Leveraging Frame Affinity for sRGB-to-RAW Video De-rendering

Chen Zhang[1*], Wencheng Han[2*], Yang Zhou[1], Jianbing Shen[2†], Cheng-zhong Xu[2], Wentao Liu[1†]
[1]SenseTime Research and Tetras.AI, [2]SKL-IOTSC, CIS, University of Macau.
zhangchen2@tetras.ai {wencheng256, shenjiangbingcg}@gmail.com
{zhouyang, liuwentao}@sensetime.com czxu@um.edu.mo

Abstract

Unprocessed RAW video has shown distinct advantages over sRGB video in video editing and computer vision tasks. However, capturing RAW video is challenging due to limitations in bandwidth and storage. Various methods have been proposed to address similar issues in single image RAW capture through de-rendering. These methods utilize both the metadata and the sRGB image to perform sRGB-to-RAW de-rendering and recover high-quality single-frame RAW data. However, metadata-based methods always require additional computation for online metadata generation, imposing severe burden on mobile camera device for high frame rate RAW video capture. To address this issue, we propose a framework that utilizes frame affinity to achieve high-quality sRGB-to-RAW video reconstruction. Our approach consists of two main steps. The first step, temporal affinity prior extraction, uses motion information between adjacent frames to obtain a reference RAW image. The second step, spatial feature fusion and mapping, learns a pixel-level mapping function using scene-specific and position-specific features provided by the previous frame. Our method can be easily applied to current mobile camera equipment without complicated adaptations or added burden. To demonstrate the effectiveness of our approach, we introduce the first RAW Video De-rendering Benchmark. In this benchmark, our method outperforms state-of-the-art RAW image reconstruction methods, even without image-level metadata.

1. Introduction

In recent years, RAW videos have become increasingly popular among professional videographers due to their ability to capture unprocessed signals from the camera at a 10-16 bit depth. This enables the production of video sequences with superior visual effects compared to sRGB videos. In addition, recent research has explored the benefits of using RAW data in computer vision tasks. Some studies have discovered that the linear relationship between scene irradiance and RAW data is better suited for tasks such as image denoising [2, 9, 12, 39], image super-resolution [7, 13, 34, 40], and low-light enhancement [5, 8, 11]. Additionally, the wide tolerance of RAW data makes perception tasks more robust in high-dynamic scenes [6, 22, 33, 36]. However, the high cost of RAW video capturing limits the widespread use of this technology. On one hand, the abundant and detailed information encoded in the RAW data takes up a huge amount of storage, increasing storage costs for users. On the other hand, real-time capturing of RAW video data puts a significant strain on the camera's bandwidth, as shown in Figure 1 (a).

To address similar issues, many techniques have been

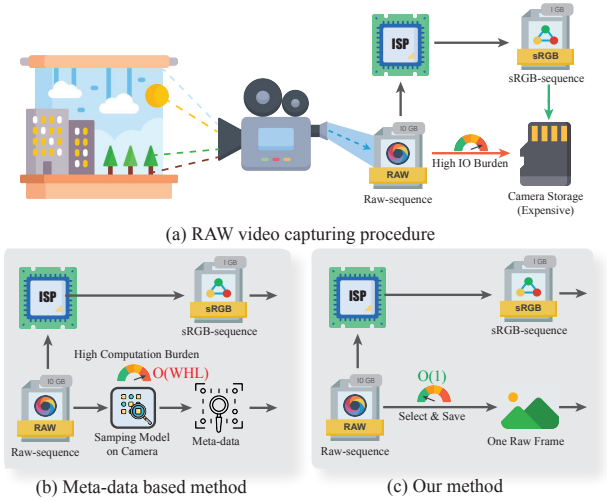

Figure 1. **Comparison of Different RAW Video Capture Methods.** (a) Saving the entire RAW video would cause significant strain on bandwidth and storage resources. (b) Sampling metadata for every frame in the video would burden the camera's computation. (c) Our method only requires saving a single extra RAW frame, which does not significantly burden the camera.

*Equal contribution. †Corresponding author. This work was supported in part by the FDCT grants 0154/2022/A3, 0002/2023/AKP, 0102/2023/RIA2, and 001/2024/SKL, the MYRG-CRG2022-00013-IOTSC-ICI grant and the SRG2022-00023-IOTSC grant.

proposed for de-rendering single-frame sRGB images to RAW data. Some approaches utilize only the sRGB data as input and learn to recover the corresponding RAW data [2, 7, 31]. These methods are convenient for various applications, as they do not require any additional information from the camera besides the sRGB images. However, since some critical information may have been lost during the ISP processing, the de-rendering from only sRGB inputs still suffers from limited performance. In contrast, other methods attempt to extract a small portion of crucial information from the RAW data as metadata and learn to recover the original RAW images from both the sRGB images and the metadata [14, 19, 23, 27]. Although these methods can produce more accurate RAW images, they require an additional sampling method to run on the camera for producing the metadata. These sampling techniques impose additional pressure on the camera, and the burden will be much more severe for RAW video capturing. The complexity of sampling metadata from RAW video is $O(WHL)$, where W and H are the width and height of the image, and L is the length of the video sequence, as shown in Figure 1 (b).

This paper presents a novel approach for RAW video de-rendering, which overcomes the limitations of previous methods and enhances storage and computation efficiency. Unlike metadata-based approaches, our method eliminates the need for an online sampler on the camera side. Instead, it utilizes a single RAW frame from the video sequence as prior information to efficiently recover detailed information from the RAW data, as depicted in Figure 1 (c). The additional select operation only incurs $O(1)$ extra computation cost. Specifically, our method utilizes the provided RAW frame and the relationship between adjacent frames to guide the reconstruction of the remaining RAW frames. By designing a deep neural network with Temporal Affinity Prior Extraction and Spatial Feature Fusion and Mapping, our method effectively leverages the prior information from the previous frame and efficiently de-renders the RAW video.

