

# Beyond Joint Demosaicking and Denoising: An Image Processing Pipeline for a Pixel-bin Image Sensor

S M A Sharif <sup>1</sup>, Rizwan Ali Naqvi <sup>2 \*</sup>, Mithun Biswas <sup>1</sup>

<sup>1</sup> Rigel-IT, Bangladesh, <sup>2</sup> Sejong University, South Korea

{sma.sharif.cse, mithun.bishwash.cse}@ulab.edu.bd, rizwanali@sejong.ac.kr

## Abstract

*Pixel binning is considered one of the most prominent solutions to tackle the hardware limitation of smartphone cameras. Despite numerous advantages, such an image sensor has to appropriate an artefact-prone non-Bayer colour filter array (CFA) to enable the binning capability. Contrarily, performing essential image signal processing (ISP) tasks like demosaicking and denoising, explicitly with such CFA patterns, makes the reconstruction process notably complicated. In this paper, we tackle the challenges of joint demosaicing and denoising (JDD) on such an image sensor by introducing a novel learning-based method. The proposed method leverages the depth and spatial attention in a deep network. The proposed network is guided by a multi-term objective function, including two novel perceptual losses to produce visually plausible images. On top of that, we stretch the proposed image processing pipeline to comprehensively reconstruct and enhance the images captured with a smartphone camera, which uses pixel binning techniques. The experimental results illustrate that the proposed method can outperform the existing methods by a noticeable margin in qualitative and quantitative comparisons. Code available: [https://github.com/sharif-apu/BJDD\\_CVPR21](https://github.com/sharif-apu/BJDD_CVPR21).*

## 1. Introduction

Smartphone cameras have illustrated a significant altitude in the recent past. However, the compact nature of mobile devices noticeably impacts the image quality compared to their DSLR counterparts [15]. Also, such inevitable hardware limitations, holding back the original equipment manufacturers (OEMs) to achieve a substantial jump in the dimension of the image sensors. In contrast, the presence of a bigger sensor in any camera hardware can drastically improve the photography experience, even in stochastic lighting conditions [26]. Consequently, numerous OEMs have exploited pixel enlarging techniques known as pixel binning

in their compact devices to deliver visually admissible images [4, 43].

In general, pixel binning aims to combine the homogeneous neighbour pixels to form a larger pixel [1]. Therefore, the device can exploit a larger sensor dimension outwardly incorporating an actual bigger sensor. Apart from leveraging a bigger sensor size in challenging lighting conditions, such image sensor design also has substantial advantages. Among them, capture high-resolution contents, producing a natural bokeh effect, enable digital zoom by cropping an image, etc., are noteworthy. This study denotes such image sensors as a pixel-bin image sensor.

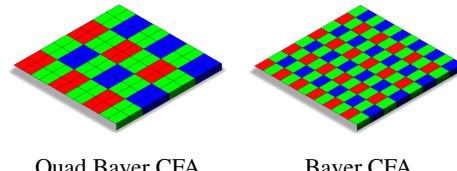


Figure 1: Commonly used CFA patterns of pixel-bin image sensors.

Despite the widespread usage in recent smartphones, including Oneplus Nord, Galaxy S20 FE, Xiaomi Redmi Note 8 Pro, Vivo X30 Pro, etc., reconstructing RGB images from a pixel-bin image sensor is notably challenging [18]. Expressly, the pixel binning techniques have to employ a non-Bayer CFA [22, 18] along with a traditional Bayer CFA [5] over the image sensors to leverage the binning capability. Fig. 1 depicts the most commonly used CFA patterns combination used in recent camera sensors. Regrettably, the non-Bayer CFA (i.e., Quad Bayer CFA [19]) has to appropriate in pixel-bin image sensors is notoriously vulnerable to produce visually disturbing artefacts while reconstructing images from the given CFA pattern [18]. Hence, combining fundamental low-level ISP tasks like denoising and demosaicking on an artefact-prone CFA make the reconstruction process profoundly complicated.

Contrarily, the learning-based methods have illustrated distinguished progression in performing image reconstruc-

\*Corresponding author



Figure 2: Example of Joint demosaicing and denoising on Quad Bayer CFA.

tion tasks. Also, they have demonstrated substantial advantages of combining low-level tasks such as demosaicing along with denoising [12, 21, 25, 9]. Most notably, some of the recent convolutional neural network (CNN) based methods [35, 16] attempt to mimic complicated mobile ISP and substantiate significant improvement in perceptual quality over traditional methods. Such computational photography advancements inspired this study to tackle the challenging JDD of a pixel-bin image sensor and go beyond.

This study introduces a novel learning-based method to perform JDD in commonly used CFA patterns (i.e., Quad Bayer CFA [19], and Bayer CFA [5]) of pixel-bin image sensors. The proposed method leverage spatial and depth-wise feature attention [40, 14] in a deep architecture to reduce visual artefacts. We have denoted the proposed deep as a pixel-bin image processing network (PIPNet) in the rest of the paper. Apart from that, we introduced a multi-term guidance function, including two novel perceptual losses to guide the proposed PIPNet for enhancing the perceptual quality of reconstructed images. Fig. 2 illustrates an example of the proposed method’s JDD performance on a non-Bayer CFA. The feasibility of the proposed method has extensively studied with diverse data samples from different colour spaces. Later, we stretched our proposed pipeline to reconstruct and enhance the images of actual pixel-bin image sensors.

The contribution of this study has summarized below:

- Proposes a learning-based method, which aims to tackle the challenging JDD on a pixel-bin image sensor.
- Proposes a deep network that exploits depth-spatial feature attentions and is guided by a multi-term objective function, including two novel perceptual losses.
- Stretches the proposed method to study the feasibility of enhancing perceptual image quality along with JDD on actual hardware.

## 2. Related work

This section briefly reviews the works that are related to the proposed method.

**Joint demosaicing and denoising.** Image demosaicing is considered a low-level ISP task, aiming to reconstruct RGB images from a given CFA pattern. However, in practical application, the image sensors’ data are contaminated with noises, which directly costs the demosaicing process by deteriorating final reconstruction results [25]. Therefore, the recent works emphasize performing demosaicing and denoising jointly rather than traditional sequential approaches.

