

Efficient Unified Demosaicing for Bayer and Non-Bayer Patterned Image Sensors

Haechang Lee^{1,4,*}, Dongwon Park^{2,*}, Wongi Jeong^{1,*},
 Kijeong Kim⁴, Hyunwoo Je⁴, Dongil Ryu⁴ and Se Young Chun^{1,2,3,†}
¹Dept. of ECE, ²INMC, ³IPAI, Seoul National University, Republic of Korea,
⁴SK hynix, Republic of Korea
 {harrylee, donglpark, wg7139, sychun}@snu.ac.kr

Abstract

As the physical size of recent CMOS image sensors (CIS) gets smaller, the latest mobile cameras are adopting unique non-Bayer color filter array (CFA) patterns (e.g., Quad, Nona, $Q \times Q$), which consist of homogeneous color units with adjacent pixels. These non-Bayer sensors are superior to conventional Bayer CFA thanks to their changeable pixel-bin sizes for different light conditions, but may introduce visual artifacts during demosaicing due to their inherent pixel pattern structures and sensor hardware characteristics. Previous demosaicing methods have primarily focused on Bayer CFA, necessitating distinct reconstruction methods for non-Bayer patterned CIS with various CFA modes under different lighting conditions. In this work, we propose an efficient unified demosaicing method that can be applied to both conventional Bayer RAW and various non-Bayer CFAs' RAW data in different operation modes. Our Knowledge Learning-based demosaicing model for Adaptive Patterns, namely KLAP, utilizes CFA-adaptive filters for only 1% key filters in the network for each CFA, but still manages to effectively demosaic all the CFAs, yielding comparable performance to the large-scale models. Furthermore, by employing meta-learning during inference (KLAP-M), our model is able to eliminate unknown sensor-generic artifacts in real RAW data, effectively bridging the gap between synthetic images and real sensor RAW. Our KLAP and KLAP-M methods achieved state-of-the-art demosaicing performance in both synthetic and real RAW data of Bayer and non-Bayer CFAs.

1. Introduction

Demosaicing (DM) is the process of interpolating single-channel input images into RGB output images within an embedded Image Signal Processor (ISP). With the growing demand for high-quality mobile camera images, CMOS image sensor (CIS) resolution has increased dramatically,

even reaching 200 million pixels in the latest smartphones. However, as image sensors cannot infinitely increase in size, pixel size has been reduced to enhance resolution. Smaller CISs are more vulnerable to noise and degradation in image restoration capabilities because they are more sensitive to variations in light reception, especially in low-light condition [14, 28, 45, 46]. As a result, modern high-end smartphones have started using image sensors that group adjacent homogeneous pixels, resulting in non-Bayer Quad, Nona, and Quad-by-Quad ($Q \times Q$) sensors [24, 45, 48], while still retaining some of the properties of the standard Bayer CFA [5] pattern. Quad, Nona, and $Q \times Q$ sensors combine the same color pixel arrays of 2×2 , 3×3 , and 4×4 respectively, resulting in homogeneous pixel units (i.e., Gr, R, B, and Gb) for each sensor, as shown in Fig. 1(a).

Demosaicing for modern non-Bayer CFAs is more complex and computationally demanding than for standard Bayer CFAs. This is because as the number of pixel arrays within each unit increases, the distance between the units becomes greater, requiring interpolation with inaccurate pixel values from distant locations. Therefore, there is growing interest in using deep learning for demosaicing methods, leading to active research on both Bayer pattern demosaicing [78, 51, 13, 8, 68, 1, 40, 76, 57, 23, 31, 22] and non-Bayer pattern demosaicing [33, 32, 25, 3, 58, 11].

However, the aforementioned methods focus on a single CFA pattern task and do not cover demosaicing tasks for other CFA patterns. Modern mobile phones with non-Bayer patterned CIS adapt their CFA modes dynamically based on lighting conditions, controlled by the CIS's ISP. Using independent models (IMs) for each pattern, tailored to different CFA modes, would demand loading and operating multiple models within the limited circuit space of the CIS. This would result in excessive memory and power consumption if the models were kept standby on the mobile application processor (AP) and switched accordingly. Moreover, the task of tuning models for each CFA would be laborious.

Currently, no existing method can handle dynamically changing CFA modes in a non-Bayer patterned CIS as a

* Equal contribution, † Corresponding author.

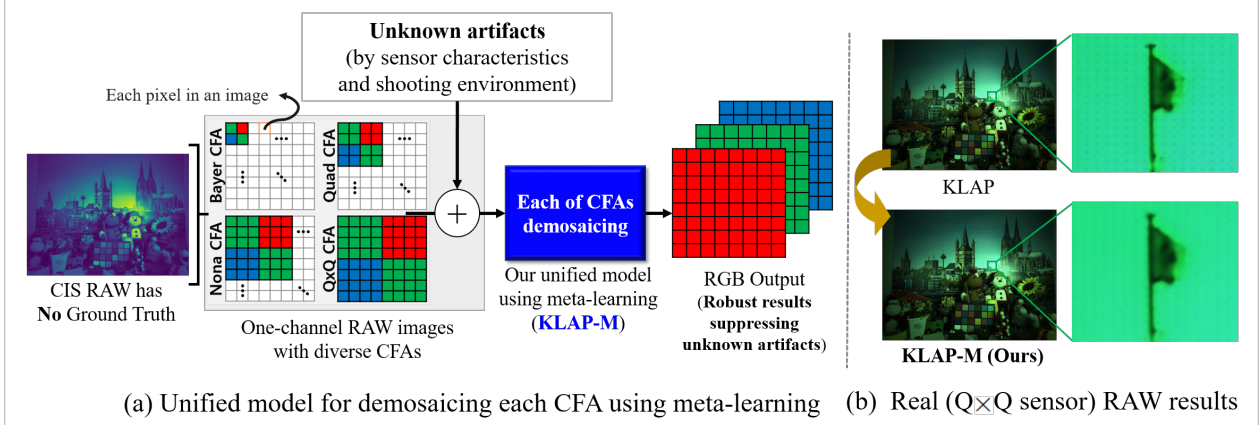


Figure 1: (a) Overview of our unified model (UM) for demosaicing all the Bayer and non-Bayer CFAs, called the **Knowledge Learning-based demosaicing model for Adaptive Patterns using Meta-test learning (KLAP-M)** with Bayer or non-Bayer patterns, even when ground truth is unavailable and unknown artifacts are present. (b) Comparing CIS RAW demosaicing results of KLAP (KLAP-M without meta-test learning) and KLAP-M (KLAP with meta-test learning).

unified model (UM). Inspired by recent works for all-in-one image restoration affected by multiple types of unknown degradation [37, 10, 36], we propose a unified demosaicing method for all Bayer and non-Bayer CFA patterns. However, these all-in-one image restoration methods do not consider real unknown artifacts, so we will further investigate them to address the scenario of real CIS RAW with ‘unknown’ artifacts, missing or mostly lacking ground truth (GT). Since such unknown artifacts may fail to yield high-quality phone camera photos, we are motivated to propose a UM with robust meta-learning-based DM methods that can handle these obstacles.

In this work, we propose efficient unified DM methods that are capable of handling various non-Bayer patterned CISs with a new pipeline to bridge the gap between synthetic and real CIS RAW images. Our proposed **Knowledge Learning-based demosaicing model for Adaptive Patterns (KLAP)** is capable of simultaneously handling multiple CFAs’ demosaicing, which consists of two following steps. Firstly, we train a baseline UM using the two-stage knowledge learning (TKL) [10], making it more efficient to find Adaptive Discriminative filters for each specific CFA Pattern (ADP). Secondly, we fine-tune the UM model trained in the first stage using ADP. ADP is a metric that applies FAIG [67] to the update logic of our neural network, allowing us to find a small set of discriminative filters that can be used as independent parameters for specific CFA DM tasks. Lastly, we propose KLAP-M, KLAP (TKL+ADP) with Meta-test learning. KLAP-M integrates self-supervised learning into KLAP to address domain gaps between synthetic RAW and real CIS RAW images caused by unknown artifacts in real-life scenarios. Our proposed meta-test learning for demosaicing consists of pixel binning loss based on CIS domain knowledge and self-supervised denoising techniques. Fig. 1(a) provides an overview of our

KLAP-M approach, which handles both Bayer and Non-Bayer patterns. Additionally, Fig. 1(b) shows the results of our meta-test learning technique, addressing the domain gap in real RAW images.

Our contributions are summarized as follows: (1) Our efficient unified network, KLAP, effectively performs demosaicing for multiple CFAs, (2) KLAP-M, a version of KLAP that incorporates a meta-learning approach, effectively reduces unknown visual artifacts in genuine CIS RAW images that are caused by diverse sensor characteristics and shooting environments, (3) KLAP and KLAP-M achieve state-of-the-art performance on the synthetic benchmark dataset and real CIS RAW samples captured by CIS chips.

2. Related Works

2.1. Deep Learning-based Demosaicing

IMs for DM only. Traditional demosaicing without applying deep learning techniques either apply a fixed DM filter to each pixel without considering other parameters as features or utilize spectral and spatial features available in neighboring pixels to interpolate the unknown pixel as closely as possible to the original [42, 19]. Due to the complexity of various CIS CFAs, traditional methods are cumbersome, leading to an increasing interest in deep learning-based demosaicing models. Stojkovic *et al.* [59] suggested IMs of each Bayer and Quad demosaicing based on CDM-Net [12]. Kim *et al.* [33, 32] applied the duplex pyramid network structure to Quad CFA and Nona CFA, respectively. Sharif *et al.* [58] proposed a GAN-based spatial-asymmetric attention for Nona CFA reconstruction. For $Q \times Q$ CFA, Cho *et al.* [11] proposed an efficient pyramidal network using progressive distillation based on PyNet [23].

Multi-tasks joint with DM. There have been proposals to combine DM methods with other closely related ISP

tasks, such as denoising (DN) and super-resolution (SR). Some [13, 66, 8, 40, 27, 34] proposed convolutional neural networks approach for joint DM and DN to improve the quality of the restored image. Ma *et al.* [41] and Xu *et al.* [71] proposed models for simultaneous DM and SR. Xing *et al.* [68] introduced a multi-task learning approach to jointly address three tasks: DM, DN, and SR. Previous studies mainly concentrate on multi-task approaches for single CFA demosaicing and known noise sources. In contrast, our proposed method introduces a unified model that handles both Bayer and non-Bayer CFAs, incorporating meta-learning to ensure robust performance even in the presence of unknown noise.

2.2. Image Restoration for Multi-tasks

IMs for multi-tasks. Beyond DM tasks, recent papers [77, 43, 75, 65, 64, 54, 9] have introduced various approaches that share a common framework capable of addressing multiple image restoration tasks, including denoising, deblurring, and deraining. While the mentioned IM excels in individual tasks, it necessitates multiple network parameters as multiple networks are needed to handle all the required tasks.

Unified model (UM) for multi-tasks. To overcome the drawbacks of IMs, Chen *et al.* [10] proposed a single UM for two-stage knowledge learning mechanism based on multi-teacher and single student approach for multiple degradations on images that contains rain, haze, and snow. Li *et al.* [36] proposed a single UM using a contrastive-based degraded encoder, called the degradation-guided restoration network (DGRN), which adaptively works with three degradations: rain, noise, and blur. Park *et al.* [47] introduced a single UM equipped with dedicated filters for degradation, achieving remarkable results in rain-noise-blur and rain-snow-haze tasks. To the best of our knowledge, there is currently no reported method that can handle all Bayer and non-Bayer demosaicing tasks using a single unified model.

