DemosaicFormer: Coarse-to-Fine Demosaicing Network for HybridEVS Camera

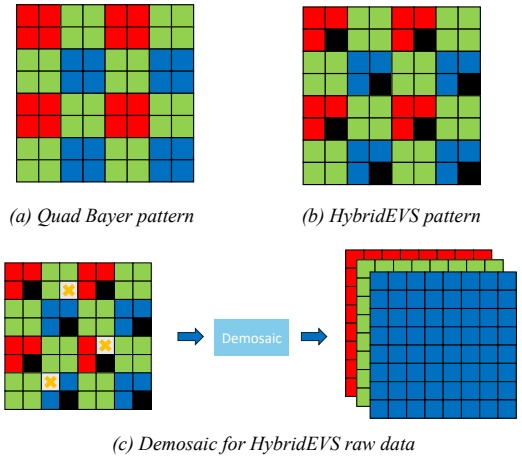
Senyan Xu*, Zhijing Sun*, Jiaying Zhu, Yurui Zhu, Xueyang Fu, Zheng-Jun Zha†
University of Science and Technology of China
{syxu, sunzhijing, zhujy53, zyr}@mail.ustc.edu.cn, {xyfu, zhazj}@ustc.edu.cn

Abstract

Hybrid Event-Based Vision Sensor (HybridEVS) is a novel sensor integrating traditional frame-based and event-based sensors, offering substantial benefits for applications requiring low-light, high dynamic range, and low-latency environments, such as smartphones and wearable devices. Despite its potential, the lack of Image signal processing (ISP) pipeline specifically designed for HybridEVS poses a significant challenge. To address this challenge, in this study, we propose a coarse-to-fine framework named DemosaicFormer which comprises coarse demosaicing and pixel correction. Coarse demosaicing network is designed to produce a preliminary high-quality estimate of the RGB image from the HybridEVS raw data while the pixel correction network enhances the performance of image restoration and mitigates the impact of defective pixels. Our key innovation is the design of a Multi-Scale Gating Module (MSGM) applying the integration of cross-scale features, which allows feature information to flow between different scales. Additionally, the adoption of progressive training and data augmentation strategies further improves model's robustness and effectiveness. Experimental results show superior performance against the existing methods both qualitatively and visually, and our DemosaicFormer achieves the best performance in terms of all the evaluation metrics in the MIPI 2024 challenge on Demosaic for Hybridevs Camera. The code is available at this repository.

1. Introduction

Event-Based Vision Sensor (EVS) detects luminance changes asynchronously and will output event data immediately, which has the advantages of low power consumption and high sensitivity, and is suitable for capturing high dynamic range visual information without blurring. However, the inability to capture color information greatly limits the application scope of event cameras. Hybrid Event-based Vision Sensor (HybridEVS) [11] is a novel hybrid sensor formed by combining traditional frame-based sensor and event-based sensor. It combines the advantages of these sensors, offering high temporal resolution, low latency, and exceptional dynamic range while still capturing color information with higher Signal-to-Noise Ratio (SNR). Compared to traditional sensors, HybridEVS can perform better in a greater range of applications because of its hybrid design. Quad Bayer pattern, as shown in Fig. 1(a) is a common type of pattern widely employed in smartphone cameras due to its ability to obtain high-quality images under low light secnary by averaging four pixels within a 2×2 neighborhood. While signal-to-noise ratio (SNR) is improved in the binning mode, the spatial resolution is reduced as a tradeoff. Defect pixels are flaws caused by the sensor's manufacturing process, where certain pixel values are inaccurate during the photoelectric conversion process.



(a) Quad Bayer pattern *(b) HybridEVS pattern*

(c) Demosaic for HybridEVS raw data

Figure 1. Illustration of two different patterns and demosacing task. (a) Quad Bayer pattern. (b) HybridEVS pattern. (c) Demosaic for HybridEVS Camera task refers to the conversion of HybridEVS pattern raw data into RGB images.

HybridEVS pattern, as shown in Fig. 1(b), is based on Quad Bayer pattern which replaces two normal pixels in the 4×4 pattern by event pixels (represented by black pixels). However, conventional general-purpose methods face chal-

*Co-first authors, †corresponding author.