In general, JDD methods are clustered into two major categories: optimization-based methods [13, 37] and learning-based methods [12, 9, 21]. However, the later approach illustrates substantial momentum over their classical counterparts, particularly in reconstruction quality. In recent work, numerous novel CNN-based methods have been introduced to perform the JDD. For example, [12] trained and a deep network with millions of images to achieve state-of-the-art results. Similarly, [21] fuse the majorization-minimization techniques into a residual denoising network, [9] proposed a generative adversarial network (GAN) along with perceptual optimization to perform JDD. Also, [25] proposed a deep-learning-based method supervised by density-map and green channel guidance. Apart from these supervised approaches, [10] attempts to solve JDD with unsupervised learning on burst images.

**Image enhancement.** Image enhancement works mostly aim to improve the perceptual image quality by incorporating colour correction, sharpness boosting, denoising, white balancing, etc. Among the recent works, [11, 44] proposed learning-based solutions for automatic global luminance and gamma adjustment. Similarly, [23] offered deep-learning solutions for colour and tone correction, and [44] presented a CNN model to image contrast enhancement. However, the most comprehensive image enhancement approach was introduced by [15], where the author enhanced downgraded smartphone images according to superior-quality photos obtained with a high-end camera system.

**Learning ISP.** A typical camera ISP pipeline exploits numerous image processing blocks to reconstruct an sRGB image from the sensor’s raw data. A few novel methods have recently attempted to replace such complex ISPs by

learning from the convex set of data samples. In [35], the authors proposed a CNN model to suppress image noises and exposure correction of images captured with a smartphone camera. Likewise, [16] proposed a deep model incorporating extensive global feature manipulation to replace the entire ISP of the Huawei P20 smartphone. In another recent work, [24] proposed a two-stage deep network to replicate camera ISP.

**Quad Bayer Reconstruction.** Reconstructing RGB images from a Quad Bayer CFA is considerably challenging. In [18] has addressed this challenging task by proposing a duplex pyramid network. It worth noting, none of the existing methods (including [18]) specialized for our target applications. However, their respective domains' success inspired this work to develop an image processing pipeline for a pixel-bin image sensor, which can perform JDD and go beyond.

### 3. Method

This section details the network design, a multi-term objective function, and implementation strategies.

#### 3.1. Network design

Fig. 3 depicts the proposed method's overview, including the novel PIPNet architecture. Here, the proposed network exploits feature correlation, also known as attention mechanism [14, 40, 8], through the novel components in U-Net [33] like architecture to mitigate visual artefacts. Overall, the method aims to map a mosaic input ( $I_M$ ) as  $G : I_M \rightarrow I_R$ . Where the mapping function ( $F$ ) learns to reconstruct an RGB image ( $I_R$ ) as  $I_R \in [0, 1]^{H \times W \times 3}$ .  $H$  and  $W$  represent the height and width of the input and output images.

**Group depth attention bottleneck block.** The novel group depth attention bottleneck (GDAB) block allowed the proposed network to go deeper by leveraging depth attention [14]. The GDAB block comprises of  $m \in \mathbb{Z}$  number of depth attention bottleneck (DAB) blocks. Where the DABs are stacked consecutively and connected with short distance residual connection; thus, the network can converge with informative features [8]. For any  $g$ -th member of a GDAB block can be represented as:

$$F_g = W_g F_{g-1} + H_g(F_{g-1}) \quad (1)$$

Here,  $W_g$ ,  $F_{g-1}$ , and  $F_g$  represent the corresponding weight matrices, input, and output features.  $H_g(\cdot)$  denotes the function of group members (i.e., DAB).

**Depth attention bottleneck block.** The proposed DAB incorporates a depth attention block along with a bottleneck block. For a given input  $X$ , the  $m$ -th DAB block aims to output the feature map  $X'$  as:

$$X'_m = B_m(X) + D_m(X) \quad (2)$$

In Eq. 2,  $B(\cdot)$  presents the bottleneck block function, which has been inspired by the well-known MobileNetV2 [34]. The main motive of utilizing the bottleneck block is to control the trainable parameters with satisfactory performance. Typically, pixel-bin image sensors are exclusively designed for mobile devices. Therefore, we stress to reduce the trainable parameters as much as possible. Apart from the bottleneck block, DAB also incorporates a depth attention block, which has denoted as  $D(\cdot)$  in Eq. 2. It is worth noting, this study proposes to adding the feature map of the depth attention block along with the bottleneck block to leverage the long-distance depth-wise attention [14, 8]. Here, depth-wise squeezed descriptor  $Z \in \mathbb{R}^C$  has been obtained by shrinking  $\hat{X} = [x_1, \dots, x_c]$  as follows:

$$Z_c = A_{GP}(x_c) = \frac{1}{C} \sum_i^C x_c(i) \quad (3)$$

Here,  $A_{GP}(c)$  presents the global average pooling, spatial dimension, and feature map.

Additionally, an aggregated global dependencies have pursued by applying a gating mechanism as follows:

$$W = \tau(W_S(\delta(W_R(Z)))) \quad (4)$$

Here,  $\tau$  and  $\delta$  represent the sigmoid and ReLU activation functions, which have applied after  $W_S(\cdot)$  and  $W_R(\cdot)$  convolutional operations, which intended to set depth dimension of features to  $C/r$  and  $C$ .

The final output of the depth attention block has obtained by applying a depth-wise attention map with a rescaling factor [8] described as follows:

$$D_c = W_c \cdot S_c \quad (5)$$

Here,  $W_c$  and  $S_c$  represent the feature map and scaling factor.

**Spatial attention block.** The spatial attention block of the proposed method has been inspired by recent convolutional spatial modules [40, 6]. It aims to realize the spatial feature attention from a given feature map  $X$  as follows:

$$F = \tau(F_S([Z_A(X); Z_M(X)]) \quad (6)$$

Here,  $F(\cdot)$  and  $\tau$  represent the convolution operation and sigmoid activation. Additionally,  $Z_A$  and  $Z_M$  present the average pooling and max pooling, which generates two 2D feature map as  $X_A \in \mathbb{R}^{1 \times H \times W}$  and  $X_M \in \mathbb{R}^{1 \times H \times W}$ .

