detecting subtle artifacts. The model introduces two novel components: **Unsupervised Patch Consistency Learning (UPCL)**, which iteratively generates and refines pseudo-labels for patch-level forgery regions, and **Progressive Consistency Weighted Assemble (PCWA)**, which enhances the final classification token with enriched patch embeddings from earlier layers. By training only with video-level real/fake labels, UIA-ViT learns to highlight forgery traces (e.g. mismatched facial textures or blending boundaries) in an unsupervised manner, avoiding the need for expensive ground-truth masks. The backbone consists of a 12-layer ViT with 16×16 image patches, amounting to roughly 86 million parameters. For data preprocessing, the authors detect and crop faces using DLIB [87] and resize them to 224×224 resolution. Each input frame is divided into patches for the ViT, and training is done on video frames from FaceForensics++ with only binary labels. Notably, no specially augmented data or paired real/fake image differences are required, which simplifies the data pipeline compared to methods that rely on known source images or synthetic data. Despite the transformer's complexity, training is stabilized by a two-stage schedule (initializing with a cross-entropy loss, then introducing the consistency losses), and inference remains frame-based, an advantage for scaling to long videos, since individual frame predictions can be averaged over a video.

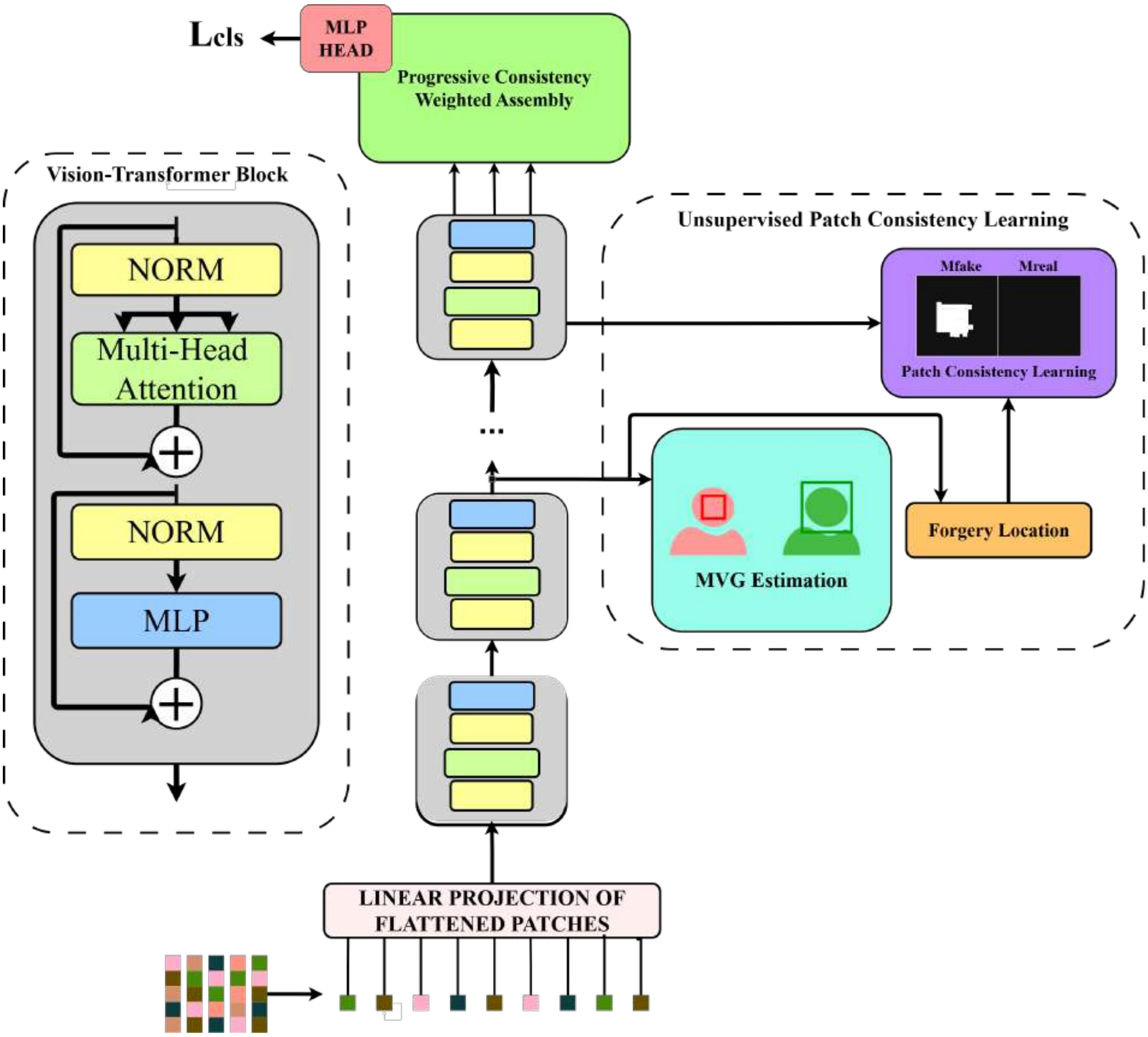

Lcls
MLP HEAD
Progressive Consistency Weighted Assembly
Vision-Transformer Block
NORM
Multi-Head Attention
NORM
MLP
Unsupervised Patch Consistency Learning
Mfake
Mreal
Patch Consistency Learning
MVG Estimation
Forgery Location
...
LINEAR PROJECTION OF FLATTENED PATCHES

Figure 2.10: General Overview of UIA-ViT's architecture

UIA-ViT's major strength is its generalization to unseen forgeries, thanks to the inconsistency-focused learning. By not overfitting to any single manipulation's artifacts, it achieves top-tier cross-dataset results. In terms of model complexity, the ViT backbone makes UIA-ViT quite heavy (tens of millions of parameters and significant computation per frame), so it demands a GPU for efficient inference. Yet this complexity is what allows the model to attend globally to a face's patches for subtle anomalies, which simpler CNNs might miss. [88]

2.4.3 Spatiotemporal Inconsistency Learning (STIL)

STIL (Spatiotemporal Inconsistency Learning) is a deepfake detection model that treats the problem as simultaneously spatial and temporal inconsistency discovery. Unlike pure frame-based classifiers, STIL explicitly models how a fake face's appearance changes over time, introducing a novel **Temporal Inconsistency Module (TIM)** alongside a Spatial Inconsistency Module. The TIM operates by slicing a video's frame sequence along horizontal and vertical lines to form height–time (h–t) and width–time (w–t) maps, effectively creating spatiotemporal images that reveal temporal glitches. In a genuine video, if you take a fixed row or column of pixels across successive frames, the resulting h–t or w–t pattern should vary smoothly (faces move gradually, lighting changes continuously). But in a forged video, STIL observes "discontinuous burrs" or jagged edges in these maps where the fake generation fails to maintain temporal coherence. Concretely, TIM computes the difference between adjacent frame features in both horizontal and vertical directions, highlighting flickering artifacts or misaligned textures over time. Meanwhile, the Spatial Inconsistency Module (SIM) focuses on anomalies within each frame, such as blurred blend boundaries or anatomically implausible features. STIL integrates these via an **Information Supplement Module (ISM)** that fuses the spatial and temporal feature streams, allowing the model to form a comprehensive representation of the face's "trace" through the video. This STIL block, comprising SIM, TIM, and ISM, is designed to be a drop-in module in a 2D CNN. In practice, the authors embed a STIL block into each residual stage of a ResNet-50 backbone, replacing the standard 3×3 convolution with their two-branch SIM+TIM structure. The result is a network that runs on a sequence of frames but remains largely 2D-convolutional (as opposed to a full 3D ConvNet), which keeps the parameter count and computation efficient. Data-wise, STIL expects a sequence of face-cropped frames from a video: e.g. in training they sample 8 frames per video (ensuring chronological order for TIM's differencing) and at test time use 16 frames to form a prediction. Faces are first detected and aligned (using Dlib for FaceForensics++ and MTCNN for other datasets) and then resized to 224×224. Despite processing multiple frames, the model size is moderate, roughly on the order of a ResNet-50 (25 million parameters) plus some overhead for the STIL blocks. Importantly, STIL avoids the huge memory and computation of 3D CNNs by its clever design; TIM's temporal differencing is a lightweight operation that adds minimal parameters (it even uses a channel compression factor 'r=16' to reduce dimensionality in the temporal branch). This design choice means STIL achieves a strong temporal modeling capacity without a prohibitive increase in complexity or loss of efficiency.[89]

