AWNet: Attentive Wavelet Network for Image ISP

AWNet: Attentive Wavelet Network for Image ISP

By Linhui Dai, B.Eng.

A Thesis Submitted to the School of Graduate Studies in the Partial Fulfillment of the Requirements for the Degree M.A.Sc. Thesis

McMaster University

M.A.Sc. Thesis (2021)

Hamilton, Ontario (Department of Electrical and Computer Engineering)

TITLE: AWNet: Attentive Wavelet Network for Image ISP

AUTHOR: Linhui Dai (McMaster University)

SUPERVISOR: Dr.Jun Chen

NUMBER OF PAGES: x, 51

To my dear family and friends.

Abstract

As the revolutionary improvement being made on the performance of smartphones over the last decade, mobile photography becomes one of the most common practices among the majority of smartphone users. However, due to the limited size of camera sensors on phone, the photographed image is still visually distinct from the one taken by the digital single-lens reflex (DSLR) camera. To narrow this performance gap, one way is to redesign the camera image signal processor (ISP) to improve the image quality. Owing to the rapid rise of deep learning, recent works resort to the deep convolutional neural network (CNN) to develop a sophisticated data-driven ISP that directly maps the phone-captured image to the DSLR-captured one. In this paper, we introduce a novel network that utilizes the attention mechanism and wavelet transform, dubbed AWNet, to tackle this learnable image ISP problem. By adding the wavelet transform, our proposed method enables us to restore favorable image details from RAW information and achieve a larger receptive field without compromising computational efficiency. The global context block is adopted in our method to learn the non-local color mapping for the generation of appealing RGB images. More importantly, this block alleviates the influence of image misalignment occurred on the provided dataset. Experimental results demonstrate the superiorities of our design in both qualitative and quantitative measurements.

Acknowledgements

I would like to acknowledge and give my warmest thanks to my beloved supervisor Dr. Jun Chen who made this work possible. His guidance and advice pave a way for me to work towards a qualified master's student and enable me to dive into my research topics. I also would like to give my appreciation to my committee members who give careful reviews and meaningful suggestions to my work.

I want to give special thanks to Chengqi Li and Xiaohong Liu for their hard-working dedication throughout the collaboration of this work. Their contributions give profound influence to this research.

Furthermore, I want to express my sincere appreciation to Xiang Song, Zhihao Shi, Huan Liu, Chenxiao Niu and all other people I worked with during my graduate study. I feel grateful to have this chance to collaborate with all these friendly, intelligent, and knowledgeable colleagues. They always give me smart views that inspire my researches.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Problem Statement

1.1 Introduction

Traditional image ISP is a critical processing unit that maps RAW images from the camera sensor to RGB images in order to accommodate the human visual system (HVS). For this purpose, a series of sub-processing units are leveraged in order to tackle different types of artifacts from photo-capturing devices, including, among others, colour shifts, signal noises, and moire effects. However, tuning each sub-processing unit requires legions of efforts from imagery experts.

Nowadays, mobile devices are equipped with high-resolution cameras to serve the ever-growing need for mobile photography. However, due to the compact space, the hardware is limited with respect to the quality of the optics and the pixel numbers. Moreover, the time of exposure is relatively short due to the instability of hand-holding. Therefore, a mobile-specific ISP has to compensate for these limitations as well. In fact, current consumer ISP systems cannot completely handle those aforementioned problems; therefore, they render the image in some lossy ways, for example, by adding noise or applying cartoonish blurring, in order to remove defects from ISP unit (See Fig. 1.1).

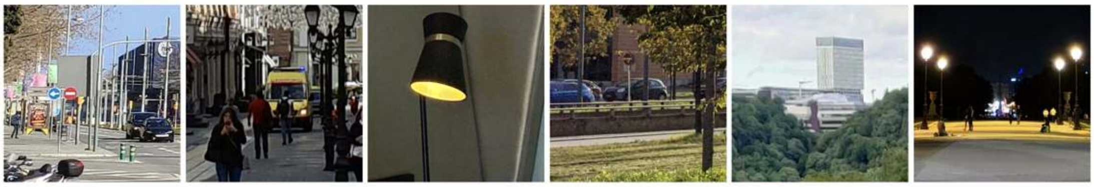


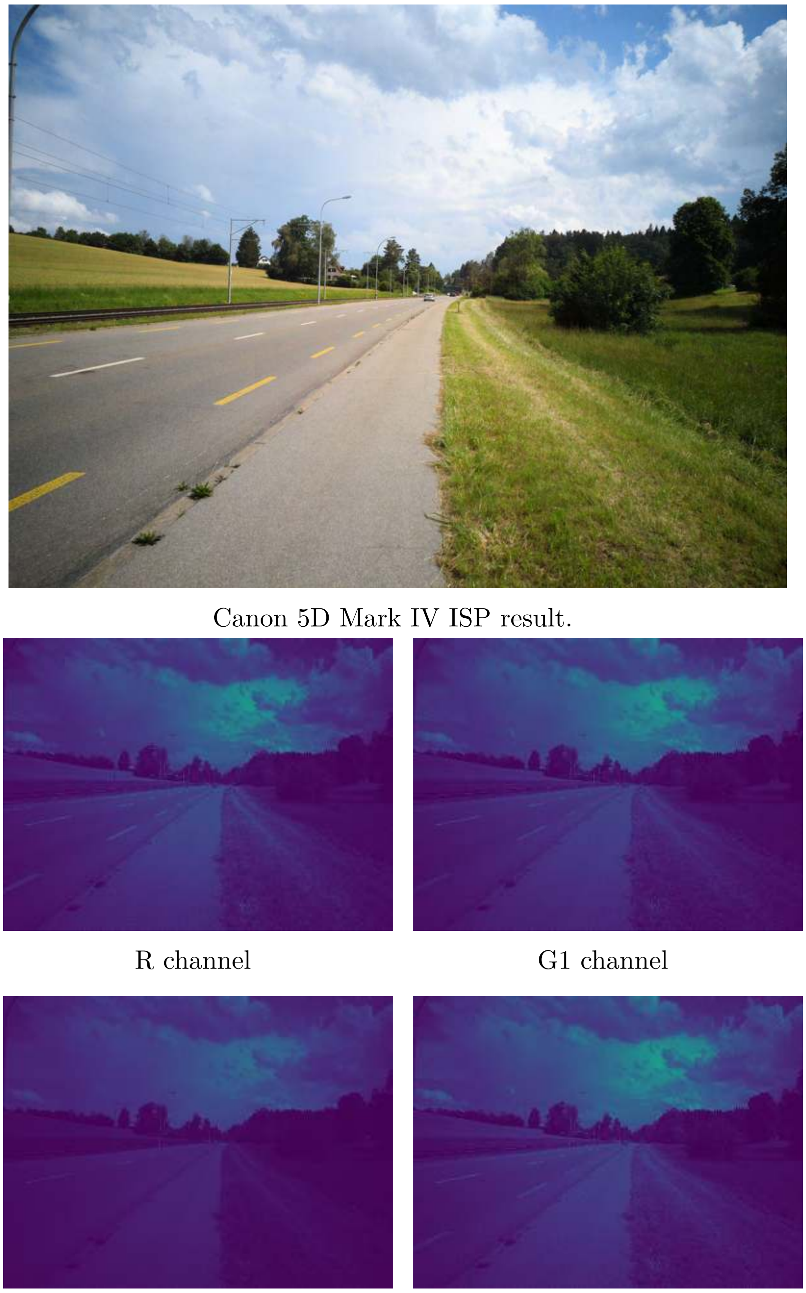
FIGURE 1.1: Artifacts on mobile photos. From left to right: cartoonish blurring (Xiaomi Mi 9, Samsung Galaxy Note10+), noise (iPhone 11 Pro, Google Pixel 4 XL), and image flattening (OnePlus 7 Pro, Huawei Mate 30 Pro). Note that the image is originally used in Ignatov et al. 2020

Recently, deep learning (DL) based methods have achieved considerable success on various image enhancement tasks, including image denoising (Abdelhamed et al. 2020; Zhang et al. 2017), image demosaicing (Gharbi et al. 2016), and super-resolution (Kim et al. 2016; Ledig et al. 2017; Lugmayr et al. 2020; Wang et al. 2019). Different from traditional image processing algorithms that commonly require prior knowledge of natural image statistics, data-driven methods can implicitly learn such information. Due to this fact, the DL-based method becomes a good fit for mapping problems (Chen et al. 2018a; Xu et al. 2019; Zhu et al. 2017). Here, learning image ISP can be regarded as an image-to-image translation problem, which can be well-addressed by the DL-based method. In ZRR dataset from (Ignatov et al. 2020), the RAW images can be decomposed into 4 channels, which are red (R), green (G1), blue (B) and green (G2) from the Bayer pattern, as shown in Fig. 1.2. Remark that 2 of 4 channels record the radiance information from green sensors. Therefore, additional operations such as demosaicing and colour correction are needed to tackle the RAW images as compared to RGB images. Moreover, due to the nature of the Bayer filter, the size of these 4 channels is down-sampled by a factor of two. In order to make the size of prediction and ground truth images consistent, an up-sampling operation is required. This can be regarded as a restoration problem, where the recovery of high-frequency information should be taken into consideration. In our observation, the misalignment between the DSLR and mobile photographed image pairs is severe even though the authors have adopted the SIFT (Lowe 2004) and RANSAC (Vedaldi and Fulkerson 2010) algorithms to mitigate this effect. It is worth mentioning that the minor misalignment between the input RAW image and ground-truth RGB image would cause a significant performance drop.

1.2 Contributions

To tackle the aforementioned problems, we introduce a novel trainable pipeline that utilizes the attention mechanism and wavelet transform. More specifically, the input of our proposed methods is a combination of a RAW image and its demosaiced counterpart as a complement, where the two-branch design is aimed at emphasizing the different training tasks, namely, noise removal and detail restoration on RAW model and the colour mapping on the demosaiced model; the discrete wavelet transform (DWT) is adopted to restore fine context details from RAW images while reserving the information in features during training; as for the colour correction and tone mapping, the res-dense connection and attention mechanism are utilized to encourage the network putting effort on the focused areas.

In summary, our main contributions are:



Canon 5D Mark IV ISP result.

R channel

G1 channel

B channel

G2 channel

FIGURE 1.2: Visualization of each channel in the RAW image and the corresponding RGB image reconstructed by AWNet. Zoom-in for better views.

1) Exploring the effectiveness of wavelet transform and non-local attention mechanism in image ISP pipeline.

2) A two-branch design to take a raw image and its demosaiced counterpart that endows our proposed method the ability to translate the RAW image to the RGB image.

3) A lightweight and fully convolutional encoder-decoder design that is time-efficient and flexible on different input sizes.

1.3 Thesis Structure

In chapter 2, we review both traditional and learnable image processing pipelines and compare those with our work to highlight our contributions. In the same chapter, we also discuss some applications in low-level image restoration fields that utilize the raw data. Then, in chapter 3, we elaborate our model design in detail, including the featured two-branch design, usage of the discrete wavelet transform, attentive residual module, and the design of loss functions. In chapter 4, we give the detailed implementation of our algorithm and the setup of our experiments. Meanwhile, we demonstrate the performance of our algorithm through comprehensive experiments. In the end, in chapter 5, we make a conclusion of our work and provide some suggestions of potential improvements for our work.

Chapter 2

Related Works

In this section, we provide a brief review of the traditional image ISP methods, some representative RAW to RGB mapping algorithms, the existing learnable imaging pipelines, some attention algorithms, and usages of wavelet transform in deep learning.

2.1 Traditional Image ISP Pipeline

Traditional ISP pipeline encompasses multiple image signal operations, including, among others, denoising, demosaicing, white balancing, colour correction, gamma correction, and tone mapping. Due to the nature of the image sensor, the existence of noise in RAW images is inevitable. Therefore, some operations are (Abdelhamed et al. 2020; Dabov et al. 2007; Zhang et al. 2017) proposed to remove the noise and improve the signal-to-noise ratio. The demosaicing operation interpolates the single-channel raw image with repeated mosaic patterns into multi-channel colour images (Gharbi et al. 2016). White balancing corrects the colour by shifting illuminations of RGB channels to make the image more perceptually acceptable

(Cheng et al. 2015). Color correction adjusts the image value by a correction matrix (Kwok et al. 2013; Rizzi et al. 2003). Tone mapping shrinks the histogram of image values to enhance image details (Rana et al. 2019; Yuan and Sun 2012). Note that all sub-processing units in the traditional image ISP pipeline require human effort to manually adjust the final result.