**Transition Layer.** The proposed network traverses different features depth to exploit the UNet like structure using upscaling or downscaling operations. The downsampling operation has obtained on an input feature map  $X_0$  as follows:

$$F_\downarrow = H_\downarrow(X_0) \quad (7)$$

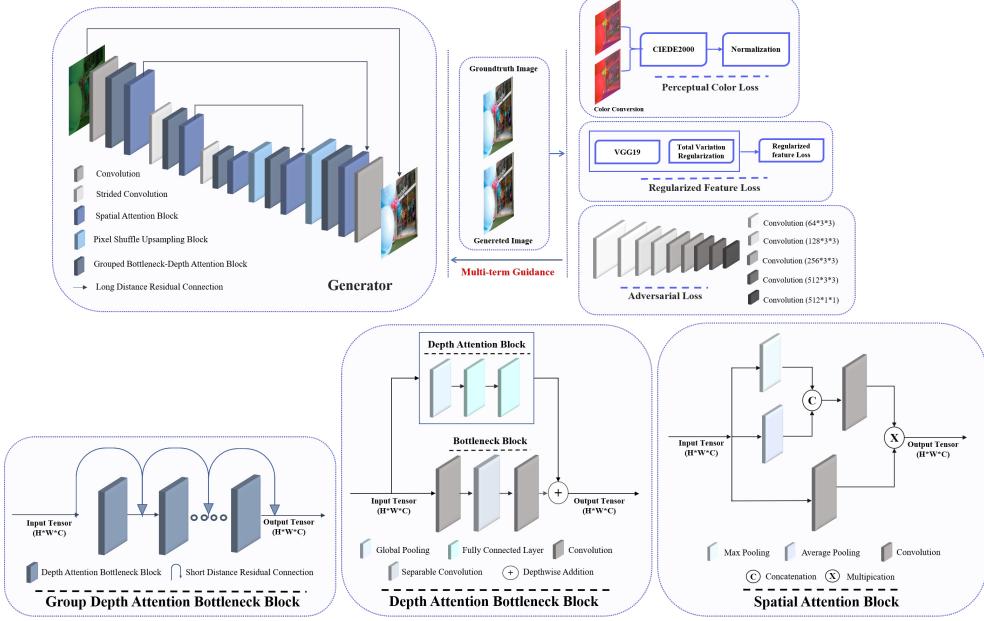


Figure 3: Overview of the proposed method, including network architecture and submodules.

Here,  $H_{\downarrow}(\cdot)$  represents a stride convolutional operation.

Inversely, the upscaling on an input feature map  $\mathbf{X}_0$  has achieved as:

$$\mathbf{F}_{\uparrow} = H_{\uparrow}(\mathbf{X}_0) \quad (8)$$

Here,  $H_{\uparrow}(\cdot)$  represents the pixel shuffle convolution operation followed by the PReLU function, which intends to avoid checkerboard artefacts [3].

**Conditional Discriminator.** The proposed PIPNet has appropriated the concept of adversarial guidance and adopted a well-established conditional Generative Adversarial Network (cGAN) [31]. The objective of the cGAN discriminator consists of stacked convolutional operations and set to maximize as:  $\mathbb{E}_{\mathbf{X}, \mathbf{Y}} [\log D(\mathbf{X}, \mathbf{Y})]$ .

### 3.2. Objective function

The proposed network  $G$  parameterized with weights  $\mathbf{W}$ , aims to minimize the training loss by appropriating the given  $P$  pairs of training images  $\{\mathbf{I}_M^t, \mathbf{I}_G^t\}_{t=1}^P$  as follows:

$$W^* = \arg \min_W \frac{1}{P} \sum_{t=1}^P \mathcal{L}_T(G(\mathbf{I}_M^t), \mathbf{I}_G^t) \quad (9)$$

Here,  $\mathcal{L}_T$  denotes the proposed multi-term objective function, which aims to improve the perceptual quality (i.e., details, texture, colour, etc.) while reconstructing an image.

**Reconstruction loss.** L1-norm is known to be useful for generating sharper images [45, 35]. Therefore, an L1-norm has adopted to calculate pixel-wise reconstruction error as follows:

$$\mathcal{L}_R = \| \mathbf{I}_G - \mathbf{I}_R \|_1 \quad (10)$$

Here,  $\mathbf{I}_G$  and  $\mathbf{I}_R$  present the ground truth image and output of  $G(\mathbf{I}_M)$  respectively.

**Regularized feature loss (RFL).**: VGG-19 feature-based loss functions aim to improve a reconstructed image's perceptual quality by encouraging it to have identical feature representation like the reference images [15, 30, 39]. Typically, such activation-map loss functions represented as follows:

$$\mathcal{L}_{FL} = \lambda_P \times \mathcal{L}_{VGG} \quad (11)$$

Where  $\mathcal{L}_{VGG}$  can be extended as follows:

$$\mathcal{L}_{VGG} = \frac{1}{H_j W_j C_j} \| \psi_t(\mathbf{I}_G) - \psi_t(\mathbf{I}_R) \|_1 \quad (12)$$

Here,  $\psi$  and  $j$  denote the pre-trained VGG network and its  $j^{th}$  layer.

It is worth noting, in Eq. 11,  $\lambda_P$  denotes the regulator of a feature loss. However, in most cases, the regulator's value has to set emphatically, and without proper tuning, it can deteriorate the reconstruction process [39]. To address this limitation, we replaced  $\lambda_P$  with a total variation regularization [36], which can be presented as follows:

$$\lambda_R = \frac{1}{H_j W_j C_j} \| \Delta O_v \| + \| \Delta O_h \| \quad (13)$$

Here,  $\| \Delta O_v \|$  and  $\| \Delta O_h \|$  present the gradients' summation in the vertical and horizontal directions calculated

over a training pair. The regularized form of Eq. 11 can be written as:

$$\mathcal{L}_{RFL} = \lambda_R \times \mathcal{L}_{VGG} \quad (14)$$

**Perceptual colour loss (PCL).** Due to the smaller aperture and sensor size, most smartphone cameras are prone to illustrate colour inconsistency in numerous instances [15]. We developed a CIEDE2000 [27] based on the perceptual colour loss to address this limitation, which intends to measure the colour difference between two images in euclidean space. Subsequently, the newly developed loss function encourages the proposed network to generate a similar colour as the reference image. The proposed perceptual colour loss can be represented as follows:

$$\mathcal{L}_{PCL} = \Delta E(\mathbf{I}_G, \mathbf{I}_R) \quad (15)$$

Here,  $\Delta E$  represents the CIEDE2000 colour difference [27].

