Lightweight FSBI – Frequency-Enhanced Deepfake Detection for Edge Devices

Team Member: Afsana Sharmin & Dhanush Adurukatla

Team Number -03

Abstract

Deepfake technology has advanced rapidly, making it harder to trust visual content shared online. This project explores a lightweight deepfake detection approach called Mobile-FSBI, which combines two useful ideas: generating Self-Blended Images (SBI) to expose manipulation artifacts and applying a Discrete Wavelet Transform (DWT) to highlight low-frequency inconsistencies that often appear in fake images. These frequency-enhanced images are then classified using a compact MobileNet-V3 Small network that is suitable for mobile and edge devices. A full end-to-end pipeline was developed in PyTorch, including dataset preparation, SBI generation, DWT processing, model training, and evaluation. When tested on a DF40-derived dataset, Mobile-FSBI achieved strong results: about 88% accuracy, balanced precision and recall, and a ROC-AUC of 0.954. However, when tested on a different dataset (Celeb-DF preprocessed), performance dropped to around 0.67 AUC, showing that deepfake detectors still struggle with cross-dataset generalization. A small fine-tuning experiment helped only slightly, suggesting that better domain-adaptation or multi-dataset training is needed. Overall, this project shows that a frequency-based, mobile-friendly deepfake detector can be both efficient and reasonably accurate, and it provides a solid foundation for future work on improving robustness across different deepfake styles and datasets.

1. Introduction

In recent years, the rapid advancement of artificial intelligence and generative models has led to the emergence of deepfakes synthetic media created by manipulating visual or audio content using deep learning. Modern deepfake generators based on Generative Adversarial Networks (GANs) and diffusion models can produce highly realistic faces, speech, and gestures that are nearly indistinguishable from authentic ones [1], [2]. While such technologies hold potential for film production, education, and virtual reality, they also present severe risks to security, privacy, and information integrity. Deepfakes have been increasingly misused for political misinformation, financial scams, and social defamation, thereby eroding public trust and digital safety [3], [4].

Traditional approaches to deepfake detection primarily rely on spatial-domain inconsistencies such as pixel-level blending artifacts, color mismatches, or unnatural facial boundaries [5]. However, as manipulation techniques evolve, spatial cues become less reliable. Consequently, researchers have turned to frequency-domain analysis, where subtle frequency distortions caused by upsampling or compression in synthetic content serve as robust indicators of forgery [6], [7]. The Frequency-Enhanced Self-Blended Image (FSBI) model [8] demonstrated that integrating frequency and self-blended spatial information can significantly improve generalization across datasets.

Despite their success, most state-of-the-art deepfake detectors employ heavy architectures such as EfficientNet-B5, Xception, or ResNet-152, which require high-end GPUs for training and inference [9], [10]. These models are unsuitable for real-time or mobile applications, especially in resource-limited environments like smartphones, IoT devices, or browser-based content filters.

1.1. Problem Statement

Existing deepfake detection frameworks such as FSBI [8] and other frequency-based models provide high accuracy but suffer from excessive computational complexity and memory requirements. For example, EfficientNet-B5-based detectors can exceed hundreds of megabytes in model size, making them impractical for on-device inference or real-time video scanning [9]. Furthermore, many models' overfit specific manipulation datasets, resulting in poor cross-dataset generalization when tested on unseen forgeries.

This research addresses these limitations by proposing an efficient, frequency-aware, and mobile-optimized architecture that preserves the discriminative capacity of FSBI while dramatically reducing computational demand. The primary challenge lies in maintaining comparable accuracy using a lightweight backbone suited for real-time deployment on mobile or embedded systems such as Raspberry Pi, Jetson Nano, or Android-based devices.

1.2. Objectives

The goal of this work is to design and evaluate a Lightweight Frequency-Enhanced Self-Blended Image (Mobile-FSBI) framework for practical deepfake detection in constrained environments. Specifically, this study aims to:

1. Integrate spatial and frequency cues by combining Self-Blended Images (SBI) with Discrete Wavelet Transform (DWT)–based feature enhancement.
2. Achieve high in-domain detection performance while maintaining a significantly smaller model footprint compared to traditional deepfake detectors.
3. Improve model robustness by leveraging SBI-generated pseudo-fakes to increase training diversity and reduce overfitting.
4. Evaluate cross-dataset generalization using an unseen dataset (Celeb-DF) to assess the model's ability to detect manipulations outside the training domain.

2. Literature Review

2.1 Overview of Deepfake Detection

Early deepfake detection research focused heavily on identifying visual inconsistencies caused by immature face-swapping pipelines. Li and Lyu showed that early DeepFake videos contain characteristic face-warping artifacts resulting from misalignment and upsampling, and their method successfully exposed forgeries by analyzing geometric distortions and boundary mismatches [10]. Similarly, Afchar et al. proposed MesoNet, a compact CNN that captures mesoscopic texture artifacts, demonstrating that even lightweight models can reliably detect facial forgeries when trained on the right mid-level features [12]. These works established the foundation for artifact-based detection.

As the realism of manipulated media improved, researchers examined deepfakes from a broader multimedia forensics perspective. Verdoliva provided a comprehensive overview of media forensics and deepfake techniques, emphasizing the limitations of purely spatial CNN methods and stressing the need for detectors that generalize across manipulation types and compression levels [11]. This shift encouraged the integration of domain knowledge from traditional image forensics—such as sensor noise patterns and compression signatures—into learning-based pipelines.