To assess the effectiveness of our proposed method, we introduce the first RAW Video De-rendering benchmark, the RVD dataset. We then evaluate the performance of the proposed method and compare it with other sRGB-to-RAW de-rendering methods. The results demonstrate that our method can produce high-fidelity RAW video sequences and outperform other methods, without relying on per-image metadata. In conclusion, the contributions we have made in this paper can be summarized into five points:

- We propose a new architecture for RAW video de-rendering. This architecture can efficiently de-render RAW video sequences using only one RAW frame and sRGB videos as input. By adopting this method, both storage and computation efficiency for RAW video capturing can be significantly improved.
- We propose a module for extracting a temporal affinity prior to obtain a reliable reference RAW frame. This module utilizes motion information between adjacent frames. The reference RAW frame can accurately serve as a reference for recovering the lost RAW information.
- We propose a module called Global and Local feature Aggregation (GLA) to extract and combine the globalized and localized features encoded in the prior information.
- We propose a new benchmark for RAW video de-rendering to comprehensively evaluate the methods for this task. To our knowledge, this is the first benchmark specifically designed for the RAW video de-rendering task.
- Our method significantly outperforms state-of-the-art image RAW de-rendering methods and achieves high-quality RAW video sequences without relying on per-image metadata.

2. Related work

2.1. RAW Data Application

RAW data, as a direct capture of raw information from a scene by a camera, has long been favored by computational photography and certain low-level vision tasks. These tasks include image super-resolution [7, 16, 30, 40], image denoising [2, 9, 10, 41–43], image deblur [4, 15], and low-light enhancement [5, 8, 11]. The main advantage of RAW data is that it is not processed by ISP (Image Signal Processor) and its response value has a linear relationship with scene irradiance. This simplifies the modeling process for degradation or enhancement models.

Recently, the advantages of RAW in visual perception tasks have been gradually discovered and explored. One basic point is that high-bit-depth RAW data retains more scene information. Another point is that ISP processing does not always have a positive effect. In the early stages, the work in [3, 17, 29, 35] examined the role of the ISP pipeline in computer vision tasks. An interesting conclusion is that most ISP processing is focused on improving visual effects and has little to do with the DNN model, and may even have a negative impact. Furthermore, some methods [18, 25, 37] used proxy networks or parameter searching to redefine ISP as a learnable process, improving the performance of RAW data in downstream tasks. The work in [6] focused on the instance segmentation task in dark scenes, and its results show that higher bit-depth is usually associated with better segmentation results. Xu et al. [33] proposed a new benchmark for RAW object detection in driving scenes and demonstrated the important influence of dynamic range on the detector.

2.2. sRGB-to-RAW Image De-rendering

There are generally two categories of single image sRGB-to-RAW de-rendering methods, based on whether metadata

is used or not.

De-rendering without metadata. Such methods are often designed for a specific camera or for vision tasks that require imprecise de-rendering. For example, Brooks et al. [2] proposed a generic process to model and invert the key steps of ISP. They then used synthesized RAW data to train a denoising network. Conde et al. [7] improved upon the aforementioned method by modeling the ISP steps as a learnable dictionary. The work in [31] introduced the normalizing flow to this task, relying on the inherent reversibility of this structure. It is approximated by a two-way process of learning rendering and de-rendering.

De-rendering with metadata. For metadata-based methods, the typical approach is to reconstruct the complete RAW image by saving a small amount of sampled data. Yuan and Sun [38] proposed a hybrid capture mode that reduces storage costs and continuous shooting burdens by saving low-resolution RAW images. The method described in [23] saves a small number of uniformly sampled pixels as metadata during capture, and uses RBF interpolation to obtain the full-resolution image during the reconstruction phase. Subsequently, [19] proposed a content-aware superpixel prediction network to improve the sampling strategy, while [14] introduced more expressive implicit neural functions to replace the RBF interpolation algorithm and achieve better performance. In addition to sampling-based methods, Nguyen and Brown's work [20] [21] proposed encoding the necessary parameters of ISP into an sRGB-JPEG file. The work of [27] considered an adaptive and learnable metadata construction method, which uses a compact representation of the latent space as metadata.

Unlike previous single image de-rendering methods, the proposed method does not impose any additional burden on the camera side. Instead, it utilizes inter-frame relationships to guide the de-rendering process, resulting in improved performance and greater efficiency.

3. RAW Video De-rendering Dataset

RAW Video De-rendering is a computer vision task that aims to reconstruct RAW video sequences using sRGB frames and other additional information. This task is valuable as it helps reduce the storage and bandwidth requirements associated with capturing RAW videos. Currently, there is a lack of a dedicated benchmark specifically designed for evaluating and developing RAW video de-rendering methods. To address this issue, we propose the introduction of the first-ever RAW Video De-rendering (RVD) benchmark. This benchmark consists of over 200 videos, captured using different devices in various scenarios. Additionally, we have included videos in challenging scenes, such as fast motion and low lighting, to provide a comprehensive evaluation of de-rendering methods even in extreme conditions. In the remaining content of this section,

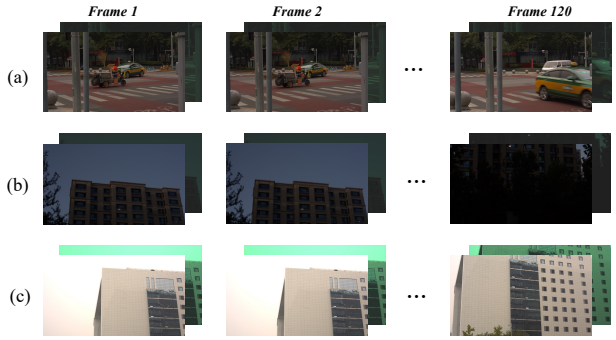

Figure 2. Typical challenging scenarios of RVD dataset. (a) contains fast moving vehicles, (b) has drastic scene change, and (c) is a high-light scenario. For each video, we list the sRGB of the first two frames and the last frame, and the corresponding RAW images which are demosaiced for visualization.

we will provide a detailed introduction to the Data Collection and Data Processing of the proposed benchmark.

