Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Evaluating Frequency and Noise-Residual Features for Robust AI-Generated Image Detection

Ahmad Abdelrahman

ahmedissa.it@hotmail.com

independet researcher https://orcid.org/0009-0000-5251-5274

Research Article

Keywords: AI-generated image detection, synthetic image forensics, cross-generator generalization, frequency analysis, noise residual, MobileNetV2, ablation study, robustness

Posted Date: November 11th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11688933/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Evaluating Frequency and Noise-Residual Features for Robust AI-Generated Image Detection

Ahmed Yousef Abdelrahman

Abstract. Adding hand-crafted forensic features, such as Fourier spectra or high-pass noise residuals, is a common way to help detectors distinguish AI-generated images from photographs. Yet it remains unclear whether these extra inputs actually help when a detector encounters generators it has not seen during training. We examine this question through a controlled ablation using MobileNetV2 and four input configurations: RGB alone, RGB with an FFT channel, RGB with an SRM residual channel, and RGB with both features. Experiments use 19,964 images from CommunityForensics-Small, representing 269 generators. We hold out 54 generators, repeat training with three random seeds, and test the models under JPEG compression and down-scaling. RGB alone gives the best clean-image accuracy: 94.77% on known generators and 93.91% on unseen ones. The three feature-augmented configurations reach 92.67%–93.56% and 93.15%–93.24%, respectively. None outperforms RGB significantly in any of the 18 comparisons after Holm correction of exact McNemar tests. The residual channel is especially vulnerable to resizing: at 50% down-scaling, accuracy on unseen generators falls from 87.31% for RGB to 80.54% with the residual alone and 82.33% with both additional channels. Importantly, 53 of the 54 held-out generators belong to the latent-diffusion family, while the single held-out GAN is detected in only 31%–37% of cases. The findings therefore support generalization mainly within a generator family, not across architectures. In this setting, the extra channels increase complexity without a measurable improvement.

Keywords: AI-generated image detection; synthetic image forensics; cross-generator generalization; frequency analysis; noise residual; MobileNetV2; ablation study; robustness

1. Introduction

Text-to-image systems can now produce images that are difficult to distinguish from photographs, and large numbers of fine-tuned generators are publicly available [1, 2]. This creates a practical challenge for detection. A model that recognizes only the generators used during training may struggle as soon as it encounters a new one. For real-world use, a detector needs to generalize beyond its training generators and remain reliable after ordinary processing steps, such as image compression and resizing [3, 4, 5].

One proposed solution is to provide a detector with forensic features alongside the original RGB image. Fourier-based features can reveal patterns associated with up-sampling in generative networks [6, 7, 8], while high-pass residuals may capture local pixel statistics that differ between natural and synthetic images [9]. These features are often added as input channels. Although the idea is reasonable, strong performance from a model that uses them does not, by itself, demonstrate that they are responsible for the improvement.

Our study builds on earlier work [10] that trained an EfficientNet-B2 detector using RGB, FFT, and artifact-residual inputs. On 40,000 images from CommunityForensics-Small, that system reported 94.53% test accuracy and 98.57% AUROC. However, its test images followed the same distribution as its training data. As the earlier paper acknowledged, those results did not establish performance on generators excluded from training or under image transformations. Nor did they isolate the effect of each additional channel. Here, we address those questions with a smaller, tightly controlled experiment.

We make five contributions. First, we compare RGB, RGB+FFT, RGB+Residual, and RGB+FFT+Residual while keeping the backbone, training data, optimization settings, and decision threshold fixed. Second, we remove duplicates before creating a generator-disjoint split that reserves 54 generators entirely for testing. Third, we test all configurations on the same images under four JPEG settings and three down-scaling levels. Fourth, we repeat each experiment with three seeds and report bootstrap intervals and paired statistical tests with multiple-comparison correction. Finally, we report an important negative result: neither additional feature improves performance consistently, and the residual feature reduces robustness under image processing.

