CLEAR: COMBINED FLICKER-BANDING AND MOIRÉ REMOVAL FOR SCREEN-CAPTURED IMAGES

Anonymous authors
Paper under double-blind review

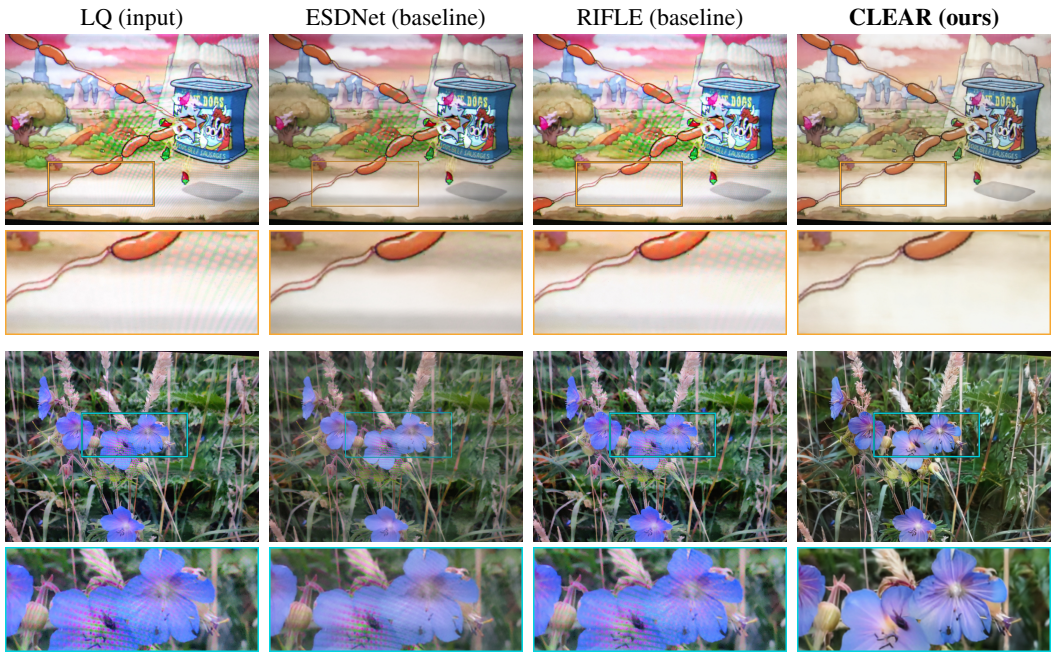


Figure 1: Joint removal of flicker-banding and moiré from screen-captured images. The inputs contain broad brightness bands and moiré patterns. ESDNet (Yu et al., 2022) and RIFLE (Zhu et al., 2025) leave visible interference, while CLEAR suppresses both artifacts and restores scene details.

ABSTRACT

Capturing display screens with mobile devices has become increasingly common, yet the resulting images often suffer from severe degradation caused by the coexistence of moiré patterns and flicker-banding, leading to significant visual quality impairment. Due to the strong coupling of these two artifacts in real imaging processes, existing methods designed for single degradations fail to generalize to such compound scenarios. In this paper, we present the first systematic study on joint removal of moiré patterns and flicker-banding in screen-captured images, and propose a unified restoration framework, named CLEAR. To support this task, we construct a large-scale dataset containing both moiré patterns and flicker-banding, and introduce an ISP-based flicker simulation pipeline to stabilize model training and expand the degradation distribution. Furthermore, we design a frequency-domain decomposition and recomposition module with a multi-level wavelet loss to enhance the modeling of compound artifacts. Comparisons with single-artifact restorers and their cascades reveal residual interference, motivating joint treatment of both degradations. Extensive experiments demonstrate that our method consistently outperforms existing image restoration approaches across multiple evaluation metrics, validating its effectiveness in complex real-world scenarios. Our code and dataset will be released.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



Figure 2: Examples of three degradation types in our MIRAGE dataset: flicker-banding, moiré and the combined one. All three types commonly occur when capturing display screens.

1 INTRODUCTION

With the rapid advancement of mobile imaging technology, smartphones have become the dominant tools for daily photography and video recording (Zhou et al., 2021). Despite improvements in sensors and image processing pipelines, mobile cameras still suffer from degradations (*e.g.*, moiré) in practical scenarios that impair image quality and usability.

One such scenario arises when capturing display screens, including computer monitors, tablets, and mobile devices. In these cases, captured images often exhibit structured dark band-like artifacts across the screen region, significantly degrading visual uniformity and obscuring screen content. Given the prevalence of screen-capturing in daily applications, understanding and addressing this degradation remains an important yet underexplored problem.

This phenomenon is commonly known as **flicker-banding**, which originates from the interaction between rolling shutter image acquisition and pulse width modulation (PWM)-based screen brightness control. Rolling shutter sensors expose the image sequentially, resulting in temporal offsets across sensor rows, while PWM controls the screen luminance by rapidly switching pixel intensities over time. When a rolling shutter camera captures a PWM-controlled screen, different sensor rows sample the screen at different temporal phases, leading to spatially structured luminance inconsistencies that manifest as banding artifacts across the screen region.

In screen-captured images, flicker-banding is often accompanied by another well-known degradation: moiré patterns. Moiré artifacts arise from spatial frequency aliasing between the display pixel grid and the camera sensor sampling lattice, typically appearing as repetitive textures or color distortions. While moiré removal has been extensively studied (Liu et al., 2020a; Yue et al., 2021a; Yu et al., 2022; Mei et al., 2025; Dong et al., 2025), flicker-banding has been relatively underexplored due to its temporal characteristics and the complexity of its formation mechanism. Recent works (Zhu et al., 2025) have explored learning-based approaches for removing banding only.

More importantly, in real-world scenarios, flicker-banding and moiré patterns frequently coexist and interact. However, experiments show that directly cascading flicker and moiré removal models fails when both degradations are present simultaneously. This is because the two artifacts are strongly coupled in both spatial and frequency domains. Flicker-banding alters the global luminance distributions, affecting moiré statistics, while high-frequency moiré structures interfere with banding detection and suppression. These observations highlight the need for a unified framework capable of jointly removing flicker-banding and moiré artifacts from the same captured image.

We identify the following key challenges. ***I. Lack of a suitable dataset.*** Existing datasets typically focus on a single type of degradation and lack samples containing both flicker-banding and moiré artifacts, making them unsuitable for joint restoration. ***II. Severe structural and perceptual degradation.*** Images affected by both degradations often exhibit extremely poor visual quality, with diverse

Table 1: Screen-capture datasets (scene: DCID/TMM22/UHDM natural/web/document; LCDMoiré text; RIFLE ads). MIRAGE uniquely spans both artifacts, six scenes, and LCD/LED/OLED.

Dataset	Size	Resolution	Scene	Degradation		Devices		Real-world
				Banding	Moiré	Camera	Display	
DCID Mei et al. (2025)	8,959	2133 × 1200	3	×	✓	3	LCD	✓
TMM22 Xu et al. (2024)	948	384 × 384	3	×	✓	4	/	✓
UHDM Yu et al. (2022)	5,000	4328 × 3248	3	×	✓	3	LCD	✓
LCDMoiré Zheng et al. (2020)	10,200	1024 × 1024	1	×	✓	/	LCD	×
RIFLE Zhu et al. (2025)	60	4096 × 3072	1	✓	×	1	LED	✓
MIRAGE	3,000	3706 × 3368	6	✓	✓	**5**	**LCD,LED,OLED**	✓

banding patterns, severe structural degradation, and color corruption, posing significant challenges to existing restoration models. ***III. Unstable training on real-world data.*** Due to the diversity of degradation patterns, directly training deep models on real captured data is highly unstable, frequently resulting in non-convergence or training collapse. This suggests that training solely on a real-world FB & moiré dataset is insufficient for reliable joint restoration.

To address these challenges, we construct a large-scale dataset: **M**oiré and fl**I**cke**R** in screen im**AGE** capture (**MIRAGE**), containing both moiré patterns and flicker-banding by capturing screen-displayed images under diverse devices, displays, and shooting conditions. Furthermore, we propose a restoration model named **C**ombined f**L**ick**E**r **A**nd moiré **R**emoval for Screen-Captured Images (**CLEAR**). To optimise the distribution of training data and expand the diversity of flicker-banding & moiré patterns, we introduce an ISP-based flicker-banding simulation pipeline. ISP means the image signal processor in the camera, and we utilize a pretrained Invertible-ISP (Xing et al., 2021) to approximate the transformation between RAW and sRGB domains. It models varied RAW-domain banding while preserving realistic moiré characteristics and stabilizing training.

As for the model design, we introduce a frequency-domain decomposition module that segments the degraded input into multiple frequency components. Since both flicker-banding and moiré artifacts have different but overlapping frequency support, we selectively attenuate artifact-dominated segments while preserving the remaining components. The recomposed representation serves as the input to the downstream model, leading to substantial performance improvements despite not fully removing the artifacts. Furthermore, we propose a multi-level wavelet loss to regularize the denoising process with orientation-aware constraints on artifact-related sub-bands. Together, these designs enable more stable and effective joint artifact removal in challenging screen captures.

Overall, our contributions are summarized as follows:

- We construct a large-scale dataset containing both moiré patterns and flicker-banding, enabling effective training and evaluation for joint artifact removal.
- We introduce an ISP-based flicker-banding simulation pipeline that stabilizes and expands degradation distributions, substantially enhancing model performance.
- We propose a frequency-domain segmentation module together with a multi-level wavelet loss, significantly improving restoration performance under compound degradations involving complex moiré patterns and flicker-banding in screen-captured images.
- Our experimental results demonstrate that the proposed model achieves clear advantages in handling compound flicker-banding and moiré scenarios, validating its strong practical value in real-world screen-capture applications and everyday use.

2 Related Work

Image Demoiréing. Early work on image demoiréing employed multi-branch architectures to address this problem (Sun et al., 2018). Subsequent methods have explored different signal domains: sRGB-based approaches (Xu et al., 2020; Liu et al., 2020b; Luo et al., 2020; Yue et al., 2021b), RAW-domain processing (Yang et al., 2025), and hybrid sRGB+RAW pipelines (Yue et al., 2023; Xu et al., 2024). From a methodological perspective, multiresolution networks (Zheng et al., 2022; Peng et al., 2024), cross-domain learning combining frequency and spatial representations (Luo et al., 2020; Xu et al., 2024), and explicit frequency-domain transforms (Zheng et al., 2022; Xu et al., 2020; Liu et al., 2025) have been widely adopted. To enhance detail recovery, multi-view fusion strategies leveraging wide-angle and ultra-wide-angle lenses (Mei et al., 2025) or focused and defocused image pairs (Dong et al., 2025) have also been proposed. Additionally, unsupervised (Yue et al., 2021a; Zhong et al., 2024) and training-free methods (Liu et al., 2020a) have been developed to reduce reliance on large-scale paired datasets for supervised demoiréing.

