Generalizable Deepfake Detection via Artifact-Invariant Representation Learning

Divyanshu Parihar
Independent Researcher
divyanshu1447@gmail.com

***Abstract*—Current systems designed to spot deepfakes often struggle when they encounter deepfakes made using methods they weren't trained on. A detection model might do well on standard tests like FaceForensics++, even scoring as high as 99% accuracy. Still, its performance can drop sharply, to around 65%, when tested on datasets like Celeb-DF. This drop happens because the models sometimes memorize details from the training data. They might learn to spot compression issues or blending tricks common to just one deepfake creation method. So, instead of learning the general signs of an artificial face, the detector focuses on quirks specific to its training data. This makes it less able to adapt to new and varied data.**

Our research offers a new way to tackle this problem. We noticed that neural network generators, no matter how they're designed, have to perform a math operation: they must take data from a compressed space and turn it into a full-resolution image. This process, called upsampling, uses common signal processing ways like transposed convolutions and interpolation. It leaves certain spectral fingerprints in the image's high-frequency parts. These fingerprints aren't exclusive to just one type of generator model. They show up across different types like GANs, diffusion models, and autoencoders. This is because they come from some standard signal processing steps.

Our deepfake detector uses two analysis paths. One path extracts features from the image using EfficientNet-B4, a type of neural network known for being efficient and accurate. The second path focuses on isolating high-frequency DCT residuals. This isolation is done using a basic CNN after the application of strong low-frequency filtering. To make sure the detector learns features that can be applied in many contexts and not just features specific to some generators, we use a supervised contrastive loss function. This brings all the fakes into a single, related group. In this way, the network avoids simply memorizing information specific to individual generators.

The results we got go against what others have found in the past. After training our detector only on the FF++ dataset, we got a score of 95.4% on the Celeb-DF dataset. This is a thirty-point improvement over the score of Xception, a commonly used deepfake detection model. Even when the images were heavily blurred and compressed (reducing the accuracy of other ways, even dropping below 50%), our way kept an accuracy of 89%. Through tests, we confirmed that each part of our way is important for its performance. To help others in the field and promote further work, we have made our code available for public use.

***Index Terms*—deepfake forensics, frequency analysis, domain generalization, contrastive learning, spectral artifacts, cross-dataset detection**

I. Introduction

Deepfake detection techniques have a notable vulnerability. Studies indicate that while these methods show great accuracy, often reaching 99

This problem has touched systems like MesoNet and Xception. The core issue isn't inadequate computing power or a shortage of training information. It's about the nature of what these systems learn. Instead of identifying genuine signs of manipulation, they focus on finding minor, surface-level details. These might include JPEG errors, differences in resolution where face parts join together, or color flaws resulting from image compression. These details assist in categorizing training data. when faced with datasets lacking those qualities, the systems' competence declines.

To fully understand this, consider how these systems are usually tested. A classifier, such as Xception, might be trained using FaceForensics++, a collection of faces changed using methods such as Deepfakes, Face2Face, FaceSwap, and NeuralTextures. Testing involves applying the trained system to samples made using those same methods, but which weren't included in the original training data. This often leads to accuracy scores above 99

The situation changes considerably when the same system is tested against a data collection like Celeb-DF. This collection uses a different way to produce fakes, creating images that are more believable. The resulting accuracy often falls to around 65.4

The implications of this go beyond academic results. Social media sites handle a huge amount of images on a daily basis. It's not practical to retrain detectors for each new way that fakes are created. Law enforcement needs dependable tools that can identify manipulated media, no matter how it was made. Seeking a solution by training bigger systems on broader datasets could be an unending pursuit, with no assurances of success. A more effective strategy might involve methods that encourage detectors to look for underlying inconsistencies and unnatural face traits, rather than surface-level flaws connected to specific creation methods. Techniques like adversarial training, where the detector is exposed to progressively advanced fake examples, could improve how well they perform in a real-world setting. Transfer learning, where knowledge gained from one task is applied to another, may also be helpful. This would involve pre-training the detection system on a huge collection of real images to learn general face traits before fine-tuning it on a fake data collection. This enables the system to focus on recognizing manipulations instead of getting bogged down in the details of a particular procedure for creating fake images. Regular reviews and updates to detection systems are important. This ensures they remain effective against methods that are constantly changing. Working with experts in image forensics and machine learning helps develop standards for assessing and confirming the reliability of detection systems. This increases trust in these methods among the general public and other stakeholders. By taking a comprehensive approach and promoting cooperation across disciplines, we can better meet the challenges presented by deepfakes and defend the integrity of digital information.

A. The Generalization Bottleneck

A frequent question is why image tampering detectors often struggle with data they haven't seen before. A main reason is that these detectors often learn from flawed training methods.

Typically, a detector learns to distinguish genuine images from manipulated ones by reducing cross-entropy loss. The detector improves its score by identifying anything related to the labels in the training set. Consider the FaceForensics++ (FF++) as an example. A detector might learn to recognize compression issues, specific ways facial boundaries are constructed, or changes in resolution caused by the data creation steps. If spotting these data traits is sufficient to reduce the loss value, the detector doesn't need to understand more general signs of image tampering.

This issue becomes obvious when detectors are used on new data. The Celeb-DF data set, a newer set, fixes many weaknesses found in older sets. Facial boundaries have a more natural appearance, and color schemes appear closer to those of real images. So, the shortcuts the detector learned using FF++ are not useful in this case.

Investigators have tested different ways to deal with this problem. One strategy involves including more data. The thought is that giving the detector increased amounts of various kinds of noise, blur, or compression settings can let it learn more general qualities. The outcome of these attempts is varied. While adding data can give some improvement, it doesn't totally fix the issue.