Data Collection. We collect the data used in this benchmark from two different sources. In the first part, we gather RAW-sRGB video pairs from existing RAW video-based tasks [40]. There are two reasons why we choose to reuse this data in our benchmark. Firstly, this data is of high quality, captured in various scenarios by different devices. By reusing this data, we can increase the diversity of the proposed benchmark without incurring additional data collection costs. Secondly, these datasets are widely recognized benchmarks for RAW video-based tasks, such as Video RAW Super-resolution. Therefore, this data can be easily used to evaluate the de-rendered RAW videos in these tasks.

However, simply relying on the data from the existing RAW video dataset is not sufficient for evaluating the RAW video de-rendering tasks. These data are not specifically collected for this task and lack comprehensive evaluations. Therefore, we manually collected some new videos from various challenging scenarios for the RAW video de-rendering task. Specifically, we collected these RAW videos using a Canon 5D Mark III DSLR camera. All videos were shot using fixed focus lenses, specifically the Canon EF 50mm f/1.8 STM. We used the automatic setting on the camera, where the aperture, shutter, and ISO were adjusted automatically based on the scene. Each video has a duration of 4 to 6 seconds, resulting in a total of 100 videos. The RAW videos are recorded with a resolution of 1600×900 pixels at a frame rate of 30 frames per second. The camera sensor's color filter array (CFA) follows the widely used RGGB Bayer format, and the RAW data is recorded with a 14-bit depth.

For the scenarios, we select some common challenges in the RAW video de-rendering task. These challenges include fast motion, high/low illumination, and more. After collecting this data, we manually label each video with the corresponding attributes. It is possible for a video to be as-

signed with more than one attribute.

Data Processing. Typically, different camera devices use different Image Signal Processing (ISP) types to generate sRGB images, resulting in significant variations in sRGB image quality. In order to ensure a fair comparison with previous works, we choose not to use the sRGB images produced by the camera itself. Instead, we employ a software ISP algorithm that has been widely used in previous RAW-image datasets to generate the sRGB videos. Our data processing involves two steps: converting the RAW video format data (in .MLV format for Canon 5D Mark III) into RAW image sequences, and using an external ISP algorithm to render the sRGB image sequences.

For the first step, we use MLV App[1] – a professional magic lantern video processing software – to convert the RAW video into a RAW image sequence in DNG format (an open and lossless RAW image format). Since the quality of the first few frames of a handheld shot may not be very high, we delete some of the early frames. For convenience, we retain 120 consecutive frames from each video, resulting in a total of 12,000 retained images for the device. To obtain a rendered sRGB image sequence, we follow the methods described in [31] and [15]. We use the rawpy toolbox, a Python version of the LibRaw library, to process all the RAW images into PNG format using the default settings. This process includes typical operations of a modern ISP, such as demosaicking, auto white balance, denoising, gamma correction, color correction, and more. Some sample images can be seen in Figure 2.

4. Method

4.1. Overview

Given a video sequence x from the sRGB domain, and the corresponding RAW video y from the RAW domain. The sRGB generation procedure f is:

$$x, \varepsilon' = f(y), \tag{1}$$

where ε' represents the extra data saved at capture time. And the invert function of this procedure is defined as the RAW video de-rendering task:

$$y = f^{-1}(x|\varepsilon). \tag{2}$$

In previous works on single image RAW de-rendering, additional data ε typically consists of sampled RAW pixels from the original RAW image, *i.e.* the metadata. This data has proven to be highly effective in recovering lost detailed information during the sRGB generation process. However, when it comes to video de-rendering, sampling additional metadata from every frame in the RAW video could impose an excessively high computational burden on the camera, making implementation difficult. To alleviate the need for extra computational burden on the camera device and boost de-rendering efficiency, we propose a new pipeline for the RAW video de-rendering. In this pipeline, we utilize the first RAW frame from the RAW video y as additional data:

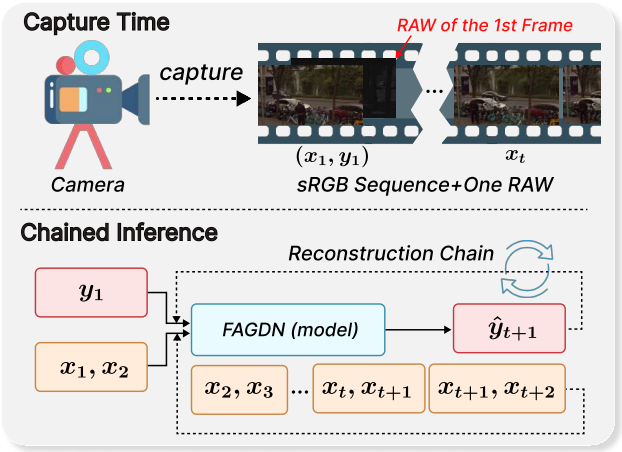

Figure 3. Video capture and inference procedure. In the capture time, we save the first frame RAW image and complete sRGB sequence. In the inference phase, except for the first frame, which uses the real RAW image, the rest use the predicted RAW image from the previous frame.

$$\varepsilon = y_1. \tag{3}$$

Figure 3 shows an overview of the proposed pipeline. When capturing the video, we only need the camera to save the sRGB frames and an additional first RAW frame, which is easily achievable. When de-rendering the RAW video, our pipeline employs a chained structure to accomplish this. Specifically, in the initialization step, our model takes sRGB frames x_1, x_2 and the first RAW frame y_1 as input and produces the well-de-rendered RAW frame $\hat{y}_2$ as output. In the subsequent steps, our model takes x_t, x_{t+1}, and the $\hat{y}_t$ produced in the last step as input and produces $\hat{y}_{t+1}$ until the entire RAW video is de-rendered. As shown in this figure, the Frame Affinity Guided De-rendering Network (FAGDN) is the core of our pipeline. In the following subsection, we will provide a detailed introduction to it.