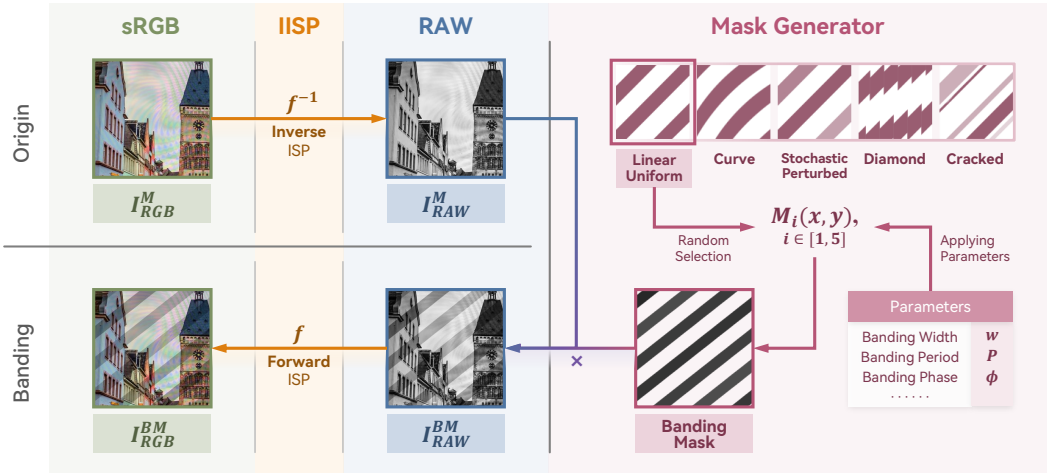


Figure 3: Overview of the simulation pipeline. Given an sRGB image with moiré, Inverse-ISP model is used to convert it to RAW format. Then, flicker-banding mask is generated and applied in the RAW domain to simulate banding artifacts. Finally, the ISP model converts the degraded RAW image back to RGB space, producing a training pair with both moiré and banding artifacts.

Image Debanding. Flicker-banding manifests as periodic luminance stripes in images captured from display screens. This artifact arises from the temporal mismatch between camera acquisition and display brightness modulation (Zhu et al., 2025). Modern smartphone cameras typically employ CMOS sensors with electronic rolling shutters (Durini, 2019), which expose sensor rows sequentially rather than simultaneously, introducing temporal offsets across the image. Meanwhile, displays modulate brightness through temporally varying mechanisms: OLED panels use pulse-width modulation (PWM) (Geffroy et al., 2006), LED displays adopt scanning refresh strategies. The interaction between these temporal dynamics produces spatially structured banding patterns. RIFLE (Zhu et al., 2025) represents the only learning-based method specifically targeting flicker-banding removal. However, its performance degrades when banding coexists with other artifacts, motivating unified approaches that address banding and moiré together.

3 Dataset: MIRAGE

To enable study of joint flicker-banding and moiré patterns removal, we construct a dataset featuring diverse degradation patterns. The data collection and capture procedures are described below.

3.1 Data Collection

We collect documents from DocBank (Li et al., 2020) and PubLayNet (Zhong et al., 2019), natural images from LSDIR (Li et al., 2023), cartoons from Danbooru2021, game images from VideoGameBunny (Taesiri & Bezemer, 2024), mobile UIs from Rico (Deka et al., 2017), and website screenshots from DSCVC (Wang et al., 2025) and screen-content-dataset (Safonov et al., 2025). We use IQA to exclude low-quality, overly dark, or noisy images, remove sensitive content to protect privacy, and rebalance sources to approximate the scene distribution of real captures.

3.2 Data Capture

To accurately reflect real-world conditions, we capture degraded images by photographing display screens with smartphone cameras. To account for variations in sensor characteristics across manufacturers, we employ five different smartphones: Vivo X200s, Xiaomi 17 Pro, Huawei Mate 50, Huawei Mate 60, and iPhone 17 Pro. To cover diverse display technologies, we use LCD, LED, and OLED screens. We systematically acquire images exhibiting flicker-banding only, moiré only, and both artifacts combined (LCD excluded, as they do not produce banding artifacts).

For each source image, we display it in full-screen mode on the target display and capture it using the smartphone camera under varying exposure settings, screen brightness levels, and viewing angles. Source images aligned to the photos serve as clean ground truth (GT).

Our dataset comprises 3,000 image pairs, with 1,000 pairs for each degradation type (flicker-banding only, moiré only, and combined artifacts). We randomly partition each subset into 930 training, 20 validation, and 50 test pairs. Due to different camera models, image resolutions vary (e.g., 4000×3000, 4096×3072, 3024×4032). A comparison with existing datasets is provided in Table 1. The key characteristics of our dataset are summarized as follows:

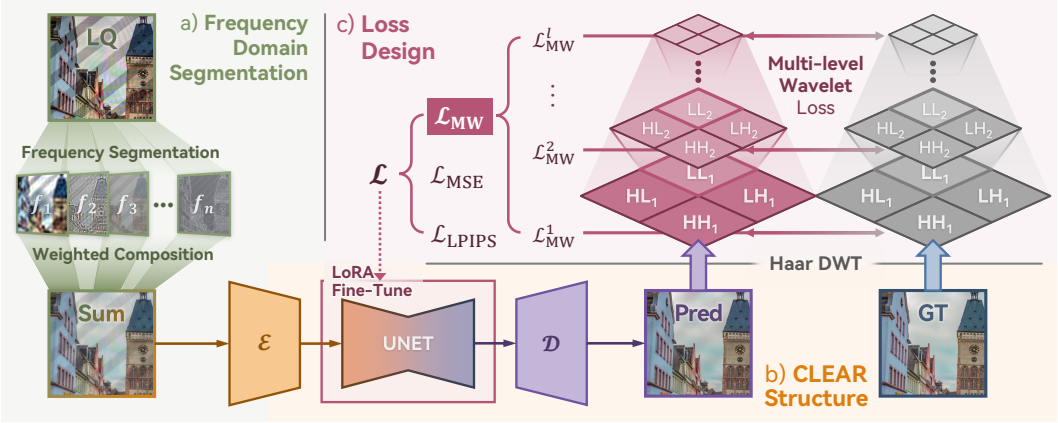


Figure 4: Overview of the proposed CLEAR framework. **(a)** Frequency-domain segmentation (FS) module. **(b)** The overall CLEAR architecture. **(c)** The CLEAR training objective combines multi-level wavelet loss (MWL) with perceptual and pixel losses.

- **Diverse scene content.** The dataset spans six scene categories: natural images, websites, documents, cartoons, video games, and mobile UI interfaces.
- **Comprehensive artifact coverage.** The dataset includes images with banding, moiré, and both artifacts combined, captured under varying viewing angles and distances to ensure diverse artifact morphologies across scenes and display types.
- **Real-world capture conditions.** Images are captured with consumer smartphones on commercial LCD, LED, and OLED displays to reflect practical screen photography.

4 METHODS

Building upon the PiSA-SR (Sun et al., 2025) backbone, we introduce the following components to enhance model performance in jointly removing flicker-banding and moiré artifacts.

4.1 ISP-BASED SIMULATION PIPELINE (ISPS)

While real-world data provides diverse compound degradations, we observe that certain banding patterns occur infrequently, which hinders robust model training. Moreover, directly training on real data with highly entangled artifacts often results in unstable convergence. To this end, we propose a physically-motivated RAW-domain simulation pipeline that synthesizes flicker-banding on moiré images, enabling controlled augmentation of underrepresented degradation types while preserving realistic artifact characteristics in the simulated captures.

RAW-Domain Simulation. A key insight of our approach is that flicker-banding (FB) artifacts originate from the temporal luminance modulation during camera sensor exposure, which is more accurately modeled in the linear RAW domain rather than the nonlinear sRGB space. Therefore, we adopt the Inverse-ISP model f^{-1} (Xing et al., 2021) to transform moiré-corrupted sRGB images I^{M}_{sRGB} from the UHDM dataset (Yu et al., 2022) into pseudo-RAW representations:

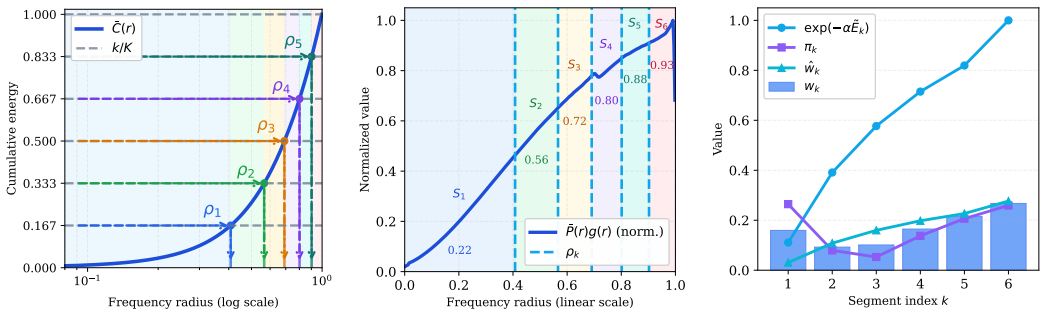
$$I^{M}_{RAW} = f^{-1}(I^{M}_{sRGB}), \tag{1}$$

enabling physically plausible FB synthesis of imaging pipeline.

Comprehensive Banding Pattern Coverage. Real-world flicker-banding exhibits diverse morphologies depending on the interaction between camera, display and shooting conditions. To ensure sufficient coverage, we design a parameterized mask generator $M(x, y)$ that produces five representative banding types, ranging from regular parallel stripes to curved, grid-like, fragmented, and stochastically perturbed structures, which could be further controlled by parameters including stripe width, period, phase, and feathering radius. This process effectively covers the vast majority of banding patterns encountered in everyday screen photography. Detailed mathematical formulations of each pattern type are provided in the supplementary material.