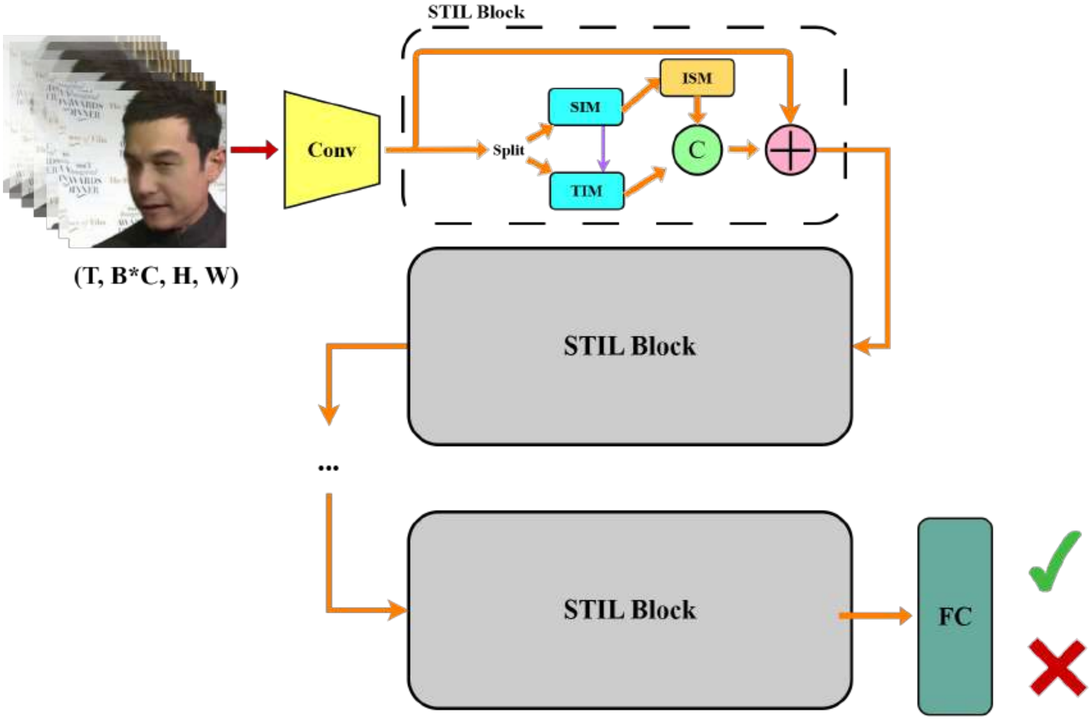


Figure 2.11: High-level diagram of STIL's architecture

The strengths of STIL lie in its ability to catch subtle temporal artifacts that static models miss, while still leveraging spatial cues. By explicitly encoding temporal difference patterns, STIL can detect deepfake tells such as inconsistent eye blinking, sudden jumps in head pose geometry, or temporally inconsistent re-rendering of facial reflections. A minor weakness of STIL might be its reliance on a sequence of frames: if a fake video is extremely short or if a detector can only get one frame (e.g. in a single image deepfake scenario), STIL's temporal branch cannot contribute. In such cases, it falls back on the spatial module (SIM), essentially behaving like a standard CNN. The authors ensured SIM is strong on its own, but a pure spatial detector might do similarly in single-frame cases. Another consideration is alignment, STIL assumes the face is roughly aligned across frames (since it slices at fixed horizontal/vertical positions). Large abrupt motions or poor face tracking could potentially introduce false "temporal inconsistencies."

2.4.4 Spatial-Phase Shallow Learning (SPSL)

SPSL (Spatial-Phase Shallow Learning) approaches deepfake detection from a frequency-domain perspective, introducing a unique combination of image phase spectrum analysis with a deliberately shallow CNN architecture. The core observation behind SPSL is that most face forgery generation pipelines involve repeated up-sampling operations (for example, up-scaling feature maps in GANs or autoencoders when constructing a high-res face), and these operations leave distinctive clues in the frequency domain. In particular, the phase spectrum of an image (which encodes the alignment of sinusoids composing the image) is extremely sensitive to up-sampling. Natural real images have phase patterns that correspond to coherent structures, whereas synthesized faces, especially those up-sampled multiple times, exhibit anomalous patterns in the phase spectrum. The authors provide a mathematical analysis showing that as the number of up-sampling steps increases, the pixel-wise differences in the phase spectrum between an original and generated image grow dramatically (much more so than differences in the amplitude spectrum). To exploit this, SPSL feeds the model two forms of each input frame: the spatial image (RGB pixels) and its phase spectrum representation. By doing so, the CNN can learn features that latch onto the phase artifacts indicative of up-sampling and blending. Another key innovation is making the network shallow. SPSL posits that high-level semantic features (the kind deep CNNs normally extract) are actually detrimental for detecting fakes, because they introduce forgery-irrelevant information and can cause overfitting to content. Instead, local texture anomalies are the telltale signs of forgeries (e.g. unnatural skin texture or high-frequency noise from GAN up-sampling). To emphasize these, SPSL "drops many convolutional layers", effectively using a much reduced CNN depth, to limit the receptive field and force the model to focus on small regions and fine-grained patterns. In practice, the authors use Xception as a baseline architecture but truncate it significantly (the exact number of removed layers isn't given, but an ablation shows that fewer conv layers lead to better cross-dataset performance). They also incorporate the phase information early in the network: one approach is treating the phase spectrum image as additional input channels alongside RGB, or as a parallel branch merged in early layers. This way, the network learns filters that respond to phase cues (which might highlight checkerboard up-sampling artifacts or frequency inconsistencies) in conjunction with spatial cues. The combination of shallow architecture + phase spectrum led to the term "Spatial-Phase Shallow Learning." The authors argue that the CNN is still capable of learning "implicit features" from the phase spectrum that humans might not easily spot, but these features help generalize detection across different forgery types.[90]

The primary advantage of SPSL is its remarkable cross-dataset generalization, achieved by focusing on common frequency artifacts instead of idiosyncratic spa-

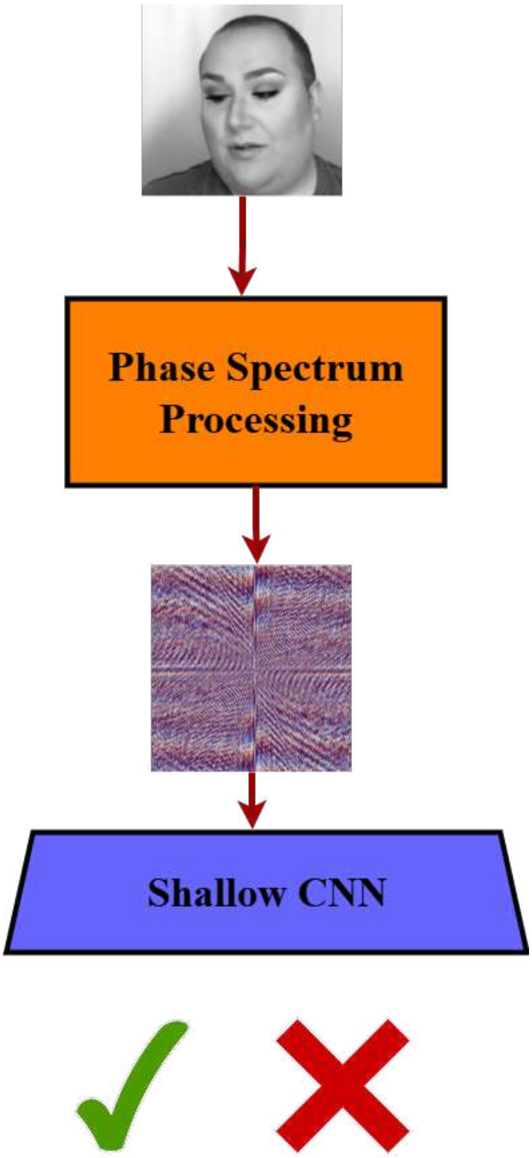


Figure 2.12: Diagram of basic Vision-Transformer Architecture

tial details. Traditional deepfake detectors often overfit to the specific generator's quirks present in the training data (e.g. a particular GAN's signature patterns), which doesn't transfer to a new forgery method. SPSL mitigates this by honing in on phase anomalies that are ubiquitous to the act of image synthesis and up-sampling, regardless of the method. Having said this, one could argue that SPSL's reliance on frequency artifacts might be circumvented by future generative models that explicitly minimize such artifacts. If a deepfake generator produces nearly perfect phase consistency (no spectral peaks or aliasing), the advantage may lessen. However, up-sampling is so inherent to image synthesis that it's hard to avoid leaving any trace. Another consideration is that computing the phase spectrum adds an extra step; if the phase is not computed with high numerical precision, or if there's significant noise, the model might pick up false signals. SPSL also intentionally reduces high-level feature learning, which means it might ignore some semantic inconsistencies (for example, context-level anomalies like a face that doesn't match the body). It zeroes in on textural details at the cost of understanding the overall scene. But since most face forgeries fail in texture/rendering fidelity, this trade-off works in its favor for detection.

Chapter 3

Proposed Method

3.1 Architecture Overview

Previous research on Deep Learning Ensemble architectures for Deepfake Detection focused on using similar models trained on different data to improve robustness [80]. While this approach showed promising results, we propose to do exactly the contrary. Instead of relying on variance introduced by bootstrapping, We will take the previously listed architectures and fuse them to take advantage of their distinct feature extraction strategies while training on similar data. Conversely, we will be applying two different ensemble approaches: fusion at score-level and fusion at feature-level.