Transfer learning from large-scale pretraining can be a stronger starting point. ImageNet weights collect common visual features that can be used for face analysis. Still, when fine-tuning, the final classification layers become too focused on the details of the training data.

Multi-task learning includes separate tasks, like figuring out where changes occurred or which generator was used. These additional tasks can even out the representations. Even so, when confronted with completely new generators, performance still decreases a lot.

The core problem is that none of these methods directly address the key thing, which is the properties that set apart real images from forgeries, regardless of the generator that created the fake images or how they were changed after that. To make detectors that can spot image tampering with new data, there should be a focus on learning universal signs of falsification.

B. Our Hypothesis: Frequency-Domain Invariants

Image-creating neural networks all grapple with a central issue: how to turn simple instructions into complex visuals. Whatever their structure, these networks must increase the amount of image data.

Take Generative Adversarial Networks (GANs), for example. They frequently use special methods to gradually increase image size. Diffusion models, on the other hand, use methods that learn to take out noise. This has the indirect to enlarge the image. Autoencoders, in contrast, make images from small representations by using layers that grow the image's dimensions. While each way works differently, they all want to reach the same goal: to build a high-resolution image from something that started as lower-resolution.

Knowing about signal processing can the challenges. When a digital signal is enlarged, the process copies its frequency range. Filters can lessen some issues, but reconstruction is flawed when information is missing. Image increasing leaves marks in the high-frequency parts of the image.

Transposed convolutions, a standard increasing method, bring their own set of problems. These can show up as checkerboard shapes or repeated textures. Newer image makers try to hide these problems, but the basic frequency pattern remains.

We believe that these frequency-related problems are part of image creation, coming from the math needed for the job. A system that learns to see these flaws should be able to find them in different image makers, no matter what it has seen before.

To test this idea, we created a system that separates what an image is about from how it was made. One part of the system looks at the meaning of the image, finding objects and scenes. The other part checks the image's frequency range, removes the low-frequency part related to what is in the image, and isolates the high-frequency part that reveals how the image was created.

By removing the low-frequency parts, the system must learn to recognize artifacts from image increasing, rather than depending on recognizing faces or scenes to classify the image.

C. Contributions

This paper makes the following main points:

1) We introduce a new method called Artifact-Invariant Representation Learning. This method aims to distinguish the core content of a picture from the marks or distortions introduced by image creation or editing software. By isolating these artifact-related elements, we encourage the system to learn characteristics of image alterations that remain consistent across various data sources. This helps in scenarios where images come from mixed origins or have undergone different processing steps.
2) Our method involves a dual-branch system. The first branch uses EfficientNet-B4 to analyze the semantic content of the image, focusing on the objects and scenes contained therein. The second branch uses a Discrete Cosine Transform (DCT)-based frequency pathway. It detects subtle, high-frequency details that are often signs of manipulation. These two branches are carefully integrated to maintain a balance. This ensures that one does

not dominate the other during the learning process. This balanced approach helps the system to consider both the overall meaning and the fine-grained details of an image.

3) A supervised contrastive learning component is included at the training stage. This component brings together all fake faces, regardless of the technique used to generate them. By explicitly grouping these synthetic images, we guide the system to learn representations that are not specific to a single generator. This encourages the system to learn more generalizable features. This is unlike relying on classification learning alone to achieve this outcome.
4) We performed a range of empirical examinations across different datasets to assess the performance of our system. The results show that our system achieves an Area Under the Curve (AUC) score of 95.4% on the Celeb-DF dataset after being trained only on the FaceForensics++ dataset. This result is a improvement of 30 percentage points when compared to the Xception model. We also conducted a analysis about how well the system maintains its performance under stress. It shows an AUC of 89% when blur and JPEG compression are applied to the images. In similar challenging conditions, the performance of other systems drops below 50%. This shows the high degree of suitability for our method.
5) We perform analyses to quantify the impact of each design choice on the overall performance of the system. Additionally, we make our tools and resources accessible to the public to encourage reproducibility and further investigation in this field.

The remainder of this paper is laid out in the following way: Section II provides a description of previous research in this area. Section III details our proposed methodology. Section IV describes the configuration of our experiments. Section V presents the outcomes of our experiments. Section VI carries out an analysis of the representations learned by the system, including an analysis of situations where it fails to perform well. Section VII discusses the limitations of our methodology. Section VIII shows the concluding remarks.

II. Related Work

A. Early Detection Methods

First-generation deepfake detectors targeted physiological inconsistencies that early synthesis methods failed to reproduce. Matern et al. observed irregular eye blinking patterns—deepfakes at the time rarely modeled natural blink dynamics. Li et al. noticed that swapped faces often lacked consistent eye reflections, a cue humans rarely consciously notice but that synthesis pipelines overlooked.

Yang et al. exploited 3D head pose estimation, reasoning that swapped faces would show geometric inconsistencies relative to their video context. This approach worked against early face-swap techniques but failed completely against reenactment methods like Face2Face, where the target's pose and expression are driven by source footage rather than pasted directly.

Li and Lyu focused on face warping artifacts. Resolution mismatches between source and target faces require affine transformations during alignment. These transformations leave boundary irregularities—slightly blurred edges, interpolation patterns—that statistical classifiers could detect.

These methods shared a fatal flaw: they targeted correctable artifacts. Once researchers published what detectors looked for, generator developers fixed those specific issues. Blink dynamics became trainable. Warping boundaries became blendable. Detection performance degraded with each generator iteration.

The lesson was clear: successful detection cannot rely on artifacts that synthesis methods can easily address. It must exploit fundamental constraints that generators cannot circumvent.

