2020

Video and Image Super-Resolution via Deep Learning with Attention Mechanism

Xuan Xu
West Virginia University, xuxu@mix.wvu.edu

Video and Image Super-Resolution via Deep Learning with Attention Mechanism

Xuan Xu

Dissertation submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources
at West Virginia University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Engineering

Xin Li, Ph.D., Chair
Mark Tseytlin, Ph.D.
Yuxin Liu, Ph.D.
Natalia A. Schmid, Ph.D.
Guodong Guo, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2020

Keywords: Deep learning, Image processing, Attention mechanism, Joint image demosaicing and super-resolution, Image super-resolution, Video super-resolution

Abstract

Video and Image Super-Resolution via Deep Learning with Attention Mechanism

Xuan Xu

Image demosaicing, image super-resolution and video super-resolution are three important tasks in color imaging pipeline. Demosaicing deals with the recovery of missing color information and generation of full-resolution color images from so-called Color filter Array (CFA) such as Bayer pattern. Image super-resolution aims at increasing the spatial resolution and enhance important structures (e.g., edges and textures) in super-resolved images. Both spatial and temporal dependency are important to the task of video super-resolution, which has received increasingly more attention in recent years. Traditional solutions to these three low-level vision tasks lack generalization capability especially for real-world data. Recently, deep learning methods have achieved great success in vision problems including image demosaicing and image/video super-resolution.

Conceptually similar to adaptation in model-based approaches, attention has received increasing more usage in deep learning recently. As a tool to reallocate limited computational resources based on the importance of informative components, attention mechanism which includes channel attention, spatial attention, non-local attention, etc. has found successful applications in both high-level and low-level vision tasks. However, to the best of our knowledge, 1) most approaches independently studied super-resolution and demosaicing; little is known about the potential benefit of formulating a joint demosaicing and super-resolution (JDSR) problem; 2) attention mechanism has not been studied for spectral channels of color images in the open literature; 3) current approaches for video super-resolution implement deformable convolution based frame alignment methods and naive spatial attention mechanism. How to exploit attention mechanism in *spectral* and *temporal* domains sets up the stage for the research in this dissertation.

In this dissertation, we conduct a systematic study about those two issues and make the following contributions: 1) we propose a spatial color attention network (SCAN) designed to jointly exploit the spatial and spectral dependency within color images for single image super-resolution (SISR) problem. We present a spatial color attention module that calibrates important color information for individual color components from output feature maps of residual groups. Experimental results have shown that SCAN has achieved superior performance in terms of both subjective and objective qualities on the NTIRE2019 dataset; 2) we propose two competing end-to-end joint optimization solutions to the JDSR problem: Densely-Connected Squeeze-and-Excitation Residual Network (DSERN) vs. Residual-Dense Squeeze-and-Excitation Network (RDSEN). Experimental results have shown that an enhanced design RDSEN can significantly improve both subjective and objective performance over DSERN; 3) we propose a novel deep learning based framework, Deformable Kernel Spatial Attention Network (DKSAN) to super-resolve videos with a scale factor as large as 16 (the extreme SR situation). Thanks to newly designed Deformable Kernel Convolution Alignment (DKC_Align) and Deformable Kernel Spatial Attention (DKSA) modules, DKSAN can get both better subjective and objective results when compared with the existing state-of-the-art approach enhanced deformable convolutional network (EDVR).

Acknowledgments

First, I would like to express my deepest appreciation to my committee chair and advisor Prof. Xin Li for giving me invaluable guidance and advices throughout my Ph.D. study in the past several years. His professionalism, expertise, and patience help me build up self-confidence and motivate me to find new insights on research. This dissertation would not be possible without his enormous effort and generous help. His rigorous and enthusiastic research attitude have always deeply guided and inspired me in my whole Ph.D. study. And I believe these experiences in West Virginia University will benefit me a lot in my future life and career.

I would also like to express my sincerest thanks to my committee members, Prof. Mark Tseytlin, Prof. Guodong Guo, Prof. Yuxin Liu and Prof. Natalia Schmid for giving me valuable suggestions for my Ph.D. study and helping me to improve this dissertation. I want to have my special thanks to Prof. Mark Tseytlin for his one-year support for my Ph.D. study.

I would like to extend my gratitude to Dr. Chiman Kwan for supplying real-world Bayer patterns collected by NASA Mars Curiosity. Also, I am thankful to all my lab mates, colleagues, faculty and staff members in Lane Department of Computer Science and Electrical Engineering, and all my friends, for their generous help and support during my Ph.D. study.

Finally, I would like to thank my family, especially my parents and my sister, who unconditionally love, support and understand me.

Contents

List of Figures

List of Tables

List of Abbreviations

SISR Single Image Super-Resolution

JDSR Joint Demosaicing and Super-Resolution

RCAN Residual Channel Attention Networks

CNN Convolutional Neural Networks

HR High Resolution

LR Low Resolution

PSNR Peak Signal-to-Noise Ratio

SSIM Structural Similarity Index

RDN Residual-Dense Networks

GAN Generative Adversarial Networks

RaGAN Relativistic average GAN

SR Super-Resolution

JDD Joint Demosaicing and Denoising

SCAN Spatial Color Attention Networks

SCAM Spatial Color Attention Module

DSERN Densely-Connected Squeeze-and-Excitation Residual Network

DSERB Densely-Connected Squeeze-and-Excitation Residual Block

RCSA Residual Channel-Spatial Attention

VSR Video Super-Resolution

MVSR Multi-Frame Video Super-Resolution

RDSEN Residual-Dense Squeeze-and-Excitation Networks

IRI Iterative-Residual Interpolation

RDSEB Residual-Dense Squeeze-and-Excitation Block

DSE Dense Squeeze-and-Excitation

DKSAN Deformable Kernel Spatial Attention Networks

GCN Graph Convolutional Networks

DKSA Deformable Kernel Spatial Attention

JSTSR Joint Video Spatial and Temporal Super-Resolution

RNN Recurrent Neural Networks

Chapter 1

Introduction

Low-level vision problems such as image super-resolution, image demosaicing and video super-resolution have been explored in the last decade. The traditional approaches usually based on interpolation and modeling. Recently, thanks to the advance in more powerful computing resources, deep learning techniques are rapidly developing. Convolutional Neural Networks (CNN) play more and more important role to solve most of the computer vision tasks.

1.1 Motivation

1.1.1 Single Image Super-Resolution

Image super-resolution aims at increasing the spatial resolution of low-resolution LR images and enhancing important structures (usually the high-frequency components such as edges and textures) which is regarded as a challenging ill-posed problem (as shown in Fig. 1.1). It has been widely applied to practical and real-world applications such as medical imaging [10], surveillance and security [11].

Attention mechanism, that allows a network to concentrate its computational resources on the most useful features and enhance the discriminative learning ability, originally inspired by the

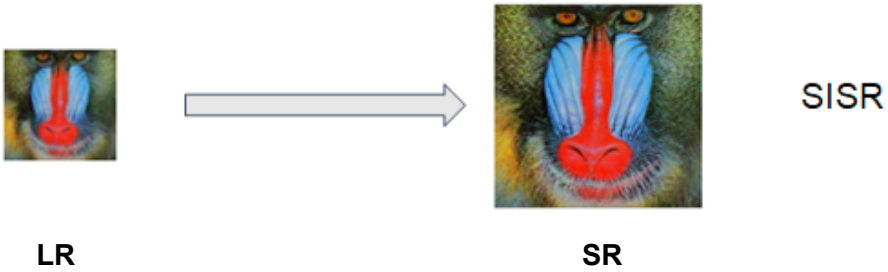

Figure 1.1: The general procedure of single image super-resolution.

behavior and the neuronal architecture of primate visual systems [12,13], has received increasingly more attention by computer vision and machine learning communities. Since the breakthrough in machine translation application [14], attention has been found to be useful to many high-level vision tasks including image captioning [15, 16], lip reading [17], image classification [18–20] and image understanding [21, 22]. The success of attention mechanism is generally attributed to prioritize the allocation of available processing resources towards the most informative components (e.g., salient regions) in an image.

Until now, attention mechanism has been under-researched for low-level vision tasks. The only few exceptions all deal with SISR (e.g., channel attention [2], channel-spatial attention [23], non-local attention [24]). The common theme behind so-called spatial or channel attention mechanism is to adaptively rescale each spatial-domain or channel-wise feature by modeling their interdependency, that will help networks pay more attention to specific features.

However, existing study about attention mechanism has not been extended for color images or across spectral bands to the best of our knowledge. The only studies about color attention we can find are [25, 26] which have focused on the application of object recognition for high-level vision tasks. The issue of how to jointly exploit spatial and spectral dependencies [27] for low-level vision tasks such as SISR seems to have not been addressed in the open literature. All previous attention strategies for SISR have only considered to directly use R,G,B color channels as input training data. In other words, the networks will simply treat all the color information among R,G,B channels equally. One potential risk of this strategy is the lack of optimization - e.g., exploiting the spectral dependency among color channels might benefit the task of deep residual learning.

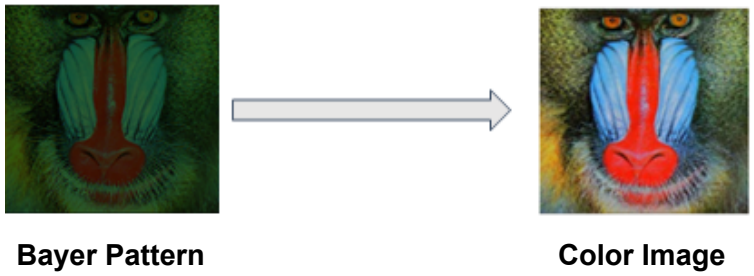

Figure 1.2: The general procedure of single image demosaicing.

Figure 1.3: Comparison of JDSR output to separately demosaic-super-resolve output. Left to right: a) HR image (ground-truth); b) 4× upscaling output by concatenating state-of-the-art demosaicing method Flex [1] with SISR method RCAN [2] (separated approach); c) 4× upscaling output of our proposed DSERN networks (joint approach).

1.1.2 Joint Image Demosaicing and Super-Resolution

Image demosaicing and single image super-resolution (SISR) are two important image processing tasks in the pipeline of color imaging. Demosaicing is a necessary step to reconstruct full-resolution color images from so-called Color filter Array (CFA) such as Bayer pattern(as shown in Fig. 1.2). SISR is a cost-effective alternative to more expensive hardware-based solution (i.e., optical zoom). Both problems have been extensively yet separately studied in the literature - from model-based methods [28–36] to learning-based approaches [2, 4, 5, 37–42]. Treating demosaicing and SISR as two independent problems may generate undesirable edge blurring as shown in Fig. 1.3. Moreover, the processes of demosaicing and SISR can be integrated and optimized together from a practical application point of view (e.g., digital zoom for smartphone cameras such as Google Pixel 3 and Huawei P30).

Inspired by the success of joint demosaicing and denoising [6], we propose to study the problem of JDSR in this dissertation and develop a principled solution leveraging the latest advances in deep learning to computational imaging. We argue that the newly formulated JDSR problem has high practical impact (e.g., to support the mission of NASA Mars Curiosity and smartphone applications). The problem of JDSR is intellectually appealing but has been under-researched so far. The only existing work we can find in the open literature is a recently published paper [43] which contained a straightforward application of ResNet [44] and considered the scaling ratio of two only.

1.1.3 Video Super-Resolution

Video Super-Resolution (VSR) aims to map LR videos to high-resolution HR videos with more details especially edges and textures (as shown in Fig. 1.4). The same as image super-resolution, VSR has also widely used in many practical applications such as surveillance [45] and HDTV. More than that, VSR has great potential to be applied to video coding and video compression. Video super-resolution can be mainly separated as two problems, video spatial super-resolution and video temporal super-resolution. Video spatial super-resolution regards to super-resolve the spatial resolution of the LR frame to HR frame to improve clarity of video; while video temporal super-resolution refers to the generation of new frames between two neighboring frames to increase the video frame rate and fluency. Different from SISR which only needs to consider the information from spatial domain, both spatial and temporal information can be utilized to recover HR videos in the task of VSR via multiple LR frames. Although single frame based video super-resolution is a possible approach for VSR, ton of temporal information such as motion compensation among the consecutive frames cannot be exploited.

In order to explore the potential benefit from temporal information of VSR, many existing approaches [46–49] conducted a sequence of consecutive LR frames including one reference frame and several neighboring frames as inputs to reconstruct HR frame corresponding to reference LR frame. One problem of this kind of methods is the camera or objects motion among different frames. Therefore, to better exploit temporal information, the consecutive frames need to be aligned before using to reconstruct HR frame. As the most famous motion estimation method,

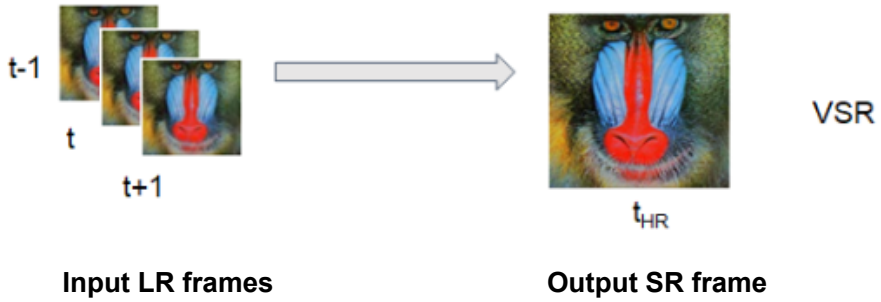

Figure 1.4: The general procedure of multi-frame video super-resolution.

optical-flow [50] is logically considered as motion estimator for several VSR approaches [51–53]. However, the disadvantages of the optical flow method are also obvious. For example, the flows can be observed when the external illumination changes even if there is no movement. In addition, the actual motion is often not observed in areas without sufficient variation of gray level. Conducting with wrong motion compensation may generate undesired blurring and artifacts for HR frames.

Recently, deformable convolution [54, 55] becomes more and more popular as an assistant module for video frame alignment, such as [56–58] have already successfully applied for vary forms of deformable convolution alignment module to temporally align neighboring frames with reference frame and get the desired motion compensation compared with optical-flow-based methods. However, such the alignment modules still learn the offsets via several normal convolution layers which may not extract the valuable information because of the fixed kernel configurations. Alternatively, deformable kernels [59] can adapt effective receptive fields by weighting the contribution of each pixel that can enhance the offset extraction compared with the normal convolution does.

1.2 Problem Statement

In this dissertation, we mainly focus on solving three major challenging problems: 1) SISR; 2) JDSR; 3) Multi-Frame Video Super-Resolution (MVSR).

1.2.1 Single Image Super-Resolution

In this work, we propose to address the above issue (Sec. 1.1.1) by developing a new architecture named Spatial Color Attention Networks (SCAN). Conceptually similar to bilateral filtering [60] in which spatial and color are treated as two independent domains, we treat spatial and color features as two complementary channels. So instead of considering channel-wise and spatial feature modulation in [23], we have developed a spatial color attention module (SCAM) to calibrate important color information from output feature maps of residual groups. Unlike existing works which treat channel-wise features across spectral bands equally, we propose to make the networks focus on informative features and exploit interdependencies among color channels. The newly developed color attention mechanism enables the network to not only focus on recovering spatially high frequency components (e.g., edges and textures) but also pay attention to vivid and sharp color information (e.g., colorful flowers and texts) in the generated HR image.

1.2.2 Joint Image Demosaicing and Super-resolution

In the second work, the motivation behind our approach is mainly two-fold. On one hand, rapid advances in deep residual learning have offered a rich set of tools for image demosaicing and SISR. For example, DenseNet [61] has been adapted to fully exploit hierarchical features for the problem of SR in SRDenseNet [62] and residual dense network (RDN) [42]; residual channel attention network (RCAN) [2] allows us to develop much deeper networks (over 400 layers) with squeeze-and-excitation (SE) blocks [19] than previous works (e.g., [4,63]). However, to the best of our knowledge, the issue of *spatio-spectral attention* mechanism has not been explicitly addressed for color images in the open literature. How to *jointly* exploit spatial and spectral dependency for JDSR in network design deserves a systematic study.

On the other hand, we propose to optimize the *perceptual quality* for JDSR because that is what really matters in real-world applications. Generative adversarial network [64] is arguably the most popular approach toward perceptual optimization and has demonstrated convincing improvement for SISR in SRGAN [5]. However, it has also been widely observed that the training of GAN suffers from stability issue which could have catastrophic impact on reconstructed images. There has been a flurry of the latest works (e.g., Relativistic average GAN (RaGAN) [65], enhanced SRGAN

(ESRGAN) [41] and perception-enhances SR (PESR) [66]) showing the potential of relativistic discriminator in stabilizing GAN and improving visual quality of SISR images. How to leverage those latest advances to optimize the perceptual quality for JDSR has practical significance.

1.2.3 Multi-Frame Video Super-Resolution

In the final work, we propose a novel Multi-Frame based Deformable Kernel Spatial Attention Network (DKSAN) for video extreme super-resolution, the upscaling factor is 16. Different from the traditional motion estimation based on optical flow [50], inspired by EDVR [56] who applies deformable convolution [55] to temporally align neighboring frames with reference frame, we designed a non-optical-flow based module, called Deformable Kernel Convolution Alignment (DKC_Align) module, to utilize deformable kernel [59] combined with deformable convolution to extract not only global but also local edge and texture features to align neighboring frames with reference frame. Moreover, we developed Deformable Kernel Spatial Attention (DKSA) module to further enhance the spatial details of reconstructed feature maps. The advance of DKSA module is that deformable kernel [59] can better represent the edge and texture features which are important for super-resolution tasks compared with general convolution based spatial attention which cannot represent local feature accurately.

1.3 Contributions

1. SISR

   • We propose to address the issue of *color attention* for SISR and demonstrate it is supplementary to the spatial and channel attention mechanisms studied in the literature;

   • Our proposed Spatial Color Attention Module (SCAM) and Residual Channel-Spatial Attention (RCSA) can be easily integrated to most existing SISR networks;

   • Experimental results have shown our SCAN can significantly outperform previous state-of-the-art RCAN [2] on real SISR competition dataset.

2. JDSR

• Network design: we propose two networks for JDSR, Densely-Connected Squeeze-and-Excitation Residual Network (DSERN) and Residual-Dense Squeeze-and-Excitation Networks (RDSEN). DSERN is novel Densely-Connection Squeeze-and-Excitation Residual Block (D_SERB) is designed to facilitate information flow in deeper and wider networks by smooth activation, which can more effectively suppress spatio-spectral aliasing. RDSEN concatenates a pre-demosaicing network (PDNet) and Residual-Dense Squeeze-and-Excitation Networks (RDSEN). The former takes a model-based demosaicing result via Iterative-Residual Interpolation (IRI) [67] as the surrogate target to facilitate deep residual learning for pre-demosaicing. Then a novel concatenation of Residual-Dense Squeeze-and-Excitation Block (RDSEB) modules is designed to facilitate information flow between the intermediate demosaicing result and the final reconstruction.

• Perceptual optimization: we have leveraged the latest advance RaGAN [65] from SISR to JDSR and studied the choices of perceptual loss function for JDSR. In addition to improved stability, we have found that Texture-enhanced RaGAN (TRaGAN) with a before-activation perceptual loss function can produce visually more pleasant results.

• Real-world application: we have applied the proposed RDSEN+TRaGAN solution to raw Bayer pattern data collected by the Mast Camera (Mastcam) of NASA Mars Curiosity Rover. Our experimental results have shown visually superior HR image reconstruction can be achieved at the scaling ratio as large as 4.

3. MVSR

• We propose a multi-frame based cascaded network called Deformable Kernel Spatial Attention Networks (DKSAN) to solve the video extreme super-resolution task (scale factor of 16).

• We designed a new deformable kernel [59] + deformable convolution [54,55] based module to align the neighboring frames with the reference frame.

• Experimental results have shown our DKSAN can significantly outperform previous state-of-the-art EDVR [56] on IntVID [68] dataset.

1.4 Dissertation Structure

The rest of this dissertation is organized as follows:

- Chapter 2 reviews the state-of-the-art SISR approaches and introduces a novel network Spatial Color Attention Networks (SCAN) to solve real world image super-resolution problem.
- Chapter 3 presents the JDSR problem and proposes two effective solutions via deep learning architecture.
- Chapter 4 describes the proposed DKSAN network with DKSA and DKC_Align modules to solve video extreme super-resolution problem with a scale factor of 16.
- Chapter 5 presents the conclusions of this dissertation and plans the future works.

Chapter 2

Single Image Super-Resolution

2.1 Related Work

2.1.1 Traditional Approaches

Traditional approaches towards SISR [32–36] suffer from notorious aliasing artifacts and edge blurring. Here we reviewed a couple of state-of-the-art SISR approaches.