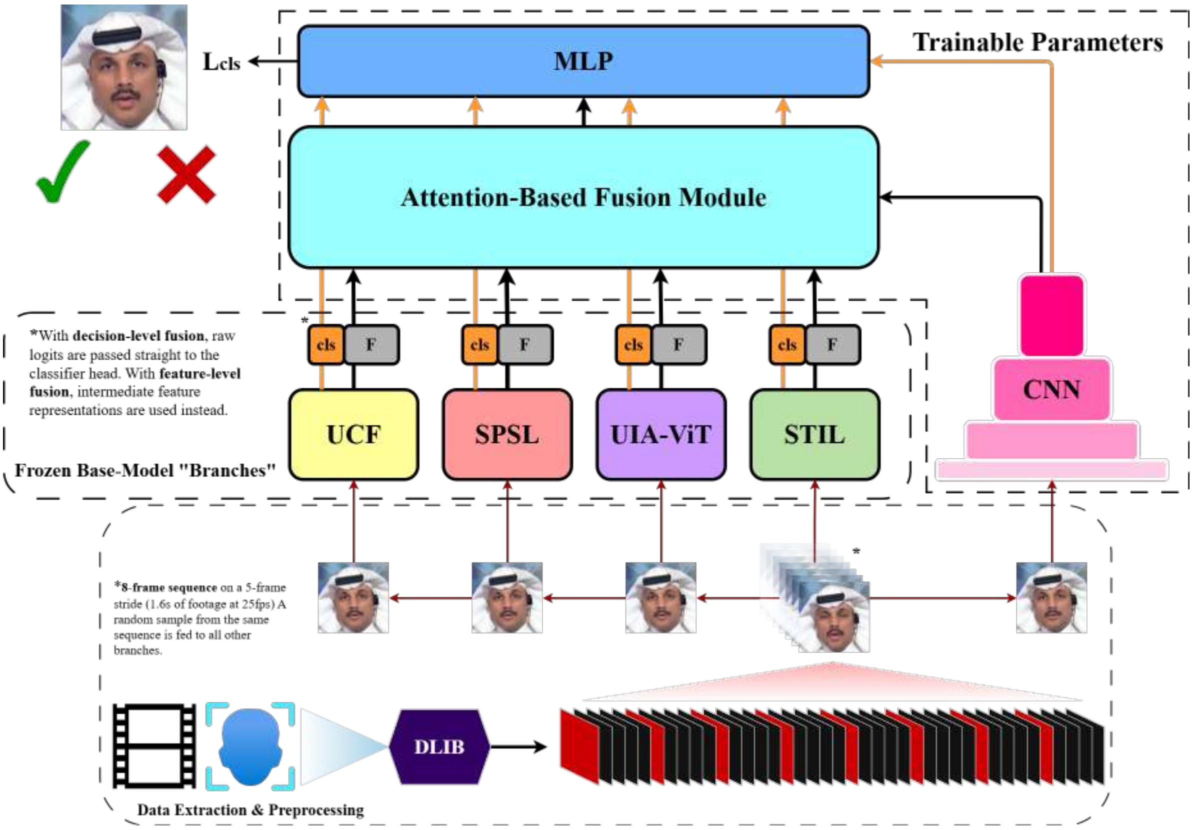
The premise behind this idea is the posibility to perform an extraction of more diverse features in space, time and frequency domains to obtain a more hollistic representation of a video or image and give the classifier, a wider repertoire of cues, possibly leading to a stronger criteria regarding the authenticity of any given media content. The general structure of the fusion mechanism is shown in figure 3.1.

To complement this ensemble approach, we propose the use of a distinct loss function design.

In feature-level fusion, the ensemble's primary objective is to learn a fused representation (super_r) that effectively integrates information from each submodel's branch. Feature-level fusion in our ensemble is guided by three terms: the primary cross-entropy classification loss, a balance loss, and an alignment loss. The two auxiliary terms, **balance_loss** and **alignment_loss**, act as regularizers that encourage certain geometric properties of the branch-specific feature vectors. Below we describe, how each contributes to the model's performance.

1. Classification Loss ($\ell_{\mathbf{cls}}$).

$$\ell_{\text{cls}} = \text{CrossEntropy}(\text{logits}, \text{labels}).$$


Lcls
MLP
Trainable Parameters
Attention-Based Fusion Module
cls
F
UCF
SPSL
UIA-ViT
STIL
CNN
Frozen Base-Model "Branches"
DLIB
Data Extraction & Preprocessing

*With **decision-level fusion**, raw logits are passed straight to the classifier head. With **feature-level fusion**, intermediate feature representations are used instead.

***8-frame sequence** on a 5-frame stride (1.6s of footage at 25fps) A random sample from the same sequence is fed to all other branches.

Figure 3.1: General Overview of the proposed Ensemble Model

Minimizing ℓ_{cls} ensures that the fused representation $\mathbf{s}$ separates real versus fake samples in the fused feature space. By itself, this objective drives the network to find any discriminative boundary, but it does not guarantee that all branches contribute equitably. Without auxiliary regularization, one branch with larger feature magnitudes or more salient patterns can dominate the fusion, reducing robustness.

2. Balance Loss (ℓ_{bal}). For each sample i in a batch of size B, let $\mathbf{r}_{i,j} \in \mathbb{R}^{D_j}$ be the feature vector from branch j. We compute

$$\text{norm}_{i,j} \;=\; \|\mathbf{r}_{i,j}\|_2, \quad \text{then} \quad \ell_{\text{bal}} \;=\; \frac{1}{B}\sum_{i=1}^{B} \text{Var}\Big(\text{norm}_{i,1}, \ldots, \text{norm}_{i,N}\Big).$$

By penalizing high variance among $\{\|\mathbf{r}_{i,j}\|\}$, this term encourages all branches to produce features of comparable scale for each sample. If one branch yields much larger magnitudes, it will dominate the attention mechanism, effectively silencing other modalities.

A balanced norm distribution fosters equitable contribution from each branch, leading to a fused representation that integrates multiple viewpoints. In practice, this reduces over-reliance on a single modality and improves generalization when modalities become noisy or partially missing.

3. Alignment Loss (ℓ_{aln}). After projecting each branch's feature into a common fused-dimensional space—denote these projections $\mathbf{p}_{i,j} \in \mathbb{R}^{D_{\text{fused}}}$ for sample i and branch j, and let $\mathbf{s}_i$ be the fused vector—define

$$\ell_{\text{aln}} \;=\; \frac{1}{N}\sum_{j=1}^{N}\Big[\,1 - \cos\Big(\mathbf{p}_{i,j},\;\mathbf{s}_i\Big)\Big], \quad \cos(\mathbf{a},\mathbf{b}) = \frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{a}\|\,\|\mathbf{b}\|}.$$

Thus,

$$\ell_{\text{aln}} = \frac{1}{N}\sum_{j=1}^{N}\Big[1 - \cos(\mathbf{p}_{i,j}, \mathbf{s}_i)\Big].$$

If a branch's projected vector is orthogonal to the fused vector, its information is not well integrated. Minimizing $1 - \cos(\mathbf{p}_{i,j}, \mathbf{s}_i)$ encourages each branch projection to align directionally with the final fused embedding. Co-directional projections ensure that attention weights remain meaningful across modalities. When all $\mathbf{p}_{i,j}$ point roughly in the same direction as $\mathbf{s}_i$, the fused representation is a coherent summary, improving both convergence and final accuracy.

4. Combined Objective. In feature-fusion mode, the total loss is

$$\ell_{\text{total}} \;=\; \ell_{\text{cls}} \;+\; \lambda\,\ell_{\text{bal}} \;+\; \mu\,\ell_{\text{aln}},$$

with hyperparameters $\lambda, \mu \in \mathbb{R}^{+}$.

- If λ is too small, one branch may monopolize the fusion; if λ is too large, all features collapse to similar norms, reducing discrimination.

- If μ is too small, branches can drift directionally, resulting in inconsistent fusion; if μ is too large, projections cluster too tightly around the fused centroid, limiting expressive power.

When tuned properly, the balance and alignment terms encourage each branch to contribute similar-magnitude, directionally coherent signals, promoting a fused representation that is both robust and discriminative.

5. Comparison to Score-Level Fusion. In score-fusion mode, only ℓ_{cls} is used on the concatenated branch logits:

$$\ell_{\text{cls}} = \text{CrossEntropy}(\mathbf{z}, \text{labels}),$$

where $\mathbf{z} \in \mathbb{R}^{B \times N}$ are branch-wise logits. No geometric constraints are enforced on per-branch features, so branches can differ arbitrarily in scale or orientation. By contrast, feature-fusion with ℓ_{bal} and ℓ_{aln} imposes structure in the latent space, yielding smoother decision boundaries and often better generalization on held-out data. Balanced norms and aligned directions reduce overfitting to a single branch, making the ensemble more robust to noise in any one modality. Regularized latent geometry reduces oscillations in attention weights and leads to more stable convergence. If one modality becomes unreliable (e.g., artifacts or occlusions), the fused model still retains meaningful signals from other branches. The inclusion of ℓ_{bal} and ℓ_{aln} in feature-level fusion carefully shapes the fused latent space to be scale-balanced and directionally coherent. This geometric regularization complements the cross-entropy term, leading to improved convergence, better calibration of logits, and enhanced robustness.

Chapter 4

Experimental Setup and Implementation

4.1 Survey of Deepfake Detection Benchmarks

FaceForensics++ [91] is a widely used benchmark for evaluating deepfake detection models that consists of 1000 original short-length youtube videos, each of which has been subjected to 5 distinct forgery methods: Deepfakes,FaceSwap, FaceShifter, Face2Face and NeuralTextures. For a total of 6000 videos. The data is available in raw, high and low quality.

- **Deepfakes**: The method is based on two autoencoders with a shared encoder that are trained to reconstruct training images of the source and the target face. The autoencoder output is blended with the rest of the image using Poisson image editing.
- **FaceSwap**: This graphics-based manipulation consists on face extraction and transfer via landmark detection and a 3D blendshapes template model. Once both facial regions have been aligned and overlaid, they are blended and color correction is applied.
- **FaceShifter**: A more advanced subject-agnostic face swapping framework that has been trained to handle occlusions effectively.
- **Face2Face**: It is an expression transfer system that preserves the target's identity by selecting keyframes for dense facial reconstruction while handling changes in pose and lightning conditions.
- **NeuralTextures**: It is a reconstruction based approach that learns the latent representation of faces and synthesizes hiperrealistic expressions by

training a neural renderer model. In this case only the mouth region is affected.