2.2 RAW Data Usage in Low-level Image Restoration

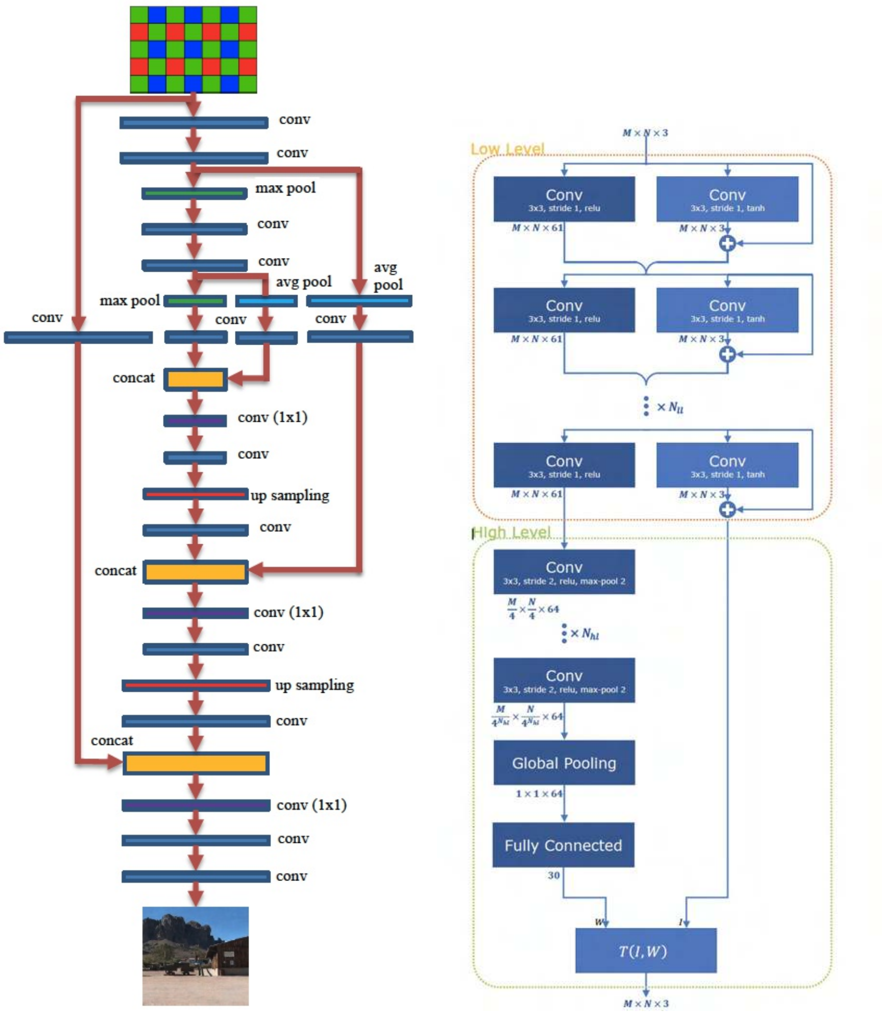
The advantages of applying RAW data on low-level vision tasks have been explored by different works in the field of image restoration. For instance, (Chen et al. 2018a) uses dark RAW image and bright colour image pairs to restore dark images from images with long exposure. In this case, the radiance information retained by raw data contributes to the restoration of image illumination. (Xu et al. 2019) takes advantage of rich radiance information from unprocessed camera data to restore high-frequency details and improve their network performance on super-resolution tasks. Their experiment reveals that using raw data as a substitute for camera processed data is beneficial on single image super-resolution tasks. Lately, (Ignatov et al. 2020; Schwartz et al. 2018) adopt unprocessed image data to enhance mobile camera imaging. Since RAW data avoids the information loss introduced by quantization in ISP, it is favourable for a neural network to restore the delicate image details. Inspired by (Ignatov et al. 2020), our work makes use of the RAW data to train our network for a learnable ISP pipeline. Instead of only taking RAW images as the input, we adopt the combination of the input data formats from (Ignatov et al. 2020) and (Schwartz et al. 2018) to encourage our network to learn different sub-tasks of image ISP, for example, noise removal, colour mapping, and detail restoration.

2.3 Deep Learning Based Image ISP Pipeline

Since CNN has achieved promising performance on plenty of low-level vision tasks (He et al. 2019; Kim et al. 2016; Ledig et al. 2017; Tao et al. 2018; Wang et al. 2019), it is natural to leverage it for the learning of camera ISP.

(Ratnasingam 2019) generates synthetic RAW images from JPEG ones and applies RAW-to-RGB mapping to restore the original RGB images. However, since this network handles a series of imaging tasks (defect pixel correction, denoising, white balancing, exposure correction, demosaicing, colour transform, and gamma encoding), we consider it is in a certain sense overloaded. See Fig. 2.1a for the illustration of the model structure.

(Schwartz et al. 2018) collects RAW low-lit images from Samsung S7 phone, and uses a neural network to improve image brightness and remove noise on demosaiced RGB images from a simple ISP pipeline. In this work, authors define a two-staged network that takes demosaiced images as inputs and lets the low-level stage produce an intermediate output, which would be further refined in the high-level stage. In this way, the network can learn a colour transformation between the raw input and refined output, which is similar to our work. Yet, our work not only learns the colour transformation but also gains the ability to recover high-level details by using a two-branch design. See Fig. 2.1b for the illustration of the model structure.



(A) Model structure of Deep Camera. Image is originally used in Ratnasingam 2019.

(B) Model structure of DeepISP. Image is originally used in Schwartz et al. 2018.

FIGURE 2.1: Networks that use RAW data.

Moreover, some previous works in AIM 2019 RAW to RGB Mapping Challenge have achieved appealing results. For example, (Uhm et al. 2019) considers using the stacked U-Nets to produce a pipeline in a coarse-to-fine manner. (Mei et al. 2019) adopts a multi-scale training strategy that recovers the image details while maintaining the global perceptual quality.

The most recent work (Ignatov et al. 2020) tries to narrow the visual quality gap between the mobile and DSLR colour images by directly translating mobile RAW images to DSLR colour ones, where RAW images are captured by Huawei P20 phone and colour ones are from Canon 5D Mark IV. In this work, authors present a multi-scale network that is trained layer-by-layer (see Fig. 2.2).

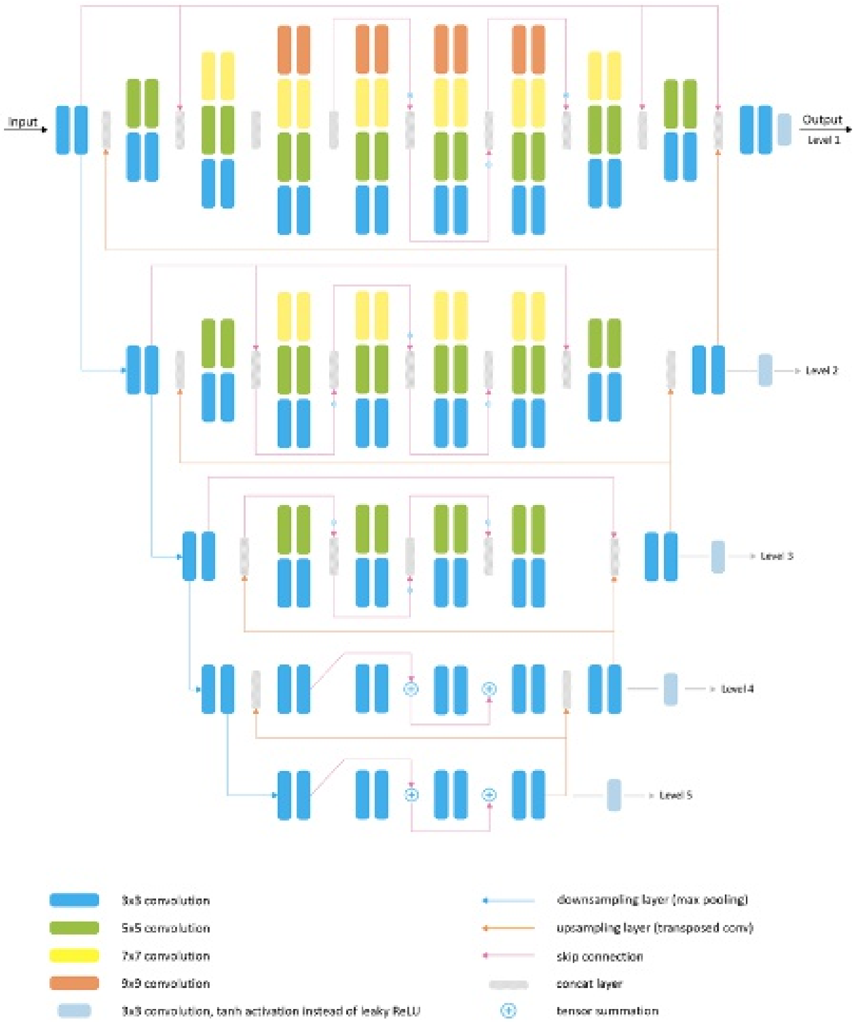


FIGURE 2.2: Structure of PyNet. Image is originally used in Ignatov et al. 2020.

Nonetheless, all previous learnable ISP methods only focus on the general mapping problem without mentioning other artifacts from the training dataset. For example, without additional operation, the misalignment between the DSLR and mobile image pairs can cause severe degradation on estimated outputs. In our work, we apply the global context block combined with the res-dense block that learns the global colour mapping to tackle misaligned image features. The added blocks enable our network to outperform the current state-of-the-art method proposed by (Ignatov et al. 2020).

2.4 Attention Mechanisms

Attention mechanisms have been employed in various deep learning models for performance enhancement in different tasks. Vaswani et al. (Vaswani et al. 2017) apply the self-attention model to machine translation. Hu et al. (Hu et al. 2018) propose a channel-wise attention mechanism called squeeze and excitation (SE) block, which yields promising results on different computer vision tasks. Specifically, the authors apply a "squeeze" operation to compress the number of channels and produce some channel descriptors. In this way, channel descriptors can aggregate feature maps across their spatial dimensions, and indicate important channels that have more informative features. The structure of the SE block can be viewed from Fig. 2.3.

Wang et al. (Wang et al. 2018) introduce a non-local module to measure the spatial information using a correlation matrix, which is then used as a form of attention to guide the contextual information aggregation. Following this work, a series of papers (Fu et al. 2019; Huang et al. 2019; Cao et al. 2019) leverage the

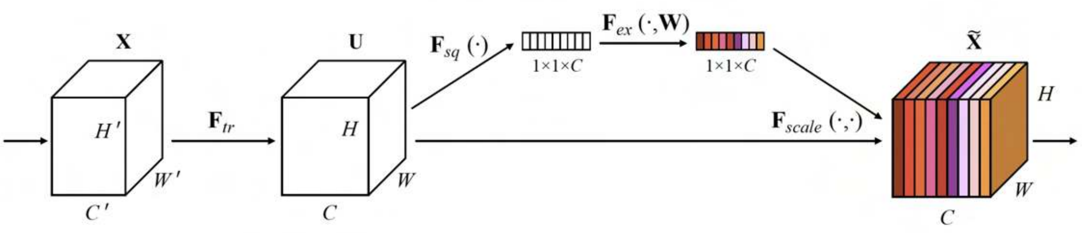


FIGURE 2.3: Structure of SE block. Image is originally used in Hu et al. 2018.

non-local module to guide spatial or channel-wise learning. Some examples of the non-local block can be seen in Fig. 2.4.