2. Related Work

Generalization of learned detectors. Wang et al. [3] demonstrated that a conventional CNN trained on images from one GAN can sometimes generalize to other CNN-based generators, particularly with blur and JPEG augmentation. Later work showed that this transfer is sensitive to model architecture and post-processing [4], and that moving from GAN images to diffusion images is especially challenging [5]. Ojha et al. [11] noted that real-versus-fake classifiers can become tied to artifacts from their training generators and explored pretrained vision-language features as an alternative. Park and Owens [1] approached the problem through training diversity: their Community Forensics dataset contains images from thousands of generators, and their experiments showed that broader generator coverage improves detection of unseen models.

Frequency and residual cues. Prior studies identified frequency-domain artifacts caused by up-sampling in GANs [6, 7], and Zhang et al. [8] investigated classifiers trained directly on spectral representations. High-pass residuals have a different origin in steganalysis, particularly the spatial rich model (SRM) filter bank of Fridrich and Kodovský [9]. They are now also used in image-forensics pipelines, either as fixed initial filters or as additional model inputs. However, studies often assess these features as part of a complete system rather than testing their individual contribution with a generator-disjoint evaluation.

Lightweight backbones. MobileNetV2 [12] and EfficientNet [13] are practical options when computational resources are limited. The earlier study [10] used a five-channel EfficientNet-B2. We use MobileNetV2 here because its lower training cost makes it feasible to run all four configurations with three seeds each on a single consumer GPU.

3. Materials and Methods

3.1. Data

We use CommunityForensics-Small [1], distributed across 186 parquet shards ranging from 1 to 4 GB. Synthetic images are organized by generator, while real images are grouped by source collection. To build a manageable sample on a free cloud notebook, we visited shards in a fixed random order (seed 2025) and read only their first 1,000 rows. We excluded shards larger than 2.5 GB to avoid loading oversized files. Sampling was capped at 40 images per generator and 300 real images per real-image shard, with targets of 10,000 synthetic and 10,000 real images. This initial collection included 272 generators.

Before storage, every image was converted to RGB, resized by bicubic interpolation until its shorter side reached 224 pixels, and center-cropped to 224×224. We avoided lossy re-encoding during this step. Representative examples appear in Figure 1.



Figure 1. Examples from the sampled pool after preprocessing. Top row: real images. Bottom row: AI-generated images.

3.2. Leakage Control and Generator-Disjoint Split

We removed duplicates before splitting the data: exact duplicates were identified using SHA-1 hashes of file bytes, and near duplicates using matching 64-bit difference hashes. In total, 29 images were removed.

We then excluded generators with fewer than six remaining images and matched the number of real images to the available synthetic images. The final dataset contained 19,964 images from 269 generators.

To prevent generator leakage, we assigned generators to splits before assigning their images. Within each architecture family, 20% of the generators were reserved for the unseen-generator test set and were excluded from training and model selection. Images from the remaining generators were divided into training, validation, and known-generator test sets in a 70/15/15 ratio. Real images have no associated generator, so we split them randomly using the same proportions. All configurations shared a single fixed split, ensuring that their scores were calculated on identical test images (Table 1).

Table 1. Generator-disjoint data split. The 54 unseen generators share no generator with the other three sets.

Split	Real images	Fake images	Generators	Used for
Train	5,526	5,578	215	Weight updates
Validation	1,184	1,206	215	Early stopping and checkpoint selection
Test, known generators	1,236	1,206	215	In-distribution test
Test, unseen generators	2,033	1,995	54	Cross-generator test
Total	9,979	9,985	269	

3.3. Input Representations

The additional features are calculated from the RGB tensor on the GPU, after applying any test-time perturbation. This matches the intended inference workflow: the feature channels reflect the image as received by the detector. We first convert RGB to grayscale, denoted by g, using weights of 0.299, 0.587, and 0.114.

RGB (3 channels). The input image is scaled to [0, 1] and normalized with ImageNet means and standard deviations.

FFT (1 channel). We take the orthonormal two-dimensional Fourier transform of g, shift the zero-frequency component to the center, and compute $\log(1 + |F(g)|)$. The resulting map is standardized separately for each image to zero mean and unit variance.

Residual (1 channel). We convert g to the 0–255 scale and convolve it, using reflection padding, with the fixed 5×5 high-pass "KV" filter from the SRM bank [9]. The resulting residual is standardized per image and clipped at ±5 standard deviations.

