Benchmarking Deepfake Detection Methods

Aditya Kumar
Rudraksh Kumawat
Priyanshu Joshi
Vishvas Patel

Content

1 Introduction

In recent years, deepfake technology has rapidly advanced allowing for the creation of highly convincing synthetic images and videos through face swapping, reenactment and other manipulation techniques. While this progress showcases the power of deep learning, it also raises serious concerns about misinformation, privacy violations and security threats.

1.1 Challenges

While deepfake detection has gained significant attention, fair and comprehensive evaluation of detection methods remains a major challenge.

Many deepfake detection approaches use different training data and are yet compared on a common test set. This inconsistency raises concerns about whether performance gains come from the method itself or the quality of training data.

Moreover, top-performing models often suffer from poor generalization and overfitting especially when evaluated on perturbed datasets. This leads to unreliable performance in real-world scenarios, where deepfakes are more varied and subtle.

Another limitation lies in the evaluation metrics because commonly used metrics like AUC and accuracy fail to capture practical aspects such as computational cost and robustness which are crucial for real-world deployment.

1.2 Objective

To overcome current evaluation limitations, this paper proposes a unified benchmark that ensures consistent training conditions and enables a fair, comprehensive comparison of 13 deepfake detection methods. The benchmark includes a standardized training dataset, a challenging Imperceptible and Diverse (ID) test set reflecting real-world complexity, and five evaluation metrics covering detection accuracy, generalization, robustness, efficiency, and inference time. Experiments were conducted using these datasets and results for the same are discussed later in this report.

2 Related Work

To build a reliable and meaningful benchmark for deepfake detection, it is essential to first understand the current landscape of deepfake generation techniques and the datasets used to train and evaluate detection models. Our focus remains primarily on face-swapping manipulation as it is one of the most common and studied forms of deepfakes.

2.1 Deepfake Creation

Modern deepfakes are created using a variety of techniques, most of which fall under three main categories: autoencoder-based, GAN-based and graphic-based manipulations.

a) Autoencoder-based manipulation

This is one of the earliest and most widely used methods for face-swapping. It works by encoding the source face into a latent representation and decoding it as the target face. Tools like FakeApp and FaceSwap rely on this principle. Datasets such as UADFV and FaceForensics++ were generated using these tools.

b) GAN-based manipulation

GANs have revolutionized face manipulation by producing high-quality, more diverse fake content. They train a generator and discriminator simultaneously to produce convincing fake faces. Notable examples include FaceShifter, FSGAN and DeepFakes which were used to generate large portions of the DeepFake-TIMIT, FaceForensics++ and ForgeryNet datasets. These methods are often better at handling identity preservation and realistic blending.

c) Graphic-based manipulation

This technique relies on more traditional graphics and animation techniques deforming and blending facial landmarks between source and target faces. While generally less realistic than GANs, these methods still play a significant role. Tools like 3D-Faceswap and the morphable-mask model are examples employed by datasets like FaceForensics++ and DFDC.

2.2 Forensic Datasets

Based on these manipulation methods, a variety of deepfake datasets have been developed. These datasets typically consist of either real videos/images or synthetic ones generated through the techniques mentioned above. Some datasets such as UADFV, Celeb-DF and DeepFake-TIMIT focus on a limited number of manipulation strategies making them less diverse but easier to use for focused studies. On the other hand, datasets like FaceForensics++, DFDC and ForgeryNet are more comprehensive, combining multiple types of manipulations and offering a more realistic evaluation environment.

To increase the challenge and realism newer datasets such as DeeperForensics-1.0, DFDC and ForgeryNet have introduced private or hidden test sets. These are designed to prevent overfitting and better simulate real-world deepfake encounters. However some lack detailed information on diversity or the visual quality of forgeries.

2.3 Forgery Detection

A wide range of detection methods have been proposed to counter the misuse of synthetic media. These methods aim to identify inconsistencies introduced during the manipulation process. Based on how they analyze the manipulated content, we categorize detection methods into two broad groups: intra and inter frame detection.