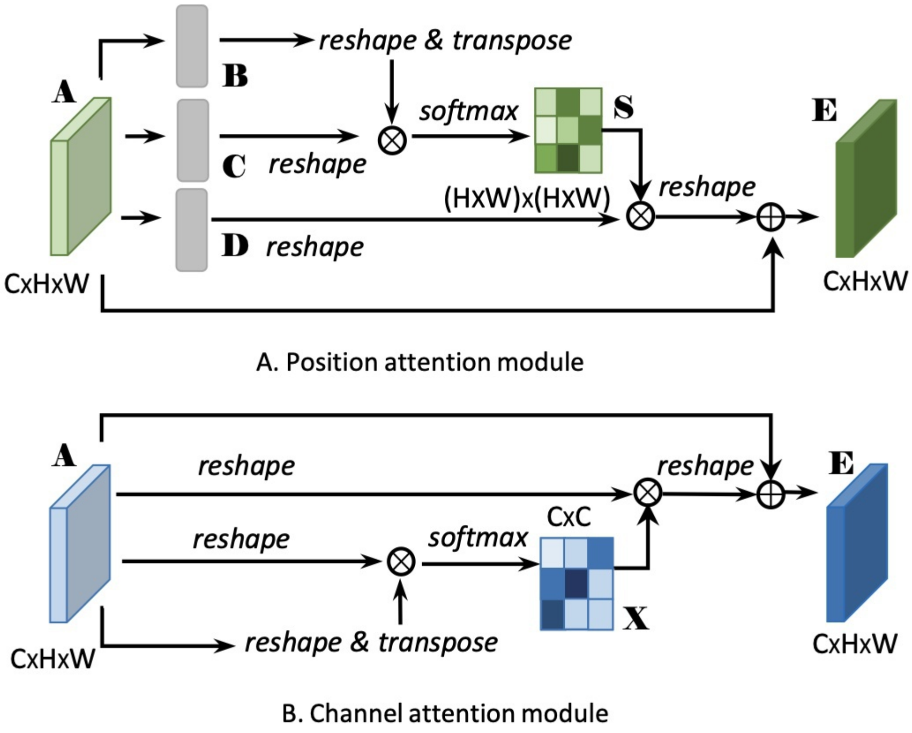


FIGURE 2.4: Example structure of non-local attention. Note that in the example work, author apply non-local operation on both channel and spatial domains. Image is originally used in Fu et al. 2019.

In low-level image restoration tasks, Zhang et al. (Zhang et al. 2018a) use channel and spatial attention blocks in single-image super-resolution tasks to make the network focus on more informative features from channel-wise and spatial-wise aspects, respectively. Similar to this setting, Liu et al. (Liu et al. 2019b) use an attention mechanism to boost their network performance in image dehazing tasks. In image denoising, Anwar et al. (Anwar and Barnes 2019) extend the idea of (Hu et al. 2018) and propose feature attention that is able to capture the channel dependencies from the global average pooling features. In our work, we adopt both the ideas of (Hu et al. 2018) and (Cao et al. 2019) to ensure that our network has the ability to distinguish the importance of different features from both spatial and channel perspectives. More details of our implementation can be seen from chapter 3.

2.5 Wavelet Transform in Deep Learning

Wavelet-based methods have been widely deployed in different low-level computer vision tasks. Many of them focus on image super-resolution problems (Ji and Fermüller 2008; Nguyen and Milanfar 2000), which utilize a sequence of low-resolution images to inference a high-resolution image. For traditional single image super-resolution task, wavelet transform is used for interpolation-based and static-based algorithms. Naik et al. (Naik and Patel 2013) introduce a modified version of classical wavelet-based interpolation method. Mallat (Mallat 1996) applies wavelet transform to extract information from data at different scales.

In the deep learning field, wavelet transform is often used to preserve the feature information for better feature map reconstruction. Gao et al. (Gao and Xiong

2016) propose a hybrid end-to-end wavelet convolution network. This network is able to generate a set of sparse code candidates and weigh these candidates to reconstruct the high-dimensional signal. In addition, Huang et al. (Huang et al. 2017) propose a wavelet-based CNN approach that can resolve a face image from a very low-resolution setting. Guo et al. (Guo et al. 2017) design a deep CNN to predict the information loss of wavelet coefficients of the low-resolution images, and use them, in conjunction with original wavelet subbands, to generate high-resolution results. Note that this network is trained in the wavelet domain, which means that both input and output of this network are wavelet subbands for low-resolution and high-resolution images, respectively.

Different from most wavelet-based CNNs, which manually extract the wavelet subbands and use those for feature reconstruction, Huang et al. (Huang et al. 2017) predict the wavelet components for high-resolution images and then reconstruct them to obtain the final result. The structure of this network is shown in Fig. 2.5.

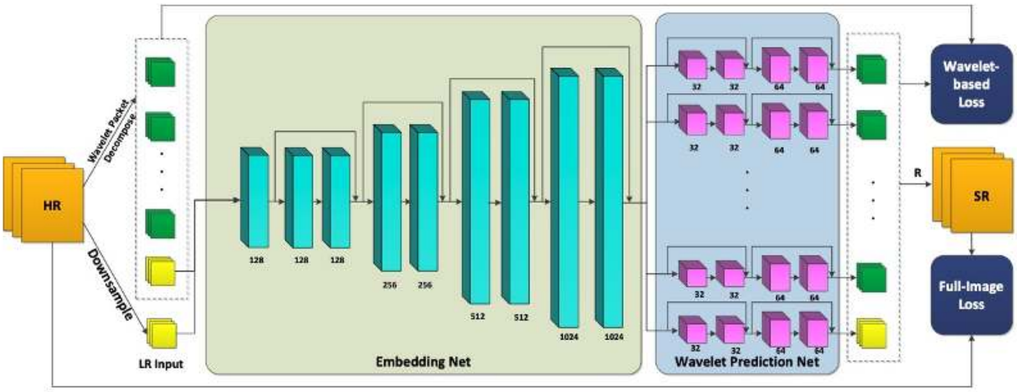


FIGURE 2.5: Structure of Wavelet-SRNet. Image is originally used in Huang et al. 2017.

Liu et al. (Liu et al. 2019a) introduce a multi-level wavelet CNN (MWCNN) model which uses wavelet transform in between the encoder and decoder layers to reconstruct the feature map. By doing this, the network gains a larger receptive field while preserving information. In addition, the computational cost is much reduced. A detailed explanation is provided in chapter 3. The structure of MWCNN can be viewed in Fig. 2.6.

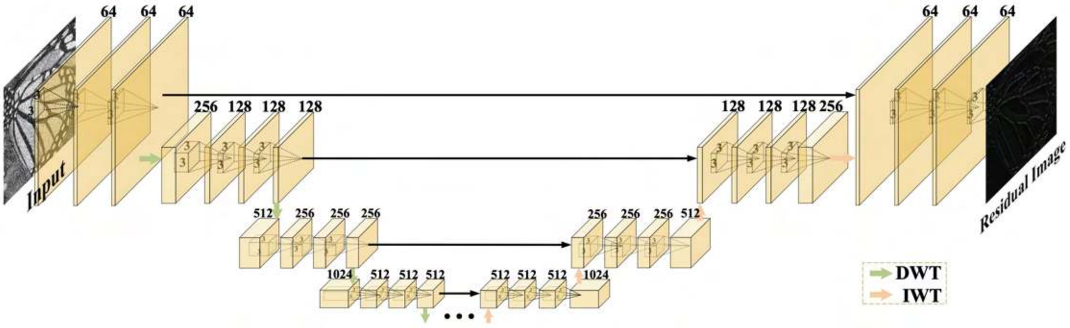


FIGURE 2.6: Structure of MWCNN. Note that image is originally used in Liu et al. 2019a.

By following the idea of (Liu et al. 2019a), Luo et al. (Luo et al. 2020) propose a deep wavelet network (AWDUN) that treats discrete wavelet transform (DWT) and inverse discrete wavelet transform (IDWT) as a sampling technique. This approach achieves stunning results on the image demoiring task. The structure of AWDUN can be viewed in Fig. 2.7.

Inspired by (Liu et al. 2019a) and (Luo et al. 2020), our method also makes use of DWT and IDWT to conduct upsampling and downsampling, respectively. The benefit is mani-fold. First, by the feature of the wavelet transform, the frequency components of the input data can reconstructed the original input in a lossless way (See chapter 3 for more explanation), thus, it is beneficial to have DWT and IDWT in the sampling module to prevent information loss. Moreover, due to the nature of wavelet transform, the process of DWT and IDWT can be treated as

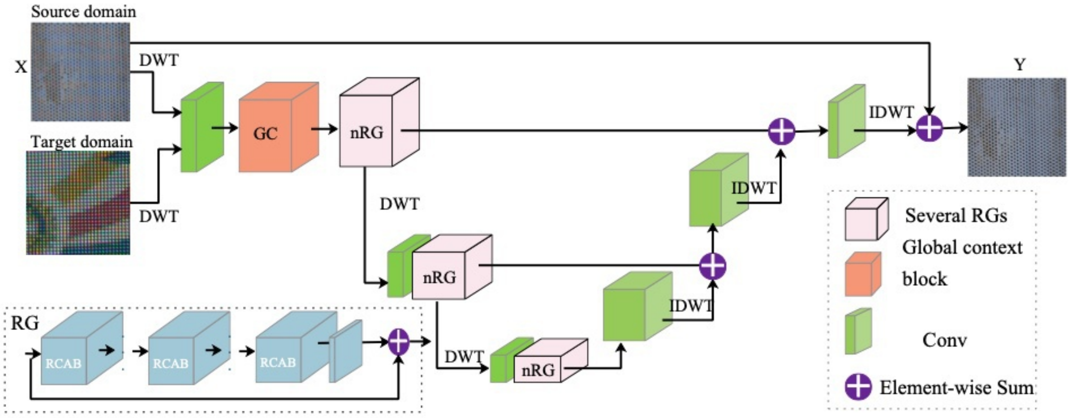


FIGURE 2.7: Structure of AWDUN. Image is originally used in Luo et al. 2020.

a good substitution of other sampling methods such as interpolation or pooling. In addition, the computational cost of DWT (IDWT) is low on GPU as it does not involve element-wise operation (other than subsampling). Different from the previous work, we design a novel sampling block that not only uses the wavelet transform but also convolutions to sample the feature. In this way, we give the ability to the network to learn the importance of the feature based on its spatial representation. Therefore, our work is operated on both spatial and frequency domains.

Chapter 3

Proposed Method

We describe the proposed method and training strategy in this section. First, the overall network architecture (shown in Fig. 3.1) and details of each network module are demonstrated, and then the rationale behind this design is illustrated. In the end, the loss functions adopted in training are introduced.

3.1 Network Structure

The proposed AWNet employs a U-Net resembled structure and consolidates the architecture by three main modules, namely global context res-dense module, residual wavelet up-sampling module, and residual wavelet down-sampling module (see Fig. 3.2 and Fig. 3.4).

The global context res-dense module consists of a residual dense block (RDB) and a global context block (GCB) (Cao et al. 2019). The effectiveness of RDB has been comprehensively examined (Liu et al. 2019b; Zhang et al. 2018b). Here, learning the residual information is beneficial to the colour-mapping performance.