4.2. Frame Affinity Guided De-rendering Network

Figure 4 illustrates the structure of the proposed Frame Affinity Guided De-rendering Network, which consists of two main steps. The first step is the extraction of temporal affinity priors. In this step, we combine the motion relationship between adjacent frames and the input from the previous frame RAW to construct a reference RAW. This is based on the observation that most parts of adjacent frames describe similar scenes. Therefore, the pixels in the previous RAW frame can provide accurate prior information for de-rendering the corresponding pixels in the subsequent frames.

The second step involves spatial feature fusion and mapping. Using the reference RAW extracted in the first step as auxiliary information, we utilize an encoder-decoder struc-

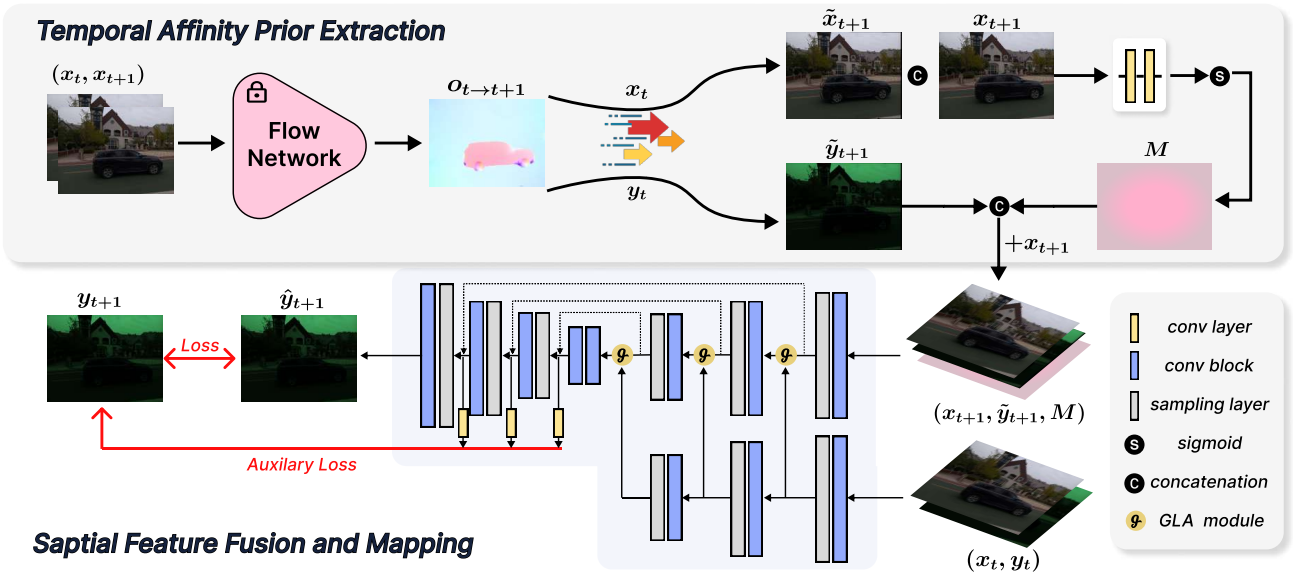

Figure 4. Overview of the proposed Frame Affinity Guided De-rendering network. It contains two main steps, the Temporal Affinity Prior Extraction step generates an reference RAW by utilizing the motion information between adjacent frames. Here, we evaluate the quality of reference RAW through a confidence map. The second step is Spatial Feature Fusion and Mapping, using the reference RAW as starting status, we learn a pixel-level mapping function with the help of sRGB image and previous frames to further repair pixels predicted inaccurately in the first stage.

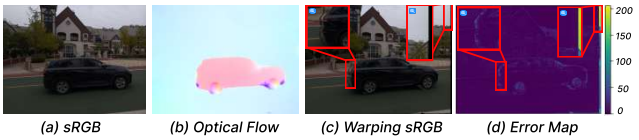

Figure 5. From left to right are the sRGB image x_t, the optical flow image $o_{t\to t+1}$, the warping sRGB image $\tilde{x}_{t+1}$ and the error map of $\tilde{x}_{t+1}$ and x_{t+1}. The red box lists errors caused by the optical flow algorithm.

ture to learn a mapping function of sRGB-to-RAW by incorporating the previous frame. To effectively utilize the scene-specific and position-specific prior information encoded in the previous frame, we introduce a Global and Local Feature Aggregation (GLA) module. This module is integrated with the encoder-decoder network, which will be further discussed later.

Temporal Affinity Prior Extraction. In order to take full advantage of the temporal relationship between the adjacent frames, we need to generate the affinity between two frames. We accomplish this by using an off-the-shelf optical flow algorithm Θ [32]. This algorithm takes the sRGB frames (x_t, x_{t+1}) as input and produces the transfer matrix $o_{t\to t+1}$:

$$o_{t\to t+1} = \Theta(x_t, x_{t+1}). \tag{4}$$

Using this transfer matrix and the previous RAW image y_t, we can apply a wrapping technique to generate a warping RAW image $\tilde{y}_{t+1}$ as an reference RAW:

$$\tilde{y}_{t+1} = \Omega(y_t, o_{t\to t+1}), \tag{5}$$

where Ω represents a warping function that applies an offset to the original pixels to obtain new coordinates. It then completes the image using an interpolation algorithm.

However, there is a natural limitation where some pixels in the subsequent frame do not have corresponding pixels in the previous frame, making the reference pixels unreliable. Figure 5 illustrates a typical example of this limitation. To overcome this issue, we propose generating a confidence map to suppress areas where matching errors occur and select reliable reference features from the reference RAW frame. This generation process is based on the observation that regions with matching errors result in high error values when warping the sRGB image, as shown in Figure 5 (d). Subsequently, we utilize a multi-layer convolutional network to generate the confidence map M:

$$\begin{aligned} \tilde{x}_{t+1} &= \Omega(x_t, o_{t\to t+1}), \\ M &= \sigma(conv([\tilde{x}_{t+1}, x_{t+1}])), \end{aligned} \tag{6}$$

where, the $conv$ and σ represent the convolutional network and sigmoid function, respectively. The reference RAW image $\tilde{y}_{t+1}$ and confidence map M will be fed to the second step for accurate RAW image reconstruction.