1) Intra-Frame Detection

These methods operate at the image level and treat deepfake detection as a binary classification task - deciding whether a given image is real or fake. Based on how they extract features, intra-frame methods can be grouped into three categories:

a) Knowledge-Driven Detection

These methods rely on human-defined clues also called domain-specific artifacts that are known to be introduced by certain manipulation techniques. The idea is to incorporate this prior knowledge into the model either through handcrafted features or guided supervision. These approaches often involve custom preprocessing steps to extract or enhance forgery-related clues before feeding them into a classifier.

b) Data-Driven Detection

These methods use deep neural networks trained on large datasets of real and fake images. They automatically learn discriminative features using image-level labels though the reasoning behind their decisions is often not transparent or easily interpretable.

c) Multi-Stream-Driven Detection

These models combine the strengths of both previous approaches. They are designed with multiple streams - one that focuses on predefined handcrafted features and another that learns hidden patterns from raw data. The outputs from these streams are fused to make a more informed decision.

2) Inter-Frame Detection

Unlike intra-frame methods that treat frames in isolation, inter-frame detection focuses on video-level analysis capturing temporal inconsistencies over sequences of frames. These methods are particularly useful for detecting unnatural transitions or motion artifacts that only appear across time. The majority of existing works are based on architectures such as CNN-RNN, CNN-CNN or 3D CNN. Some inter-frame approaches also integrate temporal domain knowledge such as optical flow, lip-reading consistency, biological signals to enhance deepfake detection.

2.4 Evaluation Metrics

Deepfake detection methods are typically evaluated using metrics like AUC, accuracy and precision which measure predictive performance. To handle real-world class imbalance, DFDC introduced Weighted Precision-Recall assigning higher penalties to false positives. However, most studies focus only on prediction accuracy often overlooking practical aspects like model size, inference time and computational cost which are crucial for real-world deployment.

2.5 Forensic Benchmarks

Several benchmarks such as Celeb-DF, DeeperForensics-1.0, FaceForensics++, DFDC, and ForgeryNet have been developed to evaluate deepfake detection models. These often rely on public leaderboards or challenge submissions. Yet, a major limitation of inconsistent training conditions remains. Many models are trained on different datasets making it hard to compare methods fairly. This report presents a unified benchmark that ensures consistent training settings and provides a comprehensive evaluation framework, as proposed in the referenced research.

3 Evaluation Datasets

As deepfake manipulation becomes increasingly accessible through advanced algorithms and open-source tools, there is a growing need for reliable forensic detection methods that can operate under real-world conditions. To simulate such conditions, a unified benchmark is proposed that combines multiple datasets containing various manipulation techniques and levels of visual quality.

3.1 Standard Dataset

The standard dataset used in this benchmark is constructed by integrating seven popular deepfake datasets, each containing both real and manipulated data. These include:

- **Autoencoder-based**: UADFV, Celeb-DF, DF-1.0
- **GAN-based**: DF-TIMIT (higher quality)
- **Mixed-manipulation**: FaceForensics++ (Raw), DFDC and ForgeryNet

Each sub-dataset is divided into training, validation, and test sets to support model training and evaluation.

To ensure balanced and unbiased benchmarking, the extracted data adheres to the following principles:

1. **Scale Proportionality**: The amount of data extracted from each sub-dataset is proportional to the original size of that dataset.
2. **Real-Fake Balance**: The ratio of real to fake samples in the extracted subset mirrors that of the original dataset.
3. **Frame-Video Consistency**: The total number of extracted frame-level data and consecutive video frames is kept approximately equal, allowing fair evaluation for both image-based and video-based detection methods.

In total, 2527384 frames are used as the base scale for training, validation, and testing.

If the original dataset provides a predefined training, validation and test split, it is used directly. Otherwise, a reasonable split is applied. After this, frame and video sequences are randomly extracted using a 14:1:1 ratio across training, validation and test sets.