Chang *et al.* [33] proposed a patch-learning-based framework to utilize similar local geometry from LR patches to reconstruct HR images. Yang *et al.* [35] introduced a novel sparse coding based image super-resolution method which trained two dictionaries for LR and HR image patches jointly to enforce the similarity between LR dictionary and HR dictionary based on sparse representations. Therefore, the HR dictionary can guide LR patch image to reconstruct Super-Resolution (SR) image through sparse representations. Bevilacqua *et al.* [32] presented a non-negative neighbor embedding method which is also based on learning LR and HR dictionary. In particular, a dictionary can be trained by the feature vector of LR patches which are represented via a weighted K nearest neighbors combination; then, this trained LR dictionary can guide input LR patches to reconstruct the corresponding HR patches. Timofte *et al.* [34] proposed fast SR method to speed-up learning dictionary based approaches without any compromise on being reconstructed super-resolved images.

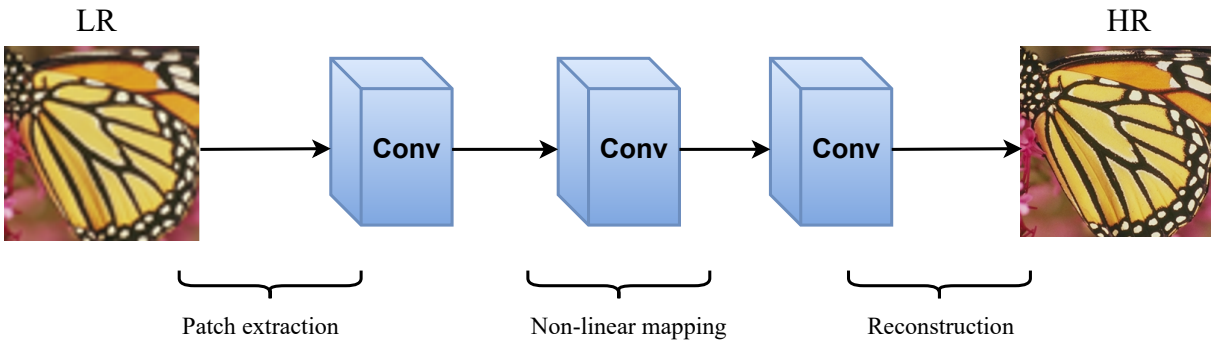

Figure 2.1: The overview of SRCNN [3] networks.

2.1.2 Deep Learning Approaches

Pre-upsampling SISR networks

Deep learning-based approaches toward single image super-resolution (SISR) have shown the reliability and advantages compared with the traditional model-based methods. As a pioneer, SRCNN [3] first introduced CNN architecture to train SISR model with pre-upsampling LR images because the mapping between low and high-dimensional space is difficult to learn directly. By applying traditional interpolation algorithms such as Bicubic, SRCNN can be end-to-end trained from LR (after upsampling) to HR image (as shown in Fig. 2.1). This training strategy significantly reduces the learning difficulty which redefined the SR problem as an image coarse-to-fine process. Although SRCNN firstly implements CNNs to SISR problem, it's still only a three layer network which limit to explore deeper features to reconstruct more detailed information. To address vanishing gradient problem when training deeper networks, VDSR [4] (shown in Fig. 2.2) utilized the concept of deep residual networks [44] to build a 20-layer network to learn only the residual between interpolated LR image and HR image, and significantly improved the results.

Post-upsamling SISR networks

Although pre-upsampling and long skip connection strategies overcome the difficulty of training, large size of input images (pre-upscaled LR images) consume a huge amount of GPU memory, and slow down the training and test running time. Also image interpolation algorithms such as Bicubic always amplifies noise and blurring.

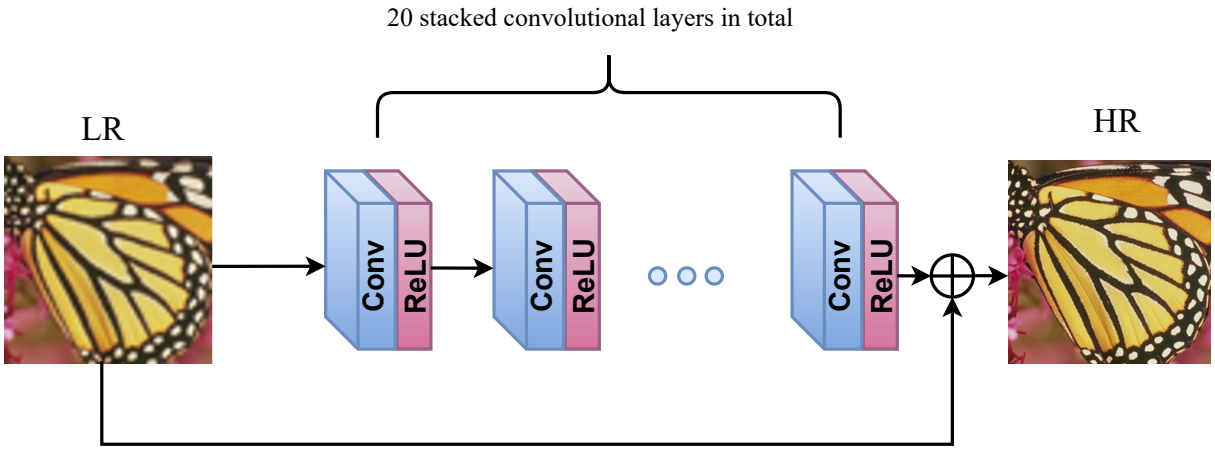

Figure 2.2: The overview of VDSR [4] networks, $\oplus$ denotes element-wise sum. The long connection between HR and LR images which can significantly reduce training complexity.

To avoid these issues and improve the computational efficiency, LapSRN [69] proposed to upscale low resolution image by a pyramid structure, this cascade of deep learning method not only saved GPU memory but also reduced the gap among LR image and each corresponding scale of HR image which has achieved a better performance on large scale factors (ex. $8\times$). EDSR [63] and its derivative network MDSR firstly introduced to stack residual blocks to address SISR problem. Residual blocks utilized short skip connection to make information flow more smoothness. Also, batch-normalization [70] layer which usually applied to reduce training difficulty and boost convergence is discarded. There are two reasons: 1) batch-normalization is designed for high-level vision problems such as classification, low-level vision problem needs the accuracy for each pixel position which normalized layer may prevent the network to find the optimized solution; 2) removing batch-normalization layer can save up to 40% GPU memory to extend a deeper and wider networks and thus get a chance to boost the performance substantially. Most recent advances include SRDenseNet [62] which applied DenseNet [61] to reach a wider network to increase the performance. Residual-Dense Networks (RDN) [42], which took the advantages of ResNet [44] and DenseNet [61], proposed the stacked residual dense block (RDB) and through local and global feature fusion to utilize deep and wide networks information to reconstruct HR images.

Attention Mechanism based approaches

It is worth highlighting previous works on attention mechanism in the existing literature. Generally speaking, the common principle underlying various attention mechanisms is to bias limited computational resources based on the importance of informative components. For example, channel attention [19] adaptively rescale the channel-wise feature by modeling their interdependency; channel-spatial attention addresses the issue of channel-wise and spatial feature modulation [23]; non-local attention [24] attempts to simultaneously exploit the local and non-local dependency within an image for the task of image restoration. Residual Channel Attention Networks (RCAN) [2] first introduced attention mechanism inspired by SENet [19] to calibrate feature maps and proposed residual in residual structure to achieve a very deep convolutional networks which achieved new state-of-the-art performance for SISR task.

To the best of our knowledge, the issue of color attention - i.e., the modeling of interdependency across different spectral channels - had not been studied in the open literature. Therefore, we propose to address this issue and develop specially tailored modules for color image restoration.

2.1.3 SISR with Generative Adversarial Network

Generative Adversarial Network (GAN), proposed by Goodfellow *et al.* [64], gives a new insight for deep learning techniques. GAN based model usually includes two parts, generator and discriminator. These two models cooperate in an adversarial manner, the generator tries to generate realistic synthetic image to fool discriminator and the discriminator works on distinguishing the ground-truth image and the fake image, until the whole networks find a trade-off point where the generator can synthesize most realistic fake image. GAN has been implemented to solve many computer vision problems, such as image synthesis [71], image-to-image translation [72], denoising [73], dehazing [74] and image super-resolution [5, 41, 66]. Here we only focus on reviewing literature related to image super-resolution with GAN.

Most of existing non-GAN based SISR approaches operate with L1 or L2 (MES) as the loss function which aims to optimize the measure index of Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM). This guides the model to produce a smoother super-resolved image which loses sharp and texture details. Besides objective measures such as PSNR/SSIM [75],

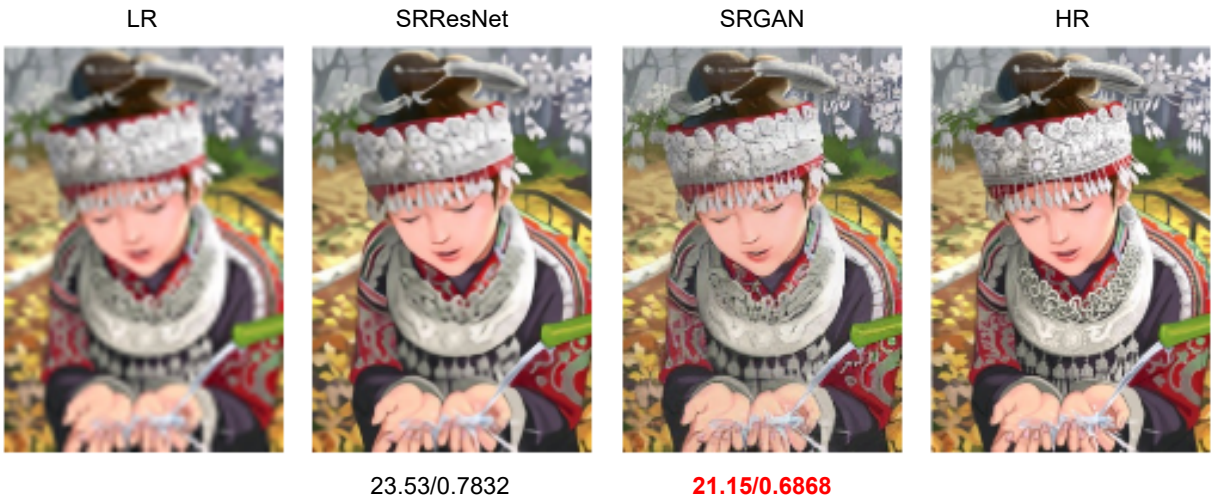

Figure 2.3: The visual results comparison among GAN and non-GAN approaches [5]. The scale-factor is 4.

SRGAN [5] introduced a novel GAN [64] based architecture and combined pre-trained VGGNet [76] to extract high-level similarity features and optimize the perceptual quality of SR images. Benefited by GAN, SRGAN can reconstruct more textures from LR images even when the measure index of PSNR / SSIM is much lower compared with non-GAN approaches. As shown in Fig. 2.3, the SRResNet has a better PSNR performance but SRGAN can generate a more realistic SR image. Because of the loss function design, one issue of SRGAN has is to generate the artificial noise when reconstructs to SR image. To solve this problem, an enhanced version of SRGAN named ESRGAN [41] adopted Relativistic average GAN (RaGAN) which was developed in [65] as well as [66]. The main difference between GAN and RaGAN is the adversarial loss function (feedback to generator and discriminator). Unlike the traditional GAN tried to estimate the probability that the image entered into the discriminator is real, RaGAN estimates the probability that the real image is relatively more realistic than a fake image which is to encourage generator to generate more realistic images.

2.1.4 Summary of SISR approaches

Table. 2.1 presents a summary of the state-of-the-art deep learning methods for the SISR task.

Method	Pre.	Pos.	Upsampling	Res.	CA	SCA	GAN
SRCNN [3]	✓		Bicubic				
VDSR [4]	✓		Bicubic	✓			
FSRCNN [77]		✓	Deconv				
LapSCN [69]		✓	Bicubic	✓			
DRRN [78]	✓		Bicubic	✓			
EDSR [63]		✓	Sub-Pixel	✓			
EnhancedNet [79]	✓		Bicubic	✓			
SRGAN [5]		✓	Sub-Pixel	✓			✓
SRDenseNet [62]		✓	Deconv	✓			
DBPN [80]		✓	Deconv	✓			
RDN [42]		✓	Sub-Pixel	✓			
RCAN [2]		✓	Sub-Pixel	✓	✓		
ESRGAN [41]		✓	Sub-Pixel	✓			✓
PESR [66]		✓	Sub-Pixel	✓			✓
RNAN [24]		✓	Sub-Pixel	✓			
SCAN(ours)		✓	Sub-Pixel	✓	✓	✓	

Table 2.1: A summarry of previous works on SISR. The “Pre.”, “Pos.”, “Res.”, “CA,” and “SCA” respectively represent pre-upsampling, post-upsampling, residual connection, channel attention and spatial color attention.

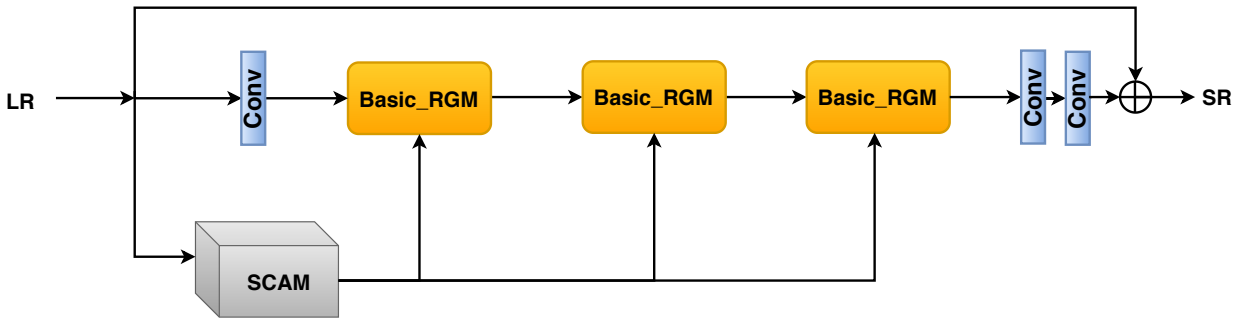

Figure 2.4: Overview of the proposed networks architecture, Basic_RGM stands for the basic residual group module which includes several residual groups, SCAM is the proposed spatial color attention module where to generate R,G,B spatial color attention map (see 2.5 for more details), $\oplus$ denotes element-wise sum.

2.2 Proposed Approach of SISR

2.2.1 Network Design

We present the designed networks in the following hierarchy: SCAN (Fig. 2.4) $\rightarrow$ Subnetwork of SCAM and Basic_RGM (Fig. 2.5) $\rightarrow$ Residual Channel-Spatial Attention (RCSA, Fig. 2.6). It should be noted that our SCAN and previous state-of-the-art RCAN [2] are similar at the coarsest level. Both SCAN and RCAN are decomposed of the residual group (RG) and residual channel attention block (RCAB). This is because we want to evaluate the validity of proposed SCAM (refer to Sec. 2.2.2) and RCSA (refer to Sec. 2.2.3) modules under the same conceptual framework.

We are hoping that this way of presentation can facilitate our explanation about why SCAN can outperform RCAN (similar to the popular ablation study) - i.e., without changing the overall structure, we can still improve the performance of SISR by designing novel fine-scale modules (SCAM and RCSA) in a plug-and-play fashion (e.g., the number of Basic_RGM modules in Fig. 2.4 can be reduced as we will show in our ablation study in Table 2.3). Meanwhile, we will focus on the key difference between the design of RCAN and SCAN - i.e., the desirable color attention mechanism. Across different hierarchies (SCAN$\rightarrow$SCAM$\rightarrow$RCSA), we will show how color attention mechanism is the theme unifying our network design and optimizing the task of deep residual learning.

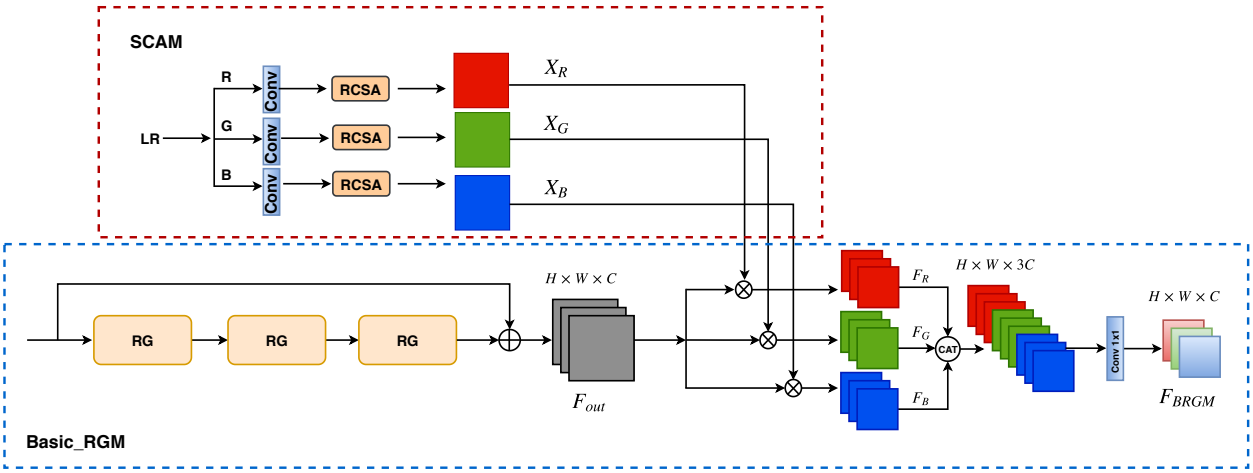

Figure 2.5: The structure of proposed Basic_RGM and SCAM modules. In the block of SCAM, RCSA stands for proposed residual channel-spatial attention module (the details are demonstrated in Fig. 2.6); $\oplus$ denotes element-wise sum, $\otimes$ denotes element-wise product, CAT denotes feature-concatenation.

2.2.2 Spatial Color Attention Module (SCAM)

In SCAM module (see Fig. 2.5), we organize the input training data (LR images) into two parts: 1) similar to the normal SISR architectures, the whole LR image is supplied as the input to the main network (see Basic_RGM block in Fig. 2.5); 2) the LR image is divided to R,G,B channels separately, which then serve as the input to the proposed RCSA module for generating spatial color attention maps X_R, X_G, X_B, for R,G,B channels respectively. Note that the second part (our new contribution) is absent in previous works on SISR because they treat all R,G,B channels equally.

Let F_{out} denote the output feature maps of the Basic_RGM block (see Fig. 2.5, the gray-colored feature map with the dimension of $H \times W$ that contains C feature maps), which is generated from the whole LR input image and fused all R,G,B information into each feature map. To re-calibrate F_{out}, we apply element-wise product between each spatial color attention map X_R, X_G, X_B and F_{out}. The process of introducing color attention can be expressed as follows:

$$F_R = F_{out} \cdot X_R \tag{2.1}$$

$$F_G = F_{out} \cdot X_G \tag{2.2}$$

$$F_B = F_{out} \cdot X_B \tag{2.3}$$

where F_R, F_G, F_B are the re-calibrated feature maps from F_{out} to represent spatial color information for each R,G,B channel (e.g., F_R represents the spatial information from red channel).

Next, to get the final output feature-map F_{BRGM} with the dimension of $H \times W \times C$, we first concatenate R,G,B feature maps and then use a 1×1 Conv layer to reduce the feature-map dimension from $3C$ to C:

$$F_{BRGM} = \mathbf{W}_D([F_R, F_G, F_B]) \tag{2.4}$$

where $\mathbf{W}_D \in \mathbb{R}^{1\times1\times C}$ is a 1×1 Conv layer used for dimensionality reduction.

By applying SCAM to the basic_RGM, the networks can fuse channel attention (already considered in RCAN [2]) and spatial color attention (new module introduced by this work) to better re-calibrate input feature maps based on the pair of training data. Note that the real SISR challenge still belongs to strongly supervised learning; therefore the objective here is the same as the original idea of applying ResNet [4] to SISR (i.e. to learn a more accurate residual representation). The new insight we attempt to bring through this work is that residual representations across spectral channels are not independent, which implies the potential of jointly learning them (as we will elaborate next).

