

NTIRE 2025 Challenge on RAW Image Restoration and Super-Resolution

Marcos V. Conde ^{*†} Radu Timofte ^{*} Zihao Lu ^{*} Xiangyu Kong Xiaoxia Xing
 Fan Wang Suejin Han MinKyu Park Tianyu Zhang Xin Luo Yeda Chen
 Dong Liu Li Pang Yuhang Yang Hongzhong Wang Xiangyong Cao
 Ruixuan Jiang Senyan Xu Siyuan Jiang Xueyang Fu Zheng-Jun Zha
 Tianyu Hao Yuhong He Ruoqi Li Yueqi Yang Xiang Yu Guanlan Hong
 Minmin Yi Yuanjia Chen Liwen Zhang Zijie Jin Cheng Li Lian Liu
 Wei Song Heng Sun Yubo Wang Jinghua Wang Jiajie Lu Watchara Ruangsang

Abstract

This paper reviews the NTIRE 2025 RAW Image Restoration and Super-Resolution Challenge, highlighting the proposed solutions and results. New methods for RAW Restoration and Super-Resolution could be essential in modern Image Signal Processing (ISP) pipelines, however, this problem is not as explored as in the RGB domain. The goal of this challenge is two fold, (i) restore RAW images with blur and noise degradations, (ii) upscale RAW Bayer images by 2x, considering unknown noise and blur. In the challenge, a total of 230 participants registered, and 45 submitted results during the challenge period. This report presents the current state-of-the-art in RAW Restoration.

1. Introduction

RAW Image Restoration has become an active topic in the computational photography field for enhancing image quality. This work focuses on three related sub-tasks: super-resolution, denoising, and deblurring. Each of these addresses image quality bottlenecks in portable camera devices: insufficient pixel density, excessive sensor noise, and motion blur due to long exposure times.

Although modern portable devices now feature image sensors with higher bit depths approaching those of professional-grade cameras, they remain constrained by physical size, power consumption, and thermal limitations. As a result, larger sensors and higher-quality optical components are impossible to be deployed in a limited space.

^{*} Marcos V. Conde ([†] corresponding author, project lead), Zihao Lu, and Radu Timofte are the challenge organizers, while the other authors participated in the challenge and survey.

^{*} University of Würzburg, CAIDAS & IFI, Computer Vision Lab.
 NTIRE 2025 webpage: <https://cvlai.net/ntire/2025>.
 Code: <https://github.com/mv-lab/AISP>

Smaller sensor areas lead to reduced light collection per pixel and lower signal-to-noise ratios (SNR). Optical resolution and distortion resistance are also limited, making it difficult to achieve both high resolution and low noise simultaneously with great sharpness. These inherent limitations underline the irreplaceable role of image restoration tasks in portable imaging systems.

At the data level, RAW images are processed with only minimal operation, primarily linear amplification and white balance which preserving a nearly linear response to the sensor signal. In contrast, images processed through the ISP (Image Signal Processing) pipeline experience a series of strongly nonlinear operations such as demosaicing, tone mapping, gamma correction, and color adjustment [17, 18, 28, 29, 68]. These operations introduce irreversible information loss and amplify quantization errors and noise textures [2, 34], constraining the performance for tasks like denoising and super-resolution when conducted in the sRGB domain. Although some recent works have demonstrated great performance on RAW image reconstruction from sRGB domain [50, 67], reconstructed RAW image still have distance from the original RAW images, which means the reconstruction model couldn't restore the original detail pattern from the sRGB image, original RAW images still have more detail information for the model to learn.

Furthermore, due to the lack of a ISP-tuning standard and varying stylistic preferences across camera manufacturers, sRGB-format images from the same sensor can differ significantly more than their RAW images. This increases the difficulty in achieving generalization and robustness for cross-sensor model when relying on sRGB inputs.

At the same time, portable devices also suffer from limited computational resources. Therefore, model size and computational complexity must be prioritized factor for model design.

Thus, in this work we are presenting the solutions submitted for the NTIRE 2025 RAW Image Restoration and Super-Resolution Challenge. We are providing information regarding the challenge setup, with the task description and the challenge data properties characterizing the challenge dataset splits. We are also listing information regarding the challenge participants, with their teams and affiliations.

In Sec. 2 and Sec. 3 we describe the challenge tracks, datasets and evaluations. In Sec. 4 and Sec. 5 we provide detailed descriptions of the best solutions.

Related Computer Vision Challenges This challenge is one of the NTIRE 2025 Workshop associated challenges on: ambient lighting normalization [63], reflection removal in the wild [70], shadow removal [62], event-based image deblurring [59], image denoising [60], XGC quality assessment [44], UGC video enhancement [56], night photography rendering [21], image super-resolution (x4) [12], real-world face restoration [13], efficient super-resolution [52], HR depth estimation [73], efficient burst HDR and restoration [36], cross-domain few-shot object detection [22], short-form UGC video quality assessment and enhancement [39, 40], text to image generation model quality assessment [25], day and night raindrop removal for dual-focused images [38], video quality assessment for video conferencing [30], low light image enhancement [45], light field super-resolution [66], restore any image model (RAIM) in the wild [42], raw restoration and super-resolution [15], and raw reconstruction from RGB on smartphones [16].

Acknowledgments

This work was partially supported by the Humboldt Foundation. We thank the NTIRE 2025 sponsors: ByteDance, Meituan, Kuaishou, and University of Wurzburg (Computer Vision Lab).

2. NTIRE 2025 RAWSR Challenge

2.1. Dataset

The challenge dataset is based on BSRRAW [18] and the NTIRE 2024 RAWSR Challenge [19]. Following previous work [18, 68, 69], we use images from the Adobe MIT5K dataset [3], which includes images from multiple Canon and Nikon DSLR cameras.

The images are manually filtered to ensure diversity and natural properties (*i.e.* remove extremely dark or overexposed images), we also remove the blurry images (*i.e.* we only consider in-focus sharp images with low ISO).

The **pre-processing** is as follows: (i) we normalize all RAW images depending on their black level and bit-depth.

<https://www.cvlai.net/ntire/2025/>

Method	Track	PSNR	SSIM	# Par.
SMFFRaw-S 4.3	Efficient	42.12	0.9433	0.18
RawRTSR 4.1	Efficient	41.74	0.9417	0.19
NAFBN 4.5	Efficient	40.67	0.9347	0.19
MambaIRv2 4.6	Efficient	40.32	0.9396	0.19
RepRAW-SR-Tiny 4.7	Efficient	40.01	0.9297	0.02
RepRAW-SR-Large 4.7	Efficient	40.56	0.9339	0.09
ECAN 4.8	Efficient	39.13	0.9057	0.09
<hr/>				
USTC 4.2	General	42.70	0.9479	1.94
SMFFRaw-S 4.3	General	42.60	0.9467	1.99
RawRTSR-L 4.1	General	42.58	0.9475	0.26
ERBSFormer 4.4	General	42.45	0.9448	3.30
ER-NAFNet 5.5	General	41.17	0.9348	-
<hr/>				
RBSFormer [33]	2024	43.649	0.987	3.3
BSRAW [18]	2024	42.853	0.986	1.5
Bicubic [19]	2024	36.038	0.952	-

Table 1. **NTIRE 2025 RAWSR Results.** We provide **PSNR/SSIM** results on the complete testing set (200 images) [19]. Efficient methods are constrained to maximum 200K parameters. As a reference, we provide the baselines from the NTIRE 2024 RAWSR Challenge [19]. All the fidelity metrics are calculated in the RAW domain. We also report parameters (Par.) in millions.

(ii) we convert (“pack”) the images into the well-known RGGB Bayer pattern (4-channels), which allows to apply the transformations and degradations without damaging the original color pattern information [43].

Training: We provide the participants 1064 1024 × 1024 × 4 clean high-resolution (HR) RAW images. The LR degraded images can be generated on-line during training using the degradation pipeline proposed in BSRRAW [18].

Such degradation pipeline considers different noise profiles, multiple blur kernels (PSFs) and a simple downsampling strategy to synthesize low-resolution (LR) RAW images [20]. The participants can apply other augmentation techniques or expand the degradation pipeline to generate more realistic training data.

2.2. Baselines

We use BSRRAW [18] and previous years top-performing methods as references. The top performing challenge solutions improve the baseline performance, however, the neural networks are notably more complex in terms of design and computation.

2.3. Results

We use three testing splits: (i) Validation, 40 1024px images using during the model development phase. (ii) Test 1MP, 200 images of 1024px resolution. (iii) The same 200 test images at full-resolution ≈ 12MP. The participants process the corresponding LR RAW images (*e.g.* 512 × 512 × 4), and submit their results. Thus, the participants do not have access to the ground-truth images.

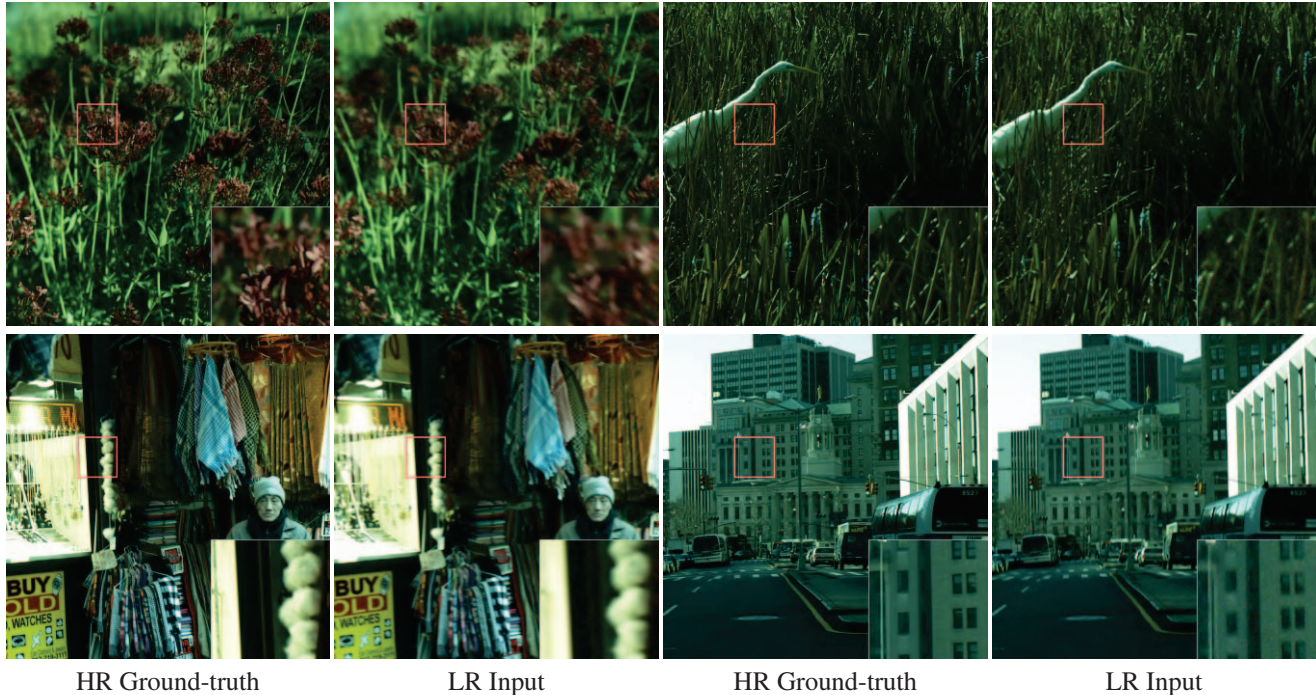


Figure 1. Samples of the NTIRE 2025 RAW Image Super-Resolution Challenge testing set.

We provide samples of the testing set in Fig. 1.

In Tab. 1 we provide the challenge benchmark. Besides fidelity metrics such as PSNR and SSIM, we also provide relevant implementation details of each method. The methods can greatly improve the RAW images quality and resolution, even in the case of full-resolution 12MP images as output. All the proposed methods are able to increase the resolution and details of the RAW images while reducing blurriness and noise. Moreover, there are not detectable color artifacts. We can conclude that (synthetic) RAW image super-resolution can be solved similarly to RAW denoising. However, more realistic downsampling remains an open challenge.

3. NTIRE 2025 RAW Image Restoration (RAWIR) Challenge

3.1. Dataset

The RAW Image Restoration (RAWIR) challenge dataset uses images from diverse sensors including: Samsung Galaxy S9 and iPhone X (RAW2RAW [1]). Additionally, we collected images from Google Pixel 10, Vivo X90, and Samsung S21 to enrich the diversity of camera sensors and build a complete testing set. The dataset covers various scenes (indoor and outdoor), lighting conditions (day and night), and subjects. All the images were manually filtered to ensure high quality, sharpness and clear details *i.e.* the images were captured under low ISO (≤ 400), in-focus

(without notable defocus or motion blur) and with proper exposure. The original RAW files were saved in DNG format, unprocessed by the smartphones' ISPs.