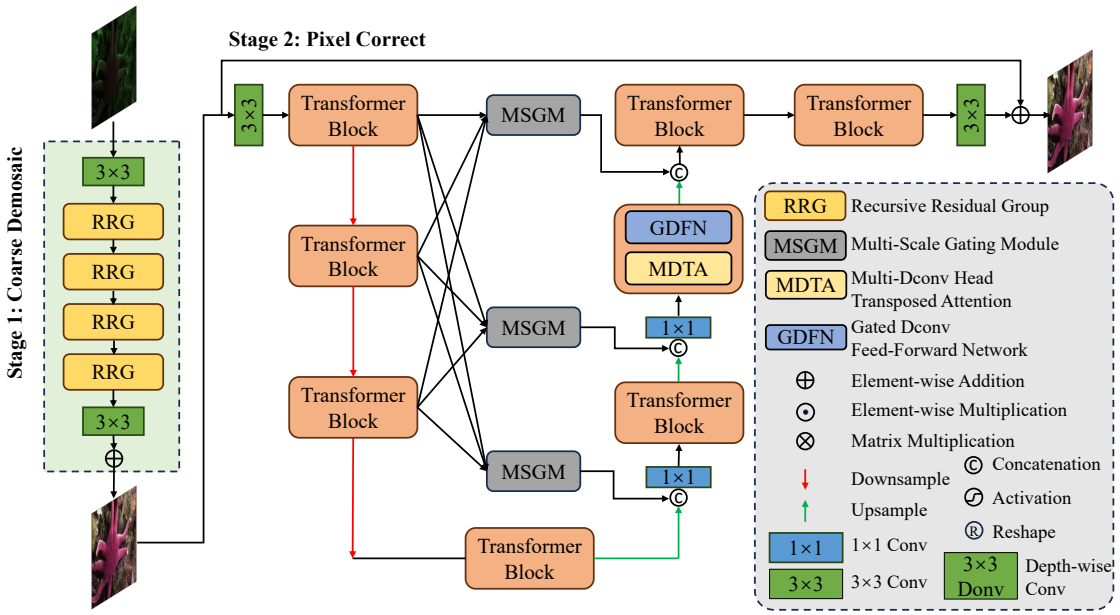


Figure 2. The architecture of our proposed DemosaicFormer to demosaic the raw data captured by HybridEVS cameras.

lenges when demosacing for HybridEVS raw data. Since Quad Bayer pattern sacrifices spatial resolution and event pixels can not record color information, demosaicing for HybridEVS raw data has less spatial and color information than demosaicing for regular raw data. On the other hand, as with any sensor, defect pixels can occur. Therefore, with the HybridEVS pattern, identifying and correcting these pixels is more challenging.

To address this challenge, we propose a coarse-to-fine framework named DemosaicFormer which comprises a coarse demosaicing network and a pixel correction network. For the coarse demosaicing stage, in order to produce a preliminary high-quality estimate of the RGB image from the HybridEVS raw data, we introduce Recursive Residual Group (RRG) [28] which employs multiple Dual Attention Blocks (DABs) to refine the feature representation progressively. For pixel correction stage, aiming to enhance the performance of image restoration and mitigate the impact of defective pixels, we introduce the Transformer Block which consists of Multi-Dconv Head Transposed Attention (MDTA) and Gated-Dconv Feed-Forward Network (GDFN). Our key innovation is the design of a novel Multi-Scale Gating Module (MSGM) applying the integration of cross-scale features, which allows feature information to flow between different scales.The main contributions of our paper are summarized as follows:

- We present a novel coarse-to-fine framework (called DemosaicFormer) to demosaic for HybridEVS raw images with defect pixels which decomposes the task into two sub-tasks: coarse demosaicing and pixel correction.
- We devise the Multi-Scale Gating Module (MSGM) to enhance the network by improving the interaction of feature information flow among different scales.
- Experimental results show that the proposed method significantly outperforms other exited solutions. In the MIPI-challenge 2024 Demosaic for HybridEVS Camera track, our DemosaicFormer achieves first place in terms of all the evaluation scores (PSNR, SSIM) and outperforms the others by a large margin.

2. Related Work

2.1. Image Signal Processing Pipeline

Image signal processing (ISP) pipeline is a series of processing steps in digital image processing that are used to convert raw images obtained from cameras or other image acquisition devices into final usable images. This pipeline typically consists of multiple stages, each performing specific image processing tasks to improve image quality, enhance specific image features, or prepare the image for subsequent processing or display. ISP includes a series of processing algorithms that process raw images to obtain RGB images, such as demosaic, denoising, gamma correction, etc. With the development of deep neural networks (DNN), many studies [9, 13, 16] use DNN to directly replace the main processing flow of ISP and convert raw images into RGB images end-to-end. CycleISP [28] uses a cyclic approach to construct a noise data set of real scenes, modeling the camera imaging pipeline in both forward (RGB2RAW) and reverse (RAW2RGB) directions.