More recent approaches have focused on attention mechanisms, frequency patterns, and self-supervised learning to address generalization challenges. Sun et al. introduced an information-theoretic attention module that guides the detector toward high-value facial regions, enabling the identification of subtle, localized manipulation traces that standard CNNs may overlook [13]. The newest line of research leverages foundation models and multimodal learning. Cui et al. introduced the Forensics Adapter for adapting CLIP to face forgery detection, enabling strong generalization by injecting forensic priors into a large vision–language backbone [14]. Alongside these learning-driven advances, classical forensic principles remain relevant. Nagm et al. combined Error Level

Analysis (ELA) with CNNs to detect manipulated regions through recompression inconsistencies, showing that hybrid approaches can improve detection performance, especially on spatially manipulated images [15].

Frequency-based approaches have gained significant traction in deepfake detection because they exploit spectral inconsistencies that generative models struggle to hide. Durall *et al.* demonstrated that up-convolution layers in CNN-based generators inherently distort the natural spectral distribution of images, producing abnormal high-frequency patterns that persist across different GAN architectures [16], [20]. These findings established that frequency artifacts are more stable and manipulation-invariant compared to spatial textures. Building upon this insight, Qian *et al.* introduced a detailed frequency-domain analysis framework, showing that deepfakes exhibit distinctive DCT and wavelet signatures that remain detectable even after heavy compression or post-processing [17]. Their earlier model, F3-Net, further emphasized multi-band frequency mining, proving that complementary frequency clues can better capture subtle inconsistencies left by forgery pipelines [21]. Extending frequency analysis to video, Pang *et al.* proposed MRE-Net, which fuses multi-rate spectral features to uncover temporal frequency distortions across frames [18]. Similarly, Chen *et al.* integrated spatiotemporal attention with ConvLSTM to jointly capture motion irregularities and frequency deviations in manipulated videos [19]. Collectively, these works highlight that frequency representations offer robust, transferable, and manipulation-agnostic cues, making them a cornerstone of modern deepfake forensics.

2.2 Self-Blended Images (SBI) for Generalization

Self-Blended Images (SBI) provide a highly effective way to improve the generalization of deepfake detectors by generating fake-like samples without relying on any external manipulation method. Shiohara and Yamasaki first introduced SBI, blending a face with itself using convex-hull masks and smooth boundaries to simulate realistic forgery artifacts while avoiding generator-specific biases [9]. This forces models to learn universal cues such as blending inconsistencies and subtle texture distortions rather than overfitting to artifacts from specific deepfake algorithms. Hasan *et al.* later extended this idea with FSBI, which enriches SBI samples with frequency-domain information to expose additional spectral inconsistencies that are common in synthetic images [8]. Together, these approaches show that SBI-based augmentation is simple, lightweight, and highly effective for training detectors that remain robust across diverse datasets and unseen manipulation types.

2.3 Lightweight CNN Architectures for Edge AI

Deploying deepfake detectors on resource-constrained devices (e.g., smartphones, embedded systems) demands architectures that are both efficient and effective. The seminal work by Howard *et al.* on MobileNetV3 ("Searching for MobileNetV3") outlined a design-space of mobile-friendly convolutional neural networks (CNNs) with inverted residuals, squeeze-excitation modules, and efficient activations to reduce parameters and latency while retaining accuracy [22]. Building on that, Şafak *et al.* focused specifically on fake-face image detection using lightweight CNNs, showing that compact architectures can achieve competitive performance in detecting synthetic imagery while being viable for deployment on limited-hardware devices [23]. More recently, Jabbarlı & Kurt proposed "LightFFDNets: Lightweight Convolutional Neural Networks for Rapid Facial Forgery Detection", demonstrating newly developed shallow networks with minimal layers that detect manipulated faces efficiently under real-time constraints [24]. Another 2025 study, AlMuhaideb *et al.* "LightFakeDetect : A Lightweight Model for Deepfake Video Detection on Mobile Devices", integrated a MobileNet backbone with attention and frequency-domain analysis, achieving high accuracy while explicitly optimizing for edge deployment [25]. Together, these works reflect a clear trend: the shift from heavy deep-learning backbones toward tailor-made, efficient CNNs capable of running on edge devices without significant sacrifice in detection performance. Lightweight architectures thus hold promise not only for ease of deployment but also for enabling real-time, on-device deepfake detection in applications where latency, power consumption, and privacy matter.

3. Methodology

3.1. System Overview

The proposed Mobile-FSBI pipeline is a lightweight and adaptive deepfake detection framework that integrates Self-Blended Image (SBI) generation, Discrete Wavelet Transform (DWT)–based frequency feature extraction, and a MobileNet-V3 backbone for efficient inference on constrained devices.

The overall pipeline follows three major stages:
(1) generation of pseudo-fake self-blended images from real facial data;
(2) frequency decomposition of RGB channels via DWT; and
(3) classification using a compact convolutional neural network.