Gain-Based Degradation Model. The generated mask is converted into spatially-varying gain coefficients that model light intensity attenuation during exposure. A baseline darkness parameter D and per-stripe brightness jitter η_k are incorporated to simulate the stochastic nature of real banding:

$$G(x, y) = \max\left(G_{min},\ 1 - M(x, y) \cdot (1 - D) \cdot \eta_k\right), \tag{2}$$



(a) $\bar{C}(r)$ with ρ_k quantiles (b) Equal-energy segments and ρ_k (c) $\pi_k, \hat{w}_k \to w_k$

Figure 5: FS parameterization (K=6): (a) cumulative-energy quantiles locate boundaries; (b) equal-energy intervals yield ρ_k; (c) prior and frequency statistics initialize segment weights before training.

where G_{min} prevents complete pixel extinction in the simulated RAW image.

We obtain the banded RAW image by applying this spatial gain to each pixel:

$$I^{BM}_{RAW}(x, y) = I^{M}_{RAW}(x, y) \cdot G(x, y). \tag{3}$$

Final composite image is produced by forward ISP as $I^{BM}_{sRGB} = f(I^{BM}_{RAW})$, expanding effective training distribution and enabling model to generalize to rare but practically important FB patterns.

4.2 Frequency Domain Segmentation (FS)

Our FS module starts from a simple observation in the frequency domain: moiré usually appears as periodic mid/high-frequency interference, while flicker-banding is more concentrated in low or mid-frequency directional components. When these two artifacts of different degradations coexist, direct spatial restoration often forces one branch to simultaneously suppress periodic artifacts and recover global luminance consistency, which is prone to optimization conflict. We therefore precondition the input in frequency space before backbone restoration to suppress both artifacts.

Based on this observation, we introduce frequency segmentation and weighted recomposition as a front-end. Given input $\mathbf{x}$, we transform it to $\mathbf{X} = \mathcal{F}(\mathbf{x})$ and use normalized radius d to index frequency locations. Smooth Butterworth boundaries avoid ringing from hard cutoffs:

$$B_{\mathrm{lp}}(d; \rho, n) = \frac{1}{1 + (d/(\rho + \varepsilon))^{2n}}, \tag{4}$$

where ρ is the cutoff radius and n controls transition sharpness. Soft boundaries avoid the ringing introduced by binary frequency masks at sharp transitions.

Radial statistics determine the equal-energy frequency boundaries:

$$\bar{C}(d) = \frac{\int_0^d \bar{P}(r)\, g(r)\, dr}{\int_0^1 \bar{P}(r)\, g(r)\, dr}, \qquad q_k = \frac{k}{K}, \qquad \rho_k = \bar{C}^{-1}(q_k), \tag{5}$$

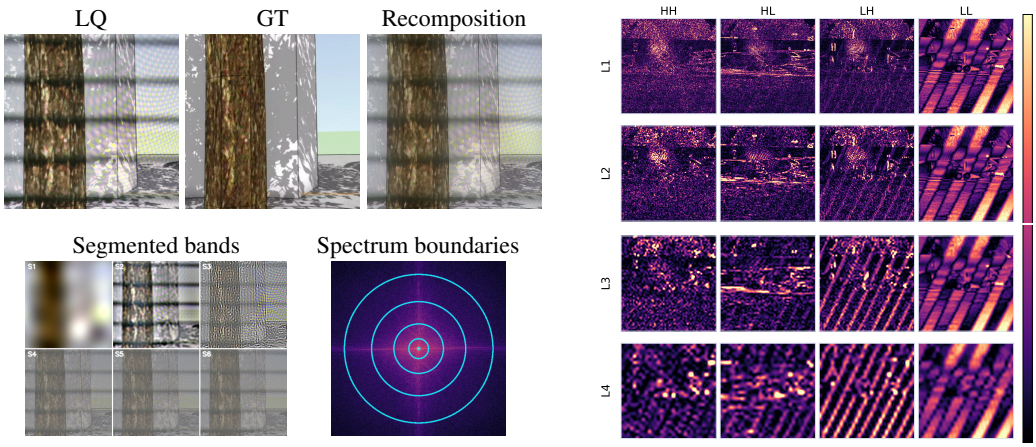
where $g(r)$ is a mild radial compensation term to reduce low-frequency dominance, K is the number of frequency segments, and q_k is the equal-energy target. Therefore, each ρ_k is obtained from a cumulative-energy quantile and serves as a split boundary.

With $\{\rho_k\}$, we build K smooth band masks $\{\mathbf{B}_k\}_{k=1}^{K}$:

$$\mathbf{B}_k = \begin{cases} B_{\mathrm{lp}}(d; \rho_1, n), & k = 1, \\ B_{\mathrm{lp}}(d; \rho_k, n) - B_{\mathrm{lp}}(d; \rho_{k-1}, n), & 2 \le k \le K - 1, \\ 1 - B_{\mathrm{lp}}(d; \rho_{K-1}, n), & k = K. \end{cases} \tag{6}$$

These masks approximate a partition of unity ($\sum_k \mathbf{B}_k \approx 1$). Each band contributes complementary spectral content. We then recombine all bands in frequency space and map back to image space:

$$\mathbf{I}_{\mathrm{sum}} = \Re\left\{\mathcal{F}^{-1}\left(\mathbf{F}_{\mathrm{sum}}\right)\right\}, \quad \mathbf{F}_{\mathrm{sum}} = \sum_{k=1}^{K} w_k \cdot (\mathbf{X} \odot \mathbf{B}_k), \tag{7}$$



(a) Frequency segmentation and weighted recomposition.

(b) Four-level (L1–L4) wavelet-loss maps.

Figure 6: Visualizations of the frequency segmentation module and multi-level wavelet supervision.

where $\{w_k\}$ are segment weights. Their initialization combines frequency statistics and task prior:

$$E_k = \mathrm{Mean}\Big(|\mathbf{X} \odot \mathbf{B}_k|^2\Big), \tag{8}$$

where $\tilde{E}_k$ is normalized energy and $\hat{w}_k$ is the recomposition weight,

$$\hat{w}_k = \frac{\exp(-\alpha \tilde{E}_k)}{\sum_j \exp(-\alpha \tilde{E}_j)}, \quad w_k^{(0)} = (1-\eta)\pi_k + \eta \hat{w}_k, \tag{9}$$

where π_k is the task prior (high keep at the lowest-frequency segment, lower keep at mid-low segments, compensatory keep at high-frequency segments), and η balances prior and statistics.

In Fig. 5a, cumulative-energy quantiles locate boundary candidates; Fig. 5b converts them to equal-energy intervals. Figure 5c shows how frequency statistics and task priors initialize recomposition.

Figure 6a shows that segmentation and weighted recomposition visibly reduce moiré and flicker-banding while retaining stable structures, giving downstream restoration a better-conditioned input.

4.3 Multi-Level Wavelet Loss (MWL)

Pixel and perceptual losses emphasize global similarity and can miss directional artifacts. We therefore use a multi-level wavelet loss: Haar decomposition isolates banding-like patterns in the directional HH/HL/LH branches for explicit penalties. For prediction $\hat{\mathbf{y}}$ and target $\mathbf{y}$, recursive Haar decomposition gives

$$\Big(\mathbf{LL}^l, \mathbf{LH}^l, \mathbf{HL}^l, \mathbf{HH}^l\Big) = \mathcal{W}\Big(\mathbf{LL}^{l-1}\Big), l = 1, \ldots, L, \tag{10}$$

where $\mathbf{LL}^0 = \hat{\mathbf{y}}$, $\mathbf{LL}_*^0 = \mathbf{y}$, $\mathcal{W}$ is one-step 2D Haar decomposition.

The resulting multi-level wavelet objective penalizes each sub-band:

$$\mathcal{L}_{\mathrm{MWL}} = \sum_{l=1}^{L} \sum_{b \in \{\mathrm{LL,LH,HL,HH}\}} w_b \, \mathcal{D}\big(\mathbf{B}_b^l, \mathbf{B}_{b,*}^l\big), \tag{11}$$

where $\mathcal{D}$ is L2 distance, w_b is the sub-band weight, and $\mathbf{B}_b^l$ denotes the b-th band at level l. In practice, this formulation improves optimization stability for mixed artifacts: LL constrains coarse luminance consistency, while HH/HL/LH emphasize directional discrepancy that is strongly correlated with flicker-banding and moiré textures. Figure 6b shows a 4-level map, with wavelet levels in rows and HH/HL/LH/LL sub-bands in columns for each level.

4.4 Overall Training Objective.

The overall objective is written in a compact form:

$$\mathcal{L}_{\mathrm{total}} = \lambda_{\mathrm{MWL}} \mathcal{L}_{\mathrm{MWL}} + \lambda_{\mathrm{rec}} (\mathcal{L}_{\mathrm{MSE}} + \lambda_{\mathrm{P}} \mathcal{L}_{\mathrm{LPIPS}}), \tag{12}$$

where $\mathcal{L}_{\mathrm{MSE}}$ is the pixel reconstruction loss, $\mathcal{L}_{\mathrm{LPIPS}}$ is the perceptual loss (Zhang et al., 2018), and $\lambda_{\mathrm{rec}}, \lambda_{\mathrm{P}}$ control their contributions to reconstruction.

Table 2: Sequential debanding and demoiréing on joint artifacts.

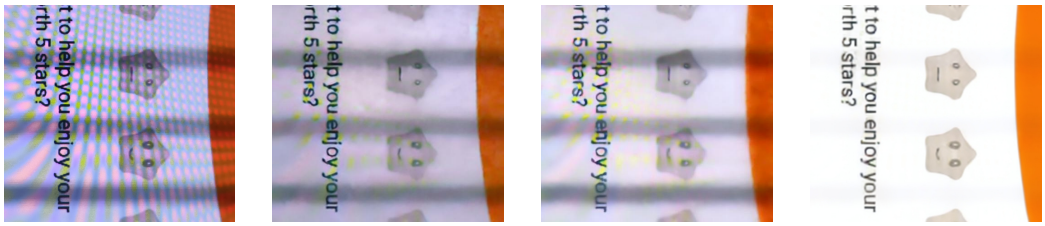
Method	ms-SSIM↑	LPIPS↓	DISTS↓
LQ	0.6969	0.4494	0.2619
$R \rightarrow E$	**0.7532**	**0.3834**	0.2574
$E \rightarrow R$	0.7528	0.3846	**0.2552**
CLEAR	**0.8158**	**0.2836**	**0.2011**

Table 3: Effect of simulated data on joint banding and moiré removal.

Training	ms-SSIM↑	LPIPS↓	DISTS↓
MIRAGE	0.7240	0.3556	0.2420
IISP	0.8028	0.3158	0.2233
Both	0.8219	0.3101	0.2211

Table 4: Effect of FS and MWL on joint banding and moiré removal.