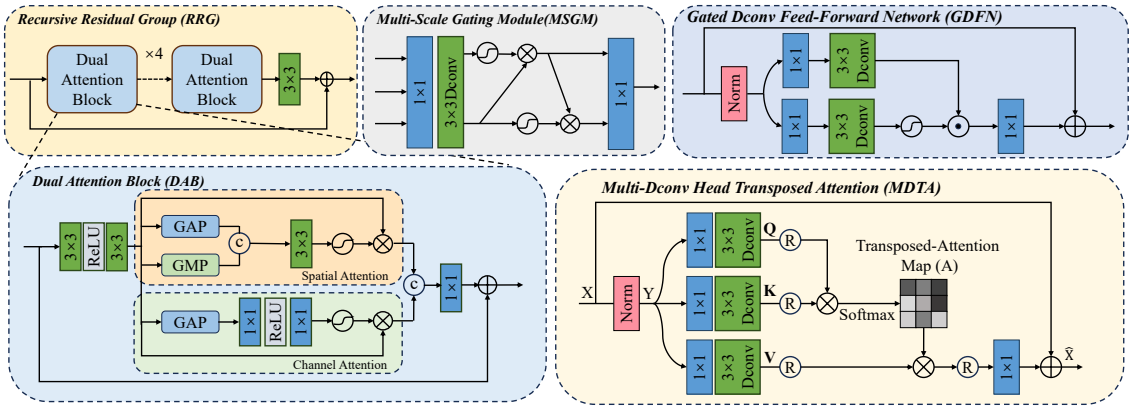


Figure 3. The structures of sub-modules in the main architecture.

2.2. Deep Learning for Image Restoration

Image restoration aims to recover its clean counterparts from a degraded image. A popular scheme is to use CNN structures to learn efficient models to capture local features of images and learn generalizable image priors. CNNs have been widely used in various image restoration tasks, including image denoising [3, 26], demosaicing [14, 31], and super-resolution [19, 32]. Chen et al. [2] used multiplication to replace or delete unnecessary activation functions such as Sigmoid, ReLU, GELU, and Softmax, and derived a nonlinear activation free network called NAFNet. Zhu et al. [36] proposed ECFNet to effectively restore UDC images which takes multi-scale images as input. MIRNet [27] is a novel architecture that learns a rich set of features incorporating contextual information from multiple scales while maintaining high resolution.

After the Transformer model shined in the field of natural language processing, Vision Transformer (ViT) [5] has also been extensively explored in high-level visual tasks, such as object detection [1, 35], image segmentation [24, 33], etc. Transformer has the ability to capture long-range dependencies between image patches and adapt to given input content. Due to these characteristics, Transformer is also used in the field of image restoration [12, 21, 34]. ShuffleFormer [23] proposes a local window Transformer based on a random shuffling strategy to model non-local interactions with linear complexity. Restormer [29] proposes an efficient Transformer-based model.

2.3. HybridEVS Visions

Event-Based Vision Sensor Camera has the advantages of low power consumption and high sensitivity, and is suitable for capturing high dynamic range visual information without blurring. There have been related works using Deep Neural Network (DNN) with RGB and event data for effective image enhancement (such as deblurring and video frame interpolation) [10, 17]. But these image processing techniques require equivalent RGB characteristics to advanced mobile RGB sensors, as well as alignment of focus between RGB and event pixels on the sensor. Based on this, Kodama et al. [11] proposed the Hybrid Event-Based Vision Sensor, which can achieve image enhancement of mixed data in mobile application processors. However, the manufacturing process of the sensor will cause defects, and there will also be some inaccurate pixel values during the photoelectric conversion process, resulting in the appearance of defective pixels. Currently, the reconstruction of HybridEVS raw data containing event pixels and defective pixels into RGB images is less explored.

3. Method

Our proposed DemosaicFormer is built with two-stage cascade framework, which gradually generates desired high-quality results for Hybridevs camera in a coarse-to-fine manner. As shown in Fig. 2, the proposed framework consists of coarse demosaicing and pixel correction network, which is based on the CycleISP [28] and Restormer [29] respectively. Different from these approaches, our two-stage framework can decompose the complex task into individual sub-tasks which can increase each network's learning ability and make the whole framework easier to converge. Furthermore, we devise the Multi-Scale Gating Module (MSGM) to transfer the feature information flow among the Transformer Blocks of cross scales. Following this, we present detailed explanations of our pipeline and the key components encompassed within proposed approach.

3.1. Overall Pipeline

In some learning ISP methods [9, 13, 16], defect pixel removal and demosaicing are often implemented in one stage due to the relatives between them. So we first feed the original raw image into the coarse demosaicing network to get

Table 1. Quantitative comparisons of methods on the official testing datasets of the MIPI-challenge 2024 Demosaic for Hybridevs Camera track. The best and the second results are boldfaced and underlined, respectively.

Rank	Methods	Metrics	
		PSNR↑	SSIM↑
1	DemosaicFormer(Ours)	**44.8464**	**0.9854**
2	2nd	<u>44.6234</u>	<u>0.9847</u>
3	3rd	44.4950	0.9845
4	4th	43.9564	0.9837
5	5th	42.6508	0.9810
6	6th	41.3279	0.9780
7	7th	41.0737	0.9752

an imperfect image in RGB space. Then, the RGB image will go through the pixel correction network which gradually restores the corrupted RGB image in a coarse to fine manner. The second stage finally outputs a desired high-quality RGB image.