We concatenate these representations at the network input (early fusion), producing three, four, four, or five channels depending on the configuration. Figure 2 illustrates the extra channels for a real image and an AI-generated image.

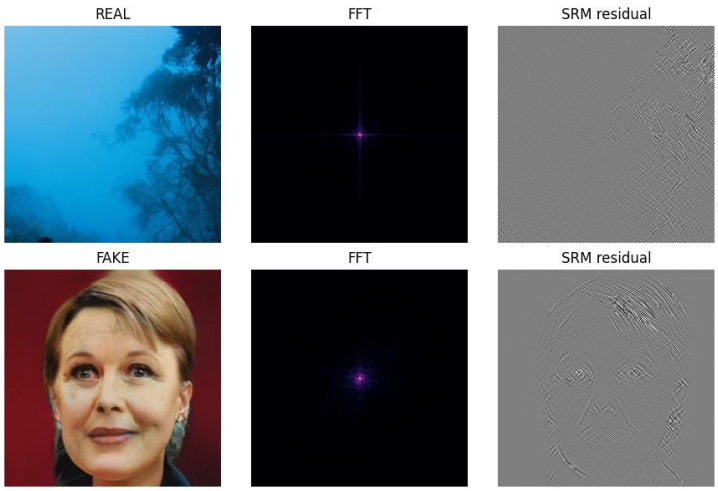


Figure 2. Input representations for a real image (top) and a generated image (bottom): RGB, standardized log-magnitude FFT, and standardized SRM noise residual.

3.4. Model and Training

Our detector uses ImageNet-pretrained MobileNetV2 [12], implemented through timm [14], with a two-class classification head and end-to-end fine-tuning (2.23 million parameters). When a configuration uses more than three input channels, we expand the first convolution. Its RGB weights remain unchanged, while each additional channel is initialized using the mean of those three pretrained filters. In this way, every configuration begins with the same pretrained RGB pathway.

All four models follow the same training procedure (Table 2). We use AdamW [15], a one-cycle learning-rate schedule [16], mixed-precision training, and gradient-norm clipping. Random horizontal flipping is the only augmentation. We intentionally exclude blur and JPEG augmentation, since these could obscure the effect of the input representation in our robustness comparisons. The checkpoint with the lowest validation loss is retained, and training stops after three epochs without improvement. Each configuration is trained with three seeds, giving twelve runs in total. On an NVIDIA Tesla T4, one epoch takes approximately 20 seconds.

Table 2. Experimental settings shared by all four configurations.

Setting	Value
Backbone	MobileNetV2 (ImageNet-pretrained), fine-tuned end to end, 2.23 M parameters
Input configurations	RGB; RGB+FFT; RGB+Residual; RGB+FFT+Residual (3, 4, 4 and 5 channels)
Input size, batch size	224×224, 64
Optimizer	AdamW, peak learning rate 0.0003, weight decay 0.0001, one-cycle schedule (10% warm-up)
Epochs	At most 8; early stopping with patience 3 on validation loss
Regularization	Random horizontal flip; gradient-norm clipping at 1.0
Seeds	42, 123, 2026
Decision threshold	0.5 on P(fake), fixed for all models and test conditions
Test perturbations	JPEG quality 90, 70, 50, 30; down-scaling to 75%, 50%, 25%

3.5. Evaluation Protocol

We evaluate the selected checkpoint from each run on both test sets under eight conditions: unmodified images; JPEG re-encoding at qualities 90, 70, 50, and 30; and down-scaling to 75%, 50%, or 25% of the original side length. For resizing, bicubic interpolation with anti-aliasing is followed by 8-bit rounding and bilinear up-scaling to 224×224. Because the transformations are deterministic, every model receives exactly the same altered images (Figure 3). We treat synthetic images as the positive class and report accuracy, F1, AUROC, sensitivity, and specificity. The classification threshold remains 0.5 throughout and is never tuned using test data. We define the generalization gap as performance on known generators minus performance on unseen generators, and a generator's detection rate as the percentage of its images predicted to be synthetic.



Figure 3. The eight test conditions applied to one test image: clean, four JPEG qualities and three down-scaling factors.