We applied the following **pre-processing** pipeline to the RAW images:

- All images were normalized based on the camera black-level and bit depth (*e.g.* 10, 12, 14 bits per pixel).
- The images were converted to the RGGB Bayer pattern (4-channels).
- We crop the images into non-overlapping (packed) patches of dimension $512 \times 512 \times 4$.

We provide 2139 clean patches for **training** models. The participants use the baseline degradation pipeline [18] to simulate realistic degradations. As part of the challenge, participants develop their own degradation pipelines to simulate more realistic blur and noise. The core components in the degradation pipeline includes different noise profiles, and multiple blur kernels (PSFs).

3.2. Baselines

We use PMRID [65] as the main baseline model. We also include NAFNet [7] into our baseline as a representative of UNet-like models. Both methods are popular efficient image denoising methods.

3.3. Challenge Results

The synthetic test dataset is generated by applying our degradation pipeline at different levels. The three degrada-

Method	Type	Test Level 1	Test Level 2	Test Level 3	# Params. (M)	# MACs (G)
Test Images		0.953 / 39.56	0.931 / 35.30	0.907 / 33.03		
PMRID []	Baseline	0.982 / 42.41	0.965 / 38.43	0.951 / 35.97	1.032	1.21
NAFNET [7]	Baseline	0.983 / 43.50	0.972 / 39.70	0.962 / 37.49	1.130	3.99
MOFA [9]	Reference	0.982 / 42.54	0.966 / 38.71	0.974 / 36.33	0.971	1.14
Samsung AI 5.1	Efficient	0.991 / 45.10	0.980 / 40.82	0.971 / 38.46	0.19	10.98
LMPR-Net 5.4	Efficient	0.989 / 42.57	0.973 / 39.17	0.961 / 37.26	0.19	2.63
Samsung AI 5.1	General	0.993 / 46.04	0.985 / 42.25	0.978 / 40.10	4.97	23.79
Miers 5.2	General	0.993 / 45.72	0.983 / 41.73	0.974 / 39.50	4.76	#N/A
Multi-PromptIR 5.3	General	0.986 / 44.80	0.978 / 41.38	0.968 / 38.96	39.92	158.24
ER-NAFNet 5.5	General	0.992 / 45.10	0.972 / 39.32	0.953 / 36.13	4.57	#N/A

Table 2. **NTIRE 2025 RAW Image Restoration Results.** We provide the **SSIM/PSNR** results on the testing set (140 images) at different degradation levels. All the metrics are calculated in the RAW domain. We also provide the efficiency information about the models. All the best performance results are highlighted in bold.

tion level settings are as follows: (i) $y = x + n$, the degradation is only noise. (ii) $y = (x * k) + n$, the degradation is noise and/or blur, with 0.3 probability of blur and 0.5 of real noise. (iii) $y = (x * k) + n$, all the images have realistic blur and noise.

In the Tab. 2, we provide the challenge benchmark and quantitative evaluation results of the challenge participants. The challenge covers a wide range of solutions: efficient (under 200K parameters), and general solutions without computational constraints. The team Samsung AI Camera wins the general and efficient track.

In Fig. 20, we provide qualitative results of the challenge methods. For the level 1 test set, most solutions showcase great performance by increasing the sharpness and reducing noise completely. For the levels 2 and 3, the solutions from team Samsung AI and Miers show the best results, while the output from the WIRTeam and ChickentRun show less effective blur removal. In general, since denoising and deblurring might require opposite operations, the models struggle to tackle both at the same time, specially blur.

In Section 4 we describe the top solutions for **RAW Super-Resolution**.

In the Section 5 we describe the best solutions for **RAW restoration**.

Note that the method descriptions were provided by each team as their contribution to this report.

4. RAW Image Super-Resolution Methods

4.1. RawRTSR: Raw Real-Time Super Resolution

Team Samsung AI

Xiaoxia Xing¹, Fan Wang¹, Suejin Han², MinKyu Park²

¹ Samsung R&D Institute China - Beijing (SRC-B)

² The Department of Camera Innovation Group, Samsung Electronics

Contact: xx.xing@samsung.com

Method Description In the RAW Super Resolution Challenge, we designed our model structure based mainly on CASR [71]. To meet the parameter and inference time requirements, we adopted the strategy of distillation and parameterization, as shown in Fig. 2. We employ X-Restormer [10] as the teacher model, with two architecturally distinct student models trained through knowledge distillation from this shared teacher. Specifically, during the training phase, the student model incorporates re-parameterized convolution blocks. For deployment, these multi-branch convolutional components are structurally converted into a unified convolution layer through parameter fusion, yielding the final submitted model. The specifications of the submitted model are detailed in Table 3

Model Framework The Raw Super Resolution task encompasses denoising, super-resolution, and blur quality degradation. We decompose these three objectives into two fundamental processes: denoising and detail enhancement. Both two networks employ a straight-through architecture integrating both denoising and detail enhancement modules.

- Efficient model, RawRTSR: As shown in Fig. 2, includes

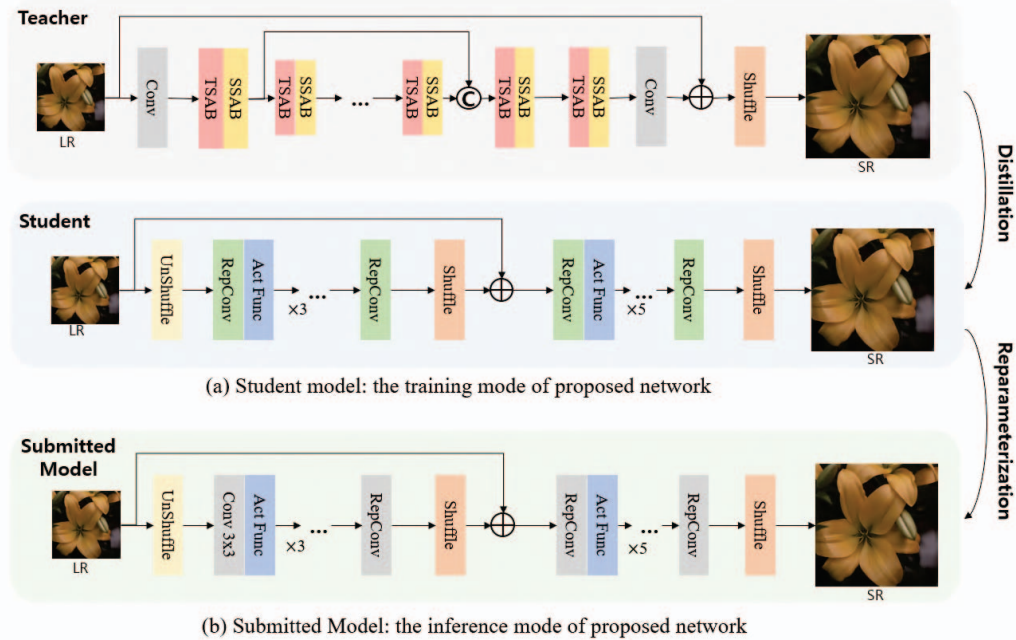


Figure 2. The overall structure of the RawRTSR network, including 0.188M parameters and running at 3.64 ms on the A100 GPU.

Type	Model	Input	Training Time	Train E2E	Extra Data	# Params. (M)	Inference Time	GPU
Efficient Model	RawRTSR 2	$512 \times 512 \times 4$	24h	Yes	No	0.19	4.45 ms	A100
General Model	RawRTSR-L 3	$512 \times 512 \times 4$	24h	Yes	No	0.26	4.44 ms	A100

Table 3. Training and Testing details of Samsung AI solution.



Figure 3. The overall structure of the RawRTSR-L network: 0.311M parameters and running at 4.44 ms on the A100 GPU.

denoising module and detail enhancement module. Denoising Module first reduces image resolution via unPixelshuffle downsampling to effectively capture global information critical for noise removal. It processes features through four convolutions for denoising, followed by resolution restoration to match the original input size through upsampling. Detail Enhancement Module employs five convolutions to recover fine textures. To prevent excessive detail loss during denoising, we explicitly incorporate residual connections from the original input. The final output is upscaled through pixel shuffle operations to achieve super-resolution reconstruction. The network maintains a maximum feature channel number of 48 throughout both modules to balance performance and complexity.

- General Model, RawRTSR-L: As shown in Fig. 3, differ-

ent from the RawRTSR, we increased the number of feature channels from 48 to 64 to enhance representational capacity. However, to prevent potential information redundancy during the denoising stage resulting from this channel expansion, we incorporated a channel attention mechanism to adaptively recalibrate feature responses.

Implementation Details We use two synthetic degradation methods to obtain low quality (LQ) images:

- Randomly add noise and blur multiple times at RAW domain.
- Convert the RAW image to RGB to add motion blur and noise and then back to RAW.

We trained our model in three steps. All steps use the PyTorch framework and A100 GPU.

- First, we train the teacher and student models separately. The LQ patches were cropped from synthetically degraded LQ images with 256×256 sizes. AdamW [47] optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 0.0001) was used with a learning rate of 0.0005. The total number of epochs was set at 800. We use the l1 loss.
- In the second step, the model was initialized with the weights trained in the first step, and feature distillation was used. The total number of epochs was set at 800. In

this step, the initial learning rate was set to 0.00005 and we use L2 loss.

- In the third stage, the model was initialized using the weights trained in the previous step. The LR patches were cropped from synthetically degraded LQ images with 512×512 sizes.

In order to ensure that the model parameters and inference time meet the requirements, the final submitted model is the student model after reparameterization.

4.2. Streamlined Transformer Network for Real-Time Raw Image Super Resolution

Team USTC-VIDAR

Tianyu Zhang¹Equal contribution., Xin Luo¹, Yeda Chen²,
Dong Liu¹

¹ University of Science and Technology of China, Hefei,
China

² Shanghai Shuangshen Information Technology Co., Ltd.

Contact: zhangtianyu@mail.ustc.edu.cn

Method Description The overall framework of our method is illustrated in Fig. 4. It is a streamlined version of RBSFormer [33], designed for efficient processing. The main branch consists of a 3×3 convolution, N cascaded transformer blocks, and an upsample block, while the residual branch contains only an upsample block. Each upsample block employs a 3×3 convolution followed by a PixelShuffle operation [58] to upscale features by a factor of 2.

The computational complexity of transformers primarily arises from self-attention modules and feed-forward networks, where feature maps are expanded for queries (Q), keys (K), and values (V) or projected for MLP selection. To mitigate these inefficiencies, we incorporate InceptionNeXt [72], which leverages partial convolution and depth-wise convolution for efficient spatial feature extraction during Q, K, and V projection. For the feed-forward networks, we adopt ShuffleNet [78] with G channel groups to reduce input projection parameters while maintaining cross-channel communication. Inspired by [33, 48], we further streamline the output projection using element-wise multiplication with a depth-wise convolution gate. We set $N = 8$ and $G = 4$.

Our method is trained on the NTIRE 2025 RAW Image Super Resolution Challenge dataset, with the degradation pipeline following [18]. For a 4-channel, 1MP RGGB RAW image, our model requires 519.72 GFLOPS and contains 1.94M parameters, aligning with the challenge’s constraints (0.2M–2M parameters). On an NVIDIA RTX 3090, a forward pass for a full-resolution image takes 96 ms.

Implementation Details The model is trained exclusively on the dataset provided by the challenge organizers, which consists of over 1,000 RAW images captured by various DSLR camera sensors. To enhance diversity, the dataset is augmented using random horizontal flips, vertical flips, and transpositions. Degraded images are simulated using the BSRRAW degradation pipeline with additional PSF kernels [26].

The training process consists of two stages. In the first stage, the model is trained for 300k steps using a batch size of 8 and a patch size of 192. The learning rate is decayed from 2×10^{-4} to 10^{-6} , and training takes approximately 12 hours on NVIDIA RTX 3090 GPUs. In the second stage, the batch size is increased to 64, and the patch size is set to 256. The model is trained for 147k steps with the learning rate decaying from 10^{-4} to 10^{-6} , requiring around 31 hours on A800 GPUs.

For both stages, we use the Adam optimizer with default hyperparameters. The loss function is a combination of Charbonnier loss and a Frequency loss [49], with the latter assigned a weight of 0.5.

4.3. SMFFRaw: Simplified Multi-Level Feature Fusion Network for RAW Image Super-Resolution

Team XJTU

Li Pang^{1,2*}, Yuhang Yang^{1,2*}, Hongzhong Wang^{1,2*},
Xiangyong Cao^{1,2†}

¹ School of Computer Science and Technology, Xi’an
Jiaotong University, Xi’an, 710049, China.

² Ministry of Education Key Laboratory of Intelligent
Networks and Network Security

Contact: caoxiangyong@mail.xjtu.edu.cn

While existing RAW image super-resolution methods have achieved remarkable restoration quality, their reliance on over-parameterized architectures leads to computational overhead, rendering them impractical for real-time processing. To bridge this gap, we propose a simplified multi-level feature fusion network for RAW image super-resolution (SMFFRaw), a computationally efficient network based on MFSSR [41]. By employing a novel iterative training strategy which gradually enhances the overall performance of the model, our method achieves 42.208/42.628 dB PSNR on the validation set with only 0.182M/1.994M parameters.