Spatial Feature Fusion and Mapping. Using the current sRGB frame and reference RAW, we develop a neural network that learns the final RAW image de-rendering process. Following the previous method for sRGB-to-RAW de-rendering [19], we utilize an encoder-decoder structure, specifically U-Net [24] like structure, to learn the mapping function in an end-to-end manner.

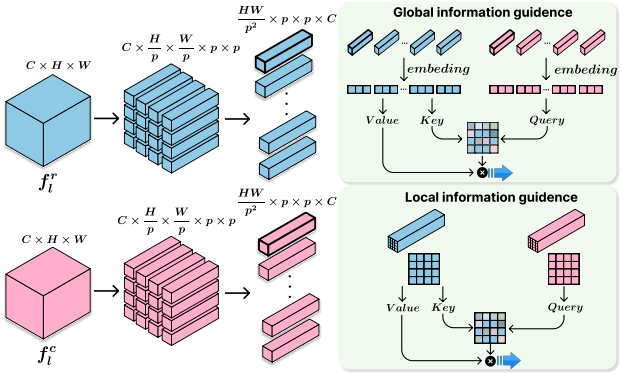

Figure 6. The implementation of Global and Local Feature Aggregation(**GLA**) module. Global information guidance models the dependencies between all patches, while local information guidance limits the relationship modeling within each pair of patches.

However, the mapping function often presents two types of challenges. Firstly, the sRGB-to-RAW mapping is scene-specific, meaning that even the same RAW pixel can be mapped to different sRGB values under different scenes. Secondly, the sRGB-to-RAW mapping is position-specific, as described in [23], due to local operations in the ISP. This means that the same RAW value can be mapped to different sRGB values even within a single image. Therefore, our structure includes two branches in the encoder. The first branch takes the current frame information $[x_{t+1}, \tilde{y}_{t+1}, M]$ as input, providing realistic information about the current status, such as lighting conditions and object position. The second branch takes the previous frame input $[x_t, y_t]$, aiming to extract global (scene-specific) and local (position-specific) prior information from the previous RAW and sRGB pair.

As shown in Figure 4, we utilize two encoder networks to extract deep features from the two input branches individually. These branches have identical structures but do not share parameters:

$$\begin{aligned} f_l^r &= \text{Encoder}_{\text{ref}}([x_t, y_t]), \\ f_l^c &= \text{Encoder}_{\text{cur}}([x_{t+1}, y_{t+1}, M]), \end{aligned} \tag{7}$$

where Encoder represents encoder network. The intermediate features are denoted as f_l^r and f_l^c, where l indexes the layer of the network. To efficiently utilize the features of the previous frame from global and local perspectives, we propose a global and local feature aggregation module, as shown in Figure 6. Specifically, for the feature $f_l^r \in \mathbb{R}^{C\times H\times W}$ and $f_l^c \in \mathbb{R}^{C\times H\times W}$, we divide them into non-overlapping patches with a size of $C \times p \times p$. To introduce scene-related features from a global perspective, we embed each patch as a vector, forming a sequence of length $\frac{HW}{p^2}$. Similar to the transformer approach[26], we use the embedding vector of the current frame as the *Query* and the embedding vector of the previous frame as the *Key* to learn an affinity matrix. This matrix is then used to extract global relevant information from the previous frame features (*Value*). For local information guidance, we implement information interaction within each patch pair. Similarly, we still use the current frame as the *Query* vector to model pixel-level dependencies. The information interaction within the patch allows the network to pay attention to neighboring pixels, making use of the video characteristics and reducing computational cost. Finally, the global and local supplementary information is added to each layer of the $\text{Encoder}_{\text{cur}}$ sequentially.

During the decoding stage, we use an equal number of convolutional and sampling layers to gradually merge features and restore resolution. At the same time, we incorporate the multi-level features of the current encoder ($\text{Encoder}_{\text{cur}}$) into the decoding process through skip connections. The resulting output of the decoder is considered the final reconstructed RAW image ($\hat{y}_{t+1}$).

Loss Function. The overall network is trained in an end-to-end manner, using ℓ, the combination of L_2 loss and $SSIM$ [28] loss, to measure the accuracy of the reconstructed RAW. In addition to supervising the output of the last layer of the decoder, we also impose additional constraints on the intermediate features. The entire loss function can be expressed by the following formula:

$$\mathcal{L} = \ell(\hat{y}_{t+1}, y_{t+1}) + \lambda \sum_{n=1}^{3} \ell(\hat{y}_{t+1}^n, y_{t+1}^n). \tag{8}$$

$\hat{y}_{t+1}^n$ denotes the output of different levels of the decoder, while y_{t+1}^n is the downsampled image of the ground truth, maintaining the same resolution. λ is a hyperparameter used to balance the contribution of the auxiliary loss and is set to 0.5.

5. Experiments

5.1. Experimental Setup

Dataset. We conduct experiments on the proposed RVD dataset. For convenience, we call the actually collected data RVD-Part1, and the data collected from [40] is called RVD-Part2. For RVD-Part1, we carefully select 15 videos (including difficult scenes such as fast motion, drastic scene changes, HDR) for testing, and remaining 85 videos for training. Each video is 120 frames with 1600×900 resolution. For the RVD-part2, we maintain the original division of training sets and test sets. Specifically, the training set contains 130 videos, each video has about 50 frames, and a total of 6308 images. The testing set has 20 videos, and 983 images. All images have resolution 1440×640.