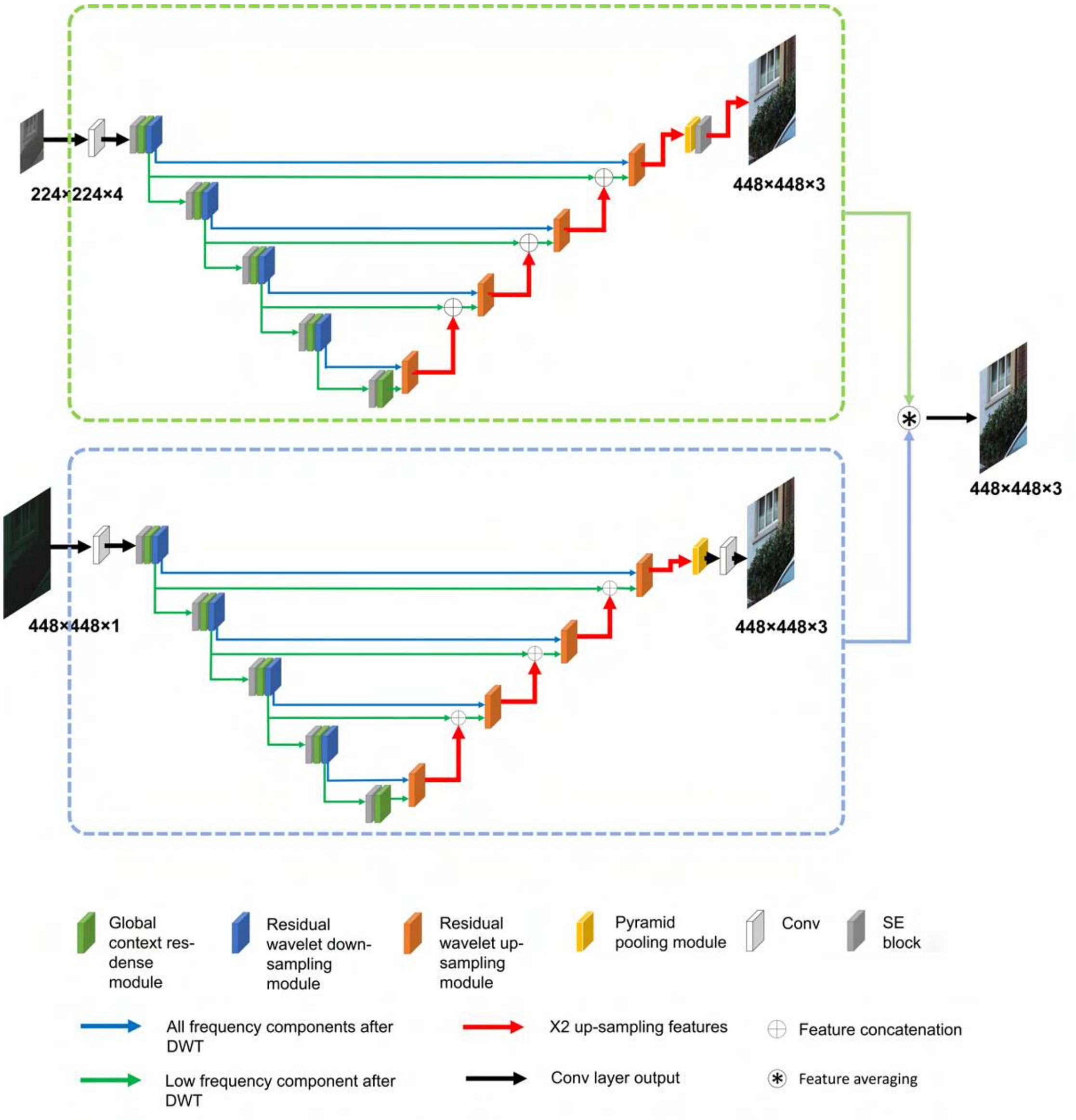

224×224×4
448×448×3
448×448×1
448×448×3
448×448×3
Global context res-dense module
Residual wavelet down-sampling module
Residual wavelet up-sampling module
Pyramid pooling module
Conv
SE block
All frequency components after DWT
X2 up-sampling features
Feature concatenation
Low frequency component after DWT
Conv layer output
Feature averaging

Figure 3.1: The main architecture of the proposed AWNet. The top and bottom ones are the demosaiced and RAW models, respectively. We take the average of both outputs from these two models to obtain the final prediction.

A total of seven convolutional layers are used in RDB, where the first six layers aim at increasing the number of feature maps and the last layer concatenates all feature maps generated from these layers. At the end of RDB, a global context

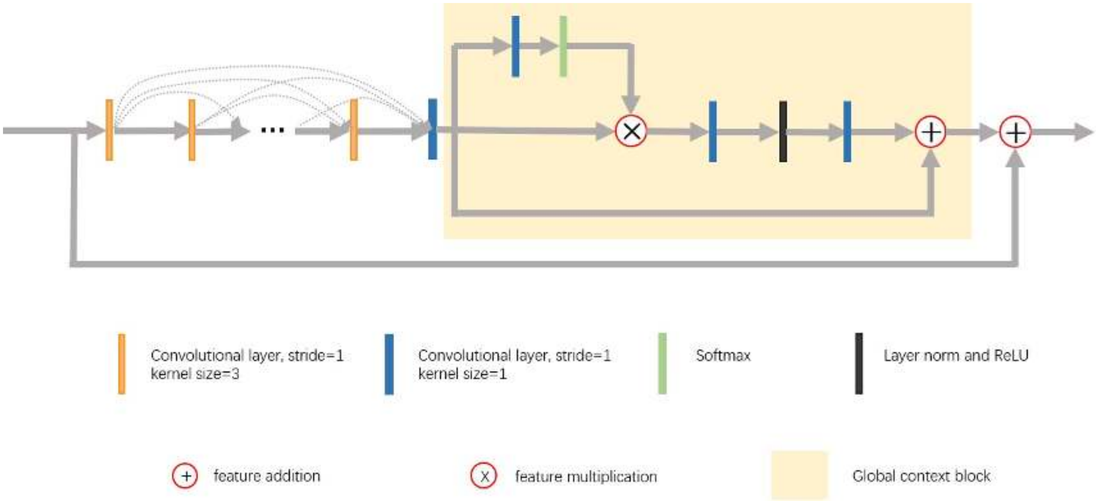


FIGURE 3.2: Our global context res-dense module contains a residual dense block (RDB) and a global context block (GCB). We observe that the RDB can benefit the color restoration from RAW images and the GCB encourages the network putting effort on learning the global color mapping. See details in chapter 4.4.

block is presented to encourage the network to learn the global colour mapping, since local colour mapping might cause the deterioration of the results due to the pixel misalignment between RAW and RGB image pairs. The reason is evident as the existence of misalignment misleads the neural network to map colour into incorrect pixel locations.

In view of the fact that the convolutional kernel only covers the local information of an image, (Wang et al. 2018) propose a non-local attention mechanism. This work can realize the dependency between long-distance pixels so that the value at a query point can be calculated by the weighted sum of the features of all positions on the input feature. However, heavy computation is required, especially when the feature map has a large size (e.g., the full resolution input image from the ZRR dataset).

To be specific, the non-local operation can be modelled as

$$y_i = \sum_{j=1}^{Np} \frac{f(x_i, x_j)}{C(X)} g(x_j). \tag{3.1}$$

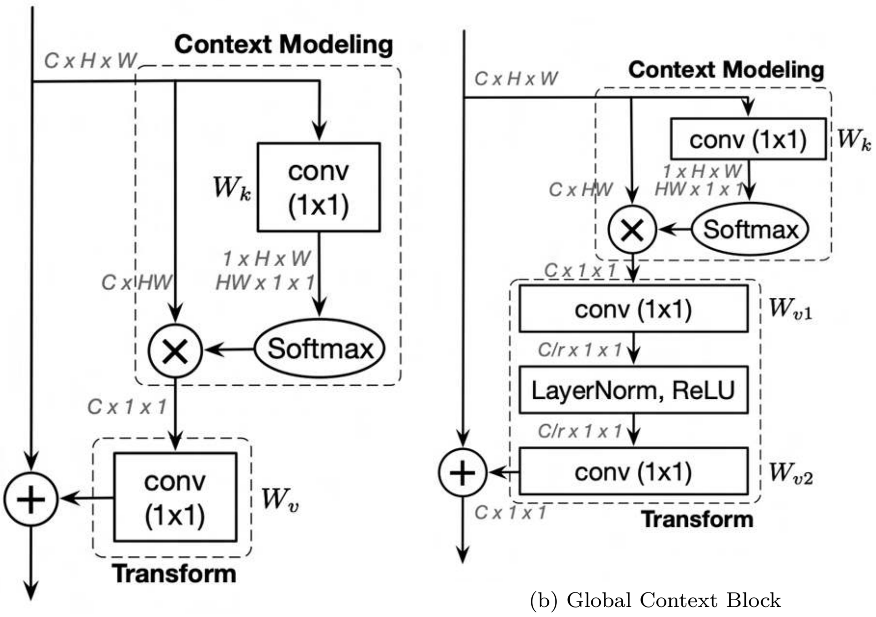
Note that y_i and x_i are the respective values of input and output features at query position i; j is the index that enumerates all possible positions; Np denotes the total number of positions ($H \times W$ for image); $g(\cdot)$ denotes the linear transformation function, e.g., 1×1 convolution; $C(X)$ is the normalization function; $f(x_i, x_j)$ denotes the function that measures the similarity between x_i and x_j. There are multiple ways to implement $f(x_i, x_j)$. The most common way is using Embedded Gaussian which is formulated as

$$f(x_i, x_j) = e^{\gamma(x_i)^T \beta(x_j)}. \tag{3.2}$$

Here, $\gamma(x_i)$ and $\beta(x_j)$ are two embeddings. $C(X)$ in this case is set to be $\sum_{\forall j} f(x_i, x_j)$. Even though the original non-local attention operation can catch the long-range dependency of each pixel, (Cao et al. 2019) claims that the attention map obtained from different query points has minor differences based on their experiments. Therefore, they propose a lightweight global context block (GCB) that simplifies the non-local module and can be combined with the global context framework and the SE block (Hu et al. 2018). To do that, they define their simplified non-local block as

$$y_i = x_i + W_z \sum_{j=1}^{Np} \frac{e^{W_k x_j}}{\sum_{m=1}^{Np} e^{W_k x_m}} x_j. \tag{3.3}$$

Here W_k and W_v are linear transformation matrices. It is obvious that Eq. 3.3 only depends on the pixel location j, which means that this attention algorithm calculates a global attentive weight for all pixels across the feature. Therefore, it is beneficial to apply this mechanism for facilitating global colour mapping. To further improve the non-local attention mechanism, GCB also applies squeeze and excitation operation to enable the network to identify which channel is more important. One can find the simplified non-local block and GCB in Fig. 3.4. The GCB encourages the network to learn key information spatial-wise and channel-wise while effectively reduce the computation complexity. These characteristics are exactly what we look for in this RAW-to-RGB mapping problem.



(a) Simplified Non-local Block

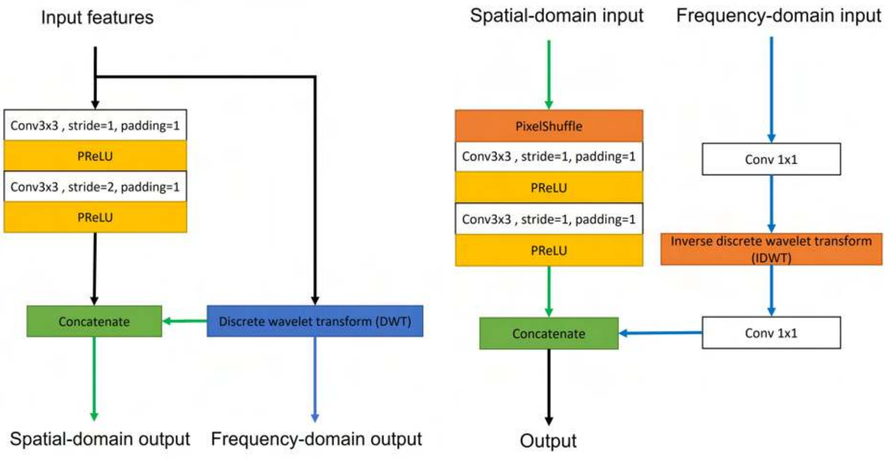
(b) Global Context Block

FIGURE 3.3: Illustration of the structure of simplified non-local block and Global Context Block. Note that the Global Context Block add squeeze-and-excitation operation after the non-local operation, which gives the network ability to do channel attention. Image is originally from Cao et al. 2019.

For up-sampling and down-sampling, we borrow the idea from the discrete wavelet transform (DWT), since the nature of DWT decomposes the input feature maps into the high-frequency and low-frequency components, in which the low-frequency one can serve as the result from average pooling (further discussion can be found in chapter 3.3). As shown in Fig. 3.4, we use the low-frequency component as part of our down-sampling feature maps and connect the high-frequency part to the up-sampling block for image recovery (i.e., inverse DWT). However, the feature maps produced by frequency-domain operation might be lack of spatial correlation. Therefore, an additional spatial convolutional layer is adopted to downsample the feature map with learned kernels. Similarly, a pixel-shuffle operation along with a spatial convolutional layer is employed for up-sampling as the complement to the IDWT. The combination of frequency-domain and spatial-domain operations facilitates the learning of abundant features in up-sampling and down-sampling blocks. At the end of the proposed method, we use a Pyramid Pooling block (Chen et al. 2018b) to further enlarge the receptive field.

3.2 Two-Branch Network

By consolidating the encoder-decoder structure with previously mentioned modules, our network is able to surpass the state-of-the-art when trained on the RAW images. However, using multiple neural networks to train on different low-level vision tasks is a more effective way to learn image ISP. One of the reasons is that feeding distinct data to different network branches can provide abundant information during training. Recently, the two-stream design has been successfully applied in various computer vision tasks, especially in the video field. Note that fusing the information from different formats of input (e.g., optical flow and image frames) can significantly improve the network performance. Inspired by (Carreira and Zisserman 2017; Feichtenhofer et al. 2016), we build AWNet based on the idea of two-branch architecture to facilitate network performance on different low-level imaging tasks by utilizing different inputs. Our two-branch design contains two encoder-decoder models, namely the RAW model and the demosaiced model. Here, the RAW model is trained on 224 × 224 × 4 RAW images, and the demosaiced branch takes 448 × 448 × 3 demosaiced images as input. For the RAW model, there is a need to make the prediction size and ground truth size consistent. Therefore, this branch pays more attention to the recovery of high-frequency details. For its counterpart, the demosaiced branch has no need to upscale the output size for consistency. Instead, this branch focuses more on the colour mapping between the demosaiced image and the RGB colour image. We train the two networks separately and average their predictions at testing. As expected, a great performance boost is observed by applying this architecture (see details in chatper 4.3).