3.6. Statistical Analysis

We report means and standard deviations across the three random seeds. For accuracy, we compute 95% percentile bootstrap confidence intervals [17] using 1,000 resamples of test images and per-image correctness averaged across seeds. We also compare each augmented-input model against RGB on the same images using an exact McNemar test [18]. For these paired tests, image-level predictions are determined by majority vote across the three seeds. The 18 tests—three feature variants, two test sets, and three conditions—are adjusted using Holm's method [19].

4. Results

4.1. Performance on Clean Images

Table 3 summarizes results on clean images. RGB alone achieves the highest accuracy, F1, and AUROC on both test sets. It also shows the least variation across seeds on known generators. Adding forensic channels reduces known-generator accuracy by approximately 1.2–2.1 percentage points. Results on unseen generators are closer: the four configurations differ by less than 0.8 points, with overlapping confidence intervals. However, the augmented models are more likely to label real images as fake. Their specificity on unseen generators ranges from 93.4% to 94.3%, compared with 95.8% for RGB, while sensitivity changes by less than a point. The extra channels therefore increase false positives without a corresponding improvement in synthetic-image detection.

Table 3. Clean test results in percent, mean ± SD over three seeds. CI: 95% bootstrap confidence interval of accuracy.

Test set	Input	Accuracy	95% CI	F1	AUROC	Sensitivity	Specificity
Known	RGB	**94.77 ± 0.48**	94.06–95.45	**94.65 ± 0.51**	**98.62 ± 0.14**	93.59 ± 0.93	95.93 ± 0.05
	RGB+FFT	93.28 ± 1.42	92.57–94.06	93.10 ± 1.53	98.02 ± 0.67	91.96 ± 2.62	94.58 ± 0.98
	RGB+Residual	93.56 ± 1.05	92.81–94.31	93.45 ± 1.09	98.10 ± 0.46	93.09 ± 1.51	94.01 ± 0.71
	RGB+FFT+Residual	92.67 ± 1.15	91.84–93.49	92.56 ± 1.12	97.93 ± 0.46	92.26 ± 0.98	93.07 ± 2.11
Unseen	RGB	**93.91 ± 0.72**	93.30–94.47	**93.73 ± 0.76**	**98.24 ± 0.29**	92.00 ± 1.11	95.79 ± 0.44
	RGB+FFT	93.24 ± 1.67	92.62–93.83	93.10 ± 1.77	97.79 ± 0.62	92.20 ± 2.83	94.26 ± 1.17
	RGB+Residual	93.15 ± 1.20	92.52–93.76	93.07 ± 1.22	97.94 ± 0.49	92.92 ± 1.27	93.38 ± 1.14
	RGB+FFT+Residual	93.20 ± 0.62	92.58–93.79	93.10 ± 0.55	97.78 ± 0.35	92.53 ± 0.53	93.85 ± 1.70

Figure 4 provides another view of the training results. Training loss drops below 0.04 in every run, but the lowest validation losses remain around 0.21–0.28 and stop improving after only a few epochs. This pattern suggests overfitting that early stopping does not fully prevent. The five-channel configuration also has the highest validation loss through most of training.

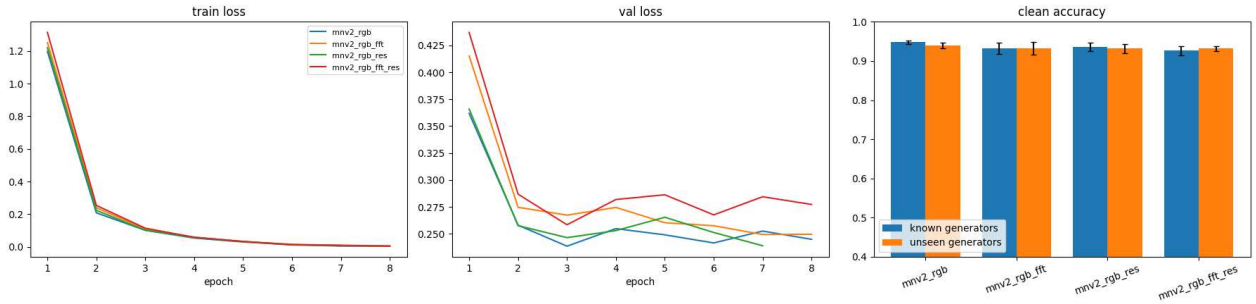


Figure 4. Training loss and validation loss averaged over seeds (left, center), and clean accuracy on known and unseen generators with standard deviation over seeds (right).