Implementation details. We are committed to converting sRGB to the original RAW data, which has not undergone any ISP operations such as demosaic and AWB. As a common practice, we package the single-channel RAW data into a four-channel [R, Gr, Gb, B] format, which serves as the

Method	Pub.	Year	Metadata	RVD-Part1		RVD-Part2	
				PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
CAM [19]	CVPR	2022	✓	41.85	0.9832	42.25	0.9856
CAM w/ fine-tuning [19]	CVPR	2022	✓	41.87	0.9836	42.26	0.9858
INF [14]	CVPR	2023	✓	47.67	0.9948	46.25	0.9939
Ours	-	-	✗	**48.86**	**0.9980**	**49.71**	**0.9983**

Table 1. Quantitative comparison with sRGB-to-RAW image de-rendering methods on the proposed RVD dataset. The ↑ represents that the larger the value, the better, and best results are marked in bold.

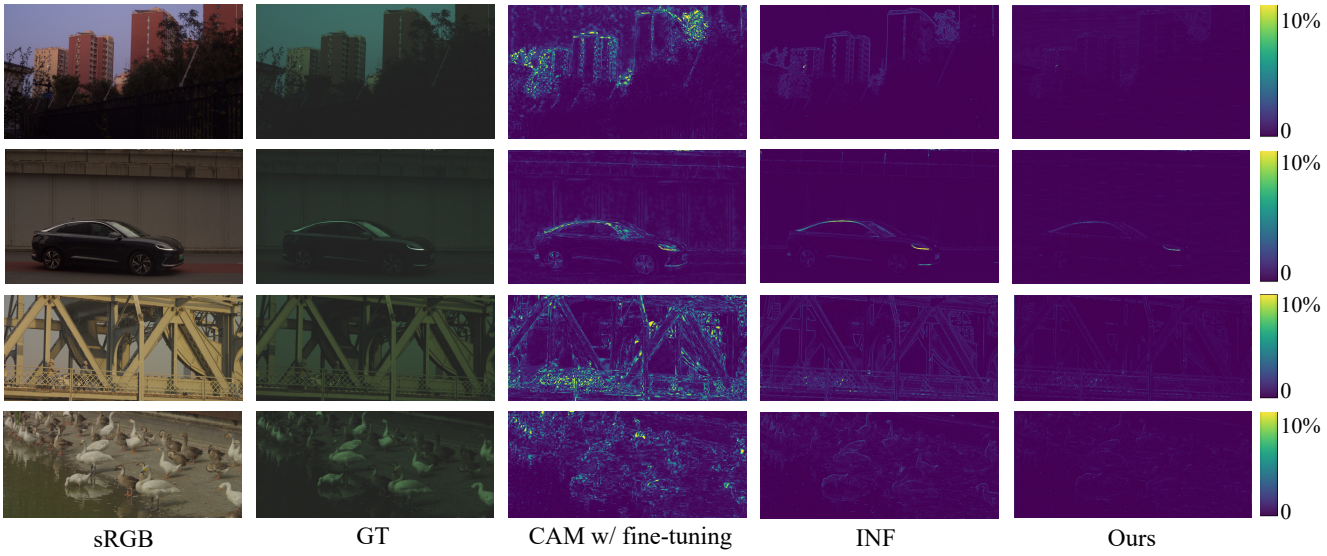

Figure 7. Visual comparison with sRGB-to-RAW image de-rendering methods. We calculate the pixel-level error maps of different methods and GT, and the RAW images are demosaiced for visualization.

basic format for training and inference. It is also easy to unpack it back to single-channel RAW for downstream tasks. The proposed method is implemented using the PyTorch framework. During training, we randomly crop patches of size 256×256 from the original input and apply random rotation and flipping as data augmentation techniques. The model is optimized using the Adam optimizer, with an initial learning rate of $1e-3$. The learning rate is reduced to one-tenth every 20 epochs. We train the model for a total of 60 epochs and select the model from the last epoch for testing. During testing, we maintain the original resolution of the image as the input. Considering the differences in sensors among different devices, we train and test the model separately on each subset.

5.2. Comparison with Image De-rendering Methods

Since our method is the first proposed for sRGB-to-RAW video de-rendering, we compare it with state-of-the-art sRGB-to-RAW image de-rendering methods: CAM [19], CAM with fine-tuning [19], and INF [14]. The source code of these methods has been made public, and since most of the sRGB-to-RAW methods are camera-dependent, we retrain and test on the new dataset.

Table 1 illustrates the quantitative comparison results of different methods. We use peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) metrics to evaluate the quality of reconstructed RAW. The results demonstrate that the proposed method achieves the best performance, even when compared to per-image metadata-based methods. Moreover, we also compute the pixel-wise difference between the reconstructed RAW and the ground truth to visually compare the performance of different methods, which can be seen in Figure 7. Overall, our method achieves satisfactory results in most scenarios. We also notice that the sampling-based method INF obtains comparable results to our method but performs poorly in the edge areas of objects. This means that interpolation-based functions have difficulty processing areas with large gradient changes.

5.3. Comparison under Challenging Scenarios

In the actual photography process, the camera faces complex and diverse scenes. Therefore, we aim to explore the robustness of the model in different challenging scenarios. **Low resolution** is a common phenomenon in post-processing or when using low-cost storage. Previous research [14] has shown that sampling-based methods tend

Condition	Method	PSNR	SSIM
Low resolution	Ours	**47.90**	**0.9978**
	INF	45.50	0.9944
Fast motion	Ours	**46.36**	**0.9976**
	INF	45.79	0.9936
Dark lighting	Ours	**53.60**	**0.9989**
	INF	51.42	0.9969

Table 2. Comparison under challenging scenarios on RVD-Part1.

warping	M	GLA	ℓ_{aux}	PSNR	SSIM
		✓	✓	42.60	0.9972
✓		✓	✓	47.68	0.9978
✓	✓		✓	46.12	0.9979
✓	✓	✓		47.91	0.9983
✓	✓	✓	✓	**49.71**	**0.9983**

Table 3. Ablation study on RVD-Part2 dataset.

to be sensitive to resolution. To test the performance of our method and INF, we construct a low-resolution version by downsampling the original dataset to half the resolution. The results indicate that our method is more robust to resolution changes, with a 1.8% PSNR drop compared to the 4.6% drop observed in INF. **Fast motion** is often regarded as a challenging scene for our method, as the first RAW frame only provides limited information for subsequent RAW reconstruction. In comparison to INF, our method shows a gain of 1.2% in the fast motion scene, which is lower than the 2.3% gain observed on the full test set. This suggests that fast motion still poses a challenge for our method. **Dark lighting** is a typical harsh environment for shooting. However, our research has found that it is not actually difficult for the sRGB-to-RAW methods, as the recovery quality in these scenarios is well above average.