2.3. Meta-learning-based Image Restoration

For image reconstruction, a large number of samples are usually necessary, but it may not be feasible in many real-world situations. Meta-learning, also known as learn-to-learn, provides a promising solution to the problem of adapting models quickly to new data. This learning method empowers models to achieve efficient task performance even with limited additional incoming data. Finn *et al.* [17] proposed an algorithm for model-agnostic meta-learning that achieved state-of-the-art performance in few-shot learning tasks. Meta-SR [20] enables super-resolution for arbitrary scale factors by applying the Meta-Upscale Module. We propose the use of meta-learning to achieve robust results, even in the presence of unknown artifacts in CIS RAW images.

3. Deep Demosaicing for Each Non-Bayer CFA

3.1. Operating Principles of Non-Bayer Sensors

With the decreasing size of camera sensors, the physical area of light captured by a pixel has been reduced. Consequently, the introduction of non-Bayer sensors allows for capturing more light. In case of $Q \times Q$ as an example, when there is sufficient light, as scenario (3) and (4) in Fig. 2, $Q \times Q$ sensors can handle the entire resolution with Bayer DM (after ‘re-mosaicing’) and direct $Q \times Q$ DM. On the other hand, especially in low-light conditions, $Q \times Q$ CIS pixels have the advantage of using ‘pixel binning’ to enhance their light sensitivity and reduce the noise [80, 74], sacrificing their resolution (but still acceptable), resulting in clear image quality with reduced noise (shown as scenario (1) and (2) in Fig. 2). Pixel binning is the merging of neighboring pixels in an image through summation or averaging in ISP, typically done by the ISP after pixel-readout. Quad DM or Bayer DM methods are specifically required in such cases. Supporting a diverse range of CFA pattern modes remains crucial in non-Bayer patterns. However, employing separate DM networks for each pattern increases network parameters, leading to larger CIS chip area. Multiple DM models necessitate frequent model switching, consuming more memory and power in mobile environment. Our proposed unified DM model handles all non-Bayer sensor patterns, including standard Bayer sensors, providing effective solutions for this issue. It offers flexibility for different CIS product lines and CFA pattern modes, reducing product development time with minimal fine-tuning required for specific product characteristics.

3.2. Data Synthesis for Demosaicing All CFAs

To train input images resembling real CIS RAW, we propose a data synthesis pipeline that generates realistic RAW-

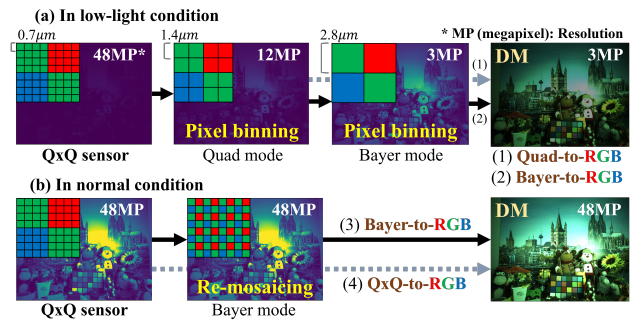


Figure 2: DM scenario in real CIS. For example, in the case of $Q \times Q$ CIS: (a) In low-light conditions, the $Q \times Q$ sensor converts its pattern to either the (1) Quad or (2) Bayer mode (pixel-binning), sacrificing resolution, and then performs DM. (b) In normal conditions, the $Q \times Q$ sensor can either re-mosaic the pattern to the Bayer mode and then perform DM or directly perform $Q \times Q$ DM, with full resolution.

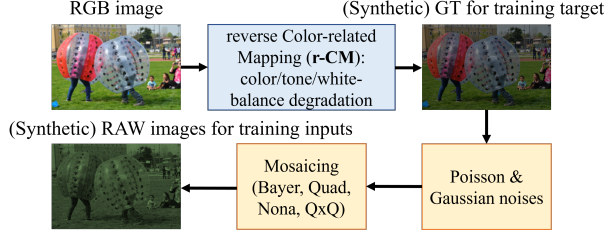


Figure 3: Overview of our realistic RAW image synthesis pipeline for Bayer and Non-Bayer demosaicing. The r-CM (reverse Color-related Mapping functions) towards RAW-like synthesis consists of invertible linear operations that relate RGB color spaces.

like images. Using a high-quality sRGB dataset, we follow the front-end of Fig. 3 to generate synthetic RAW-like images. This involves applying four reverse color-related mapping functions (r-CM) from the ISP chain, including color tone degradation, inverse gamma correction, inverse color correction, and inverse auto white balance correction functions. We analyzed and adjusted the previous ISP chains, resulting in a pipeline structure similar to previous methods. [63, 62, 55, 6, 70]. Using this method, we generate RGB synthetic GT labels for demosaicing training. Furthermore, we add Gaussian and Poisson noise to simulate various types of real noise [55, 14, 6, 70]. Each image is then converted into a mosaic pattern for Bayer, Quad, Nona, and Q×Q CFA, as depicted in the bottom row of Fig. 3. This process generates the training inputs. The reverse color mapping (r-CM) consists of linear operations and can be easily "re-reversed" to obtain the original color mapping (CM). CM makes final output images only after DM that closely resemble human-viewed realistic images. Our proposed synthetic dataset generation pipeline considers demosaicing for both Bayer and Non-Bayer patterns and incorporates a realistic noise model that combines Gaussian and Poisson noise. More detailed information is Sec. S.1 in the supplementary material.

3.3. Domain Gap in Synthetic and Real CIS RAW

Synthetic data-trained models often struggle with real data due to the domain gap issue, a persistent problem in image restoration tasks [55, 6, 26]. The domain gap arises from variations in sensor hardware characteristics due to differences in circuit structure, manufacturing processes, and component variations across CIS brands and product lines. The upper image in Fig. 1(b) shows visual artifacts in real CIS RAW, mainly caused by crosstalk effects [29, 30, 38] between inner and outer pixels (details in Sec. S.2 in the supplementary). Moreover, unknown artifacts can emerge in different shooting environments and vary across CIS types. To address this, we propose a meta-learning method to minimize the domain gap, enabling the

effective handling of unexpected unknown artifacts.

4. Unified Deep Demosaicing for Multiple Bayer and Non-Bayer CFAs

Fig. 4 displays the proposed single unified DM method for all Bayer and non-Bayer sensor patterns (KLAP) and its additional meta-learning during inference framework for robustness (KLAP-M). In Step 1 as Fig. 4(a), our approach augments the network capacity of the integrated model using the Two-stage Knowledge Learning [10] (TKL). This maximizes the effectiveness of the Adaptive Discriminative filters for each specific CFA Pattern (ADP) discovered in the subsequent step. In Step 2 as Fig. 4(b), we further enhance the UM using a small number of specialized network kernels for each DM task. Lastly, as Fig. 4(c), we introduce a meta-test learning framework that ensures robust DM output in the presence of unknown artifacts.

4.1. Step 1: Two-stage Knowledge Learning

This step aims to train the baseline of unified DM model (baseline UM) for all CFAs using the two-stage knowledge learning [10] (TKL), with independent DM models for each CFA (IMs). The IMs, with the same network architecture, have independent network parameters ($\{\theta_i\}_{i=1}^k$) dependent on each CFA-specific DM task (k). Note that the IM achieves high performance as a specialized model for each task, but requires a model k times larger than UM (θ_{um}).

First, we pre-train each individual IM based on NAFNet [9], renowned for its high performance despite having few network parameters. Then, in the knowledge collection (KC) stage, set the IMs specialized for each CFA DM task as the teacher network and UM as the student network to learn and collect knowledge from the teacher. In the knowledge examination (KE) stage after KC, train only using the student network and GT labels without guidance from the teacher network. We applied TKL method to increase the model's capacity after feature-level guidance for each CFA pattern, in order to maximize the effect of top filter detection in FAIG (Filter Attribution method based on Integral Gradient) [67] (see actual results in Tab. 1).

4.2. Step 2: Adaptive Discriminative Filters for a specific CFA Pattern

Xie *et al.* [67] proposed FAIG, which can detect discriminative filters of specific degradation. FAIG measures integrated gradient (IG) [60, 61] between baseline and target models. Inspired by FAIG and its application in another domain [47], we applied CNN for Adaptive Discriminative filters for a specific CFA Pattern (ADP) using the leveraged FAIG method. FAIG score is as follows: $FS_j = FAIG_j(\theta_{um}, \theta_i, x_i)$, for multiple CFA filters $i = 1, \dots, k$

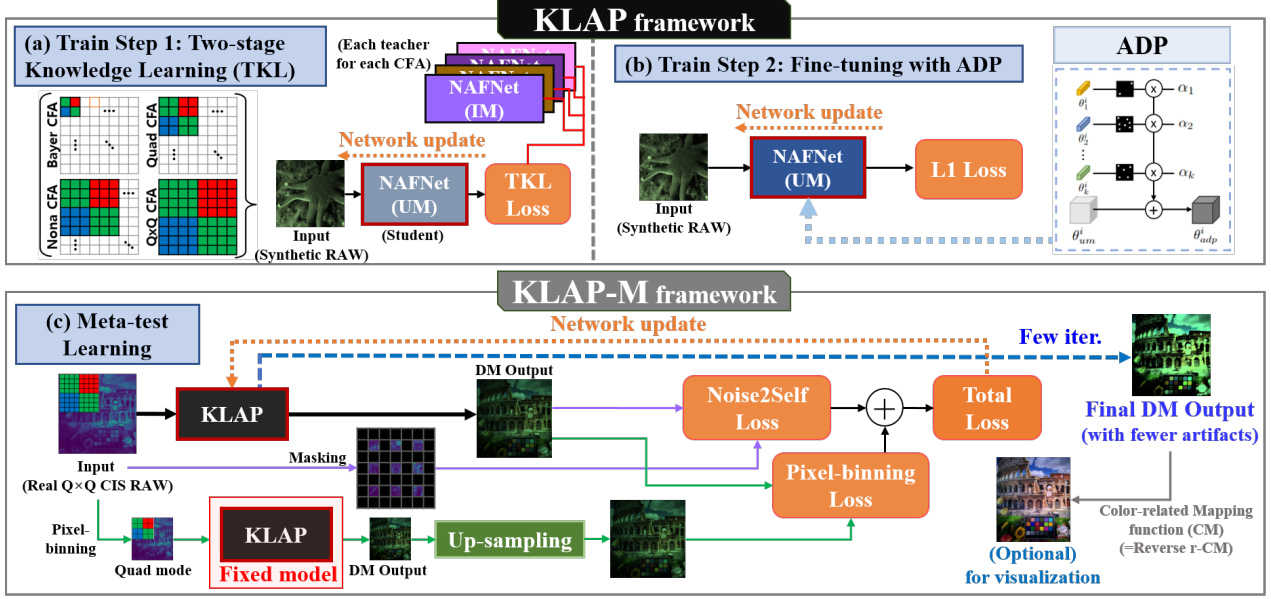


Figure 4: The overview of our proposed unified DM model, Knowledge Learning-based demosaicing model for Adaptive Pattern (KLAP) and KLAP with Meta-test learning (KLAP-M). KLAP consists of 2 steps: (a) two-stage knowledge learning (TKL) for training baselines, (b) fine-tuning using Adaptive Discriminant filters for each specific CFA Pattern (ADP). (c) KLAP-M employs meta-learning to reduce unknown artifacts in real RAW images during inference.

and all kernels j . Once the FAIG scores are computed, they are then ranked in descending order. The top $q\%$ of kernels are selected for each demosaicing process, with q representing a fixed value between 0.5 and 5.