Model	ms-SSIM↑	LPIPS↓	DISTS↓
Baseline	0.7909	0.3101	0.2211
MWL	0.7979	0.2901	0.2087
FS	0.8082	0.2958	0.2158
MWL + FS	0.8158	0.2836	0.2011



LQ $R \rightarrow E$ $E \rightarrow R$ CLEAR

Figure 7: Visual comparison with combined methods.

We adopt a parameter-efficient fine-tuning strategy using LoRA (Hu et al., 2022), which inserts low-rank adaptation matrices into the diffusion U-Net (Chen et al., 2026b;a). During training, only the LoRA parameters are optimized, while the encoder and decoder remain frozen.

5 Experiments

5.1 Experimental Setup

Data Construction. The primary dataset used for training and testing is our proposed MIRAGE dataset, which covers scenes in which flicker-banding and moiré coexist. For our proposed simulation pipeline, we use UHDM (Yu et al., 2022) for introducing the real-world moiré artifacts. Note that the MIRAGE testing images are cropped to 512×512 to amplify differences across evaluation metrics.

Evaluation Metrics. Reference-based metrics are used for evaluation, including SSIM (Wang et al., 2004a), ms-SSIM (Wang et al., 2004b), LPIPS (Zhang et al., 2018), DISTS (Ding et al., 2020), FSIM (Zhang et al., 2011), and GMSD (Xue et al., 2014). As mentioned in RIFLE (Zhu et al., 2025), non-reference metrics are not suitable for evaluation on flicker-banding.

Implementation Details. We set the rank to 32 with a learning rate of 5×10^{-5} for LoRA finetuning. The training process is performed using images of resolution 512×512.

Compared Methods. Lacking dedicated methods for simultaneous flicker-banding and moiré removal, we compare methods from related restoration tasks (*e.g.*, deraining, demoiréing, debanding, and super-resolution). All are retrained on our dataset for a fair comparison. The models include transformer-based, multi-step diffusion, and one-step diffusion architectures.

5.2 Comparison with Combined Models

We cascade RIFLE (Zhu et al., 2025) (debanding) and ESDNet (Yu et al., 2022) (demoiréing) in both orders on joint artifacts. Table 2 and Figure 7 compare them with CLEAR; $R \rightarrow E$ denotes RIFLE then ESDNet, and $E \rightarrow R$ the reverse. Neither cascade matches CLEAR: the first stage can alter signals needed by the second, and separate stages struggle with coupled degradations.

5.3 Main Results

Quantitative Results. On the cropped MIRAGE test set (Table 5), **CLEAR** leads on all six reported metrics. Its visible debanding gains exceed the changes in conventional scores, which measure overall distance to ground truth but are relatively insensitive to flicker-banding. Global image agreement can conceal local luminance stripes, so standard scores do not fully reflect the visual improvement.

Visual Comparison. In Fig. 8, CLEAR removes severe flicker-banding and suppresses moiré even around dense stripes and fine textures; competing methods leave residual artifacts or blur and distort structure. The advantage is clearest where periodic moiré overlaps luminance bands.

5.4 Ablation Study

ISP-based simulation consistently improves structural and perceptual quality over training on real MIRAGE alone (Table 3). Combining simulated and real data performs best: simulation broadens the range of banding patterns, while real pairs preserve capture effects that are difficult to model. FS and MWL each improve the baseline across structural and perceptual metrics (Table 4). FS conditions the input spectrum, while MWL penalizes directional residuals across scales. Their combination

Table 5: Results on the cropped MIRAGE test set. All methods are retrained on MIRAGE; red and blue mark the best and second-best scores for each metric.

Method	Task	SSIM↑	ms-SSIM↑	LPIPS↓	DISTS↓	FSIM↑	GMSD↓
LQ	-	0.5680	0.6969	0.4494	0.2619	0.7724	0.2086
NeRD-Rain	Deraining	0.6685	0.7254	0.5446	0.3130	0.7801	0.1723
ESDNet	Demoiré	**0.7307**	**0.7912**	0.4492	0.2721	0.8161	0.1709
MAT	Super-resolution	0.6845	0.7527	0.4680	0.2798	0.7962	0.1720
ResShift	Super-resolution	0.7099	0.7404	0.4548	0.2709	0.7912	0.1812
PiSA-SR	Super-resolution	0.6781	0.7240	**0.3556**	0.2420	0.7714	0.2090
InvSR	Super-resolution	0.6947	0.7126	0.4009	0.2800	0.7701	0.1998
RIFLE	Debanding	0.6816	0.7872	0.3745	**0.2371**	**0.8251**	**0.1679**
CLEAR	Debanding & Demoiré	**0.7600**	**0.8158**	**0.2836**	**0.2011**	**0.8352**	**0.1616**

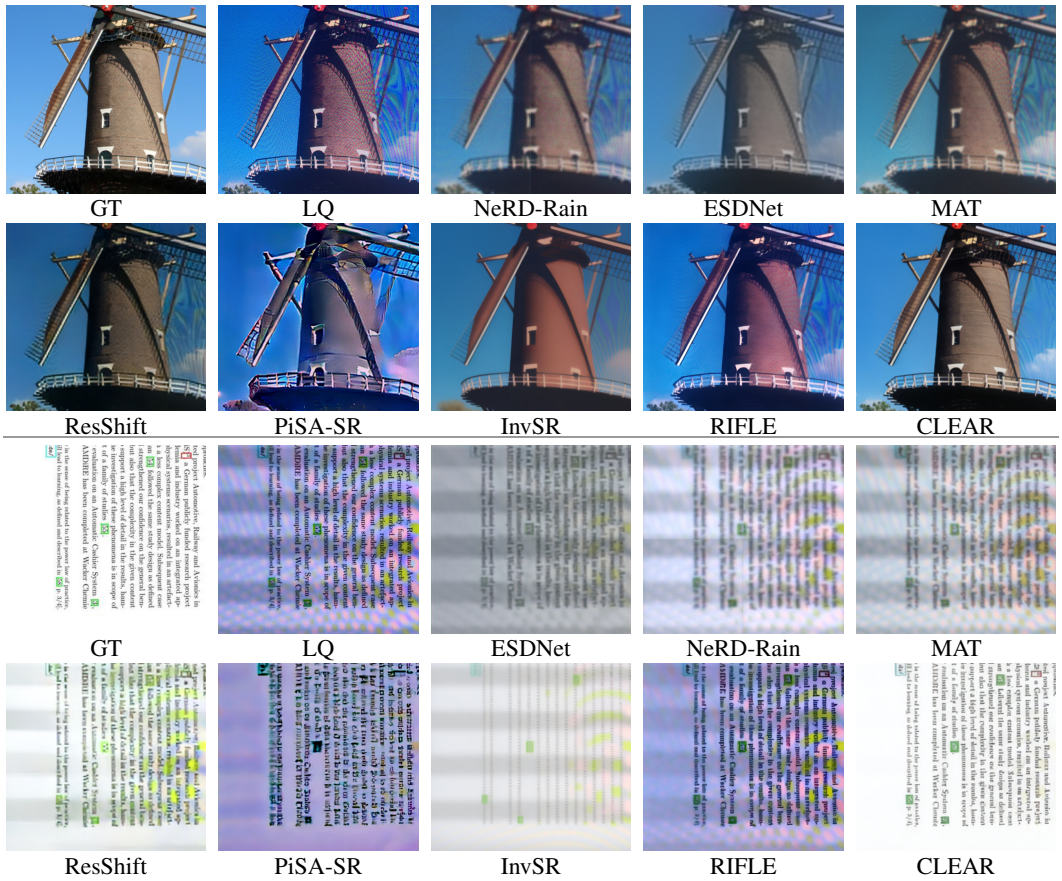


Figure 8: Joint banding and moiré removal on MIRAGE. LQ and GT denote degraded and clean images; comparison methods are retrained on MIRAGE.

performs best, indicating that the two components address complementary aspects of compound degradation. These complementary gains, together with the cascade results, support end-to-end treatment of compound degradation over separate single-artifact processing.

6 Conclusion

We study joint flicker-banding and moiré removal in screen captures, construct a real-world dataset, and propose CLEAR as a unified end-to-end model. Existing methods and debanding–demoiréing cascades underperform; CLEAR achieves stronger metrics and cleaner results, with fewer residual artifacts and better structural fidelity across varied scenes. These results show the value of treating both degradations together in practical screen-capture restoration.

REFERENCES

Tayebeh Azadmousavi and Ebrahim Ghafar-Zadeh. Complementary metal–oxide–semiconductor-based magnetic and optical sensors for life science applications. *Sensors*, 2024.

Zheng Chen, Bowen Chai, Rongjun Gao, Mingtao Nie, Xi Li, Bingnan Duan, Jianping Fang, Xiaohong Liu, Linghe Kong, and Yulun Zhang. Dvface: Spatio-temporal dual-prior diffusion for video face restoration. *arXiv preprint arXiv:2604.14560*, 2026a.

Zheng Chen, Ruofan Yang, Jin Han, Dehua Song, Zichen Zou, Chunming He, Yong Guo, and Yulun Zhang. Diffst: Spatiotemporal-aware diffusion for real-world space-time video super-resolution. *arXiv preprint arXiv:2605.13182*, 2026b.

Peng Dai, Xin Yu, Lan Ma, Baoheng Zhang, Jia Li, Wenbo Li, Jiajun Shen, and Xiaojuan Qi. Video demoireing with relation-based temporal consistency. In *CVPR*, 2022.

Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. Rico: A mobile app dataset for building data-driven design applications. In *UIST*, 2017.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. Image quality assessment: Unifying structure and texture similarity. *TPAMI*, 2020.

Xuan Dong, Xiangyuan Sun, Xia Wang, Jian Song, Ya Li, and Weixin Li. Video demoireing using focused-defocused dual-camera system. *TPAMI*, 2025.

Daniel Durini. *High performance silicon imaging: Fundamentals and applications of CMOS and CCD sensors*. Woodhead Publishing, 2019.

Bernard Geffroy, Philippe Le Roy, and Christophe Prat. Organic light-emitting diode (oled) technology: materials, devices and display technologies. In *PI*, 2006.

Bin He, Ce Wang, Boxin Shi, and Ling yu Duan. Fhde2net: Full high definition demoireing network. In *ECCV*, 2020.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. Docbank: A benchmark dataset for document layout analysis. *arXiv preprint arXiv:2006.01038*, 2020.

Yawei Li, Kai Zhang, Jingyun Liang, Jiezhang Cao, Ce Liu, Rui Gong, Yulun Zhang, Hao Tang, Yun Liu, Denis Demandolx, Rakesh Ranjan, Radu Timofte, and Luc Van Gool. Lsdir: A large scale dataset for image restoration. In *CVPRW*, 2023.