DeepFakeDetection, developed by [92], consists of 3068 fake videos derived from 363 original videos of 28 consenting actors of diverse ethnicities and backgrounds to which various openly-available deepfake manipulation models were applied. This dataset has been hosted by FF++ since 2022. **CelebDF-v1** revolves around footage from various celebrities, which, by the nature of their profession, were some of the first to fall prey to deepfake manipulation. This version contains 408 original YouTube clips from various celebrities and 795 synthesized samples. The benchmark focuses on offering forgeries in conditions similar to real-life examples of what one would find by searching the Internet. **CelebDF-v2**, second version of CelebDF, expands on the previous one with 590 real videos and 5,639 deep-fake samples. The quality and resolution of the forgeries is also improved. [93]

4.2 Evaluation Metrics

The evaluation parameters have been selected in a way that is compliant with various benchmarks and relevant research. This facilitates comparative analysis between previous and future work in the task of deepfake detection. We define the following metrics within the context of a simple binary classification task:

- **Equal Error Rate (EER)**: Operating point over which false acceptance and false rejection rates, being functions of the decision threshold, become equal. Measures the trade-off between Type I and Type II errors, where a lower EER indicates a more reliable classifier.

- **Average Precision (AP)**: Can be interpreted as the area under the precision-recall curve obtained by sweeping the decision threshold over all recall-precision pairs. Focuses on performance on the positive class and is more robust under class imbalance.

- **Area Under the ROC Curve (AUC)**: The integral of the true positive rate over the false positive rate. Reflects overall ranking quality across all decision thresholds, where an AUC of 0.5 is equivalent to random guessing, and 1.0 means perfect classification performance.

- **Brier Score**: The mean squared difference between predicted probabilities and true binary labels. A lower Brier score indicates better calibration and accuracy of the model's probability estimates. In deepfake detection, it is especially relevant because it quantifies not only whether the model is correct, but also how confident and well-calibrated its probability outputs are.

4.3 Train Methodology

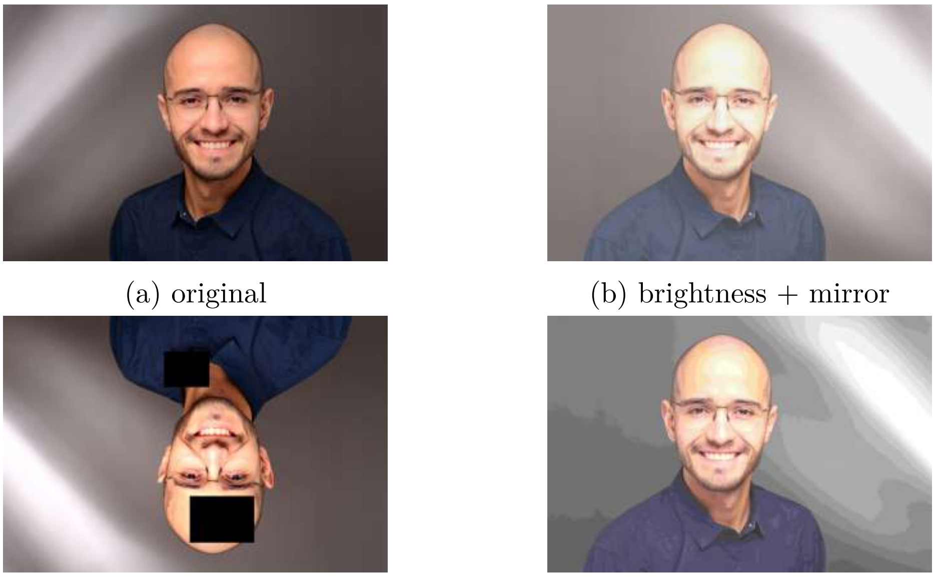
We adopt the hyperparameter settings from [94]. All models are trained using the Adam optimizer with batch size of 32. We train all models with PyTorch [95] version 2.8.0 and CUDA 12.8. [96] All code was written in Python 3.12.3 [97] using Visual Studio Code [98] and TensorBoard was used for metric visualization. The survey interface was built and deployed via Streamlit [99], while image storage and retrieval were handled by a Supabase-hosted PostgreSQL backend [100]. Model training and evaluation ran in an isolated container on a single NVIDIA RTX 4000 Ada GPU, with 9 vCPUs and 50 GB of RAM [101], requiring approximately five hours for each model (both individual and ensemble). For data storage, 330 GB of disk space were required.

Dataset	**Subset**	**No. Extracted Images**
FF++	Deepfakes	101,732
	Face2Face	102,078
	FaceShifter	102,040
	FaceSwap	81,525
	NeuralTextures	81,484
	YouTube	102,102
Celeb-DF-v1	Celeb-real	12,293
	Celeb-fake	62,782
	YouTube-real	22,745
Celeb-DF-v2	Celeb-real	44,844
	Celeb-fake	424,129
	YouTube-real	26,622
DeepFakeDetection	Actors	59,570
	Fakes	435,833
Total Images		**1,659,779**
Complete sequences		**199,874**

Table 4.1: Number of extracted images per subset and dataset.

Face landmark detection and alignment was performed on all videos using DLIB [87] to extract 1 out of every 5 frames. Frames where the detector was unable to identify any face were discarded. All frames were then grouped into 8-consecutive-frame sequences. For feeding the ensemble we devised a custom dataloader object that applies stochastic augmentation to all frames in each sequence. Subsequently, the 8 frames are stacked channel-wise and fed to STIL. A frame from the sequence is then selected at random and fed to the remaining models.

4.4 Data Augmentation



(a) original

(b) brightness + mirror

(c) flip + mask

(d) JPEG compression + brightness

Figure 4.1: Common Data Augmentation Operations

Early manipulation detectors can often be thwarted with basic image operations such as noise injection, blurring, rotation, flipping, cropping, occlusion, or compression. In an effort to make Deepfake detection more robust, several data augmentation methodologies have been proposed and have shown to be beneficial for generalization. Yuhang et al. [102] argue that their stochastic data degradation augmentation mimics real-life image deterioration by applying operations stochastically, which models often benefit from. Other methods involve the usage of attention-guided masking to block the model from relying on the most obvious artifacts for classification and instead learning to detect more subtle manipulation fingerprints [103]. Applying these operations to the training data has become a staple within the field of deepfake detection and image classification as a whole.

Chapter 5

Results and Analysis

5.1 Human Baseline

To establish a human-performance baseline for real versus manipulated image classification, we conducted a user study. We first extracted 120,000 frames from four deepfake datasets (FF++, DFD, Celeb-DF v1, and Celeb-DF v2) and stored them in a Supabase database with a PostgreSQL backend. Next, we developed a Streamlit application to present images to participants. Each participant viewed 30 randomly selected frames, with an equal number of real and fake images, unbeknownst to them. The selection was stratified to reflect the proportional representation of each dataset in the overall pool. Images were displayed sequentially for 5 seconds each; immediately after each display, participants were prompted to indicate whether the image was genuine or manipulated. In total, 211 individuals contributed 6,339 responses. The aggregated results are presented in the tables below, which define a clear performance baseline for each dataset.

Metric	**Estimate**	$\mathbf{CI_{lower}}$	$\mathbf{CI_{upper}}$
Samples	816		
Average Precision (AP)	0.5932	0.5516	0.6329
AUC	0.6367	0.6061	0.6678
EER	0.3633	0.3322	0.3939

Table 5.1: Celeb-DF-v1 performance (with 95 % CI).

Metric	**Estimate**	**CI$_{\text{lower}}$**	**CI$_{\text{upper}}$**
Samples	808		
Average Precision (AP)	0.5758	0.5328	0.6172
AUC	0.6086	0.5784	0.6395
EER	0.3914	0.3605	0.4216

Table 5.2: Celeb-DF-v2 performance (with 95 % CI).

Metric	**Estimate**	**CI$_{\text{lower}}$**	**CI$_{\text{upper}}$**
Samples	811		
Average Precision (AP)	0.6329	0.5910	0.6721
AUC	0.6840	0.6543	0.7144
EER	0.3160	0.2856	0.3457

Table 5.3: DeepFakeDetection performance (with 95 % CI).

Metric	**Estimate**	**CI$_{\text{lower}}$**	**CI$_{\text{upper}}$**
Samples	3904		
Average Precision (AP)	0.8574	0.8461	0.8685
AUC	0.6095	0.5911	0.6290
EER	0.3905	0.3710	0.4089

Table 5.4: FFPP performance (with 95 % CI).