We propose ADP, implemented by the masks M_c that are selected kernels using FAIG as illustrated in Fig. 4 and defined as follows:

$$\theta_{adp}^i = \theta_{um}^i + \sum_{c=1}^k \alpha_c \theta_c^i * M_c^i \quad (1)$$

where i is kernel index, $*$ is point-wise multiplication, θ_{um}^i refers to the pre-trained integrated model in Step 1, and α_c is a coefficient for a specific CFA pattern and is set up either as 1 or 0. Note that in a real non-Bayer CIS on a mobile device, the pattern mode i is determined by the mobile AP after detecting the lighting conditions. Also θ_c^i is an additional kernel for specific CFA pattern. The ratio q in the mask is determined empirically to be 1%. For example, the ratio of 1% in the mask is 1% for 4 demosaicing types, our proposed method uses an additional 4% of the entire network parameters as compared to the baseline UM. More detailed information is in the supplementary Sec. S.3. Our proposed KLAP, a combination of TKL and ADP, achieves state-of-the-art performance in various CFA DM tasks by replacing only relevant CNN kernels in UM from TKL.

4.3. Meta-learning during Inference

As shown in Fig. 4, we propose meta-learning during inference (meta-test learning) to mitigate unknown artifacts caused by sensor characteristics or shooting environments. By performing a few network updates during inference, this approach produces robust results. Our proposed meta-learning during inference consists of pixel binning loss and Noise2Self (N2S) loss, one of the self-supervised denoising techniques. As mentioned in Sec. 3.1, pixel binning compensates for resolution loss by increasing the light sensitivity, thus reducing noise. Based on CIS domain knowledge, we propose a self-supervised denoising method using a pixel binning loss to remove unknown artifacts.

$$\mathcal{L}_{pix} = |G(x_{J^c}, \theta_{adp}) - U(G(m(x_{J^c}), \theta_{adp'}))|, \quad (2)$$

where x and J^c denote the CIS RAW data and mask used by N2S, m and U represent average-based pixel-binning operation and up-sampling operation, respectively. G is a unified network structure and θ_{adp} is network parameters of ADP. $\theta_{adp'}$ is the initial network parameters that are not updated.

Additionally, we apply modified N2S loss to maintain robustness against noise (Poisson and Gaussian noise) that may occur depending on the shooting environment and to prevent blur caused by pixel binning loss:

$$\mathcal{L}_{N2S} = |G(x_{J^c}, \theta_{adp})_J - x_J| \quad (3)$$

where x_J and x_J^c are represent independent images using the mask scheme. Additional information about the pixel binning loss and N2S loss can be found in the supplementary material (See Sec. S.4.2.)

The total loss for meta-learning during inference is as follows:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{pix}} \mathcal{L}_{\text{pix}} + \lambda_{\text{N2S}} \mathcal{L}_{\text{N2S}} \quad (4)$$

where λ_{pix} and λ_{N2S} are used to balance different loss conditions and is experimentally found through visualization.

5. Experimental Results

As stated in Sec. 3.2, we generate synthetic DF2K Bayer and Non-Bayer CIS (DF2K-CIS) dataset utilizing DF2K, a combination of two open source datasets, DIV2K [2] and Flickr2K [39]. The training set comprises 2,500 images, with a validation set of 50 images and a test set of 1000 images. Furthermore, we propose to use the DF2K-CIS test dataset with strong noise to evaluate the effectiveness of our proposed meta-test learning in generating robust results. The DF2K-CIS strong noise test dataset comprises 200 images with noise parameters four times larger than those used in training. Then, we evaluate our proposed meta-learning method, KLAP-M, using 7 Q×Q CIS RAW images (48MP) with a resolution of 8000×6000 , 1 Quad CIS RAW image, and 3 Bayer CIS RAW images (50MP) with a resolution of 8192×6144 , all of which are 10-bit images captured directly by each type of CIS chip. In the meta-test learning, KLAP-M is trained using the loss function in Eq.(4) with $\lambda_{\text{pix}} = 1$ and $\lambda_{\text{N2S}} = 0.02$. Note that Meta-test (KLAP-M) does not utilize IMs but instead employs a unified model, and we conducted KLAP-M evaluations on each new full image for each sensor type. More implementation details and demosaicing RAW results can be found in Sec. S.5, S.7 and S.8. of the supplementary materials

5.1. Results on Synthetic RAW Dataset

5.1.1 Comparison of Ablation Studies and KLAP with Other Methods

Ablation study for KLAP. We perform ablation studies on the proposed KLAP approach based on NAFNet [9], including TKL and ADP, as shown in Fig. 4 (a) and (b), using the DF2K-CIS test dataset. Tab. 1 summarized the performance of PSNR (dB) and the number of parameters (Million). Baseline UM is a simple integrated model trained on all tasks, while total IMs require 4 times more network parameters than UM. Baseline UM-Large (Baseline UM-L) refers to a modified version of NAFNet [9] with increased network blocks. In TKL and ADP in the table, each step is independently applied to the baseline UM. TKL-to-IM refers to the re-trained IMs after applying TKL.

Using TKL and ADP independently leads to only a marginal improvement of 0.05 dB and 0.11 dB, respec-

Table 1: Ablation study for our proposed KLAP and Quantitative performance comparison (Chen [10] and Li [36]) on DF2K-CIS test dataset in terms of PSNR (dB) and the number of parameters (Million). Baseline-UM is a simple unified model. TKL is Baseline UM applying TKL, and ADP is Baseline UM applying ADP independently. TKL-to-IM involves fine-tuning IM after applying TKL. Chen [10] and Li [36] are based on MSBDN [15] and AirNet, respectively, while other experiments are based on NAFNet [9]. Note that Avg. denotes mean of all CFA’s PSNR, and Par. denotes the number of parameters.

Method	Ba.	Qu.	No.	QxQ	Avg.	Par.
IM	42.18	41.80	41.14	41.42	41.64	68.4
TKL-to-IM	42.36	41.89	41.58	41.60	41.86	68.4
Baseline UM	41.90	41.40	41.03	41.09	41.35	17.1
Baseline UM-L	41.95	41.44	41.08	41.13	41.40	19.4
Chen [10]	41.43	40.89	40.54	40.49	40.84	28.7
Li [36]	38.28	38.08	38.23	36.94	37.88	7.6
TKL	41.89	41.44	41.11	41.15	41.40	17.1
ADP	42.06	41.50	41.14	41.16	41.46	17.8
KLAP (Ours)	42.25	41.75	41.42	41.41	41.71	17.8

tively, compared to Baseline UM. Our proposed KLAP (TKL+ADP) further improved performance by 0.4 dB with a slightly increased number of network parameters compared to Baseline UM. Notably, Our KLAP achieved significantly higher performance than Baseline UM-L (41.71dB vs. 41.40dB) with fewer parameters (17.8M vs. 19.4M). In addition, fine-tuning each IM with pre-trained TKL resulted in a notable improvement compared to the original IMs, attributed to the inclusion of contrastive learning loss in TKL. Our proposed KLAP method, which combines TKL and ADP, significantly improves demosaicing performance for all CFAs.

Comparisons among other unifying methods. We evaluate the performance of our KLAP with NAFNet [9] on a DF2K-CIS test dataset and summarize the results in Tab. 1 in terms of PSNR (dB) and the number of parameters. We use the official codes provided by the authors of Airnet [36] and Chen [10]. The Chen [10] method uses the MSBDN-based TKL method. Despite a slight increase in network parameters by 0.7M (about 4%) in NAFNet, our KLAP yields significantly improved performance by 0.4 dB compared to the IM method. Notably, our KLAP yields the highest PSNR among all-in-one methods [10, 36] while using smaller network parameters compared to existing methods applied to NAFNet networks. Fig. 5 shows DM results on synthetic datasets for visual comparisons. We adjust CM in Sec. 3.2 for visualization. The images on the 1st to 4th rows are input synthetic RAW images and their DM outputs of UM, Chen [10], Li [36], and our KLAP are on the 2nd, 3rd, 4th, 5th column of Fig. 5, respectively. This shows that our KLAP outperforms other state-of-the-art unifying methods on DF2K-CIS test datasets.

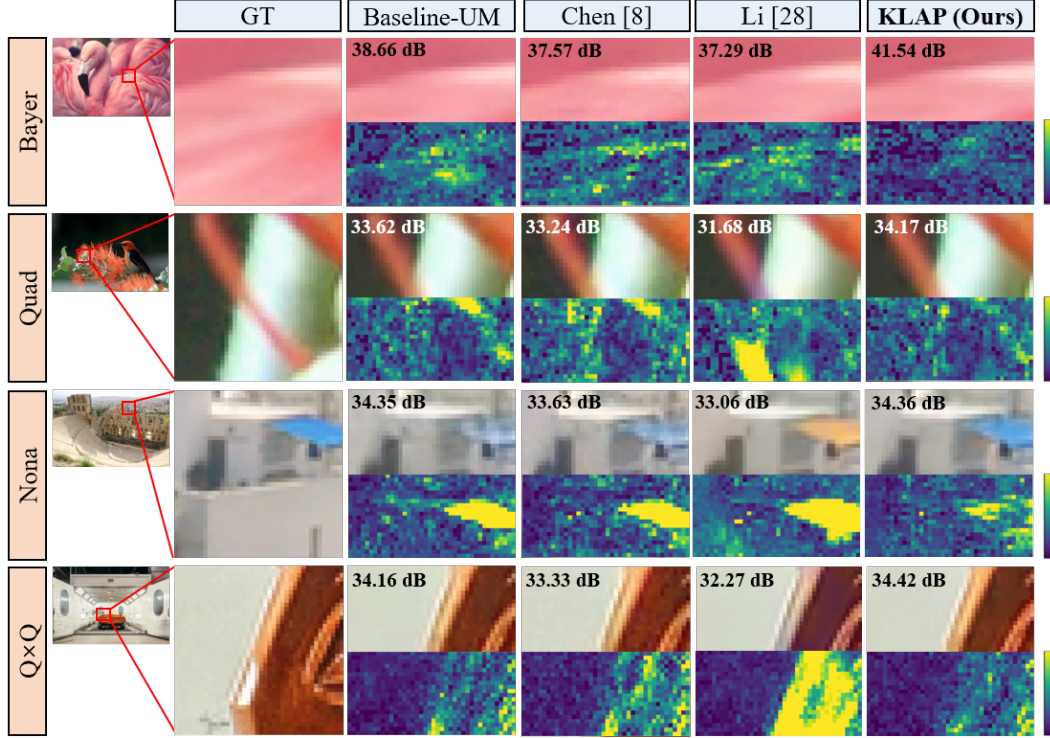


Figure 5: Comparisons of demosaiced images (**top**) from different methods and their difference maps (**bottom**) on the synthetic RAW (DF2K-CIS) test dataset. The PSNR (dB) value displayed in the top-left corner is for the entire image. As shown in the figure above, our proposed KLAP achieves the best performance in synthetic RAW test dataset.