2.2.3 Residual Channel-Spatial Attention (RCSA)

To implement RCSA module (see Fig. 2.6), we have followed the basic structure of SENet [19] and RCAN [2] which sets up a regular residual block including channel attention mechanism. More specifically, we first squeeze input feature maps with global average pooling:

$$Q_C = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_C(i, j) \tag{2.5}$$

where C is the number of feature maps, Q_C is the c-th element of $Q \in \mathbb{R}^C$, $F_C(i, j)$ is the pixel value of the c-th feature at position (i, j) from input feature maps $F_{Re'} \in \mathbb{R}^{H\times W\times C}$. Then we propose to implement a simple gating mechanism as adopted by previous works including SENet [19] and RCAN [2]:

$$SE = \sigma(\mathbf{W}_E(\delta(\mathbf{W}_S(Q)))) \tag{2.6}$$

where σ refers to a sigmoid function, δ denotes the ReLU function, $\mathbf{W}_S \in \mathbb{R}^{1\times1\times\frac{C}{r}}$ is the *squeeze* Conv layers with weights and $\mathbf{W}_E \in \mathbb{R}^{1\times1\times C}$ is the *expand* Conv layers with weights, r is the

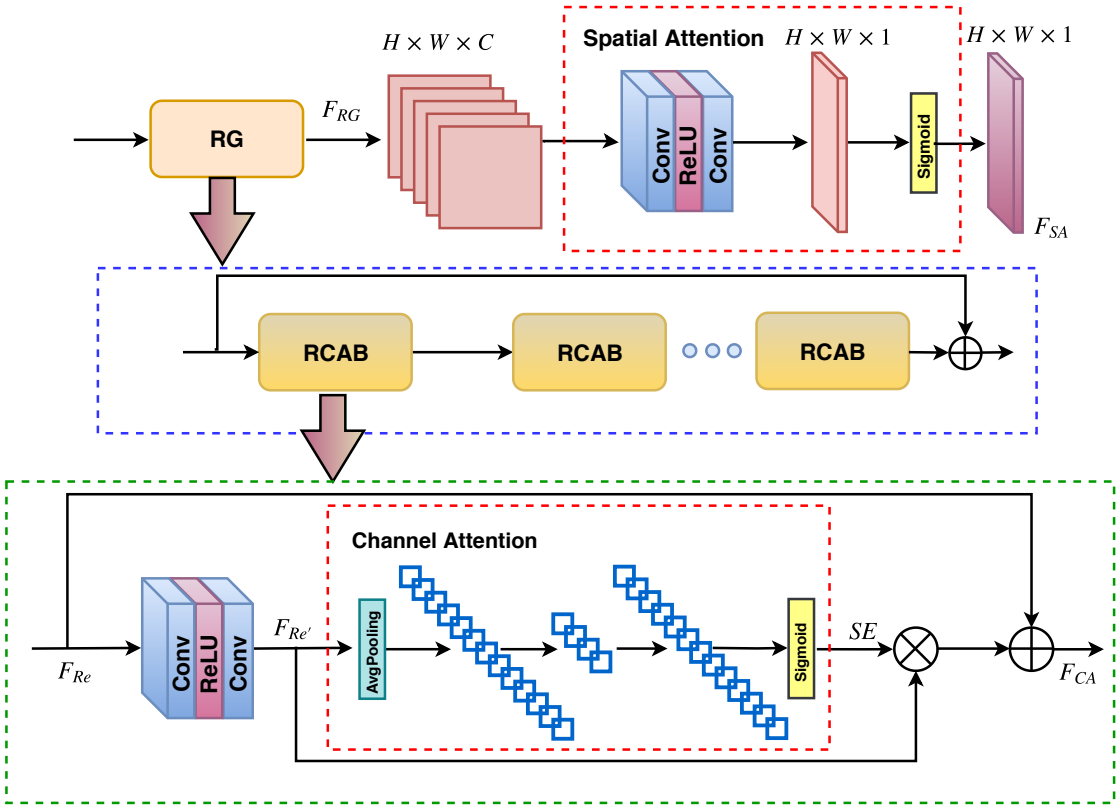

Figure 2.6: The structure of proposed RCSA module which includes the implementation of RG and RCAB blocks. The two red blocks show the structures of channel and spatial attentions. $\otimes$ denotes element-wise product and $\oplus$ denotes element-wise sum.

reduction ratio to reduce the dimension of Q (the parameter r controls the trade-off between the capacity and the complexity [19]). Finally, we can rescale the feature maps F_{Re} by:

$$F_{CA} = SE \cdot F_{Re'} + F_{Re} \tag{2.7}$$

where F_{Re} is the input feature map to RCAB block, F_{CA} is the output from RCAB block which is a rescaled feature maps by channel attention module (see Fig. 2.6). Note that Eq. (2.7) is different from previous works such as RCAN because we will have a separate channel/spatial attention mechanism for each R,G,B channel respectively.

Next, we can apply spatial attention to the rescaled feature map. Unlike previous works in which R,G,B feature maps are treated equally, we note that the input of RCSA is a single channel of RGB image. Therefore, our approach generates the output feature map (so-called spatial color attention map) which focuses on re-calibrating single color channel information from the feature maps of F_{out}. In this fashion, the outputs of SCAM module naturally fit the grey-colored feature

map in Fig. 2.5 (refer to section 2.2.2). More specifically, we have

$$F_{SA} = \sigma(\mathbf{W}_{SA}(\delta(\mathbf{W}_{SA}(F_{RG})) \tag{2.8}$$

where F_{SA} is the output spatial color attention feature map which can be represented as X_R, X_G, X_B based on the corresponding R,G,B channels, $\mathbf{W}_{SA} \in \mathbb{R}^{1\times1\times C}$ is the Conv layer with weight, σ refers to a sigmoid function, δ denotes the ReLU function. F_{RG} is the output of RG (refer to Fig. 2.6). In summary, newly designed spatial color attention map is expected to more effectively learn the joint residual representations across spectral channels.

2.3 NTIRE2019 Real-World Dataset

Most of SISR methods are basically trained with synthetic dataset. Bicubic interpolation is the most widely used downsampling method to syntheses LR images from original HR images. Many state-of-the-art approaches such as EDSR [63], LapSRN [69], RDN [42], RCAN [2] etc. used this way to generate different scale of LR images. One critical problem of this LR data generation is that the mapping between low and HR data is fixed (Bicubic, Bilinear or any interpolation methods). It means if the test data is not acquired by the certain interpolation method which used to generate training dataset, the trained model will not super-resolve desired SR image.

In this SISR work, we have used the real-world paired image dataset provided by NTIRE2019 challenge. It includes 60 pairs of images for training, 20 pairs of images for validation and another 20 pairs of images for testing; both HR and LR images are collected by standard DSLR cameras, which means the LR image is not synthetic but captured from the real-world (likely with a different focal length). This is in sharp contrast with previous SISR challenges which generate LR images from HR images (e.g., DIV2K [81]) using model-based methods (e.g., Bicubic interpolation). The new real-world dataset is arguably more closely related to the real-world SISR tasks - e.g., the scaling factors between LR and HR images are unknown (in theory it is determined by the ratio of focal lengths). The sample images are shown in Fig. 2.7.

We also note that HR images for test data (i.e., the ground-truth) is not provided; therefore LR images in test data have already been scaled to the same size/resolution as the corresponding HR images by the competition organizer. Accordingly, we have opted to report our PSNR/SSIM

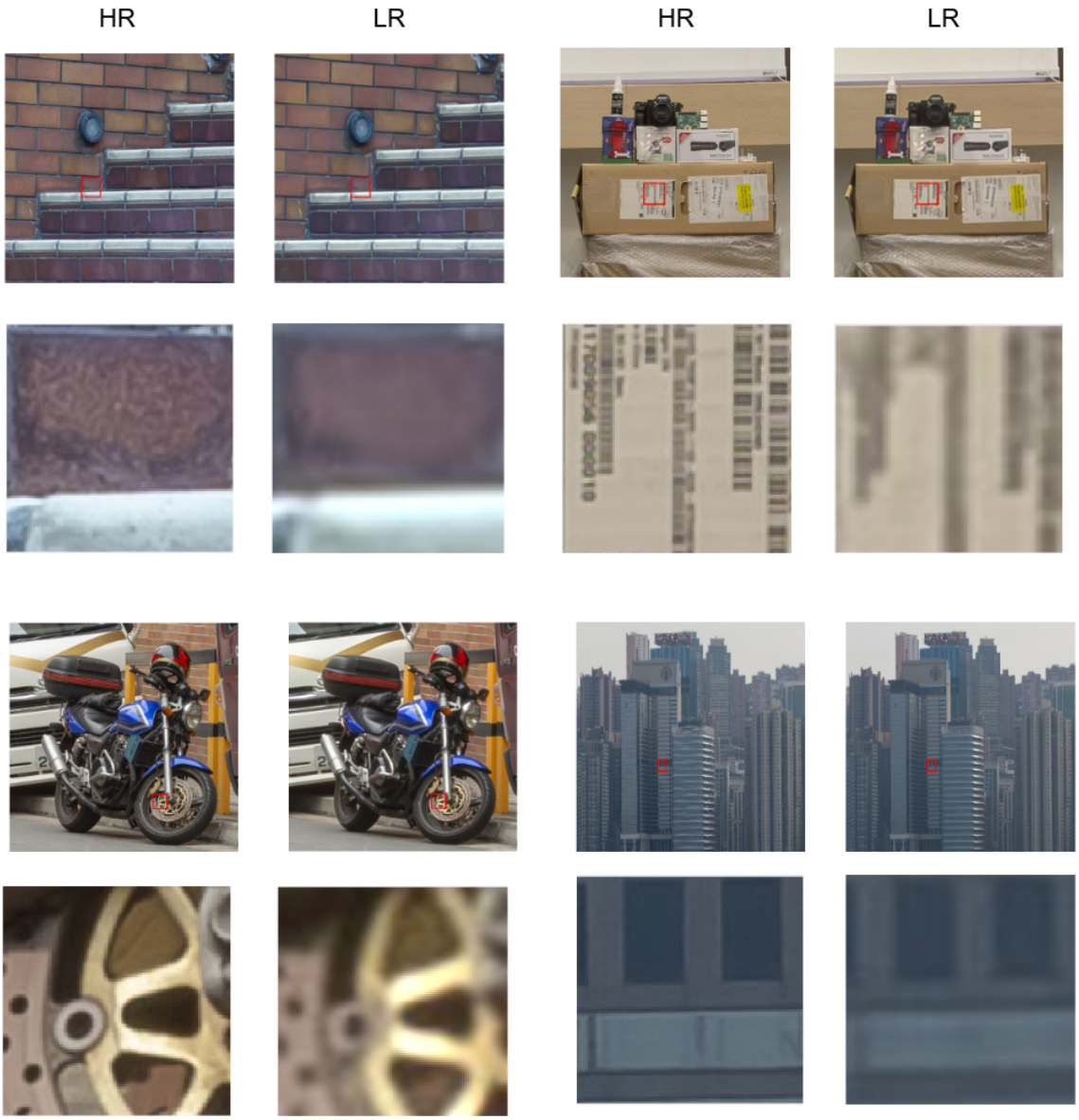

Figure 2.7: The sample data from NTIRE2019 real-world dataset. The downsampling factor of LR images are unknown.

experimental results based on 20 paired validation data (for which ground-truth is available) and report perceptual index (PI) score [82] based on 20 test data since PI is a no-reference image quality metric (no HR image is needed).

2.4 Experimental Results

In this section, we demonstrate the network setting, training details, ablation study and experimental results of proposed SISR problem.

2.4.1 Implementation details

In our proposed SCAN networks, we have set Basic_RGM to 3, each Basic_RGM includes 3 residual groups (RG). And every RG contains 20 RCAB blocks which is the same as the original RCAN [2]. In RCSA module, we have used one RG with 6 RCAB blocks inside. Most of the kernel size of Conv layers are 3×3 with 64 filters ($C = 64$) except few exceptions as shown in Fig. 2.6 (e.g., in the spatial attention block, the two Conv layers have only 1 filter that means $C = 1$, and 1×1 of kernel size). In channel attention block, the reduction ratio is $r = 16$. The last layer filter of the whole network is set to be 3 in order to output super-resolved color images. Note that the original RCAN has 10 RGs; due to the limitation of GPU memory, we have only adopted 9 RGs in our current implementation of SCAN (3 Basic_RGM, totally amount to 9 RGs).

In our training process, we first randomly crop both the input and ground-truth RGB images with small patches such as 128×128, with a batch size of 16; then we augment the training set by standard geometric transformations (e.g., flipping and rotation). Our model is trained and optimized by ADAM [83] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-8}$. The initial learning rate is set to 1×10^{-4}, the decay factor is set to 5, which decreases the learning rate by half after $[384k, 576k, 768k, 883k, 998k]$ steps; the MSE loss function is applied to minimize the error between HR and SR images. All reported experiment results are trained by PyTorch [84] on 4 NVIDIA TITAN Xp GPUs. The total training time is around 35 hours.

Patch size	48×48	96×96	128×128
PSNR	29.37	29.55	29.59

Table 2.2: The influence of different cropped patch-size used (48 × 48, 96 × 96 and 128 × 128) for training process.

2.4.2 Effect of Patch Size for Training

We explored the effect of different patch-size cropped for the model training. Table 2.2 shows the results of 48 × 48, 96 × 96 and 128 × 128 patch-size used. Note that all the training settings are exactly same besides the patch size of training data. From the results we find that large patch size leads a better PSNR performance. However, due to the constraint with limited GPU memory, 128×128 is the largest patch size we can train at this point.

2.4.3 Ablation Study

In order to better illustrate the benefit of spatial color attention map step by step, we have compared different strategies to evaluate the validity of proposed SCAM. We have implemented four competing models in our experiments (trained by the same dataset): 1) baseline RCAN [2]: training without SCAM (all settings follow the original RCAN); 2) SCAN_1: training with only one-time calibration with SCAM (one Basic_RGM with 9 RGs inside); 3) SCAN_2: training with two-times calibration with SCAM (two Basic_RGM, first one has 5 RGs and the second one has 4 RGs); 4) SCAN_3: training with three-times calibration with SCAM (the proposed SCAN, please refer to Fig. 2.4).

Table. 2.3 shows the results of the four strategies mentioned above. Without SCAM, the RCAN can achieve the average PSNR of 29.31 dB; after adding SCAM, SCAN_1 can improve the initial PSNR results to 29.49 dB (**0.18 dB** gained when compared with RCAN); keep increasing calibration time to 2 and 3, we observe that the PSNR results are further improved. Finally, we have achieved the best PSNR result of 29.59 dB with the proposed SCAN (i.e., SCAN_3 in Table 2.3).

Method	No. of RGs	No. of SCAM calibration used	PSNR	SSIM
RCAN	10	N/A	29.31	0.8606
SCAN_1	9	1	29.49	0.8628
SCAN_2	9	2	29.52	0.8641
SCAN_3	9	3	29.59	0.8650

Table 2.3: Investigations of how to set spatial color attention modules (SCAM) .

2.4.4 Comparison Against State-of-the-Art

We have compared our proposed SCAN with current state-of-the-art SISR approach RCAN [2]. The original RCAN is trained to super-resolve LR image by a specific scale factor (i.e., $2\times$, $3\times$), but the LR and HR image in the new real-world dataset from NTIRE2019 have the same size. Therefore, we have to remove the upscale module from the original RCAN to make sure both the input and the output have the same size. The comparison results in terms of PSNR are shown in Table 2.4.

The baseline result is the average PSNR between the (scaled) LR images and the corresponding HR images. The "+" in RCAN+ and SCAN+ stands for self-ensemble strategy used to further improve results (similar strategies have been adopted in previous works [2, 42, 63, 85]). From Table 2.4, our proposed SCAN and SCAN+ have the best PSNR/SSIM performance. When compared with RCAN and RCAN+, our proposed SCAN+ can significantly improve the PSNR performance by as much as 0.44 dB and 0.33 dB respectively. Even without activating the strategy of self-ensemble, SCAN is still noticeably better than RCAN and RCAN+.

Beside quantitative PSNR/SSIM results, we have also included the subjective quality results comparison in Fig. 2.8 and Fig. 2.9. For image "cam2_09" in Fig. 2.8, we can see that RCAN suffers from severe edge blurring artifacts and text color distortions. Our proposed SCAN can reconstruct colorful texts with fewer blurring artifacts and less color distortion. For image "cam1_07" in Fig. 2.8, our SCAN is capable of recovering more edge details than RCAN (e.g., the sharpness of wall-pattern). For another image "cam2_05" in Fig. 2.9 (note that this example is really challenging - even ground-truth HR image has suffered a little bit of edge blurring), our SCAN can reconstruct

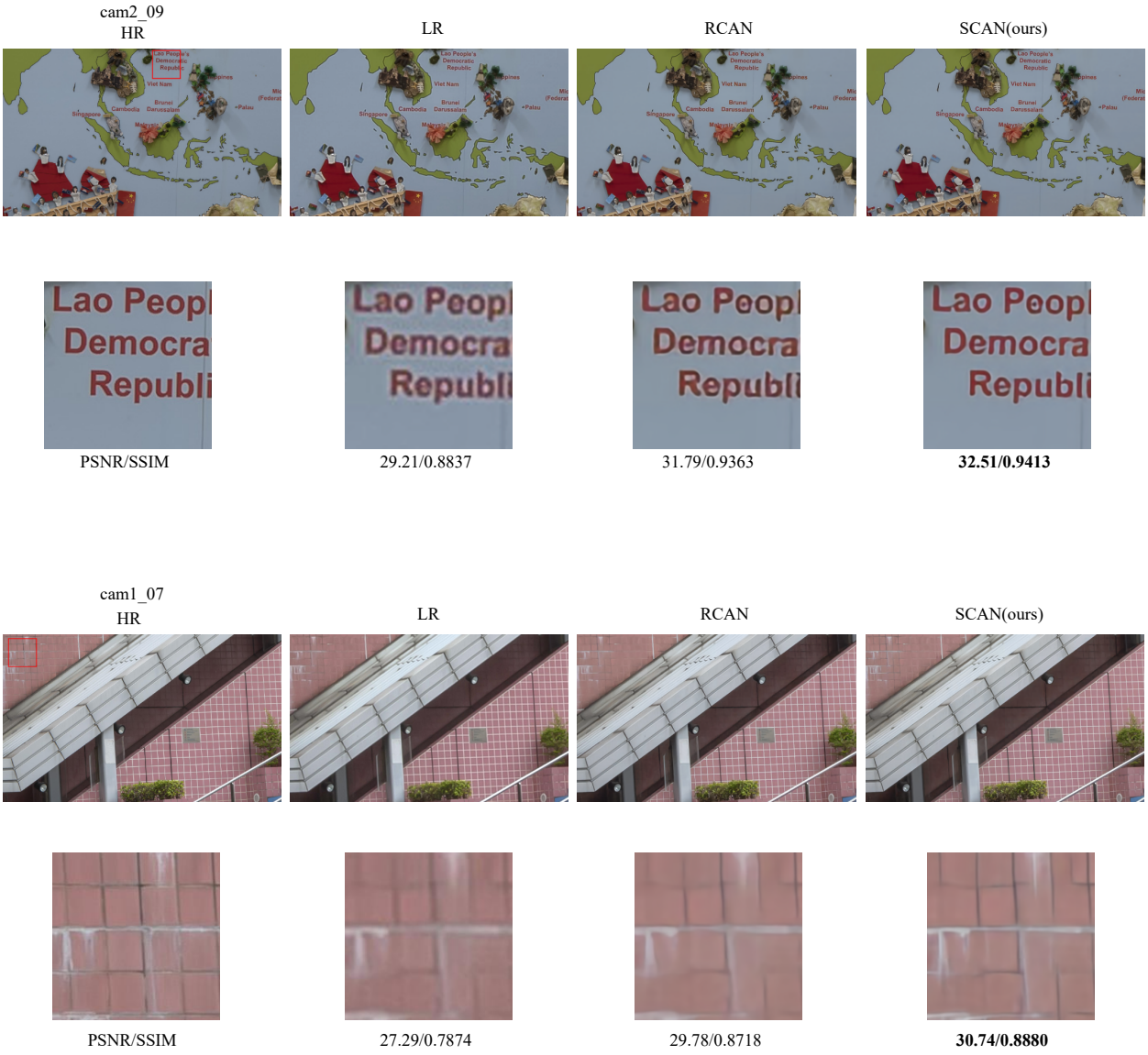

Figure 2.8: Visual results for validation data "cam1_07" and "cam2_09".