**Adversarial loss.** Adversarial guidance is known to be capable of recovering texture and natural colours while reconstructing images. Therefore, we encouraged our model to employ a cGAN based cross-entropy loss as follows:

$$\mathcal{L}_G = - \sum_t \log D(\mathbf{I}_G, \mathbf{I}_R) \quad (16)$$

Here,  $D$  denotes the conditional discriminator, which aims to perform as a global critic.

**Total loss.** The final multi-term objective function ( $\mathcal{L}_T$ ) has calculated as follows:

$$\mathcal{L}_T = \mathcal{L}_R + \mathcal{L}_{RFL} + \mathcal{L}_{PCL} + \lambda_G \cdot \mathcal{L}_G \quad (17)$$

Here,  $\lambda_G$  presents adversarial regulators and set as  $\lambda_G = 1e-4$ .

### 3.3. Implementation details

The generator of proposed PIPNet traverses between different feature depth to leverage the UNet like structure as  $d = (64, 126, 256)$ , where the GDAB blocks of the proposed network comprise  $m = 3$  number of DAB block (also refer as group density in a later section). Every convolution operation in the bottleneck block of a DAB block incorporates  $1 \times 1$  convolutional and a  $3 \times 3$  separable convolution, where each layer is activated with a LeakyReLU function. Additionally, the spatial attention block, downsampling block, and discriminator utilize  $3 \times 3$  convolutional operation. A swish function has activated the convolution operations of the discriminator. Also, every  $(2n-1)^{th}$  layer of the discriminator increases the feature depth and reduces the spatial dimension by 2.

## 4. Experiments

The performance of the proposed method has been studied extensively with sophisticated experiments. This section details the experiment results and comparison for JDD.

### 4.1. Setup

To learn JDD for a pixel-bin image sensor, we extracted 741,968 non-overlapping image patches of dimension 128 × 128 from DIV2K [2] and Flickr2K [38] datasets. The image patches are sampled according to the CFA patterns and contaminated with a random noise factor of  $\mathcal{N}(\mathbf{I}_G|\sigma)$ . Here,  $\sigma$  represents the standard deviation of a Gaussian distribution, which is generated by  $\mathcal{N}(\cdot)$  over a clean image  $\mathbf{I}_G$ . It has presumed that the JDD has performed in sRGB colour space before colour correction, tone mapping, and white balancing. The model has implemented in the PyTorch [32] framework and optimized with an Adam optimizer [20] as  $\beta_1 = 0.9$ ,  $\beta_2 = 0.99$ , and learning rate =  $1e-4$ . The model trained for  $10 \sim 15$  epoch depending on the CFA pattern with a constant batch size of 12. The training process accelerated using an Nvidia Geforce GTX 1060 (6GB) graphical processing unit (GPU).

### 4.2. Joint demosaicing and denoising

We conducted an extensive comparison with the benchmark dataset for the evaluation purpose, including BSD100 [29], McM [41], Urban100 [7], Kodak [42], WED [28], and MSR demosaicing dataset [17]. We used only lin-RGB images from the MSR demosaicing dataset to verify the proposed method's feasibility in different colour spaces (i.e., sRGB and linRGB). Therefore, we denoted the MSR demosaicing dataset as linRGB in the rest of the paper. Apart from that, four CNN-based JDD methods (Deepjoint [12], Kokkinos [21], Dong [9], DeepISP [35]) and a specialized Quad Bayer reconstruction method (DPN [18]) have studied for comparison. Each compared method's performance cross-validated with three different noise levels  $\sigma = (5, 15, 25)$  and summarized with the following evaluation metrics: PSNR, SSIM, and DeltaE2000.

#### 4.2.1 Quad Bayer CFA

Performing JDD on Quad Bayer CFA is substantially challenging. However, the proposed method aims to tackle this challenging task by using the novel PIPNet. Table. 1 illustrates the performance comparison between the proposed PIPNet and target learning-based methods for Quad Bayer CFA. It is visible that our proposed method outperforms the existing learning-based methods in quantitative evaluation on benchmark datasets. Also, the visual results depicted in Fig. 4 confirm that the proposed method can reconstruct visually plausible images from Quad Bayer CFA.