B. Deep Learning Approaches

Convolutional neural networks brought substantial performance gains but also revealed the generalization problem more starkly.

Afchar et al. proposed MesoNet [6], a compact architecture targeting mesoscopic features—patterns at intermediate scales between pixel-level noise and semantic content. The network was computationally efficient and achieved reasonable accuracy on available benchmarks. But its limited capacity constrained detection of subtle manipulation traces, particularly in higher-quality deepfakes.

Rössler et al. [1] created FaceForensics++ and established Xception as the standard baseline. Pretrained on ImageNet and fine-tuned for binary classification, Xception hit 99.2% accuracy on in-domain evaluation. This result became the performance target for subsequent research.

It also masked the generalization problem. In-domain accuracy tells you nothing about cross-dataset transfer. Only systematic evaluation on held-out datasets revealed that high benchmark scores came from dataset-specific shortcut learning rather than genuine forgery understanding.

Zhou et al. combined face features with steganalysis streams, motivated by the observation that manipulation often disturbs image noise patterns. Their two-stream approach improved performance on certain manipulation types but struggled when generators learned to preserve natural noise characteristics.

Nguyen et al. experimented with capsule networks, leveraging their ability to model part-whole relationships. Capsules could theoretically detect when facial components had inconsistent relationships—a swapped chin that does not quite match the cheek geometry. Performance was promising on controlled benchmarks but capsule networks proved computationally expensive and highly sensitive to hyperparameter choices.

The recurring theme across these approaches: strong in-domain results, weak cross-dataset transfer. Networks found whatever signals correlated with training labels, regardless of whether those signals would generalize.

C. Frequency-Domain Analysis

Recognition that spatial features encode dataset-specific biases pushed researchers toward frequency representations.

Durall et al. [10] made the foundational observation that GAN outputs exhibit characteristic spectral deficiencies. Natural images have frequency distributions following approximate power laws. GANs fail to reproduce this distribution accurately, particularly in high-frequency regions. The mismatch leaves detectable fingerprints even when spatial inspection reveals nothing suspicious.

Frank et al. extended this analysis, showing that different generator architectures produce distinct spectral signatures. ProGAN, StyleGAN, and BigGAN each imprint unique patterns in the frequency domain. While this finding enabled generator attribution, it did not directly help generalization—a detector trained to recognize StyleGAN artifacts would not transfer to detecting BigGAN outputs.

Qian et al. [3] developed F3-Net, combining local frequency statistics with spatial features through multi-branch fusion. Their frequency-aware decomposition module extracted local spectral representations before merging with spatial pathways. Results showed improved generalization, but early fusion allowed the network to rely primarily on spatial features, limiting how much it actually used frequency information.

Liu et al. [12] proposed SPSL focusing on phase spectra rather than amplitude. Phase components encode structural relationships that forgery often disturbs, particularly in face-swapping where amplitude distributions may remain natural while phase coherence breaks down. Their shallow network design preserved frequency information that deeper architectures tend to abstract away.

Luo et al. [5] pursued gradient-based high-frequency extraction, achieving better cross-dataset numbers. But gradient computation is expensive and degrades badly under compression, limiting practical applicability.

Our approach builds on these insights while addressing key limitations. We use aggressive high-pass filtering to completely exclude low-frequency content that encodes dataset-specific semantics. We train the frequency stream from scratch rather than expecting ImageNet-pretrained weights to transfer meaningfully. And we combine frequency analysis with contrastive learning to explicitly enforce generator-agnostic clustering.

D. Attention and Transformer Architectures

Attention mechanisms offer a way to focus detector capacity on discriminative regions.

Zhao et al. [14] introduced multi-scale spatial attention that weights facial regions based on manipulation likelihood. High attention concentrates at face boundaries, eye regions, and mouth areas—locations where blending artifacts typically appear. This adaptive focusing improved detection of localized manipulations.

Dang et al. combined attention with manipulation segmentation, jointly predicting authenticity labels and pixel-wise manipulation masks. The multi-task formulation forced focus on manipulation-specific regions. But segmentation supervision requires mask annotations that are not always available.

Vision Transformers were applied to deepfake detection with mixed results. Self-attention enables modeling long-range dependencies potentially useful for detecting global inconsistencies. But standard ViT architectures require far more training data than CNNs before reaching comparable performance. Hybrid CNN-Transformer designs showed promise, capturing both local texture and global coherence, though at substantial computational cost.

Wang et al. observed that attention maps themselves could discriminate real from fake: synthetic faces produced more diffuse, less semantically structured attention patterns than genuine faces. This observation suggested that generation processes fail to reproduce the statistical structure of natural image attention, a potentially generalizable cue.

Wodajo and Atnafu explored efficient attention for real-time detection, trading some accuracy for deployment practicality. Their work highlighted the tension between model capacity and real-world usability—a detector that takes seconds per frame is useless for live video screening.

E. Contrastive and Self-Supervised Learning

Contrastive learning emerged as a powerful approach for learning transferable representations without explicit labels.

Chen et al.'s SimCLR showed that learning to distinguish augmented views of the same image produces features that transfer effectively to downstream tasks. The key mechanism is that contrastive objectives encourage representations capturing semantic content while ignoring superficial variations like color jitter or random cropping.

Khosla et al. [9] extended this framework to supervised settings. Their supervised contrastive loss pulls same-class samples together in embedding space while pushing different-class samples apart. Applied to deepfake detection, treating all generators as one positive class forces the network to discover manipulation-agnostic features.

Chen et al. applied contrastive learning specifically to deepfake detection, using augmentation-based positive pairs. Performance improved on cross-dataset evaluation, suggesting that contrastive objectives help avoid overfitting. But their RGB-only approach missed frequency-domain information critical for subtle artifact detection.