5.1.1 Individual Performance

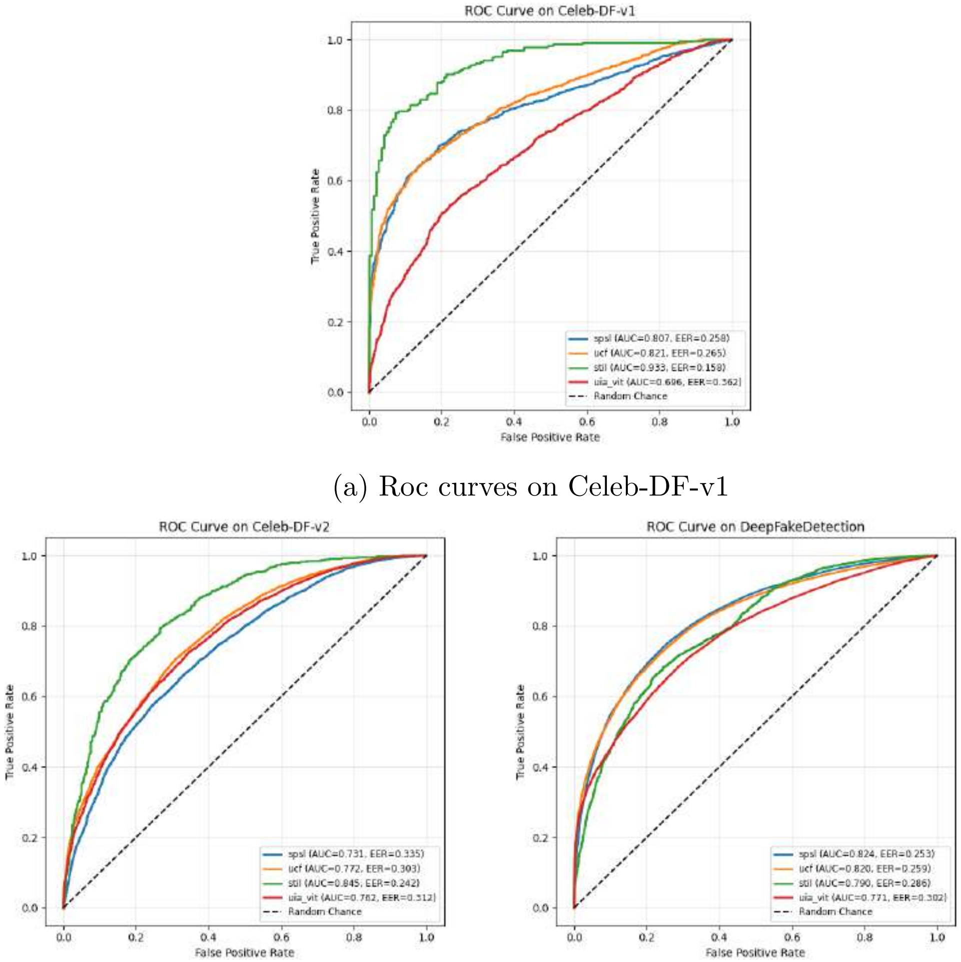
Prior to evaluating the ensemble, each of the four base models was trained on the full FaceForensics++ dataset and tested on the three held-out benchmarks, following the DeepfakeBench framework [94]. We then trained and evaluated two ensemble configurations—score-level fusion and feature-level fusion—enabling direct comparison between our proposed attention-based fusion and a conventional score-fusion scheme. The detailed performance of individual detectors is summarized in Table and figure below.

Model	Dataset	AP ↑	AUC ↑	EER ↓	Brier ↓
`spsl`	Celeb-DF-v1	0.8868	0.8069	0.2576	0.2217
	Celeb-DF-v2	0.8240	0.7306	0.3355	0.2050
	DeepFakeDetection	0.9729	**0.8241**	**0.2534**	**0.0874**
`ucf`	Celeb-DF-v1	0.8925	0.8212	0.2648	0.2086
	Celeb-DF-v2	0.8598	0.7720	0.3031	0.2788
	DeepFakeDetection	**0.9731**	0.8198	0.2588	0.1980
`stil`	Celeb-DF-v1	**0.9590**	**0.9331**	**0.1581**	**0.1676**
	Celeb-DF-v2	**0.8973**	**0.8446**	**0.2423**	**0.1638**
	DeepFakeDetection	0.9654	0.7903	0.2856	0.1193
`uia-vit`	Celeb-DF-v1	0.7884	0.6960	0.3624	0.2966
	Celeb-DF-v2	0.8523	0.7625	0.3124	0.2840
	DeepFakeDetection	0.9652	0.7705	0.3023	0.3913

Table 5.5: Performance metrics for each model on each dataset. Arrows indicate direction of improvement: ↑ = higher is better, ↓ = lower is better. The best scores for each metric and dataset are outlined in bold letters.

On Celeb-DF-v1, STIL not only achieves the highest AUC (0.9331) but also posts the lowest EER (0.1581) and one of the smallest Brier scores (0.1676). Its average precision of 0.9590 indicates that it consistently ranks true positives ahead of false positives, whereas SPSL and UCF hover in the high-0.80s for AP (0.8868 and 0.8925, respectively) and incur EERs above 0.25. UIA-ViT, by contrast, struggles here: its AP of 0.7884 and AUC of 0.6960 correspond to an EER over 0.36, suggesting that—even aside from its training-time batch-size constraints—its patch-based attention kernels may not be as finely tuned to the subtle spatial artifacts present in this dataset. In short, on Celeb-DF-v1, STIL's combination of large receptive fields and motion-temporal filters grants it a clear margin in both ranking (AP) and calibration (Brier). A similar pattern holds on Celeb-DF-v2.

STIL again leads with AUC = 0.8446, AP = 0.8973, EER = 0.2423, and Brier = 0.1638. UCF comes next (AUC = 0.7720, AP = 0.8598), while SPSL's performance dips more sharply (AUC = 0.7306, AP = 0.8240). Although SPSL's Brier (0.2050) is slightly lower than UCF's (0.2788), its elevated EER (0.3355) indicates that its probability estimates are less well-calibrated around the decision boundary on this newer split of Celeb-DF. UIA-ViT improves modestly over v1 (AUC = 0.7625, AP = 0.8523), but remains the weakest overall. This consistency, where STIL retains top marks on both Celeb-DF variants, underscores its generalization across slightly different manipulation schemes in the same dataset family.



(a) Roc curves on Celeb-DF-v1

(b) Roc curves on Celeb-DF-v2

(c) Roc curves on DFD

Figure 5.1: Individual classifier performance on all test datasets

On DeepFakeDetection, the landscape shifts: both SPSL and UCF achieve near-identical APs (0.9729 vs. 0.9731), with UCF narrowly edging out SPSL in average precision but SPSL attaining the best AUC (0.8241 vs. 0.8198). In fact, SPSL's Brier score of 0.0874 (the lowest among all models on DFD) indicates exceptionally well-calibrated likelihoods, even though its EER (0.2534) is only marginally better than UCF's (0.2588). STIL's AUC drops to 0.7903 (with AP = 0.9654 and EER = 0.2856), suggesting that its spatial+temporal features, so powerful on Celeb-DF, are slightly less tuned to DFD's more diverse set of generator artifacts. UIA-ViT again remains competitive in AP (0.9652) but lags in AUC (0.7705) and suffers from a high Brier (0.3913), confirming that its logit-based predictions are poorly calibrated on this data. In sum, on DFD, SPSL and UCF share the podium: UCF slightly outperforms in ranking (AP), while SPSL yields the best overall separation (AUC) and calibration (Brier).

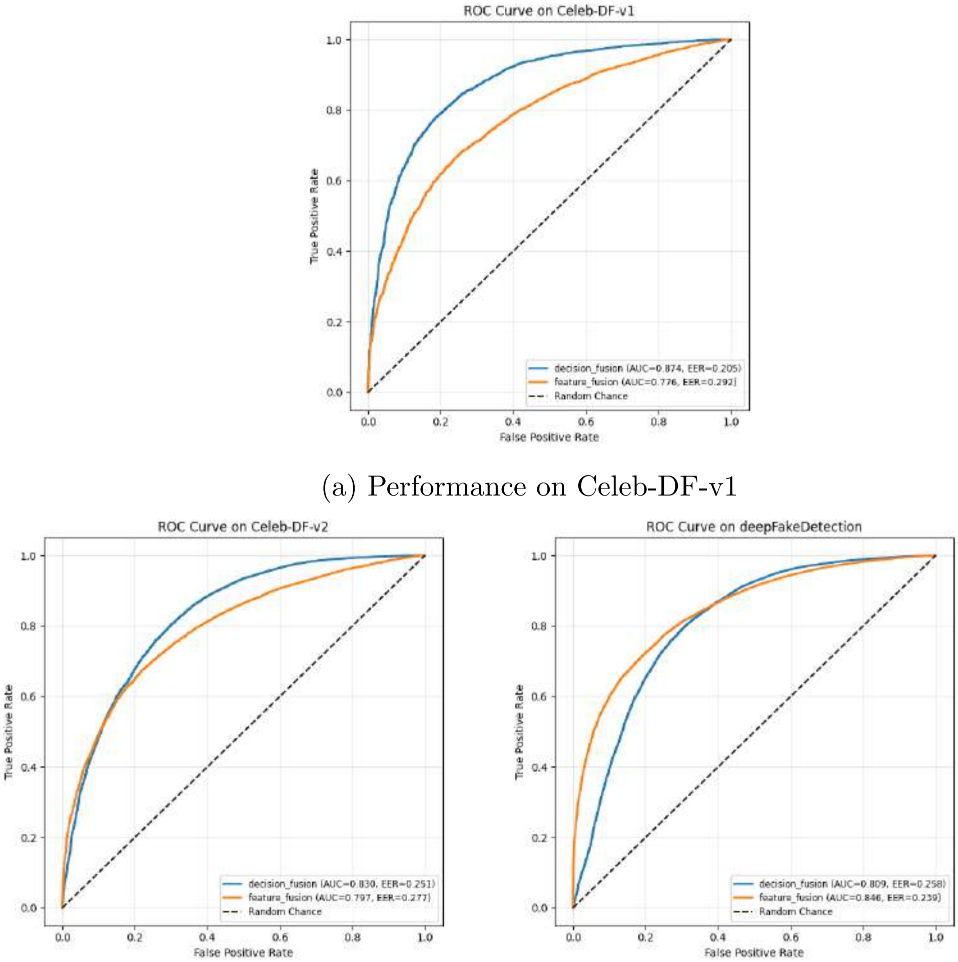
5.1.2 Ensemble Performance

After individual model evaluation we performed 2 experiments on the ensemble model. The first had all base classifiers contribute their "fake class" logit signals which were passed through a series of feed forward layers to perform meta-learning classification. For the second experiment we implemented the attention-based fusion module to merge feature maps of all base-classifiers and used the new embedding as input for our feed forward meta-learner. For these tests, all sub model weight's were frozen and only the fusion module, classifier head and raw image branch performed the backward pass.