Input features
Conv3x3 , stride=1, padding=1
PReLU
Conv3x3 , stride=2, padding=1
PReLU
Concatenate
Discrete wavelet transform (DWT)
Spatial-domain output
Frequency-domain output
Spatial-domain input
Frequency-domain input
PixelShuffle
Conv3x3 , stride=1, padding=1
PReLU
Conv3x3 , stride=1, padding=1
PReLU
Conv 1x1
Inverse discrete wavelet transform (IDWT)
Concatenate
Conv 1x1
Output

(a) Residual Wavelet Down-sampling Block (b) Residual Wavelet Up-sampling Block

FIGURE 3.4: Illustration of our up-sampling and down-sampling modules in Fig. 3.1. The residual design enables our model to operate in frequency-domain and spatial-domain that facilitates the learning of abundant features in up-sampling and down-sampling blocks.

3.3 Discrete Wavelet Transform

To elaborate on the reason for choosing DWT in our design opinion, we introduce the connection between DWT and traditional pooling operation. In 2D discrete wavelet transform, there are four filters, i.e., f_{LL}, f_{LH}, f_{HL}, and f_{HH}, that can be used to decomposed an image (Mallat 1989).

To be more specific, in Haar DWT, these four filters are defined as

$$f_{LL} = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right), f_{LH} = \left(\begin{smallmatrix} -1 & -1 \\ 1 & 1 \end{smallmatrix}\right), f_{HL} = \left(\begin{smallmatrix} -1 & 1 \\ -1 & 1 \end{smallmatrix}\right), f_{HH} = \left(\begin{smallmatrix} 1 & -1 \\ -1 & 1 \end{smallmatrix}\right). \tag{3.4}$$

We see that f_{LL}, f_{LH}, f_{HL}, and f_{HH} are orthogonal to each other. By convolving the image with each filter, a full-size image x is split into 4 sub-bands, i.e., x_{LL}, x_{LH}, x_{HL}, and x_{HH}. Due to the nature of DWT, this process can be expressed as

$$\begin{aligned} x_{LL} &= (f_{LL} \circledast x) \downarrow_2, \\ x_{LH} &= (f_{LH} \circledast x) \downarrow_2, \\ x_{HL} &= (f_{HL} \circledast x) \downarrow_2, \\ x_{HH} &= (f_{HH} \circledast x) \downarrow_2, \end{aligned} \tag{3.5}$$

where $\circledast$ represents a convolutional operation and $\downarrow_2$ indicates down-sampling by a scale factor of 2. It is evident that the DWT operation can be treated as a convolutional downsampling operator with stride equal to 2.

According to Eqn. 3.5, the (m, n)-th value of x_{LL} after 2D Haar wavelet transform can be defined as

$$\begin{aligned}
x_{LL}(m,n) &= x(2m-1,2n-1) + x(2m-1,2n) + x(2m,2n-1) + x(2m,2n),\\
x_{LH}(m,n) &= -x(2m-1,2n-1) - x(2m-1,2n) + x(2m,2n-1) + x(2m,2n),\\
x_{HL}(m,n) &= -x(2m-1,2n-1) + x(2m-1,2n) - x(2m,2n-1) + x(2m,2n),\\
x_{HH}(m,n) &= x(2m-1,2n-1) - x(2m-1,2n) - x(2m,2n-1) + x(2m,2n).
\end{aligned} \tag{3.6}$$

Even though DWT is a subsampling operation, the original feature x can be restored by IDWT, i.e., $x = IDWT(x_{LL}, x_{LH}, x_{HL}, x_{HH})$. Here IDWT can be defined as

$$\begin{aligned}
x(2m-1,2n-1) &= (x_{LL}(m,n) - x_{LH}(m,n) - x_{HL}(m,n) + x_{HH}(m,n))/4,\\
x(2m-1,2n) &= (x_{LL}(m,n) - x_{LH}(m,n) - x_{HL}(m,n) + x_{HH}(m,n))/4,\\
x(2m,2n-1) &= (x_{LL}(m,n) - x_{LH}(m,n) - x_{HL}(m,n) + x_{HH}(m,n))/4,\\
x(2m,2n) &= (x_{LL}(m,n) - x_{LH}(m,n) - x_{HL}(m,n) + x_{HH}(m,n))/4.
\end{aligned} \tag{3.7}$$

It is obvious that the restoration is lossless if you have all 4 frequency components. Moreover, by defining x_p to be the feature map after p-level of average pooling, the (m, n)-th value of x_p can be expressed as

$$\begin{aligned} x_p(m, n) = 0.25 \times (&x_{p-1}(2m-1, 2n-1) + x_{p-1}(2m-1, 2n) \\ &+ x_{p-1}(2m, 2n-1) + x_{p-1}(2m, 2n)). \end{aligned} \tag{3.8}$$

As we can see, Eq. (3.8) is highly correlated with the equation of low frequency component in Eq. (3.7). In comparison, by taking four subbands into account, the pooling operation discards all the high-frequency components and only makes use of the low-frequency part. Therefore, the information loss in traditional pooling operations is severe. To alleviate this problem, we design our up-sampling and down-sampling modules in a way that uses both wavelet transform and convolutional operation to manage to scale (See Fig. 3.4). To be more specific, our downsampling block takes the input data and decomposes it into four subbands, where the low-frequency component will not only be circulated as a downsampled feature but will also be used as high-level features with other frequency components and skip-connected to the upsampling block. Within the upsampling block, the low-level feature will be used to reconstruct the high-level features, by using both IDWT and pixel shuffling (Shi et al. 2016).

By doing that, our network can learn from both spatial and frequency information. Our experiments reveal the superior performance of this design (see details in chapter 4.4).

3.4 Loss Function

In this section, we introduce our three loss functions and the multi-scale loss strategy. We denote I as the target RGB image and $\tilde{I}$ as the predicted result from our method.

Pixel loss We adopt the Charbonnier (Bruhn et al. 2005; Zhang et al. 2018a) loss as an approximate L_1 term for our loss function to better handle outliers and improve the performance. From previous experiments, we realize that Charbonnier loss can efficiently improve the performance of the signal-to-noise ratio of reconstructed images. In addition, Charbonnier loss has been applied in multiple image reconstruction tasks and outperforms the traditional L_2 penalty (Zhang et al. 2018a). The Charbonnier penalty function is defined as:

$$L_{char} = \sqrt{(\tilde{I} - I)^2 + \epsilon^2}, \tag{3.9}$$

where we set ϵ to $1e - 3$. Note that using only the pixel loss on RAW-to-RGB mapping results in blurry images as reported in (Uhm et al. 2019). Thus, we redeem this problem by adding other feature loss functions.

Perceptual loss. To deal with the pixel misalignment problem from ZRR dataset (See Fig. 3.5), we also employ perceptual loss. The loss function is defined as

$$L_P = L_{MSE}(F(\tilde{I}) - F(I)), \tag{3.10}$$

where F denotes the pretrained VGG-19 network, $\tilde{I}$ and I represent the predicted image and ground truth, respectively. As misaligned images are processed by the pre-trained VGG network, the resulting downsampled feature maps have fewer variants in terms of the misalignment. Therefore, adding a L_2 term on such feature maps is beneficial for the network to recognize the global information and minimize the perceptual difference between the reconstructed image and the ground truth image.

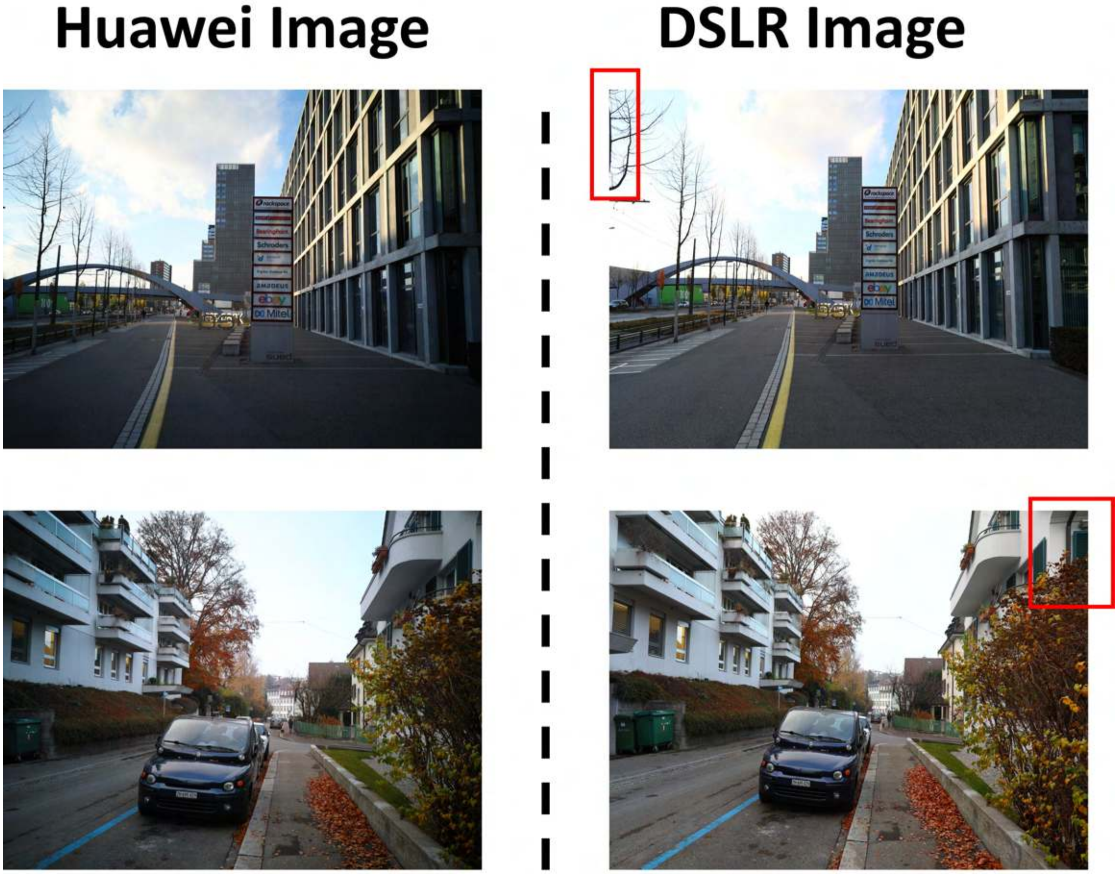


FIGURE 3.5: Some examples for the misalignment problem from ZRR dataset. Misaligned areas are highlighted by red boxes.

SSIM loss. We also employ the structural similarity (SSIM) loss L_{SSIM} (Wang et al. 2003) that is aiming to reconstruct the RGB images by enhancing on structural similarity index. The resulting images are more perceptually acceptable than the predictions without applying SSIM loss. Note that the SSIM loss can be defined as:

$$L_{SSIM} = 1 - F_{SSIM}(\tilde{I} - I), \tag{3.11}$$

where F denotes the function of calculating the structural similarity index.

Multi-scale loss function. Inspired by (Qian et al. 2018), we apply supervision on outputs from different decoder layers to refine reconstructed images of different sizes. For each scale level, we focus on different restoration aspects, thus different loss combinations are applied. In our RAW model, there are 5 up-sampling operations, which form feature maps in 6 different scales, named as scale 1-6 from small to large. Similarly, there are 5 different scales presented in the demosaiced model and we name those as scales 1-5.

1). Scale 1-2 process feature maps that are down-scaled by a factor of 16 and 32. The feature maps at this scale contain less context information compared with ground truth. Thus, we mainly focus on global colour and tone mapping. These layers are supervised only by Charbonnier loss, which can be written as:

$$L_{1,2} = L_{char}. \tag{3.12}$$

2). Scale 3-4 are computed on feature maps with down-scaled factors of 4 and 8; since these features are smaller as compared to the size of ground truth yet contain richer information than the scale 1-2, we apply a loss combination that incorporates perceptual and Charbonnier losses to perform global mapping while maintaining the perceptual quality. The loss function of these layers is defined as:

$$L_{3,4} = L_{char} + 0.25 \times L_P. \tag{3.13}$$

3) In scale 5-6, the size of feature maps is close or equal to the original one, thus we are able to pay more attention to the recovery of image context in addition to the color mapping. We choose a more comprehensive loss combination at this level, which can be expressed as:

$$L_{5,6} = L_{char} + 0.25 \times L_P + 0.05 \times L_{SSIM}. \tag{3.14}$$

Note that we manually choose the coefficients of different loss terms. The total loss function can be expressed as:

$$L_{total} = \sum_{n=1}^{k} L_n, \tag{3.15}$$

where k is equal to 5 and 6 for demosaiced model and RAW model, respectively.