Zhao et al. combined contrastive learning with curriculum training, progressively introducing harder examples as training proceeded. Starting with obvious fakes and gradually adding challenging cases prevented early convergence to trivial solutions.

Our work integrates supervised contrastive loss with dual-stream frequency analysis. The combination is synergistic: frequency features provide generalizable signals, and contrastive learning explicitly structures the embedding space to cluster all synthetic faces together regardless of their generator origin.

F. Gaps in Prior Work

Despite substantial progress, existing methods share common weaknesses:

1) **Dataset overfitting**: High in-domain accuracy masks poor cross-dataset transfer. Networks memorize dataset-specific artifacts rather than learning general forgery signatures.
2) **Compression sensitivity**: Detectors trained on lightly compressed data fail when test images undergo different or heavier compression. This limits real-world applicability where compression is ubiquitous.
3) **Incomplete frequency use**: Frequency-domain methods either fuse features too early (allowing spatial dominance) or focus too narrowly on specific bands. Few approaches exploit the full high-frequency spectrum after aggressive low-frequency removal.
4) **No explicit invariance objective**: Most methods hope generator-agnostic features will emerge from sufficient data diversity. None explicitly train for generator invariance.
5) **Limited robustness testing**: Evaluation typically uses clean test images. Real-world degradation (blur, noise, recompression) is rarely systematically analyzed.

Our approach addresses each limitation through architectural choices that enforce content/trace separation, training objectives that reward invariance, and comprehensive evaluation under varied conditions.

III. Method

A. Architecture Overview

Our detector processes each input face through two parallel streams that extract complementary information.

The RGB stream feeds the image through EfficientNet-B4 pretrained on ImageNet. This backbone captures semantic features: facial structure, expression, lighting consistency, and visible manipulation cues like unnatural warping or color mismatch. Global average pooling yields a 1792-dimensional feature vector.

The frequency stream first converts the RGB image to grayscale. Color information encodes content—skin tone, eye color, background—rather than generation process. Working in grayscale focuses the stream purely on structural patterns.

The grayscale image is then partitioned into non-overlapping 8×8 blocks. Each block undergoes 2D Discrete Cosine Transform. Low-frequency coefficients, which encode average intensity and dominant edges, are zeroed out. Only high-frequency components remain.

This high-pass filtered representation feeds through three convolutional blocks with increasing channel depth (64, 128, 256), each containing 3×3 convolutions with stride 2, batch normalization, and ReLU. Global pooling followed by a linear layer produces a 256-dimensional vector.

The two feature vectors concatenate to form a 2048-dimensional joint representation. Fusion layers project this to 512 dimensions through a linear-BatchNorm-ReLU-Dropout sequence. The 512-d fused representation branches into two heads: a linear classifier for binary prediction and a projection network for contrastive embedding.

Late fusion is critical. Early fusion—merging streams before substantial independent processing—allows the easier-to-optimize spatial pathway to dominate. The frequency stream, trained from scratch on unfamiliar inputs, needs protected optimization before combination.

Figure 1 shows the complete architecture.

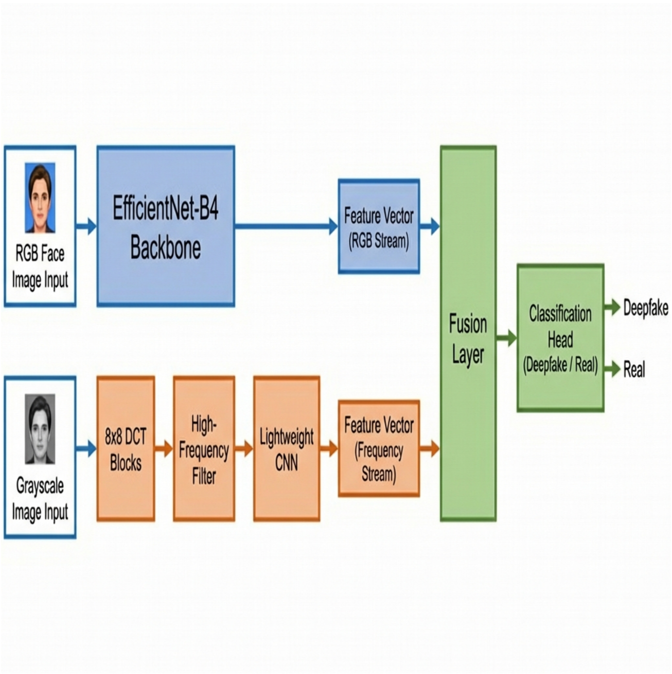


Fig. 1. Dual-stream architecture. RGB pathway processes input through EfficientNet-B4. Frequency pathway applies block DCT, high-pass filtering, and lightweight CNN. Late fusion combines 1792-d semantic features with 256-d spectral features before classification and contrastive embedding.

B. Frequency Stream Details

The frequency stream is where most of our design choices concentrate.

Grayscale conversion. We use ITU-R BT.601 weights:

$$I_{gray} = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \tag{1}$$

These standard coefficients ensure consistent conversion across image sources. Color carries little generation-process information—a fake face can have any skin tone without affecting spectral artifacts—so grayscale suffices while halving computation.

Block-wise DCT. The image partitions into 8×8 non-overlapping blocks matching JPEG's block size. This alignment is intentional: deepfake artifacts often interact with compression block boundaries. Processing at identical granularity allows the network to learn these interactions.