3.2 Imperceptible and Diverse Test (ID Test) Set

To evaluate how well detection methods perform against realistic deepfakes, the Imperceptible and Diverse Test (ID Test) Set was developed. This set was built using carefully selected hard to detect examples from the 7 public datasets and an additional private dataset. The private samples were generated using two manipulation techniques: FSGAN (GAN-based) and MegaFS (autoencoder-based) trained on CelebA and FaceForensics++ data respectively.

To ensure high visual realism, we used a two-stage filtering process:

1. Detection-Based Filtering

We first applied two trained models - Xception (data-driven) and Face X-ray (knowledge-driven) to select fake samples that were often misclassified as real. This step helped us shortlist convincingly manipulated videos that fooled current detectors.

2. Human Perception Test

Next, 30 participants with experience in deepfake detection rated the realism of these videos on a scale of 1 to 5. Based on their scores, we retained only the most convincing fake samples. This final selection included 976 fake videos and 2348 real videos.

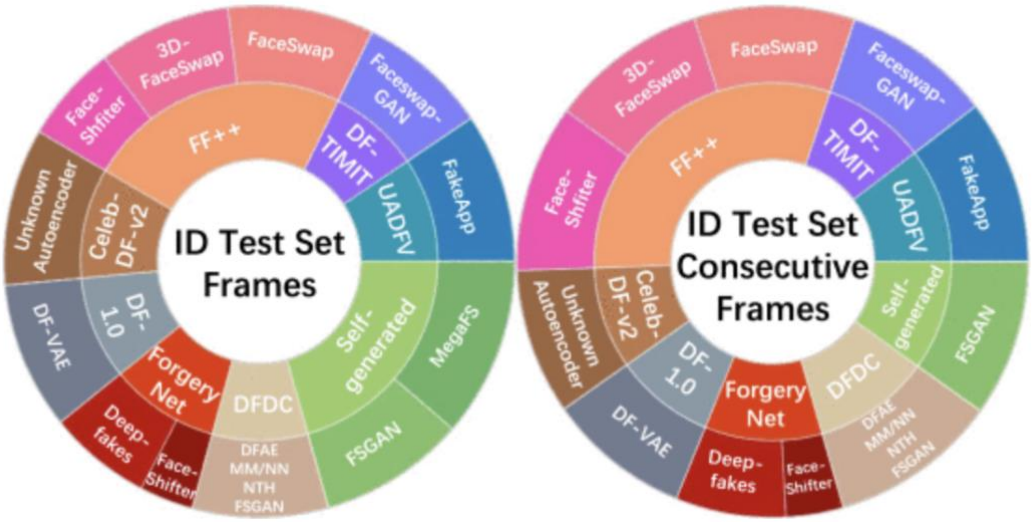
From these, we extracted 25697 fake and 25697 real images ensuring a balanced amount of frame and video level data, and even representation across different manipulation types. The given table and figure present detailed data distribution and manipulation coverage across 13 manipulation techniques.

Overview of the Data Distribution According to Manipulation Strategy in the ID Test Set

	AE	GAN	Graphic	Unknown
Video	522	202	22	230
Image	10,514	10,778	2171	2234

"Unknown" refers to data lacking relevant information of manipulation type from the source dataset.

Distribution of frame data and consecutive frame data for each manipulation method in the ID test set.



3.3 Perturbed ID Test Set

To evaluate the robustness of detection models in real-world conditions, we introduce a perturbed version of the ID Test Set. This version includes six types of commonly encountered distortions that mimic the effects of media uploads to online platforms.