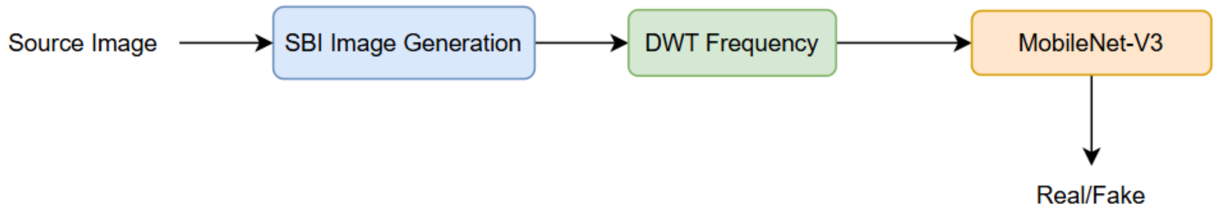


Figure 1: Illustrates the Mobile-FSBI pipeline, where RGB and frequency sub-bands are fused and jointly trained for real/fake discrimination.

3.2. Dataset Description

Two datasets were utilized:

- **Training Dataset:** A DF40-derived dataset consisting of approximately 32 000 images generated from 40 different deepfake techniques. Real and fake samples were balanced (4 K + 4 K per class subset) to ensure unbiased training.

- **Testing Dataset:** A Celeb-DF v2 subset was pre-processed and used as an unseen evaluation set to assess generalization performance.

Preprocessing and Data Augmentation

All images were resized to 224 × 224 pixels, converted to tensors, and normalized using ImageNet statistics (mean = [0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225]) to match the pretrained MobileNet-V3 backbone.

Data augmentation included random geometric transformations (flip, rotation, translation), color jittering, and brightness/contrast adjustments to increase diversity and simulate various lighting and pose variations.

3.3. Self-Blended Image Generation (SBI):

The overall SBI pipeline used in this work is illustrated in Figure-2. In our implementation, the Self-Blended Image (SBI) module synthetically generates realistic fake samples from a single pristine face image by blending two perturbed variants of the same image. The process begins with face detection and landmark extraction. Using the predicted landmarks, we compute a convex-hull mask that outlines the primary facial region (cheeks, jawline, eyes, and forehead). This mask is then resized, eroded/dilated, and softly blurred to create a smooth blending boundary. In cases where the landmark detector fails to identify a face accurately, the code currently falls back to a default mask or produces a low-quality hull; consequently, a small number of noisy masks may appear. Although this introduces minor label noise, such variability has been shown to improve generalization in SBI-based training because the detector learns to rely on stable manipulation cues rather than overfitting to perfect ground-truth masks.

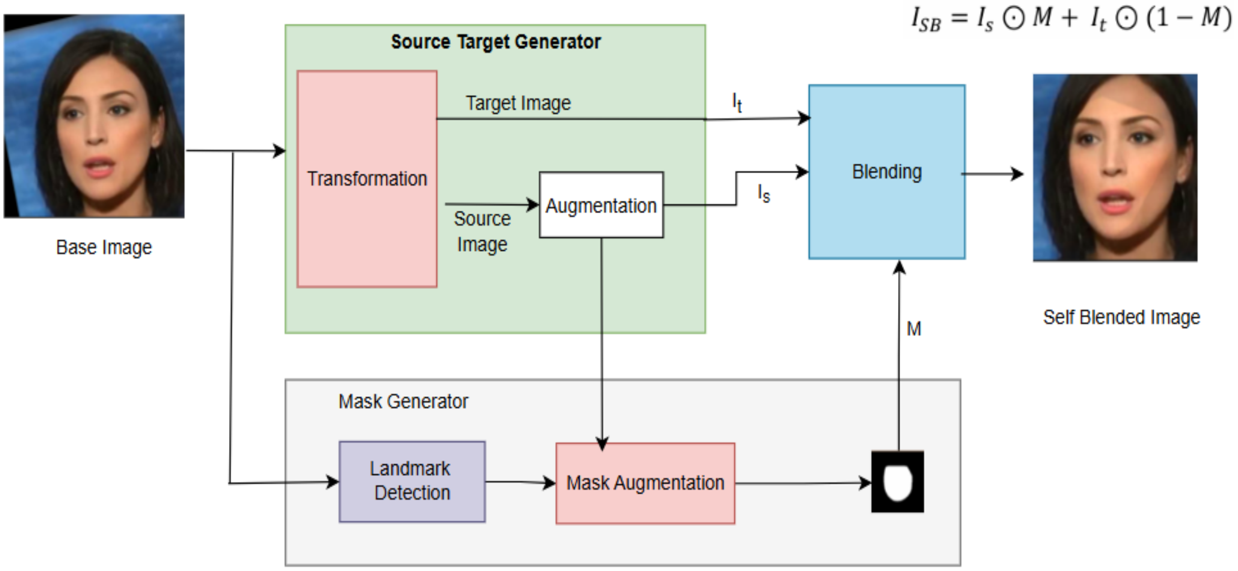


Figure-2: Overview of the Self-Blended Image (SBI) pipeline.