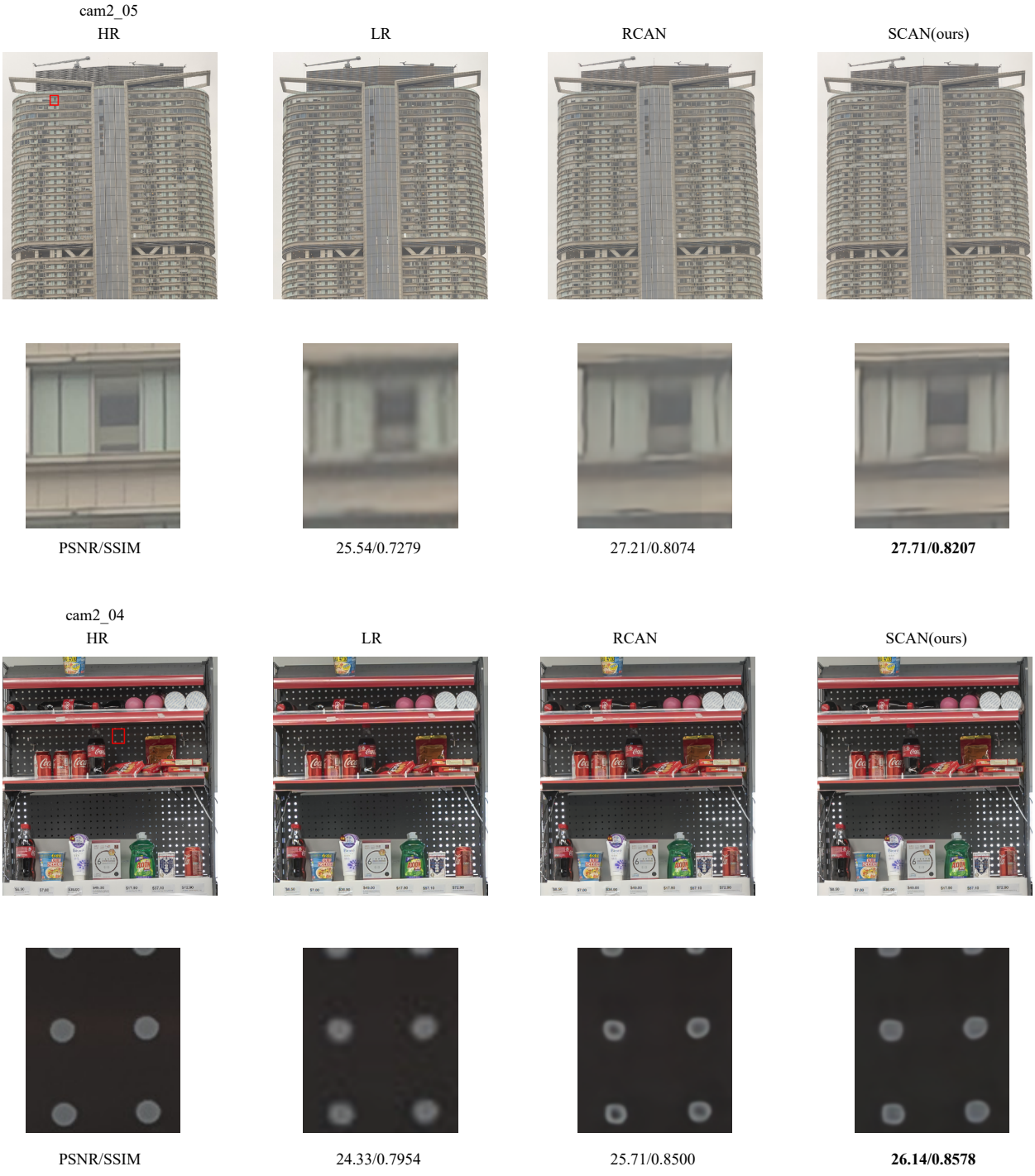

Figure 2.9: Visual results for validation data “cam2_05” and “cam2_04”.

	Baseline	RCAN	RCAN+	SCAN	SCAN+
PSNR	27.78	29.31	29.42	<u>29.59</u>	**29.75**
SSIM	0.8163	0.8606	0.8632	<u>0.8650</u>	**0.8687**

Table 2.4: Quantitative results of PSNR and SSIM for all methods. The higher the value of the metrics, the better performance is. **Bold** font indicates the best result and underline indicates the second best result.

the large-scale building structure details much better (e.g., the horizontal roof structure above the window and vertical edges on both sides of the window). For image "cam2_04" in Fig. 2.9, we can see that the dots in ground-truth image contain solid color; while in RCAN reconstructed image, they become hollow dots. One possible interpretation is that for fine structures like small dots, it takes both spatial and color attention mechanism to ensure the structural consistency among them. By contrast, our SCAN can still faithfully reconstruct those solid dots.

Finally, because the HR images for test data are not released, we cannot report the PSNR based results on test data. Alternatively, to evaluate the quantitative results among our method, baseline and RCAN on test data, we have used a new objective metric called Perceptual Index (PI) [82] (a no-reference image quality metric) which was recently developed to measure perceptual quality for SISR (e.g., the 2018 PIRM Challenge [86]). The PI score is defined by

$$\mathrm{PI} = \frac{1}{2}((10 - \mathrm{MA}) + \mathrm{NIQE}) \tag{2.9}$$

where MA denotes a no-reference quality metric [87] and NIQE refers to Naturalness Image Quality Evaluator [88]. Unlike PSNR or SSIM [75], the lower PI score, the better perceptual quality.

Table 2.5 includes the PI comparison between ours and other competing methods. SCAN reaches the lowest PI score, which implies the highest perceptual quality. Fig. 2.10 includes the PI comparison among baseline (LR), RCAN and SCAN on images "cam1_07" and "cam2_04" from test dataset (HR images are not released, so we will not be able to evaluate the fidelity or accuracy of SR reconstruction). But it can still be observed that SCAN is capable of delivering the most visually pleasant reconstruction of fine-detailed structures in basket on image "cam1_07". On another image "cam2_04", our proposed SCAN can significantly reduce the blurring of white-colored texts when compared with RCAN.

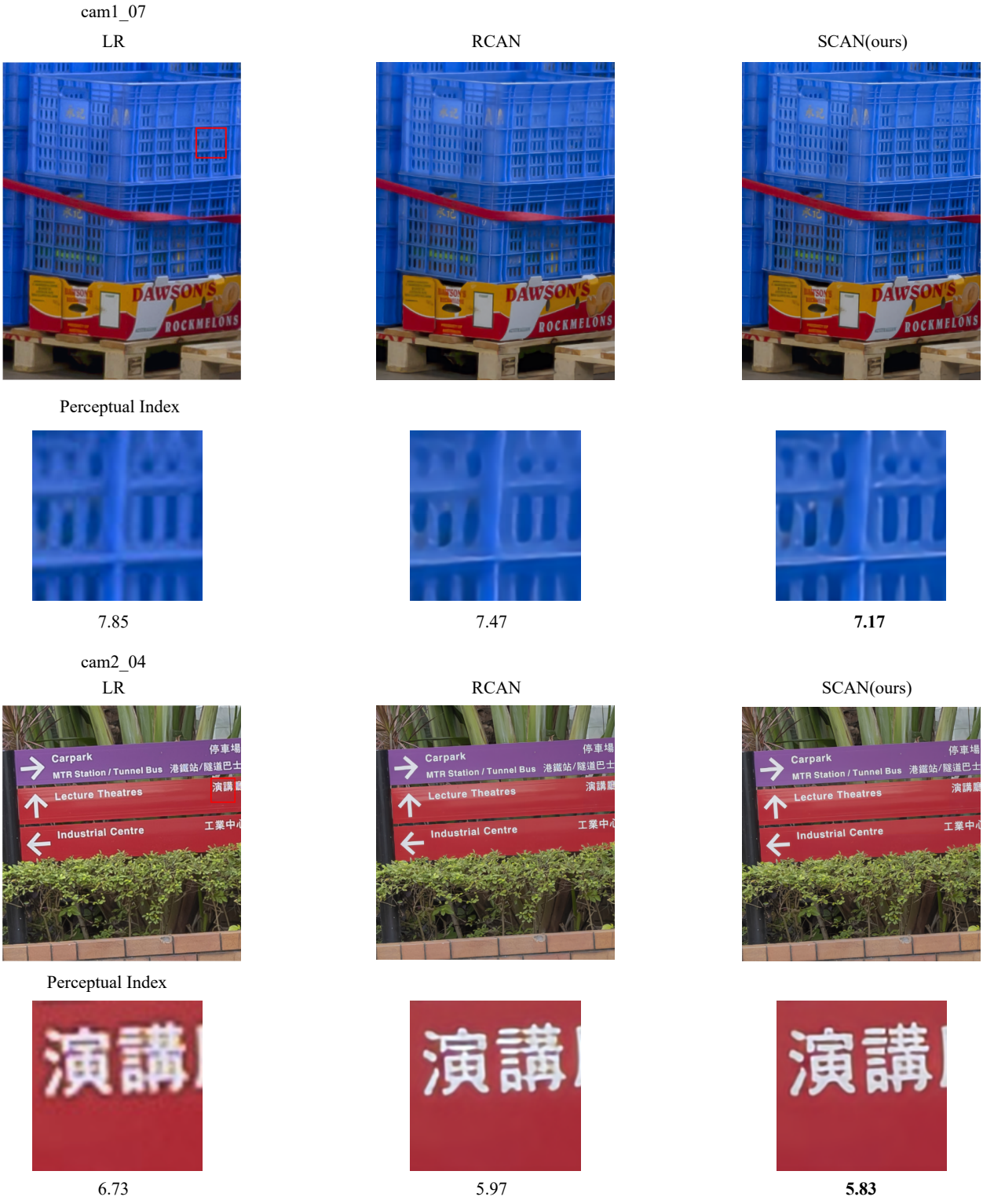

Figure 2.10: The visual results for test data "cam1_07" and "cam2_04". The results are based on perceptual index (PI) score since the HR image is not available. The lower PI score indicates the better perceptual quality.

	Baseline	RCAN	SCAN
PI Score	7.36	6.79	**6.68**

Table 2.5: Quantitative results of averaged perceptual index scores for all methods. The lower score is better. **Bold** font indicates the best result.

2.5 Summary

This chapter provided a state-of-the-art approach SCAN to solve the problem of single image super-resolution on real-world image dataset. The newly designed SCAM module can better utilize the color information of R, G, B three channels and learn the color attention map to better calibrate internal feature maps. Our proposed networks can significantly improve both objective and subjective results compared with the previous state-of-the-art RCAN.

Chapter 3

Joint Low-Level Vision Problems

3.1 Related Work

Both image demosaicing and super-resolution have been studied in decades in the open literature. In this section, we review image demosaicing and joint low-vision problems separately.

3.1.1 Image Demosaicing

Existing approaches toward image demosaicing also can be classified into two categories: model-based methods [28–31] and learning-based methods [37, 38, 40].

Li [30] proposed iterative demosaicing algorithm with a spatially adaptive stopping criterion to prevent color misregistration and reduce artifacts. Zhang and Wu [28] presented a demosaicing method to optimize directional filtering of the green-red and green-blue difference signals by linear minimum mean square-error. Ye and Ma [31] introduced an iterative residual interpolation strategy instead of color-component difference fields to all R,G,B three channels to reconstruct color images.

Model-based approaches rely on hand-crafted parametric models which often suffer from lacking of the generalization capability to handle varying characteristics in color images (i.e., the potential model-data mismatch). Recently, deep learning methods showed the advantages in image demosaicing field. Inspired by single image super-resolution model SRCNN [3], DMCNN [89]

utilized super-resolution based CNN model and ResNet [44] to investigate image demosaicing problem. CDM-CNN [90] introduced to apply residual learning [44] with a two-phase network architecture which firstly recovers green channel as guidance prior and then uses this guidance prior to reconstruct the RGB channels.

3.1.2 Joint Low-Level Vision

In addition to exploring image demosaicing methods only, there are several works studying Joint Demosaicing and Denoising (JDD) problem to jointly address demosaicing and denoising together with end-to-end optimization. Dong *et al.* [91] developed a deep neural network with GAN [64] and perceptual loss functions to solve JDD problems. Inspired by classical image regularization and majorization-minimization optimization, Kokkinos and Lefkimmiatis [39] proposed a deep neural network to solve JDD problem. Deep learning based image demosaicing techniques have shown convincingly improved performance over model-based ones on several widely-used benchmark dataset (e.g., Kodak and McMaster [6]). However, the issue of suppressing spatio-spectral aliasing has not been addressed in the open literature as far as we know and the problem of JDSR has been under-researched so far with the only exception of [43], the potential benefits of JDSR are still worth to be further explored.

3.2 Proposed Approach of JDSR V1 – DSERN

The hierarchy of JDSR network design goes like: DSERN (Fig. 3.1) $\rightarrow$ D_RG subnetwork (Fig. 3.2a) $\rightarrow$ D_SERB module (Fig. 3.2b) $\rightarrow$ DSE block (Fig. 3.3).

3.2.1 DSERN: Deeper and Wider are Better

Channel attention mechanism has been successfully applied in both high-level (e.g., SENet [19] and LS-CNN [92]) and low-level (e.g., RCAN [2]) vision tasks. A channel attention module first squeezes the input feature map and then activates one-time reduction-and-expansion to excite the squeezed feature map. Such strategy is not optimal for recovering missing high-frequency information in SISR when the network is very deep; meanwhile, JDSR problem requires simultaneous

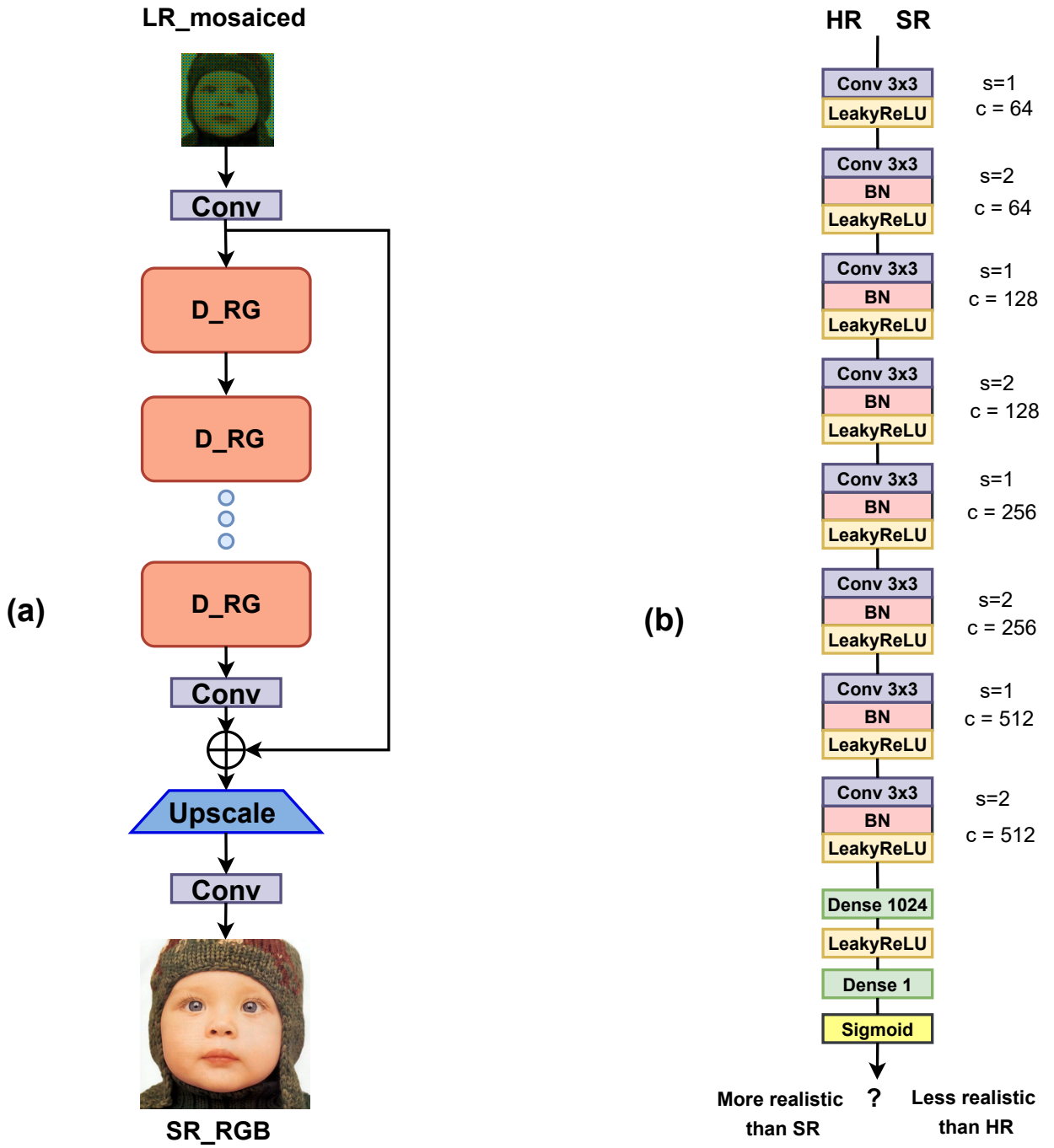

Figure 3.1: Overview of the proposed DSERN network architecture, the generator is shown in (a), D_RG stands for Dense_Residual_Group and $\oplus$ denotes element-wise sum; (b) is the structure of discriminator used (s is the stride and c is the number of feature maps).

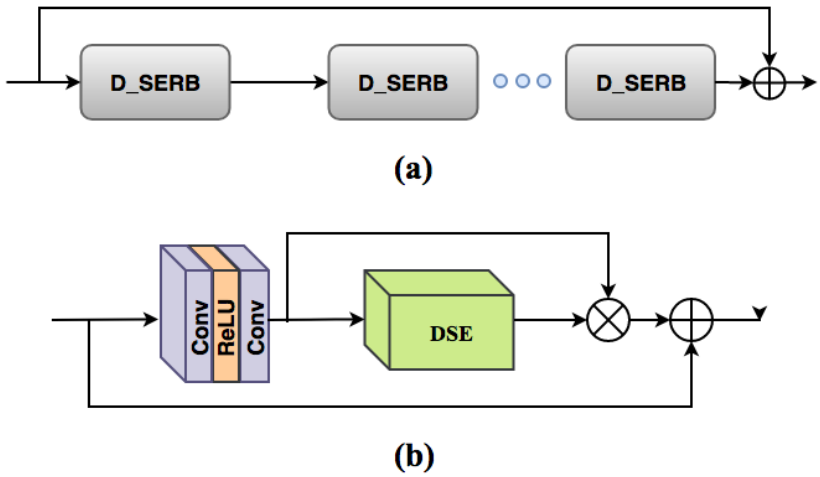

Figure 3.2: Structure of (a) D_RG subnetwork and (b) D_SERB module where $\otimes$ denotes element-wise product and $\oplus$ denotes element-wise sum respectively.

recovery of incomplete color information across R, G, B bands, which requires extra attention toward the dependency among spectral bands. How to generalize the channel attention mechanism from spatial-only to joint *spatio-spectral* serves as the key motivation behind our approach.

As discussed in [2], high-frequency components often correspond to regions in an image such as textures, edges, corners and so on. Conventional Conv layers have limited capability of exploiting contextual information outside the local receptive field especially due to missing data in Bayer pattern. To overcome this difficulty, we propose to design a new Densely-Connected Squeeze-and-Excitation Residual Block (D_SERB) as shown in Fig. 3.2b and Fig. 3.3. The proposed D_SERB is designed to implement a *deeper* and *wider* spatio-spectral channel attention mechanism for the purpose of more effectively suppressing spatio-spectral aliasing in LR Bayer pattern.

Unlike SENet [19] and RCAN [2] (using one-time reduction-and-expansion), D_SERB uses multiple expansion modules after reduction to assure more faithful information recovery when the network gets deeper and wider. As shown in Fig. 3.2, we have kept both long skip and short skip connections like RCAN in order to make the overall training stable and facilitate the information flow both inside and outside the D_SERB modules. Although similar idea of local feature fusion existed in residual dense block of RDN [42], our hybrid design - i.e., the Dense SE (DSE) block combining the ideas in RDN and RCAN - is novel from the perspective of achieving joint spatio-spectral attention for JDSR. Spatio-spectral channel attention mechanism in the proposed D_SERB

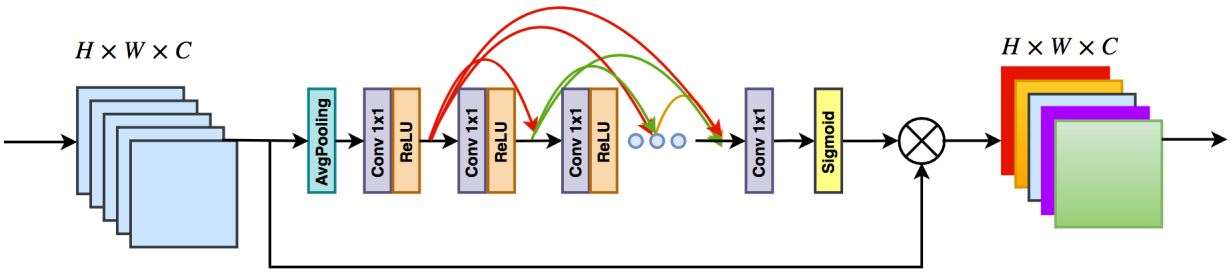

Figure 3.3: Flowchart of Dense Squeeze-and-Excitation (DSE) block (⊗ denotes element-wise product).

module can help to recalibrate input features via channel statistics [19] across spectral bands. In SISR, one-time reduction-and-expansion operation might be sufficient for capturing channel-wise dependencies for LR color images; however our JDSR task aims at recovering two-third of missing data in spectral bands in addition to high-frequency spatial details, which calls for the design of deeper and wider networks.