Types of perturbations applied:

- **Colour Contrast Adjustment** (enhancement range: 0.5 - 2.5)
- **Colour Saturation Adjustment** (range: 0.0 - 2.5)
- **Gaussian Blur** (kernel size: 10 - 20)
- **JPEG Compression** (quality: 30 - 95)
- **White Gaussian Noise** (mean: -0.3 - 0.3, variance: 0 - 1)
- **H.264 Video Compression** (CRF levels: 23 and 40)

These perturbations are applied with random strengths to better simulate the unpredictable, lossy processing typically performed by social media platforms. Since these platforms often introduce compression and filtering effects, evaluating detection methods under such noise helps assess their performance to real-world degradations.

4 Detection Algorithms

To fairly evaluate how well different deepfake detection methods perform, this benchmark includes 13 forensic approaches (11 algorithms) that represent a wide range of detection strategies. These methods were selected based on their frequent use in past research and their significance in the field.

Intra-Frame Detection:

- **Knowledge-Driven Models:** HeadPose, Face X-ray and FWA
- **Data-Driven Models:** Xception, MesoNet, PatchForensics, FFD and Multiple-Attention
- **Multi-Stream-Driven Model:** M2TR

Inter-Frame Detection: Convolutional LSTM and LRNet

4.1 Pre-Processing

Before training deepfake detection models, it is essential to process the data properly. This process includes both general pre-processing steps common to most methods and method-specific steps tailored to individual algorithms.

General steps like extracting frames from videos, cropping faces, and aligning them are carried out for all methods. These are implemented using widely used tools like Dlib and OpenCV-Python.

Some algorithms need extra steps to extract meaningful clues or generate additional supervision signals. Some of them are:

Face X-ray: This method relies on detecting the subtle blending boundary that appears in manipulated images. To do this, it generates a special mask which is created by comparing the fake image to its original counterpart, identifying the manipulated regions and then applying a Gaussian blur and normalization. The result highlights areas that are likely fake, helping the model focus on tampered regions during training.

FWA: This method uses a self-supervised strategy to create fake looking examples called negative samples during training. To do this, a face image is first aligned at multiple

scales and one version is randomly selected. Then, it is blurred and warped back to its original size. Finally, to mimic typical deepfake artifacts, colour adjustments are made and either the full face or specific facial regions are pasted back into the original image. These manipulated samples help the model learn to detect subtle inconsistencies caused by deepfake generation.

FFD: It needs a binary mask that marks the exact areas of tampering. These are made by comparing manipulated images with their original versions, calculating pixel-wise differences, converting them into grayscale and applying a threshold. The result tells the model exactly where to look for forgery clues.

4.2 Implementation Details

To ensure fair comparisons, all selected detection algorithms were retrained on a consistent training set. As most of these algorithms did not have open-source training code, they were reimplemented based on the original descriptions provided in their respective papers. For the few algorithms with accessible source code, the official implementations were directly adopted.

Hyperparameters were initialized following the settings reported in the original works. For parameters dependent on training iterations such as learning rate decay or warm-up steps, adaptive values were computed in proportion to the relative size of the benchmark training data. For other hyperparameters such as batch size and initial learning rate, the original values were retained.

Each model was trained until convergence was achieved on the validation set and the version with the best validation performance was selected for benchmark evaluation.

Face X-ray is a knowledge-driven intra-frame detection method that uses blending boundaries referred to as Face X-rays as visual artifacts to identify manipulated regions in fake images. The model was implemented using the HRNet-W48-C backbone.

FWA is a knowledge-driven intra-frame detection method that detects deepfakes by identifying resolution inconsistencies between the manipulated face region and the surrounding background. It employs a self-supervised strategy to generate negative examples that emphasize these artifacts. ResNet50 is used as the backbone model to perform the classification.

HeadPose is a knowledge-driven intra-frame detection method that uses 3D head pose inconsistencies as a forgery clue. It extracts head pose features by comparing estimates from the entire face and the central facial region, both derived using facial landmarks. These differences are then used to train a Support Vector Machine (SVM) classifier to distinguish real faces from manipulated ones.