Method Description The specific network architecture of SMFFRaw is shown in Figure 5, which consists of three main components including shallow feature extraction, deep feature extraction and reconstruction. More details are provided in the following.

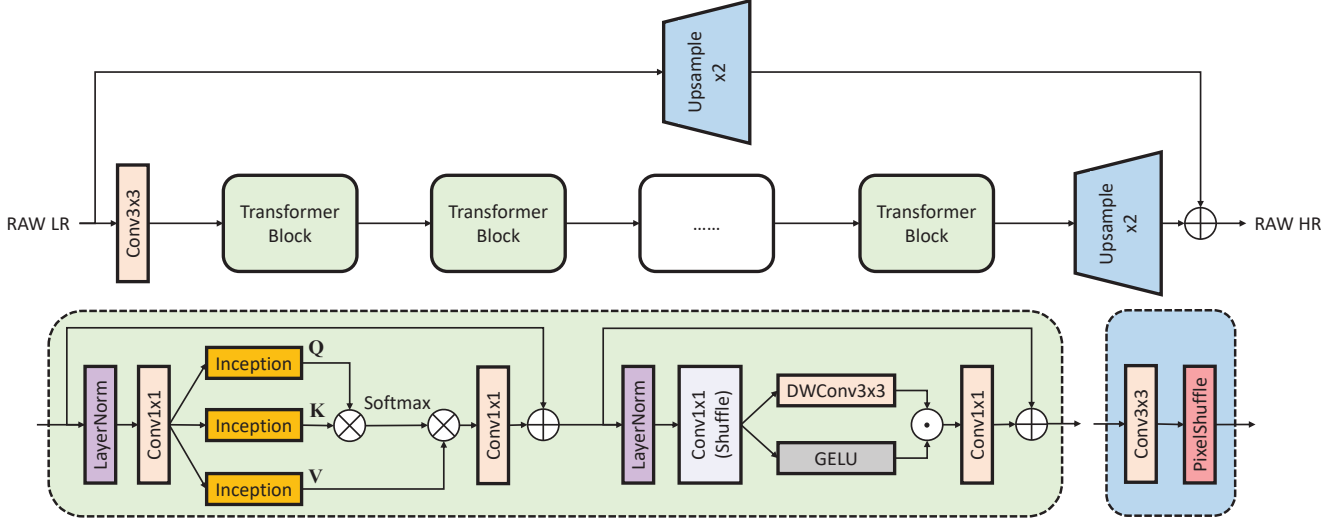


Figure 4. Team USTC framework for RAW image super resolution.

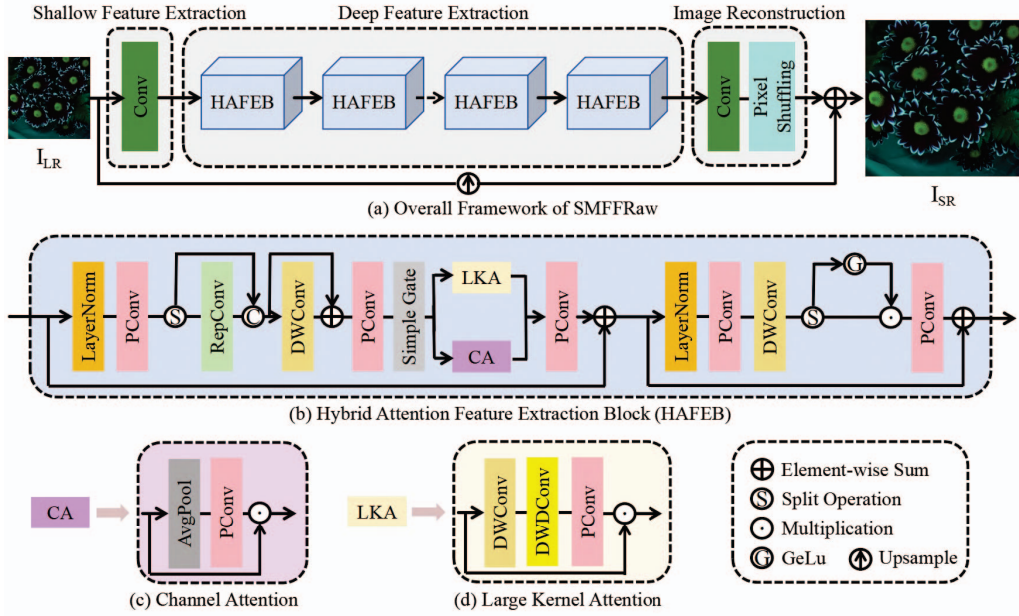


Figure 5. The proposed network architecture of SMFFRaw by Team XJTU. (a) Overall framework. (b) Hybrid Attention Feature Extraction Block (HAFEB). (c) Channel Attention (CA). (d) Large Kernel Attention (LKA).

Shallow Feature Extraction: Given a degraded input image I_{LR} , a simple 3×3 convolutional operation is used to extract shallow features F_0 .

Deep Feature Extraction: Deep features are extracted using a sequential of Hybrid Attention Feature Extraction Block (HAFEB) modules, each of which consists of operations such as Point-wise Convolution (Pconv), Depth-wise Convolution (DWconv), Reparameterized Convolution (RepConv), Channel Attention (CA) and Large Kernel Attention (LKA) as demonstrated in Figure 5(b). No reparameterization is applied during the inference phase.

parameterization is applied during the inference phase.

Reconstruction: In the reconstruction stage, the feature map is first upsampled using a 3×3 convolutional layer with pixel shuffle [58], and then added to the bilinearly interpolated input to produce the final result I_{SR} . This design reduces training complexity while enhancing the network’s SR performance.

Implementation Details The model is trained solely using the dataset provided by the challenge organizers. Our

Input	Training Time	Train E2E	Extra Data	# Params (M)	GPU
(512, 512, 4)	99h	Yes	No	0.182	RTX 4090
(512, 512, 4)	174h	Yes	No	1.994	RTX 4090

Table 4. Training details of SMFFRaw showing input dimensions, computational requirements, and model complexity.

Training Phase	Mixup	Downsample	Noise	Blur	Patch Size	Batch Size	Iterations	Training Loss
Phase 1	✓	✓	-	-	512	8	372K	Charbonnier L1+Frequency
Phase 2	✓	✓	✓	-	512	8	372K	Charbonnier L1+Frequency
Phase 3	✓	✓	✓	✓	512	8	372K	Charbonnier L1+Frequency
Phase 4	-	✓	✓	✓	1024	4	266K	Charbonnier L1+Frequency
Phase 5	-	✓	✓	✓	1024	4	266K	MSE+Frequency

Table 5. Illustration of the progressive training strategy of SMFFRaw, including augmentation techniques (i.e., mixup), degradation process (i.e., downsample, noise, blur), training patch size, batch size, training iterations and loss functions across different phases.

Method	Validation IMP	Test IMP	# Params. (M)	FLOPs (G)	Runtime (ms)
SMFFRaw-Small	42.208/0.983	Unknown	0.182	43.58	3.5
SMFFRaw-Large	42.628/0.984	Unknown	1.994	478.75	3.8

Table 6. Performance of Team XJTU SMFFRaw solutions, including PSNR/SSIM on the validation set (40 images), parameter number, FLOPs and runtime.

training pipeline consists of five stages (see Table 5). To increase data diversity, we apply common augmentations (rotation, flipping) along with mixup [76]. Degradation pipeline proposed by BSRaw [18] is used to generate RAW-degraded image pairs. The model is trained using the Adam optimizer [35] with an initial learning rate of 1e-3, which decays to 1e-6 using Cosine Annealing. The network is trained using a combination of Charbonnier loss [64] and frequency loss [31] for the first four stages, similar to RBSFormer [33], and a combination of MSE and frequency loss for the final stage. All experiments are conducted employing PyTorch on RTX 4090 GPUs. The model efficiency results are presented in Table 6 and more training details are provided in Table 4.

4.4. An Enhanced Transformer Network for Raw Image Super-Resolution

Team EGROUP

Ruixuan Jiang, Senyan Xu, Siyuan Jiang, Xueyang Fu,
Zheng-Jun Zha

University of Science and Technology of China

Contact: rxjiang21@gmail.com

Method Description Our approach leverages the RBSFormer[32] architecture to directly process RAW images for super-resolution tasks. By operating in the RAW domain rather than on sRGB images processed by ISP, we

avoid the complexities of non-linear transformations that make degradation modeling challenging.

We maintain the three-component structure of RBSFormer[32]. The overall pipeline can be described as follows:

Firstly, given a raw low-resolution image with degradation $I_{LR} \in \mathbb{R}^{H \times W \times 4}$, we apply shallow feature extraction:

$$F_s = \text{Conv}_{3 \times 3}(I_{LR}) \quad (1)$$

Next, we use transformer blocks to extract deep features:

$$F_i = \mathcal{H}_{tb_i}(F_{i-1}), i = 1, 2, \dots, K \quad (2)$$

$$F_d = \text{Conv}_{3 \times 3}(F_K) \quad (3)$$

Finally, we reconstruct the HR image by aggregating the features:

$$I_{HR} = \mathcal{H}_{rec}(I_{LR}, F_d) = \text{Up}(F_s + F_d) \quad (4)$$

We utilized the official training dataset provided by the organizers. Our focus has been on optimizing the training process through careful data augmentation in the RAW domain, implementing random noise and blur degradation patterns.

Our evaluation shows competitive results on the validation set, achieving 42.54 dB PSNR and 0.98 SSIM with only 3.3M parameters and reasonable computational requirements, making our approach both effective and efficient for RAW image super-resolution.

Implementation Details AdamW optimizer with β_1 and β_2 equal to 0.9 and 0.999 is used to train our model. The initial learning rate is set as 7×10^{-4} . We adopt the cosine annealing strategy to train the models, where the learning

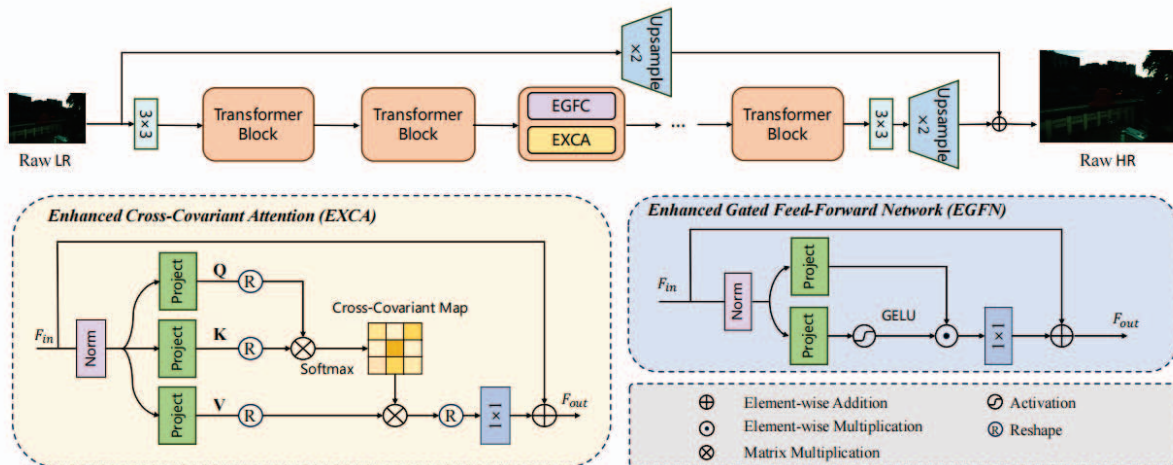


Figure 6. The architecture of the RBSFormer [32] used by Team EGROU for RAW image super-resolution.

rate gradually decreases from the initial learning rate to 1×10^{-6} . All experiments are implemented by PyTorch 1.11.0 with two NVIDIA 4090 GPUs. We used batch-size of 8 and crop-size of 192. We utilized the official training dataset provided by the organizers. We trained with L1 loss for 100k iterations, then fine-tuned with FFT loss for 20k iterations.

4.5. A fast neural network to do super-resolution based on NAFSSR

Team NJU

Xiang Yu, Guanlan Hong

Nanjing University

Contact: a2417831512@gmail.com

Team NJU RSR proposed a CNN framework for raw image super-resolution, with a design based on NAFBlock proposed in NAFSSR [14], adopting reparameterization during inference, fusing the additional parameters of Batch Normalization into the CNN layer before to achieve efficient inference.

Method Description The solution used the data provided in the NTIRE 2025 RAW Image Super Resolution challenges, with the degradation pipeline described in [18].

The architectural configuration of the Team NJU RSR is depicted in Fig 7. In the proposed method, a NAFSSR-based design is redesigned to handle single view picture. The team adopted the NAFBlock by redesign the Simple-Gate component with a CNN layer and GeLU activation function and removing FFN component to constrain the parameters of the block to build a light-weight model. The

adopted NAFBlock is shown in Fig 8. To make the training more easily and inference more efficiently, they replace Layer Normalization with Batch Normalization, which can be fused into the near CNN layer.