3.2.2 Dense Squeeze-and-Excitation Residual Block

The key to deeper and wider networks lies in the design of D_SERB module - i.e., how to use multiple expansions after reduction to assure more faithful information recovery both inside and outside D_SERB modules? As shown in Fig. 3.2b), we propose a Dense Squeeze-and-Excitation (DSE) block in which the channel size can be expanded *step by step* (see Fig. 3.3). The key advantages of this newly designed DSE block include: 1) the reduced channel descriptor can be smoothly activated *multiple times* and therefore more faithful information across spatio-spectral domain is accumulated; 2) dense-connection can increase the network *depth and width* without running into the notorious vanishing-gradient problem [93]; 3) both information flow and network stability, which are important to a principled solution to JDSR, can be jointly improved by introducing dense connections to SE residual blocks (so we can train even deeper than RCAN [2]).

More specifically, to implement the DSE block, we first apply global average pooling to *squeeze* input feature maps. Let us denote the input feature maps by $\mathbf{U} = [u_1, u_2, ..., u_C]$, which contains C feature maps with the dimension of $H \times W$. Then the global average pooling output

$z \in \mathbb{R}^C$ can be calculated by:

$$z_C = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mathbf{u}_C(i, j) \tag{3.1}$$

where z_C is the c-th element of z, $u_C(i, j)$ is the pixel value of the c-th feature at position (i, j) from input feature maps. Then we propose to implement a simple gating mechanism as adopted by previous works including SENet [19] and RCAN [2]:

$$s = \sigma(\mathbf{W}_E(\delta(\mathbf{W}_S(z)))) \tag{3.2}$$

where σ refers to a sigmoid function, δ denotes the ReLU function. Note that both $\mathbf{W}_S$ and $\mathbf{W}_E$ are Conv layers with weights $\mathbf{W}_S \in \mathbb{R}^{1\times1\times\frac{C}{r}}$ and $\mathbf{W}_E \in \mathbb{R}^{1\times1\times C}$, r is the reduction ratio to reduce the dimension of z (details about this hyperparameter controlling the tradeoff between the capacity and the complexity can be found in SENet [19]).

In order to achieve deeper and wider channel attention, we propose a novel strategy of activating the reduced features *step by step* (instead of one-shot) with dense connections. As shown in Fig. 3.3, after reducing z by a factor of r, we can gradually expand (i.e., smooth activation) the feature map N times where $N = r - 1$. The detailed procedure of our proposed DCA module can be written as:

$$X_0 = \delta(\mathbf{W}_S(z)), \tag{3.3}$$

$$X_i = \delta(\mathbf{W}_i(X_{i-1})), \text{ where } i \in [1, N] \tag{3.4}$$

$$\hat{s} = \sigma(\mathbf{W}_E([X_0, X_1, X_2, ..., X_N])) \tag{3.5}$$

where σ, δ, $\mathbf{W}_E$, $\mathbf{W}_S$ are the same as Eq. (3.2), $\mathbf{W}_i \in \mathbb{R}^{1\times1\times\frac{C}{r}}$ and $[X_0, X_1, X_2, ..., X_N]$ refers to the concatenation of feature maps by each DSE layer.

Finally, we can rescale the input feature map by

$$\hat{\mathbf{U}} = \hat{s} \cdot \mathrm{U} \tag{3.6}$$

With the new DSE block, we can train even deeper network than RCAN [2] thanks to the improved information flow.

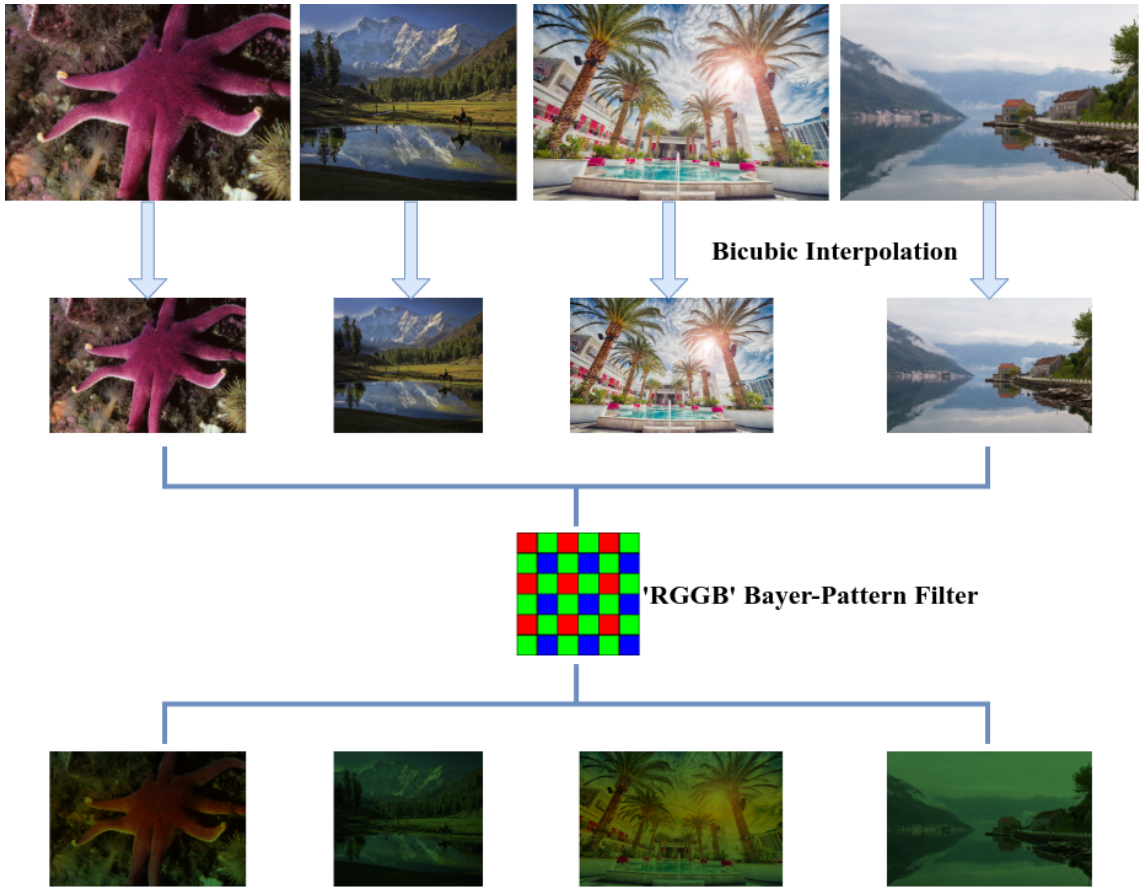

Figure 3.4: The general process to generate low-resolution training data. First to downsample HR image to LR image with desired scale-factor (ex. 2, 3, 4) by Bicubic interpolation, then generate mosaiced image with RGGB Bayer filter by padding zero for missed color pixels.

3.3 Dataset Setup

DIV2K dataset [81] is widely used as training dataset in single image super-resolution (SISR) methods. It includes 800 2K-resolution RGB images with a large diversity of contents for training, 100 2K images for validation and 100 images for testing. For our proposed JDSR approach, we only use 800 images to train.

For testing, we have generally evaluated both popular image super-resolution benchmark datasets including Set5 [32], Set14 [94], B100 [95], Urban100 [96], and Manga109 [97], and popular image demosaicing datasets such as McMaster [98] and Kodak PhotoCD. Also to verify the effectiveness of our proposed networks, we then have evaluated the challenging patches database provided by [6] where they detected and cropped very challenging patches from the web for demosaicing performance test.

To pre-process training and testing data, we downsample original HR images by a factor of $2\times$, $3\times$, $4\times$ using Bicubic interpolation then generate the 'RGGB' Bayer pattern as shown in Fig. 3.4.

Based on previous work [89] and our own study (refer to next paragraph), supplying three-channels separately as the input (instead of the mosaicked single-channel composition) works better for the proposed network architecture.

Note that we have to be careful about four different spatial arrangements of Bayer patterns [99]) in our definition of feature maps. One can either treat the Bayer pattern like a gray-scale image (one-channel setting) which ignores the important spatial arrangement of R/G/B; or take spatial arrangement as a priori knowledge and pad missing values across R,G,B bands by zeroes (three-channel setting). As shown in Fig. 3.5, the former has the tendency of producing color misregistration artifacts, which suggests the latter works better. Our experimental result has confirmed a similar finding previously reported in [89].

3.4 Experimental Results and Limitation

In this section, we demonstrate the network setting, training details, ablation study and experimental results of proposed JDSR problem.

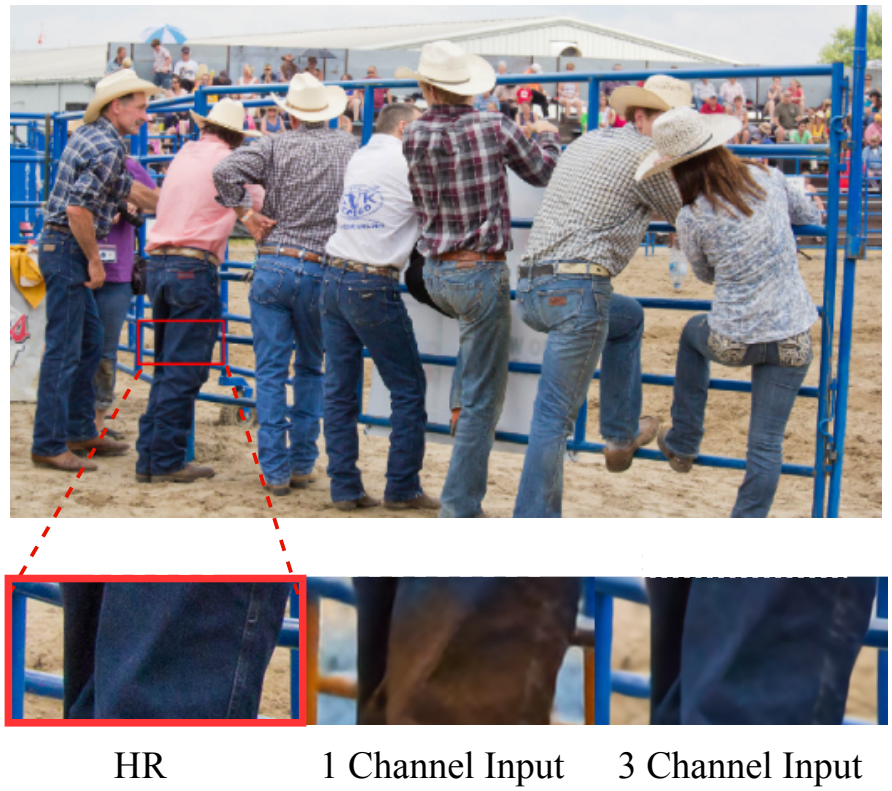

Figure 3.5: Visual comparison of training data effect, the bottom images, from left to right, are HR image, SR image generated by one-channel feature map (raw Bayer-pattern), SR image generated by three-channel feature map (Bayer-pattern with zero padding).

3.4.1 Implementation Details

In our proposed DSERN networks, we have kept the basic setting same as RCAN [2]: D_RG is set to 10 and every D_RG contains 20 D_SERBs. All kernel size of Conv layers is 3×3 with 64 filters ($C = 64$) except the Conv layers in our DSE modules. The reduction ratio is $r = 16$. The upscale module we have used is the same as [100]. The last layer filter is set to 3 in order to output super-resolved color images.

In our PyTorch implementation of DSERN, we first randomly crop the 3-channel Bayer patterns as small patches with the size of 48×48, and crop the corresponding HR color images, with a batch size of 16; then we augment the training set by standard geometric transformations (flipping and rotation). Our model is trained and optimized by ADAM [83] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-8}$. The initial learning rate is set to 1×10^{-4}, the decay factor is set to 5, which decreases the learning rate by half after [80k, 120k, 150k, 180k] steps; the L_1 loss function is applied

Figure 3.6: Visual results among competing approaches for Manga109 dataset at a scaling factor of 2.

to minimize the error between HR and SR images.

Because the codes of RDSR [43] are not publicly available, we have tried our own best to reproduce RDSR using PyTorch while keeping the batch size (16), patch size (64×64) and number of residual blocks (24) the same as the original work [43]. The learning rate and decay steps in RDSR implementation are the same as those in our DSERN. This way, we have striven to make the experimental comparison against RDSR [43] as fair as possible.

3.4.2 PSNR/SSIM Comparisons

It is convenient to further improve the performance of our DSERN by a so-called self-ensemble strategy (as done in previous works [2,42,63,85]). The improved results are denoted as "DSERN+". We have compared our methods against two benchmark methods: a separated (brute-force) approach Flex [1] + RCAN [2], recently published in the literature RDSR [43]. To evaluate the results of Flex [1] + RCAN [2] approach, we first demosaiced the LR mosaiced images by using Flex to get LR color images, then super-resolved them by applying a pre-trained RCAN model.

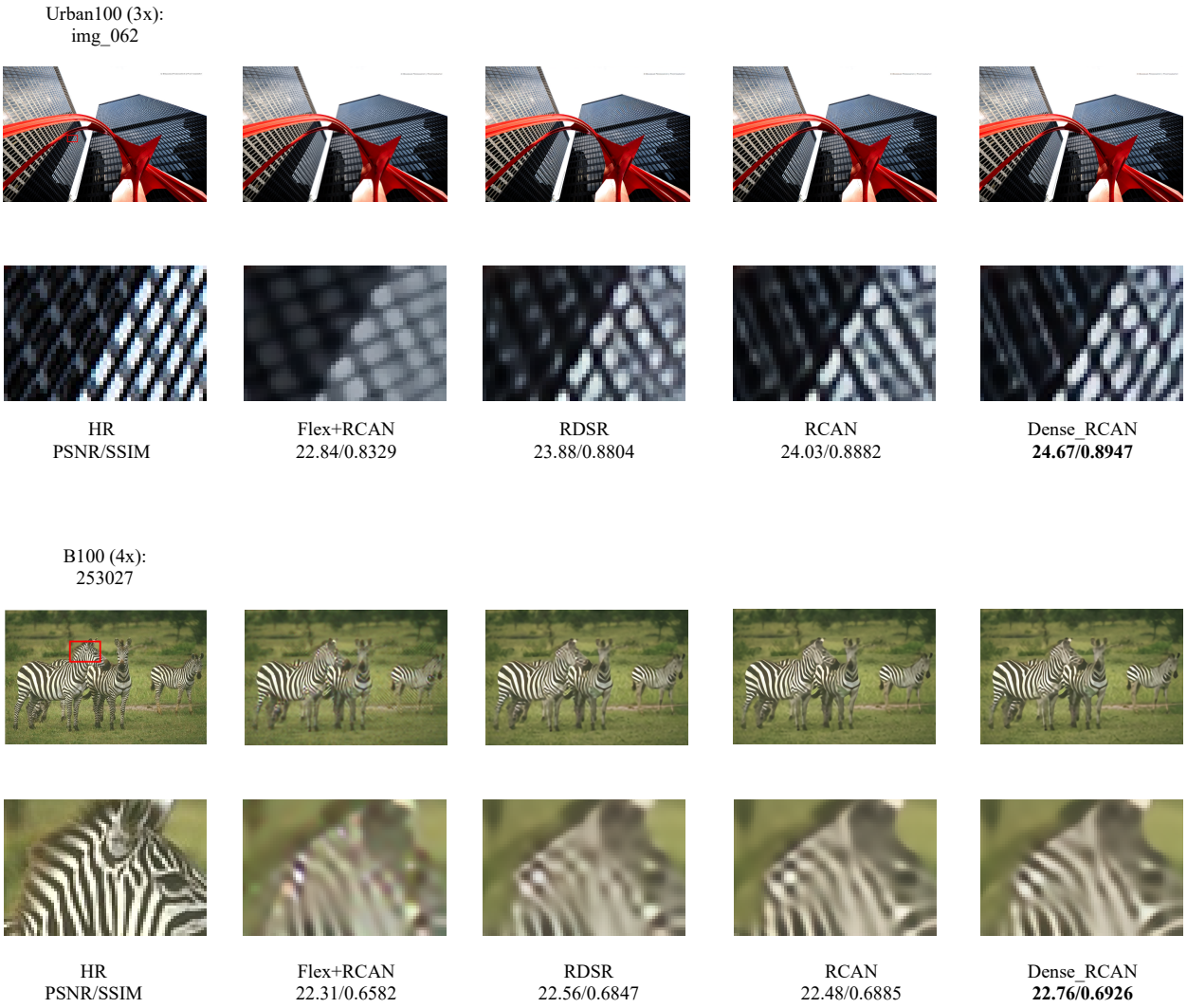

Figure 3.7: Visual results among competing approaches for Urban100 and B100 datasets at a scaling factor of 3 and 4.

Method	Scale	Set5	Set14	B100	Urban100	Manga109	McM	PhotoCD
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
FlexIPS [1]+RCAN [2]	x2	35.18/0.9387	31.24/0.8776	31.00/0.8647	31.23/0.9119	30.32/0.9199	34.80/0.9301	43.02/0.9610
RDSR [43]	x2	36.29/0.9485	32.56/0.9008	31.56/0.8850	31.20/0.9148	36.14/0.9625	35.90/0.9423	43.74/0.9655
RCAN [2]	x2	36.54/0.9499	32.74/0.9032	31.68/0.8878	31.74/0.9200	36.65/0.9643	36.18/0.9445	43.91/**0.9661**
DSERN (ours)	x2	36.55/0.9500	32.71/0.9031	31.70/0.8879	31.78/0.9207	36.72/0.9652	36.23/0.9448	43.90/**0.9661**
DSERN+ (ours)	x2	**36.62/0.9504**	**32.80/0.9041**	**31.73/0.8884**	**31.94/0.9221**	**36.89/0.9658**	**36.33/0.9456**	**43.92/0.9661**
FlexISP+RCAN	x3	31.21/0.8731	28.55/0.7884	27.31/0.7310	25.68/0.7800	27.58/0.8647	31.25/0. 8661	40.32/0.9402
RDSR	x3	33.05/0.9103	29.54/0.8211	28.61/0.7859	27.64/0.8375	31.69/0.9225	32.21/0.8842	40.90/0.9458
RCAN	x3	33.24/0.9125	29.67/0.8241	28.69/0.7882	27.90/0.8436	32.06/0.9267	32.42/0.8874	41.11/0.9469
DSERN (ours)	x3	33.27/0.9127	29.67/0.8240	28.70/0.7884	27.92/0.8439	32.06/0.9268	32.41/0.8875	41.10/0.9469
DSERN+ (ours)	x3	**33.35/0.9134**	**29.73/0.8251**	**28.74/0.7892**	**28.05/0.8462**	**32.27/0.9286**	**32.52/0.8888**	**41.13/0.9471**
FlexISP+RCAN	x4	29.57/0.8376	26.94/0.7177	26.68/0.6896	25.29/0.7503	26.69/0.8427	27.78/0.7651	38.28/0.9201
RDSR	x4	30.87/0.8712	27.91/0.7589	27.16/0.7151	25.65/0.7695	28.86/0.8800	30.10/0.8328	38.81/0.9258
RCAN	x4	31.04/0.8746	27.98/0.7613	27.20/0.7175	25.91/0.7784	29.12/0.8856	30.24/0.8367	39.01/0.9271
DSERN (ours)	x4	31.02/0.8747	27.99/0.7620	27.20/0.7177	25.92/0.7788	29.13/0.8857	30.25/0.8368	39.02/0.9273
DSERN+ (ours)	x4	**31.12/0.8761**	**28.06/0.7635**	**27.24/0.7188**	**26.05/0.7819**	**29.36/0.8888**	**30.35/0.8387**	**39.08/0.9277**

Table 3.1: PSNR/SSIM comparison among different competing methods. **Bold** font indicates the best result and underline the second best.

Method	Scale	Set5	Set14	B100	Urban100	Manga109	McM
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
ResNet	x2	36.48/0.9498	32.71/0.9030	31.67/0.8876	31.65/0.9201	36.48/0.9642	36.11/0.9443
RCAN	x2	36.54/0.9499	**32.74/0.9032**	31.68/0.8878	31.74/0.9200	36.65/0.9643	36.18/0.9445
DSERN (ours)	x2	**36.55/0.9500**	32.71/0.9031	**31.70/0.8879**	**31.78/0.9207**	**36.72/0.9652**	**36.23/0.9448**

Table 3.2: Ablation study for ResNet, ResNet with CA (RCAN) and ResNet with proposed DSERN. **Bold** font indicates the best result.

Note that we have used the pre-trained RCAN weights provided by the authors on GitHub.

Table 3.1 shows PSNR/SSIM comparison results for scaling factors of $2\times$, $3\times$ and $4\times$. It can be seen that our DSERN+ method perform the best for all datasets and scale factors. Even without self-ensemble, the performance of DSERN still leads all of datasets and scaling factors. We observe that moderate PSNR/SSIM gains $(0.2-0.5dB)$ over previous state-of-the-art. Since PSNR/SSIM metrics do not always faithfully reflect the visual quality of images, we have also included the subjective quality comparison results for image "TotteokiNoABC" in Fig. 3.6. It can be readily observed that for the top of the pink sock, only our DSERN can faithfully recover stripe patterns; both the brute-force approach (Flex+RCAN) and RDSR have produced severe blurring artifacts. Taking another example, Fig. 3.7 shows the comparison at two other scaling factors ($3\times$ and $4\times$). For "img_062", we observe that all approaches contain noticeable visual distortion, but our DSERN method can recover more shape details than other competing approaches; for "253027", zebra pattern recovered by DSERN appears to have the highest quality. For more visual comparison, see Fig. 3.8 and Fig. 3.9 which show more visual comparison among various competing approaches (please zoom in for a detailed comparison).