Mesonet-4/MesoInception-4 are intra-frame data-driven detection methods that focus on mesoscopic-level image features. MesoNet-4 uses a shallow architecture with four convolutional and pooling layers followed by a dense layer while MesoInception-4 modifies the initial layers with inception modules incorporating dilated convolutions. Both architectures aim to capture subtle manipulation artifacts in facial images.

Patch Resnet Layer1/Patch Xception Block2 are intra-frame data-driven detection methods that operate on patch-level predictions by truncating standard ResNet and Xception models at early layers. This approach helps the models focus on local manipulation clues. During training, patch-level labels guide the learning process while in the testing phase, predictions are averaged across patches to produce a final image-level decision.

Xception is an intra-frame data-driven detection method that uses XceptionNet as its backbone for binary classification of real and fake images. It leverages deep separable convolutions to efficiently capture subtle artifacts introduced during manipulation.

FFD is an intra-frame knowledge-driven detection that enhances manipulation detection by guiding the model to focus on regions likely to contain forgeries. It introduces dedicated attention layers designed to highlight subtle inconsistencies within tampered areas. For this benchmark, XceptionNet is used as the backbone network.

Multiple-Attention is an intra-frame level knowledge-driven detection method that treats deepfake detection as a fine-grained classification problem. It introduces a multi-attentional network designed to capture subtle visual artifacts often present in manipulated media. The architecture includes multiple attention maps that help focus on discriminative facial regions, densely connected convolutional layers to enhance fine-grained texture features and bilinear attention pooling to combine both low-level

and high-level representations effectively. To further improve learning, it employs a region-independent loss for each attention map and an AGDA mechanism to encourage diversity in attended regions. For this benchmark, EfficientNet-b4 is used as the backbone network.

Convo LSTM is an inter-frame level method that uses temporal information to detect deepfakes. It combines InceptionV3 for extracting features from each frame and an LSTM to capture frame-to-frame inconsistencies. For better focus, 20 consecutive face crops are used instead of raw frames. The model is trained using Adam optimizer with a learning rate of 1e-5 and batch size 4.

LRNet is an inter-frame level detection method that focuses on temporal patterns in facial landmark movements. It uses a two-stream RNN to model geometric features extracted from face landmarks enhanced by a calibration module based on optical flow to improve feature reliability. While the original setup used 60 frames, this benchmark uses 20 consecutive frames for fairness across all inter-frame models. All other settings follow the original implementation.

M2TR is an intra-frame level multi-stream-driven detection method. It combines spatial and frequency domain analysis using a two-stream architecture: a multi-scale transformer captures spatial forgery clues while frequency filters extract manipulation patterns in the frequency domain. These complementary features are fused through a cross-modality fusion block for final classification. In this benchmark, only classification loss is used for training following the official implementation with other settings kept consistent with the original paper.

To verify the correctness of our implementations especially for methods without full open-source code, we conducted validation experiments. Most detection algorithms such as Face X-ray, MesoInception-4, Multiple-attention and Conv LSTM showed comparable or improved performance relative to their original versions. However, minor discrepancies were observed in FWA on UADFV and Xception. FWA performed well on high-quality DeepfakeTIMIT samples but poorly on lower-quality UADFV indicating its reliance on training image quality. Similarly, Xception showed a performance gap, aligning with earlier reports highlighting its sensitivity to dataset variations.

5 Evaluation Metrics

To address the limitations of traditional evaluation methods and better assess deepfake detection in real-world conditions, we employ a set of metrics that go beyond simple accuracy. These include both widely used metrics like AUC (Area Under the ROC Curve) and additional analysis tools that measure robustness, efficiency, and practicality.

AUC (Area Under the Curve)

A standard metric in deepfake detection, AUC is used because it is unaffected by class imbalance. It represents the area under the ROC curve and the ROC curve plots the relationship of False Positive Rate (FPR) vs True Positive Rate (TPR) of a deepfake detection model at all classification thresholds.

$$TPR = True\ Positive / (True\ Positive + False\ Negative)$$

$$FPR = False\ Positive / (False\ Positive + True\ Negative)$$

It evaluates how well a model distinguishes between real and fake across different thresholds. We use both frame-level AUC for image-based models and segment-level AUC for video-based models. For video-level methods, we compute segment-level AUC by averaging image predictions within a video segment.