For a 4-channel RGGB RAW image patch of size 256×256 , the computational cost of the NAFBN amounts to 11.90 GFLOPS, being characterized by a number of 189K trainable parameters after fusing Batch Normalization layers. On a consumer-grade gaming GPU, the NVIDIA RTX3090, the forward pass needed of a 4-channel RGGB RAW image of size 256×256 estimation needs 7.19 ms after simply fusing Batch Normalization layers and 5.19 ms using half precision with almost no performance drop comparing with 9.49 ms without fusion operation.

Implementation Details The experiments are based on the Pytorch framework for implementation, with the experiments being performed on single vGPU-32 device provided by the platform AutoDL. The NAFBN model uses 12 blocks of NAFBlocks of width 48. The training procedure is based on the Adamw optimizer with the decay parameters $\beta_1 = 0.9$ and $\beta_2 = 0.99$. The momentum of Batch Normalization is set to 0.03 to avoid the performance drop due to limited batch size. The initial learning rate is 1×10^{-3} and changes with Cosine Annealing scheme to 1×10^{-6} , with the training procedure covering 50K iterations in a time-frame of around 7 hours.

The data augmentation includes random patch cropping of size 32×32 , random white balance, applied in the unit interval normalized images, random horizontal or vertical flips, random right-angle rotations and exposure adjustment by linearly scaling the images with the factor tuned to the $[-0.1, 0.1]$ interval, which are all set with probability of

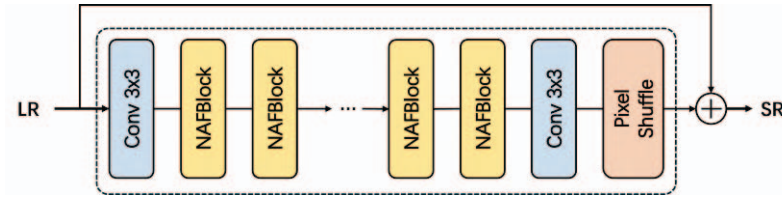


Figure 7. NAFBN proposed by Team NJU RSR.

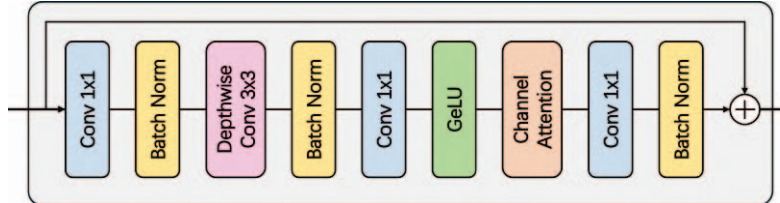


Figure 8. Adopted NAFBlock used by Team NJU RSR.

0.5 to be applied. To achieve a better performance on relatively lower resolution images, random downsample with AvePool2d and bicubic interpolation is added during each patch cropping procedure with the probability of 0.3. The training objective is based on the L_1 loss.

4.6. A efficient neural network baseline report using Mamba

Team TYSL

Minmin Yi, Yuanjia Chen, Liwen Zhang, Zijie Jin

E-surfing Vision Technology Co., Ltd, China

Contact: 441095434@qq.com

We implemented the MambaIRv2 method on the raw data with the aim of providing a baseline from a different perspective for the competition.

providing a baseline from a different perspective for the competition. This method did not use any additional data, and we simplified its architecture to obtain a lightweight model with a size of less than 0.2m. The architecture of this model is shown in Figure 9, where embedding dim = 32, $m = 4$, and $n = 2$. We chose the Mamba model because of its great potential for lightweighting and because no one had ever used it on raw data before. In addition, we have done a lot of work on downsampling. We have tried many downsampling methods, including 1. directly performing bicubic downsampling on each channel, 2. using AvgPool2D as provided in the competition, and 3. using bicubic downsampling with bias, as shown in Figure 2. However, among these methods, AvgPool2D performs much better than the others. Since we do not know whether the test set images in the competition are synthetic or real data, if they are synthetic data, the downsampling in the synthesis method will have a significant impact.

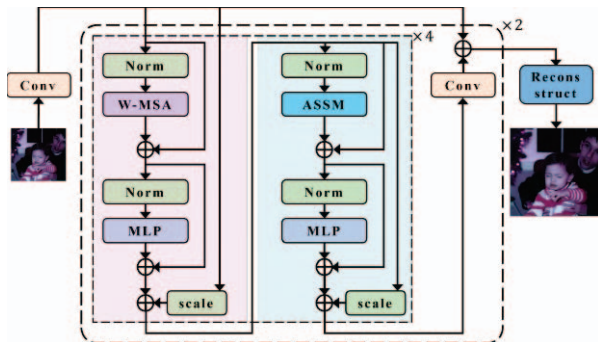


Figure 9. MambaIRv2 structure

Method Description We implemented the MambaIRv2[24] method on raw data with the aim of

Implementation Details We propose a new downsampling method based on the intuition that the central pixel value of a region is closer to the region’s average pixel value. Taking the smallest 4x4 unit as an example, after downsampling, it transforms into a 2x2 unit with an RGGGB arrangement. In this structure, the red pixel at the top-left corner after downsampling represents the average value of the top-left 2x2 pixels before downsampling, as shown in Figure 10(b). If we directly apply bicubic downsampling or ‘avgpool2d’ to the red channel of the original image, the resulting value should approximate the corresponding position in the original 4x4 image, specifically the pixel at coordinate (1,1), as illustrated in Figure 10(a). Notably, this value ends up at the bottom-right of the red pixel in the downsampled 2x2 image rather than at its center. There-

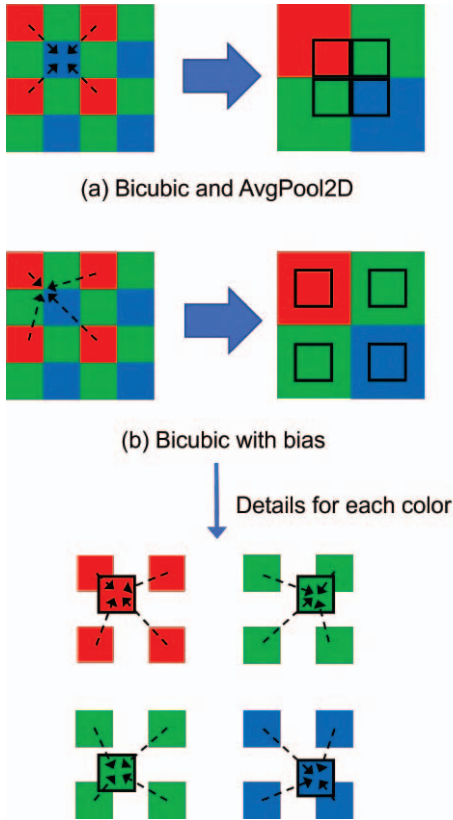


Figure 10. Downsampling structure

fore, if we perform bicubic interpolation using the nearest 16 points, with the interpolation point positioned at the center of the downsampled red pixel, we believe this approach achieves more precise downsampling.

As for other settings, we conducted our experiments on the server with several A100 GPUs, using the PyTorch framework. The batch size was set to 64, and the learning rate was configured as $8e-4$, resulting in a total training time of approximately 26 hours. We identified that the primary factor contributing to the slow training speed was the prolonged image degradation process. Utilizing the provided training set, we adhered to the given degradation pipeline, except for the downsampling step, and did not apply any image enhancement.

4.7. RepRawSR: Accelerating Raw Image Super-Resolution with Reparameterization

Team EiffLowCVer

Jiajie Lu
Politecnico di Milano

Contact: lujiajie010@126.com

Method	FLOPs	Val (Self)	Val
NAFnet-1.9M(baseline)	9.68G	40.21	41.70
RepLarge-97k	24.42G	39.22	40.80
RepTiny-21k	5.65G	39.00	–

Table 7. Ablation study by Team Team EiffLowCVer. PSNR results on the self-organized validation set (40 images selected from the training set) and the validation set (40 images). Testing of the Tiny-21k model on Codalab was not conducted due to submission limits in the test phase.

Method Description We designed two lightweight variants of previous SYEnet[23] for raw image super-resolution, incorporating structural reparameterization and efficient network design. The Large-97k model uses 32 channels as the global feature dimension and adds an additional feature extraction module before the original SYEnet[23] blocks, resulting in higher PSNR performance. The Tiny-21k model employs 16-channel feature extraction modules, achieving faster processing speeds and minimal computational complexity with only 21k parameters after reparameterization.

We conducted secondary development based on SYEnet and found that common operations, such as increasing the number of processing blocks or simply enlarging the channel numbers, tend to cause instability during training, resulting in suboptimal performance. To stabilize the training process and achieve better results, we carried out extensive experiments and ultimately trained the following two models on the provided training dataset.

- RepTiny-21k: The original SYEnet used only one feature extraction module, which limiting feature extraction capability. Our Tiny-21k model increases this to four and introduces skip connections (red arrows) to mitigate gradient vanishing. We set the channel number to 16 for efficiency. With an input of (512,512,4), our model achieves 5.65G FLOPs, 21k parameters, and 39.00 dB PSNR on a self-organized validation set
- RepLarge-97k: Another approach to enhance the model is increasing the channel width instead of network depth. We set the channel number to 32, used only one Feature Extraction Module, and incorporated the FEBlock—a preprocessing module from SYEnet designed for super-resolution task. This configuration improved PSNR by 0.22 dB on the self-organized validation set. However, due to the wider channels and added FEBlock, the parameters and FLOPs increased to 5 times the original.

Based on the experimental results, we observe that increasing the number of feature extraction modules while employing skip connections for training stability enables Tiny-21k to achieve an optimal balance between performance and speed.

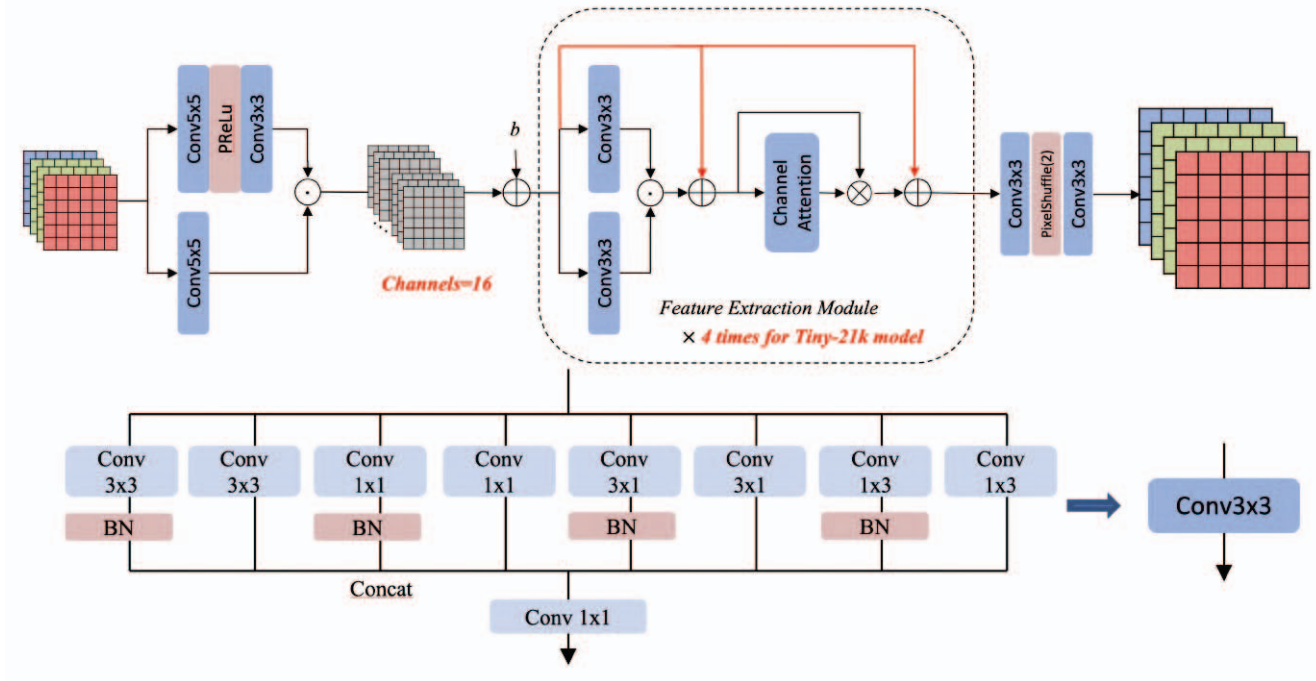


Figure 11. Main branch of RepRawSR proposed by Team EffiLowCVer.

Input	Training Time	Train E2E	Extra Data	# Params. (M)	GPU
(256,256,3)	22h	Yes	No	0.021 Million for Tiny-21k	3090
(256,256,3)	26h	Yes	No	0.097 Million for Large-97k	3090

Table 8. The overall training details of RepRawSR.