In detail, for coarse demosaicing stage, given a HybridEVS raw image of $\mathbf{I}_{raw} \in \mathbb{R}^{H\times W\times 1}$, we extend it to RGB space $\mathbf{I}_{raw}^{RGB} \in \mathbb{R}^{H\times W\times 3}$, a coarse demosaicing network noted as $\mathbf{F}_{cd}$ is employed to simply eliminate the defect pixels and restore the raw image to RGB space $\mathbf{I}_{rest}^{RGB}$.

$$\mathbf{I}_{rest}^{RGB} = F_{cd}(Extend(\mathbf{I}_{raw})) \quad (1)$$

After that, $\mathbf{I}_{rest}^{RGB}$ is taken as the input of pixel correction stage and a pixel correction network noted as $\mathbf{F}_{pc}$ is adopted to correct pixel and refine the imperfect image.

$$\mathbf{I}_{output}^{RGB} = F_{pc}(\mathbf{I}_{rest}^{RGB}) \quad (2)$$

Finally, we get the desired images $\mathbf{I}_{output}^{RGB} \in \mathbb{R}^{H\times W\times 3}$. The whole two-stage framework can be formulated as:

$$\mathbf{I}_{output}^{RGB} = F_{pc}(F_{cd}(Extend(\mathbf{I}_{raw}), \theta_{cd}), \theta_{pc}) \quad (3)$$

Here θ_{cd}, θ_{pc} denote the learnable parameters in $\mathbf{F}_{cd}$ and $\mathbf{F}_{pc}$. By decomposing complex demosaic tasks, our DomisaicFormer achieves outstanding results.

3.2. Coarse Demosaicing Network

The Coarse Demosaicing Network aims to produce a preliminary high-quality estimate of the RGB image from the raw data. Inspired by [6, 15, 30], we introduce Recursive residual group (RRG) [28] which employs multiple Dual Attention Blocks (DABs) to refine the feature representation progressively. As shown in Fig. 3, the DAB is a comprehensive attention unit within the RRG that utilizes both spatial[22] and channel[8] attention mechanisms. The overall process of DAB is:

$$T_{DAB} = T_{in} + Conv(Concat([CA(U), SA(U)])) \quad (4)$$

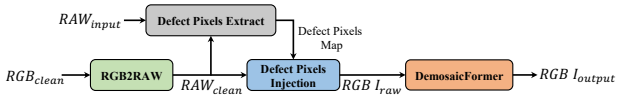


Figure 4. Train model using synthesized augmented data.

where $U \in \mathbb{R}^{H\times W\times C}$ denotes tensors of features maps obtained by applying two 3×3 conv layers on input tensor $T_{in} \in \mathbb{R}^{H\times W\times C}$, $Conv(\cdot)$ is the last 1×1 conv layer.

3.3. Pixel Correction Network

The output of coarse demosacing stage still suffers from the impact of defect pixels because the first stage can't perfectly tackle joint demosaic and defect pixels removal tasks. Pixel Correction Network is aimed to enhance the performance of image restoration and mitigate the impact of defective pixels. Existing CNN-based image restoration methods have achieved impressive results [2, 4, 27, 36]. However, these approaches exhibit shortcomings in capturing long-range dependencies and non-local similarities. In contrast, Transformer methods have shown exceptional ability over the past few years with great performance. However, directly applying a conventional Transformer has more computational overhead which comes from the self-attention layer. Moreover, regular Transformer architectures always overlook the integration of cross-scale features, which is crucial for effective image restoration. To address this problem, inspired by [29], we introduce the Transformer Block which consists of Multi-Dconv Head Transposed Attention (MDTA), Gated-Dconv Feed-Forward Network (GDFN) and Multi-Scale Gating Module (MSGM).

Multi-Dconv Head Transposed Attention(MDTA), shown in Fig. 3 has linear complexity implemented by applying conventional SA [18] across channels dimension which is the key design of MDTA. As another important component of MDTA, depth-wise convolutions generate the global attention map emphasizing on the local context before computing attention.

Gated-Dconv Feed-Forward Network(GDFN), shown in Fig. 3, is utilised to transform features after MDTA, which is different from the regular feed-forward network(FN)[5]. To improve representation learning, gating mechanism and depthwise convolutions are applied in GDFN.The gating mechanism is structured as the Hadamard product (element-wise multiplication) of two parallel pathways consisting of linear transformation layers. Similar to MDTA, all pathways include 3×3 depth-wise convolutions to encode information from spatially neighboring pixel positions, useful for learning local image structure for effective restoration. One of these pathways is activated with the Gaussian Error Linear Unit (GELU)[7] .

Table 2. Quantitative comparisons of methods on the official validation datasets of the MIPI-challenge 2024 Demosaic for Hybridevs Camera track. The MACs and FLOPs is computed using a 128×128 image as input by calflops tool. The best and the second results are boldfaced and underlined, respectively.