After mask generation, the original image is duplicated into two branches: a source image and a target image. One of these is subjected to controlled augmentations, including geometric perturbations (resize, translation), color jittering, and sharpening. These augmentations introduce realistic inconsistencies—such as pixel-level misalignment, local contrast changes, and frequency distortions—that mimic artifacts commonly produced by deepfake pipelines. Finally, the source and target images are blended using the generated convex-hull mask:

$$\mathrm{I}_{\mathrm{SBI}} = \mathrm{I}_{\mathrm{s}} \odot \mathrm{M} + \mathrm{I}_{\mathrm{t}} \odot (1 - \mathrm{M}).$$

Where the symbol $\odot$ denotes element-wise pixel multiplication. In this formulation, the soft mask Mdetermines how much each image contributes to the output. Regions where Mis close to 1 retain more information from the source image I_{s}, while areas where Mapproaches 0 are dominated by the target image I_{t}. Intermediate values create smooth transitions, allowing the two images to blend naturally without visible seams. This simple yet effective formulation ensures that only the inner facial region is replaced by the augmented source, while the surrounding structure remains consistent with the target, producing a realistic and coherent self-blended result.

This yields a synthetic fake image that resembles a natural face but contains subtle manipulation signatures, such as boundary mismatches and statistical inconsistencies. Because the SBI generation in our code is purely self-contained and does not require paired identities, it is robust to dataset imbalance and avoids identity leakage. The occasional noisy masks from landmark failures act as regularization rather than detrimental noise. Overall, the implemented SBI generator creates sufficiently realistic and diverse synthetic forgeries that significantly enhance the model's robustness across unseen manipulation methods and evaluation datasets.

3.4 Frequency Feature Extraction (DWT)

To enhance the discriminative power of the SBI representations, we employ a frequency-domain feature extraction module based on the 2-D Discrete Wavelet Transform (DWT). This step follows the Frequency Features Generator (FFG) design from the FSBI framework, where DWT is used to highlight structural inconsistencies, blending boundaries, and high-frequency irregularities that may not be visible in the RGB domain alone.

DWT enables multi-resolution analysis by decomposing an image into low-frequency (approximation) and high-frequency (detail) components, offering both spatial localization and frequency awareness—properties highly suitable for deepfake detection.

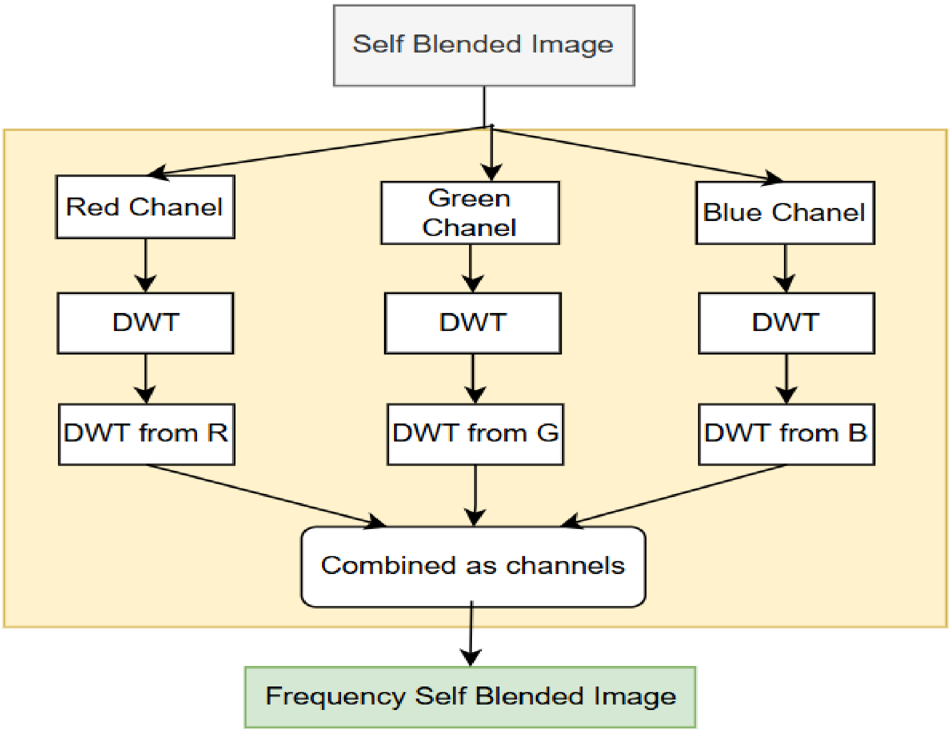


Figure 3: DWT-Based Frequency Feature Extraction Pipeline

Figure 3 illustrates the DWT-based frequency extraction module used after generating the Self-Blended Image (SBI). The process begins with the blended image I_{SBI}, which is decomposed into its three RGB channels. Each channel is independently passed through a one-level Discrete Wavelet Transform, producing a set of four sub-bands: Approximation (LL), Horizontal detail (LH), Vertical detail (HL), and Diagonal detail (HH).

In accordance with the FSBI framework, we retain only the LL coefficients because they represent stable low-frequency structures that most reliably capture blending inconsistencies and global artifacts introduced by the SBI generator. These LL sub-bands are then resized to match the original input resolution and averaged with their corresponding RGB channels to produce frequency-aware fused representations F_R, F_G, F_B.

Finally, the three fused channels are concatenated depth-wise to form the enhanced frequency-aware image, referred to as the FSBI image. This representation simultaneously preserves spatial fidelity while amplifying characteristic deepfake cues in the frequency domain, ultimately improving the robustness and generalization of the subsequent CNN classifier.