Implementation Details

- **Optimizer and Learning Rate:** We used the Adam optimizer with an initial learning rate of 8×10^{-4} and adopted CosineAnnealingRestartLR to schedule the learning rate.
- **GPU:** NVIDIA GeForce RTX 3090 24Gb
- **Datasets:** We used only the 1,064 RAW images provided by the organizers as the dataset, with 40 images randomly selected as the validation set during training.
- **Training Time:** 22 hours
- **Training Strategies:** We adopted a multi-stage training strategy. In the first stage, during 100,000 training steps, we randomly cropped 256×256 GT patches, applied random rotation and flipping, and then generated LQ images using the degradation pipeline provided by the organizers as model input. In the second stage, the patch size was increased to 384×384 , and training continued for an additional 50,000 steps.
- **Efficiency Optimization Strategies:** To stabilize training, an additional tail generated a second predicted image from intermediate feature maps, which was included in the loss calculation. This branch is removed during inference.

4.8. ECAN: Efficient Channel Attention Network for RAW Image Super-Resolution

Team CUEE-MDAP

Watchara Ruangsang

*Multimedia Data Analytics and Processing Unit,
Department of Electrical Engineering, Faculty of
Engineering,*

Chulalongkorn University, Thailand

Contact: watchara.knot@gmail.com

Method Description Many high-performing super-resolution (SR) models suffer from high computational complexity. As noted in [54], "many approaches are complex and are difficult to use in practical scenarios." Furthermore, "computational cost and model parameters are the most challenging limitations in real-world applications" [55]. ECAN aims to create an "efficient Super-resolution algorithm" [54] for the NTIRE 2025 Efficient Track ($<0.2M$ parameters).

ECAN is a CNN-based model. No external pre-trained

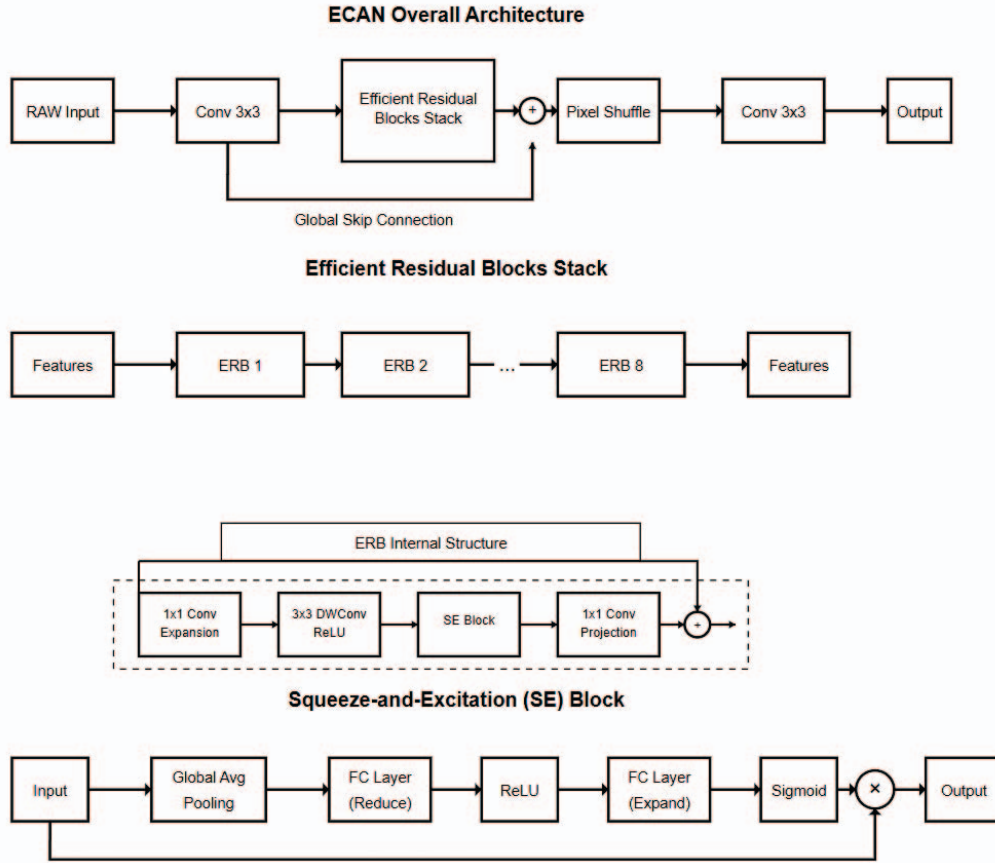


Figure 12. The overall architecture of the proposed ECAN model, showing the main pipeline and details of the EfficientResidualBlock with the integrated SEBlock.

models were used. It was trained end-to-end on the NTIRE 2025 RAW training dataset using standard augmentations and a specific degradation pipeline.

The ECAN architecture (Fig. 12) uses four stages: (1) Shallow Feature Extraction (3x3 Conv on 4-channel RAW input), (2) Deep Feature Extraction (8 ‘EfficientResidualBlocks’ with global skip connection), (3) Upsampling (PixelShuffle), and (4) Reconstruction (3x3 Conv to 4-channel RAW). Each ‘EfficientResidualBlock’ uses an inverted residual structure with depthwise separable convolutions (inspired by MobileNetV2 [57]) and a Squeeze-and-Excitation (SE) block [27] for channel attention. This focus on channel interaction aligns with our group’s work on Multi-FusNet [55] and channel attention networks [79].

ECAN has only **93,092 parameters** ($\approx 0.093\text{M}$), significantly below the 0.2M limit. The computational cost is estimated at 21.82 GMACs (or 43.65 GFLOPs) for a 512x512x4 input (scaling to 1MP output). Inference time is approximately **8.25 ms per output megapixel** on an NVIDIA RTX 4090 (measured without AMP).

Implementation Details

- **Framework:** PyTorch
- **Optimizer:** AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$), weight decay 1×10^{-4} .
- **Learning Rate:** Initial 4×10^{-4} , cosine annealing to 1×10^{-7} .
- **GPU:** NVIDIA RTX 4090 (24GB).
- **Datasets:** NTIRE 2025 RAW training set. Augmentation: Random 90/180/270 rotations, horizontal flips. Degradation: Gaussian Blur ($\sigma \leq 4.0, p = 0.7$) + Gaussian Noise ($level \leq 0.04, p = 0.95$). No extra data used.
- **Training Time:** 600 epochs (1.6 hours).
- **Training Strategies:** End-to-end from scratch, Automatic Mixed Precision (AMP). Input patch size 128x128, batch size 64. L1 Loss. Gradient clipping at 1.0.

5. RAW Image Restoration Methods

5.1. Efficient RAW Image Restoration

Team SamsungAI

Xiangyu Kong¹, Xiaoxia Xing¹,
Suejin Han², MinKyu Park²

¹ Samsung R&D Institute China - Beijing (SRC-B)

² The Department of Camera Innovation Group, Samsung Electronics

Contact: xiangyu.kong@samsung.com

Method Description In the RAW Restoration Challenge, we designed our model structure based mainly on Nafnet [7], a lightweight network for RAW image restoration, available in two variants: ERIRNet-S (5M parameters) and ERIRNet-T (0.2M parameters). To satisfy parameter constraints while preserving performance under diverse mobile RAW degradations, we reduced Nafnet parameters and implemented the distillation strategy. X-Restormer [10] is adopted as the teacher model, with two architecturally distinct student models trained through knowledge distillation from this shared teacher.

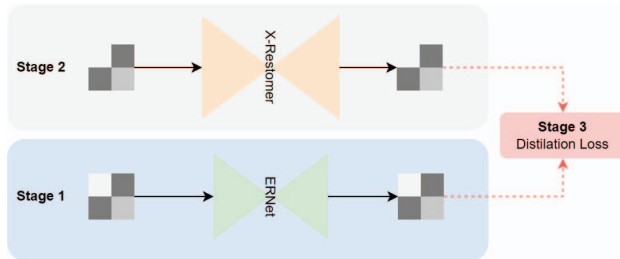


Figure 13. Training Stage Description by Samsung AI

Model Framework Raw image restoration addresses coupled degradation processes involving noise suppression and blur correction. Methodologically, we select the NafBlock architecture from Nafet [7] as architecturally pivotal components, given their demonstrated SOTA efficacy in joint denoising and deblurring tasks. To achieve parameter efficiency without compromising restoration fidelity, we designed networks based on a narrow-and-deep architectural principle, reducing channel dimensions while preserving layer depth, which resulted in two architecture variants with distinct parameter configurations as shown in Table 9:

- ERIRNet-S is a simplified version of NAFNet with reduced channel width and fewer encoder-decoder blocks for improved efficiency.
- ERIRNet-T further reduces complexity by decreasing the number of blocks and using smaller FFN expansion ratios. It also replaces PixelUnshuffle layers with Con-

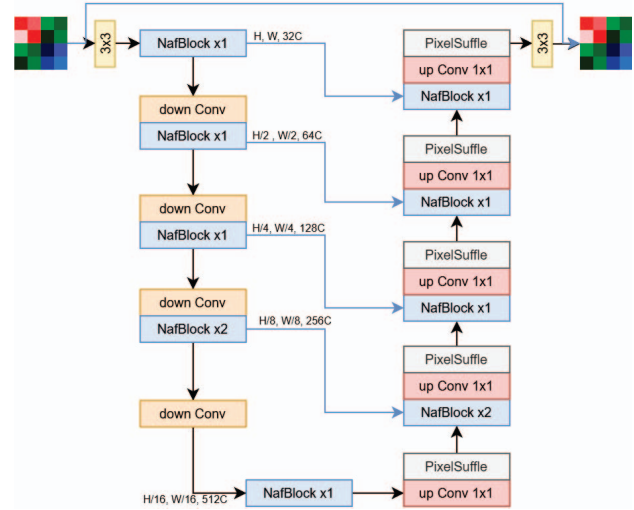


Figure 14. Architectures of ERIRNet-S, with reduced channels and fewer blocks. Proposal by Samsung AI.

vTranspose, enabling deeper architectures under a strict parameter budget.

Implementation Details We trained our model in three steps. The training process consisted of three stages, as shown in Fig. 13:

- **Stage 1 – Train Base Model:** Each ERIRNet variant was trained independently using the original ground truth supervision.
- **Stage 2 - Train Teacher Model:** We have trained a Teacher Model based on X-Restormer and fine-tuned the teacher model on each mobile device.
- **Stage 3 – Distillation with Teacher Model:** We applied knowledge distillation using a teacher model. The model was initialized with the weights trained in the Stage 1. The teacher’s outputs were used as targets to guide both ERIRNet-S and ERIRNet-T.

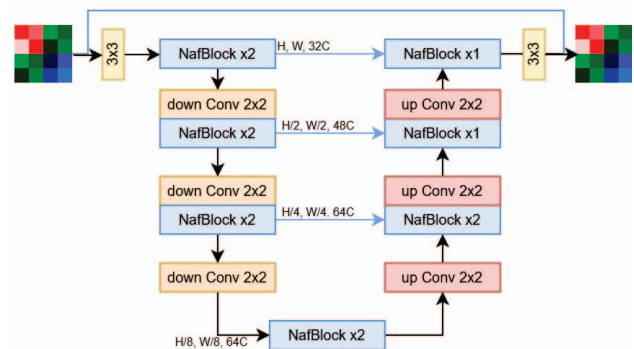


Figure 15. Architectures of ERIRNet-T, with ConvTranspose and reduced FFN expansion. Proposal by Samsung AI.

Model	Input	Training Time	Train E2E	Extra Data	FLOPs (GMac)	# Params. (M)	GPU
ERIRNet-S	(512, 512, 4)	24h	Yes	No	23.79	4.97	A100
ERIRNet-T	(512, 512, 4)	26h	Yes	No	10.98	0.19	A100

Table 9. Training details of SamsungAI solution.

We used the PyTorch framework for training and utilized A100 GPU. We used the Adam optimizer [35] with $\beta_1 = 0.5$, $\beta_2 = 0.999$ and L1 loss for supervision. The initial learning rate was set to $1e - 4$ at stage 1 and reduced to $1e - 5$ on stage 3. The model was trained for 1000, 2000, and 1000 epochs in stages 1, 2, and 3, respectively. We implemented a MultiStepLR scheduler to manage learning rate decay, conducting training with a batch size of 16 on NVIDIA A100 GPU.

5.2. Modified SwinFIR-Tiny for Raw Image Restoration

Team Miers

Cheng Li, Lian Liu, Wei Song, Heng Sun
Xiaomi Inc.

Contact: licheng8@xiaomi.com

Method Description Our method is based on an improvement of SwinFIR-Tiny[76]. By aggregating the outputs of different RSTB modules to enhance the model’s feature representation capability, and also incorporating the HAB module from HAT[8] via zero convolution, and applying the reparameterization technique[23], the final model has a parameter count of 4.76M and achieved a PSNR of 41.73 on the CodaLab server.