4.2. Cross-Generator Generalization

Generalization gaps are small across all four configurations: no more than about 0.9 percentage points for accuracy and 0.4 points on the AUROC ×100 scale (Table 4). Some feature-augmented models have a smaller gap than RGB, but this should not be mistaken for stronger generalization. Their performance on unseen generators is not better; rather, their accuracy on known generators is lower.

Table 4. Generalization gap (known minus unseen), mean ± SD over three seeds. Accuracy in percentage points, AUROC ×100.

Input	Known acc.	Unseen acc.	Gap (acc.)	Known AUROC	Unseen AUROC	Gap (AUROC)
RGB	94.77 ± 0.48	93.91 ± 0.72	0.86 ± 0.33	98.62 ± 0.14	98.24 ± 0.29	0.38 ± 0.23
RGB+FFT	93.28 ± 1.42	93.24 ± 1.67	0.05 ± 0.28	98.02 ± 0.67	97.79 ± 0.62	0.23 ± 0.11
RGB+Residual	93.56 ± 1.05	93.15 ± 1.20	0.41 ± 0.73	98.10 ± 0.46	97.94 ± 0.49	0.16 ± 0.24
RGB+FFT+Residual	92.67 ± 1.15	93.20 ± 0.62	−0.53 ± 0.55	97.93 ± 0.46	97.78 ± 0.35	0.15 ± 0.14

The composition of the unseen set is crucial when interpreting these small gaps. Of the 54 held-out generators, 53 are latent-diffusion systems [2], as are nearly all training generators. Across these 53, mean

detection rates are about 93%–94%, and between 42 and 45 of the 54 held-out generators achieve detection rates of at least 90% (Table 5). The remaining model, GigaGAN [20], is detected only 31%–37% of the time. Because this result comes from just one GAN and 40 test images, it signals a possible transfer problem but does not quantify cross-architecture generalization.

There is also an informative exception within the diffusion family. For SDXS [21], a one-step distilled generator, the RGB detector recognizes only 29% of its images. Adding FFT raises this figure to 76%, while combining both channels produces 59%. These are results for a single generator with a small test sample, so we treat them as a useful observation rather than evidence of a general advantage.

Table 5. Detection rate on unseen generators (share of a generator's test images classified as fake), averaged over three seeds.

Group	RGB	RGB+FFT	RGB+Residual	RGB+FFT+Residual
Latent diffusion, mean of 53 generators	0.927	0.932	0.936	0.937
GAN, 1 generator (GigaGAN)	0.367	0.358	0.367	0.308
Median over all 54 generators	0.957	0.948	0.958	0.958
Generators detected ≥ 90% (of 54)	42	43	45	45
SDXS (one-step distilled latent diffusion)	0.286	0.762	0.349	0.587

4.3. Robustness to JPEG Compression and Down-Scaling

Table 6 and Figure 5 show how accuracy changes after JPEG compression and resizing. All configurations lose performance under these transformations. Even JPEG quality 90 reduces accuracy by roughly four to six points, which is unsurprising given that we did not use compression augmentation during training [3]. Under JPEG compression, RGB and the five-channel model remain fairly close; at qualities 90 and 70, the latter is nominally ahead by at most 0.8 points, within the variability between seeds. RGB+FFT and RGB+Residual generally fare worse. At quality 30 on unseen generators, their accuracies fall to 80.8% and 79.6%, respectively, compared with 84.3% for RGB.

Resizing reveals much larger differences. When unseen-generator images are reduced to 25% of their original size, RGB still reaches 83.5% accuracy. RGB+FFT falls to 78.6%, RGB+FFT+Residual to 73.3%, and RGB+Residual to 67.3%, with considerable variability across seeds in the last configuration. The same ranking appears for known generators. AUROC tells a similar story under resizing: at 25% scale on unseen generators, RGB achieves 0.918, compared with 0.851–0.874 for the augmented models. At JPEG quality 30, the five-channel model has a slightly higher AUROC than RGB (0.938 versus 0.927), but this does not improve accuracy at the fixed decision threshold; we did not test that AUROC difference for statistical significance.