Methods	**#Params (M)**	**MACs (G)**	**FLOPs (G)**	Metrics	
				PSNR↑	**SSIM↑**
CycleISP[28]	2.8	46.0	93.4	41.32	0.98
MIMO-UNet[4]	8.9	21.1	41.2	40.75	0.98
MIMO-UNet*[4]	8.9	21.7	41.7	41.27	0.98
ECFNet[36]	9.1	21.7	42.5	41.45	0.98
NAFNet[2]	67.8	15.8	31.6	41.19	0.98
MIRNet[27]	5.9	34.9	70.0	40.92	0.98
Restormer[29]	26.1	35.2	70.6	<u>41.73</u>	<u>0.98</u>
ShuffleFormer[23]	50.6	20.7	41.6	41.70	0.98
DemosaicFormer(Ours)	30.3	85.1	171.5	**42.01**	**0.98**

Multi-Scale Gating Module (MSGM) Inspired by ResNet, some methods supplement the original features in the encoder to the decoder through skip connection. This can reduce the difficulty of network optimization and improve network performance. In some cases, features are even transferred across scales, feeding features from the encoder into different scales of the decoder. In this paper, inspired by NAFNet [2], we furthermore introduce a simple gating mechanism into cross-scale feature fusion increasing the nonlinearity of fusion. Based on the gating mechanism, we can extract the features needed by different scale decoders which improves the correction effect of the network. Specifically, as shown in Fig. 3, our Multi-Scale Gating Module (MCGM) up-samples or down-samples the features at different scales according to the required shape of the module output, then concatenates them at the channel dimension and adjusts the number of channels using 1×1 convolution. Inspired by the simple gate in NAFNet, we divide them into two equal parts for 3×3 depth-wise convolution. Each feature is multiplied by the sigmoid change of the other feature, and finally the two parts of the features are transformed into the required enhancement features using 1×1 convolution. Formally, the MCGM at the shallowest scale can be presented as

$$\begin{aligned} F &= Conv(Concat([(TB_1^{out}), (TB_2^{out})^{\uparrow}, (TB_3^{out})^{\uparrow}])), \\ F_1, F_2 &= Split(DConv(F)), \\ F_1 &= Sigmoid(F_1) \times F_2, \\ F_2 &= Sigmoid(F_2) \times F_1, \\ F_e &= Conv(Concat([F_1, F_2])), \end{aligned} \tag{5}$$

where $TB_i^{out}, i = 1, 2, 3$ denotes the output of the n^{th} scale transformer block, $\uparrow$ denotes the up-sampling operation, $Concat(\cdot)$ denotes the concatenation operation along the channel dimension, $Dconv(\cdot)$ denotes the depth-wise convolution, $Split(\cdot)$ denotes the chunk operation.

Table 3. Quantitative comparisons of different training objects. The best result is boldfaced.

Model	**Training Description**	**PSNR↑**
A	Indiv. Train. & Joint FT	41.99
B	Joint Train. w. Ext. Sup.	40.76
C	Joint Train. (default)	**42.01**

3.4. Joint Training of DemosaicFormer

Given the intrinsic interdependence of coarse demosaicing and pixel correction, it is impractical to disentangle them completely into separate subtasks. Hence, in our DemosaicFormer, we employ a joint training approach which will be discussed in Section 4.4. We utilize ℓ_1 loss, which is widely used in many image restoration and enhancement tasks[2, 13, 27–29, 36]. The loss function for the joint optimization is:

$$\mathcal{L}_1(I, \hat{I}) = \frac{1}{N} \sum_{p \in P} |I_(p) - \hat{I}_(p)|, \tag{6}$$

where p is the index of the pixel and P is the patch; I and $\hat{I}$ represent the ground-truth and restored result by our DemosaicFormer with N pixels, respectively.

4. Experiments

4.1. Dataset

We conduct the experiments strictly following the setting of the MIPI-challenge 2024 Demosaic for Hybridevs Camera track[25]. The training data consists of 800 pairs of Hybridevs's input data and label result with a resolution of 2K. Both the input and label have the same spatial resolution.

Table 4. Ablation study of the Training Strategies. The best and the second results are boldfaced and underlined, respectively.

	Progressive Training	Data Augmentation	Finetune Stage	PSNR↑	
				VAL SET	TEST SET
DemosaicFormer				42.01	-
DemosaicFormer		50% Prob.		42.39	42.61
DemosaicFormer		✓		43.10	42.54
DemosaicFormer-s1	✓	✓		<u>43.17</u>	<u>42.63</u>
DemosaicFormer-s2	✓	✓	✓	**43.26**	**42.98**

Table 5. Ablation study of the connection manner in the different level. The best and the second results are boldfaced and underlined, respectively.

Level	Connection Manner	PSNR↑
Arch	Pixel Correct First	<u>42.92</u>
	Coarse Demosaic First(default)	**43.10**
	Parallel Connection	42.85
Block	Simple Concatenation	42.93
	Single Gating Fusion	<u>42.99</u>
	Multi-Scale Gating Module(default)	**43.10**

The input is of 10bits in the ".bin" format and ranges from [0, 1023], and the corresponding ground truth is of 8bits in the ".png" format. The validation and testing sets consist of 50 images each, and each set contains images of varying resolutions. In the testing set, the resolution of images is not fixed, ranging from 1280×720 to 5760×5760. Note that the ground truth data corresponding to the validation and testing dataset is not publicly available.