We selected the SwinFIR-Tiny[76] architecture as our baseline framework for evaluating the efficacy of our proposed methodology. The complete network architecture is systematically illustrated in Figure 16. Our innovative scheme incorporates four Residual Swin Transformer Blocks (RSTBs), which have been extensively validated through empirical analysis to demonstrate exceptional performance in hierarchical feature extraction. Each RSTB consists of 5 or 6 Hybrid Attention Blocks (HABs) and 1 HSFB. Each HAB integrates a multi-head self-attention mechanism and a locally constrained attention module. These complementary mechanisms play a pivotal role in capturing multi-scale contextual information, thereby significantly enhancing the network’s representation capabilities across different spatial dimensions.

To further optimize the architectural efficiency, we designed a hierarchical FeatureFusion strategy. This novel structure systematically aggregates the outputs from each RSTB block, effectively mitigating the issue of shallow feature degradation during deep feature extraction. By pre-

serving critical information across hierarchical stages, this fusion mechanism ensures seamless information flow and promotes more discriminative feature abstraction at multiple scales. Experimental evaluations on challenging datasets demonstrate that this architectural configuration achieves substantial performance improvements in handling severely blurred scenes, producing sharper and more visually consistent outputs.

Complementing these core components, our design integrates specialized enhancements including the Channel Attention Block (CAB[8]) and CovRep5[23] module. These innovations are specifically engineered to improve noise robustness and blur resilience, enabling precise reconstruction of degraded inputs with superior fidelity.

Implementation details Our code is implemented using the PyTorch framework and is modified based on the SwinFIR project. The dataset is divided into a training set and a validation set, with 2,099 samples in the training set and 40 samples in the validation set. The data degradation pipeline is sourced from BSRRAW[18], and we have enhanced the noise levels within it. Specifically, we made the following adjustments:

- The value of `log_max_shot_noise` for shot noise was increased from -3 to -2;
- For heteroscedastic Gaussian noise, the range of `sigma_1_range` was extended from (5e-3, 5e-2) to (5e-3, 1e-1), and the range of `sigma_2_range` was expanded from (1e-3, 1e-2) to (1e-3, 5e-2).

The entire training process was conducted on a machine equipped with four H800 80G GPUs. Data augmentation was performed using mixup, and the Adam optimizer was used. The loss function used was the Charbonnier loss. Our approach is based on the SwinFIR-Tiny model and involves four stages of development:

Firstly, we used the original SwinFIR-Tiny model as the baseline, employing the original data degradation method. The initial learning rate was set to $2e-4$, with a batch size of 8 and an input size of 180×180 . The model was trained for 250K iterations.

Secondly, we added a Feature Fusion module, a Channel Attention module, and a ConvRep5 module to the SwinFIR-Tiny model. The model trained in the first stage was used as the initial parameter set. The original data degradation method was retained, with an initial learning rate of $2e-4$, a batch size of 8, and an input size of 180×180 . The model

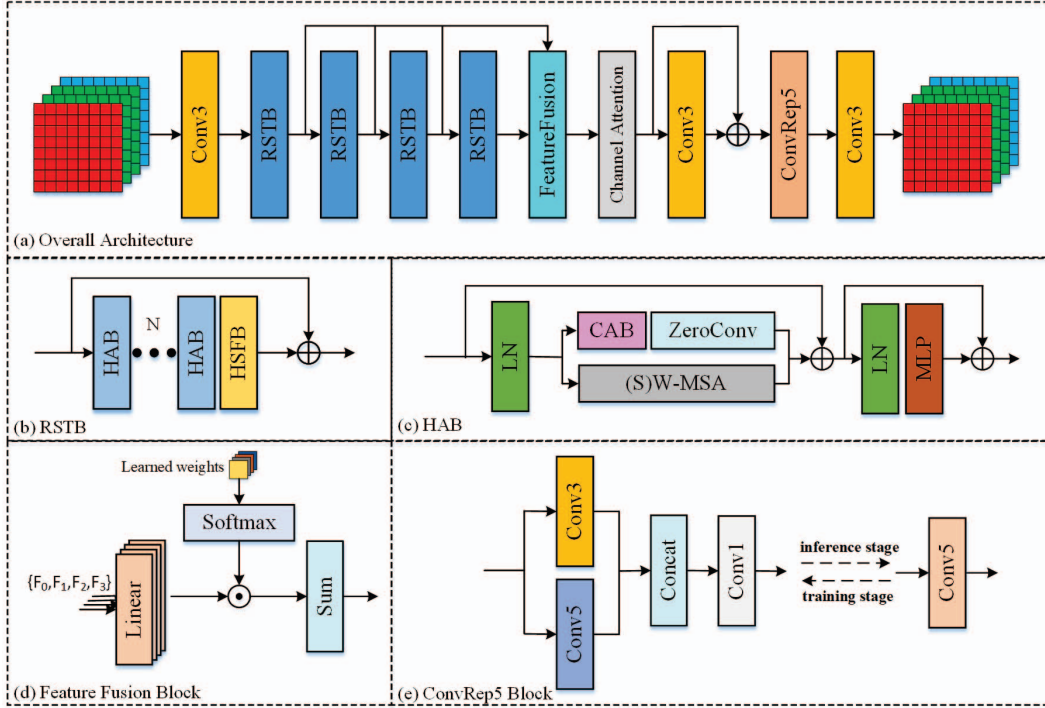


Figure 16. CABATTswinFIR proposed method by Team Miers (Xiaomi Inc.).

was trained for 170K iterations. Third, building on the second stage, we introduced the CAB module using zero convolution and increased the noise intensity. The initial learning rate was set to $3e-5$, with a batch size of 8 and an input size of 180×180 . The model was trained for 140K iterations.

Finally, based on the third stage, we further adjusted the initial learning rate to $2e-5$, reduced the batch size to 2, and increased the input size to 360×360 . The model was trained for 15K iterations. The final model that we submitted utilized reparameterization techniques to convert the ConvRep5 module into a standard 5×5 convolution. The total number of parameters in the final model is 4.76M.

5.3. Multi-PromptIR: Multi-scale Prompt-based Raw Image Restoration

Team WIRTeam

Yubo Wang, Jinghua Wang
Harbin Institute of Technology (Shenzhen)

Contact: wangyubo@stu.hit.edu.cn

Method Description Building on the latest improvements in image restoration research, as presented in [11, 37], we introduce the multi-scale prompt-based raw image restoration (Multi-PromptIR) method. It transforms a degraded raw image $\mathbf{I} \in \mathbb{R}^{H \times W \times 4}$ into a high-quality,

clear image $\bar{\mathbf{I}} \in \mathbb{R}^{H \times W \times 4}$. The core of our method is a four layers encoder-decoder architecture, where Transformer Blocks [75] are integrated to enable cross-channel global feature extraction. During the encoding stage, drawing from the success of leveraging multi-resolution images in image restoration [11], we incorporate images at reduced resolutions ($1/2$, $1/4$, and $1/8$ of the original size). This additional information stream significantly enriches the encoding process. In the decoding phase, we adopt a specialized prompt mechanism [51], which consists of a Prompt Generation Module (PGM) and a Prompt Interaction Module (PIM).

The model is based PromptIR [51] and Restormer [75], which combines CNNs and Transformers with specific design demonstrated in Figure 17. The total number of parameters in the final model is 39.92 M.

Implementation details

- **Framework:** Pytorch
- **Optimizer and Learning Rate:** AdamW (the initial learning rate is 2×10^{-4} and reduces to 1×10^{-6} with the cosine annealing schedule.)
- **GPU:** 1 * NVIDIA A100 (80G)
- **Datasets:** We merely adopt the datasets provided by organizers. We apply psf blur and add noise to the original images to synthetic degradations.
- **Training and Inference**

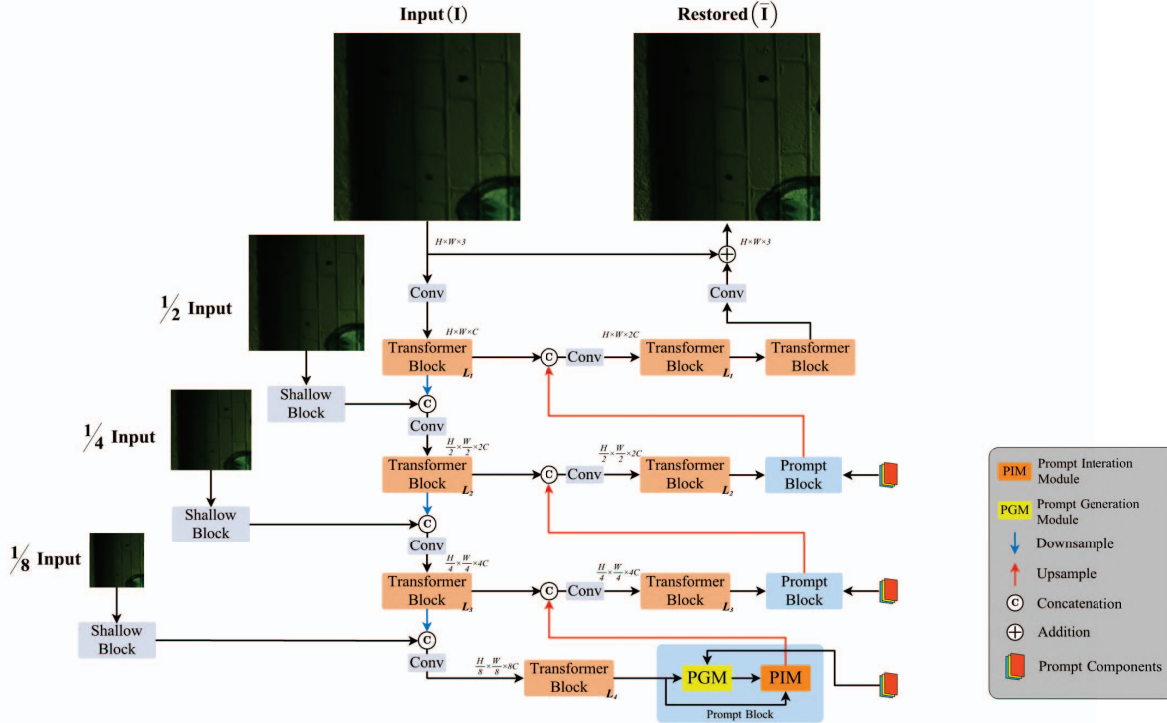


Figure 17. Overall architecture proposed by the Team WIRTeam.

We train the model in the end-to-end manner for 700 epochs. The training procedure is based on the AdamW optimizer with the decay parameters $\beta_1 = 0.9$ and $\beta_2 = 0.99$. The initial learning rate is 2×10^{-4} and gradually reduces to 1×10^{-6} with the cosine annealing strategy. Horizontal and vertical flips are adopted for data augmentation in the dataloader.

We split the input degraded images into 256*256 patches and restore them after the process of our proposed architecture.

- **Training Time:** Around 12 hours

5.4. LMPR-Net: Lightweight Multi-Stage Progressive RAW Restoration

Team WIRTeam

Yubo Wang, Jinghua Wang
Harbin Institute of Technology (Shenzhen)

Contact: wangyubo@stu.hit.edu.cn

Method Description Considering the great potential of multi-stage feature interaction in the image restoration process, we propose a lightweight model for raw image restoration based on MPRNet [74]. The multi-stage model can decompose the raw image restoration task into multiple sub-tasks to address various degradation information (such as

noise, blurring, and other unknown degradations) in this challenge. In the LMPR-Net, the original resolution block (ORB) is composed of convolution and channel attention mechanism to achieve the extraction of cross-channel key features, and the SAM can efficiently refine the incoming features at each stage. For the sake of lightness, we have simplified some components and set the hidden dimension to be 8. Besides, we introduce depthwise over-parameterized convolution [4] to increase the training speed and improve the expressive power of the model without significantly increasing the computational cost. During the training process, we use the Charbonnier loss [5] for constraint and to avoid the restored image from being overly smooth.

We select MPRNet [74] as the baseline for our method. Our proposed (shown in the Fig 18 solution builds on MPRNet, introducing targeted improvements that not only enhance performance but also significantly reduce model complexity, thereby delivering a lightweight architecture with low parameter count and flops while keeping excellent raw restoration performance. And the total number of parameters in the final model is 0.19M with the MACs of 2.63G FLOPs.