Table 6. Accuracy in percent under JPEG compression (quality Q) and down-scaling (scale S), mean over three seeds. Best value per column and test set in bold.

Test set	Input	Clean	Q90	Q70	Q50	Q30	S75	S50	S25
Known	RGB	**94.77**	88.77	85.69	**85.16**	**84.71**	**90.76**	**87.33**	**83.51**
	RGB+FFT	93.28	**88.83**	84.96	83.07	81.85	88.77	85.87	79.21
	RGB+Residual	93.56	87.74	84.73	82.38	80.85	87.84	80.29	68.70
	RGB+FFT+Residual	92.67	88.63	**86.10**	84.97	83.69	87.81	81.61	73.45
Unseen	RGB	**93.91**	88.18	85.35	**84.79**	**84.27**	**90.18**	**87.31**	**83.54**
	RGB+FFT	93.24	88.63	84.49	82.57	80.79	88.89	85.86	78.57
	RGB+Residual	93.15	87.41	83.31	81.32	79.60	87.50	80.54	67.34
	RGB+FFT+Residual	93.20	**88.94**	**85.83**	84.56	83.06	88.17	82.33	73.30

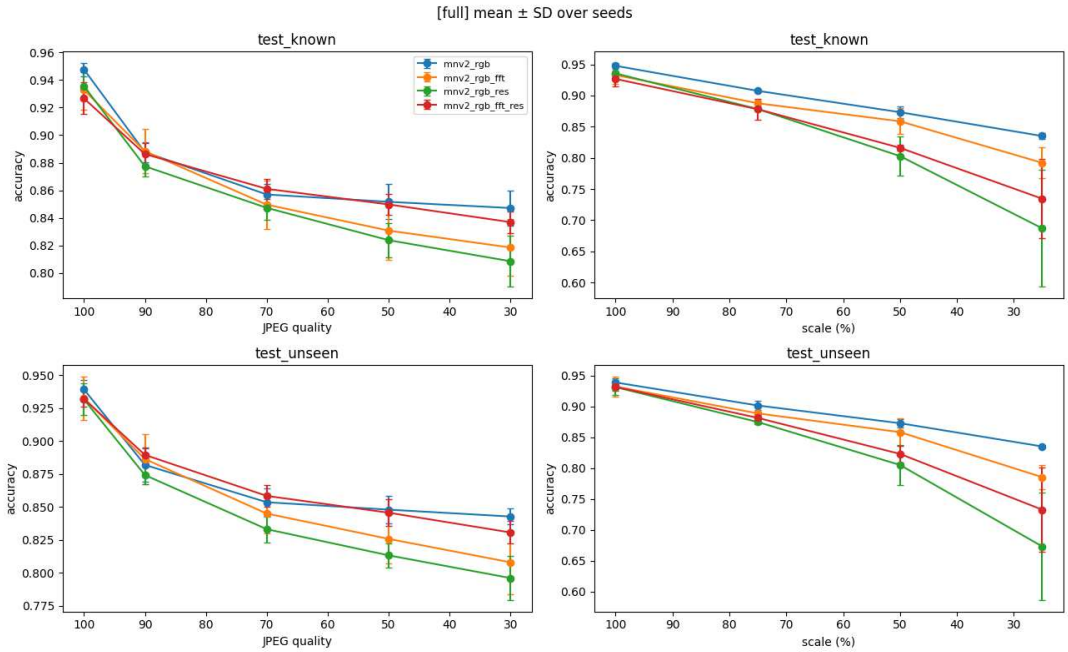


Figure 5. Accuracy under JPEG compression (left) and down-scaling (right) on known generators (top) and unseen generators (bottom). Points are means over three seeds and bars are standard deviations.