3.4.3 Ablation Studies

To demonstrate the effect of proposed DSE module, we study the networks: 1) only based on ResNet; 2) ResNet with channel attention module (RCAN); 3) ResNet with proposed DSE module (DSERN). All three networks are trained under same setting for fair comparison. The general SR benchmark datasets are used, scale factor is 2. From Table 3.2, we have found that ResNet has similar performance on Set5, Set14 and B100 to more advanced RCAN and DSERN. But when

Figure 3.8: Visual quality comparison of JDSR results among competing approaches at a scaling factor of 4.

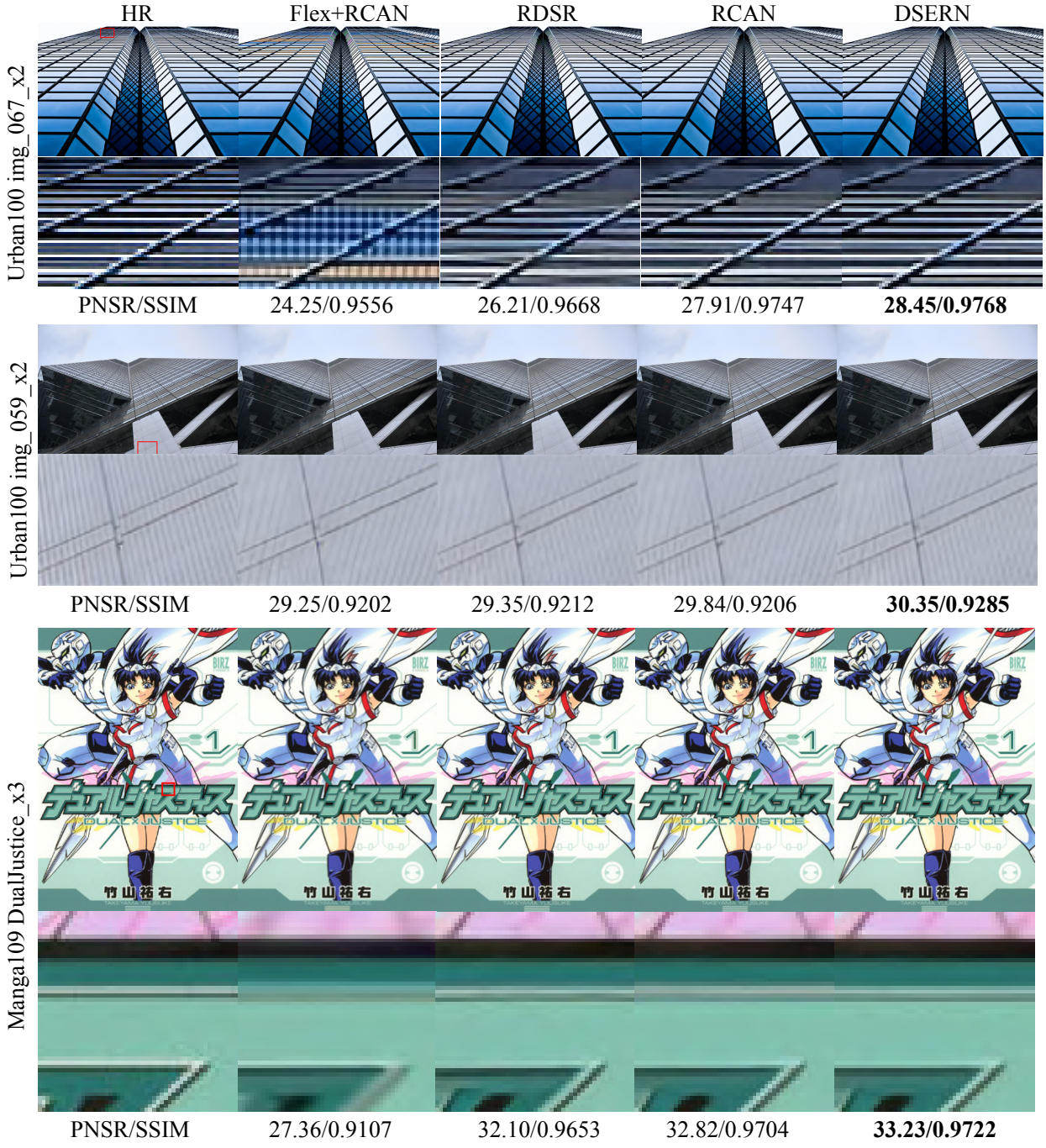

Figure 3.9: Visual quality comparison of JDSR results among competing approaches at a scaling factor of 3 or 4.

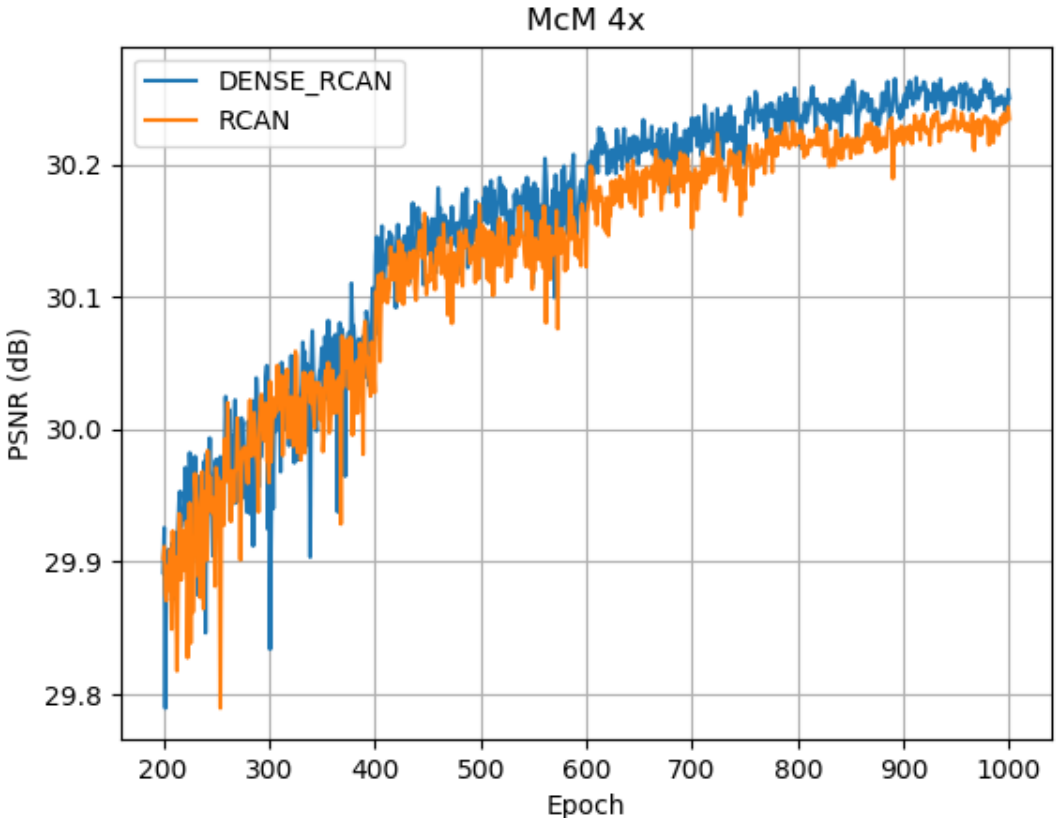

Figure 3.10: Performance comparison during training to valid the efficiency of the proposed DSE module on McM dataset. The scale factor is 4.

compared on Urban100, Manga109 and McM, RCAN and DSERN have better performance than ResNet; and the proposed DSERN has the best performance on most benchmark datasets.

Also in Fig. 3.10, it shows the comparison of training process between RCAN with or without DSE module. At first 400 epochs, DSE module did not show advantages, but after that, especially at the end of training, DSE module helped to improve the whole network performance, because DSE can reconstruct deeper information which benefit by deeper and wider networks design.

3.4.4 Limitation

Although our proposed DSERN network shows some reliable results, there remain some limitations that need to be addressed. First of all, training and testing processes are slow, because of the network complexity, which limits its practical application; secondly, it is difficult to directly learn mapping between a mosaiced image and a high-resolution RGB color image because mosaiced

image lost 2/3 of information.

3.5 Proposed Approach of JDSR V2 – RDSEN

To tackle the limitation of DSERN, we re-design a JDSR network, called RDSEN to boost the running time of training and testing. The hierarchy of our network design is described by the following flow diagram: Overview of the proposed network (Fig. 3.11) $\rightarrow$ PDNet subnetwork (Fig. 3.12) $\rightarrow$ RDSEN (Fig. 3.13) $\rightarrow$ RDSEB with channel attention (Fig. 3.14).

3.5.1 Pre-demosaicing Network

One challenging issue in JDSR is that not only high-frequency components but also two-third of color pixels are missing. This issue can lead to undesirable distortion or artifact in the reconstructed full-resolution color image. Inspired by recent work CBDNet [101], we have designed a pre-demosaicing network (PDNet) for initially demosaicing the Bayer pattern as a pre-processing step to reduce the gap between LR CFA data and HR color image. As shown in Fig. 3.11 before the RDSEN module, we have adopted a model-based demosaicing method called iterative-residual interpolation (IRI) [67] to generate an intermediate demosaicing result, which will be used as the input to the refinement module. This intermediate demosaicing results will be refined by PDNet as shown in Fig. 3.12 (conceptually similar to ResNet [44]). In the PDNet, we opt to separately process Red, Green, and Blue channels. For Red and Blue channel, we use a convolution layer with stride of 4 to shrink the corresponding Bayer pattern and then upscale them with a factor of 2; for Green channel we shrink it by a convolution layer with stride of 2. This is because the Red and Blue channels each contains one-fourth information and G channel contains one-half information. Then we concatenate RGB feature maps and fuse them with a 1×1 kernel of convolution layer. Finally, we upscale the fused feature maps back to the same size as input CFA data.

3.5.2 Residual-Dense Squeeze-and-Excitation Network

Unlike DSERN using a multiple expansion attention strategy to smoothly conduct information flow, which is missing to utilize global important features thus inefficient, we propose a new

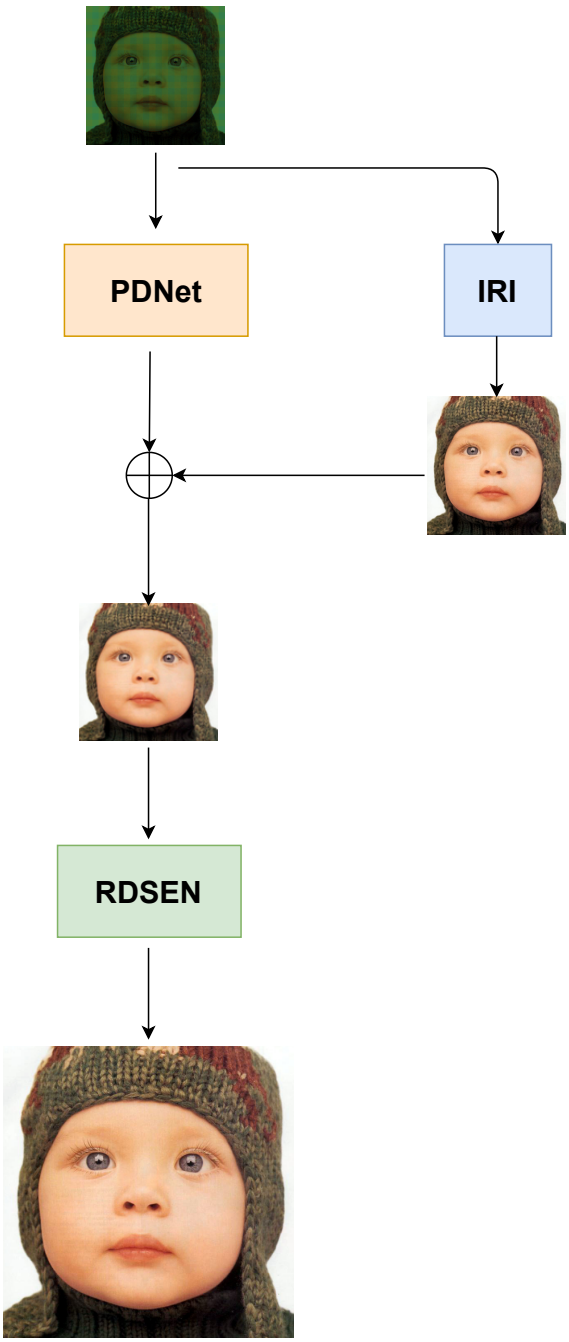

Figure 3.11: Overview of the proposed RDSEN with PDNet network architecture, $\oplus$ means element-wise sum.

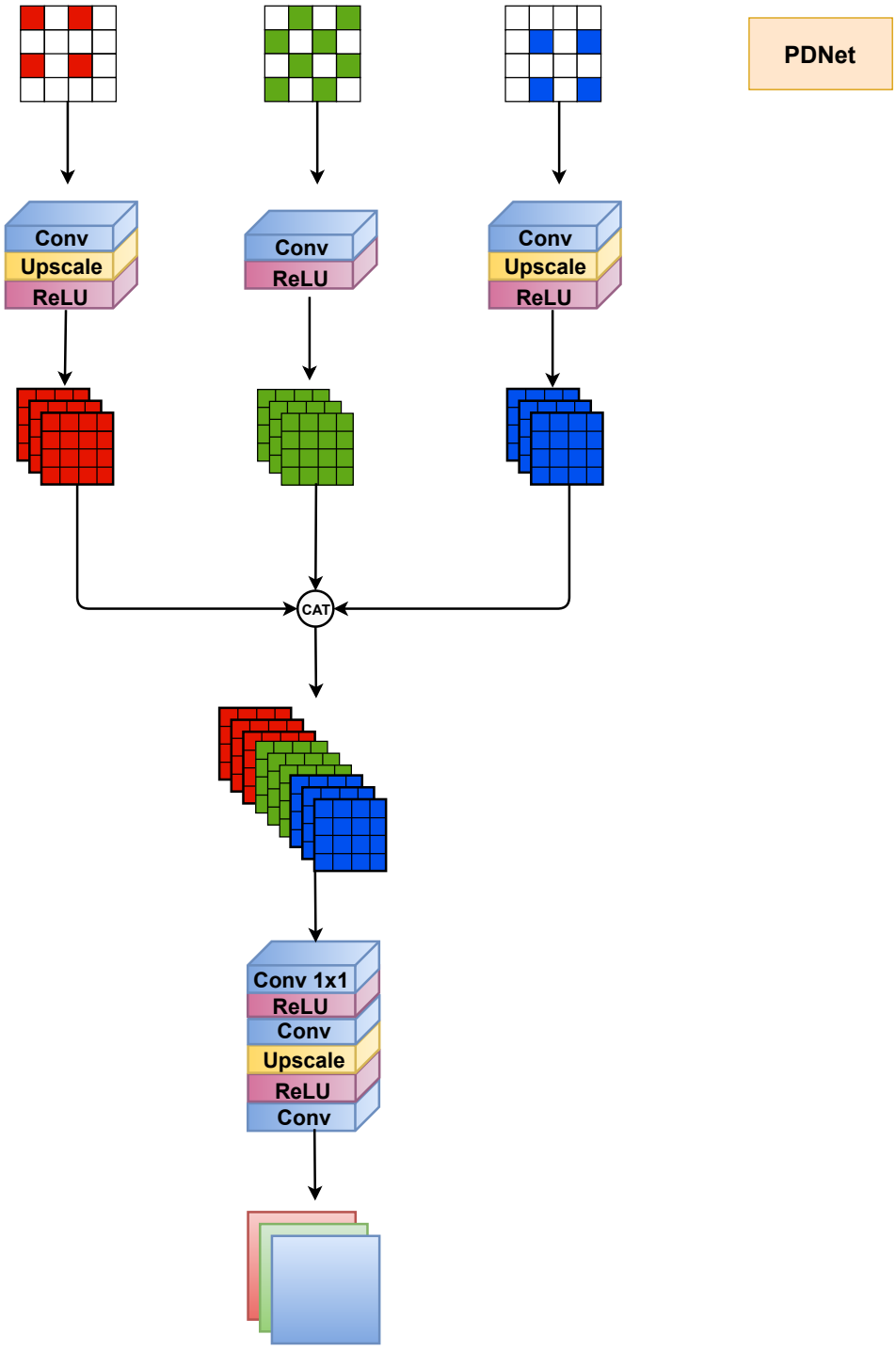

Figure 3.12: Structure of PDNet, 'CAT' is feature concatenation.

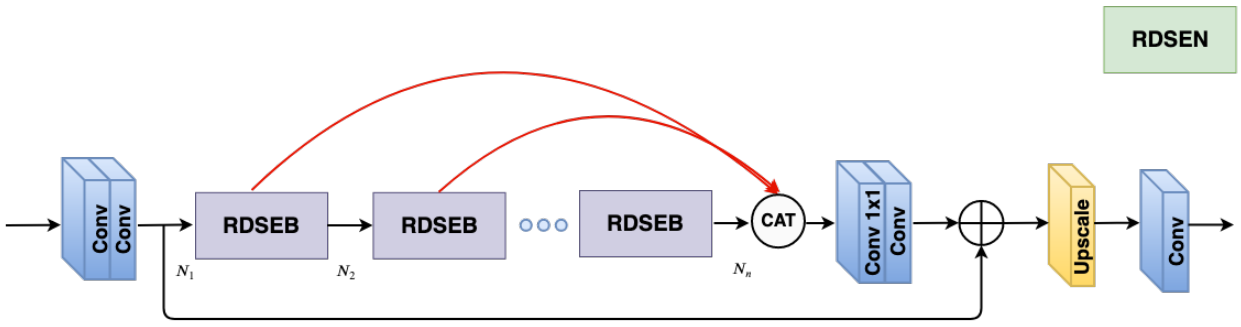

Figure 3.13: Structure of RDSEN, 'CAT' is feature concatenation and $\oplus$ denotes element-wise sum respectively.

Residual-Dense Squeeze-and-Excitation Network (RDSEN) as shown in Fig. 3.13 and Fig. 3.14 to overcome the limitation of DSERN. RDSEN includes a couple of RDSEB blocks attempts to fuse both *local and global* information to achieve more efficient deeper and wider networks by exploiting both local and global attention features. The same as DSERN and [2], both long skip and short skip connections are used to make sure the training stable and information conduction between deep and front layers (as shown in Figs. 3.13 and 3.14). The RDSEB block combining the ideas from RDN [42] and RCAN [2] is novel because it represents an alternative approach to strike an improved tradeoff between cost (in terms of network parameters) and performance (in terms of visual quality).

Our design of concatenating RDSEB modules also has its merit from the perspective of exploiting joint spatio-spectral attention for JDSR. We have experimentally verified that such design of deeper and wider networks [102] based on concatenation of multiple RDSEB modules indeed helps the boosting of our JDSR performance.