Implementation details

- **Framework:** PyTorch
- **Optimizer and Learning Rate:** AdamW(the initial

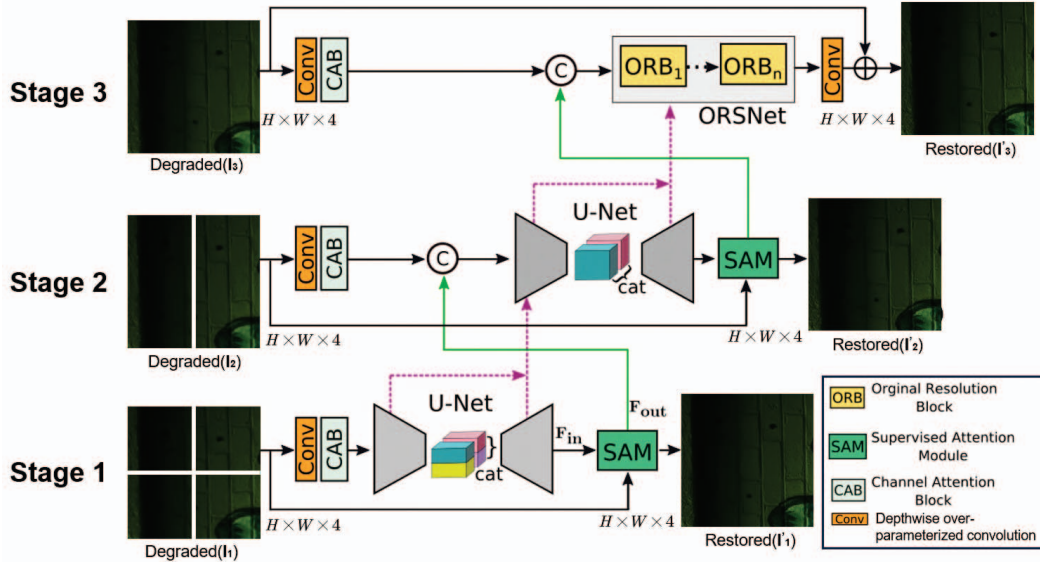


Figure 18. Overall architecture of the LMPR-Net method proposed by the Team WIRTeam.

learning rate is 2×10^{-4} and reduces to 1×10^{-6} with the cosine annealing schedule.)

- **GPU:** NVIDIA GeForce RTX 4090 (24G) * 1
- **Datasets:** We merely adopt the datasets provided by organizers. We apply psf blur and add noise to the original raw images to synthetic degradations. Horizontal and vertical flips are used for data augmentation.
- **Training and Inference** We train our model in an end-to-end manner for 600 epochs. The training process employs the AdamW optimizer with decay parameters set to $\beta_1 = 0.9$ and $\beta_2 = 0.99$. The initial learning rate is set at 2×10^{-4} , which gradually decreases to 1×10^{-6} following a cosine annealing schedule. Data augmentation involves horizontal and vertical flips.

During the inference stage, we partition the input degraded images into 256×256 patches. After passing through our proposed architecture, these processed patches are seamlessly merged to return to the original size.

- **Training Time:** About 10 hours.

5.5. ER-NAFNet Raw Restoration

Team ER-NAFNet

Tianyu Hao¹, Yuhong He², Ruoqi Li³, Yueqi Yang⁴

¹ Huazhong University of Science and Technology

² Northeastern University

³ Dalian University of Technology

⁴ Institute of Automation, Chinese Academy of Sciences

Contact: tianyuhao.times@gmail.com

Method Description The team introduces ER-NAFNet, a U-shaped framework designed for Efficient raw image Restoration. The architecture and compression mechanisms of ER-NAFNet are built upon the NAFNet block proposed in NAFNet[6](see Fig. 19), enhancing its efficiency and performance in image restoration task. The ER-NAFNet is trained and learned directly from the 4-channel RGGB RAW data with a complex blurring and noise degradation pipeline, which the noise model, blur kernel and degradation model are demonstrated in AISP[20].

Dataset Degradation method, inspired by [53] [61] and [77], in order to speed up training process and enable the model to learn real degradation, the Bayer pattern RAW images are cropped with 512×512 . We use a sequence of degradation operations composed of multiple blurring, hardware specific added noise, random digital gain by luma or exposure compensations. When adding noise, in order to better handle noises in dark areas, we capture dark frames using multiple mobile sensors to model the real dark current noise. In addition, we remove the pixels in the highlight area in order to prevent highlight color cast.

Model Framework The architectural configuration of the model proposed by Team ChickenRun is illustrated in Fig. 19. This network is structured into three main components: shallow feature extraction, deep feature extraction, and raw reconstruction. Initially, a low-quality raw image $I^{LQ} \in \mathcal{R}^{H \times W \times 4}$ is provided as input. The team applies a 3×3 convolutional filter to extract shallow feature encodings F^s . Next, a classic U-shaped architecture with skip connections is employed to perform deep feature extraction. Finally, a 3×3 convolution is applied to reconstruct the high-quality

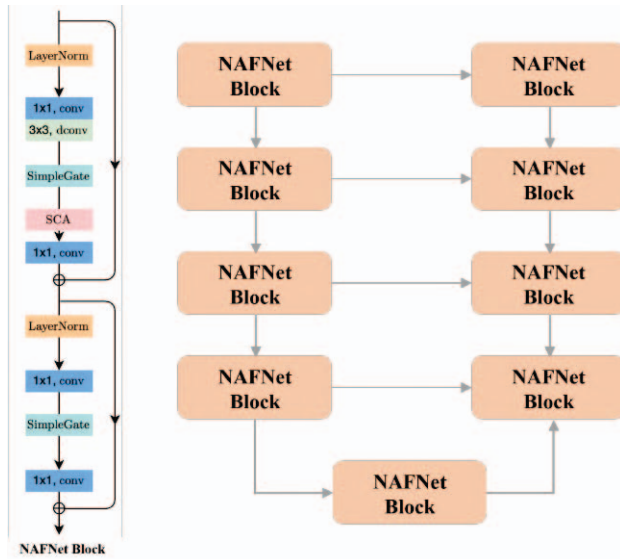


Figure 19. The overall network architecture proposed by Team ER-NAFNet.

(HQ) image.

As depicted in Fig. 19, the NAFNet (Non-Attention Feature Network) Block plays a pivotal role in the U-shaped architecture, addressing the limitations of traditional attention mechanisms in neural networks. The NAFNet Block integrates a series of convolutional operations, including 1x1 and 3x3 dilated convolutions, to effectively capture both fine-grained details and large-scale patterns within the input data. By eliminating the reliance on complex attention mechanisms, the NAFNet Block achieves superior performance with significantly reduced computational overhead. Furthermore, the incorporation of the SimpleGate and Simple Channel Attention (SCA) modules enables the network to focus on the most relevant features while suppressing irrelevant ones, thereby enhancing the overall quality of the feature representations.

Implementation details The solution is optimized solely on the NTIRE 2024 official challenge data [20], using the proposed Development Phase submission set for validation. The experiments are based on the Pytorch framework for implementation. The model uses a width 16, with [2, 2, 4, 8] encoder blocks and [2, 2, 2, 2] decoder blocks at each stage. The middle block num is set to 6. Since the degraded raw images are characterized by low quality, with a considerable level of details being lost, a data augmentation technique is applied, to improve the training procedure in terms of stability, convergence, and the achieved performance level. The strategy combines simple horizontal or vertical flips with channel shifts and mixup augmentations. The training objective is based on the L2 loss.

The optimization technique used is the AdamW [46] optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 0.00001) with the cosine annealing strategy, where the learning rate gradually decreases from the initial learning rate 3×10^{-4} to 1×10^{-6} for 300000 iterations. The training batch size is set to 12 and patch size is 512. All experiments are conducted on A100 GPUs.

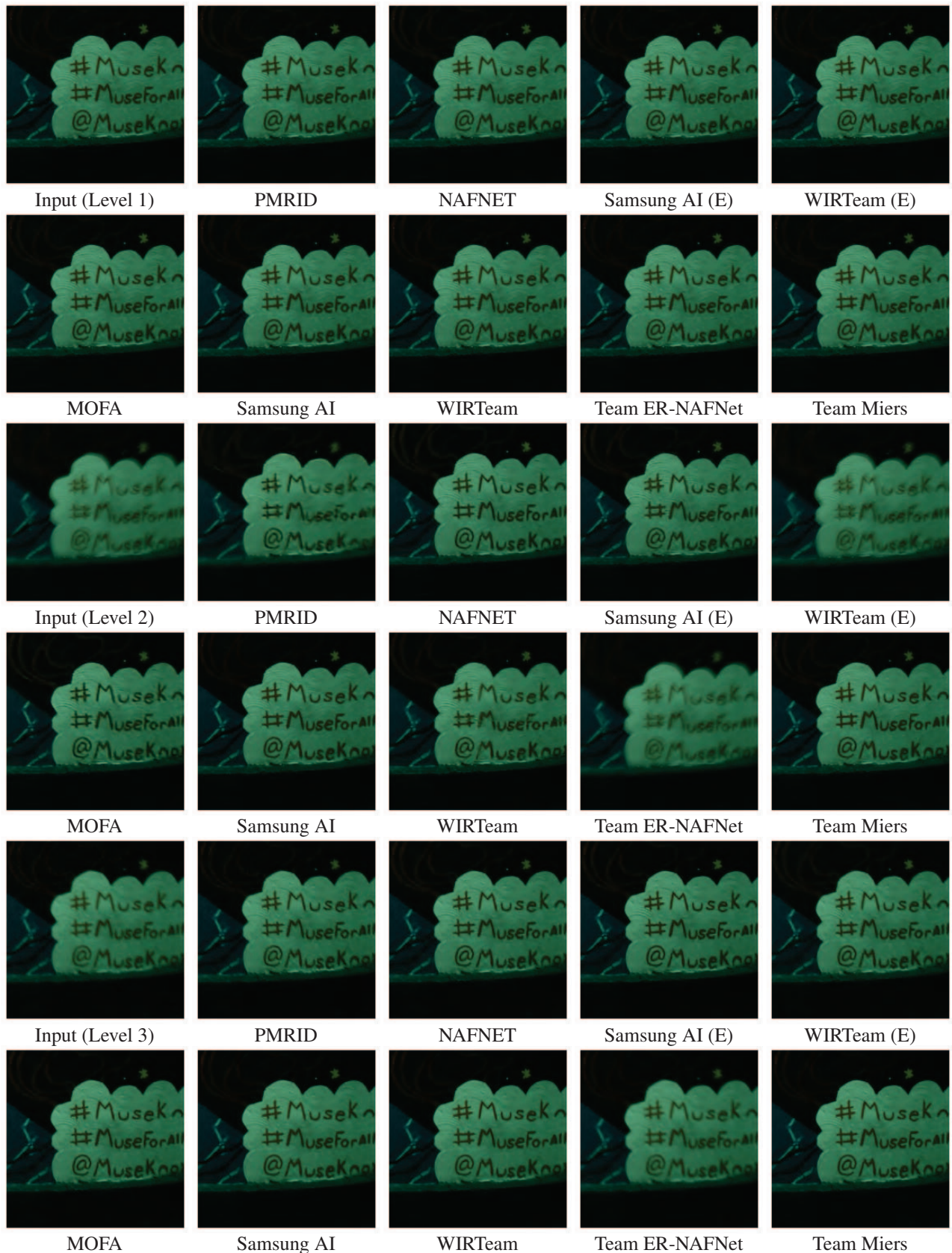


Figure 20. Visual comparison using the **NTIRE 2025 Challenge on RAW Image Restoration Challenge** testing set (`gp_2-11.npy`). RAW images have 512×512 resolution and 4-channels (RGGB Bayer pattern). RAW images are visualized using bilinear demosaicing, gamma correction and tone mapping. (E) indicates efficient methods.