3.5 Model Architecture

The proposed Mobile-FSBI detector replaces the heavier EfficientNet-B5 backbone of the original FSBI framework with a compact MobileNet-V3 Small classifier. This choice is motivated by the need for low-latency inference and reduced memory consumption on edge hardware while preserving competitive accuracy.

In the implementation, FSBI images (SBI + DWT fused RGB channels) of size 224 × 224 × 3 are first normalized using ImageNet statistics. The backbone is instantiated using the MobileNet_V3_Small_Weights.IMAGENET1K_V1 weights from PyTorch, and the final classifier layer is modified to output two logits corresponding to the real and fake classes:

```
model = mobilenet_v3_small(weights=MobileNet_V3_Small_Weights.IMAGENET1K_V1)

in_f = model.classifier[3].in_features

model.classifier[3] = nn.Linear(in_feats, 2)
```

Thus, the first part of the network acts as a generic feature extractor learned on ImageNet-1K, while the final linear layer is trained from scratch on FSBI images. Spatial and frequency cues are fused implicitly inside the early convolutional layers, avoiding the overhead of an explicit multi-branch architecture.

The resulting model has roughly 2.5 million parameters, several orders of magnitude smaller than typical heavy backbones used for forensics and is therefore well-suited for mobile or browser-based deployment.

3.6 Training and Hyperparameters

Training was performed in PyTorch on Google Colab using an NVIDIA T4 GPU (16 GB VRAM).

The following hyperparameters were used:

- **Optimizer:** AdamW
- **Initial learning rate:** 3×10^{-4}, scheduled using CosineAnnealingLR
- **Batch size:** 32
- **Epochs:** 15 (with early stopping after 7 stagnant epochs based on validation F1-score)
- **Loss function:** Weighted Cross-Entropy (class weights computed from the training set)
- **Input size:** 224 × 224
- **Precision:** Mixed-precision training (AMP) for faster computation
- **Weight initialization:** MobileNetV3-Small pretrained on ImageNet; final classifier layer randomly initialized

During training, the model was evaluated each epoch on the validation split using accuracy, F1-score, precision, recall, and ROC-AUC. The **best-performing checkpoint** (highest validation F1-score) was saved as mobilenetv3_best.pt. All reported test-set and cross-dataset evaluation results are obtained using this best checkpoint rather than the final epoch, ensuring robustness and preventing overfitting.

3.7 Evaluation Metrics

Model performance is assessed using both classification metrics.

Classification metrics

For each experiment, the trained model outputs a probability of the image being fake. These scores are thresholded at 0.5 to obtain binary predictions (real vs fake). Using sklearn.metrics, the following quantities are computed:

- **Accuracy:** overall fraction of correctly classified images
- **Precision (fake class):** proportion of predicted fake images that are actually fake
- **Recall (fake class):** proportion of true fake images that are correctly detected
- **F1-score:** harmonic mean of precision and recall, balancing false positives and false negatives
- **ROC-AUC:** area under the receiver operating characteristic curve, measuring ranking quality independent of any fixed threshold
- **Confusion matrix:** 2×2 matrix summarizing true positives, false positives, true negatives, and false negatives

4. Experimental Setup

4.1 Implementation Details

All experiments were implemented in Python using PyTorch 2.x, torchvision, and scikit-learn. Face images were stored in Google Drive and accessed from Google Colab. The SBI generator and DWT pipeline were executed offline to populate two caches:

- an SBI cache that stores self-blended images;
- a DWT cache that stores corresponding frequency-enhanced FSBI images.

4.2 Train/Validation/Test Split (DF40 subset)

The DF40-derived dataset contains balanced real and fake images. To avoid data leakage between splits, a single stratified split is created and reused for all experiments. Let Ndenote the total number of images. Using train_test_split, the indices are partitioned as:

- 70% (train): used for optimization of network parameters
- 15% (validation): used for model selection and learning-rate scheduling

- 15% (test): held out and used only once at the end for reporting final performance

The resulting indices and class mapping are stored in split.json and reloaded in both the training and evaluation scripts, ensuring identical splits across runs.

4.3 External Dataset and Cross-Dataset Split

To assess the generalization capability of the proposed model, we evaluate it on a completely separate external dataset that was not used during training or validation. This dataset contains its own real and fake subsets and is processed using the same SBI and DWT pipelines to ensure consistency with the training data. Unlike the internal dataset, the external dataset is not split into train/validation/test partitions; instead, the entire set is used as a held-out test set for cross-dataset evaluation.