3.5.3 Residual-Dense Squeeze-and-Excitation Block

Different from Densely-Connected Squeeze-and-Excitation Residual Block (DSERB) blocks based on multi-expansion for channel attention mechanism, which inefficiently enhance local information only to help recover high-frequency components, as shown in Fig. 3.14, the newly designed RDSEB blocks exploit short skip connection and multiple concatenations after channel attention mechanism to fuse local and global information. The main advantage of RDSEB is that DSERB extends and concatenates features at the channel attention level (see Fig. 3.2) but RDSEB

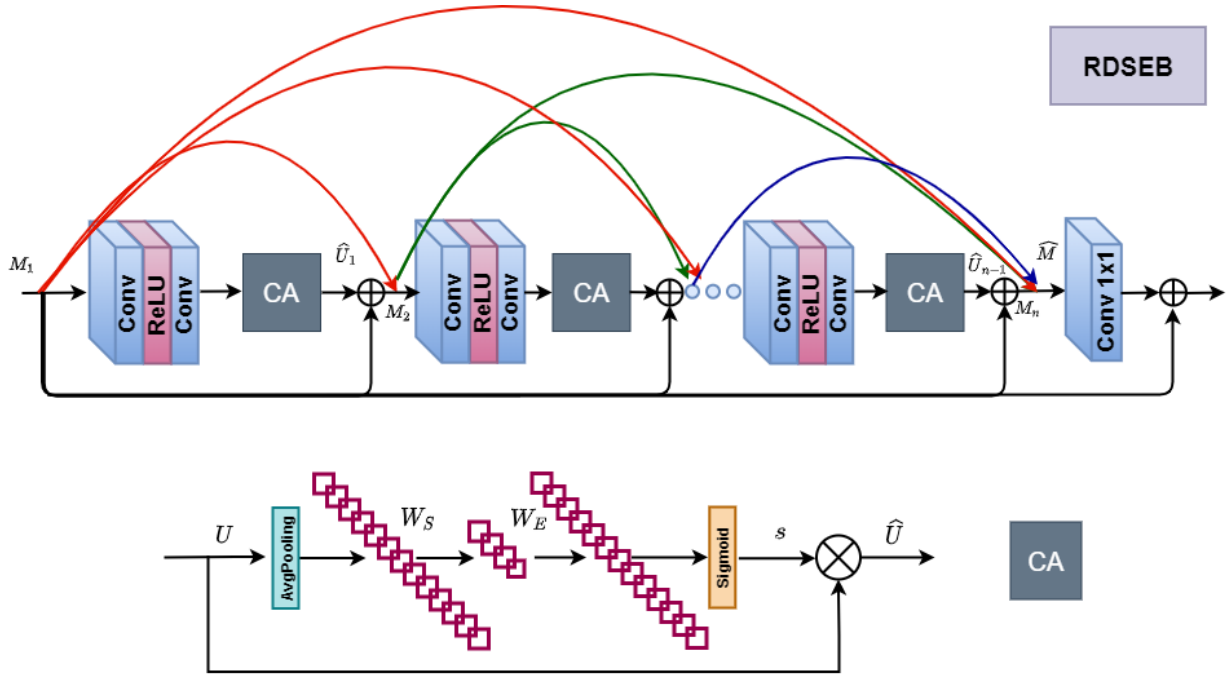

Figure 3.14: Flowchart of Residual-Dense Squeeze-and-Excitation Block (RDSEB) and Channel Attention (CA) module ($\otimes$ denotes element-wise product).

does this at the feature level (see Fig. 3.14) which can acquire richer information under less model complexity compared with DSERB.

To implement RDSEB, we first apply the same CA block as described in Sec. 3.2.2. Then, in order to achieve more efficient deeper and wider channel attention than DSERB, we conduct a new strategy of connecting each output of channel attention block not only with short skip connection (residual) but also dense-connection as shown in Fig. 3.14. To formalize this problem, define U as the input feature map to CA module, the rescaled input feature map $\hat{\mathbf{U}}$ can be expressed as:

$$\hat{\mathbf{U}} = s \cdot \mathrm{U} \tag{3.7}$$

where '$\cdot$' stands for element-wise product.

Finally, to implement dense-connection, we define M_1 as the input feature map of RDSEB block. Then the output feature map $\hat{M}$ can be written as the following equations:

$$M_i = \hat{\mathbf{U}}_{i-1} + M_1, \text{ where } i \in [2,n] \tag{3.8}$$

$$\hat{M} = [M_1, M_2, ..., M_n] \tag{3.9}$$

where $[M_1, M_2 ... M_n]$ refers to the concatenation of feature maps, $\hat{\mathbf{U}}_{i-1}$ is the corresponding output of CA module at the $i-1$-st stage as shown in Fig. 3.14. With the new RDSEB block, we can train a deeper and wider network thanks to the improved information flow.

3.6 Perceptual Optimization: Relativistic Discriminator and Loss Function

3.6.1 Texture-enhanced Relativistic average GAN (TRaGAN)

The discriminator D in standard GAN [64] only estimates the probabilities of real/fake images, and the interaction between generator and discriminator is interpreted as a two-player minimax game. It can be expressed as $D(x) = \sigma(C(x))$, where σ is sigmoid function, $C(x)$ is non-transformed layer, x is the input image. Such idea has been successfully applied to the problem of SISR such as SRGAN [5] in which the super-resolved image (fake version) is compared against the ground-truth (real version). In other words, discriminator D serves as a judge for perceptual optimization of generator (as shown in Fig. 3.1(b)).

Unlike standard GAN, relativistic average GAN (RaGAN) [65] can make the discriminator D to estimate the probability based on both real and fake images, making a real image more realistic than a fake one (on the average). According to [65], RaGAN can not only generate more realistic images but also stabilize the training progress. Recently, the benefit of RaGAN over conventional GAN has been demonstrated for SISR in [41] and [66]. Here we propose to leverage the idea of RaGAN to JDSR and demonstrate how relativistic discriminator can work with the proposed RDSEN (generator) for the purpose of perceptual optimization (overlooked in RDN [42] and RCAN [2]).

To implement RaGAN, we represent the real and fake images by x_r and x_f respectively; then

we can formulate the output of a modified discriminator $\hat{D}$ for RaGAN by:

$$\hat{D}(x_r) = \sigma(C(x_r) - \mathbb{E}_{x_f}[C(x_f)]) \tag{3.10}$$

$$\hat{D}(x_f) = \sigma(C(x_f) - \mathbb{E}_{x_r}[C(x_r)]) \tag{3.11}$$

where $\mathbb{E}_{x_f}$ and $\mathbb{E}_{x_r}$ are the expectation functions. It follows that the discriminator loss function L_D^{RaGAN} and adversarial loss function L_G^{RaGAN} can be written as:

$$L_D^{RaGAN} = -\mathbb{E}_{x_r}[\log(\hat{D}(x_r)] - \mathbb{E}_{x_f}[\log(1 - \hat{D}(x_f))] \tag{3.12}$$

$$L_G^{RaGAN} = -\mathbb{E}_{x_r}[\log(1 - \hat{D}(x_r))] - \mathbb{E}_{x_f}[\log(\hat{D}(x_f)] \tag{3.13}$$

It has been observed that the class of texture images is often more difficult for SISR due to spatial aliasing [66]. One way of achieving better texture reconstruction is through attention mechanism at the image level - i.e., to emphasize (i.e., increase the weight) difficult samples and overlook (i.e., down-weighting) easy ones. Such idea of weighting can be conveniently incorporated into the RaGAN package because the PyTorch implementation allows an optional weight input. More specifically, we propose to consider the following weighted function with a new hyperparameter γ tailored for *T*exture enhancement:

$$\begin{aligned} L_G^{TRaGAN} = &-\sum_i (\hat{D}(x_r))^{\gamma} \log(1 - \hat{D}(x_r)) \\ &-\sum_i (1 - \hat{D}(x_f))^{\gamma} \log(\hat{D}(x_f)) \end{aligned} \tag{3.14}$$

3.6.2 Perceptual Loss Function

We have implemented the following perceptual loss function based on [5, 41, 66, 103]. With a pre-trained VGG19 model [76], we can extract high-level perceptual features of both HR and SR images from the 4-*th* convolutional layer of VGG19 before the activation function is applied. Inspired by [41], we propose to extract high-level features before the activation function layer because it can further improve the performance. Let's define perceptual loss as L_{vgg} and L_1-norm distance as L_1. Then the total loss for our generator L_G can be formulated as follows:

$$L_G = L_{vgg} + \lambda_1 L_G^{TRaGAN} + \lambda_2 L_1 \tag{3.15}$$

where coefficients λ_1 and λ_2 are used to balance different loss terms. The term of L_{vgg} denotes $L_{vgg} = \Phi(f(SR), f(HR))$. Φ denotes the mean-squared error function (MSE), $f(SR)$ and $f(HR)$

Method	Scale	Set5	Set14	B100	Manga109	McM	PhotoCD
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
FlexIPS [1]+RCAN [2]	x2	35.18/0.9387	31.24/0.8776	31.00/0.8647	30.32/0.9199	34.80/0.9301	43.02/0.9610
DemoNet [6]+RCAN [2]	x2	35.92/0.9458	32.27/0.8971	31.38/0.8823	35.50/0.9590	35.34/0.9362	43.53/0.9642
RDSR [43]	x2	36.29/0.9485	32.56/0.9008	31.56/0.8850	36.14/0.9625	35.90/0.9423	43.74/0.9655
RCAN [2]	x2	36.54/0.9499	32.74/0.9032	31.68/0.8878	36.65/0.9643	36.18/0.9445	43.91/0.9661
DSERN (ours)	x2	36.55/0.9500	32.71/0.9031	31.70/0.8879	36.72/0.9652	36.23/0.9448	43.90/0.9661
DSERN+ (ours)	x2	36.62/0.9504	32.80/0.9041	31.73/0.8884	**36.89**/0.9658	36.33/0.9456	43.92/0.9661
RDSEN (ours)	x2	**37.40/0.9575**	**32.91/0.9128**	**32.00/0.8972**	36.86/**0.9716**	**37.38/0.9565**	**44.70/0.9716**
FlexISP+RCAN	x3	31.21/0.8731	28.55/0.7884	27.31/0.7310	27.58/0.8647	31.25/0.8661	40.32/0.9402
DemoNet+RCAN	x3	32.16/0.9030	29.24/0.8137	28.42/0.7801	30.75/0.9112	31.65/0.8739	40.74/0.9445
RDSR	x3	33.05/0.9103	29.54/0.8211	28.61/0.7859	31.69/0.9225	32.21/0.8842	40.90/0.9458
RCAN	x3	33.24/0.9125	29.67/0.8241	28.69/0.7882	32.06/0.9267	32.42/0.8874	41.11/0.9469
DSERN (ours)	x3	33.27/0.9127	29.67/0.8240	28.70/0.7884	32.06/0.9268	32.41/0.8875	41.10/0.9469
DSERN+ (ours)	x3	33.35/0.9134	29.73/0.8251	28.74/0.7892	**32.27**/0.9286	32.52/0.8888	41.13/0.9471
RDSEN (ours)	x3	**33.75/0.9218**	**29.91/0.8337**	**28.84/0.7993**	32.14/**0.9330**	**33.21/0.9032**	**41.60/0.9521**
FlexISP+RCAN	x4	29.57/0.8376	26.94/0.7177	26.68/0.6896	26.69/0.8427	27.78/0.7651	38.28/0.9201
DemoNet+RCAN	x4	30.33/0.8596	27.58/0.7488	26.94/0.7081	27.81/0.8590	29.49/0.8187	38.67/0.9243
RDSR	x4	30.87/0.8712	27.91/0.7589	27.16/0.7151	28.86/0.8800	30.10/0.8328	38.81/0.9258
RCAN	x4	31.04/0.8746	27.98/0.7613	27.20/0.7175	29.12/0.8856	30.24/0.8367	39.01/0.9271
DSERN (ours)	x4	31.02/0.8747	27.99/0.7620	27.20/0.7177	29.13/0.8857	30.25/0.8368	39.02/0.9273
DSERN+ (ours)	x4	31.12/0.8761	28.06/0.7635	27.24/0.7188	**29.36**/0.8888	30.35/0.8387	39.08/0.9277
RDSEN (ours)	x4	**31.63/0.8863**	**28.26/0.7725**	**27.39/0.7284**	29.28/**0.8903**	**30.74/0.8523**	**39.41/0.9317**

Table 3.3: PSNR/SSIM comparison among different competing methods. **Bold** font indicates the best result and underline the second best. Note that DSERN and DSERN+ are the results of JDSR V1.

are the high-level features extracted from VGG19-54 layer. Note that although similar loss functions were considered in previous studies including [41] and [66], their experiments include synthetic LR images only. In this dissertation, we will demonstrate the effectiveness of the proposed perceptual optimization for JDSR on real-world data next.

3.7 Experimental results

3.7.1 Implementation details

In our proposed RDSEN networks, we set the number of RDSEB blocks as 16; and each block includes 6 residual-dense SE modules. Most kernel size of Conv layers is 3×3 with 64 filters ($C = 64$) except those described in particular: the Conv layers in CA modules and Conv layers marked as '1×1' with a 1×1 kernel size. The reduction ratio is $r = 16$. The upscale module we have used is the same as [100]. The last layer filter is set to 3 in order to output super-resolved color images. For the discriminator setting, we have implemented the same discriminator network structure as SRGAN [5]. All kernel size of Conv layers is 3×3.

PyTorch implementation is still used in this experiment. Note that we configured all training setup the same as Sec. 3.4.1 for generator training. To train GAN-based networks, we have used the trained RDSEN to initialize the generator of GAN to get a better initial SR image for discriminator. The same learning rate and decay strategies are adopted here. λ_1 and λ_2 in Eq. (3.15) are set to 5×10^{-3} and 1×10^{-2} respectively as [41].

3.7.2 Training Dataset

In this experiment, we still used DIV2K dataset [81] as the training set, which includes 800 images (2K resolution). For testing, we have evaluated both popular image super-resolution benchmark datasets including Set5 [32], Set14 [94], B100 [95], and Manga109 [97], and popular image demosaicing datasets such as McMaster [98] and Kodak PhotoCD. To pre-process training and testing data, we downsample original HR images by a factor of $2\times$, $3\times$, $4\times$ using Bicubic interpolation then generate the 'RGGB' Bayer pattern. All experiments are implemented using PyTorch framework [84] and trained on NVIDIA Titan Xp GPUs. As an indicator of the overall computational complexity, the training time of our RDSEN lies somewhere between that of RDN [42] and RCAN [2] as shown in Table. 3.4. We have verified for all competing networks, it takes around 1000 epochs to reach the convergence.

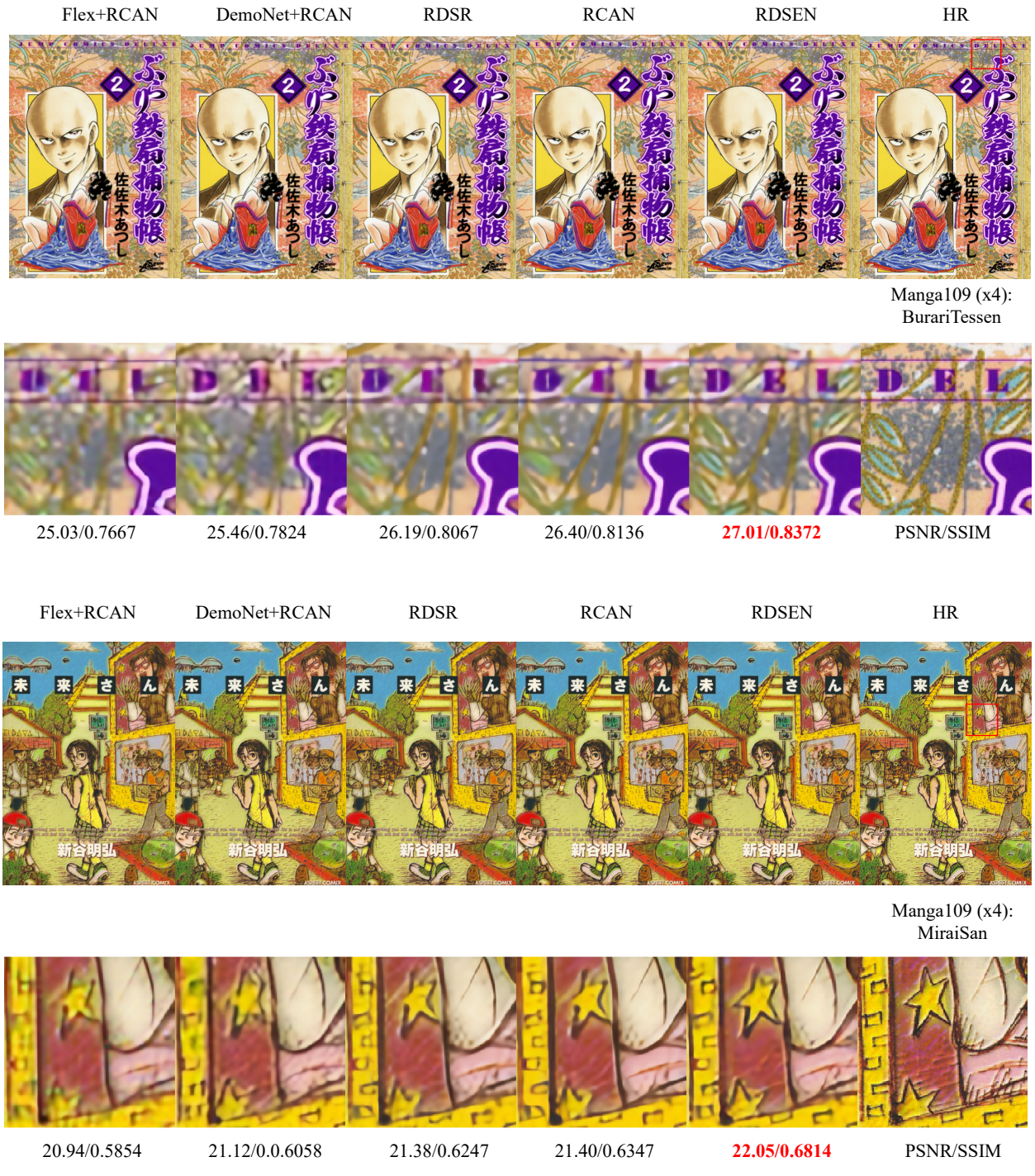

Figure 3.15: Visual results among competing approaches for Manga109 dataset at a scaling factor of 4.

Method	RDN	RCAN	RDSEN
Time (s)	128	160	130

Table 3.4: Training time compassion of RCAN, RDN and proposed RDSEN, per epoch.

Figure 3.16: Visual results among competing approaches for McM atasets at a scaling factor of 2 and 3.

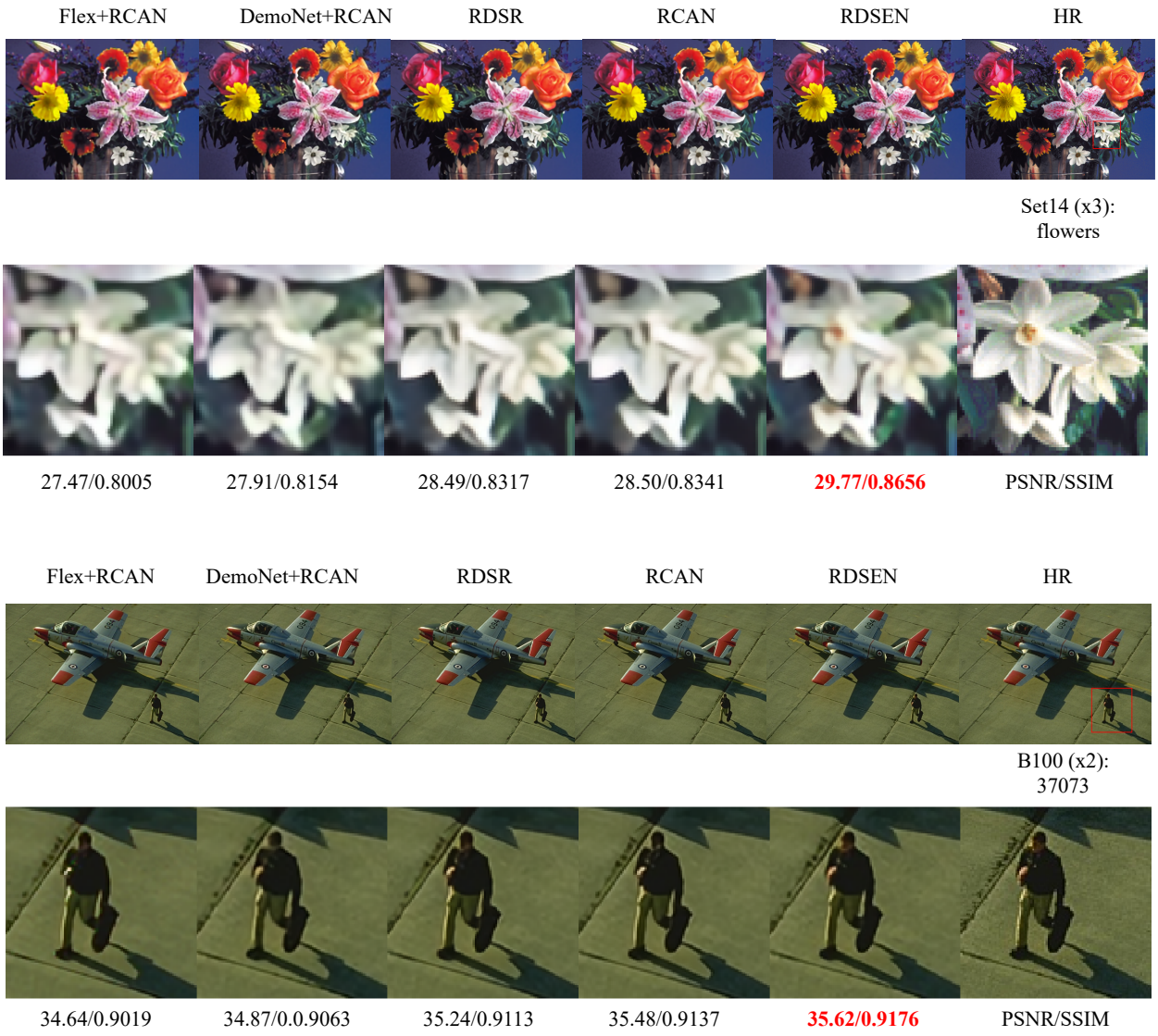

Figure 3.17: Visual results among competing approaches for Set14 and B100 datasets at a scaling factor of 3 and 2.