Perturbation Vs AUC

This graph illustrates the robustness of deepfake detection models under various real-world distortions, such as Gaussian blur, JPEG compression, color changes, and video compression. These perturbations simulate typical degradations introduced during uploading or sharing on social platforms. For each perturbation type in the ID test set, the AUC score is computed to evaluate the model's ability to maintain detection performance. A model that shows smaller drops in AUC under different perturbations is considered more robust and better suited for real-world deployment.

FLOPs Vs AUC

The FLOPs (Floating Point Operations) vs AUC graph provides insights into the practicability of forensic classifiers where FLOPs is a commonly used metric to calculate the computational complexity of deep learning models. An ideal model would be both accurate and computationally lightweight.

Number of Parameters Vs AUC

The correlation between the number of parameters and AUC provides a deeper understanding of practicability in which the number of parameters quantifies learnable parameters in a deep learning model. This graph helps us evaluate memory efficiency. Models that achieve high AUC with fewer parameters are more suitable for deployment on resource-constrained devices.

Inference Time Vs AUC

The inference time vs AUC graph reveals the efficiency of forensic classifiers where the inference time is estimated by testing a single face image or a single image segment consisting of sequential face images. This metric evaluates how fast a model can make predictions, which is critical for real-time or large-scale detection scenarios. Shorter inference times with strong AUC scores are preferred.

All evaluations were conducted in a consistent computing environment using an NVIDIA 2080Ti GPU and 128 GB RAM to ensure fair comparison across methods.

6 Evaluation Results

A comprehensive evaluation was conducted to assess the detection ability, generalization, robustness, practicability and efficiency-effectiveness trade-off of various benchmarked deepfake detection methods. Additionally, model interpretability was analyzed through class activation maps. The study provides a unified comparison framework ensuring consistent training and testing conditions across all methods. This allows for fair benchmarking and more meaningful insights into model behavior and deployment potential. Evaluation results were structured into the following key components:

6.1 Forgery Detection Ability

The core of any detection model is its ability to identify manipulated content accurately. Our intra-domain evaluation on the standard dataset demonstrates that most deep learning-based approaches perform exceptionally well in this setting. Methods like M2TR showed the best overall performance achieving an impressive 99.1% (frame-level) and 99.4% (segment-level) AUC because of its multi-stream design and multi-scale feature extraction. In contrast, methods like FWA-ResNet50 underperformed, largely due to their reliance on simple artifacts and lack of diverse supervision.

Notably, models trained on datasets like DFDC and ForgeryNet struggled more due to the higher complexity and variability in their respective test splits. This highlights the importance of consistent training and testing setups. Under such controlled conditions, the large performance gaps reported in earlier studies for example, between Meso4 and FFD were substantially reduced, highlighting the need for standardized evaluation to ensure fair comparisons.

Evaluation was also extended to face reenactment datasets (e.g., FF++/Face2Face and FF++/NeuralTextures) where most detection methods maintained strong performance reaffirming their applicability to different manipulation types.

6.2 Generalization Ability

Cross-domain generalization remains a critical challenge. When models trained on the standard dataset were tested on our ID test set, we observed significant performance drops across all 13 methods. M2TR again led with 67.9% frame-level AUC, demonstrating its stronger generalization potential due to its multi-modal architecture. Other top performers such as Multiple-attention and Patch-based variants also maintained relatively stable performance suggesting that focusing on high-frequency artifacts helps extract more transferable features.

A diagnostic in-domain evaluation on a subset of the ID test set revealed that about 29% of the performance drop stemmed from domain shift and 10.6% from higher-quality, subtler manipulations. This indicates that while domain differences are the primary factor, better quality forgeries also contribute to reduced accuracy.