References

- [1] Mahmoud Afifi and Abdullah Abuolaim. Semi-supervised raw-to-raw mapping. *arXiv preprint arXiv:2106.13883*, 2021. [3](#)
- [2] Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11036–11045, 2019. [1](#)
- [3] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *CVPR*, 2011. [2](#)
- [4] Jinming Cao, Yangyan Li, Mingchao Sun, Ying Chen, Dani Lischinski, Daniel Cohen-Or, Baoquan Chen, and Changhe Tu. Do-conv: Depthwise over-parameterized convolutional layer. *IEEE Transactions on Image Processing*, 31:3726–3736, 2022. [17](#)
- [5] Pierre Charbonnier, Laure Blanc-Feraud, Gilles Aubert, and Michel Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of 1st international conference on image processing*, pages 168–172. IEEE, 1994. [17](#)
- [6] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European conference on computer vision*, pages 17–33. Springer, 2022. [18](#)
- [7] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European conference on computer vision*, pages 17–33. Springer, 2022. [3](#), [4](#), [14](#)
- [8] Xiangyu Chen, Xintao Wang, Wenlong Zhang, Xiangtao Kong, Yu Qiao, Jiantao Zhou, and Chao Dong. Hat: Hybrid attention transformer for image restoration. *arXiv preprint arXiv:2309.05239*, 2023. [15](#)
- [9] Xiangyu Chen, Ruiwen Zhen, Shuai Li, Xiaotian Li, and Guanghui Wang. Mofa: A model simplification roadmap for image restoration on mobile devices. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1322–1332, 2023. [4](#)
- [10] Xiangyu Chen, Zheyuan Li, Yuandong Pu, Yihao Liu, Jiantao Zhou, Yu Qiao, and Chao Dong. A comparative study of image restoration networks for general backbone network design. In *European Conference on Computer Vision*, pages 74–91. Springer, 2024. [4](#), [14](#)
- [11] Xiang Chen, Jinshan Pan, and Jiangxin Dong. Bidirectional multi-scale implicit neural representations for image deraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25627–25636, 2024. [16](#)
- [12] Zheng Chen, Kai Liu, Jue Gong, Jingkai Wang, Lei Sun, Zongwei Wu, Radu Timofte, Yulun Zhang, et al. NTIRE 2025 challenge on image super-resolution (x4): Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. [2](#)
- [13] Zheng Chen, Jingkai Wang, Kai Liu, Jue Gong, Lei Sun, Zongwei Wu, Radu Timofte, Yulun Zhang, et al. NTIRE 2025 challenge on real-world face restoration: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. [2](#)
- [14] Xiaojie Chu, Liangyu Chen, and Wenqing Yu. Nafsr: Stereo image super-resolution using nafnet, 2022. [9](#)
- [15] Marcos Conde, Radu Timofte, et al. NTIRE 2025 challenge on raw image restoration and super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. [2](#)
- [16] Marcos Conde, Radu Timofte, et al. Raw image reconstruction from RGB on smartphones. NTIRE 2025 challenge report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. [2](#)
- [17] Marcos V Conde, Florin Vasluianu, Javier Vazquez-Corral, and Radu Timofte. Perceptual image enhancement for smartphone real-time applications. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1848–1858, 2023. [1](#)
- [18] Marcos V Conde, Florin Vasluianu, and Radu Timofte. Bsrw: Improving blind raw image super-resolution. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8500–8510, 2024. [1](#), [2](#), [3](#), [6](#), [8](#), [9](#), [15](#)
- [19] Marcos V Conde, Florin Vasluianu, and Radu Timofte. Deep RAW image super-resolution. a NTIRE 2024 challenge survey. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2024. [2](#)
- [20] Marcos V Conde, Florin Vasluianu, and Radu Timofte. Toward efficient deep blind raw image restoration. In *2024 IEEE International Conference on Image Processing (ICIP)*, pages 1725–1731. IEEE, 2024. [2](#), [18](#), [19](#)
- [21] Egor Ershov, Sergey Korchagin, Alexei Khalin, Artyom Panshin, Arseniy Terekhin, Ekaterina Zaychenkova, Georgiy Lobarev, Vsevolod Plokhotnyuk, Denis Abramov, Elisey Zhdanov, Sofia Dorogova, Yasin Mamedov, Nikola Banic, Georgii Perevozchikov, Radu Timofte, et al. NTIRE 2025 challenge on night photography rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. [2](#)
- [22] Yuqian Fu, Xingyu Qiu, Bin Ren Yanwei Fu, Radu Timofte, Nicu Sebe, Ming-Hsuan Yang, Luc Van Gool, et al. NTIRE 2025 challenge on cross-domain few-shot object detection: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. [2](#)
- [23] Weiran Gou, Ziyao Yi, Yan Xiang, Shaoqing Li, Zibin Liu, Dehui Kong, and Ke Xu. Syenet: A simple yet effective network for multiple low-level vision tasks with real-time performance on mobile device. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12182–12195, 2023. [11](#), [15](#)
- [24] Hang Guo, Yong Guo, Yaohua Zha, Yulun Zhang, Wenbo Li, Tao Dai, Shu-Tao Xia, and Yawei Li. Mambairv2: Attentive state space restoration. *arXiv preprint arXiv:2411.15269*, 2024. [10](#)
- [25] Shuhao Han, Haotian Fan, Fangyuan Kong, Wenjie Liao, Chunle Guo, Chongyi Li, Radu Timofte, et al. NTIRE 2025

- challenge on text to image generation model quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [26] Michal Hradiš, Jan Kotera, Pavel Zemčík, and Filip Šroubek. Convolutional neural networks for direct text deblurring. In *Proceedings of BMVC*, 2015. 6
- [27] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018. 13
- [28] Andrey Ignatov, Luc Van Gool, and Radu Timofte. Replacing mobile camera isp with a single deep learning model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 536–537, 2020. 1
- [29] Andrey Ignatov, Cheng-Ming Chiang, Hsien-Kai Kuo, Anastasia Sycheva, and Radu Timofte. Learned smartphone isp on mobile npus with deep learning, mobile ai 2021 challenge: Report. In *CVPR Workshops*, pages 2503–2514, 2021. 1
- [30] Varun Jain, Zongwei Wu, Quan Zou, Louis Florentin, Henrik Turbell, Sandeep Siddhartha, Radu Timofte, et al. NTIRE 2025 challenge on video quality enhancement for video conferencing: Datasets, methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [31] Siyuan Jiang, Senyan Xu, and Xingfu Wang. Rbsformer: Enhanced transformer network for raw image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 6479–6488, 2024. 8
- [32] Siyuan Jiang, Senyan Xu, and Xingfu Wang. Rbsformer: Enhanced transformer network for raw image super-resolution. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 6479–6488, 2024. 8, 9
- [33] Siyuan Jiang, Senyan Xu, and Xingfu Wang. Rbsformer: Enhanced transformer network for raw image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6479–6488, 2024. 2, 6, 8
- [34] Hakki Can Karaimer and Michael S Brown. A software platform for manipulating the camera imaging pipeline. In *ECCV*, pages 429–444, 2016. 1
- [35] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 8, 15
- [36] Sangmin Lee, Eunpil Park, Angel Canelo, Hyunhee Park, Youngjo Kim, Hyungju Chun, Xin Jin, Chongyi Li, Chun-Le Guo, Radu Timofte, et al. NTIRE 2025 challenge on efficient burst hdr and restoration: Datasets, methods, and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [37] Bingchen Li, Xin Li, Yiting Lu, Ruoyu Feng, Mengxi Guo, Shijie Zhao, Li Zhang, and Zhibo Chen. Promptcir: blind compressed image restoration with prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6442–6452, 2024. 16
- [38] Xin Li, Yeying Jin, Xin Jin, Zongwei Wu, Bingchen Li, Yufei Wang, Wenhan Yang, Yu Li, Zhibo Chen, Bihan Wen, Robby Tan, Radu Timofte, et al. NTIRE 2025 challenge on day and night raindrop removal for dual-focused images: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [39] Xin Li, Xijun Wang, Bingchen Li, Kun Yuan, Yizhen Shao, Suhang Yao, Ming Sun, Chao Zhou, Radu Timofte, and Zhibo Chen. NTIRE 2025 challenge on short-form ugc video quality assessment and enhancement: Kwaisr dataset and study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [40] Xin Li, Kun Yuan, Bingchen Li, Fengbin Guan, Yizhen Shao, Zihao Yu, Xijun Wang, Yiting Lu, Wei Luo, Suhang Yao, Ming Sun, Chao Zhou, Zhibo Chen, Radu Timofte, et al. NTIRE 2025 challenge on short-form ugc video quality assessment and enhancement: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [41] Yunxiang Li, Wenbin Zou, Qiaomu Wei, Feng Huang, and Jing Wu. Multi-level feature fusion network for lightweight stereo image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6489–6498, 2024. 6
- [42] Jie Liang, Radu Timofte, Qiaosi Yi, Zhengqiang Zhang, Shuaizheng Liu, Lingchen Sun, Rongyuan Wu, Xindong Zhang, Hui Zeng, Lei Zhang, et al. NTIRE 2025 the 2nd restore any image model (RAIM) in the wild challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [43] Jiaming Liu, Chi-Hao Wu, Yuzhi Wang, Qin Xu, Yuqian Zhou, Haibin Huang, Chuan Wang, Shaofan Cai, Yifan Ding, Haoqiang Fan, et al. Learning raw image denoising with bayer pattern unification and bayer preserving augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2
- [44] Xiaohong Liu, Xionguo Min, Qiang Hu, Xiaoyun Zhang, Jie Guo, et al. NTIRE 2025 XGC quality assessment challenge: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [45] Xiaoning Liu, Zongwei Wu, Florin-Alexandru Vasluianu, Hailong Yan, Bin Ren, Yulun Zhang, Shuhang Gu, Le Zhang, Ce Zhu, Radu Timofte, et al. NTIRE 2025 challenge on low light image enhancement: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [46] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 19
- [47] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 5
- [48] Xu Ma, Xiyang Dai, Yue Bai, Yizhou Wang, and Yun Fu. Rewrite the stars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5694–5703, 2024. 6

- [49] Xintian Mao, Yiming Liu, Fengze Liu, Qingli Li, Wei Shen, and Yan Wang. Intriguing findings of frequency selection for image deblurring. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1905–1913, 2023. 6
- [50] Seonghyeon Nam and Seon Joo Kim. Modelling the scene dependent imaging in cameras with a deep neural network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1717–1725, 2017. 1
- [51] Vaishnav Potlapalli, Syed Waqas Zamir, Salman Khan, and Fahad Khan. Promptir: Prompting for all-in-one image restoration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 16
- [52] Bin Ren, Hang Guo, Lei Sun, Zongwei Wu, Radu Timofte, Yawei Li, et al. The tenth NTIRE 2025 efficient super-resolution challenge report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [53] Dongwei Ren, Kai Zhang, Qilong Wang, Qinghua Hu, and Wangmeng Zuo. Neural blind deconvolution using deep priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3341–3350, 2020. 18
- [54] Watchara Ruangsang and Supavadee Aramvith. Efficient super-resolution algorithm using overlapping bicubic interpolation. In *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)*, pages 1–2. IEEE, 2017. 12
- [55] Watchara Ruangsang, Supavadee Aramvith, and Takao Onoye. Multi-fusnet of cross channel network for image super-resolution. *IEEE Access*, 11:56287–56299, 2023. 12, 13
- [56] Nickolay Safonov, Alexey Bryntsev, Andrey Moskalenko, Dmitry Kulikov, Dmitriy Vatolin, Radu Timofte, et al. NTIRE 2025 challenge on UGC video enhancement: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [57] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 13
- [58] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016. 6, 7
- [59] Lei Sun, Andrea Alfano, Peiqi Duan, Shaolin Su, Kaiwei Wang, Boxin Shi, Radu Timofte, Danda Pani Paudel, Luc Van Gool, et al. NTIRE 2025 challenge on event-based image deblurring: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [60] Lei Sun, Hang Guo, Bin Ren, Luc Van Gool, Radu Timofte, Yawei Li, et al. The tenth ntire 2025 image denoising challenge report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [61] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *Computer Vision—ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part IV 12*, pages 111–126. Springer, 2015. 18
- [62] Florin-Alexandru Vasluianu, Tim Seizinger, Zhuyun Zhou, Cailian Chen, Zongwei Wu, Radu Timofte, et al. NTIRE 2025 image shadow removal challenge report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [63] Florin-Alexandru Vasluianu, Tim Seizinger, Zhuyun Zhou, Zongwei Wu, Radu Timofte, et al. NTIRE 2025 ambient lighting normalization challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [64] Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 8
- [65] Yuzhi Wang, Haibin Huang, Qin Xu, Jiaming Liu, Yiqun Liu, and Jue Wang. Practical deep raw image denoising on mobile devices. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI*, pages 1–16. Springer, 2020. 3
- [66] Yingqian Wang, Zhengyu Liang, Fengyuan Zhang, Lvli Tian, Longguang Wang, Juncheng Li, Jungang Yang, Radu Timofte, Yulan Guo, et al. NTIRE 2025 challenge on light field image super-resolution: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [67] Yazhou Xing, Zian Qian, and Qifeng Chen. Invertible image signal processing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6287–6296, 2021. 1
- [68] Xiangyu Xu, Yongrui Ma, and Wenxiu Sun. Towards real scene super-resolution with raw images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1723–1731, 2019. 1, 2
- [69] Xiangyu Xu, Yongrui Ma, Wenxiu Sun, and Ming-Hsuan Yang. Exploiting raw images for real-scene super-resolution. *IEEE transactions on pattern analysis and machine intelligence*, 44(4):1905–1921, 2020. 2
- [70] Kangning Yang, Jie Cai, Ling Ouyang, Florin-Alexandru Vasluianu, Radu Timofte, Jiaming Ding, Huiming Sun, Lan Fu, Jinlong Li, Chiu Man Ho, Zibo Meng, et al. NTIRE 2025 challenge on single image reflection removal in the wild: Datasets, methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. 2
- [71] Kihwan Yoon, Ganzorig Gankhuyag, Jinman Park, Haengseon Son, and Kyoungwon Min. Casr: Efficient cascade network structure with channel aligned method for 4k real-time single image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7911–7920, 2024. 4

- [72] Weihao Yu, Pan Zhou, Shuicheng Yan, and Xinchao Wang. Inceptionnext: When inception meets convnext. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 5672–5683, 2024. [6](#)
- [73] Pierluigi Zama Ramirez, Fabio Tosi, Luigi Di Stefano, Radu Timofte, Alex Costanzino, Matteo Poggi, Samuele Salti, Stefano Mattoccia, et al. NTIRE 2025 challenge on hr depth from images of specular and transparent surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025. [2](#)
- [74] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *CVPR*, 2021. [17](#)
- [75] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022. [16](#)
- [76] Dafeng Zhang, Feiyu Huang, Shizhuo Liu, Xiaobing Wang, and Zhezhu Jin. Swinfr: Revisiting the swinir with fast fourier convolution and improved training for image super-resolution. *arXiv preprint arXiv:2208.11247*, 2022. [8](#), [15](#)
- [77] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3217–3226, 2020. [18](#)
- [78] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018. [6](#)
- [79] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *European Conference on Computer Vision*, pages 286–301, 2018. [13](#)