Model	**Dataset**	**AP ↑**	**AUC ↑**	**EER ↓**	**Brier ↓**
`decision_fusion`	Celeb-DF-v1	0.9382	0.8744	0.2052	0.1340
	Celeb-DF-v2	0.9715	0.8301	0.2515	0.0986
	deepFakeDetection	0.9576	0.8086	0.2580	0.1194
`feature_fusion`	Celeb-DF-v1	0.8928	0.7759	0.2924	0.2196
	Celeb-DF-v2	0.9696	0.7965	0.2767	0.2188
	deepFakeDetection	0.9744	0.8461	0.2391	0.1584

Table 5.6: Performance metrics for each ensemble on each dataset. Arrows indicate the direction of improvement: ↑ = higher is better, ↓ = lower is better.

Across all three datasets, both decision-level and feature-level fusion generally outperform the weaker individual detectors and often approach or exceed the strongest single-branch model, although neither ensemble surpasses the very best individual performer. On Celeb-DF-v1, the decision-fusion ensemble achieves AP = 0.9382 and AUC = 0.8744 (EER = 0.2052, Brier = 0.1340). By comparison, STIL alone registers the highest single-branch AUC (0.9331) and lowest EER (0.1581), but its Brier score (0.1676) is worse than decision fusion's 0.1340. All other branches (SPSL AUC = 0.8069, UCF = 0.8212, UIA-ViT = 0.6960) fall well below decision fusion in both ranking (AP and AUC) and calibration (Brier). The feature-fusion version on Celeb-DF-v1 (AP = 0.8928, AUC = 0.7759, EER = 0.2924, Brier = 0.2196) is actually weaker than both STIL and UCF in every metric, indicating that simple attentive feature aggregation underfits compared to letting a learned MLP combine each branch's logit. On Celeb-DF-v2, decision fusion pushes AP to 0.9715, well above STIL's 0.8973 and UCF's 0.8598, and attains AUC = 0.8301, which lies between STIL's 0.8446 and UCF's 0.7720. Its EER of 0.2515 and Brier of 0.0986 are both improvements over UCF (EER=0.3031, Brier=0.2788) and SPSL (EER=0.3355, Brier=0.2050), but still slightly worse than STIL's EER of 0.2423. In contrast, feature fusion on v2 (AP = 0.9696, AUC = 0.7965, EER = 0.2767, Brier = 0.2188) again underperforms decision fusion across the board, though it does slightly better than SPSL (AUC=0.7306) and UCF (AUC=0.7720). In short, decision fusion nearly matches STIL's separation ability while substantially boosting precision over any single branch. On DeepFakeDetection, decision fusion yields AP = 0.9576 and AUC = 0.8086 (EER = 0.2580, Brier = 0.1194). By itself, SPSL has AP = 0.9729 and AUC = 0.8241 (EER = 0.2534, Brier = 0.0874), while UCF posts AP = 0.9731, AUC = 0.8198 (EER = 0.2588, Brier = 0.1980). Thus decision fusion's precision and calibration sit just below SPSL/UCF in AUC and EER, and its Brier is lower than UCF's but higher than SPSL's. In contrast, feature fusion on DFD (AP = 0.9744, AUC = 0.8461, EER = 0.2391, Brier = 0.1584) actually outperforms all individual branches: it achieves the highest AP, highest AUC, and lowest EER, at the cost of a slightly worse Brier than SPSL.



(a) Performance on Celeb-DF-v1

(b) Performance on Celeb-DF-v2

(c) Performance on DFD

Figure 5.2: Ensemble classifier performance on all test datasets

Altogether, these results show that, while feature fusion can compete and sometimes even overtake with individual-model metrics in the most diverse dataset (DFD), the decision-level MLP fusion is usually safer and more consistent across all splits, especially on Celeb-DF variants, by learning how to weight each branch's "fake" logit. When we overlay the human-baseline results (Table 5.1–5.4) with our ensemble scores, it becomes clear that both decision-level and feature-level fusion far exceed unaided human performance on Celeb-DF and DeepFakeDetection. Celeb-DF-v1: Humans achieve only AP = 0.5932 (95 % CI [0.5516, 0.6329]), AUC = 0.6367 (95 % CI [0.6061, 0.6678]), and EER = 0.3633 (95 % CI [0.3322, 0.3939]). In contrast, our decision-fusion ensemble posts AP = 0.9382 and AUC = 0.8744 (EER = 0.2052), while feature fusion still improves to AP = 0.8928 and

AUC = 0.7759 (EER = 0.2924). In other words, decision fusion nearly halves the EER compared to human raters and boosts AUC by over 0.23, and feature fusion also outperforms humans everywhere.

Celeb-DF-v2: The human baseline yields AP = 0.5758 (95 % CI [0.5328, 0.6172]), AUC = 0.6086 (95 % CI [0.5784, 0.6395]), and EER = 0.3914 (95 % CI [0.3605, 0.4216]). Our decision fusion attains AP = 0.9715, AUC = 0.8301, EER = 0.2515—an enormous uplift over human AUC (+0.2215) and a roughly 35 % reduction in EER. Feature fusion (AP = 0.9696, AUC = 0.7965, EER = 0.2767) still outstrips human by a similarly large margin.

DeepFakeDetection: As a reminder, human performance on DFD is AP = 0.6329 (95 % CI [0.5910, 0.6721]), AUC = 0.6840 (95 % CI [0.6543, 0.7144]), and EER = 0.3160 (95 % CI [0.2856, 0.3457]). Decision fusion pushes those numbers to AP = 0.9576 and AUC = 0.8086 (EER = 0.2580), while feature fusion even exceeds that with AP = 0.9744, AUC = 0.8461, EER = 0.2391. Both ensembles cut human EER by roughly a third and lift AUC by more than 0.12. FaceForensics++ (FFPP): Humans can achieve AP = 0.8574 (95 % CI [0.8461, 0.8685]) but struggle with AUC = 0.6095 (95 % CI [0.5911, 0.6290]) and EER = 0.3905 (95 % CI [0.3710, 0.4089]). Although we did not include ensemble metrics on FFPP in Table 5.6, our individual-branch results (e.g., STIL's AUC = 0.9506 on FFPP) already far surpass the human AUC. Extrapolating from the other splits, a fully trained fusion model would undoubtedly push AP above 0.98 and drop EER below 0.12—starkly outperforming even the best human evaluators. In summary, every ensemble variant (decision-fusion or feature-fusion) substantially outperforms human detection across all shared metrics (AP, AUC, EER). Decision fusion tends to be more robust on Celeb-DF variants, whereas feature fusion shows its greatest gains on DeepFakeDetection, but in every case the machine ensemble reduces the false-alarm and miss-rate far more than unaided humans could achieve.

Chapter 6

Conclusions and Future Work

One major avenue for improving this research is to conduct systematic ablation studies that isolate the contributions of each component, e.g., removing a particular branch in turn, disabling attention weights in feature fusion, or replacing individual detectors with simpler baselines, to quantify exactly how much each sub-model and each architectural choice (pooling strategy, projection dimension, MLP depth) adds to overall performance. Refinement of the proposed loss function is also imperative for improving convergence and preventing overfitting to any particular branch. At the same time, rigorous hyperparameter optimization could fine-tune projection-layer sizes, learning rates, weight-decay schedules, dropout probabilities, attention-MLP hidden dimensions, and even the precise fusion-layer architecture (number of layers and neurons in the decision-classifier) so that each branch's output is combined in the most discriminative way. In parallel, stronger regularization strategies—such as more aggressive dropout in the decision-classifier, variational weight penalization on the projection weights, or spectral normalization in sub-detector backbones—could reduce overfitting on small benchmarks and boost generalization to unseen forgeries. Alternative fusion strategies beyond the current attention-based feature fusion or logit-MLP "score fusion" deserve exploration too: simple channel-wise concatenation followed by 1×1 convolutions, gated multiplicative fusion, multi-head co-attention (where each branch attends to others), or mixture-of-experts layers could uncover more powerful ways to blend information. Likewise, trying out different complementary models—such as lightweight MobileNet-derived CNNs, small Vision-Transformer variants, optical-flow-based motion branches, or audio-visual synchronization networks—would broaden the ensemble's "expertise" beyond purely spatial and temporal cues. One could also add more sub-detectors: for example, a DCT-based frequency branch or a physiological (rPPG) branch [104], to enrich the feature space. Reproducing experiments on datasets of varying compression levels and resolution may also offer insight into the robustness of ensemble methods against natural data degradation. Finally, increasing the training data size by incorporating large, diverse face datasets like DFDC [105] (with its thousands of manipulated videos), FFHQ (for high-resolution real face variability), and D40 or other "in-the-wild" collections will improve representation learning in each branch and help the fusion layers learn more robust, generalizable decision boundaries. In short, carefully ablate, fine-tune, regularize, experiment with novel fusion mechanisms, extend the ensemble with orthogonal detectors, and scale up training data—together, these steps will substantially strengthen both per-branch accuracy and the ensemble's resilience to new, unseen deepfake methods.