5.4. Ablation Study

We conduct ablation experiments to verify the effectiveness of key designs in the proposed method, and the quantitative results are shown in Table 3. We first investigate the role of the temporal affinity prior extraction, which generates a reference RAW image by utilizing the motion information. We remove this step and directly input the sRGB image x_{t+1} into $\text{Encoder}_{\text{cur}}$ to do the verification. The results in first line show this step is all-important to our pipeline, that is, PSNR decreased by 6.17. The second line represents that we do not consider the quality of the RAW and directly input the reference RAW image $\tilde{y}_{t+1}$ to the next stage.

In the spatial feature fusion and mapping step, the previous frame provides global and local supplementary information for image reconstruction by the GLA module. To demonstrate the necessity of introducing a previous frame into this step, we remove the previous frame encoder $\text{Encoder}_{\text{ref}}$ and directly learn the mapping function from the current frame inputs. The result can be seen in the third line of Table 3. Furthermore, we add the auxiliary loss function to constraint the intermediate outputs, and the results can be seen in the fourth line when deleting it from the pipeline.

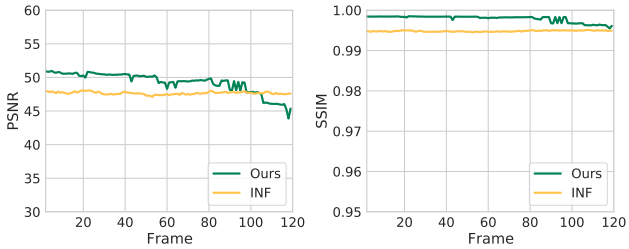

Figure 8. Average PSNR and SSIM in time dimension.

5.5. Robustness over Time

Our method relies on the first frame RAW to provide real reference information in the chained inference phase. However, as time increases, there will inevitably be accumulated information loss. In Figure 8, we calculate the average PSNR ($psnr^i = (\sum_{v=1}^{V} psnr_v^i)/V$) and SSIM ($ssim^i = (\sum_{v=1}^{V} ssim_v^i)/V$) along the time dimension. Here, $i \in [2, 120]$ indexes the order of frames, and V represents the number of all testing videos.

As depicted in the curve, our method exhibits good robustness over time within a relatively long period of 100 frames. In comparison to INF, the method based on single-image metadata, our method still maintains a relatively high quality. However, the performance noticeably declines when the period becomes too long. This phenomenon is attributed to the inherent characteristics of our method. As an optional solution and for future work, we intend to investigate a previous frame selection strategy. This strategy would involve saving reference RAW frames at intervals to achieve a stable reconstruction quality.

6. Conclusion

In this paper, we introduce a new task called sRGB-to-RAW video de-rendering. Existing methods for converting sRGB images to RAW format are not applicable when applied to the new video de-rendering task. To address this, we propose a frame affinity guided reconstructed network with two main steps. The first step involves extracting temporal affinity prior by utilizing motion information between adjacent frames to obtain a reference RAW image. The second step focuses on spatial feature fusion and mapping, where a pixel-level mapping function is learned using scene-specific and position-specific prior information from the previous frame. To evaluate our proposed method, we create a benchmark dataset that includes a wide range of RAW-sRGB videos. Experimental results on this dataset demonstrate the superior reconstruction quality achieved by our method.

References

[1] https://mlv.app/.

[2] Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11036–11045, 2019.

[3] Mark Buckler, Suren Jayasuriya, and Adrian Sampson. Reconfiguring the imaging pipeline for computer vision. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 975–984, 2017.

[4] Mingdeng Cao, Zhihang Zhong, Yanbo Fan, Jiahao Wang, Yong Zhang, Jue Wang, Yujiu Yang, and Yinqiang Zheng. Towards real-world video deblurring by exploring blur formation process. In *European Conference on Computer Vision*, pages 327–343. Springer, 2022.

[5] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3291–3300, 2018.

[6] Linwei Chen, Ying Fu, Kaixuan Wei, Dezhi Zheng, and Felix Heide. Instance segmentation in the dark. *International Journal of Computer Vision*, pages 1–21, 2023.

[7] Marcos V Conde, Steven McDonagh, Matteo Maggioni, Ales Leonardis, and Eduardo Pérez-Pellitero. Model-based image signal processors via learnable dictionaries. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 481–489, 2022.

[8] Xingbo Dong, Wanyan Xu, Zhihui Miao, Lan Ma, Chao Zhang, Jiewen Yang, Zhe Jin, Andrew Beng Jin Teoh, and Jiajun Shen. Abandoning the bayer-filter to see in the dark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17431–17440, 2022.

[9] Hansen Feng, Lizhi Wang, Yuzhi Wang, and Hua Huang. Learnability enhancement for low-light raw denoising: Where paired real data meets noise modeling. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1436–1444, 2022.

[10] Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand. Deep joint demosaicking and denoising. *ACM Transactions on Graphics (ToG)*, 35(6):1–12, 2016.

[11] Haofeng Huang, Wenhan Yang, Yueyu Hu, Jiaying Liu, and Ling-Yu Duan. Towards low light enhancement with raw images. *IEEE Transactions on Image Processing*, 31:1391–1405, 2022.