6.3 Robustness to Perturbations

Real-world scenarios often include distortions like compression, blurring or noise. To evaluate robustness, we tested models on perturbed versions of the ID test set. White Gaussian noise had the most damaging effect, while blur and JPEG compression caused minimal disruption for knowledge-driven methods like Face X-ray and FFD. These results suggest that data-driven models are more dependent on high-frequency clues and are more susceptible to perturbations.

Interestingly, LRNet which operates on geometric landmark data instead of raw pixels showed remarkable stability across all perturbation types. This supports the hypothesis that abstract, low-dimensional representations can offer resilience against visual degradation.

6.4 Practicability Analysis

To understand which models are viable in resource-constrained environments, we analyzed FLOPs and parameter count relative to performance. M2TR once again proved effective balancing high AUC with reasonable computational demands, largely due to its efficient design involving EfficientNet-b4 and lightweight transformer components.

Patch Resnet Layer1 also stood out as a practical solution offering competitive performance with minimal computational and memory overhead making it an attractive choice for real-time applications or deployment on mobile devices.

6.5 Efficiency-Effectiveness Trade-Off

We compared the inference time against AUC to evaluate each model's real-world usability. While M2TR achieved the best accuracy, it required longer inference times due to its complex architecture. In contrast, Patch Resnet Layer1 provided a good compromise offering near-top performance with rapid inference.

Video-level methods were generally slower as they needed to process multiple frames. However, methods like LRNet which utilize landmark features instead of full images achieved faster inference while retaining acceptable accuracy indicating a direction for future lightweight model design.

6.6 Classification Decision Interpretability

To explore how models make decisions, we used Grad-CAM visualizations on real and fake samples. Models trained with weak supervision (like M2TR and Xception) tended to focus on specific local regions often lacking interpretability. On the other hand, models like Face X-ray and Multiple-attention trained with explicit supervision or attention mechanisms highlighted more meaningful forgery regions.

Cross-domain visualization revealed that many models, especially those lacking robust supervision failed to attend to relevant forgery regions when evaluated on unseen data. This further explains the generalization gap observed earlier.

In summary, the evaluation reveals that while many deepfake detection methods perform well in controlled settings, their generalization to real-world scenarios is still limited. M2TR shows top-tier accuracy and generalization due to its multi-stream design but at higher computational cost. On the other hand, models like Patch Resnet Layer1 offer a better trade-off between efficiency and effectiveness.

Robustness tests show that models relying on high-frequency features are sensitive to perturbations, whereas those using geometric or supervised features are more stable. Visualization analysis also suggest that models with guided supervision focus better on meaningful forgery regions.

Overall, no method outperforms across all criteria highlighting the need for balanced approaches that combine accuracy, robustness and practicality for real-world deployment.

7 Discussion

Although a comprehensive and consistent benchmark has been established in this work to systematically assess the strengths and limitations of widely adopted deepfake detection methods, several recently proposed algorithms are not included in the evaluation. This is primarily due to the lack of publicly available source codes or pre-trained models, which poses challenges for reproducing and benchmarking these approaches within a unified framework. To address this limitation, an online deepfake detection platform is currently under development based on the proposed benchmark. This platform aims to serve as a standardized evaluation hub for the community, enabling researchers to incorporate and assess their methods under uniform experimental conditions. The platform will be continuously updated to integrate more challenging manipulation types, newly developed detection algorithms, and broader evaluation metrics, thereby facilitating the advancement of deepfake forensics in a reproducible and scalable manner.

8 Conclusion

This work presents a unified benchmark for evaluating deepfake detection methods, offering a fair and consistent basis for comparison. Through large-scale analysis, it highlights the varying strengths and weaknesses of existing models and underscores the challenges they face in real-world conditions.

To further support the community, an online platform is being developed using this benchmark. It will provide a standardized environment for testing new approaches, encouraging broader collaboration and continual progress in the field of deepfake detection.