5.1.2 Performance and Selected Filter Locations

To demonstrate the superiority of FAIG [67] over random selection, we evaluate various mask selection strate-

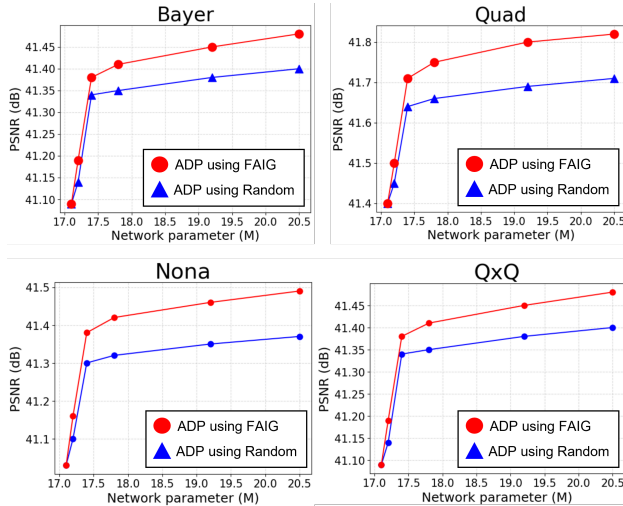


Figure 6: Performance comparisons among different filter location selections (0%, 0.1%, 0.5%, 1%, 3%, and 5%, respectively) for UM with ADP: Random selection method and FAIG adjusting ADP on DF2K CIS test dataset.

gies in our ADP method on synthetic datasets with both Bayer and non-Bayer patterns. The mask selection ratios are set to 0.1%, 0.5%, 1%, 3%, and 5%. We use a UM with TKL-based NAFNet [9] and add adaptive network kernels in proportion to the q ratio. Two mask selection methods are investigated: random selection and the FAIG method introduced in Sec. S.3. Fig. 6 summarizes our results, indicating that our ADP adopting FAIG outperforms random filter selection, underscoring the effectiveness of selecting discriminative filters for each CFA DM task. This implies that discriminative filters can be defined as task-specific (in our case, each CFA DM) filters, rather than randomly selected filters.

5.1.3 Analysis of Robustness in Strong Noise

To validate the robustness of Meta-test learning in KLAP-M, we evaluate KLAP-M on the DF2K-CIS with strong noise test dataset and summarize the results in a table. The DF2K-CIS with strong noise dataset has four times larger noise parameters compared to the DF2K-CIS training dataset. As shown in Tab. 2, KLAP shows slightly more robust results compared to existing methods. Furthermore, when KLAP-M is applied, it achieves an average improvement of 1.8 dB in PSNR with only 10 iterations.

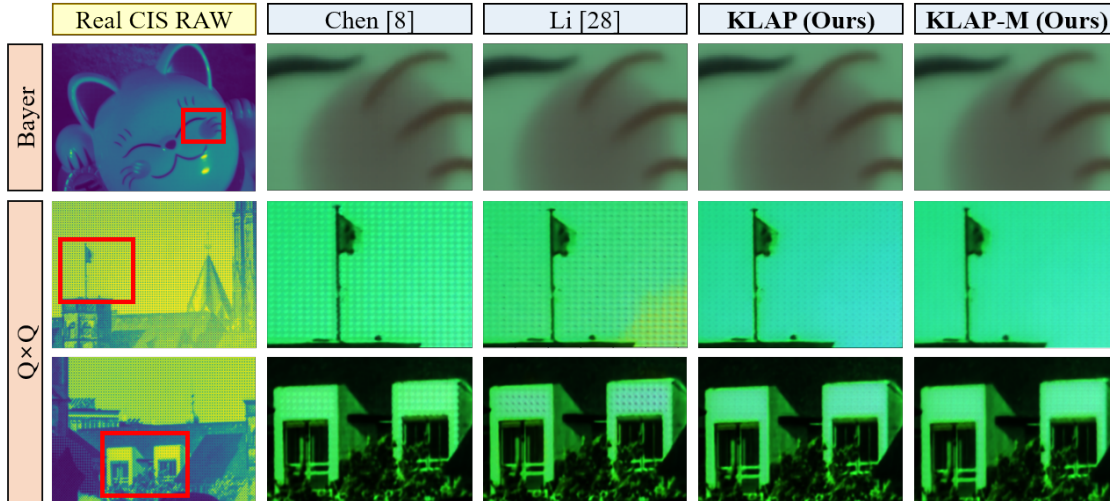


Figure 7: Qualitative DM results on the real CIS RAW. Note that KLAP with meta-test learning (KLAP-M) shows robust performance in real CIS RAW, despite of existence of sensor-generic unknown artifacts.

Table 2: Performance comparisons among different methods of robustness with strong noise in terms of PSNR (dB) on DF2K-CIS test dataset with strong noise. The noise parameters used in the test are four times larger than the noise parameters used in the training, and the number of meta-learning iterations in KLAP-M is fixed to 10.

CFA	Chen [10]	Li [36]	KLAP	KLAP-M
Bayer	32.60	31.61	32.98	33.32
Quad	32.48	31.58	32.93	35.41
Nona	32.44	31.64	32.88	35.06
$Q \times Q$	32.45	31.38	32.86	35.41

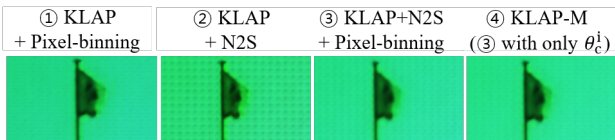


Figure 8: Ablation study of our proposed KLAP-M. The comparison shows the effect of each component of meta-learning in KLAP-M.

5.2. Results on Real CIS RAW

We evaluate the performance of our KLAP with meta-learning on a real RAW dataset and present the results in Figure 7. The number of iterations for meta-learning is fixed at 45. In the Bayer case, our method, as well as Chen [10] and Li [36]’s methods, show robust results on real data. However, In the case of demosaicing $Q \times Q$, Chen and Li’s methods are unable to alleviate artifacts, while our method significantly mitigates resulting artifacts during inference by reducing domain gap through meta-learning. Figure 8 shows the ablation study of KLAP-M and demonstrates

superior performance compared to other method combinations. Note that the Bayer output is an image that has been squared by 0.7 from the original outputs for visual comparison purposes. We represent the two $Q \times Q$ output images, with their pixel values (range of 0 to 1) cubed, to compare the artifact mitigation performance with other models.

5.3. Limitations

To utilize deep learning-based DM models for CIS, the requirement of a specialized circuit with embedded AI accelerators can be a limiting factor.

6. Conclusion

Our proposed demosaicing method uses task-specific kernels to cover all CFAs and incorporates a meta-testing framework to produce efficient and robust results. This approach boasts low computational complexity, robustness to unknown artifacts, and high-quality demosaiced images.

Acknowledgments This work was supported in part by the National Research Foundation of Korea(NRF) grants funded by the Korea government(MSIT) (NRF-2022R1A4A1030579), Basic Science Research Program through the NRF funded by the Ministry of Education(NRF-2017R1D1A1B05035810) and Creative-Pioneering Researchers Program through Seoul National University. The CIS RAW data and CIS domain knowledge were supported by CIS Development Representative at SK hynix.

References

- [1] SM A Sharif, Rizwan Ali Naqvi, and Mithun Biswas. Beyond joint demosaicking and denoising: An image processing pipeline for a pixel-bin image sensor. In *CVPR*, 2021.
- [2] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *CVPRW*, 2017.
- [3] Boaz Arad, Radu Timofte, Rony Yahel, Nimrod Morag, Amir Bernat, Yaqi Wu, Xun Wu, Zhihao Fan, Chenjie Xia, Feng Zhang, et al. Ntire 2022 spectral demosaicing challenge and data set. In *CVPRW*, 2022.
- [4] Joshua Batson and Loic Royer. Noise2self: Blind denoising by self-supervision. In *ICML*, 2019.
- [5] Bryce E Bayer. Color imaging array. *United States Patent 3,971,065*, 1976.
- [6] Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *CVPR*, 2019.
- [7] Jaeseok Byun, Sungmin Cha, and Taesup Moon. Fbi-denoiser: Fast blind image denoiser for poisson-gaussian noise. In *CVPR*, 2021.
- [8] Jierun Chen, Song Wen, and S-H Gary Chan. Joint demosaicking and denoising in the wild: The case of training under ground truth uncertainty. In *AAAI*, 2021.
- [9] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. *ECCV*, 2022.
- [10] Wei-Ting Chen, Zhi-Kai Huang, Cheng-Che Tsai, Hao-Hsiang Yang, Jian-Jiun Ding, and Sy-Yen Kuo. Learning multiple adverse weather removal via two-stage knowledge learning and multi-contrastive regularization: Toward a unified model. In *CVPR*, 2022.
- [11] Minhyeok Cho, Haechang Lee, Hyunwoo Je, Kijeong Kim, Dongil Ryu, Jinsu Kim, Jonghyun Bae, and Albert No. Pynet-qxq: A distilled pynet for qxq bayer pattern demosaicing in cmos image sensor. *arXiv preprint arXiv:2203.04314*, 2022.
- [12] Kai Cui, Zhi Jin, and Eckehard Steinbach. Color image demosaicking using a 3-stage convolutional neural network structure. In *IEEE ICIP*, 2018.
- [13] Valéry Dewil, Adrien Courtois, Mariano Rodríguez, Thibaud Ehret, Nicola Brandonisio, Denis Bujoreanu, Gabriele Facciolo, and Pablo Arias. Video joint denoising and demosaicing with recurrent cnns. In *WACV*, 2023.
- [14] Steven Diamond, Vincent Sitzmann, Frank Julca-Aguilar, Stephen Boyd, Gordon Wetzstein, and Felix Heide. Dirty pixels: Towards end-to-end image processing and perception. *ACM Transactions on Graphics TOG*, 2021.
- [15] Hang Dong, Jinshan Pan, Lei Xiang, Zhe Hu, Xinyi Zhang, Fei Wang, and Ming-Hsuan Yang. Multi-scale boosted dehazing network with dense feature fusion. In *CVPR*, 2020.
- [16] Egor Ershov, Alex Savchik, Denis Shepelev, Nikola Banić, Michael S Brown, Radu Timofte, Karlo Košćević, Michael Freeman, Vasily Tesalin, Dmitry Bocharov, et al. Ntire 2022 challenge on night photography rendering. In *CVPR*, 2022.
- [17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.
- [18] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. *NeurIPS*, 2018.
- [19] Felix Heide, Markus Steinberger, Yun-Ta Tsai, Mushfiquur Rouf, Dawid Pająk, Dikpal Reddy, Orazio Gallo, Jing Liu, Wolfgang Heidrich, Karen Egiazarian, et al. Flexisp: A flexible camera image processing framework. *ACM ToG*, 2014.
- [20] Xuecai Hu, Haoyuan Mu, Xiangyu Zhang, Zilei Wang, Tieniu Tan, and Jian Sun. Meta-sr: A magnification-arbitrary network for super-resolution. In *CVPR*, 2019.
- [21] Tao Huang, Songjiang Li, Xu Jia, Huchuan Lu, and Jianzhuang Liu. Neighbor2neighbor: Self-supervised denoising from single noisy images. In *CVPR*, 2021.
- [22] Andrey Ignatov, Grigory Malivenko, Radu Timofte, Yu Tseng, Yu-Syuan Xu, Po-Hsiang Yu, Cheng-Ming Chiang, Hsien-Kai Kuo, Min-Hung Chen, Chia-Ming Cheng, et al. Pynet-v2 mobile: Efficient on-device photo processing with neural networks. In *ICPR*, 2022.
- [23] Andrey Ignatov, Luc Van Gool, and Radu Timofte. Replacing mobile camera isp with a single deep learning model. In *CVPRW*, 2020.
- [24] Dongyoung Jang, Donghyuk Park, Seungwon Cha, Heesang Kwon, Mihye Kim, Seungwook Lee, Haewon Lee, Seonok Kim, Nakyung Lee, Jinhwa Han, et al. 0.8 μm -pitch cmos image sensor with dual conversion gain pixel for mobile applications. In *IISW*, 2019.
- [25] Kyeonghoon Jeong, Jonghyun Kim, and Moon Gi Kang. Color demosaicing of rgbw color filter array based on laplacian pyramid. *Sensors*, 2022.
- [26] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *CVPRW*, 2020.
- [27] Qiyu Jin, Gabriele Facciolo, and Jean-Michel Morel. A review of an old dilemma: Demosaicking first, or denoising first? In *CVPRW*, 2020.
- [28] Jaejin Jung, Sinhwon Lim, Jiyong Kim, Kwisung Yoo, Won-tak Choi, Youngsun Oh, Juhyun Ko, and Kyoungmin Koh. A 1/1.33-inch 108mpixel cmos image sensor with 0.8 μm unit nonacell pixels. In *IEEE ISCAS*, 2022.
- [29] Mehdi Khabir, Hamzeh Alaibakhsh, and Mohammad Azim Karami. Electrical crosstalk analysis in a pinned photodiode cmos image sensor array. *Applied Optics*, 2021.
- [30] Mehdi Khabir and Mohammad Azim Karami. Characterization and analysis of electrical crosstalk in a linear array of cmos image sensors. *Applied Optics*, 2022.
- [31] Byung-Hoon Kim, Joonyoung Song, Jong Chul Ye, and Jae-Hyun Baek. Pynet-ca: enhanced pynet with channel attention for end-to-end mobile image signal processing. In *EC-CVW*, 2020.
- [32] Irina Kim, Dongpan Lim, Youngil Seo, Jeongguk Lee, Yun-seok Choi, and Seongwook Song. On recent results in demosaicing of samsung 108mp cmos sensor using deep learning. In *IEEE TENSYP*, 2021.
- [33] Irina Kim, Seongwook Song, Soonkeun Chang, Sukhwan Lim, and Kai Guo. Deep image demosaicing for submicron image sensors. *JIST*, 2019.
- [34] Filippas Kokkinos and Stamatis Lefkimmiatis. Deep image demosaicking using a cascade of convolutional residual denoising networks. In *ECCV*, 2018.