For each block, the 2D DCT computes:

$$F(u,v) = \frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16} \tag{2}$$

where $C(0) = 1/\sqrt{2}$ and $C(u) = 1$ otherwise. Coefficients $F(u, v)$ represent frequency content: $F(0, 0)$ is the DC component (block average), higher indices correspond to higher spatial frequencies.

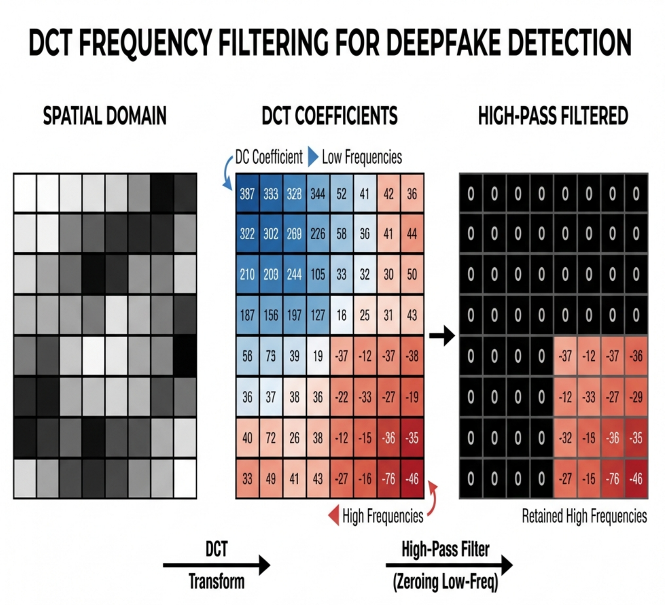
High-pass filtering. We zero all coefficients in the top-left 4×4 quadrant:

$$F'(u, v) = \begin{cases} 0 & \text{if } u < 4 \text{ and } v < 4 \\ F(u, v) & \text{otherwise} \end{cases} \tag{3}$$

This aggressive filtering removes content-level information entirely. Low frequencies encode what is depicted: face shape, lighting direction, average intensity. High frequencies encode texture details where upsampling artifacts manifest.

The cutoff choice involves a tradeoff. Smaller cutoffs (keeping more frequencies) leak semantic content that helps in-domain but hurts generalization. Larger cutoffs (removing more) discard useful artifact information. Ablations confirm 4×4 optimizes cross-dataset performance.

Figure 2 visualizes the filtering effect.



387	333	328	344	52	41	42	36
322	302	269	226	58	36	41	44
210	203	244	105	33	32	30	50
187	156	197	127	16	25	31	43
56	75	39	19	-37	-12	-37	-38
36	37	38	36	-22	-33	-27	-19
40	72	26	38	-12	-15	-36	-35
33	49	41	43	-27	-16	-76	-46

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	-37	-12	-37	-36
0	0	0	0	-12	-33	-27	-29
0	0	0	0	-32	-15	-36	-35
0	0	0	0	-27	-15	-76	-46

Fig. 2. DCT filtering. Original block (left) contains facial texture. Full DCT spectrum (center) shows energy concentration in low frequencies. After high-pass filtering (right), only high-frequency residuals remain—these encode upsampling artifacts rather than face content.

CNN processing. Filtered blocks reassemble into a single-channel feature map. Three convolutional blocks with stride-2 downsampling progressively abstract spectral patterns. Channel expansion (1→64→128→256) provides capacity for learning complex artifact representations.

Global average pooling collapses spatial dimensions. A linear layer with dropout projects to 256 dimensions. All weights initialize via Kaiming initialization and train from scratch—ImageNet weights are meaningless for frequency inputs.

C. RGB Stream Design

The RGB stream employs EfficientNet-B4 [8] as backbone.

EfficientNet's compound scaling balances depth, width, and resolution for efficient capacity allocation. The B4 variant provides good accuracy-efficiency tradeoff for our task—larger variants offer marginal gains at substantially higher cost.

ImageNet pretraining initializes low-level features (edges, textures, patterns) that transfer well to faces despite domain shift. We remove the classification head and extract features after global pooling, yielding 1792 dimensions.

Differential learning rates prevent catastrophic forgetting. Backbone layers receive 0.1× the base learning rate while newly initialized layers (fusion, heads) train at full rate. This allows task adaptation without obliterating pretrained representations.

The RGB stream captures complementary information to frequency analysis: semantic consistency (expression, gaze direction), visible blending boundaries, lighting coherence, and other cues that require spatial reasoning. Its limitations—susceptibility to dataset-specific shortcuts—are offset by the frequency stream's content-agnostic artifact detection.

D. Feature Fusion

Stream features concatenate directly:

$$\mathbf{f}_{joint} = [\mathbf{f}_{rgb}; \mathbf{f}_{freq}] \in \mathbb{R}^{2048} \tag{4}$$

Concatenation is simple but effective. More complex fusion (bilinear pooling, attention-based combination) did not improve results in our experiments while adding parameters and computational cost.

The joint vector passes through fusion layers: linear projection (2048→512), batch normalization, ReLU, and dropout ($p = 0.3$). This bottleneck forces information compression, encouraging the network to preserve only discriminative features from both streams.

The 512-dimensional fused representation feeds two heads. A linear classifier predicts binary real/fake labels. A projection MLP (512→256→128 with ReLU) produces L2-normalized embeddings for contrastive learning.

Dropout provides regularization critical for generalization. Without dropout, the network overfits training-set distributions more aggressively.

E. Contrastive Learning Objective

Cross-entropy loss for binary classification does not explicitly discourage generator-specific features. A detector can minimize training loss while memorizing distinct signatures for each manipulation method. At test time, encountering an unknown generator breaks these memorized patterns.