Chapter 4

Implementation and Experiment Results

We conduct comprehensive experiments to demonstrate that the proposed method performs favourably against the baseline model (Ignatov et al. 2020) in terms of quantitative and qualitative comparisons on the ZRR dataset.

4.1 Datasets

To enhance smartphone images, the Zurich dataset from AIM 2020 Learned Smartphone ISP Challenge (Ignatov et al. 2020) provides 48043 RAW-RGB image pairs (of size $448 \times 448 \times 1$ and $448 \times 448 \times 3$, respectively). The training data is divided into 46,839 image pairs for training and 1,204 ones for testing. In addition, 168 full resolution image pairs are used for perceptual validation. For data preprocessing and augmentation, we normalize the input data and perform vertical and horizontal flipping.

TABLE 4.1: Validation scores by different model ensembles. We use red text to indicate the best performance and blue text to indicate the second best performance. We adopt weights from the last row for the testing stage during the AIM2020 Learned Smartphone ISP Challenge.

RAW model PSNR (dB) / SSIM	Demosaiced model PSNR (dB) / SSIM	Ensemble Score PSNR (dB) / SSIM
21.36 / 0.7429	21.30 / 0.7455	21.60 / 0.7818
21.36 / 0.7429	21.38 / 0.7522	21.92 / 0.7761
21.36 / 0.7429	21.52 / 0.7484	21.95 / 0.7788
21.36 / 0.7429	21.58 / 0.7488	21.79 / 0.7818
21.38 / 0.7451	**21.58 / 0.7488**	**21.97 / 0.7784**

TABLE 4.2: The result of AIM2020 Learned Smartphone ISP Challenge for the two tracks. Our method can achieve high MOS while remaining competetive in PSNR and SSIM metrics.

	Track 1			Track 2			
Rank	Method	PSNR	SSIM	Method	PSNR	SSIM	MOS
1	Airia_CG	22.2574	0.7913	MW-ISPNet	21.574	0.777	4.7
2	skyb	21.9263	0.7865	**AWNet**	**21.861**	**0.7807**	**4.5**
3	MW-ISPNet	21.9149	0.7842	Baidu	21.9089	0.7829	4.0
4	Baidu	21.9089	0.7829	skyb	21.734	0.7891	3.8
5	**AWNet**	**21.8610**	**0.7807**	STAIR	21.569	0.7846	3.5

4.2 Training Details

Our model is trained on PyTorch framework with Intel i7, 32GB of RAM, and two NVIDIA RTX2080 Ti GPUs. The batch size is set to 6 and 2 for the RAW model and the demosaiced model, respectively. Except for that, our two models share the same training strategy. We employ Adam optimizer (Kingma and Ba 2014) with $\beta_1 = 0.9, \beta_2 = 0.999$ and set the initial learning rate as 1×10^{-4}. We decrease the learning rate by half every 10 epochs and train for 50 epochs in total.

4.3 Ensemble Strategy

Inspired by (Timofte et al. 2016), we apply a self-ensemble mechanism during the validation and testing stage of the AIM2020 Learned Smartphone ISP Challenge. Specifically, we use ensembles comprised of 8 variants (original, rotated 90°, rotated 180°, rotated 270°, rotated 90° & flipped, rotated 180° & flipped, and rotated 270° & flipped ones). After that, we average out the ensemble outputs and obtain our final result. To evaluate the benefit of ensembles, we apply our method to the validation dataset (without ground truth) during the development stage to validate our methods by calculating the PSNR values. In our experiments, the non-ensembles version of the RAW model and the demosaiced model in Track 1 achieves 21.55 dB and 21.68 dB on the validation dataset (without ground truth), respectively. Subsequently, by averaging out the results from both models, the PSNR can be boosted to 21.97 dB. To achieve optimal ensemble result, for each model, we prepare weights with different PSNR scores and then carry out experiments to test different combinations of weights across two models (see Table 4.1 for details). At the final testing stage, we choose the 21.36 dB (RAW model) and 21.52 dB (demosaiced model) weights to generate predictions. Fig. 4.1 shows the qualitative and quantitative results from these models and their ensemble outcomes (tested on offline validation data from provided ZRR dataset). Table 4.2 shows the result of AIM2020 Learned Smartphone ISP Challenge (Ignatov, Timofte, et al. 2020) for the two tracks. We are ranked in the 5^{th} and 2^{nd} place in tracks 1 and 2, respectively.

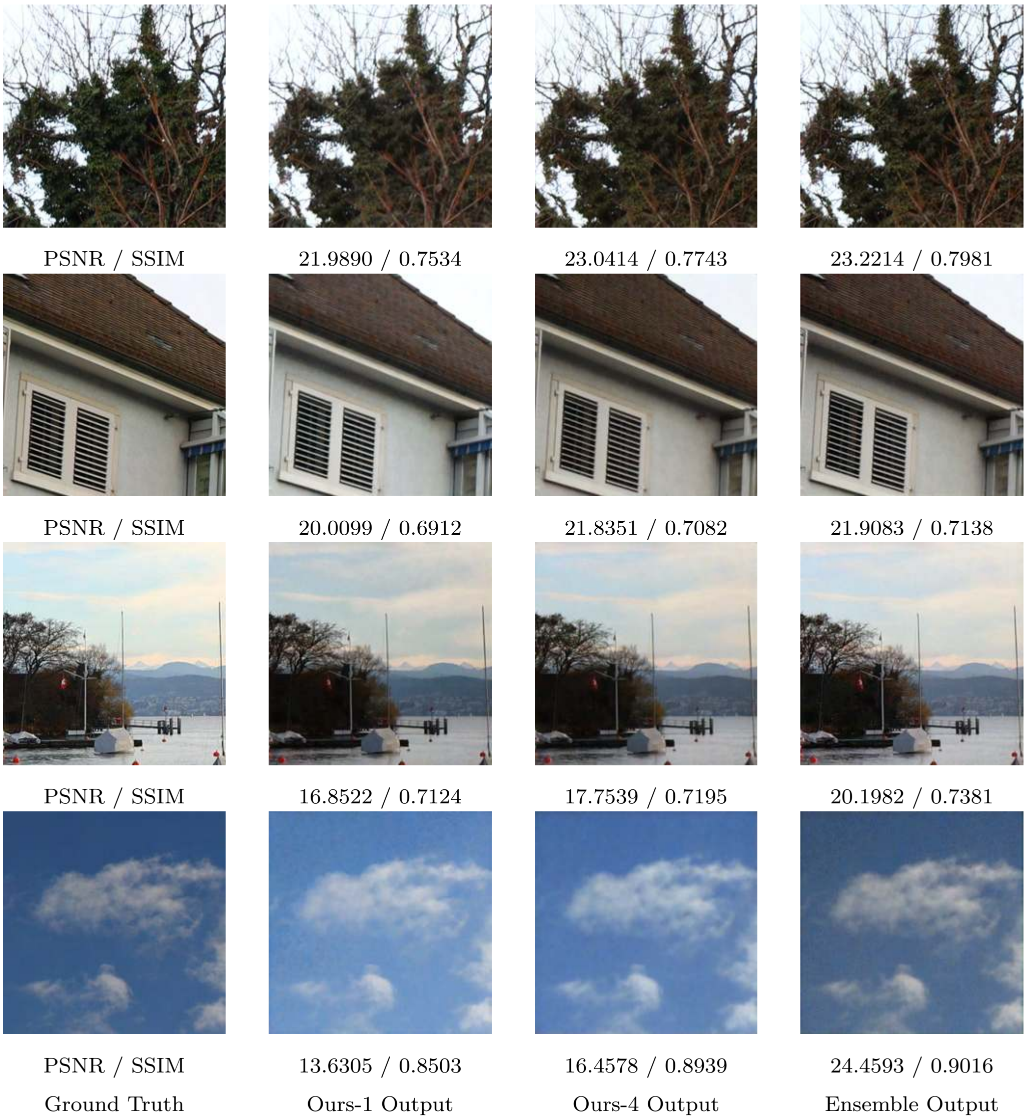


FIGURE 4.1: PSNR/SSIM and visual comparisons of reconstructed images from different network models. Ours-4 and Ours-1 denote our demosaiced and RAW models, respectively.

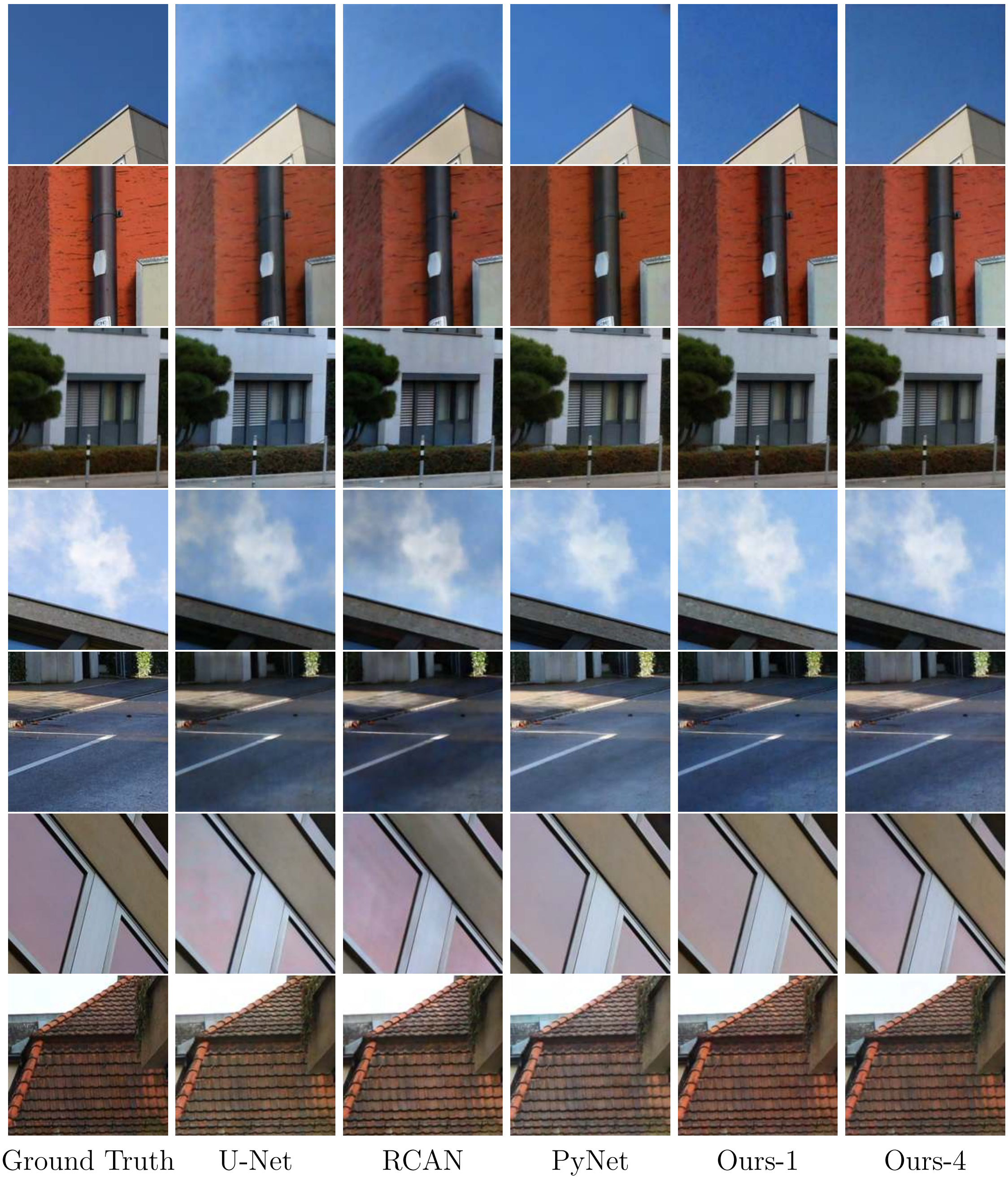


FIGURE 4.2: Qualitative comparisons of reconstructed images from different networks. Ours-4 and Ours-1 denote our demosaiced and RAW models, respectively.