Lin Liu, Shanxin Yuan, Jianzhuang Liu, Liping Bao, Gregory Slabaugh, and Qi Tian. Self-adaptively learning to demoiré from focused and defocused image pairs. In *NeurIPS*, 2020a.

Shuai Liu, Chenghua Li, Nan Nan, Ziyao Zong, and Ruixia Song. Mmdm: Multi-frame and multi-scale for image demoiréing. In *CVPRW*, 2020b.

Zhongqi Liu, Bolun Zheng, Quan Chen, Qianyu Zhang, Xu Jia, Jiyong Zhang, and Chenggang Yan. Pyramid learnable bandpass filters for ultra-high-definition image demoiréing. *TCSVT*, 2025.

Xiaotong Luo, Jiangtao Zhang, Ming Hong, Yanyun Qu, Yuan Xie, and Cuihua Li. Deep wavelet network with domain adaptation for single image demoireing. In *CVPRW*, 2020.

Yanting Mei, Zhilu Zhang, Xiaohe Wu, and Wangmeng Zuo. Image demoiréing using dual camera fusion on mobile phones. In *ICME*, 2025.

Robert A. Meyers (ed.). *Encyclopedia of Physical Science and Technology*. Academic Press, third edition, 2001.

Yan-Tsung Peng, Chih-Hsiang Hou, You-Cheng Lee, Aiden J. Yoon, Zihao Chen, Yi-Ting Lin, and Wei-Cheng Lien. Image demoiréing via multiscale fusion networks with moiré data augmentation. *Sensors*, 2024.

Nikolai Safonov, Mikhail Rakhmanov, and Dmitriy Vatolin. Screen content video dataset and benchmark. In *ACMMM*, 2025.

Lingchen Sun, Rongyuan Wu, Zhiyuan Ma, Shuaizheng Liu, Qiaosi Yi, and Lei Zhang. Pixel-level and semantic-level adjustable super-resolution: A dual-lora approach. In *CVPR*, 2025.

Yujing Sun, Yizhou Yu, and Wenping Wang. Moiré photo restoration using multiresolution convolutional neural networks. *TIP*, 2018.

Mohammad Reza Taesiri and Cor-Paul Bezemer. Videogamebunny: Towards vision assistants for video games. *arXiv preprint arXiv:2407.15295*, 2024.

An Gia Vien, Hyunkook Park, and Chul Lee. Dual-domain deep convolutional neural networks for image demoireing. In *CVPRW*, 2020.

Feifeng Wang, Liquan Shen, Zhaoyi Tian, Shiwei Wang, Qi Teng, Yao Zhu, and Chengtao Zhou. Dscvc: Deep screen content video compression. *TCSVT*, 2025.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 2004a.

Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *ACSSC*, 2004b.

Yazhou Xing, Zian Qian, and Qifeng Chen. Invertible image signal processing. In *CVPR*, 2021.

Dejia Xu, Yihao Chu, and Qingyan Sun. Moiré pattern removal via attentive fractal network. In *CVPRW*, 2020.

Shuning Xu, Binbin Song, Xiangyu Chen, Xina Liu, and Jiantao Zhou. Image demoiréing in raw and srgb domains. In *ECCV*, 2024.

Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C. Bovik. Gradient magnitude similarity deviation: A highly efficient perceptual image quality index. *TIP*, 2014.

Qirui Yang, Fangpu Zhang, Yeying Jin, Qihua Cheng, Pengtao Jiang, Huanjing Yue, and Jingyu Yang. Dsdnet: Raw domain demoiréing via dual color-space synergy. In *ACMMM*, 2025.

Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Jiajun Shen, Jia Li, and Xiaojuan Qi. Towards efficient and scale-robust ultra-high-definition image demoiréing. In *ECCV*, 2022.

Huanjing Yue, Yijia Cheng, Fanglong Liu, and Jingyu Yang. Unsupervised moiré pattern removal for recaptured screen images. In *Neurocomputing*, 2021a.

Huanjing Yue, Yan Mao, Lipu Liang, Hongteng Xu, Chunping Hou, and Jingyu Yang. Recaptured screen image demoiréing. *TCSVT*, 2021b.

Huanjing Yue, Yijia Cheng, Yan Mao, Cong Cao, and Jingyu Yang. Recaptured screen image demoiréing in raw domain. *TMM*, 2023.

Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: a feature similarity index for image quality assessment. *TIP*, 2011.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

Bolun Zheng, Shanxin Yuan, Gregory Slabaugh, and Aleš Leonardis. Image demoireing with learnable bandpass filters. In *CVPR*, 2020.

Bolun Zheng, Shanxin Yuan, Chenggang Yan, Xiang Tian, Jiyong Zhang, Yaoqi Sun, Lin Liu, Aleš Leonardis, and Gregory Slabaugh. Learning frequency domain priors for image demoireing. *TPAMI*, 2022.

Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. Publaynet: largest dataset ever for document layout analysis. In *ICDAR*, 2019.

Yunshan Zhong, Yuyao Zhou, Yuxin Zhang, Fei Chao, and Rongrong Ji. Learning image demoiréing from unpaired real data. In *AAAI*, 2024.

Yuqian Zhou, David Ren, Neil Emerton, Sehoon Lim, and Timothy Large. Image restoration for under-display camera. In *CVPR*, 2021.

Libo Zhu, Zihan Zhou, Xiaoyang Liu, Weihang Zhang, Keyu Shi, Yifan Fu, and Yulun Zhang. Rifle: Removal of image flicker-banding via latent diffusion enhancement. In *NeurIPS*, 2025.

A APPENDIX

A.1 FORMATION MECHANISM

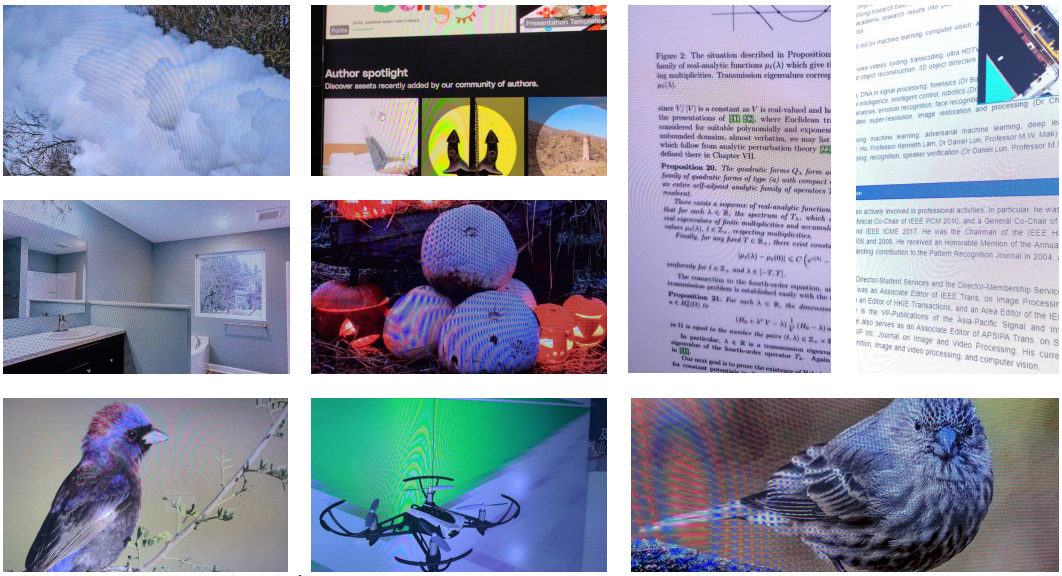


Figure 9: Example of moiréing patterns, such as mesh patterns, ripple patterns, color splotches etc.

A.1.1 FORMATION MECHANISM OF MOIRÉING

Moiréing artifacts result from the superposition of two or more periodic structures with similar spatial frequencies (Sun et al., 2018). This pattern occurs even when these similar structures overlap with each other with only a slight offset and the bigger the offset, the more obvious the moiréing patterns are. Moiréing patterns have various shapes, such as stripes, circles and so on (Vien et al., 2020). Even though the moiréing artifact is an optical phenomenon, this pattern is significant in the computer field.

Similar to the optical formation of moiréing artifacts, the digital moiréing artifacts appear when taking photos on the screen. This digital pattern results from the interference between two overlapping pixel matrices (He et al., 2020): the pixel grids of the display screens and the pixel grids of the cameras. When smartphone cameras take photos, the periodic pixel structure of the display screens interacts with that of the cameras. If these two kinds of grids don't match each other properly, due to the slight difference of camera angles, distance and frequency, moiréing artifacts appear.

In the frequency domain, moiréing artifacts usually cross a wide range of frequency and exhibit non-uniform and anisotropic spatial distribution. Different from common image noise, moiréing patterns don't distribute randomly but present structural characteristics and are often accompanied by color distortion (Dai et al., 2022). Due to the interaction between two structures, moiréing artifacts have an obvious local variation characteristic that within the same image, certain regions may appear as high-frequency, fine stripes, while other regions may show low-frequency, broad ripples. Some examples of moiréing artifacts in different shapes are shown in the Figure 9 below.

A.1.2 FORMATION MECHANISM OF FLICKER-BANDING

Flicker-banding shows as light and dark stripes in the photos, such as cracked banding, striped flickering, grid-like artifacts, curve banding and so on. When taking pictures, you may find an interesting fact that the shorter the exposure time is, the darker the stripes are. Changing the exposure time, you may get some different banding. There are some examples of flicker-banding in different shapes shown in the Figure 10 below. Flicker-banding appears due to the temporal inconsistency between cameras and display screens (Zhu et al., 2025). The following paragraphs will introduce the formation principle of flicker-banding from the perspective of both the smartphone cameras and display screens.

Firstly, from the capture perspective, smartphone cameras usually use the Complementary Metal-Oxide Semiconductor (CMOS) sensors with an electronic rolling shutter (Durini, 2019). CMOS consists of an integrated circuit design style depending on groups of p-channel (PMOS) and n-channel

(NMOS) metal-oxide-semiconductor field-effect transistors (MOSFETs) (Azadmousavi & Ghafar-Zadeh, 2024). CMOS sensors expose each line of pixels rather than expose the whole entity when taking photos, which means that each line of pixels on the cameras capture the lightness of the display screens not at the same time. As a result, time offset appears when capturing.