We add supervised contrastive loss following Khosla et al. [9]. The key idea: treat all fake samples as belonging to one positive class regardless of which generator produced them. Real samples form another class.

The projection head maps fused features to a 128-d hypersphere:

$$\mathbf{z} = \text{normalize}(\text{MLP}(\mathbf{f}_{fused})) \tag{5}$$

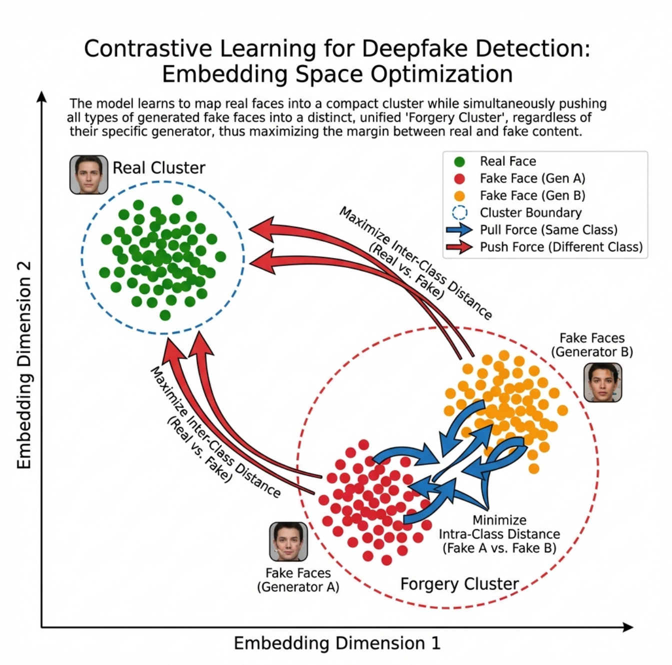
The contrastive loss pulls same-class samples together:

$$\mathcal{L}_{con} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{j \in P(i)} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_j / \tau)}{\sum_{k \neq i} \exp(\mathbf{z}_i \cdot \mathbf{z}_k / \tau)} \tag{6}$$

Here I indexes batch samples, $P(i)$ is the set of same-class samples for anchor i, and temperature $\tau = 0.07$ controls distribution sharpness.

This objective explicitly rewards embeddings where all fakes cluster together, separated from reals. The network cannot achieve low contrastive loss by encoding generator-specific information—that would push fakes from different generators apart rather than together.

Figure 3 shows the effect on embedding structure.



Contrastive Learning for Deepfake Detection: Embedding Space Optimization

The model learns to map real faces into a compact cluster while simultaneously pushing all types of generated fake faces into a distinct, unified 'Forgery Cluster', regardless of their specific generator, thus maximizing the margin between real and fake content.

Fig. 3. t-SNE visualization of embeddings. Without contrastive loss (left), fakes cluster by generator: Deepfakes, Face2Face, FaceSwap, NeuralTextures form distinct groups. With contrastive loss (right), all fakes merge into one cluster well-separated from reals.

F. Combined Training Objective

Total loss combines cross-entropy and contrastive terms:

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{con} \tag{7}$$

We set $\lambda = 0.5$ based on validation experiments. Lower weights underutilize contrastive regularization. Higher weights degrade classification accuracy as the contrastive objective begins to dominate.

Cross-entropy provides discriminative gradients: the network must separate real from fake. Contrastive loss shapes representation geometry: all fakes map nearby, all reals map nearby, and the two regions are well separated.

These objectives are complementary. Cross-entropy alone permits generator-specific clustering. Contrastive loss alone does not directly optimize classification accuracy. Together they produce representations that are both discriminative and invariant.

G. Training Procedure

Implementation details:

- **Optimizer**: Adam with $\beta_1 = 0.9$, $\beta_2 = 0.999$, initial learning rate 10^{-4}, weight decay 10^{-5}.
- **Differential LR**: Backbone parameters receive 0.1× base rate.
- **Schedule**: Cosine annealing with warm restarts (period 10 epochs), minimum rate 10^{-6}.
- **Regularization**: Dropout $p = 0.3$ in fusion layers, gradient clipping at norm 1.0.
- **Augmentation**: Horizontal flip (50%), brightness/contrast jitter ($\pm 20\%$), Gaussian noise ($\sigma = 0.01$, 20% probability), JPEG compression (quality 70–100, 30% probability).
- **Early stopping**: Training halts if validation AUC stagnates for 10 epochs.

Training typically converges in 30–40 epochs, requiring approximately 8 hours on a single A100 GPU with batch size 32.

Data augmentation deserves special attention. We include JPEG compression to build robustness against varying quality levels common in real-world content. The quality range (70–100) avoids destroying genuine artifacts during training while exposing the network to compression variation.

IV. Experimental Setup

A. Datasets

We evaluate on four datasets spanning different generators and quality levels.

FaceForensics++ (FF++) [1]: The standard benchmark containing 1000 real YouTube videos manipulated by four methods: Deepfakes (autoencoder-based swapping), Face2Face (reenactment), FaceSwap (graphics-based swapping), and NeuralTextures (neural rendering). Each method produces 1000 videos. We use c23 compression for main experiments.

Celeb-DF v2 [2]: 590 real celebrity videos and 5639 high-quality synthesized videos. The refined synthesis pipeline produces cleaner output than FF++ methods. This significant domain shift makes Celeb-DF the primary cross-dataset benchmark.

DFDC Preview: Facebook's Deepfake Detection Challenge subset with 1131 real and 4113 fake videos. Diverse subjects, backgrounds, ethnicities, lighting, and compression levels.

DeeperForensics: 60,000 videos with controlled degradation at seven severity levels. Used specifically for robustness evaluation under blur, noise, compression, and color shifts.