- [35] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. *ICML*, 2018.
- [36] Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *CVPR*, 2022.
- [37] Ruoteng Li, Robby T Tan, and Loong-Fah Cheong. All in one bad weather removal using architectural search. In *CVPR*, 2020.
- [38] Yihao Li, Junyu Long, Yun Chen, Yan Huang, and Ni Zhao. Crosstalk-free, high-resolution pressure sensor arrays enabled by high-throughput laser manufacturing. *Advanced Materials*, 2022.
- [39] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, 2017.
- [40] Lin Liu, Xu Jia, Jianzhuang Liu, and Qi Tian. Joint demosaicing and denoising with self guidance. In *CVPR*, 2020.
- [41] Karima Ma, Michael Gharbi, Andrew Adams, Shoaib Kamil, Tzu-Mao Li, Connelly Barnes, and Jonathan Ragan-Kelley. Searching for fast demosaicking algorithms. *ACM TOG*, 2022.
- [42] Shruti H Mahajan and Varsha K Harpale. Adaptive and non-adaptive image interpolation techniques. In *ICCCA*, 2015.
- [43] Chong Mou, Qian Wang, and Jian Zhang. Deep generalized unfolding networks for image restoration. In *CVPR*, 2022.
- [44] Hao Ni, Jingkuan Song, Xiaopeng Luo, Feng Zheng, Wen Li, and Heng Tao Shen. Meta distribution alignment for generalizable person re-identification. In *CVPR*, 2022.
- [45] Youngsun Oh, Munhwan Kim, Wonchul Choi, Hana Choi, Honghyun Jeon, Junho Seok, Yujung Choi, Jaejin Jung, Kwisung Yoo, Donghyuk Park, et al. A 0.8 μm nonacell for 108 megapixels cmos image sensor with fd-shared dual conversion gain and 18,000 e-full-well capacitance. In *IEEE IEDM*, 2020.
- [46] Tetuya Okawa, S Ooki, H Yamajo, M Kawada, M Tachi, K Goi, T Yamasaki, H Iwashita, M Nakamizo, T Ogasahara, et al. A 1/2inch 48m all pdaif cmos image sensor using 0.8 μm quad bayer coding 2×2 oc1 with 1.0 lux minimum af illuminance level. In *IEEE IEDM*, 2019.
- [47] Dongwon Park, Byung Hyun Lee, and Se Young Chun. All-in-one image restoration for unknown degradations using adaptive discriminative filters for specific degradations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- [48] Hye Yeon Park, Yunki Lee, Jonghoon Park, Hyunseok Song, Taesung Lee, Hyung Keun Gweon, Yunji Jung, Jeongmin Bae, Boseong Kim, Junwon Han, et al. Advanced novel optical stack technologies for high snr in cmos image sensor. In *IEEE VLSI Technology and Circuits*, 2022.
- [49] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019.
- [50] Juewen Peng, Zhiguo Cao, Xianrui Luo, Hao Lu, Ke Xian, and Jianming Zhang. Bokehme: When neural rendering meets classical rendering. In *CVPR*, 2022.
- [51] Mara Pistellato, Filippo Bergamasco, Tehreem Fatima, and Andrea Torsello. Deep demosaicing for polarimetric filter array cameras. *IEEE TIP*, 2022.
- [52] Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *CVPR*, 2017.
- [53] Vitchyr H Pong, Ashvin V Nair, Laura M Smith, Catherine Huang, and Sergey Levine. Offline meta-reinforcement learning with online self-supervision. In *ICML*, 2022.
- [54] Kuldeep Purohit, Maitreya Suin, AN Rajagopalan, and Vishnu Naresh Boddeti. Spatially-adaptive image restoration using distortion-guided networks. In *ICCV*, 2021.
- [55] Jaesung Rim, Geonung Kim, Jungeon Kim, Junyong Lee, Seungyong Lee, and Sunghyun Cho. Realistic blur synthesis for learning image deblurring. *ECCV*, 2022.
- [56] Corban G Rivera, David A Handelman, Christopher R Ratto, David Patrone, and Bart L Paulhamus. Visual goal-directed meta-imitation learning. In *CVPR*, 2022.
- [57] Eli Schwartz, Raja Giryes, and Alex M Bronstein. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE TIP*, 2018.
- [58] SMA Sharif, Rizwan Ali Naqvi, and Mithun Biswas. Sagan: Adversarial spatial-asymmetric attention for noisy non-bayer reconstruction. *arXiv preprint arXiv:2110.08619*, 2021.
- [59] Ana Stojkovic, Ivana Shopovska, Hiep Luong, Jan Aelterman, Ljubomir Jovanov, and Wilfried Philips. The effect of the color filter array layout choice on state-of-the-art demosaicing. *Sensors*, 2019.
- [60] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Gradients of counterfactuals. *arXiv preprint arXiv:1611.02639*, 2016.
- [61] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, 2017.
- [62] Ethan Tseng, Ali Mosleh, Fahim Mannan, Karl St-Arnaud, Avinash Sharma, Yifan Peng, Alexander Braun, Derek Nowrouzezahrai, Jean-Francois Lalonde, and Felix Heide. Differentiable compound optics and processing pipeline optimization for end-to-end camera design. *ACM TOG*, 2021.
- [63] Ethan Tseng, Yuxuan Zhang, Lars Jebe, Xuaner Zhang, Zhihao Xia, Yifei Fan, Felix Heide, and Jiawen Chen. Neural photo-finishing. *ACM TOG*, 2022.
- [64] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxim: Multi-axis mlp for image processing. In *CVPR*, 2022.
- [65] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *CVPR*, 2022.
- [66] Xun Wu, Zhihao Fan, Jiesi Zheng, Yaqi Wu, and Feng Zhang. Learning to joint remosaic and denoise in quad bayer cfa via universal multi-scale channel attention network. In *ECCVW*, 2023.
- [67] Liangbin Xie, Xintao Wang, Chao Dong, Zhongang Qi, and Ying Shan. Finding discriminative filters for specific degradations in blind super-resolution. *NeurIPS*, 2021.

- [68] Wenzhu Xing and Karen Egiazarian. End-to-end learning for joint image demosaicing, denoising and super-resolution. In *CVPR*, 2021.
- [69] Ying Xiong, Kate Saenko, Trevor Darrell, and Todd Zickler. From pixels to physics: Probabilistic color de-rendering. In *CVPR*, 2012.
- [70] Jun Xu, Yuan Huang, Ming-Ming Cheng, Li Liu, Fan Zhu, Zhou Xu, and Ling Shao. Noisy-as-clean: Learning self-supervised denoising from corrupted image. *IEEE TIP*, 2020.
- [71] Xuan Xu, Yanfang Ye, and Xin Li. Joint demosaicing and super-resolution (jdsr): Network design and perceptual optimization. *IEEE TCI*, 2020.
- [72] Fengxiang Yang, Zhun Zhong, Zhiming Luo, Yuanzheng Cai, Yaojin Lin, Shaozi Li, and Nicu Sebe. Joint noise-tolerant learning and meta camera shift adaptation for unsupervised person re-identification. In *CVPR*, 2021.
- [73] Qingyu Yang, Guang Yang, Jun Jiang, Chongyi Li, Ruicheng Feng, Shangchen Zhou, Wenxiu Sun, Qingpeng Zhu, Chen Change Loy, Jinwei Gu, et al. Mipi 2022 challenge on quad-bayer re-mosaic: Dataset and report. In *ECCVW*, 2022.
- [74] Yoonjong Yoo, Jaehyun Im, and Joonki Paik. Low-light image enhancement using adaptive digital pixel binning. *Sensors*, 2015.
- [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 *CVPR*, 2022.
- [76] 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 *CVPR*, 2020.
- [77] 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.
- [78] Tao Zhang, Ying Fu, and Cheng Li. Deep spatial adaptive network for real image demosaicing. In *AAAI*, 2022.
- [79] Yuyang Zhao, Zhun Zhong, Fengxiang Yang, Zhiming Luo, Yaojin Lin, Shaozi Li, and Nicu Sebe. Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification. In *CVPR*, 2021.
- [80] Zhimin Zhou, Bedabrata Pain, and Eric R Fossum. Frame-transfer cmos active pixel sensor with pixel binning. *IEEE T-ED*, 1997.
- [81] Magauiya Zhussip, Shakarim Soltanayev, and Se Young Chun. Extending stein’s unbiased risk estimator to train deep denoisers with correlated pairs of noisy images. *NeurIPS*, 2019.
- [82] Magauiya Zhussip, Shakarim Soltanayev, and Se Young Chun. Training deep learning based image denoisers from undersampled measurements without ground truth and without image prior. In *CVPR*, 2019.