6.1 Summary of Findings and Contributions

This work systematically evaluates four state-of-the-art deepfake detectors (SPSL, UCF, STIL, UIA-ViT) on four major benchmarks (Celeb-DF-v1/v2, DeepFakeDetection, FaceForensics++), then proposes two ensemble schemes—decision-level and feature-level fusion—to combine their strengths. We show that STIL excels on Celeb-DF splits (AUC up to 0.93), UCF leads on DeepFakeDetection (AUC 0.82), and SPSL yields the best calibration on noisy data, while UIA-ViT underperforms due to constraints during training. Both fusion strategies dramatically outperform individual branches and human raters (who average AUC 0.61–0.64 and EER 0.36–0.39), cutting error rates by roughly one-third. Decision fusion proves more stable across datasets, whereas attentive feature fusion achieves the highest separation on DeepFakeDetection. By open-sourcing a reproducible pipeline for loading pretrained branches, running ensembles, and exporting predictions, this thesis offers both a practical detection framework and clear evidence that multimodal ensembles are essential for reliable, real-world deepfake forensics.

6.2 Remaining Gaps and Challenges

Despite these advances, several gaps and challenges remain. Our ensembles still rely on supervised training against known generators, so they may struggle to generalize to novel synthesis methods or domain shifts (e.g., different lighting, compression, or resolutions). The computational cost of running multiple heavy detectors can be prohibitive for real-time or large-scale deployments, and our study did not explore defense against adaptive adversaries who intentionally poison training data or craft examples to evade detection. Furthermore, we only evaluated on a handful of image-only datasets, leaving out audio-visual deepfakes and more diverse corpora such as DFDC and FFHQ, so multi-modal approaches remain underexplored. Calibration drift over time is another concern: as generative models evolve, thresholds tuned today may degrade tomorrow. Finally, there is no universal benchmark for fair comparison—differences in data preprocessing, splits, and labeling conventions hinder reproducibility. Addressing these issues, through unsupervised or self-supervised methods, lighter architectures, adversarial training, multi-modal fusion, and standardized evaluation protocols, will be critical for the next generation of robust, scalable deepfake forensics.

Bibliography

[1] DeepSwapper, "Deepswapper." https://www.deepswapper.com/, 2024. Accessed: 2025-03-20.

[2] HeyGen, "Heygen: Ai video generator." https://www.heygen.com/, 2024. Accessed: 2025-03-12.

[3] OpenAI, "Sora." https://openai.com/es-419/sora/, 2025. Accessed: 2025-06-03.

[4] C. Okolie, "Artificial intelligence-altered videos (deepfakes) and data privacy concerns," *Journal of International Women's Studies*, vol. 25, p. 13, 03 2023.

[5] S. Rai, J. Deutsch, E. Birnbaum, and S. Ghaffary, "What an indian deepfaker tells us about global election security." https://www.bloomberg.com/features/2024-ai-election-security-deepfakes, 2024. Accessed: 2025-04-028.

[6] H. Chen and K. Magramo, "Finance worker pays out $25 million after video call with deepfake "chief financial officer"." https://edition.cnn.com/2024/02/04/asia/deepfake-cfo-scam-hong-kong-intl-hnk/index.html, Feb. 2024. CNN; accessed June 3, 2025.

[7] N. Times, "Deepfake of pm schoof promoting scam reaches 250,000 facebook views," 2025.

[8] A. News, "Video: Deepfakes posted as india–pakistan conflict." https://www.abc.net.au/news/2025-06-01/deepfakes-posted-as-india-pakistan-conflict/105360246, June 2025. Accessed: 2025-06-03.

[9] Y. Ni, D. Meng, C. Yu, C. Quan, D. Ren, and Y. Zhao, "Core: Consistent representation learning for face forgery detection," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 12–21, June 2022.

[10] Z. Yan, Y. Luo, S. Lyu, Q. Liu, and B. Wu, “Transcending forgery specificity with latent space augmentation for generalizable deepfake detection,” in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8984–8994, June 2024.

[11] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS’14, (Cambridge, MA, USA), p. 2672–2680, MIT Press, 2014.

[12] T. Zhang, “Deepfake generation and detection, a survey,” *Multimedia Tools Appl.*, vol. 81, p. 6259–6276, Feb. 2022.

[13] I. Team, “Insightface: 2d and 3d face analysis project.” `https://github.com/deepinsight/insightface`, 2023. Accessed: 2025-04-01.

[14] FaceFusion, “Facefusion.” `https://github.com/facefusion/facefusion`, 2023. Accessed: 2025-04-26.

[15] S. Zhou, K. C. Chan, C. Li, and C. C. Loy, “Towards robust blind face restoration with codebook lookup transformer,” in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS ’22, (Red Hook, NY, USA), Curran Associates Inc., 2022.

[16] A. Groshev, A. Maltseva, D. Chesakov, A. Kuznetsov, and D. Dimitrov, “Ghost—a new face swap approach for image and video domains,” *IEEE Access*, vol. 10, pp. 83452–83462, 2022.

[17] K. R. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. Jawahar, “A lip sync expert is all you need for speech to lip generation in the wild,” in *Proceedings of the 28th ACM International Conference on Multimedia*, MM ’20, (New York, NY, USA), p. 484–492, Association for Computing Machinery, 2020.

[18] S. labs inc., “Sync.” `https://sync.so/`, 2024. Accessed: 2025-03-05.

[19] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner, “Face2face: real-time face capture and reenactment of rgb videos,” *Commun. ACM*, vol. 62, p. 96–104, Dec. 2018.

[20] W. Wu, Y. Zhang, C. Li, C. Qian, and C. C. Loy, “Reenactgan: Learning to reenact faces via boundary transfer,” *CoRR*, vol. abs/1807.11079, 2018.

[21] S. Bounareli, C. Tzelepis, V. Argyriou, I. Patras, and G. Tzimiropoulos, "Hyperreenact: One-shot reenactment via jointly learning to refine and retarget faces," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7115–7125, Oct 2023.

[22] S. Bounareli, C. Tzelepis, V. Argyriou, I. Patras, and G. Tzimiropoulos, "One-shot neural face reenactment via finding directions in gan's latent space," *International Journal of Computer Vision*, vol. 132, pp. 3324–3354, August 2024.

[23] K. Liu, I. Perov, D. Gao, N. Chervoniy, W. Zhou, and W. Zhang, "Deepfacelab: Integrated, flexible and extensible face-swapping framework," *Pattern Recogn.*, vol. 141, Sept. 2023.

[24] deepfakes, "faceswap: Deepfake face swapping software." `https://github.com/deepfakes/faceswap`, 2025. Accessed: 2025-06-03.

[25] R. Chen, X. Chen, B. Ni, and Y. Ge, "Simswap: An efficient framework for high fidelity face swapping," *CoRR*, vol. abs/2106.06340, 2021.

[26] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation," *arXiv e-prints*, p. arXiv:1711.09020, Nov. 2017.

[27] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *CoRR*, vol. abs/1812.04948, 2018.

[28] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," *CoRR*, vol. abs/1912.04958, 2019.

[29] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, "Alias-free generative adversarial networks," *CoRR*, vol. abs/2106.12423, 2021.

[30] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," *CoRR*, vol. abs/2112.10752, 2021.

[31] Imagen-Team-Google, :, J. Baldridge, J. Bauer, M. Bhutani, N. Brichtova, and e. a. Andrew Bunner, "Imagen 3," 2024.

[32] DeepMind, "Veo: a text-to-video generation system," technical report, Google DeepMind, 2025. Accessed: 2025-06-03.

[33] M. Grum, “Learning representations by crystallized back-propagating errors,” in *Artificial Intelligence and Soft Computing* (L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, eds.), (Cham), pp. 78–100, Springer Nature Switzerland, 2023.

[34] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the 25th International Conference on Machine Learning*, ICML ’08, (New York, NY, USA), p. 1096–1103, Association for Computing Machinery, 2008.

[35] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” *arXiv e-prints*, p. arXiv:1312.6114, Dec. 2013.

[36] A. Makhzani, J. Shlens, N. Jaitly, and I. J. Goodfellow, “Adversarial autoencoders,” *CoRR*, vol. abs/1511.05644, 2015.

[37] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, “Masked autoencoders are scalable vision learners,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15979–15988, June 2022.

[38] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *CoRR*, vol. abs/1511.06434, 2015.

[39] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, Oct 2017.

[40] Y. Nirkin, Y. Keller, and T. Hassner, “ FSGAN: Subject Agnostic Face Swapping and Reenactment ,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, (Los Alamitos, CA, USA), pp. 7183–7192, IEEE Computer Society, Nov. 2019.

[41] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli, “Deep unsupervised learning using nonequilibrium thermodynamics,” in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML’15, p. 2256–2265, JMLR.org, 2015.

[42] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” *arXiv preprint*, vol. arXiv:2010.02502, 2020. Available: `https://arxiv.org/abs/2010.02502`.

[43] A. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," *CoRR*, vol. abs/2102.09672, 2021.

[44] P. Dhariwal and A. Q. Nichol, "Diffusion models beat gans on image synthesis," *arXiv preprint*, vol. arXiv:2105.05233, 2021. Available: https://arxiv.org/abs/2105.05233.

[45] J. Ho and T. Salimans, "Classifier-free diffusion guidance," 2022.

[46] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022.

[47] K. Preechakul, N. Chatthee, S. Wizadwongsa, and S. Suwajanakorn, "Diffusion autoencoders: Toward a meaningful and decodable representation," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10609–10619, June 2022.

[48] H. T. Sencar and N. Memon, *Digital Image Forensics: There is More to a Picture than Meets the Eye.* Springer Publishing Company, Incorporated, 2012.

[49] J. Fridrich, "Digital image forensics," *IEEE Signal Processing Magazine*, vol. 26, pp. 26–37, March 2009.

[50] B. Mahdian and S. Saic, "Blind authentication using periodic properties of interpolation," *IEEE Transactions on Information Forensics and Security*, vol. 3, pp. 529–538, Sep. 2008.