4.4. Statistical Comparison with the RGB Baseline

The paired statistical tests are reported in Table 7. Not one augmented-input configuration significantly outperforms RGB across the 18 comparisons. The only positive accuracy difference is 0.33 points for the five-channel model on known generators at JPEG quality 50, and its adjusted p-value is 1.00. In contrast, nine comparisons show significantly worse results after correction. At 50% down-scaling, both configurations containing the residual channel lose approximately 5.0–7.8 percentage points on the two test sets. RGB+Residual and RGB+FFT also lose around 1.8–4.1 points at JPEG quality 50, while the five-channel model is about 2.1 points worse on clean known-generator images.

Table 7. Accuracy difference to the RGB baseline in percentage points (seed-majority vote, identical images) with Holm-adjusted exact McNemar *p*-values in parentheses. Bold: significant at 0.05.

Input	Test set	Clean	JPEG quality 50	Down-scaling 50%
RGB+FFT	Known	−1.11 (0.119)	**−1.76 (0.031)**	−0.20 (1.000)
	Unseen	0.00 (1.000)	**−2.43 (<0.001)**	−0.52 (1.000)
RGB+Residual	Known	−0.86 (0.277)	**−2.95 (<0.001)**	**−7.78 (<0.001)**
	Unseen	−0.72 (0.209)	**−4.12 (<0.001)**	**−7.27 (<0.001)**
RGB+FFT+Residual	Known	**−2.05 (<0.001)**	+0.33 (1.000)	**−5.86 (<0.001)**
	Unseen	−0.52 (0.618)	−0.99 (0.277)	**−5.04 (<0.001)**

5. Discussion

The extra channels did not improve generalization. For the MobileNetV2 backbone and dataset examined here, the answer is clear: adding FFT or residual channels does not provide a consistent benefit. These channels are computed entirely from RGB pixels, so they do not introduce new information; they present existing information in a different form. An end-to-end-trained CNN may already be able to learn the high-frequency features it needs from RGB alone, consistent with previous findings on RGB-based detectors [3, 4]. The extra representations then increase the input dimensionality and, in several runs, the variation across seeds.

The residual channel reduced robustness. One possible explanation is that the SRM filter emphasizes very fine pixel-level patterns—the same patterns that resizing and interpolation tend to remove. After down-scaling and restoring an image to its original size, the standardized residual may reflect the interpolation process more than the image's origin. A detector relying on this feature could therefore lose a useful signal. JPEG quantization may create a related problem. We did not directly test this mechanism, so

it remains a hypothesis. FFT appears less harmful, although early fusion may not be the best way to combine a global spectrum with spatial convolution; separate branches or late fusion could behave differently.

Changes in the decision threshold may explain part of the apparent robustness loss. Accuracy declines more sharply than AUROC after perturbation. For RGB on unseen generators, JPEG quality 30 reduces accuracy from 93.9% to 84.3%, whereas AUROC falls from 0.982 to 0.927. The detector still separates many real and fake images in its ranking, but a fixed threshold of 0.5 becomes less suitable. Calibration using perturbed validation images or training with compression augmentation might recover some of the lost accuracy.

Relation to the earlier study. On known generators, our RGB-only MobileNetV2 reaches 94.77% accuracy and 98.62% AUROC. These values are close to the 94.53% accuracy and 98.57% AUROC reported for the five-channel EfficientNet-B2 in [10]. The experiments use different backbones, samples, data splits, and threshold-selection procedures, so the figures cannot be compared directly. Still, they raise the possibility that the earlier model's performance did not depend strongly on its FFT and artifact features. A matched ablation using EfficientNet-B2 would be needed to test that possibility.

What the unseen-generator result means. The detector loses relatively little accuracy when evaluated on the 54 held-out generators, supporting earlier evidence that diverse generator training improves transfer to new models [1]. Yet most of the held-out systems are variants of the same latent-diffusion family. The low detection rate on GigaGAN shows why the distinction matters: strong transfer within one family should not be presented as reliable transfer across architectures. The SDXS result also leaves open the possibility that frequency features are valuable for particular generators even if they do not help on average.