Efficient Unified Demosaicing for Bayer and Non-Bayer Patterned Image Sensors (Supplementary Material)

Haechang Lee^{1,4,*}, Dongwon Park^{2,*}, Wongi Jeong^{1,*},
Kijeong Kim⁴, Hyunwoo Je⁴, Dongil Ryu⁴ and Se Young Chun^{1,2,3,†}
¹Dept. of ECE, ²INMC, ³IPAI, Seoul National University, Republic of Korea,
⁴SK hynix, Republic of Korea
{harrylee,donglpark,wg7139,sychun}@snu.ac.kr

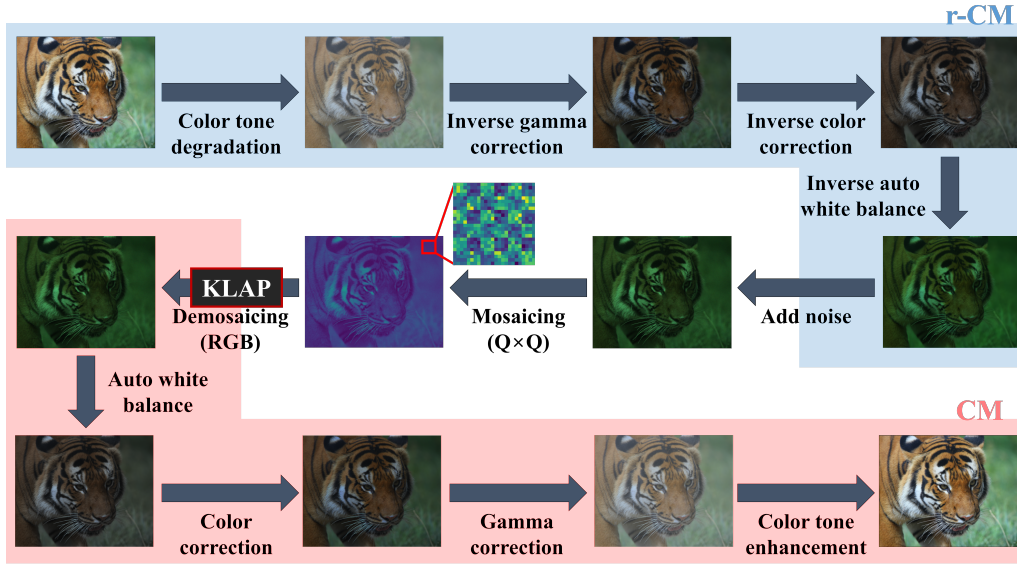


Figure S.1: Overview of our pipeline for synthesizing realistic RAW images, specifically for $Q \times Q$ patterns.

S.1. Detailed Data Synthesis for Demosaicing All CFAs

As described in our paper, we generate synthetic ground truth (GT) by sequentially applying a 4-step **reverse Color-related Mapping (r-CM)** process. Then, we add mixed Poisson and Gaussian noise and performed mosaicing (*i.e.*, CFA patterning) on the entire image to create synthetic RAW-like images (as shown in the blue shaded area in Fig. S.1). The r-CM process consists of the following modules: color tone degradation, inverse gamma correction, inverse color correction, and inverse auto white balance functions. The color matrix (CM) is the inverse of the reverse color matrix (r-CM) and can only be applied to the output of the demosaicing (DM) model.

Note that we need to use r-CM for data synthesis on the open-source dataset to generate GT images, while CM can

be “optionally” applied after demosaicing for better visualization in our paper.

Color tone degradation. Typically, color enhancement is performed in the latter part of the ISP chain. Therefore, we position the color tone degradation function at the beginning of r-CM. Inspired by [6], we adopt a tone mapping function that uses a simple inverse smoothing curve, to perform color tone degradation on open-source dataset images in the r-CM process. Note that the color tone enhancement function in CM is the inverse of color tone degradation in r-CM.

Inverse gamma correction. In the ISP chain, gamma correction is applied to image data to correct for the non-linear perception of brightness by the human eye. We use a gamma value setting of 2.2, which is standard for most cameras [16, 50, 69, 52]. In r-CM, the inverse function of gamma correction is applied, while in CM, standard gamma correction is performed.

* Equal contribution, † Corresponding author.

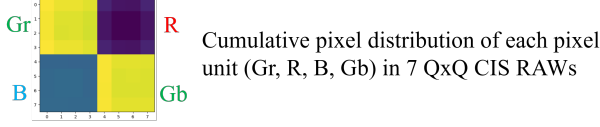


Figure S.2: The cumulative pixel value distribution of each homogeneous pixel unit (Gr, R, B, and Gb) in 7 Q×Q CIS RAW image samples. In our CIS RAW data, we observe a significant difference in signal values between inner and outer pixels in each Gr, R, B, and Gb pixel unit, which is mainly caused by crosstalk effect.

Inverse color correction. We use a color correction function to adjust the colors captured by a camera’s sensor to appear as they would to the human eye. The specific function we used is as follows:

$$\begin{pmatrix} R_{corrected} \\ G_{corrected} \\ B_{corrected} \end{pmatrix} = A \begin{pmatrix} R \\ G \\ B \end{pmatrix},$$

where A is a 3x3 color correction matrix (CCM), which is applied to the pixel values (R, G, and B) to obtain the corrected RGB values ($R_{corrected}$, $G_{corrected}$, and $B_{corrected}$). We obtain the CCM information from the CIS manufacturing company and apply it to our inverse color correction function after calculating the CCM’s inverse.

Inverse auto white balance. We empirically adjusted the gains for R, G, and B channels in the auto white balance function to make white portions of the CIS RAW appear white as perceived by the human eye. The inverse auto white balance in r-CM is obtained by reversing the values applied in the white balance module of the CM process.

Noise synthesis. We use the following practical mixed Poisson and Gaussian noise model [55, 6, 70]:

$$\begin{aligned} x_n &= \text{Poisson}(\gamma y_n) / \gamma + \epsilon_n, \\ \epsilon &\sim \mathcal{N}(0, \sigma_\epsilon^2 I), \quad n = 1, \dots, N, \end{aligned} \quad (1)$$

where y and x are clean image and corrupted image, respectively. Poisson generates pixel intensity-dependent Poisson noise caused by photon sensing, γ is a gain parameter which depends on the sensor and analog gain. ϵ is signal independent Gaussian noise with standard deviation σ , and N is the number of samples. DF2K-CIS train and test datasets are generated using the following imaging parameters: $\gamma = 0.01$ and $\sigma = 0.02$. DF2K-CIS with strong noise test dataset are generated with parameters that are 4 times larger than those of DF2K-CIS: $\gamma = 0.04$ and $\sigma = 0.08$.

S.2. Domain Gap Example: Inherent Grid Artifacts in CIS RAW

The differences in the distribution of pixels within each pixel unit are primarily caused by "crosstalk" effects, which

result from mutual interference of each pixel signal in CIS hardware [29, 30, 38]. As shown in Fig. S.2, in CIS Q×Q RAW (before demosaicing), we observe that the signals in the center of each pixel unit, especially in the R channel, are stronger than those in the outer pixels, while the edges of each pixel unit, particularly the four corners, are weaker. In addition to the cause of crosstalk phenomenon, the asymmetry between the inner and outer pixels in each pixel unit can vary across CIS devices, and this can manifest in various forms depending on the circuit configuration, component characteristics, product lines, and process capability of the CIS chip. The difference in pixel values in each of the homogeneous color units in CIS RAW may be causing grid artifacts.

S.3. Adaptive Discriminative Filter-based Model for Specific CFA Pattern (ADP)

S.3.1. Filter Attribution Integrated Gradients

Xie *et al.* [67] propose FAIG, which identifies discriminative filters of specific degradation in blind super-resolution (SR) by computing integrated gradient (IG) [61, 60] between the baseline and desired models. In FAIG, the baseline model is denoted as θ_{from} and the model being updated is denoted as θ_{to} for each desired task. The function $\rho(\beta)$, where $\beta \in [0, 1]$, represents an uninterrupted straight line between the baseline and target models. In that case, any certain route in $\rho(\beta)$ is represented by $\rho(\beta) = \beta\theta_{from} + (1 - \beta)\theta_{to}$, where $\rho(1) = \theta_{from}$ and $\rho(0) = \theta_{to}$. The FAIG on the continuous line space between two models is discretized as follows:

$$\begin{aligned} &\text{FAIG}_i(\theta_{from}, \theta_{to}, x) \\ &\approx \left| \frac{1}{N} [\theta_{from} - \theta_{to}]_i \sum_{t=0}^{N-1} \left[\frac{\partial \mathcal{L}(\rho(\beta_t), x)}{\partial \rho(\beta_t)} \right]_i \right|, \end{aligned} \quad (2)$$

where N represents the total number of steps used in the integral approximation, and N is set to 100 as in FAIG. β_t and i are t/N and the kernel index, respectively. We apply FAIG, originally proposed for denoising and deblurring, to multiple CFA sensor patterns in our demosaicing tasks.

S.3.2. Mask Ratio of FAIG in ADP

We choose a mask ratio (q) as 1% in ADP for each CFA in our KLAP framework, to balance demosaicing performance and efficiency (as shown in Sec. 4.2 and Fig. 4(b)). Increasing q improves performance but with diminishing returns and increased parameters (Tab. S.1). Compared to Baseline-UM (B.UM), our proposed method using mask ratio 1% for all 4 demosaicing types requires an additional 4% of network parameters.

Furthermore, our KLAP achieves significantly better results even when increasing the size of the Baseline UM method by 3.5 times, as shown in Fig. S.3.

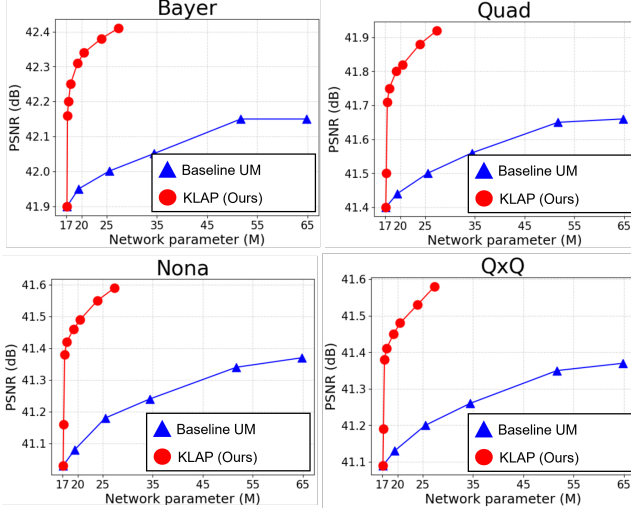


Figure S.3: Performance comparisons between Baseline UM with increased network sizes (17.1M, 19.4M, 25.5M, 34.4M, 51.7M, and 64.8M) and KLAP (Ours) with mask ratios $q\%$ (0%, 0.1%, 0.5%, 1%, 3%, 5%, 10% and 15%, respectively) on DF2K-CIS test dataset. The network size of KLAP (Ours) with mask ratios of $q\%$ (0%, 0.1%, 0.5%, 1%, 3%, 5%, 10%, and 15%) are 17.1M, 17.2M, 17.8M, 17.8M, 19.2M, 20.5M, 23.9M, and 27.4M, respectively. Our approach produces significantly higher performance results even with 3.5 times larger Baseline UM method.