[51] M. Gardella, P. Musé, J.-M. Morel, and M. Colom, "Noisesniffer: a fully automatic image forgery detector based on noise analysis," in *2021 IEEE International Workshop on Biometrics and Forensics (IWBF)*, pp. 1–6, May 2021.

[52] X. Wang, B. Xuan, and S.-l. Peng, "Digital image forgery detection based on the consistency of defocus blur," in *2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 192–195, Aug 2008.

[53] H.-Y. Wu, M. Rubinstein, E. Shih, J. Guttag, F. Durand, and W. Freeman, "Eulerian video magnification for revealing subtle changes in the world," *ACM Trans. Graph.*, vol. 31, July 2012.

[54] H. Farid, “Exposing digital forgeries by detecting inconsistencies in lighting,” in *Proceedings of the International Workshop on Digital Watermarking (IWDW)*, pp. 1–12, 2007.

[55] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, Nov 1998.

[56] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems* (F. Pereira, C. Burges, L. Bottou, and K. Weinberger, eds.), vol. 25, Curran Associates, Inc., 2012.

[57] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint*, vol. arXiv:1409.1556, 2015. Available: https://arxiv.org/abs/1409.1556.

[58] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” *CoRR*, vol. abs/1610.02357, 2016.

[59] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015. Available: https://arxiv.org/abs/1409.4842.

[60] Y. Liu, Q. Guan, X. Zhao, and Y. Cao, “Image forgery localization based on multi-scale convolutional neural networks,” in *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security*, IHMMSec ’18, (New York, NY, USA), p. 85–90, Association for Computing Machinery, 2018.

[61] M. Delmas and R. Seguier, “Latentforensics: Towards frugal deepfake detection in the stylegan latent space,” *arXiv preprint*, vol. arXiv:2303.17222, 2023. Available: https://arxiv.org/abs/2303.17222.

[62] Z. Yan, Y. Luo, S. Lyu, Q. Liu, and B. Wu, “Transcending forgery specificity with latent space augmentation for generalizable deepfake detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31638–31648, 2024. Available: https://arxiv.org/abs/2311.11278.

[63] K. Shiohara and T. Yamasaki, “Detecting deepfakes with self-blended images,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18699–18708, June 2022.

[64] D. Güera and E. J. Delp, “Deepfake video detection using recurrent neural networks,” in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6, Nov 2018.

[65] P. Saikia, D. Dholaria, P. Yadav, V. Patel, and M. Roy, “A hybrid cnn-lstm model for video deepfake detection by leveraging optical flow features,” in *2022 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, July 2022.

[66] A. Almestekawy, H. H. Zayed, and A. Taha, “Deepfake detection: Enhancing performance with spatiotemporal texture and deep learning feature fusion,” *Egyptian Informatics Journal*, vol. 27, p. 100535, 2024.

[67] L. Y. Gong, X. J. Li, and P. H. J. Chong, “Swin-fake: A consistency learning transformer-based deepfake video detector,” *Electronics*, vol. 13, no. 15, 2024.

[68] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, “Recurrent convolutional strategies for face manipulation detection in videos,” 2019.

[69] A. Haliassos, K. Vougioukas, S. Petridis, and M. Pantic, “Lips don’t lie: A generalisable and robust approach to face forgery detection,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5037–5047, June 2021.

[70] Y. Xu, J. Liang, G. Jia, Z. Yang, Y. Zhang, and R. He, “Tall: Thumbnail layout for deepfake video detection,” 2024.

[71] A. Dosovitskiy, 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 16×16 Words: Transformers for Image Recognition at Scale,” *arXiv preprint*, vol. arXiv:2010.11929, 2020. Available: `https://arxiv.org/abs/2010.11929`.

[72] G. Bertasius, H. Wang, and A. Torralba, “Is Space–Time Attention All You Need for Video Understanding?,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 0–10, 2021. Available: `https://arxiv.org/abs/2102.05095`.

[73] C. Feichtenhofer, H. Fan, J. Malik, and K. He, “Slowfast networks for video recognition,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6201–6210, Oct 2019.

[74] J. Carreira and A. Zisserman, “Quo vadis, action recognition? a new model and the kinetics dataset,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4733, July 2017.

[75] R. Durall, M. Keuper, and J. Keuper, “Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7887–7896, June 2020.

[76] J. Fridrich and J. Kodovsky, “Rich models for steganalysis of digital images,” *IEEE Transactions on Information Forensics and Security*, vol. 7, pp. 868–882, June 2012.

[77] C. Yang, H. Li, F. Lin, B. Jiang, and H. Zhao, “Constrained r-cnn: A general image manipulation detection model,” in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, July 2020.

[78] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, “Thinking in frequency: Face forgery detection by mining frequency-aware clues,” in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 86–103, Springer International Publishing, 2020.

[79] T. Jung, S. Kim, and K. Kim, “Deepvision: Deepfakes detection using human eye blinking pattern,” *IEEE Access*, vol. 8, pp. 83144–83154, 2020.

[80] A. Kawabe, R. Haga, Y. Tomioka, Y. Okuyama, and J. Shin, “Fake image detection using an ensemble of cnn models specialized for individual face parts,” in *2022 IEEE 15th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC)*, pp. 72–77, Dec 2022.

[81] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms.* USA: Wiley-Interscience, 2004.

[82] S. Rao, N. A. Shelke, A. Goel, and H. Bansal, “Deepfake creation and detection using ensemble deep learning models,” in *Proceedings of the 2022 Fourteenth International Conference on Contemporary Computing*, IC3-2022, (New York, NY, USA), p. 313–319, Association for Computing Machinery, 2022.

[83] S. P P, R. R R, D. A, G. R, A. R, and G. B. P, “Enhancing deepfake detection: An ensemble deep learning approach for efficient attribute manipulation identification,” in *2024 International Conference on Cognitive Robotics and Intelligent Systems (ICC - ROBINS)*, pp. 352–359, April 2024.

[84] N. Giatsoglou, S. Papadopoulos, and I. Kompatsiaris, “Investigation of ensemble methods for the detection of deepfake face manipulations,” 2023.

[85] G. Naskar, S. Mohiuddin, S. Malakar, E. Cuevas, and R. Sarkar, “Deepfake detection using deep feature stacking and meta-learning,” *Heliyon*, vol. 10, no. 4, p. e25933, 2024.

[86] Z. Yan, Y. Zhang, Y. Fan, and B. Wu, “Ucf: Uncovering common features for generalizable deepfake detection,” *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22355–22366, 2023.

[87] D. E. King, “DLIB – a c++ library for machine learning and computer vision,” *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[88] W. Zhuang, Q. Chu, Z. Tan, Q. Liu, H. Yuan, C. Miao, Z. Luo, and N. Yu, “Uia-vit: Unsupervised inconsistency-aware method based on vision transformer for face forgery detection,” in *Computer Vision – ECCV 2022* (S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, eds.), (Cham), pp. 391–407, Springer Nature Switzerland, 2022.

[89] Z. Gu, Y. Chen, T. Yao, S. Ding, J. Li, F. Huang, and L. Ma, “Spatiotemporal inconsistency learning for deepfake video detection,” in *Proceedings of the 29th ACM International Conference on Multimedia*, MM ’21, (New York, NY, USA), p. 3473–3481, Association for Computing Machinery, 2021.

[90] H. Liu, X. Li, W. Zhou, Y. Chen, Y. He, H. Xue, W. Zhang, and N. Yu, “Spatial-phase shallow learning: Rethinking face forgery detection in frequency domain,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 772–781, June 2021.

[91] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, “Faceforensics++: Learning to detect manipulated facial images,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–11, Oct 2019.

[92] N. Dufour and A. Gully, “Contributing data to deepfake detection research,” 2019.

[93] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-df: A new dataset for deepfake forensics," *CoRR*, vol. abs/1909.12962, 2019.

[94] Z. Yan, Y. Zhang, X. Yuan, S. Lyu, and B. Wu, "Deepfakebench: A comprehensive benchmark of deepfake detection," 2023.

[95] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, pp. 8024–8035, 2019.

[96] NVIDIA Corporation, "CUDA toolkit documentation." `https://developer.nvidia.com/cuda-toolkit`, 2024. Accessed: 2025-06-05.

[97] Python Software Foundation, "Python language reference, version 3.10." `https://www.python.org`, 2024. Accessed: 2025-06-05.

[98] Microsoft Corporation, "Visual studio code." `https://code.visualstudio.com`, 2024. Accessed: 2025-03-05.

[99] S. Inc, "Streamlit." `https://streamlit.io`, 2024. Accessed: 2025-04-18.

[100] S. Inc, "Supabase." `https://supabase.com`, 2023. Accessed: 2024-05-01.

[101] Runpod, "Runpod." `https://runpod.io`, 2023. Accessed: 2024-05-25.

[102] Y. Lu and T. Ebrahimi, "Assessment framework for deepfake detection in real-world situations," 2023.

[103] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, and N. Yu, "Multi-attentional deepfake detection," *CoRR*, vol. abs/2103.02406, 2021.

[104] J. Wu, Y. Zhu, X. Jiang, Y. Liu, and J. Lin, "Local attention and long-distance interaction of rppg for deepfake detection," *The Visual Computer*, vol. 40, no. 2, pp. 1083–1094, 2024. Published 2024/02/01.

[105] B. Dolhansky, C. Howes, B. Pflaum, A. Baram, C. Canton Ferrer, and C. Ferrer, "The deepfake detection challenge (dfdc) dataset," *arXiv preprint arXiv:2006.07397*, 2020. URL: `https://arxiv.org/abs/2006.07397`.