4.4 Performance Comparisons and Ablation Studies

We conduct an experiment by first comparing it with other state-of-the-arts to demonstrate the superior performance of our method. After that, we provide solid justification for the effectiveness of wavelet transform and global context blocks. Our proposed method is tested on offline validation data that is provided during the development stage. We choose some popular network architectures from different computer vision tasks, including UNet and RCAN, for comparisons. The qualitative comparisons can be seen from Table 4.3, and Fig. 4.2 shows the qualitative comparison between our method and other state-of-the-arts. As we can see, both U-Net and RCAN have some colour mapping artifacts, which manifests the incapability of mapping colour into RGB space correctly in a pixel-to-pixel manner. For example, in the first row of Fig. 4.2, the colour of the sky is inaccurately predicted. Although the PyNet performs better in the colour mapping aspect, it tends to obscure the image details. This artifact is obvious in the second, the third, and the last row of images. Beneficial from DWT and GCB blocks, the proposed method remedies these artifacts, which are present in other state-of-the-arts. Moreover, the RAW model provides more fine image details whereas the demosaiced model has a better matching in colour space; this reveals the effectiveness of our design.

To validate that the wavelet transform and GCB blocks manage to improve the output performance, two corresponding experiments are conducted. The first one is to remove wavelet transform and GCB blocks (see Fig. 3.4) from the residual

Table 4.3: Quantitative results from different models. Both of our proposed models outperform the state-of-the-arts. Ours-4 and Ours-1 indicate our demosaiced and RAW models, respectively.

Models	PSNR (dB) / SSIM
U-Net	21.01 / **0.7520**
RCAN	20.85 / 0.7510
PyNet	21.17 / 0.7460
Ours-1	**21.58** / 0.7488
Ours-4	21.38 / 0.7451

wavelet up-sampling module, residual wavelet down-sampling module, and global context res-dense module; another one is to restore GCB blocks and leave wavelet transform blocks absent. As shown in Table 4.4, by adding GCB blocks, both of our models can be boosted by 0.1 dB in terms of PSNR metric. The performance can be further improved by 0.2 dB with the inclusion of the DWT block. Note that all these variants are trained in the same way as before and tested on the offline validation dataset from AIM2020 Learned Smartphone ISP Challenge.

In order to visualize the effectiveness of our two-branch design, we compare the validation results from our RAW and demosaic models and their ensembles using the proposed training strategy. Fig. 4.3 reveals that the ensemble of RAW and the demosaiced model is able to combine the advantages of each model and produce images with accurate colour and fine details. To demonstrate the success of our loss selection when dealing with image misalignment, we train the demosaiced model by applying only L_{char} at the original scale level. As Fig. 4.3 shown, the results with only L_{char} leads to blurry image details and inaccurate colour mapping

To better demonstrate the qualitative result of the proposed algorithm on the real-world camera image, we test the 4-channel model, 3-channel model, and their

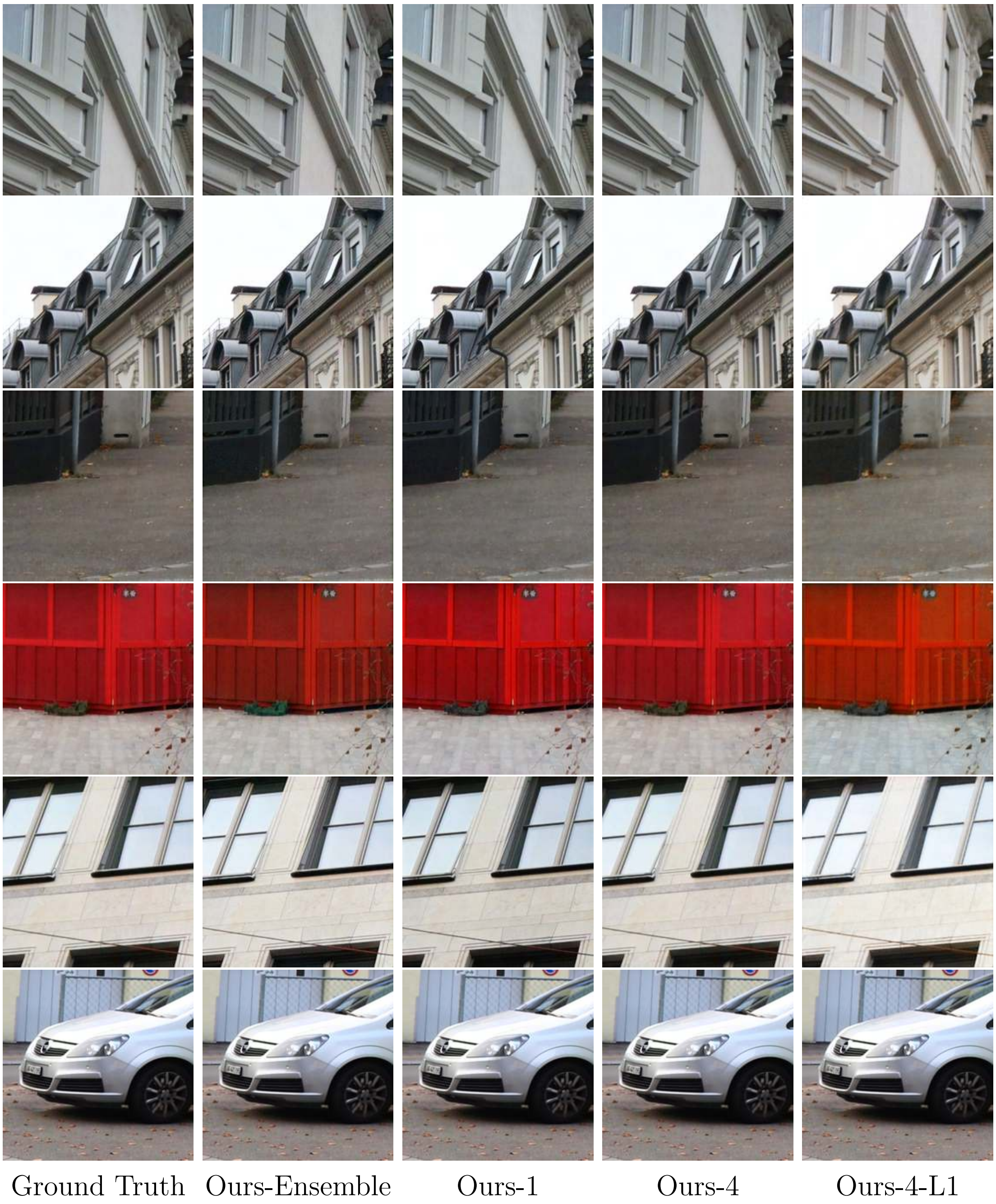


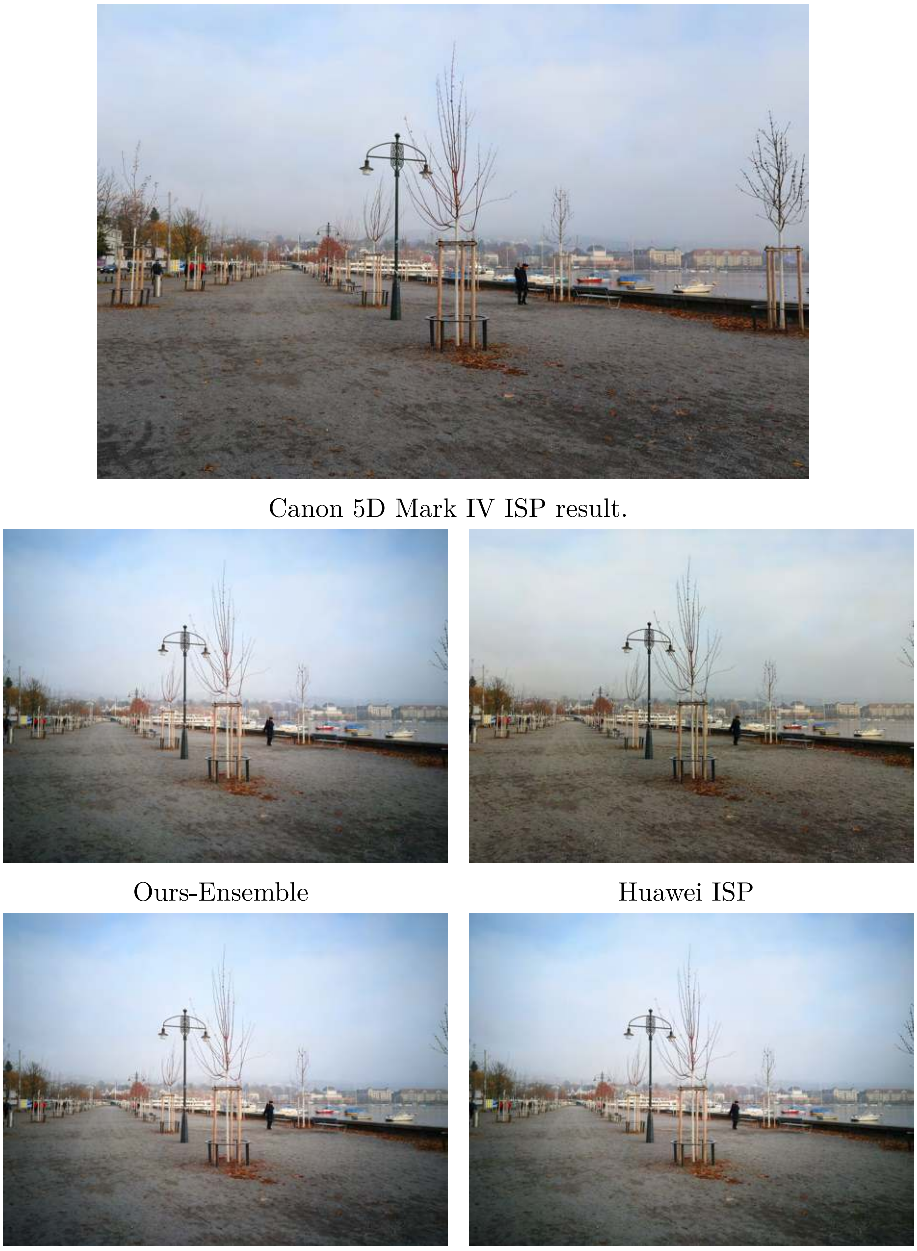
FIGURE 4.3: Qualitative comparisons of reconstructed images from different proposed models. Ours-Ensemble the averaging result of the RAW model and the demosaiced model. Ours-4 and Ours-1 denote our demosaiced and RAW models, respectively. Ours-4-L1 stands for the demosaiced model with only L_{char} loss.

Table 4.4: The benefit of using DWT and GCB blocks is evident. Both of our models can receive approximate 0.3 dB gains.

Model	Operation	PSNR (dB) \ SSIM
Demosaiced model	w/o DWT and w/o GCB	21.13 / 0.7398
	w/o DWT	21.22 / 0.7421
	proposed model	**21.38 / 0.7451**
RAW model	w/o DWT and w/o GCB	21.22 / 0.7325
	w/o DWT	21.31 / 0.7398
	proposed model	**21.58 / 0.7488**

ensemble results for full-resolution RAW images provided in the ZRR dataset (Ignatov et al. 2020). As shown in Fig. 4.5, the ensemble results from our AWNet display a more accurate image colour compared with Huawei ISP images that make our result more perceptually acceptable.

Our qualitative and quantitative results validate the superiority of our two-branch design as well as the effectiveness of wavelet transform block and attention mechanism, in the application of learning RAW-to-RGB colour mapping.



Canon 5D Mark IV ISP result.