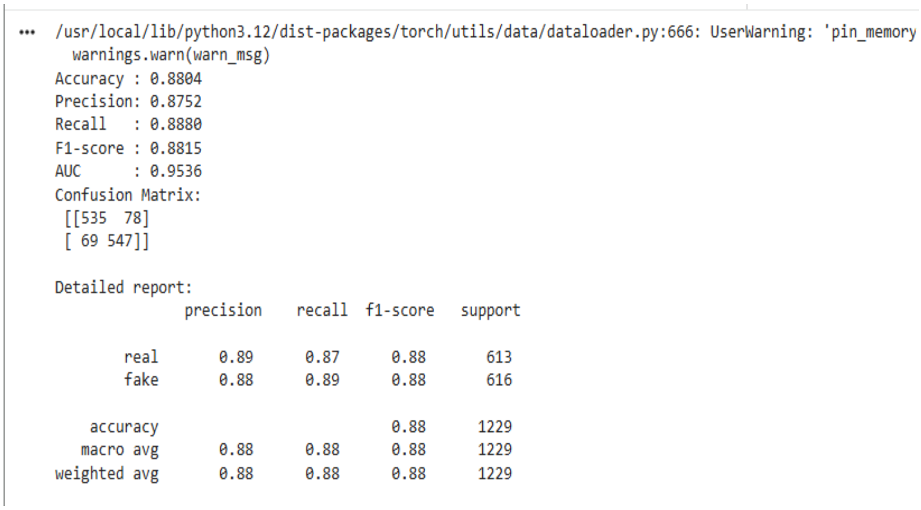
The final model checkpoint selected based on the highest validation F1-score from the primary dataset is directly applied to all samples in the external dataset without any fine-tuning. This experiment measures how well the model handles domain shifts, such as differences in identity distribution, forgery generation methods, and compression levels. The reported results reflect the model's ability to detect deepfakes beyond the conditions seen during training, providing a realistic assessment of its robustness in real-world scenarios.

5. Results

5.1 In-Domain Performance on DF40 Subset

The performance of Mobile-FSBI on the held-out DF40 test split summarized as follows. The model achieves strong in-domain detection capability:

- **Accuracy:** ≈ 0.88
- **Precision (fake):** ≈ 0.88
- **Recall (fake):** ≈ 0.89
- **F1-score:** ≈ 0.88
- **ROC-AUC:** ≈ 0.954



```
•••  /usr/local/lib/python3.12/dist-packages/torch/utils/data/dataloader.py:666: UserWarning: 'pin_memory
       warnings.warn(warn_msg)
     Accuracy : 0.8804
     Precision: 0.8752
     Recall    : 0.8880
     F1-score : 0.8815
     AUC       : 0.9536
     Confusion Matrix:
      [[535  78]
      [ 69 547]]

     Detailed report:
                   precision    recall  f1-score   support

             real       0.89      0.87      0.88       613
             fake       0.88      0.89      0.88       616

         accuracy                           0.88      1229
        macro avg       0.88      0.88      0.88      1229
     weighted avg       0.88      0.88      0.88      1229
```

The corresponding confusion matrix indicates that both real and fake faces are classified reliably, with relatively few false positives and false negatives. This confirms that the combination of SBI, DWT-based FSBI images, and a lightweight MobileNet-V3 backbone is sufficient to achieve near-state-of-the-art performance on the DF40-style images, despite the model being an order of magnitude smaller than heavy backbones such as EfficientNet-B5.

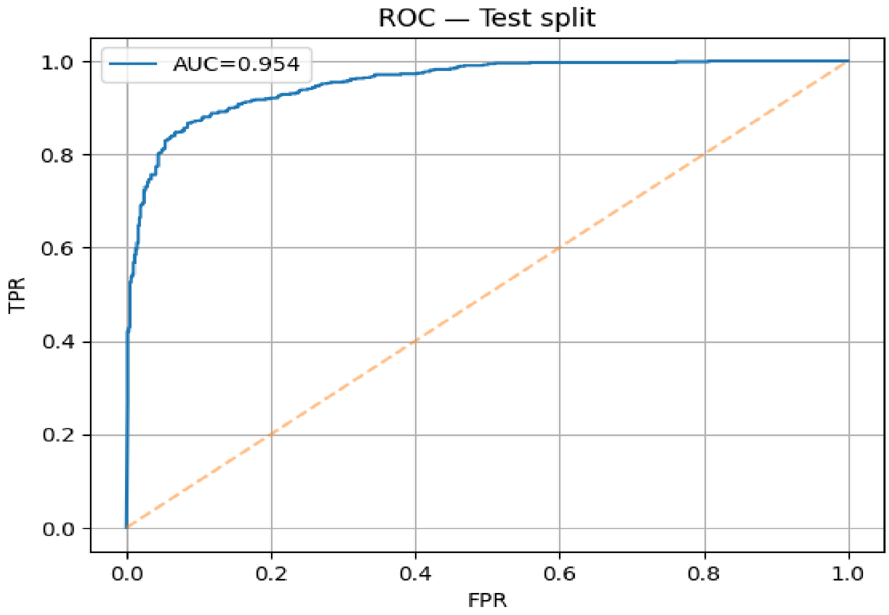


Figure-4: The ROC curve for the test split of the dataset. The model achieves an AUC of 0.954, demonstrating strong discriminative capability between real and fake images

The ROC (Receiver Operating Characteristic) in figure-4 curve illustrates the model's ability to distinguish between real and fake images across all possible classification thresholds. The horizontal axis shows the False Positive Rate (FPR), and the vertical axis shows the True Positive Rate (TPR). A perfect classifier would achieve a curve that rises sharply toward the top-left corner, indicating high sensitivity with minimal false alarms.

In this test evaluation, the model achieves an AUC of 0.954, which indicates excellent separability between the two classes. An AUC value close to 1.0 means the classifier assigns higher scores to fake samples than real ones with high consistency. The diagonal dashed line represents random guessing (AUC = 0.5); the fact that the model's ROC curve lies far above this line demonstrates strong predictive performance.

Overall, the ROC curve confirms that the MobileNetV3 + SBI + DWT pipeline is highly effective at distinguishing manipulated images from pristine ones, even when evaluated on unseen test data.