However, since the lightness of display screens also changes periodically, sampling at different points in time captures different phases of the screen's brightness waveform. Then from the perspective of display screens, display screens don't emit light continuously but shine with certain frequency. This is a potential conflict in time offset with the mechanism of the smartphone cameras. For instance, LCD screens apply precisely controlled voltage to change the alignment of liquid crystal modules. This change will in turn change the polarization state of the light from the backlight module (Meyers, 2001). Finally, this light passes the polarizers combined with the layer of the liquid crystal and the amount of light to pass this layer will be selectively blocked or allowed. This process causes the change of lightness and darkness on each pixel, which shows the images we can see on the screen.

Other technology of display matrix, such as the OLED display screens. The basic method for OLED is the Pulse-Width Modulation (PWM), which controls the lightness and darkness of each pixel with certain high frequency (Geffroy et al., 2006). Thus, by controlling the time of lightness in one cycle, the average lightness human beings can see is affected. Also, for the LED matrix display, it operates in a multiplexed scanning mode. This means that rather than illuminate all pixels at the same time, rows or columns of pixels are bright at the certain moment with certain frequency. Therefore, we can see that both the display method of screens and capture of the cameras aren't continuous. So we can get the conclusion that flicker-banding occurs when camera and display refresh rates are misaligned.



Figure 10: Example of banding patterns, such as jagged bands, striped flickering etc.

A.2 Simulation Pipeline

In this part, more details about the formulas to add banding to the raw images with moiréing artifacts will be introduced. In this pipeline, five types of banding artifacts are added: linear uniform banding, diamond banding, curve banding, cracked banding and stochastic perturbed banding. Some examples of each type are shown in Figure 11.

To process tilted patterns in images, the first step of adding any of these banding is to change the coordinate system. Suppose the width and height of the image is W and H, the original coordinate of the pixel is $(x, y)(x \in [0, W], y \in [0, H])$ and the new coordinate after putting the center of images as the origin of the new system is (x_c, y_c). The relation between the new coordinate and the original one is shown below:

$$x_c = x - \frac{W-1}{2}, \qquad y_c = y - \frac{H-1}{2}. \tag{13}$$

Based on the new coordinate, a local coordinate system that rotates with the object is constructed, which is called 2D rotation transformation mathematically. Suppose the inclination angle of the stripes relative to the vertical direction as θ and the coordinate after rotation is (u, v). The formula for this rotation is shown below:

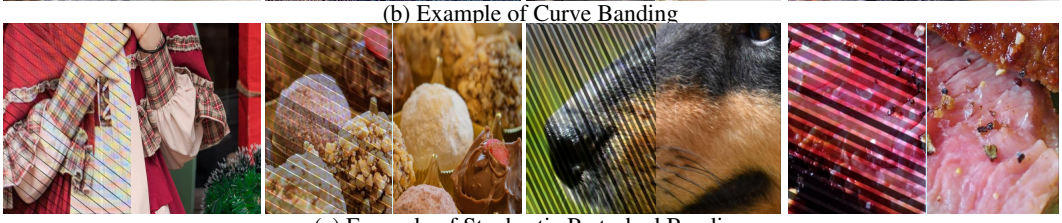
$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_c \\ y_c \end{bmatrix}, \tag{14}$$



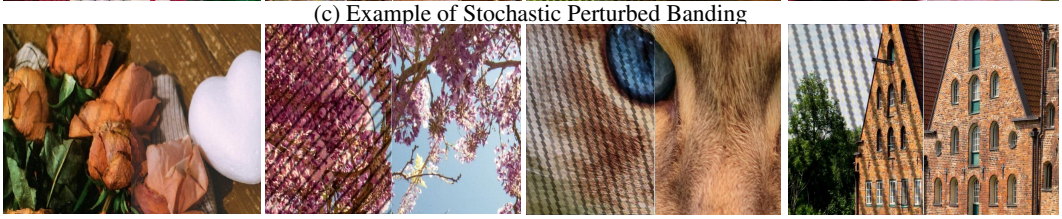
(a) Example of Linear Uniform Banding



(b) Example of Curve Banding



(c) Example of Stochastic Perturbed Banding



(d) Example of Diamond Banding



(e) Example of Cracked Banding

Figure 11: Example of results from the simulation pipeline

To avoid setting the pixel too dark, we introduce a minimum gain G_{min}, which is set to be 0.02. Set the smooth masking function, which is used to make the edges of banding smoother, as $M(x, y)$, the baseline intensity factor as D, which is selected randomly in our pipeline, and the random noise factor of each stripe as η_k. The gain applied to every pixel $G(x, y)$ can be defined as:

$$G(x, y) = \max\left(G_{min},\ 1 - M(x, y) \cdot (1 - D) \cdot \eta_k\right). \tag{15}$$

In our pipeline, we use the following equation to calculate the output $I_{out}(x, y)$ from the input image $I_{in}(x, y)$:

$$I_{out}(x, y) = I_{in}(x, y) \cdot G(x, y). \tag{16}$$

The following paragraphs will introduce the corresponding formulas for the variables mentioned above of each type.

A.2.1 Linear Uniform Banding

To generate stripes with period P (period is the result of the addition of stripe width and stripe gap) and phase ϕ, the stripe index k of the current pixel and its signed distance $d(x, y)$ relative to the center of that stripe need to be calculated. Suppose the width of stripes w. The formula to calculate the stripe index k is shown below:

$$k = \text{round}(\frac{v - \phi}{P}). \tag{17}$$

From the above equation, the ideal stripe center position corresponding to this index v_c can be written as:

$$v_c = k \cdot P + \phi. \tag{18}$$

The equation below clarifies the current pixel's signed distance $d(x, y)$ relative to the center of the stripe k:

$$d(x, y) = |v(x, y) - v_c| - \frac{w}{2}. \tag{19}$$

If $d(x, y) < 0$, this means that the current pixel is in the stripe k and vice versa.

Then to avoid aliasing effect, the Smoothstep function is applied to create a feathered edge. Suppose the feather radius f and the radius after normalization t. The formula about t is shown as below:

$$t = \text{clamp}(\frac{d(x, y) + f}{2f}, 0, 1). \tag{20}$$

Then the Smoothstep function $S(x, y)$ can be written as $S(x, y) = t^2(3 - 2t)$. Furthermore, the smooth masking function $M(x, y)$ is defined, representing the occupancy of the banding:

$$M(x, y) = 1 - S(x, y). \tag{21}$$

Here, $M(x, y) \in [0, 1]$, where 1 means the center of the banding, the darkest and 0 is the background. Bringing all the variables back to the equation 16, we can get the formula for adding the linear uniform banding.

A.2.2 Curve Banding

Based on the new coordinate (u, v) obtained in the first step, we introduce a spatial modulation term, an offset function $C(u)$ dependent on u, to realize the effect of curve banding. The effective traverse coordinate $v'(u, v)$ can be written as:

$$v'(u, v) = v - C(u). \tag{22}$$

In our pipeline, we realize two kinds of curve modes: quadratic mode and sinusoidal mode. The quadratic mode simulates geometric distortion resembling a bow shape. Let A as the bending amplitude and L as the normalized size. The middle point of pixels in the whole images on the u axis is expressed as u_{mid}. The offset function of quadratic mode $C_{quad}(u)$ is shown below:

$$C_{quad}(u) = \pm A \cdot \left(\frac{u - u_{mid}}{L} \right)^2. \tag{23}$$

The second mode, the sinusoidal mode, is designed to simulate the wavy interference pattern. Though in our simulation pipeline we don't apply this mode to produce images, but our program can realize this effect. Let λ be the wave length and ϕ be the initial phase of the curve. The sinusoidal mode $C_{sin}(u)$ can be expressed in the equation below:

$$C_{sin}(u) = A \cdot \sin \left(\frac{2\pi u}{\lambda} + \varphi \right). \tag{24}$$

Similar to the stripe index k calculated in the subsection A.2.1, suppose the phase of the whole stripe Φ_{stripe}, the stripe index k can be written as:

$$k = \text{round} \left(\frac{v' - \phi_{stripe}}{P} \right). \tag{25}$$

Since the center of the stripe is no longer a constant, but a curve $C(u) + const$, the distance from the pixel to the center of the bending stripe $d(x, y)$ is written as:

$$d(x, y) = |v' - (k \cdot P + \phi_{stripe})| - \frac{w}{2}. \tag{26}$$

Also, using the smoothstep function same as the subsection A.2.1, we can get the smooth masking function $M(x, y)$:

$$M(x, y) = 1 - S \left(\text{clamp} \left(\frac{d + f}{2f}, 0, 1 \right) \right). \tag{27}$$

Finally, with all the variables calculated above, the output image can be expressed with the formula 16.

A.2.3 STOCHASTIC PERTURBED BANDING

Stochastic perturbed banding is supposed to have random position, width, shape and edges. Based on the model of curve banding, we add the process of discrete inter-stripe variations and continuous intra-stripe modulation.

Firstly, to realize discrete inter-stripe variations, we introduce two random variables: position jitter Δp_k and width jitter Δw_k, both of which follow uniform distribution. Suppose the spacing and width jitter amplitude respectively as J_{pos} and J_{width} and the original base width as $\bar{w}$. So we can get the modified center of the stripe v_c and base width w_k for the ith stripe:

$$\begin{cases} v_c(k) = k \cdot P + \phi + \Delta p_k \\ w_k = \bar{w} + \Delta w_k \end{cases}, \tag{28}$$

$$\text{where } \Delta p_k \sim \mathcal{U}(-J_{\text{pos}}, J_{\text{pos}}), \quad \Delta w_k \sim \mathcal{U}(-J_{\text{width}}, J_{\text{width}}).$$

Secondly, to simulate the slight jitter when using the rolling shutter of smartphone cameras, we introduce the continuous low-frequency modulation. Suppose the width of the stripe at the u position is $w_l(u,k)$ and the wiggle amplitude as A_{wiggle}. A smooth noise function $\Omega(u)$ is defined:

$$w_l(u,k) = w_k + A_{wiggle} \cdot \Omega(u). \tag{29}$$

Then we introduce the independent edge roughness to simulate the fact that edges of stripes may not be smooth. Noise $\zeta_{top}(u)$ and $\zeta_{bot}(u)$are added respectively to the top edge of stripes E_{top} and the bottom edge E_{bot}:

$$\begin{cases} E_{top} = v_c(k) + \frac{w_l(u,k)}{2} + \sigma_{edge} \cdot \zeta_{top}(u) \\ E_{bot} = v_c(k) - \frac{w_l(u,k)}{2} + \sigma_{edge} \cdot \zeta_{bot}(u) \end{cases}. \tag{30}$$

The minimum distance $d_{complex}$ from the pixel v' to the edges are defined as:

$$d_{complex}(x,y) = -\min\left(E_{top} - v',\ v' - E_{bot}\right). \tag{31}$$

Replacing the d in the equation 27 with the $d_{complex}$, we can get the smooth masking function $M(x,y)$ for the stochastic perturbed banding. Taking these variables back into the equation 16, we can finally get the formula for adding the stochastic perturbed banding.