Method	$\sigma$	<b>BSD100</b>	<b>WED</b>	<b>Urban100</b>	<b>McM</b>	<b>Kodak</b>	<b>linRGB</b>
		PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE
Deepjoint [12]	5	34.69/0.9474/2.45	30.99/0.9115/3.30	31.04/0.9272/3.43	31.16/0.8889/3.21	32.39/0.9310/3.01	40.03/0.9694/1.59
Kokkinos [21]		34.54/0.9662/2.25	32.18/0.9428/2.77	32.16/0.9501/3.13	32.84/0.9280/2.49	33.05/0.9532/2.74	38.46/0.9670/1.45
Dong [9]		33.79/0.9604/2.39	31.22/0.9312/3.03	31.48/0.9395/3.24	31.80/0.9117/2.81	32.72/0.9494/2.85	38.40/0.9477/1.75
DeepISP [35]		36.78/0.9711/1.94	32.25/0.9392/2.93	32.52/0.9494/3.10	32.32/0.9143/2.89	34.25/0.9547/2.45	42.02/0.9819/1.27
DPN [18]		37.71/0.9714/1.81	33.60/0.9490/2.50	33.73/0.9560/2.78	34.37/0.9354/2.26	35.73/0.9597/2.22	42.08/0.9767/1.16
<b>PIPNet</b>		<b>39.43/0.9803/1.42</b>	<b>35.08/0.9609/2.08</b>	<b>35.66/0.9670/2.25</b>	<b>35.55/0.9470/1.90</b>	<b>37.21/0.9699/1.79</b>	<b>44.14/0.9827/0.94</b>
Deepjoint [12]	15	32.68/0.8883/2.81	29.97/0.8678/3.61	30.02/0.8819/3.72	30.19/0.8432/3.51	31.02/0.8655/3.35	36.94/0.9100/1.88
Kokkinos [21]		33.57/0.9396/2.56	31.38/0.9183/3.07	31.40/0.9262/3.38	32.03/0.9026/2.79	32.17/0.9208/3.04	37.72/0.9453/1.75
Dong [9]		32.49/0.9280/2.67	30.27/0.9004/3.32	30.52/0.9110/3.50	30.80/0.8755/3.08	31.58/0.9128/3.13	35.66/0.9061/2.05
DeepISP [35]		34.73/0.9443/2.30	31.31/0.9174/3.20	31.44/0.9276/3.37	31.47/0.8909/3.13	32.74/0.9226/2.81	39.44/0.9608/1.59
DPN [18]		35.08/0.9451/2.17	32.27/0.9265/2.80	32.22/0.9340/3.07	33.16/0.9132/2.53	33.60/0.9274/2.57	39.43/0.9613/1.44
<b>PIPNet</b>		<b>36.68/0.9586/1.79</b>	<b>33.55/0.9416/2.41</b>	<b>33.85/0.9481/2.56</b>	<b>34.21/0.9277/2.18</b>	<b>34.89/0.9422/2.14</b>	<b>41.86/0.9728/1.18</b>
Deepjoint [12]	25	30.15/0.8006/3.51	28.42/0.8008/4.16	28.44/0.8140/4.28	28.69/0.7745/4.03	29.00/0.7724/4.02	33.73/0.8246/2.51
Kokkinos [21]		31.99/0.9070/2.95	30.12/0.8868/3.45	30.09/0.8961/3.74	30.74/0.8697/3.17	30.74/0.8827/3.43	36.03/0.9152/2.08
Dong [9]		31.49/0.8870/3.00	29.50/0.8607/3.60	29.72/0.8755/3.80	29.67/0.8205/3.38	30.76/0.8685/3.43	31.74/0.8419/2.65
DeepISP [35]		32.85/0.9111/2.79	30.17/0.8885/3.60	30.17/0.8994/3.81	30.42/0.8650/3.51	31.19/0.8839/3.29	37.23/0.9343/2.09
DPN [18]		33.33/0.9059/2.57	31.07/0.8930/3.18	31.09/0.9019/3.40	31.91/0.8811/2.88	31.98/0.8840/2.96	37.49/0.9280/1.79
<b>PIPNet</b>		<b>34.62/0.9353/2.14</b>	<b>32.13/0.9198/2.75</b>	<b>32.28/0.9285/2.89</b>	<b>32.88/0.9072/2.49</b>	<b>33.04/0.9140/2.50</b>	<b>39.44/0.9565/1.47</b>

Table 1: Quantitative evaluation of JDD on Quad Bayer CFA. A higher value of PSNR and SSIM indicates better results, while lower DeltaE indicates more colour consistency.



Figure 4: Qualitative evaluation of JDD on Quad Bayer CFA.

#### 4.2.2 Bayer CFA

As mentioned earlier, the pixel-bin image sensors have to employ Bayer CFA in numerous instances. Therefore, the proposed PIPNet has to perform JDD evenly on Bayer CFA. Table. 2 illustrates the JDD performance of the proposed method and its counterparts on Bayer CFA. The proposed method depicts the consistency in Bayer CFA as well. Also, it can recover more details while performing JDD on Bayer CFA without producing any visually disturbing artefacts, as shown in Fig. 5.

#### 4.3. Network analysis

The practicability of the proposed network and its novel component has been verified by analyzing the network performance.

##### 4.3.1 Ablation study

An ablation study was conducted by removing all novel components like attention mechanism (AM), PCL, and RFL from the proposed method and later injecting them into the

network consecutively. Fig. 6 depicts the importance of each proposed component through visual results. Apart from that Table. 3 confirms the practicability of novel components introduced by the proposed method. For simplicity, we combined all sRGB datasets and calculated the mean over the unified dataset while performing JDD on challenging Quad Bayer CFA.

##### 4.3.2 Group density vs performance

Despite being significantly deeper and wider, the proposed PIPNet comprises 3.3 million parameters. The bottleneck block employed in GDAB allows our network to control the trainable parameter. Nevertheless, the number of parameters can be controlled by altering the group density (GD) of the GDAB blocks, as shown in Fig. 7. Additionally, Table. 4 illustrates the relation between GD and performance in both colour spaces while performing JDD.

#### 5. Image reconstruction and enhancement

Typically, smartphone cameras are susceptible to produce flat, inaccurate colour profiles and noisy images com-