S.4. Meta-learning during Inference

S.4.1. Definition of the term “meta-test”

In our paper, we name the process of fine-tuning with meta-learning during inference as meta-test learning in KLAP-M. The term “meta-test” typically refers to the process of improving performance on various generalization scenarios with only a few trials on unseen data [72, 18, 79, 44, 53, 56]. In general, the meta-test process works

Table S.1: Investigation of experiments according to KLAP (Ours) with filter location selection ratios (*i.e.*, mask selection ratios in FAIG [67]; $q\%$) in the DF2K-CIS test dataset. B.UM denotes the baseline UM. Note that Avg. and Par. denotes mean of all CFAs’ PSNR (dB) and the number of parameters (M).

Method	q	Ba.	Qu.	No.	QxQ	Avg.	Par.
B.UM	0	41.90	41.40	41.03	41.09	41.35	17.1
KLAP	0.1	42.16	41.50	41.16	41.19	41.50	17.2
KLAP	0.5	42.20	41.71	41.38	41.38	41.67	17.4
KLAP	1	42.25	41.75	41.42	41.41	41.71	17.8
KLAP	3	42.31	41.80	41.46	41.45	41.75	19.2
KLAP	5	42.34	41.82	41.49	41.48	41.78	20.5
KLAP	10	42.38	41.88	41.55	41.53	41.83	23.9
KLAP	15	42.41	41.92	41.59	41.58	41.87	27.4

in conjunction with the meta-training process. The meta-training process optimizes the model to improve the accuracy of meta-test samples using source data. We use ADP in the second step of our KLAP framework to only adjust the important kernel (θ_c) for each CFA demosaicing during training in order to improve the accuracy of meta-test. This can be seen as a type of meta-training process. In our paper, we define the process of fine-tuning only θ_c in KLAP during model inference as meta-test learning to achieve robust results even for undefined artifacts caused by CIS device features and shooting environments.

S.4.2. Noise2Self and Pixel Binning Loss.

To aid in a more thorough understanding in our meta-test learning process, KLAP-M, we provide a more detailed explanation of Noise2Self (N2S) [4] loss and pixel-binning loss in Fig. S.4 (a) and (b).

Noise2Self (N2S) loss. We choose Noise2Self [4] among many self-supervised denoising methods [82, 81, 35, 21, 4, 7]. To calculate N2S loss, the L1 loss is computed between an output of an image inputted into KLAP, where the empty pixels of x_{J^c} are interpolated, and x_J . In our study, we utilize the same masking scheme for each J as outlined in the N2S paper [4]. Each J samples a single pixel selected within each 4×4 window (*i.e.*, 6.25% of the number of pixels in each image). In the original N2S method, the interpolation function for x_{J^c} use a 3×3 kernel to compute the average value of the surrounding pixels for interpolation. However, we consider the characteristics of the RGB channel and calculate the average value of the surrounding values corresponding to that channel for interpolation. In the case of Bayer, we set a size of window to 6×6 and use 5×5 kernel for interpolation to prevent overlap. The use of N2S loss term has the effect of removing independent noise.

Pixel-binning loss. As mentioned in Sec. 3.1 in our paper, pixel binning is applied differently depending on the input pattern status of the CFA. Similarly, the proposed pixel binning loss based on CIS domain knowledge is also applied differently according to the CFA pattern. When using the average-based pixel binning operation (m), the $Q \times Q$ CFA pattern is converted to Quad or Bayer pattern. Nona and Quad patterns are converted to Bayer pattern. Note that pixel binning operation (m) does not exist in the Bayer pattern. The upsampling operation (U) employs a bilinear function to restore the original resolution, which may have been altered due to the pixel binning operation (m).

S.5. Implementation Details

In our experiments, we use a patch size of 240×240 to cover all of Bayer, Quad, Nona, and $Q \times Q$ CFAs. The model is trained using the ADAM optimizer with a batch size of 32 and an initial learning rate of 2×10^{-4} . We apply the cosine annealing learning rate decay technique with a

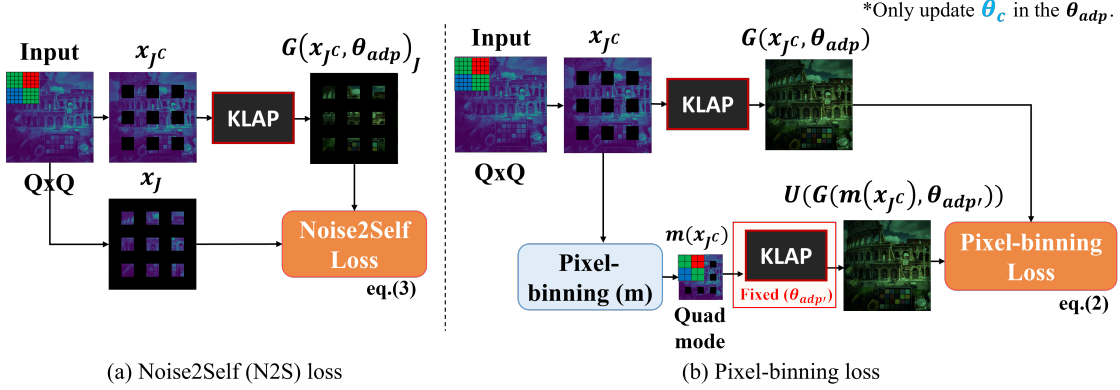


Figure S.4: The specific processes for calculating 2 loss functions in our proposed method, KLAP-M, which is KLAP with meta-test learning: (a) Noise2Self (N2S) loss, and (b) Pixel-binning loss.

Table S.2: Performance comparisons with existing DM methods for specific sensors in IMs

Method	Bayer	Quad	Nona	QxQ	Par.
IM [11]	37.03	37.38	36.65	36.44	4.2
IM [32]	41.33	40.81	39.85	37.02	13.8
IM [73]	41.89	41.19	40.60	40.74	83.0
KLAP	42.25	41.75	41.42	41.41	17.8

minimum learning rate of 1×10^{-6} . To ensure a fair comparison, we evaluate the proposed method on the same conditions with an NVIDIA A6000 GPU using PyTorch [49]. Our architecture is similar to the NAFNet [9] architecture, which has state-of-the-art performance in IM-based image restoration. We use the official codes provided by the authors of Chen [10] and Li [36]. In the Table 1, TKL denotes the method of applying NAFNet-based TKL, and Chen denotes the method of applying MBSDN-based TKL.

S.6. Demosaicing Methods Comparison

As the pioneers in integrated DM tasks for various sensor CFAs, we compared our proposed KLAP method with state-of-the-art integrated image restoration methods [10, 36] due to the lack of existing unified DM research. We conduct experiments on recent DM methods [11, 32] for specific sensors, as well as [65], one of the winners in the MIPI ‘22 [73] competition. The results, as shown in Tab. S.2 indicates that none surpassed ours.

S.7. Additional RAW Evaluation

MIPI ‘22 competition [73] emphasizes Quad-to-Bayer *re-mosaicing*, not demosaicing, so the definition of ground truth (GT) differs from our research focus. Nevertheless, we conducted inference on the MIPI inputs using KLAP and KLAP-M, as shown in Fig. S.5, effectively reducing vi-

sual artifacts and validating their performance. The MIPI challenge uses synthetic inputs without RGB GT, emphasizing the challenges of acquiring real CIS RAW data. This highlights the importance of our self-supervised learning approach for *real* sensor RAW in real-world scenarios.

S.8. Results

Fig. S.6 illustrates the qualitative results of the Baseline-UM (2nd column) with NAFNet, existing methods (Chen (3rd column), Li (4th column)) and our proposed KLAP (5th column), evaluated on the synthetic RAW (DF2K-CIS) test dataset. Fig. S.7 presents the qualitative results of prior arts (Chen (2nd column), Li (3rd column)) and our proposed KLAP (4th column) and KLAP-M (5th column) on the synthetic RAW (DF2K-CIS) with strong noise test dataset. Our proposed KLAP method visually outperform other state-of-the-art methods on DK2K-CIS test dataset, and our proposed KLAP-M method shows visually superior results compared to other state-of-the-art methods on the DK2K-CIS test with strong noise dataset, thanks to meta-learning during inference. Fig. S.8 shows our proposed KLAP-M inference output on the real CIS RAW data. Without meta-learning applied (0 iteration) in Fig. S.8, artifacts exist. On the other hand, as the number of meta-learning iterations increases, the artifacts gradually disappear. We selected 45 iterations for real data using this method. Furthermore, we observed that similar results were obtained even with further increases in iterations.

Fig. S.9 shows our proposed KLAP-M inference output on the real CIS RAW data set.

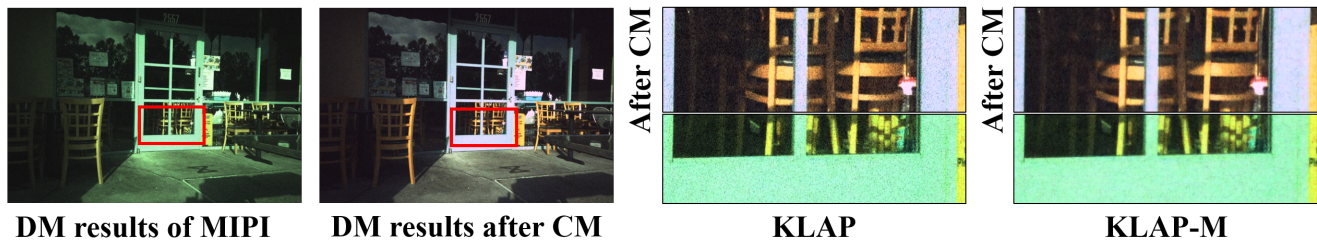


Figure S.5: Results of KLAP and KLAP-M on MIPI '22 Quad.