A.2.4 DIAMOND BANDING

Diamond banding is defined as a 2D pattern rather than a linear uniform stripe, which is L1 norm mathematically. Rather than only consider the v axis, both the stripe index k_u and k_v is considered on the u and v axis. Suppose the period on the u and v axis respectively as P_u and P_v and the center of the u and v axis as u_c and v_c. It's easy to get $k_v = \text{round}\left(\frac{v-\phi}{P_v}\right)$ and $k_u = \text{floor}\left(\frac{u}{P_u}\right)$. The equation of the center of the u and v axis as u_c and v_c can be obtained:

$$\begin{cases} v_c = k_v \cdot P_v + \phi \\ u_c = k_u \cdot P_u + \frac{P_u}{2} \end{cases}. \tag{32}$$

It' easy to get the coordinate relative to the center of the diamond (δ_u, δ_v):$\delta_u = u - u_c$ and $\delta_v = v - v_c$. Suppose w as the biggest width of the stripes and α as the shear slope, which is determined by the ratio of the height and width of the diamond. So we can get the boundary condition for a pixel to lie inside a diamond:

$$|\delta_v| + \alpha \cdot |\delta_u| \leq \frac{w}{2}. \tag{33}$$

From the formula above, we can define the distance between the pixel and the boundary of the diamond $d_{diamond}$:

$$d_{diamond} = (|\delta_v| + \alpha \cdot |\delta_u|) - \frac{w}{2}. \tag{34}$$

If $d_{diamond} < 0$, the pixel is in the diamond and vice versa.

Similar to the model mentioned in the subsection A.2.1, the smooth masking function $M(x,y)$ can be written as:

$$M(x,y) = 1 - S\left(\text{clamp}\left(\frac{d_{diamond} + f}{2f}, 0, 1\right)\right). \tag{35}$$

With all these variables mentioned above, the eqaution 16 can be written into the form of producing the diamond banding.

A.2.5 CRACKED BANDING

The cracked banding is a kind of pattern with the central area continuous and small cracks on the sides. Firstly, we need to define the body of the banding. Set the center keep ratio as $R_{keep} \in (0, 1)$ and the half width of the central area as $w_{core} = \frac{w}{2} \cdot R_{keep}$. Suppose the center of the banding as v_c and feathered smoothstep function as $S(\cdot)$. So the smooth masking function of the core area $M_{core}(x, y)$ can be defined as:

$$M_{core} = 1 - S(|v - v_c| - w_{core}). \tag{36}$$

Secondly, on both sides of the center, we generate N stripes of small parallel crack patterns. For the ith stripe, the distance from the crack center to the stripe main center v_c is set as δ_i, the basic width as $\bar{w}_i$, the dither intensity as γ and the dynamic width as $w_i(u)$. To simulate the natural irregularity, random noise $J(u)$ is introduced and the equation can be written as below:

$$w_i(u) = \bar{w}_i \cdot (1 + \gamma \cdot J(u)). \tag{37}$$

Then the distance between the pixel and the center of the ith crack can be written as:

$$d_i(x, y) = |v - (v_c + \delta_i)| - \frac{w_i(u)}{2}. \tag{38}$$

Moreover, to create a discontinuous crack effect, longitudinal breaks are added. We introduce a binary break mask $B_i(u)$:

$$B_i(u) = \begin{cases} 0, & \text{if } u \in \text{Break Regions} \\ 1, & \text{otherwise} \end{cases}. \tag{39}$$

From the equation above, we can get the smoothing mask function $M_{crack,i}(x, y)$ for the ith crack:

$$M_{crack,i}(x, y) = M_{raw,i}(x, y) \cdot B_i(u). \tag{40}$$

Finally, the smoothing mask function M_{final} of the whole crack banding can be written as:

$$M_{final} = \max\left(M_{core}, \max_{i=1}^{N}\{M_{crack,i}\}\right). \tag{41}$$

With all the variables above, it's easy to get the formula for producing cracked banding from the equation 16.

A.3 ADDITIONAL VISUAL COMPARISON

In this subsection, additional visual comparison is presented with flicker-banding and moiré images(LQ), clean images(GT) and other compared methods on the **MIRAGE** testing dataset in Figs. 12–13.

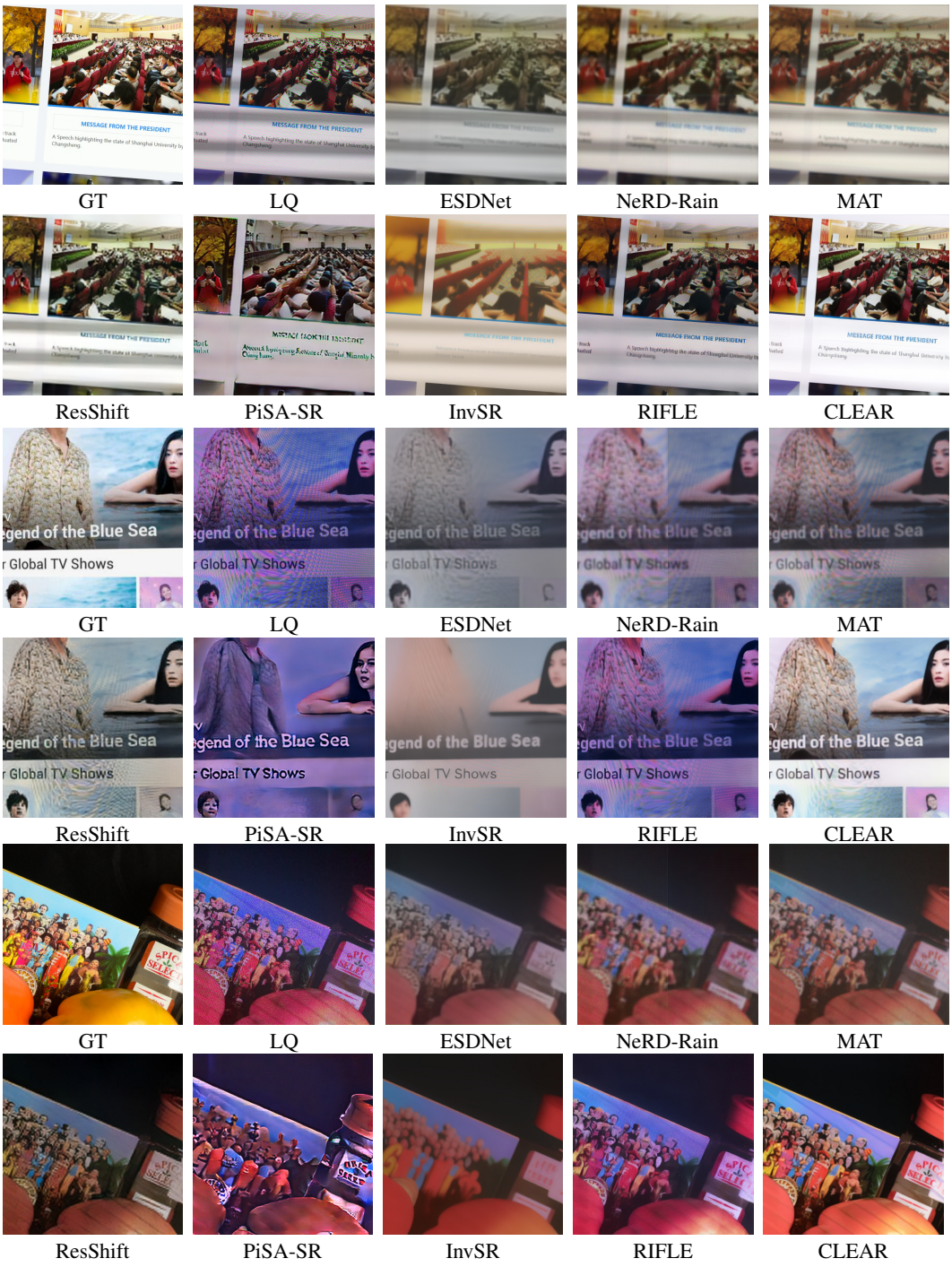


Figure 12: Visual comparison with flicker-banding & moiré images (LQ), clean images (GT), and other compared methods on the MIRAGE testing dataset. Compared methods are retrained with MIRAGE's training dataset. CLEAR gains great advantages over other methods.