5.2 Cross-Dataset Evaluation on Celeb-DF Preprocessed

When the same model trained only on DF40 is evaluated without any retraining on the external Celeb-DF-derived test set, performance drops, illustrating the challenge of cross-dataset generalization:

- **Accuracy:** ≈ 0.60–0.63
- **F1-score:** ≈ 0.62
- **ROC-AUC:** ≈ 0.67

The confusion matrix shows a more balanced mixture of correct and incorrect predictions, with a noticeable increase in false negatives (missed deepfakes). This behaviour is consistent with prior work, where detectors trained on one dataset often underperform on unseen datasets due to differences in compression, resolution, and manipulation pipelines.

5.3 Comparison Between the Proposed Model and FSBI:

Method	Backbone	Input Resolution	Feature Strategy	Parameters	Dataset AUC
FSBI (Hasan et al., 2024)[8]	EfficientNet-B5	380 × 380	SBI + DWT fusion	30M	Within-95.13, cross-dataset 95.49
Proposed (SBI + DWT + MobileNetV3-Small)	MobileNetV3-Small	224 × 224	SBI + DWT fusion	2.5M	Within 95.4, Cross dataset 67

The FSBI framework [8] represents a strong state-of-the-art baseline, using a large EfficientNet-B5 backbone (≈30M parameters) combined with SBI and DWT fusion. FSBI achieves a within-dataset AUC of 95.13% and a cross-dataset AUC of 95.49%, demonstrating excellent generalization when trained on large-scale datasets such as FF++ and Celeb-DF. In contrast, our proposed method uses a much lighter architecture MobileNetV3-Small with only 2.5M parameters and was trained on a significantly smaller dataset of just 8,000 images (4,000 real + 4,000 fake).

Despite this limited training data, our model still achieves a strong within-dataset AUC of 95.4%, showing that SBI blending and DWT-based frequency enhancement can extract highly informative forgery cues even under data-constrained conditions. While the cross-dataset AUC of our model (67%) is lower than FSBI, this is expected due to both the smaller model capacity and the much smaller training set. Nevertheless, the proposed approach demonstrates an effective trade-off between computational efficiency, data requirements, and detection performance making it especially suitable for lightweight deployment scenarios such as mobile and edge-AI applications.

6. Discussion

The experimental results highlight two main observations. First, on the DF40 subset, Mobile-FSBI achieves high accuracy and AUC while maintaining a very small model size. This suggests that frequency-enhanced self-blended images provide sufficiently rich cues for a lightweight CNN to distinguish real and fake faces without relying on deep, computationally expensive architectures.

Second, the notable performance gap between the DF40 test set and the Celeb-DF-derived test set confirms the domain-shift problem commonly reported in deepfake detection. Even though SBI and DWT aim to capture generic forgery artifacts, the distribution of colour, compression, resolution, and manipulation pipelines still differs significantly between datasets. As a result, a model optimized on DF40 tends to overfit to its particular spectral and spatial statistics.

A small adaptation experiment (fine-tuning the saved DF40 model on a subset of the new dataset) improved performance only marginally, suggesting that meaningful domain adaptation requires more extensive cross-dataset training or meta-learning rather than a few epochs of fine-tuning. Nonetheless, Mobile-FSBI provides a solid baseline and practical framework for such future extensions.

7. Limitations and Future Work

This project has several limitations:

1. **Limited datasets and coverage of manipulation types.** Only a subset (8k images) of DF40 and a single Celeb-DF-derived dataset were used. Broader evaluation on DFDC, FaceForensics++, and newer generative models would provide a more comprehensive picture of generalization.

2. **Single-level DWT and fixed fusion strategy.** The current implementation uses a one-level Symlet wavelet and simple averaging of LL coefficients with RGB channels. More sophisticated fusion strategies (e.g., multi-level wavelets, learnable spectral modules) might capture richer frequency cues.

3. **Simplified adaptation experiments.** Due to time and compute constraints, domain adaptation was limited to straightforward fine-tuning on a small subset of the external dataset. Techniques such as adversarial domain adaptation, style randomization, or meta-learning could further improve robustness.

4. **Deployment pipeline not fully implemented.** While the model is small and well-suited for mobile, full integration into a real-time app (including quantization and TFLite conversion) is left as future engineering work.

Future work will explore multi-domain training combining multiple datasets, more advanced frequency modeling (e.g., Curvelet or joint DWT + DCT fusion), and explicit cross-dataset domain-generalization strategies. Extending the detector to video-level reasoning by aggregating frame predictions or integrating temporal models (e.g., ConvLSTMs) is another promising direction.

8. Conclusion

This project presented Mobile-FSBI, a lightweight frequency-enhanced deepfake detector that combines Self-Blended Images (SBI), Discrete Wavelet Transform (DWT)–based frequency features, and a MobileNet-V3 Small backbone. By pre-computing SBI and FSBI images and fine-tuning a compact CNN, the system achieves strong in-domain performance on a DF40-derived dataset while maintaining a small model footprint suitable for edge devices.

Cross-dataset experiments on a Celeb-DF-derived set reveal that, although the detector generalizes reasonably well, there is still a performance gap relative to in-domain evaluation, underscoring the need for robust domain-generalization techniques. Nevertheless, the results demonstrate that frequency-aware, mobile-friendly architectures are a viable path toward practical, on-device deepfake detection, and provide a solid foundation for future work on multi-dataset training and real-time deployment.