Model	$\sigma$	BSD100		WED		Urban100		McM		Kodak		linRGB	
		PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE
Deepjoint [12]	5	35.73/0.9661/2.21	32.96/0.9439/2.67	32.44/0.9468/2.99	33.56/0.9283/2.48	33.74/0.9519/2.62	38.99/0.9690/1.66	35.37/0.9462/2.20	35.24/0.9596/2.51	39.77/0.9716/1.56	33.88/0.9244/2.44	35.72/0.9646/2.56	38.07/0.9381/1.81
Kokkinos [21]		36.81/0.9712/2.14	34.78/0.9561/2.43	34.30/0.9552/2.86	35.37/0.9462/2.20	35.24/0.9596/2.51	39.77/0.9716/1.56	33.94/0.9486/2.85	35.29/0.9606/2.09	43.72/0.9826/1.11	36.27/0.9707/2.08	37.21/0.9590/1.67	43.97/0.9826/1.06
Dong [9]		36.66/0.9710/2.29	33.72/0.9461/2.60	33.94/0.9486/2.85	33.88/0.9244/2.44	35.72/0.9646/2.56	38.07/0.9381/1.81	34.05/0.9497/2.38	35.29/0.9606/2.09	43.72/0.9826/1.11	36.27/0.9707/2.08	37.21/0.9590/1.67	43.97/0.9826/1.06
DeepISP [35]		38.60/0.9771/1.60	34.05/0.9497/2.38	33.53/0.9550/2.67	34.08/0.9327/2.31	35.29/0.9606/2.09	43.72/0.9826/1.11	36.71/0.9697/1.79	37.21/0.9590/1.67	43.97/0.9826/1.06	36.27/0.9707/2.08	37.21/0.9590/1.67	43.97/0.9826/1.06
DPN [18]		40.38/0.9837/1.25	36.71/0.9697/1.79	36.27/0.9707/2.08	37.21/0.9590/1.67	38.11/0.9739/1.61	43.97/0.9826/1.06	37.51/0.9731/1.87	38.13/0.9612/1.52	43.77/0.9768/1.41	46.41/0.9870/0.82	39.37/0.9768/1.41	46.41/0.9870/0.82
PIPNet		<b>41.75/0.9851/1.11</b>	<b>37.84/0.9717/1.61</b>	<b>37.51/0.9731/1.87</b>	<b>38.13/0.9612/1.52</b>	<b>39.37/0.9768/1.41</b>	<b>46.41/0.9870/0.82</b>						
Deepjoint [12]	15	33.50/0.9084/2.83	31.62/0.8988/3.15	31.20/0.9016/3.51	32.20/0.8829/2.92	31.99/0.8872/3.28	36.56/0.9201/2.15	32.93/0.9259/2.89	32.54/0.9264/3.30	33.52/0.9133/2.65	33.28/0.9202/2.98	37.82/0.9434/1.96	32.30/0.9277/2.62
Kokkinos [21]		34.52/0.9377/2.62	32.93/0.9259/2.89	32.54/0.9264/3.30	33.52/0.9133/2.65	33.28/0.9202/2.98	37.82/0.9434/1.96	33.86/0.9379/2.67	32.20/0.9182/3.18	31.87/0.8792/2.80	33.27/0.9270/2.96	34.10/0.9009/2.28	32.90/0.9410/2.23
Dong [9]		35.29/0.9410/2.23	32.46/0.9201/2.83	32.04/0.9260/3.12	32.57/0.9010/2.74	33.17/0.9192/2.68	39.81/0.9563/1.63	34.60/0.9462/2.22	34.33/0.9485/2.46	35.30/0.9355/2.04	35.28/0.9414/2.07	41.51/0.9726/1.36	36.90/0.9577/1.75
DeepISP [35]		35.29/0.9410/2.23	32.46/0.9201/2.83	32.04/0.9260/3.12	32.57/0.9010/2.74	33.17/0.9192/2.68	39.81/0.9563/1.63	34.60/0.9462/2.22	34.33/0.9485/2.46	35.30/0.9355/2.04	35.28/0.9414/2.07	41.51/0.9726/1.36	36.90/0.9577/1.75
DPN [18]		36.90/0.9577/1.75	34.62/0.9462/2.22	34.33/0.9485/2.46	35.30/0.9355/2.04	35.28/0.9414/2.07	41.51/0.9726/1.36	37.59/0.9620/1.60	<b>35.19/0.9494/2.07</b>	<b>34.93/0.9522/2.30</b>	<b>35.81/0.9391/1.92</b>	<b>35.96/0.9478/1.88</b>	<b>43.02/0.9735/1.13</b>
PIPNet		<b>35.21/0.9379/2.01</b>	<b>33.24/0.9251/2.50</b>	<b>33.02/0.9300/2.71</b>	<b>33.95/0.9147/2.30</b>	<b>33.74/0.9186/2.31</b>	<b>40.46/0.9547/1.46</b>						

Table 2: Quantitative evaluation of JDD on Bayer CFA. A higher value of PSNR and SSIM indicates better results, while lower DeltaE indicates more colour consistency.

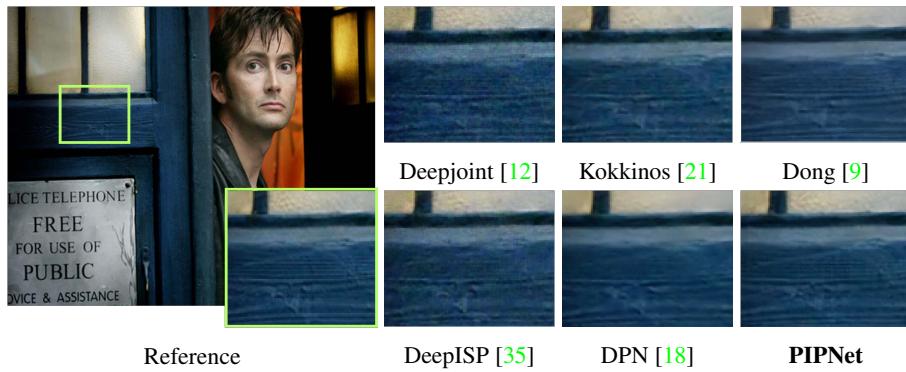


Figure 5: Qualitative evaluation of JDD on Bayer CFA.

Model	sRGB		linRGB	
	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE
Base	24.51/0.7436/8.15	25.68/0.6258/8.71		
Base + AM	32.82/0.9208/2.75	38.95/0.9430/1.77		
Base + AM + PCL	33.96/0.9344/2.40	41.01/0.9647/1.31		
Base + AM + PCL + RFL	<b>34.64/0.9436/2.22</b>	<b>41.82/0.9727/1.18</b>		

Table 3: Ablation study on sRGB and linRGB images. Each component proposed throughout this study has an evident impact on network performance.

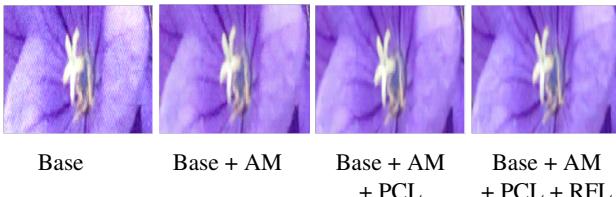


Figure 6: Each proposed component plays a crucial role in JDD (best viewed in zoom).

paring to professional cameras [15, 16]. To address this lim-

GD	Parameters	sRGB		linRGB	
		PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE	PSNR/SSIM/DeltaE
1	2,778,863	33.03/0.9239/2.73	38.30/0.9364/1.89		
2	3,118,703	33.49/0.9308/2.61	39.79/0.9608/1.64		
<b>3</b>	<b>3,458,543</b>	<b>34.64/0.9436/2.22</b>	<b>41.82/0.9727/1.18</b>		

Table 4: Group density vs model performance. The number of DAB blocks can impact network performance by making a trade-off between parameters and accuracy.