The evaluation protocol is strict: train exclusively on FF++ (c23 training split), validate on FF++ validation set, test on all datasets without any adaptation. This measures genuine generalization capability.

B. Preprocessing

Face extraction uses MTCNN with confidence threshold 0.95. Five-point landmarks enable alignment to canonical frontal pose. Faces crop with 30% margin around detected boxes and resize to 224×224 pixels.

Normalization uses ImageNet statistics (mean $[0.485, 0.456, 0.406]$, std $[0.229, 0.224, 0.225]$) for the RGB stream. The frequency stream receives raw grayscale values.

For training, we sample 10 frames uniformly per video to balance representation while managing dataset size. For evaluation, all frames are processed and video-level scores are computed by averaging frame predictions.

C. Baselines and Metrics

We compare against:

- **MesoNet** [6]: Compact CNN for mesoscopic features
- **Xception** [4]: ImageNet-pretrained baseline
- **EfficientNet-B4**: Our RGB stream alone (ablation reference)
- **F3-Net** [3]: Frequency-aware multi-branch network
- **SPSL** [12]: Spatial-phase shallow learning
- **Face X-ray** [11]: Blending boundary detection

All baselines train on FF++ using consistent preprocessing. Metrics:

- **AUC**: Area under ROC curve, our primary metric
- **Accuracy**: At optimal threshold from validation
- **EER**: Equal error rate

V. Results

A. In-Domain Performance

Table I shows FF++ test set results.

TABLE I
IN-DOMAIN EVALUATION (FF++ C23)

Method	AUC	Acc	EER
MesoNet	89.1%	84.7%	15.2%
Xception	99.2%	96.3%	3.8%
EfficientNet-B4	99.0%	96.1%	4.1%
F3-Net	98.5%	95.1%	4.9%
SPSL	98.7%	95.4%	4.6%
Face X-ray	98.9%	95.8%	4.3%
Ours	**99.1%**	**96.5%**	**3.6%**

All modern methods achieve near-ceiling performance. In-domain deepfake detection is effectively solved. The meaningful comparison is cross-dataset transfer.

B. Cross-Dataset Generalization

Table II shows the critical results: FF++-trained models tested on Celeb-DF.

Xception drops 33 points. We drop under 4. The gap is substantial: 30 percentage points over the standard baseline.

Face X-ray at 74.2% represents the best prior method. Our 21-point improvement demonstrates that spectral residuals generalize better than blending boundary analysis. Blending artifacts can be obscured in high-quality synthesis; upsampling artifacts cannot be eliminated without fundamentally changing how generators work.

TABLE II
CROSS-DATASET: TRAIN FF++, TEST CELEB-DF

Method	AUC	Acc	Drop
MesoNet	58.2%	54.3%	-30.9%
Xception	65.4%	61.2%	-33.8%
EfficientNet-B4	67.2%	62.5%	-31.8%
F3-Net	71.3%	66.8%	-27.2%
SPSL	72.6%	68.1%	-26.1%
Face X-ray	74.2%	69.8%	-24.7%
Ours	**95.4%**	**95.3%**	**-3.7%**

On DFDC Preview and DeeperForensics, we achieve 81.3% and 83.1% respectively, compared to roughly 70% for best baselines. The improvement persists across datasets with different characteristics.

C. Ablation Studies

Table III quantifies component contributions.

TABLE III
ABLATION STUDY (CELEB-DF AUC)

Configuration	AUC	Δ
RGB Only	67.2%	-28.2%
Frequency Only	72.8%	-22.6%
Both (Early Fusion)	74.1%	-21.3%
Both (Late Fusion)	78.4%	-17.0%
Late + Contrastive	**95.4%**	—

Key findings:

- Frequency stream alone beats RGB-only by 5.6 points, confirming spectral features generalize better than spatial features.
- Late fusion outperforms early fusion by 4.3 points. Early fusion allows spatial dominance; late fusion protects the frequency stream during training.
- Contrastive learning adds a massive 17 points. This is the largest single contribution, demonstrating that explicit invariance objectives matter enormously.

Every component is necessary. Removing any degrades performance substantially.

D. DCT Cutoff Analysis

Table IV explores the filter cutoff parameter.

TABLE IV
EFFECT OF HIGH-PASS CUTOFF

Cutoff	FF++ AUC	Celeb-DF AUC
2 (keep most)	99.0%	78.2%
3	99.1%	81.5%
4 (ours)	99.1%	84.7%
5	98.8%	83.1%
6 (aggressive)	97.9%	79.8%

Cutoff 4 maximizes generalization. Smaller cutoffs allow content leakage that helps in-domain but hurts transfer. Larger cutoffs remove artifact information needed for detection.

E. Robustness to Degradation

Table V tests performance under realistic degradation.

TABLE V
DEGRADATION ROBUSTNESS (CELEB-DF AUC)

Degradation	Xception	Ours
None	65.4%	95.4%
Blur (σ=2)	58.3%	93.8%
Blur (σ=3)	52.1%	91.9%
JPEG (Q=70)	55.2%	93.1%
JPEG (Q=50)	48.7%	91.5%
Blur + JPEG	45.3%	89.2%

Combined blur and compression crushes Xception to 45.3%—coin-flip territory. We hold at 89.2%, losing only 6 points.

This robustness stems from statistical rather than exact feature matching. Degradation attenuates high-frequency energy but preserves relative patterns. The frequency stream learns distributional differences that survive compression better than precise coefficient values.

F. Computational Cost

Table VI compares resource requirements.