Data augmentation. Due to our inability to accurately model defective pixels, inspired by [28], we extract the defect pixels map from the training data of the challenge to generate more diverse and realistic inputs. As shown in Fig. 4, at the training phase, we randomly rotate and flip ground-truth images(RGB_{clean}) of training split, then sample them according to HybridEVS pattern, and randomly cover the sampled images with defect pixels map. The augmentation technology is applied at the initial training of our proposed approach for improving the model's generalization and robustness. Note that the models trained with different data augmentation strategies are different, as seeing in the section 4.2.

4.2. Implementation Details

We implement our proposed network via the PyTorch 1.8 platform. Adam optimizer with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.99$ is adopted to optimize our network. Additionally, motivated by [29], we introduce the progressive training strategy. The training phase of our network could be divided into two stages:

(1) **Initial training of DemosaicFormer**. We use a progressive training strategy at first. We start training with patch size 80×80 and batch size 84 for 58K iterations. The patch size and batch size pairs are updated to $[(128^2, 30), (160^2, 18), (192^2, 12)]$ at iterations [36K, 24K, 24K]. The initial learning rate is 5×10^{-4} and remains unchanged when patch size is 80. Later the learning rate changes with Cosine Annealing scheme to 1×10^{-7}. For data augmentation, we use our data augmentation mentioned above. The first stage is performed on the NVIDIA 4090 device. We obtain the best model at this stage as the initialization of the second stage.

(2) **Fine-tuning DemosaicFormer**.We start training with patch size 192×192 and batch size 12. The initial learning rate is 1×10^{-4} and changes with Cosine Annealing scheme to 1×10^{-7}, including 20K iterations in total. We use the entire training data from the challenge without any data augmentation technologies. Exponential Moving Average (EMA) is applied for the dynamic adjustment of model parameters. The second stage is performed on the NVIDIA 4090 device.

To better distinguish between the model results, we label the two stages as DemosaicFormer-s1 and DemosaicFormer-s2, respectively.

4.3. Evaluation Metrics

We employ two reference-based metrics which are widely applied in similar tasks[2, 4, 12, 13, 27, 29], to assess the efficacy of our method: Peak Signal-to-Noise Ratio (PSNR), the structural similarity (SSIM) [20]. Higher values of PSNR and SSIM indicate better performance in image restoration tasks. Note that due to the evaluation settings of the challenge, we are unable to obtain the exact SSIM value, but it does not affect the ordering of SSIM.

4.4. Comparations

Table 1 presents a comprehensive comparison of various solutions on the MIPI-challenge 2024 Demosaic for Hybridevs Camera track. Evidently, our approach outperforms all others across all evaluation metrics on the official testing datasets, showcasing superior performance. Specifically, our method achieves a remarkable improvement, surpass-

Figure 5. Visual comparison results of Demosaic for Hybridevs Camera on the evaluation dataset of MIPI-challenge 2024 track. Note that brighter means bigger error.

ing the second-place method by 0.2230 dB in PSNR.

Besides, in Table 2, we demonstrate comparable performance methods on the official validation datasets when compared to some ISP methods and general image restoration methods. For a fair comparison, note that all methods utilize HybridEVS's raw data expanded into RGB space as input without any data augmentation techniques. Our method consistently demonstrates outstanding performance. Compared to the method Restormer and ShuffleFormer, we obtain 0.28dB and 0.31dB gain in PSNR. Furthermore, in Fig. 5,6, to more intuitively show our excellent performance, we generate the residual map representing the disparity between the predicted output and the ground truth. The comparison clearly demonstrates that our technology produces superior visual results and outperforms others in terms of visual quality. Especially, our method reconstructs finer details more effectively and shows less departure from the ground truth, demonstrating its efficiency in image restoration.

For training objects, Table 3 presents the results of employing various training objects for DemosaicFormer. Model A is two-phase training procedure where Coarse Demosaic Network is initially trained to convert raw data into RGB images, followed by joint finetuning, which extends training duration. Model B denotes joint training with extra constraint loss at Coarse Demosaic stage. Model C, in contrast, represents joint training devoid of any extra constraints. The comparison clearly demonstrates that directly joint training can produce better results with less time durtation. Specifically, the model is encouraged to jointly optimize both the demosaicing task and any auxiliary tasks, thereby leveraging the interconnectedness inherent between two stages.

4.5. Ablation Study

We conduct plenty of ablation experiments to verify the effect of each component of our method. Note that in the ablation study with the absence of other annotations, we train the model with our data augmentation technology and without progressive training manner for convenience.