Figure 7: Impact of GD while performing JDD (best viewed in zoom).

itation and study the feasibility of a learning-based method on an actual pixel-bin image sensor (i.e., Sony IMX586), we stretch our PIPNet as a two-stage network. Stage-I of

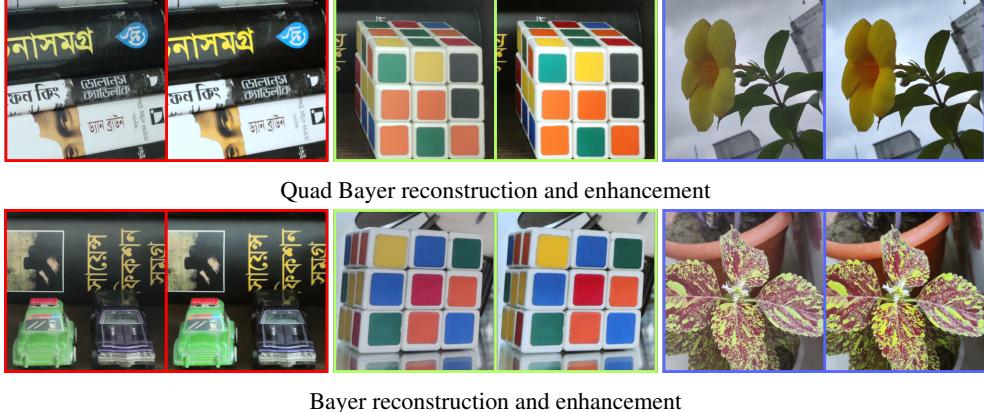


Figure 8: Qualitative comparison between pixel-bin image sensor output (i.e., Oneplus Nord) and the proposed PIPNet+. In every image pairs, Left: Oneplus Nord and Right: Results obtained by PIPNet+.

the extended network performs JDD, as described in section. 4, and stage-II aims to enhance reconstructed images’ perceptual quality by correcting colour profile, white balancing, brightness correction, etc. The extended version of PIPNet is denoted as PIPNet+. It worth noting, the PIPNet+ comprises the same configuration (i.e., hyperparameters, GD, etc.) as its one stage variant; however, it has trained with smartphone-DSLR image pairs from the DPED dataset [15], as suggested in a recent study [24]. Our comprehensive solution’s feasibility has compared with a recent smartphone (i.e., Oneplus Nord), which utilizes the pixel binning technique with actual hardware. We also develop an android application to control the binning process while capturing images for our network evaluation. Additionally, the captured images were resampled according to the CFA pattern prior to the model inference.

## 5.1. Visual results

Fig. 8 illustrates a visual comparison between Oneplus Nord and our PIPNet+. The proposed method can improve the perceptual quality of degraded images captured with an actual pixel-bin image sensor while performing ISP tasks like demosaicking, denoising, colour correction, brightness correction, etc.

## 5.2. User study

Apart from the visual comparison, we perform a blind-fold user study comparing Oneplus Nord and our proposed method. Also, we develop a blind-fold online testing method, which allows the users to pick an image from pairs of Oneplus Nord and our reconstructed image. The testing evaluation process is hosted online publicly by an anonymous user. Thus, the unbiased user opinion can be cast to calculate the mean opinion score (MOS) for both CFA patterns. Table. 5 illustrates the MOS of our proposed

method and Oneplus Nord. The proposed method outperforms Oneplus Nord in blind-fold testing by a substantial margin. Also, it confirms that Quad Bayer reconstruction is far more challenging than a typical Bayer reconstruction. Therefore, the traditional ISP illustrates deficiencies by producing visually pleasing images on such CFA patterns, while the proposed method can deliver more acceptable results.

CFA	Method	MOS ↑
Quad Bayer	Oneplus Nord	1.30
	<b>PIPNet+</b>	<b>3.70</b>
Bayer	Oneplus Nord	1.50
	<b>PIPNet+</b>	<b>3.50</b>

Table 5: A user study on Oneplus Nord and PIPNet+. Higher MOS indicates better user preference.

## 6. Conclusion

This study tackled the challenging task of performing JDD by incorporating a learning-based method specialized for a pixel-bin image sensor. We introduced a novel deep network that employed attention mechanisms and guided by a multi-term objective function, including two novel perceptual losses. Also, we stretched our proposed method to enhance the perceptual image quality of a pixel-bin image sensor along with reconstruction. The experiment with different CFA patterns illustrates that the proposed method can outperform the existing approaches in qualitative and quantitative comparison. Despite revealing new possibilities, we conducted our experiments mostly with simulated data collected by traditional Bayer sensors. Hence, the performance of the proposed network can differ in some complicated cases. It has planned to counter the data limitation by collecting a real-world dataset using pixel-bin image sensors for further study.