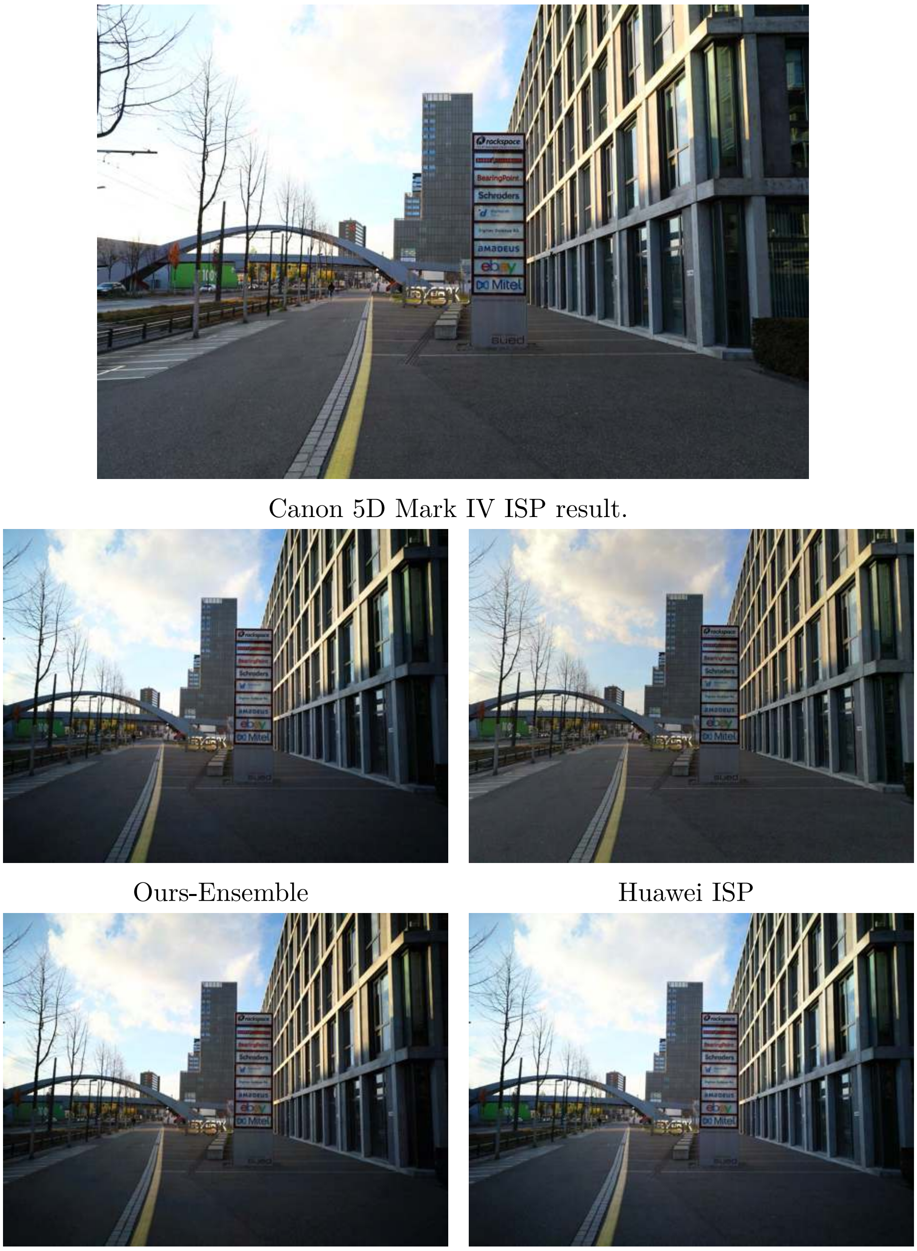
Ours-Ensemble

Huawei ISP

Ours-1

Ours-4

Figure 4.4: Visualization of full-resolution results generated by different methods. Ours-Ensemble represents the averaging result of the RAW model and the demosaiced model. Ours-4 and Ours-1 denote our demosaiced and RAW models, respectively.



Canon 5D Mark IV ISP result.

Ours-Ensemble

Huawei ISP

Ours-1

Ours-4

Figure 4.5: Visualization of full-resolution results generated by different methods. Ours-Ensemble represents the averaging result of the RAW model and the demosaiced model. Ours-4 and Ours-1 denote our demosaiced and RAW models, respectively.

Chapter 5

Conclusion and Future Work

In this paper, we propose a novel two-branch network structure, named AWNet, which can effectively enhance smartphone images. We embed wavelet transform blocks into the scaling modules associated with convolutional operations that enable our network to learn from both the spatial and frequency domains. As a consequence, our model can mitigate the information loss while processing the feature. In addition, the presence of GCB blocks improves the robustness of our network in dealing with the misalignments existent in the ZRR dataset. Our work can shed some light on the application of wavelet transform in the image ISP problem.

As for future work, our network is able to tackle other low-level imaging tasks, such as image denoising and super-resolution. The improvement of our work can be done in 2 aspects. First, we can shrink the model size by using more efficient convolutional operations such as depth-wise and point-wise convolution (Howard et al. 2017). Meanwhile, as our inference pipeline is done in separated stages, it will be more efficient to revise the pipeline into an end-to-end design. We leave these topics for future research.

Bibliography

Abdelhamed, A., Afifi, M., Timofte, R., and Brown, M. S. (2020). Ntire 2020 challenge on real image denoising: Dataset, methods and results. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 496–497.

Anwar, S. and Barnes, N. (2019). Real image denoising with feature attention. In: *Proceedings of the IEEE International Conference on Computer Vision*, 3155–3164.

Bruhn, A., Weickert, J., and Schnörr, C. (2005). Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. *International Journal of Computer Vision* 61(3), 211–231.

Cao, Y., Xu, J., Lin, S., Wei, F., and Hu, H. (2019). Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 0–0.

Carreira, J. and Zisserman, A. (2017). Quo vadis, action recognition? a new model and the kinetics dataset. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6299–6308.

Chen, C., Chen, Q., Xu, J., and Koltun, V. (2018a). Learning to see in the dark. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3291–3300.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018b). Encoder-decoder with atrous separable convolution for semantic image segmentation. In: *Proceedings of the IEEE European Conference on Computer Vision*, 801–818.

Cheng, D., Price, B., Cohen, S., and Brown, M. S. (2015). Beyond white: Ground truth colors for color constancy correction. In: *Proceedings of the IEEE International Conference on Computer Vision*, 298–306.

Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. (2007). Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing* 16(8), 2080–2095.

Feichtenhofer, C., Pinz, A., and Zisserman, A. (2016). Convolutional two-stream network fusion for video action recognition. In: *Proceedings of the IEEE European Conference on Computer Vision*, 1933–1941.

Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., and Lu, H. (2019). Dual attention network for scene segmentation. In: *Conference on Computer Vision and Pattern Recognition*, 3146–3154.

Gao, X. and Xiong, H. (2016). A hybrid wavelet convolution network with sparse-coding for image super-resolution. In: *2016 IEEE International Conference on Image Processing (ICIP)*, 1439–1443.

Gharbi, M., Chaurasia, G., Paris, S., and Durand, F. (2016). Deep joint demosaicking and denoising. *ACM Transactions on Graphics (TOG)* 35(6), 1–12.

Guo, T., Mousavi, H. S., Vu, T. H., and Monga, V. (2017). Deep Wavelet Prediction for Image Super-Resolution. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1100–1109.

He, B., Wang, C., Shi, B., and Duan, L.-Y. (2019). Mop moire patterns using mopnet. In: *Proceedings of the IEEE International Conference on Computer Vision*, 2424–2432.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861.*

Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7132–7141.

Huang, H., He, R., Sun, Z., and Tan, T. (2017). Wavelet-SRNet: A Wavelet-Based CNN for Multi-scale Face Super Resolution. In: *2017 IEEE International Conference on Computer Vision (ICCV)*, 1698–1706.

Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., and Liu, W. (2019). Ccnet: Criss-cross attention for semantic segmentation. In: *International Conference on Computer Vision*, 603–612.

Ignatov, A., Timofte, R., et al. (2020). AIM 2020 Challenge on Learned Image Signal Processing Pipeline. In: *European Conference on Computer Vision Workshops.*

Ignatov, A., Van Gool, L., and Timofte, R. (2020). Replacing mobile camera isp with a single deep learning model. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 536–537.

Ji, H. and Fermüller, C. (2008). Robust wavelet-based super-resolution reconstruction: theory and algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31(4), 649–660.

Kim, J., Kwon Lee, J., and Mu Lee, K. (2016). Accurate image super-resolution using very deep convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1646–1654.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kwok, N. M., Shi, H., Ha, Q. P., Fang, G., Chen, S., and Jia, X. (2013). Simultaneous image color correction and enhancement using particle swarm optimization. *Engineering Applications of Artificial Intelligence* 26(10), 2356–2371.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4681–4690.

Liu, P., Zhang, H., Lian, W., and Zuo, W. (2019a). Multi-level wavelet convolutional neural networks. *IEEE Access* 7, 74973–74985.

Liu, X., Ma, Y., Shi, Z., and Chen, J. (2019b). Griddehazenet: Attention-based multi-scale network for image dehazing. In: *Proceedings of the IEEE International Conference on Computer Vision*, 7314–7323.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60(2), 91–110.

Lugmayr, A., Danelljan, M., and Timofte, R. (2020). Ntire 2020 challenge on real-world image super-resolution: Methods and results. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 494–495.

Luo, X., Zhang, J., Hong, M., Qu, Y., Xie, Y., and Li, C. (June 2020). Deep Wavelet Network With Domain Adaptation for Single Image Demoireing. In:

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.

Mallat, S. (1996). Wavelets for a vision. *Proceedings of the IEEE* 84(4), 604–614.

Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11(7), 674–693.

Mei, K., Li, J., Zhang, J., Wu, H., Li, J., and Huang, R. (2019). Higher-resolution network for image demosaicing and enhancing. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops.* IEEE, 3441–3448.

Naik, S. and Patel, N. (2013). Single image super resolution in spatial and wavelet domain. *arXiv preprint arXiv:1309.2057.*

Nguyen, N. and Milanfar, P. (2000). A wavelet-based interpolation-restoration method for superresolution (wavelet superresolution). *Circuits, Systems and Signal Processing* 19, 321–338.

Qian, R., Tan, R. T., Yang, W., Su, J., and Liu, J. (2018). Attentive generative adversarial network for raindrop removal from a single image. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2482–2491.

Rana, A., Singh, P., Valenzise, G., Dufaux, F., Komodakis, N., and Smolic, A. (2019). Deep tone mapping operator for high dynamic range images. *IEEE Transactions on Image Processing* 29, 1285–1298.

Ratnasingam, S. (2019). Deep camera: A fully convolutional neural network for image signal processing. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops.*

Rizzi, A., Gatta, C., and Marini, D. (2003). A new algorithm for unsupervised global and local color correction. *Pattern Recognition Letters* 24(11), 1663–1677.

Schwartz, E., Giryes, R., and Bronstein, A. M. (2018). DeepISP: Toward learning an end-to-end image processing pipeline. *IEEE Transactions on Image Processing* 28(2), 912–923.

Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., and Wang, Z. (2016). Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1874–1883.

Tao, X., Gao, H., Shen, X., Wang, J., and Jia, J. (2018). Scale-recurrent network for deep image deblurring. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8174–8182.

Timofte, R., Rothe, R., and Van Gool, L. (2016). Seven ways to improve example-based single image super resolution. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1865–1873.

Uhm, K.-H., Kim, S.-W., Ji, S.-W., Cho, S.-J., Hong, J.-P., and Ko, S.-J. (2019). W-Net: Two-stage U-Net with misaligned data for raw-to-RGB mapping. In: *Proceedings of the IEEE International Conference on Computer Vision Workshop*. IEEE, 3636–3642.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In: *Conference on Neural Information Processing Systems*, 5998–6008.

Vedaldi, A. and Fulkerson, B. (2010). VLFeat: An open and portable library of computer vision algorithms. In: *Proceedings of the 18th ACM International Conference on Multimedia*, 1469–1472.

Wang, X., Girshick, R., Gupta, A., and He, K. (2018). Non-local neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7794–7803.

Wang, X., Chan, K. C., Yu, K., Dong, C., and Change Loy, C. (2019). Edvr: Video restoration with enhanced deformable convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

Wang, Z., Simoncelli, E. P., and Bovik, A. C. (2003). Multiscale structural similarity for image quality assessment. In: *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*. Vol. 2. Ieee, 1398–1402.

Xu, X., Ma, Y., and Sun, W. (2019). Towards real scene super-resolution with raw images. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1723–1731.

Yuan, L. and Sun, J. (2012). Automatic exposure correction of consumer photographs. In: *Proceedings of the IEEE European Conference on Computer Vision*. Springer, 771–785.

Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L. (2017). Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing* 26(7), 3142–3155.

Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., and Fu, Y. (2018a). Image super-resolution using very deep residual channel attention networks. In: *Proceedings of the IEEE European Conference on Computer Vision*, 286–301.

Zhang, Y., Tian, Y., Kong, Y., Zhong, B., and Fu, Y. (2018b). Residual dense network for image super-resolution. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2472–2481.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *Proceedings of the IEEE International Conference on Computer Vision*, 2223–2232.