6. Limitations

Several limitations shape how these findings should be interpreted. First, all but one held-out generator belong to the same architecture family as the training models, so we cannot claim broad cross-architecture generalization. Second, we study only MobileNetV2 with early fusion; a larger backbone or a different fusion strategy might yield other results. Third, the 19,964-image dataset is a subset of a much larger collection. We sampled from the beginning of visited shards and skipped shards larger than 2.5 GB, which may exclude some high-resolution real-image sources. Fourth, differences in acquisition, resolution, or compression history between the real and synthetic collections could create source-related shortcuts. Fifth, resizing everything to 224×224 may itself weaken the high-frequency signals being evaluated. Sixth, three seeds provide limited statistical power for seed-level inference; the paired tests are therefore conducted at image level. Seventh, the JPEG and resizing perturbations are controlled simulations rather than complete social-media processing pipelines. Finally, matching perceptual hashes removes some near duplicates but cannot remove images generated from the same prompt with different visual content.

7. Conclusion

In this generator-disjoint, multi-seed study, we find no consistent advantage from adding FFT or SRM noise-residual channels to a lightweight RGB detector. The residual channel, in particular, makes the model more vulnerable to down-scaling and JPEG compression. Using about 11,000 training images, RGB-only MobileNetV2 reaches 93.9% accuracy on the held-out generator test set, which consists almost entirely of latent-diffusion models. All configurations, however, perform poorly on the single held-out GAN. The results suggest that extra forensic channels should be justified through controlled comparisons rather than assumed to help. Future experiments should hold out entire generator families, repeat the comparison with EfficientNet-B2 [10], investigate early versus late feature fusion, and include compression and resizing augmentation to determine whether these cues become useful when robustness is built into training.

Data and code availability. CommunityForensics-Small is publicly available through the Hugging Face Hub (OwensLab/CommunityForensics-Small). The author can provide the notebook used for sampling, splitting, training, and generating the tables and figures upon request.

References

[1] Park, J., & Owens, A. (2025). Community Forensics: Using thousands of generators to train fake image detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

[3] Wang, S.-Y., Wang, O., Zhang, R., Owens, A., & Efros, A. A. (2020). CNN-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 8695–8704).

[4] Gragnaniello, D., Cozzolino, D., Marra, F., Poggi, G., & Verdoliva, L. (2021). Are GAN generated images easy to detect? A critical analysis of the state-of-the-art. In *IEEE International Conference on Multimedia and Expo (ICME)*.

[5] Corvi, R., Cozzolino, D., Zingarini, G., Poggi, G., Nagano, K., & Verdoliva, L. (2023). On the detection of synthetic images generated by diffusion models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

[6] Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., & Holz, T. (2020). Leveraging frequency analysis for deep fake image recognition. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, PMLR 119 (pp. 3247–3258).

[7] Durall, R., Keuper, M., & Keuper, J. (2020). Watch your up-convolution: CNN based generative deep neural networks are failing to reproduce spectral distributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 7890–7899).

[8] Zhang, X., Karaman, S., & Chang, S.-F. (2019). Detecting and simulating artifacts in GAN fake images. In *IEEE International Workshop on Information Forensics and Security (WIFS)*.

[9] Fridrich, J., & Kodovský, J. (2012). Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security, 7*(3), 868–882.

[10] Abdelrahman, A. Y. (2026). AI-Generated Image Detection Using EfficientNet-B2 with RGB,FFT and Artifact Features. *Preprints*. https://doi.org/10.20944/preprints202610.3096.v1

[11] Ojha, U., Li, Y., & Lee, Y. J. (2023). Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 24480–24489).

[12] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4510–4520).

[13] Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, PMLR 97 (pp. 6105–6114).

[14] Wightman, R. (2019). PyTorch Image Models. GitHub repository. https://doi.org/10.5281/zenodo.4414861

[15] Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*.

[16] Smith, L. N., & Topin, N. (2019). Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, Proc. SPIE 11006.

[17] Efron, B., & Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*. Chapman & Hall.

[18] McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika, 12*(2), 153–157.

[19] Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics, 6*(2), 65–70.

[20] Kang, M., Zhu, J.-Y., Zhang, R., Park, J., Shechtman, E., Paris, S., & Park, T. (2023). Scaling up GANs for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 10124–10134).

[21] Song, Y., Sun, Z., & Yin, X. (2024). SDXS: Real-time one-step latent diffusion models with image conditions. *arXiv preprint* arXiv:2403.16627.