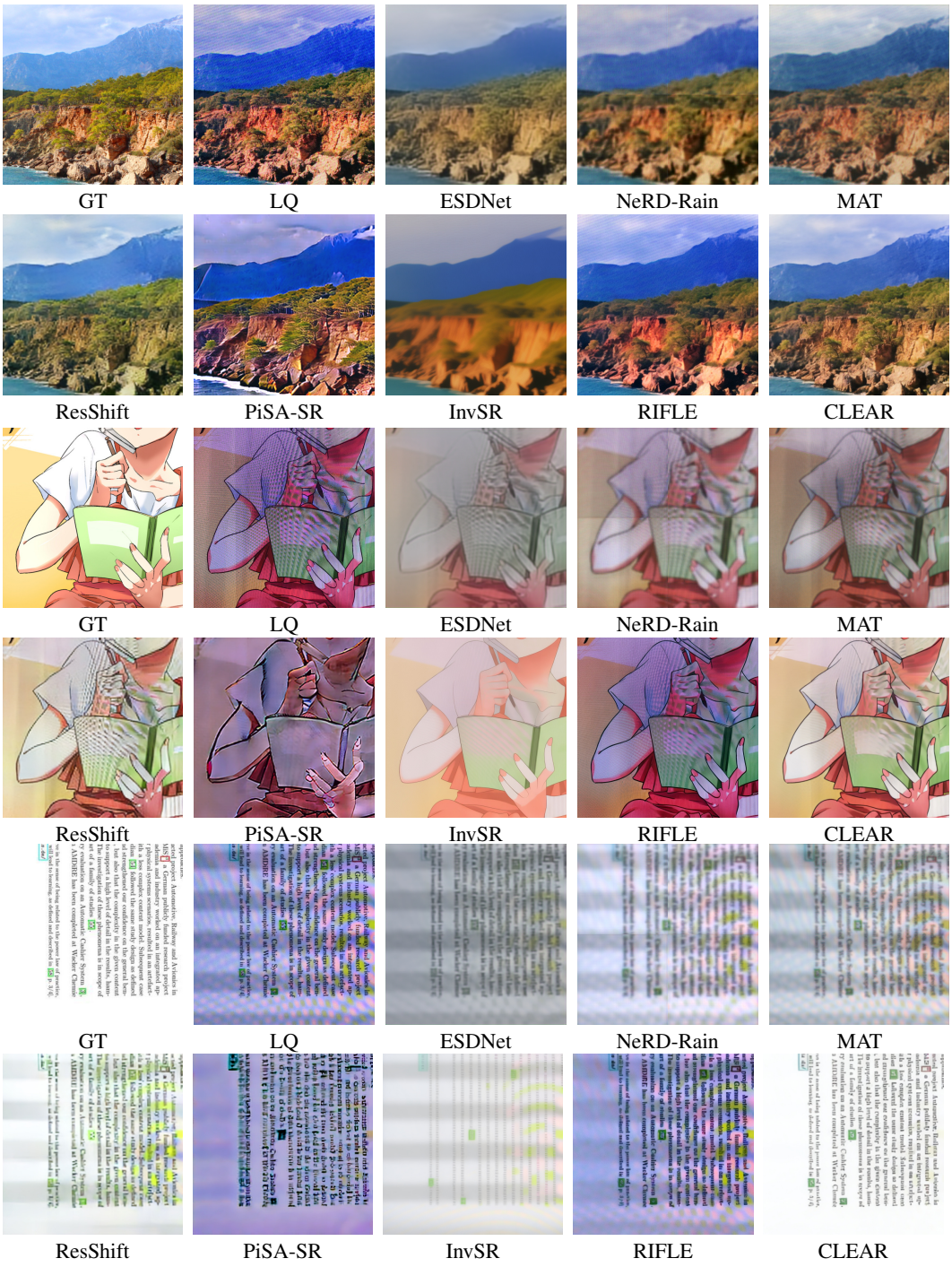


Figure 13: Visual comparison with flicker-banding & moiré images (LQ), clean images (GT), and other compared methods on the MIRAGE testing dataset. Compared methods are retrained with MIRAGE's training dataset. CLEAR gains great advantages over other methods.

A.4 RESULTS FOR DEBANDING

This subsection will show results of debanding with CLEAR. Visual comparison between CLEAR, flicker-banding images (LQ), clean images (GT) and RIFLE on the MIRAGE testing dataset in Figure 14. We can see that CLEAR has a better performance in image debanding.

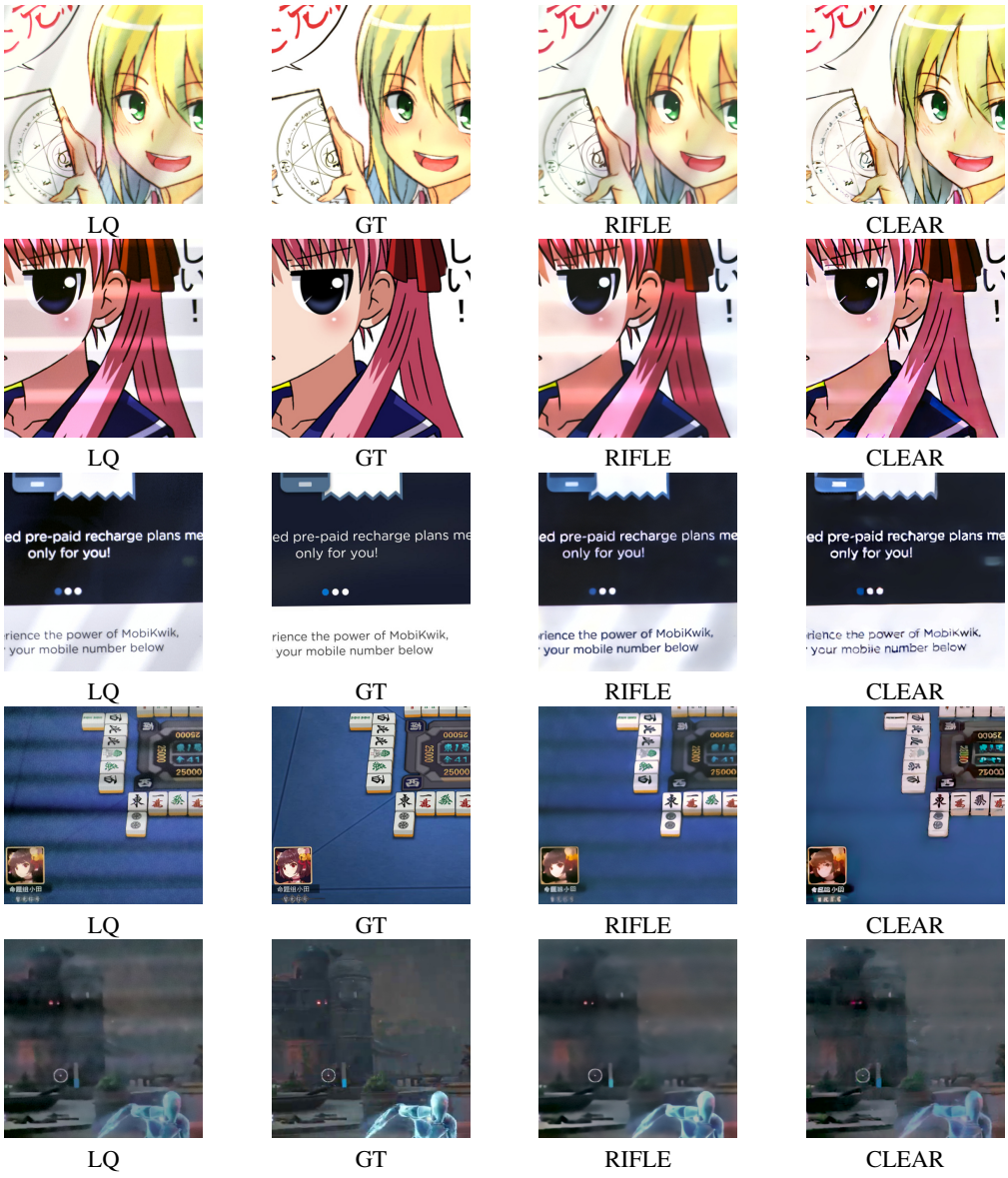


Figure 14: Visual comparison between CLEAR, flicker-banding images (LQ), clean images (GT) and RIFLE on the MIRAGE testing dataset.

A.5 RESULTS FOR DEMOIREING

This subsection will represent comparison between CLEAR, moire images (LQ), clean images (GT) and ESDNet on the MIRAGE testing dataset as shown in Figure 15. We can see that CLEAR performs better in image demoireing.

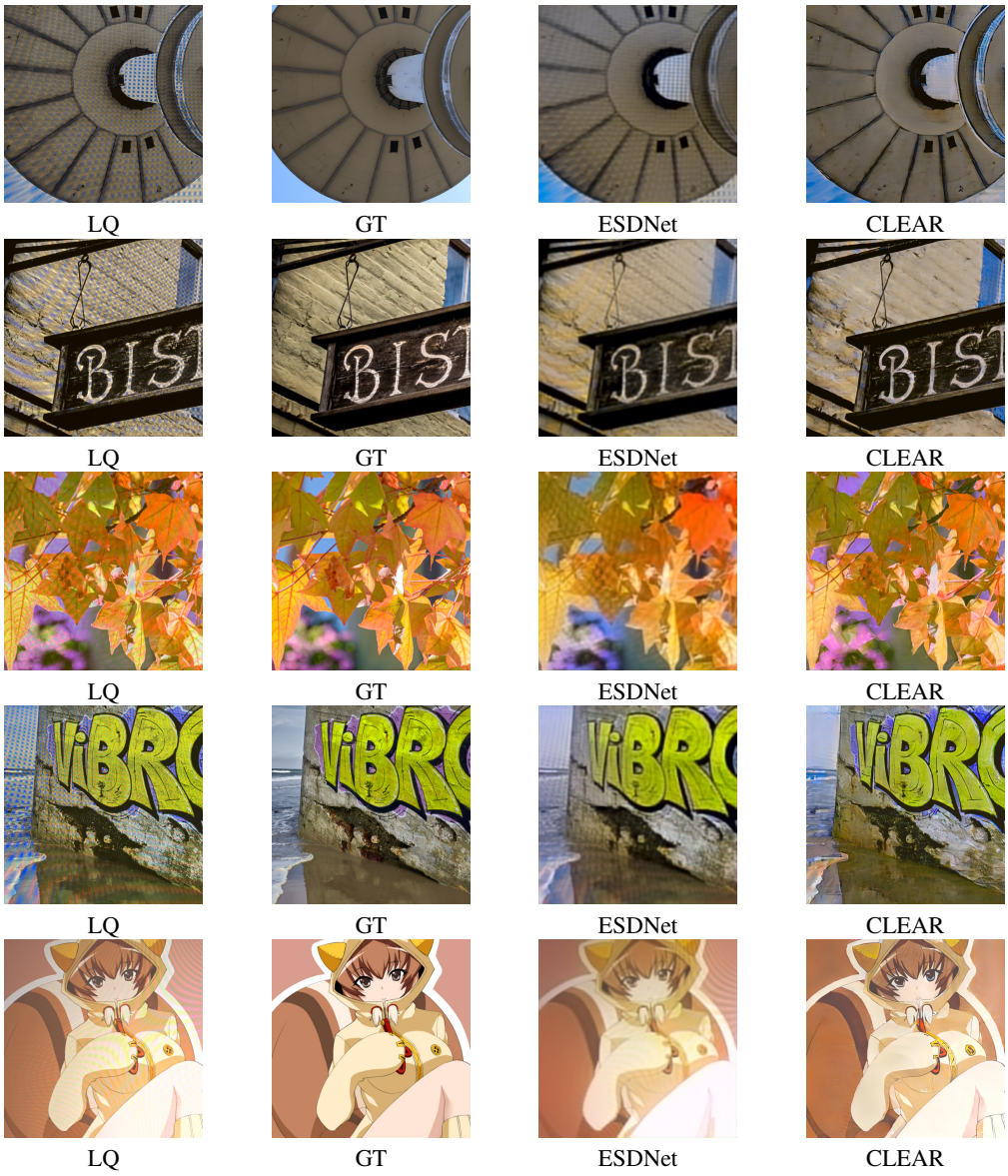


Figure 15: Visual comparison between CLEAR, moire images (LQ), clean images (GT) and ESDNet on the MIRAGE testing dataset.