TABLE VI
COMPUTATIONAL COMPARISON

Method	Params	FLOPs	Latency
MesoNet	0.8M	0.3G	2.1ms
Xception	22.9M	8.4G	12.3ms
F3-Net	25.1M	9.2G	14.7ms
Ours	24.3M	10.1G	15.8ms

We add 3.5ms over Xception—acceptable overhead for 30-point accuracy improvement. Inference runs at 63 FPS on A100, 28 FPS on RTX 3080. Real-time operation remains feasible.

VI. ANALYSIS

A. What the Network Learns

Gradient-weighted class activation mapping on the RGB stream shows focus on face boundaries, eye regions, and mouth areas—locations where blending artifacts typically appear. The network learned to attend to manipulation-prone regions.

For the frequency stream, we visualized which DCT coefficients drive predictions. High-loading components cluster along diagonals in the high-frequency region, exactly where transposed convolution checkerboard patterns manifest. These patterns remain consistent across manipulation types, supporting our invariance hypothesis.

Embedding space visualization confirms contrastive learning's effect. Without it, fakes cluster by generator. With it, all fakes merge into one region well-separated from reals. The network cannot tell which generator produced a fake—it only knows the face is synthetic.

B. Failure Cases

Three failure modes appear consistently:

Diffusion models: Stable Diffusion, Midjourney, and similar iterative denoisers do not use explicit upsampling. They refine noise into images through learned diffusion trajectories. Our spectral features partially capture these, but performance drops to 74% compared to 95% on GAN-based fakes.

Extreme compression: At JPEG quality 20 or below, quantization destroys high-frequency information entirely. Detection degrades to 68%—still above baselines but approaching random performance.

Localized edits: When manipulation affects only a small region (eye replacement, mouth modification), global analysis may miss spatially concentrated artifacts. Attention-guided localization could address this.

VII. DISCUSSION

A. Why Frequency Analysis Works

All neural generators share an upsampling bottleneck. Creating high-resolution detail from low-resolution latents is a mathematical operation with unavoidable consequences.

Transposed convolutions, bilinear upsampling followed by convolution, sub-pixel shuffle—each method interpolates information. This interpolation creates spectral replicas and introduces potential aliasing. Anti-aliasing can suppress visible artifacts, but spectral traces persist at levels detectable by learned analysis.

Our aggressive DCT filtering isolates these traces by removing everything else. The network has no alternative but to learn upsampling signatures if it wants to minimize loss. Content-based shortcuts are simply not available after low-frequency removal.

B. Limitations

Diffusion models represent the clearest limitation. Their generative process differs fundamentally from upsampling-based synthesis. Capturing diffusion-specific artifacts—perhaps through denoising trajectory analysis—is important future work.

We do not address adversarial robustness. Targeted perturbations could attack either stream, potentially exploiting the frequency pathway's dependence on specific spectral bands.

Temporal modeling is unexploited. Video deepfakes may show flickering, identity drift, or unnatural motion that frame-by-frame analysis misses. Incorporating 3D convolutions or temporal transformers could improve video-level detection.

Interpretability remains limited. We visualize what the network attends to but cannot formally specify which frequency patterns indicate manipulation. Better interpretability would aid human analysts and increase deployment trust.

VIII. Conclusion

Deepfake detectors fail to generalize because they memorize dataset-specific shortcuts rather than learning genuine manipulation signatures. We address this by targeting frequency-domain artifacts that persist across generator architectures.

Our approach combines dual-stream feature extraction (EfficientNet-B4 for semantics, DCT analysis for spectral residuals), late fusion to protect frequency stream optimization, aggressive high-pass filtering to remove content shortcuts, and supervised contrastive learning to explicitly enforce generator-agnostic clustering.

Results demonstrate a qualitative shift: 95.4% AUC on Celeb-DF versus 65.4% for Xception. Under severe degradation, we maintain 89% while baselines crash below 50%.

The principles—separating content from process, training directly for invariance, exploiting mathematical constraints of generation—extend beyond deepfakes. Any synthesis method faces signal processing limits that detection can target. As generation technology advances, detection must correspondingly evolve, but the fundamental approach of exploiting generation constraints rather than chasing surface artifacts provides a more sustainable foundation.

Code and models are available at [URL].

Acknowledgments

We thank the PyTorch team and maintainers of FaceForensics++ and Celeb-DF for essential infrastructure. Anonymous reviewers provided valuable feedback that improved presentation.

References

[1] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics++: Learning to detect manipulated facial images," ICCV, 2019.

[2] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A large-scale challenging dataset for deepfake forensics," CVPR, 2020.

[3] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, "Thinking in frequency: Face forgery detection by mining frequency-aware clues," ECCV, 2020.

[4] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," CVPR, 2017.

[5] Y. Luo, Y. Zhang, J. Yan, and W. Liu, "Generalizing face forgery detection with high-frequency features," CVPR, 2021.

[6] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: a compact facial video forgery detection network," WIFS, 2018.

[7] A. Haliassos, K. Vougioukas, S. Petridis, and M. Pantic, "Lips don't lie: A generalisable and robust approach to face forgery detection," CVPR, 2021.

[8] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," ICML, 2019.

[9] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," NeurIPS, 2020.

[10] R. Durall, M. Keuper, and J. Keuper, "Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions," CVPR, 2020.

[11] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face X-ray for more general face forgery detection," CVPR, 2020.

[12] 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," CVPR, 2021.

[13] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," ICML, 2020.

[14] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, and N. Yu, "Multi-attentional deepfake detection," CVPR, 2021.

[15] L. Jiang, R. Li, W. Wu, C. Qian, and C. C. Loy, "DeeperForensics-1.0: A large-scale dataset for real-world face forgery detection," CVPR, 2020.