## References

- [1] Gennadiy A Agranov, Claus Molgaard, Ashirwad Bahukhandi, Chiajen Lee, and Xiangli Li. Pixel binning in an image sensor, June 20 2017. US Patent 9,686,485. 1
- [2] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 126–135, 2017. 5
- [3] Andrew Aitken, Christian Ledig, Lucas Theis, Jose Caballero, Zehan Wang, and Wenzhe Shi. Checkerboard artifact free sub-pixel convolution: A note on sub-pixel convolution, resize convolution and convolution resize. *arXiv preprint arXiv:1707.02937*, 2017. 4
- [4] Sandor L Barna, Scott P Campbell, and Gennady Agranov. Method and apparatus for improving low-light performance for small pixel image sensors, June 11 2013. US Patent 8,462,220. 1
- [5] Bryce E Bayer. Color imaging array, July 20 1976. US Patent 3,971,065. 1, 2
- [6] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5659–5667, 2017. 3
- [7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3213–3223, 2016. 5
- [8] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11065–11074, 2019. 3
- [9] Weishong Dong, Ming Yuan, Xin Li, and Guangming Shi. Joint demosaicing and denoising with perceptual optimization on a generative adversarial network. *arXiv preprint arXiv:1802.04723*, 2018. 2, 5, 6, 7
- [10] Thibaud Ehret, Axel Davy, Pablo Arias, and Gabriele Facciolo. Joint demosaicking and denoising by fine-tuning of bursts of raw images. In *Int. Conf. Comput. Vis.*, pages 8868–8877, 2019. 2
- [11] Xueyang Fu, Delu Zeng, Yue Huang, Yinghao Liao, Xinghao Ding, and John Paisley. A fusion-based enhancing method for weakly illuminated images. *Signal Process.*, 129:82–96, 2016. 2
- [12] Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédéric Durand. Deep joint demosaicking and denoising. *ACM Trans. Graph.*, 35(6):1–12, 2016. 2, 5, 6, 7
- [13] Keigo Hirakawa and Thomas W Parks. Joint demosaicing and denoising. *IEEE Trans. Image Process.*, 15(8):2146–2157, 2006. 2
- [14] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7132–7141, 2018. 2, 3
- [15] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks. In *Int. Conf. Comput. Vis.*, pages 3277–3285, 2017. 1, 2, 4, 5, 7, 8
- [16] Andrey Ignatov, Luc Van Gool, and Radu Timofte. Replacing mobile camera isp with a single deep learning model. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 536–537, 2020. 2, 3, 7
- [17] Daniel Khashabi, Sebastian Nowozin, Jeremy Jancsary, and Andrew W Fitzgibbon. Joint demosaicing and denoising via learned nonparametric random fields. *IEEE Trans. Image Process.*, 23(12):4968–4981, 2014. 5
- [18] Irina Kim, Seongwook Song, Soonkeun Chang, Sukhwan Lim, and Kai Guo. Deep image demosaicing for submicron image sensors. *J. Imaging Sci. Techn.*, 63(6):60410–1, 2019. 1, 2, 3, 5, 6, 7
- [19] Yongnam Kim and Yunkyoung Kim. High-sensitivity pixels with a quad-wrgb color filter and spatial deep-trench isolation. *Sensors*, 19(21):4653, 2019. 1, 2
- [20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5
- [21] Filippos Kokkinos and Stamatios Lefkimiatis. Deep image demosaicking using a cascade of convolutional residual denoising networks. In *Eur. Conf. Comput. Vis.*, pages 303–319, 2018. 2, 5, 6, 7
- [22] Assaf Lahav and David Cohen. Color pattern and pixel level binning for aps image sensor using  $2 \times 2$  photodiode sharing scheme, Aug. 10 2010. US Patent 7,773,138. 1
- [23] Joon-Young Lee, Kalyan Sunkavalli, Zhe Lin, Xiaohui Shen, and In So Kweon. Automatic content-aware color and tone stylization. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2470–2478, 2016. 2
- [24] Zhetong Liang, Jianrui Cai, Zisheng Cao, and Lei Zhang. Cameranet: A two-stage framework for effective camera isp learning. *arXiv preprint arXiv:1908.01481*, 2019. 3, 8
- [25] Lin Liu, Xu Jia, Jianzhuang Liu, and Qi Tian. Joint demosaicing and denoising with self guidance. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2240–2249, 2020. 2
- [26] X Liu, Boyd Fowler, Hung Do, Steve Mims, Dan Laxson, and Brett Frymire. High performance cmos image sensor for low light imaging. In *International Image Sensor Workshop*, pages 327–330, 2007. 1
- [27] M Ronnier Luo, Guihua Cui, and Bryan Rigg. The development of the cie 2000 colour-difference formula: Ciede2000. *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, 26(5):340–350, 2001. 5
- [28] Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang. Waterloo exploration database: New challenges for image quality assessment models. *IEEE Trans. Image Process.*, 26(2):1004–1016, 2016. 5
- [29] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to

- evaluating segmentation algorithms and measuring ecological statistics. In *Int. Conf. Comput. Vis.*, volume 2, pages 416–423, July 2001. 5
- [30] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In *Eur. Conf. Comput. Vis.*, pages 768–783, 2018. 4
- [31] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 4
- [32] Pytorch. PyTorch Framework code. <https://pytorch.org/>, 2016. Accessed: 2020-11-14. 5
- [33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3
- [34] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4510–4520, 2018. 3
- [35] Eli Schwartz, Raja Giryes, and Alex M Bronstein. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE Trans. Image Process.*, 28(2):912–923, 2018. 2, 3, 4, 5, 6, 7
- [36] David Strong and Tony Chan. Edge-preserving and scale-dependent properties of total variation regularization. *Inverse Probl.*, 19(6):S165, 2003. 4
- [37] Hanlin Tan, Xiangrong Zeng, Shiming Lai, Yu Liu, and Maojun Zhang. Joint demosaicing and denoising of noisy bayer images with admm. In *IEEE Int. Conf. Image Process.*, pages 2951–2955. IEEE, 2017. 2
- [38] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 114–125, 2017. 5
- [39] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Eur. Conf. Comput. Vis.*, pages 0–0, 2018. 4
- [40] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Eur. Conf. Comput. Vis.*, pages 3–19, 2018. 2, 3
- [41] Wei Wu, Zheng Liu, Wail Gueaieb, and Xiaohai He. Single-image super-resolution based on markov random field and contourlet transform. *J. Electron. Imaging*, 20(2):023005, 2011. 5
- [42] Akira Yanagawa, Alexander C Loui, Jiebo Luo, Shih-Fu Chang, Dan Ellis, Wan Jiang, Lyndon Kennedy, and Keansub Lee. Kodak consumer video benchmark data set: concept definition and annotation. *Columbia University ADVENT Technical Report*, pages 246–2008, 2008. 5
- [43] Yoonjong Yoo, Jaehyun Im, and Joonki Paik. Low-light image enhancement using adaptive digital pixel binning. *Sensors*, 15(7):14917–14931, 2015. 1
- [44] Lu Yuan and Jian Sun. Automatic exposure correction of consumer photographs. In *Eur. Conf. Comput. Vis.*, pages 771–785. Springer, 2012. 2
- [45] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Trans. Comput. Imag.*, 3(1):47–57, 2016. 4