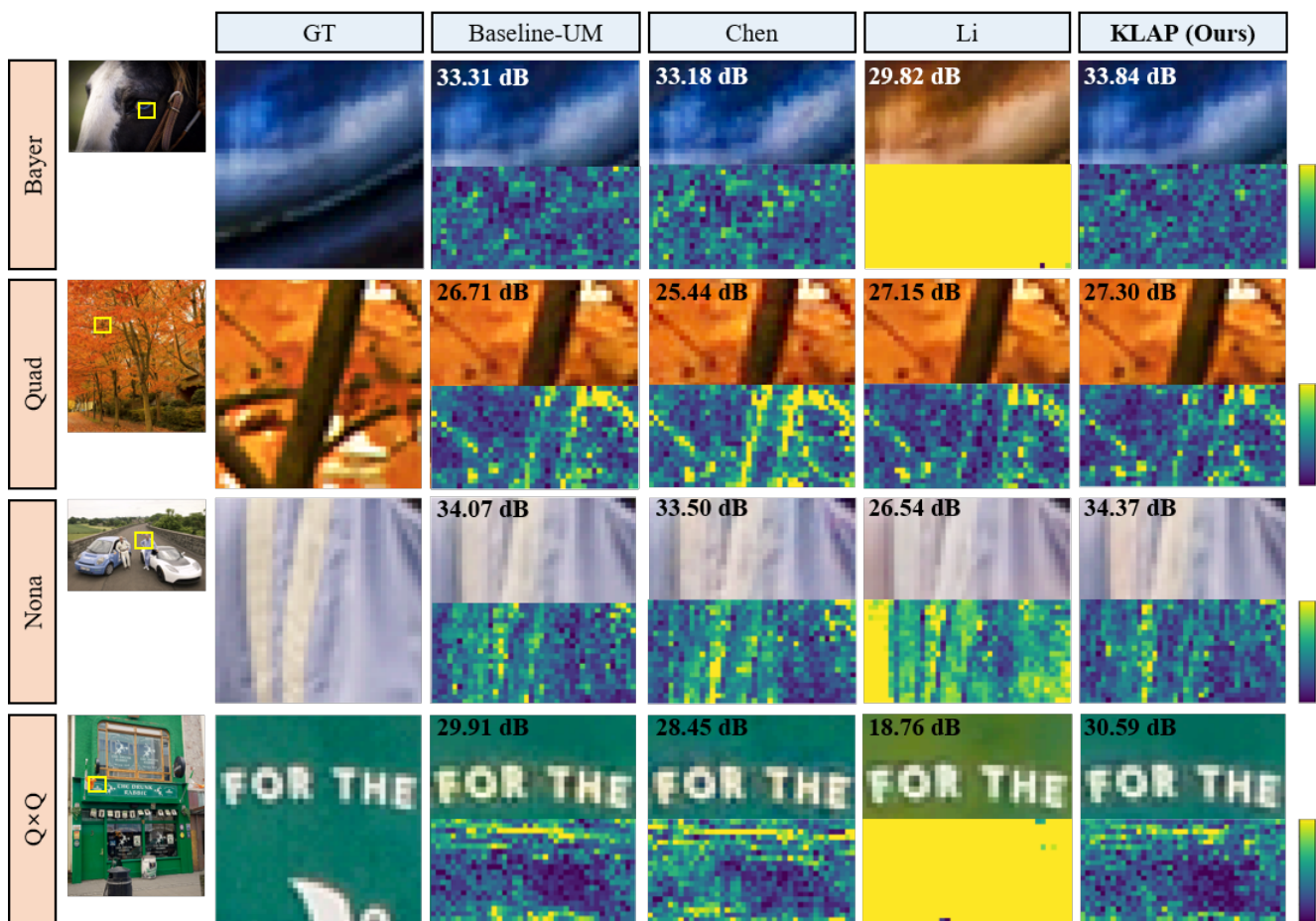


Figure S.6: Comparisons of demosaiced images (**top**) from different methods and their difference maps (**bottom**) on the synthetic RAW (DF2K-CIS) test dataset. The PSNR (dB) values displayed in the top-left corner of each image are calculated using the entire image.

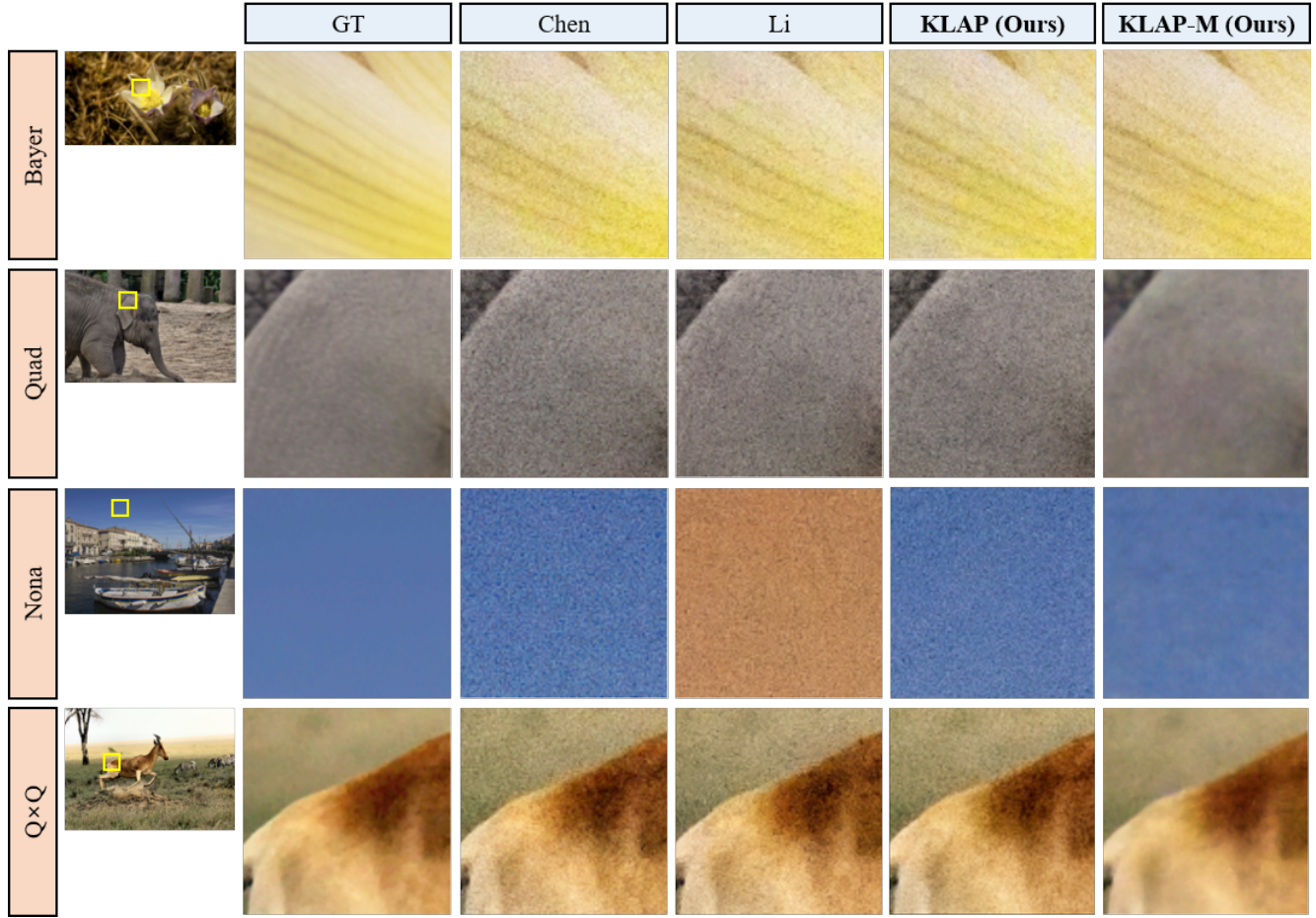


Figure S.7: Comparisons among different methods of robustness on DF2K-CIS with strong noise test dataset. The noise parameters used in the test are four times larger than the noise parameters used in the training. The number of meta-learning iterations in KLAP-M is set to 10, based on empirical determination through visualization of outputs in our experiments.



Figure S.8: The demosaiced output images of $Q \times Q$ CIS RAW (48MP) with various iterations of KLAP-M.

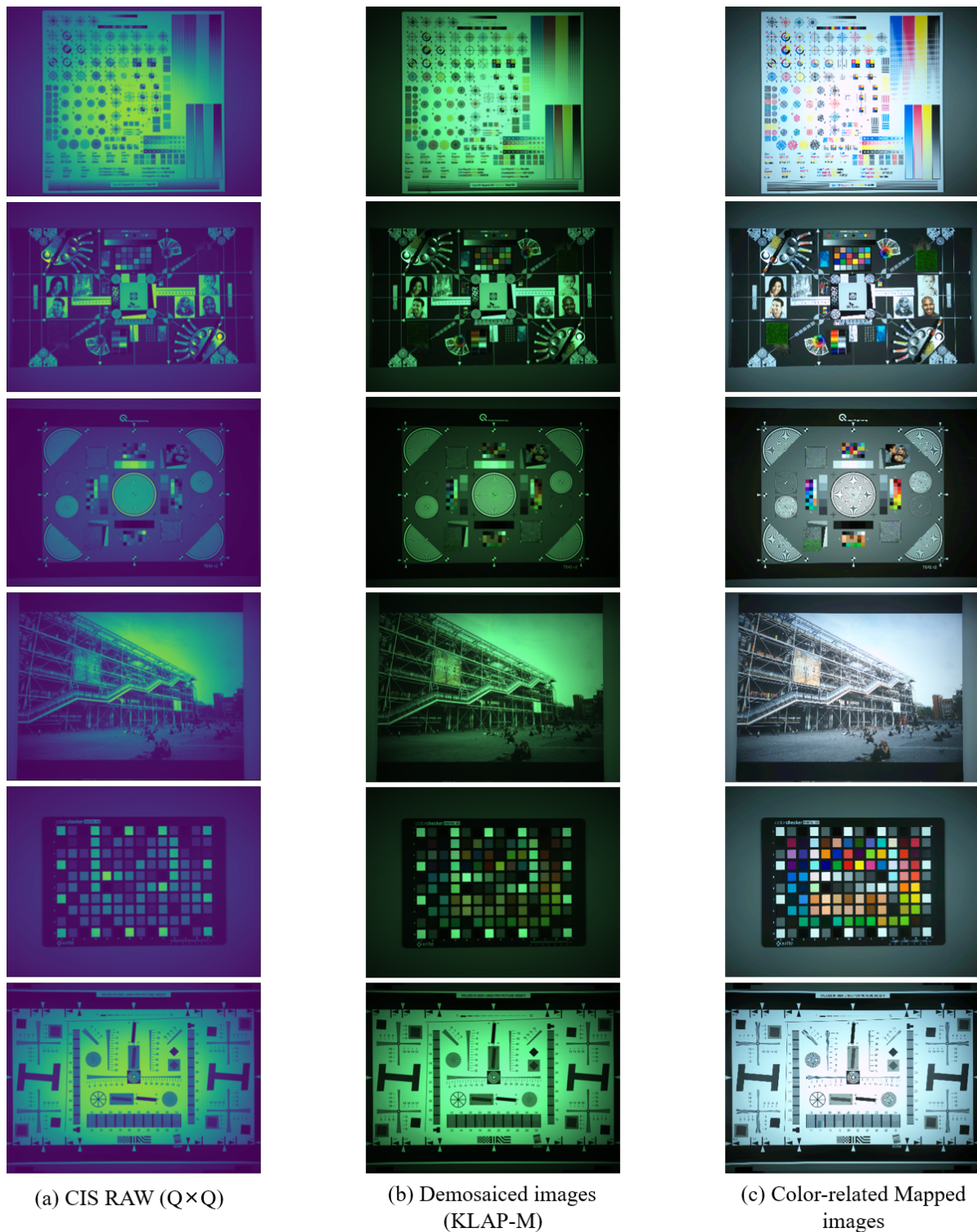


Figure S.9: Additional images of CIS RAW data. (a) CIS $Q \times Q$ RAW data, (b) demosaiced output images obtained using KLAP-M inference, and (c) the same images as in (b) after applying CM (Color-related Mapping function). Note that in (c), it can be perceptually observed that CM works well not only on synthetic RAW images but also on real CIS RAW images.