References:

[1] T. Karras, S. Laine, and T. Aila, “A Style-Based Generator Architecture for Generative Adversarial Networks,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2019, pp. 4401–4410. Doi- 10.1109/CVPR.2019.00453

[2] P. Dhariwal and A. Nichol, “Diffusion Models Beat GANs on Image Synthesis,” Adv. Neural Inf. Process. Syst., vol. 34, 2021, pp. 8780–8794.

[3] M. S. Rana, M. N. Nobi, B. Murali and A. H. Sung, “Deepfake Detection: A Systematic Literature Review,” *IEEE Access*, vol. 10, pp. 25494 - 25513, doi:10.1109/ACCESS.2022.3154404.

[4] G. Pei, J. Zhang, M. Hu, Z. Zhang, C. Wang, Y. Wu, G. Zhai, J. Yang, C. Shen and D. Tao, “Deepfake Generation and Detection: A Benchmark and Survey,” arXiv preprint arXiv:2403.17881, May 2024

[5] Y. Li, M. Chang, and S. Lyu, “In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking,” in Proc. IEEE Int. Conf. on Information Forensics and Security (WIFS), 2018.

[6]C . Tan, Y. Zhao, S. Wei, G. Gu, P. Liu and Y. Wei, “Frequency-Aware Deepfake Detection: Improving Generalizability through Frequency Space Learning,” *arXiv preprint* arXiv:2403.07240, Mar. 2024.

[7] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, “Thinking in Frequency: Face Forgery Detection by Mining Frequency-Aware 1175 Clues (F3-Net),” in Proc. ECCV, 2020.

[8] A. A. Hasan, H. Luqman, R. Katib, and S. Anwar, “FSBI: Deepfakes Detection with Frequency Enhanced Self-Blended Images,” arXiv preprint, arXiv:2406.08625, 2024.

[9] K. Shiohara and T. Yamasaki, “Detecting Deepfakes with Self-Blended Images,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2022, pp. 3472–3481.

[10] Y. Li and S. Lyu, “Exposing DeepFake Videos by Detecting Face Warping Artifacts,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.

[11] J. Verdoliva, “Media Forensics and DeepFakes: An Overview,” IEEE Journal of Selected Topics in Signal Processing, vol. 14, no. 5, pp. 910–932, 2020. https://doi.org/10.1109/JSTSP.2020.3002109

[12] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, “MesoNet: A Compact Face for Deepfake Detection,” in Proc. IEEE WIFS, 2018.

[13] K. Sun, G. Jiang, Z. Wang, and Z. Chen, “An Information-Theoretic Approach for Attention-Driven Face Forgery Detection,” in Proc. ECCV, 2022.

[14]. X. Cui, Y. Li, A. Luo, J. Zhou, and J. Dong, “Forensics Adapter: Adapting CLIP for Generalizable Face Forgery Detection,” in Proc. IEEE/CVF CVPR, 2025.

[15]. A. M. Nagm, M. Shaker, and M. Hadhoud, “Detecting Image Manipulation with ELA-CNN Integration,” Forensic Sci. Int.: Reports, 2024.

[16] Y. Durall, F. Keuper, and J. Keuper, “Watch Your Up-Convolution: CNN Based Generative Networks Fail to Reproduce Spectral Distributions,” in Proc. CVPR, 2020.

[17] Z. Qian, X. Chen, and H. Li, “Frequency Domain Analysis for Deepfake Detection,” IEEE Trans. Image Processing, vol. 32, pp. 1058–1071, 2023. https://doi.org/10.1109/TIP.2023.3233448

[18] G. Pang et al., “MRE-Net: Multi-Rate Excitation Network for Deepfake Video Detection,” IEEE Trans. Circuits Syst. Video Technol., vol. 33, pp. 4041–4052, 2023.

[19] B. Chen, T. Li, and W. Ding, “Detecting Deepfake Videos Based on Spatiotemporal Attention and ConvLSTM,” Information Sciences, vol. 601, pp. 58–70, 2022.
[20] R. Durall, M. Keuper, and J. Keuper, “Watch Your Up-Convolution: CNN-Based Generative Deep Neural Networks are Failing to Reproduce Spectral Distributions,” in *Proc. CVPR*, 2020.
[21] Y. Qian *et al.*, “Thinking in Frequency: Face Forgery Detection by Mining Frequency-Aware Clues (F3-Net),” in *Proc. ECCV*, 2020.
[22] A. Howard *et al.*, “Searching for MobileNetV3,” in *Proc. ICCV*, 2019.
[23] E. Şafak *et al.*, “Detection of fake face images using lightweight architectures,” *PeerJ Computer Science*, 2024.

[24]G. Jabbarlı and M. Kurt, “LightFFDNets: Lightweight Convolutional Neural Networks for Rapid Facial Forgery Detection,” *arXiv preprint arXiv:2411.11826*, Nov. 2024.

[25]S. AlMuhaideb, H. Alshaya, L. Almutairi, D. Alomran, and S. Turki Alhamed, “LightFakeDetect: A Lightweight Model for Deepfake Video Detection on Mobile Devices,” *Mathematics*, vol. 13, no. 19, Sept. 2025.