[12] Xin Jin, Jia-Wen Xiao, Ling-Hao Han, Chunle Guo, Ruixun Zhang, Xialei Liu, and Chongyi Li. Lighting every darkness in two pairs: A calibration-free pipeline for raw denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13275–13284, 2023.

[13] Bruno Lecouat, Jean Ponce, and Julien Mairal. Lucas-kanade reloaded: End-to-end super-resolution from raw image bursts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2370–2379, 2021.

[14] Leyi Li, Huijie Qiao, Qi Ye, and Qinmin Yang. Metadata-based raw reconstruction via implicit neural functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18196–18205, 2023.

[15] Chih-Hung Liang, Yu-An Chen, Yueh-Cheng Liu, and Winston H Hsu. Raw image deblurring. *IEEE Transactions on Multimedia*, 24:61–72, 2020.

[16] Xiaohong Liu, Kangdi Shi, Zhe Wang, and Jun Chen. Exploit camera raw data for video super-resolution via hidden markov model inference. *IEEE Transactions on Image Processing*, 30:2127–2140, 2021.

[17] Zhenhong Liu, T Park, HS Park, and NS Kim. Ultra-low-power image signal processor for smart camera applications. *Electronics Letters*, 51(22):1778–1780, 2015.

[18] Ali Mosleh, Avinash Sharma, Emmanuel Onzon, Fahim Mannan, Nicolas Robidoux, and Felix Heide. Hardware-in-the-loop end-to-end optimization of camera image processing pipelines. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7529–7538, 2020.

[19] Seonghyeon Nam, Abhijith Punnappurath, Marcus A Brubaker, and Michael S Brown. Learning srgb-to-raw-rgb de-rendering with content-aware metadata. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17704–17713, 2022.

[20] Rang MH Nguyen and Michael S Brown. Raw image reconstruction using a self-contained srgb-jpeg image with only 64 kb overhead. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1655–1663, 2016.

[21] Rang MH Nguyen and Michael S Brown. Raw image reconstruction using a self-contained srgb–jpeg image with small memory overhead. *International journal of computer vision*, 126:637–650, 2018.

[22] Emmanuel Onzon, Fahim Mannan, and Felix Heide. Neural auto-exposure for high-dynamic range object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7710–7720, 2021.

[23] Abhijith Punnappurath and Michael S Brown. Spatially aware metadata for raw reconstruction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 218–226, 2021.

[24] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[25] Ethan Tseng, Felix Yu, Yuting Yang, Fahim Mannan, Karl ST Arnaud, Derek Nowrouzezahrai, Jean-François Lalonde, and Felix Heide. Hyperparameter optimization in black-box image processing using differentiable proxies. *ACM Trans. Graph.*, 38(4):27–1, 2019.

[26] 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.

[27] Yufei Wang, Yi Yu, Wenhan Yang, Lanqing Guo, Lap-Pui Chau, Alex C Kot, and Bihan Wen. Raw image reconstruction with learned compact metadata. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18206–18215, 2023.

[28] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P. Simoncelli. Image quality assessment: From error

visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[29] Chyuan-Tyng Wu, Leo F Isikdogan, Sushma Rao, Bhavin Nayak, Timo Gerasimow, Aleksandar Sutic, Liron Ainkedem, and Gilad Michael. Visionisp: Repurposing the image signal processor for computer vision applications. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 4624–4628. IEEE, 2019.

[30] Wenzhu Xing and Karen Egiazarian. End-to-end learning for joint image demosaicing, denoising and super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3507–3516, 2021.

[31] Yazhou Xing, Zian Qian, and Qifeng Chen. Invertible image signal processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6287–6296, 2021.

[32] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, Fisher Yu, Dacheng Tao, and Andreas Geiger. Unifying flow, stereo and depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[33] Ruikang Xu, Chang Chen, Jingyang Peng, Cheng Li, Yibin Huang, Fenglong Song, Youliang Yan, and Zhiwei Xiong. Toward raw object detection: A new benchmark and a new model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13384–13393, 2023.

[34] Xiangyu Xu, Yongrui Ma, Wenxiu Sun, and Ming-Hsuan Yang. Exploiting raw images for real-scene super-resolution. *IEEE transactions on pattern analysis and machine intelligence*, 44(4):1905–1921, 2020.

[35] Masakazu Yoshimura, Junji Otsuka, Atsushi Irie, and Takeshi Ohashi. Dynamicisp: dynamically controlled image signal processor for image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12866–12876, 2023.

[36] Masakazu Yoshimura, Junji Otsuka, Atsushi Irie, and Takeshi Ohashi. Rawgment: noise-accounted raw augmentation enables recognition in a wide variety of environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14007–14017, 2023.

[37] Ke Yu, Zexian Li, Yue Peng, Chen Change Loy, and Jinwei Gu. Reconfigisp: Reconfigurable camera image processing pipeline. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4248–4257, 2021.

[38] Lu Yuan and Jian Sun. High quality image reconstruction from raw and jpeg image pair. In *2011 International Conference on Computer Vision*, pages 2158–2165. IEEE, 2011.

[39] Huanjing Yue, Cong Cao, Lei Liao, Ronghe Chu, and Jingyu Yang. Supervised raw video denoising with a benchmark dataset on dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2301–2310, 2020.

[40] Huanjing Yue, Zhiming Zhang, and Jingyu Yang. Real-rawvsr: Real-world raw video super-resolution with a benchmark dataset. In *European Conference on Computer Vision*, pages 608–624. Springer, 2022.

[41] Feng Zhang, Bin Xu, Zhiqiang Li, Xinran Liu, Qingbo Lu, Changxin Gao, and Nong Sang. Towards general low-light raw noise synthesis and modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10820–10830, 2023.

[42] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7):3142–3155, 2017.

[43] Yi Zhang, Hongwei Qin, Xiaogang Wang, and Hongsheng Li. Rethinking noise synthesis and modeling in raw denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4593–4601, 2021.