Effects of the Connection Manner. As shown in Table 5, we verify the validity of the DemosaicFormer connection manner at different levels, including the sequential choice of the two-stage network(arch-level) and the effectiveness

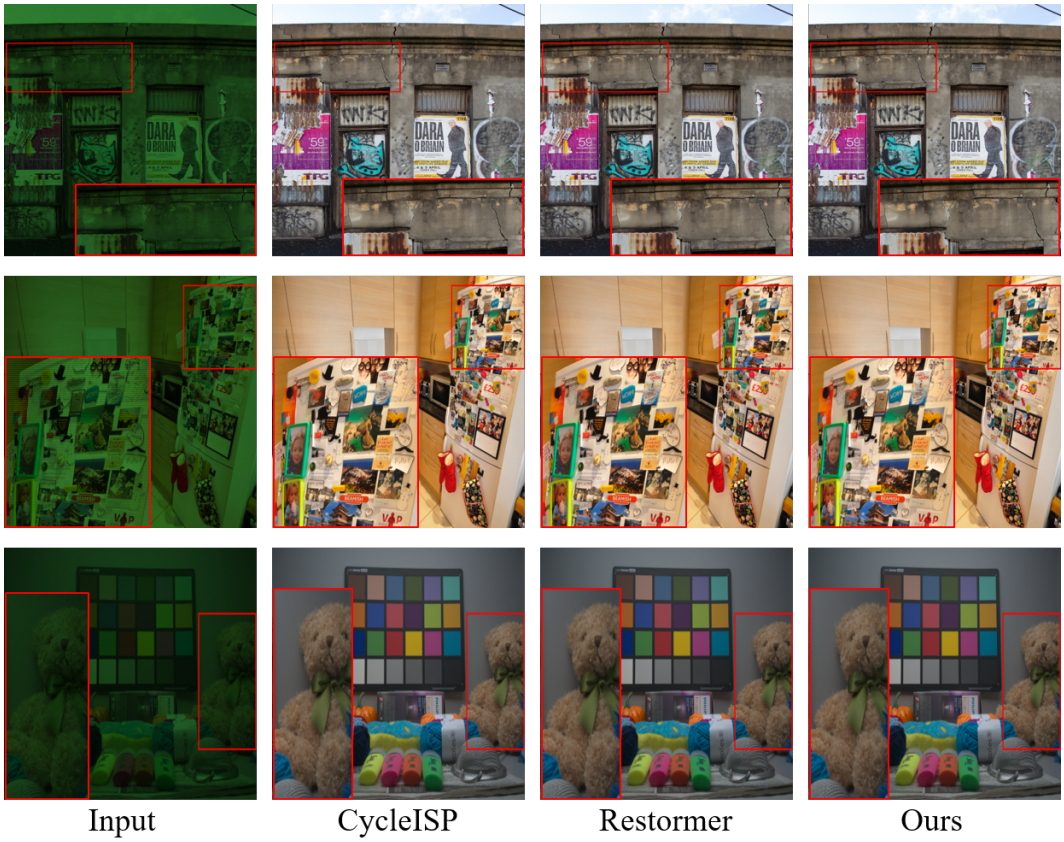


Figure 6. Visual comparison results of Demosaic for Hybridevs Camera on the testing dataset of MIPI-challenge 2024 track.

of the MSGM module(block-level). In connection manner of arch-level, we compare the performance of the model using different two-stage connection approaches which include exchanging the order of coarse demosaicing and pixel correction and processing the two branches in parallel. It is evident that coarse demosaicing before the pixel correction results in significant performance gains. Because of the sparsity nature of defect pixels, the initial demosaicing process is not significantly affected, while also providing more detailed color information for the post-processing. Parallel processing causes degraded performance by disrupting the progressive processing flow created by cascading.

Furthermore, in connection manner of block-level, effectiveness of the MSGM module is verified by replacing it with Simple Concatentation and Single Gating Fusion. The MSGM module incorporates multi-scale feature information and adaptively selects features based on the hierarchy of the output, obtaining 0.17dB gain in PSNR.

Effects of the Training Strategies. Following [28, 29], we additionally adopt the progressive training strategy, various data augmentation strategies and fine-tune model to enhance the model performance. As shown in Table 4, the models trained at different stages are marked as DemosaicFormer-s1 and DemosaicFormer-s2. Experiments show that training with progressively larger patches often results in higher gains in generalization performance. Our data augmentation technology greatly improves model's performance by increasing generalization and robustness. After initial training with progressive learning and data augmentation, fine-tuning the model on the original training set facilitates better adaptation to the real data distribution, obtaining 0.09dB and 0.35dB gain in PSNR on challenge official val set and test set, respectively.

5. Conclusion

In this paper, we present DemosaicFormer, an effective coarse-to-fine network for demosaicing HybridEVS's raw data. Built with a two-stage cascade framework comprising coarse demosaicing and pixel correction networks, DemosaicFormer decomposes the complex task into sub-tasks, and formulates Multi-Scale Gating Module(MSGM). Besides, the adoption of progressive training and data augmentation strategies further improves the model's robustness and effectiveness. DemosaicFormer achieves the best performance in terms of all the evaluation metrics in the MIPI-challenge 2024 Demosaic for Hybridevs Camera track.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 3

[2] 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, 5, 6

[3] Shen Cheng, Yuzhi Wang, Haibin Huang, Donghao Liu, Haoqiang Fan, and Shuaicheng Liu. Nbnet: Noise basis learning for image denoising with subspace projection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4896–4906, 2021. 3

[4] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4641–4650, 2021. 4, 5, 6

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3, 4

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[7] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 4

[8] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 4

[9] 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. 2, 3

[10] Zhe Jiang, Yu Zhang, Dongqing Zou, Jimmy Ren, Jiancheng Lv, and Yebin Liu. Learning event-based motion deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3320–3329, 2020. 3

[11] Kazutoshi Kodama, Yusuke Sato, Yuhi Yorikado, Raphael Berner, Kyoji Mizoguchi, Takahiro Miyazaki, Masahiro Tsukamoto, Yoshihisa Matoba, Hirotaka Shinozaki, Atsumi Niwa, et al. 1.22 μm 35.6 mpixel rgb hybrid event-based vision sensor with 4.88 μm-pitch event pixels and up to 10k event frame rate by adaptive control on event sparsity. In *2023 IEEE International Solid-State Circuits Conference (ISSCC)*, pages 92–94. IEEE, 2023. 1, 3

[12] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844, 2021. 3, 6

[13] Zhetong Liang, Jianrui Cai, Zisheng Cao, and Lei Zhang. Cameranet: A two-stage framework for effective camera isp learning. *IEEE Transactions on Image Processing*, 30:2248–2262, 2021. 2, 3, 5, 6

[14] Bolin Liu, Xiao Shu, and Xiaolin Wu. Demoir\'eing of camera-captured screen images using deep convolutional neural network. *arXiv preprint arXiv:1804.03809*, 2018. 3

[15] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. Progressive image deraining networks: A better and simpler baseline. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3937–3946, 2019. 4

[16] Eli Schwartz, Raja Giryes, and Alex M Bronstein. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE Transactions on Image Processing*, 28(2):912–923, 2018. 2, 3

[17] Stepan Tulyakov, Alfredo Bochicchio, Daniel Gehrig, Stamatios Georgoulis, Yuanyou Li, and Davide Scaramuzza. Time lens++: Event-based frame interpolation with parametric non-linear flow and multi-scale fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17755–17764, 2022. 3

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[19] Longguang Wang, Yingqian Wang, Xiaoyu Dong, Qingyu Xu, Jungang Yang, Wei An, and Yulan Guo. Unsupervised degradation representation learning for blind super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10581–10590, 2021. 3

[20] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

[21] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17683–17693, 2022. 3

[22] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 4

[23] Jie Xiao, Xueyang Fu, Man Zhou, Hongjian Liu, and Zheng-Jun Zha. Random shuffle transformer for image restoration. In *International Conference on Machine Learning*, pages 38039–38058. PMLR, 2023. 3, 5

[24] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090, 2021. 3

[25] Wu Yaqi, Fan Zhihao, Chu Xiaofeng, Ren Jimmy S., Li Xiaoming, Yue Zongsheng, Li Chongyi, Zhou Shangcheng, Feng Ruicheng, Dai Yuekun, Yang Peiqing,

Loy Chen Change, et al. Mipi 2024 challenge on demosaic for hybridevs camera: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 5

[26] Zongsheng Yue, Qian Zhao, Lei Zhang, and Deyu Meng. Dual adversarial network: Toward real-world noise removal and noise generation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 41–58. Springer, 2020. 3

[27] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Learning enriched features for real image restoration and enhancement. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 492–511. Springer, 2020. 3, 4, 5, 6

[28] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Cycleisp: Real image restoration via improved data synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2696–2705, 2020. 2, 3, 4, 5, 6, 8

[29] 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. 3, 4, 5, 6, 8

[30] He Zhang, Vishwanath Sindagi, and Vishal M Patel. Multi-scale single image dehazing using perceptual pyramid deep network. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 902–911, 2018. 4

[31] Tao Zhang, Ying Fu, and Cheng Li. Deep spatial adaptive network for real image demosaicing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3326–3334, 2022. 3

[32] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018. 3

[33] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021. 3

[34] Yupeng Zhou, Zhen Li, Chun-Le Guo, Song Bai, Ming-Ming Cheng, and Qibin Hou. Srformer: Permuted self-attention for single image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12780–12791, 2023. 3

[35] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 3

[36] Yurui Zhu, Xi Wang, Xueyang Fu, and Xiaowei Hu. Enhanced coarse-to-fine network for image restoration from under-display cameras. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*, pages 130–146. Springer, 2023. 3, 4, 5