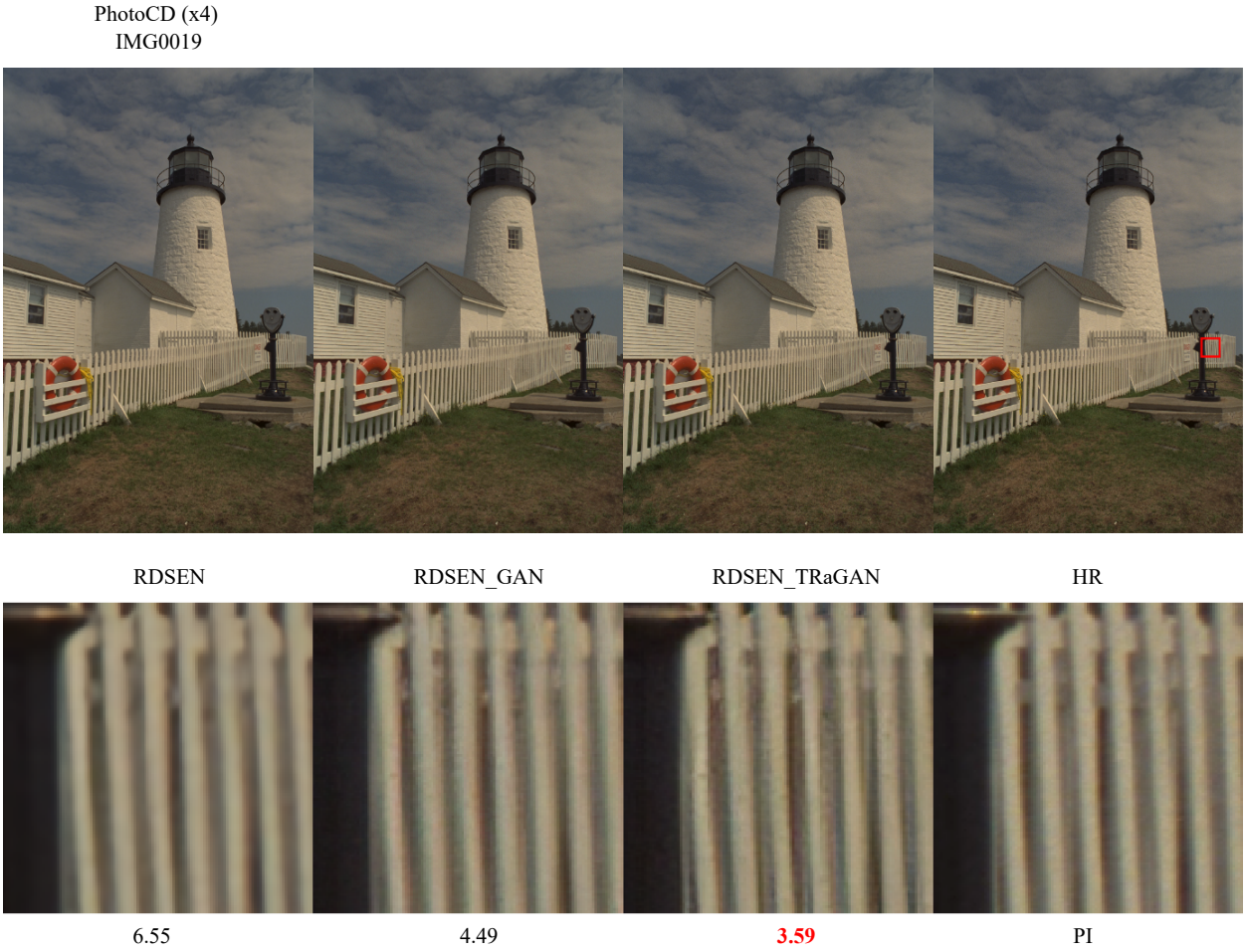

Figure 3.18: Visual comparison results among competing approaches for PhotoCD dataset at a scaling factor of 4.

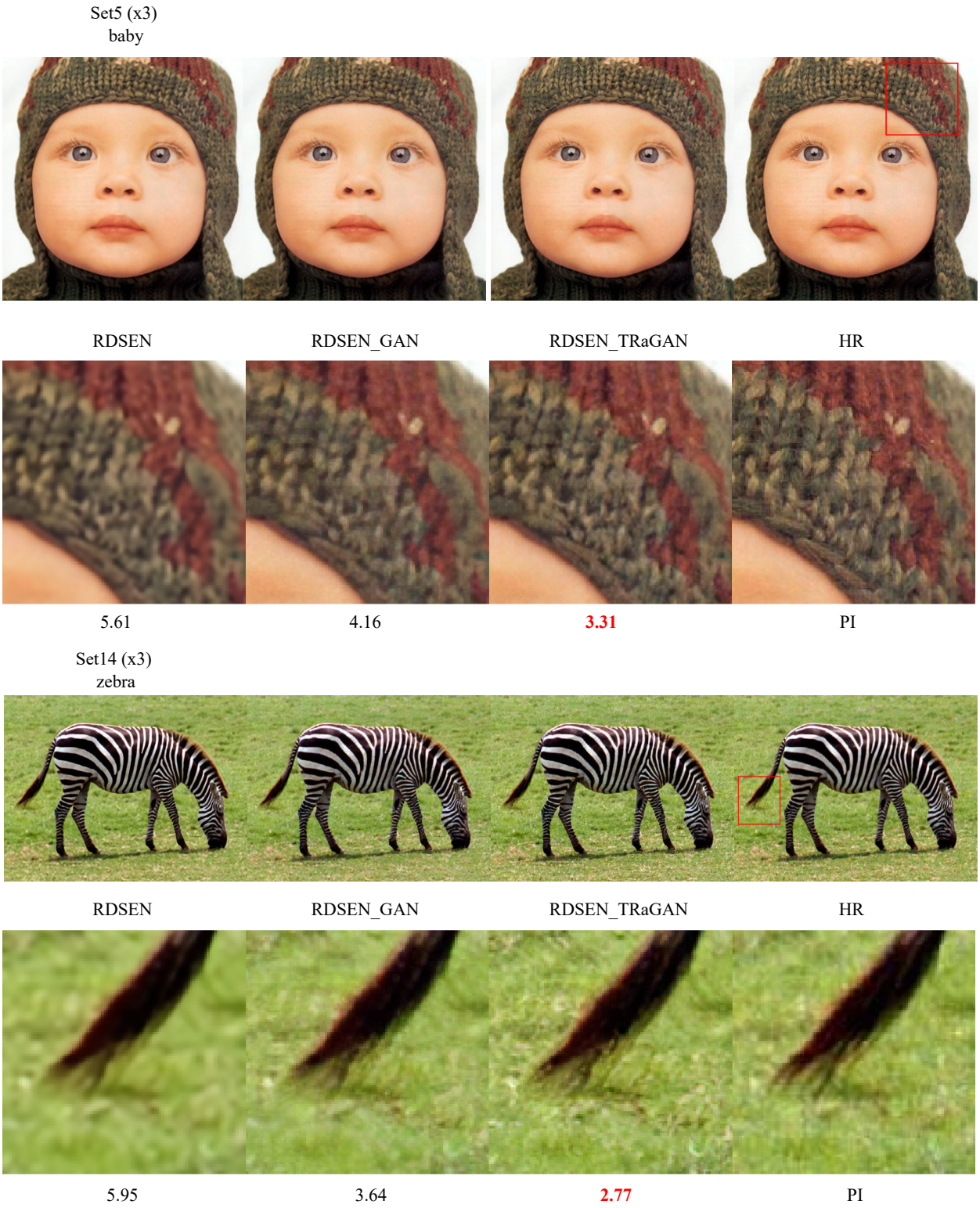

Figure 3.19: Visual comparison results among competing approaches for Set5 and Set14 datasets at a scaling factor of 3.

3.7.3 PSNR/SSIM Comparisons

We have compared our methods against four benchmark methods: separated (brute-force) approaches Flex [1] + RCAN [2] and DemoNet [6] + RCAN [2], recently published literature RDSR [43], and state-of-the-art SR approach RCAN [2]. To evaluate the results of DemoNet [6] + RCAN [2] approach, we first demosaiced the LR mosaiced images by using a pre-trained demosaicing network DemoNet to get LR color images, then super-resolved them by applying a pre-trained RCAN model. Note that we have used the pre-trained DemoNet and RCAN weights provided by the authors on GitHub.

The quantitative results are shown in Table 3.3 with the scaling factors of $2\times$, $3\times$ and $4\times$. It can be easily observed that our RDSEN method perform the best for most of datasets and scale factors. We observe that significant PSNR/SSIM gains (up to $1.2dB$) over previous state-of-the-art. Here we need to point that, not like DSERN, the upgraded version of RDSEN method did not apply any ensemble strategies, that is why on Manga109 dataset, RDSEN is not the best approach. As mentioned in Sec. 3.4.2, PSNR/SSIM metrics cannot always reflect the real visual quality of images, we have also included the subjective quality comparison results for images "BurariTessen" and "MiraiSan" in Fig. 3.15. For the first row of Fig. 3.15, it can be readily observed that for the top of the letters, only our RDSEN can faithfully recover text details; brute-force approaches (Flex+RCAN and DemoNet+RCAN), RDSR and RCAN have produced severe blurring artifacts; for the second row, only our method can reconstruct the yellow stars faithfully. Taking another example, Fig. 3.16 shows the comparison at two other scaling factors ($3\times$ and $2\times$). For "McM(x3)_16", we observe that all approaches contain color artifacts between the flower and grass, but our RDSEN method can recover more realistic details than other competing approaches; for "McM(x2)_1", pattern recovered by RDSEN appears to have the highest quality and most detailed textures. For more visual comparison, see Fig. 3.17 which shows more convincing visual comparison among various competing approaches (please zoom in for detailed evaluation).

3.7.4 Perceptual Index (PI) Comparisons

Most recently, a new objective metric called Perceptual Index (PI) [82] has been developed for perceptual SISR (e.g., the 2018 PIRM Challenge [86]). The PI score is defined by

$$\mathrm{PI} = \frac{1}{2}((10 - \mathrm{MA}) + \mathrm{NIQE}) \tag{3.16}$$

where MA denotes a no-reference quality metric [87] and NIQE referred to Natural Image Quality Evaluator [88]. Note that the lower PI score, the better perceptual quality (i.e., contrary to SSIM metric [75]). Objective comparison of competing JDSR methods in terms of PI is shown in Table 3.5. We have observed that GAN-based methods produce the lowest PI scores for all datasets and scaling factors. Fig. 3.18 provides the visual comparison with image "IMG0019" ($4\times$). It can be observed that GAN-based methods can recover sharper edges and overcome the issue of over-smoothed regions. Additionally, TRaGAN is capable of achieving even lower PI scores than the standard GAN. Fig. 3.19 shows two more results to demonstrate the advanced ability to recover texture details of GAN based methods, especially of TRaGAN.

3.7.5 Challenging dataset evaluation

In [6], the authors argue that the existing benchmark datasets are lacking of challenge. To make the demosaicing performance convincing, they proposed challenging patches where they detected and cropped the challenging patches from web images for image demosaicing tasks.

In our tasks, we first generated the Bayer patterns, then tested them with scale factor of 4 (demosaiced and super-resovled by 4 $\times$). From Fig. 3.20, we can see that comparing with the separated method which have severe blurring and aliasing issues, our RDSEN can better reconstruct both the textures and edges and avoids aliasing issues.

3.7.6 Ablation Studies

Similar to Sec. 3.4.3, we conduct another ablation study shown in Table 3.6 to demonstrate the effect of proposed RDSEB module. We can find that our RDSRN has the best performance on all benchmark dataset.

Methods	Scale	Set5	Set14	B100	Manga109	McM	PhotoCD
FlexISP [1]+RCAN [2]	x2	4.16	4.14	3.34	4.97	3.51	5.42
DemoNet [6]+RCAN [2]	x2	4.13	3.76	3.31	3.99	3.48	5.59
RDSEN (ours)	x2	4.17	3.81	3.28	4.07	3.27	5.65
RDSEN_GAN (ours)	x2	3.41	2.95	**2.34**	3.53	2.59	4.85
RDSEN_TRaGAN (ours)	x2	**3.06**	**2.90**	2.35	**3.45**	**2.52**	**4.72**
FlexISP [1]+RCAN [2]	x3	6.98	5.70	6.18	5.43	5.14	6.42
DemoNet+RCAN	x3	6.31	5.18	4.97	4.63	5.19	6.61
RDSEN (ours)	x3	5.71	4.74	4.48	4.53	4.57	6.52
RDSEN_GAN (ours)	x3	3.78	2.94	2.39	3.44	2.60	4.96
RDSEN_TRaGAN (ours)	x3	**3.58**	**2.81**	**2.36**	**3.37**	**2.44**	**4.78**
FlexISP [1]+RCAN [2]	x4	7.42	6.63	6.30	5.28	7.15	6.88
DemoNet+RCAN	x4	7.21	6.23	6.28	5.43	6.22	7.04
RDSEN (ours)	x4	6.18	5.94	5.92	5.00	5.68	6.87
RDSEN_GAN (ours)	x4	4.50	3.31	2.84	3.65	2.84	5.01
RDSEN_TRaGAN (ours)	x4	**4.24**	**3.11**	**2.55**	**3.45**	**2.72**	**4.44**

Table 3.5: Objective performance comparison among different methods in terms of Perceptual Index (the lower the better). **Bold** indicates the best result and underline the second best.

Method	Scale	Set5	Set14	B100	Manga109	McM
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
ResNet	x2	36.48/0.9498	32.71/0.9030	31.67/0.8876	36.48/0.9642	36.11/0.9443
RCAN	x2	36.54/0.9499	32.74/0.9032	31.68/0.8878	36.65/0.9643	36.18/0.9445
RDSEN (ours)	x2	**37.40/0.9575**	**32.91/0.9128**	**32.00/0.8972**	**36.86/0.9716**	**37.38/0.9565**

Table 3.6: Ablation study for ResNet, ResNet with CA (RCAN) and ResNet with proposed RDSEN. **Bold** font indicates the best result.

Figure 3.20: Visual quality comparison of JDSR results among challenging patches provided by [6].

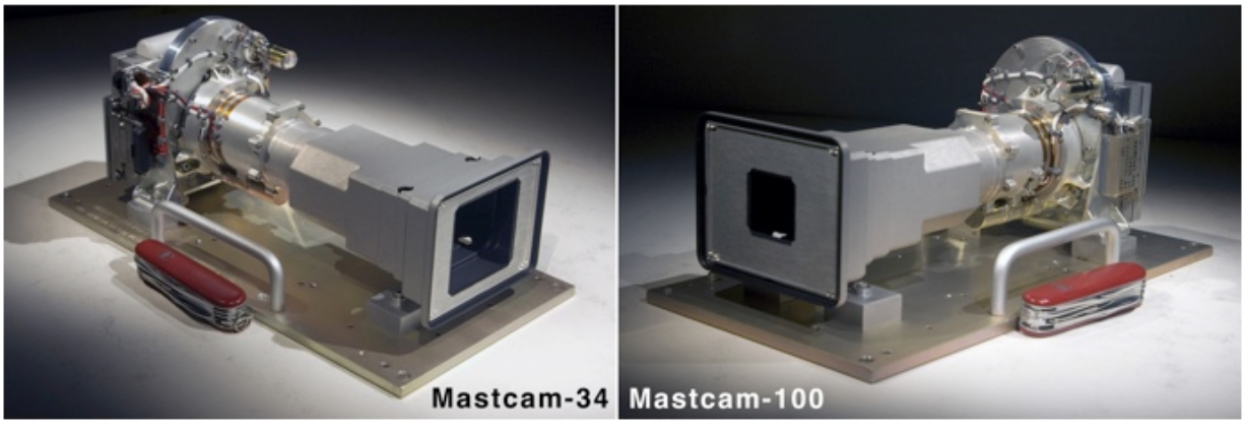

Figure 3.21: Fixed-focal length Mastcams [7].

3.7.7 Performance on the Real-world Data

Finally, we have tested our proposed JDSR technique on some real-world data collected by the Mastcam of NASA Mars Curiosity (as shown in Fig. 3.21). The raw data are 'RGGB' Bayer pattern sized by 1600×1200. Due to hardware constraints, the left camera and the right camera of Mastcam have different focal lengths (the left is about 3 times weaker than the right). To compensate such a "lazy-eye" effect on raw Bayer patterns, it is desirable to develop a joint demosaicking and SR technique with at least a scaling factor of 3 (in order to support high-level standard stereo-based vision tasks such as 3D reconstruction and object recognition). Our proposed JDSR algorithm is a perfect fit for this task, which shows the great potential of computer vision and deep learning in deep space exploration.

The visual comparison results are shown in Fig. 3.22 for a scaling factor of 4. It can be seen that the brute-force approach (Flex+RCAN) suffers from undesired artifacts especially around the edge of rocks. Our proposed RDSEN method can overcome this difficulty but the results appear over-smoothed. RDSEN_GAN improves the visual quality to some degree - e.g., more fine details are present and sharper edges can be observed. Replacing GAN by TRaGAN can further improve the visual quality not only around the textured regions (e.g., roads and rocks) but also in the background (e.g., terrain appears visually clearer and sharper). Let's take two more visual results, Fig. 3.23 and Fig. 3.24 among Flex+RCAN, RDSEN, RDSEN_GAN and RDSEN_TRaGAN approaches, the proposed RDSEN_TRaGAN always lead to the best visual qualities.

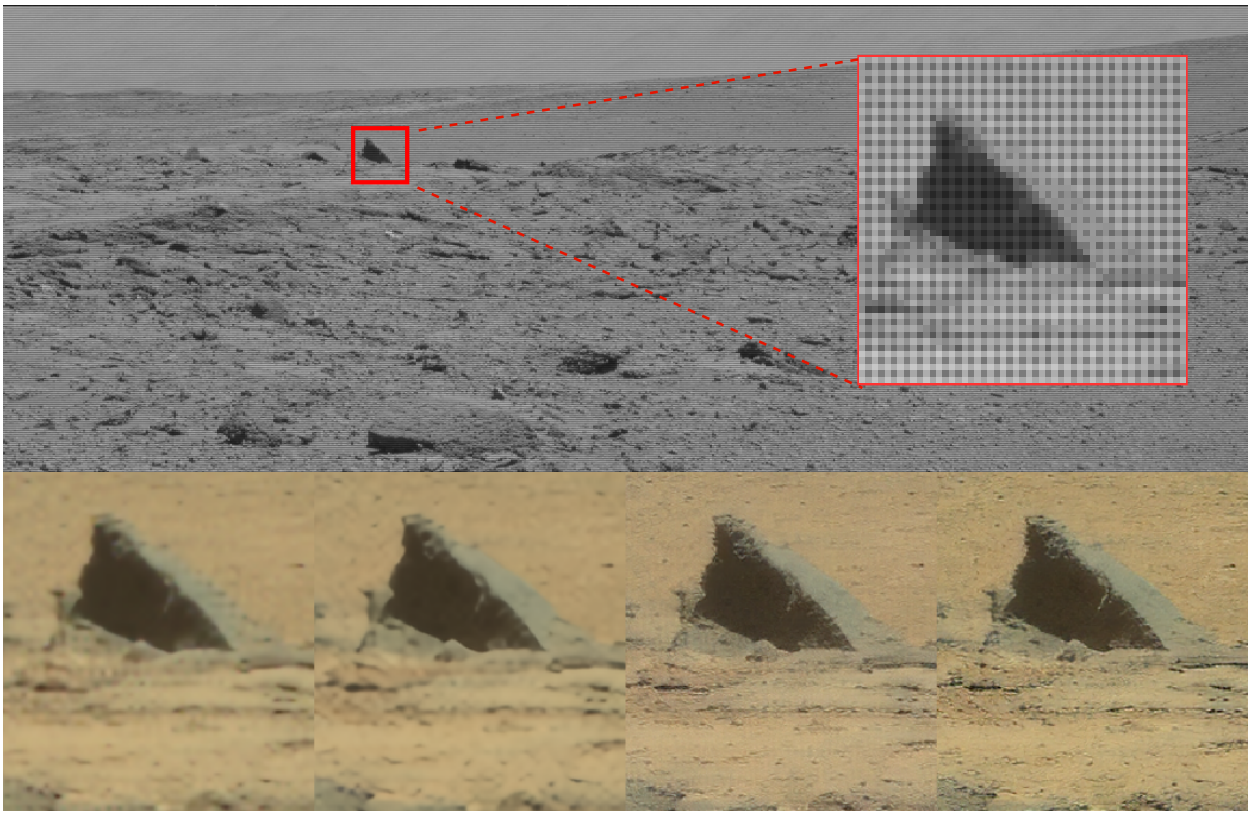

Figure 3.22: Visual quality comparison of JDSR results on real-world Bayer pattern collected by NASA Mars Curiosity (4×).

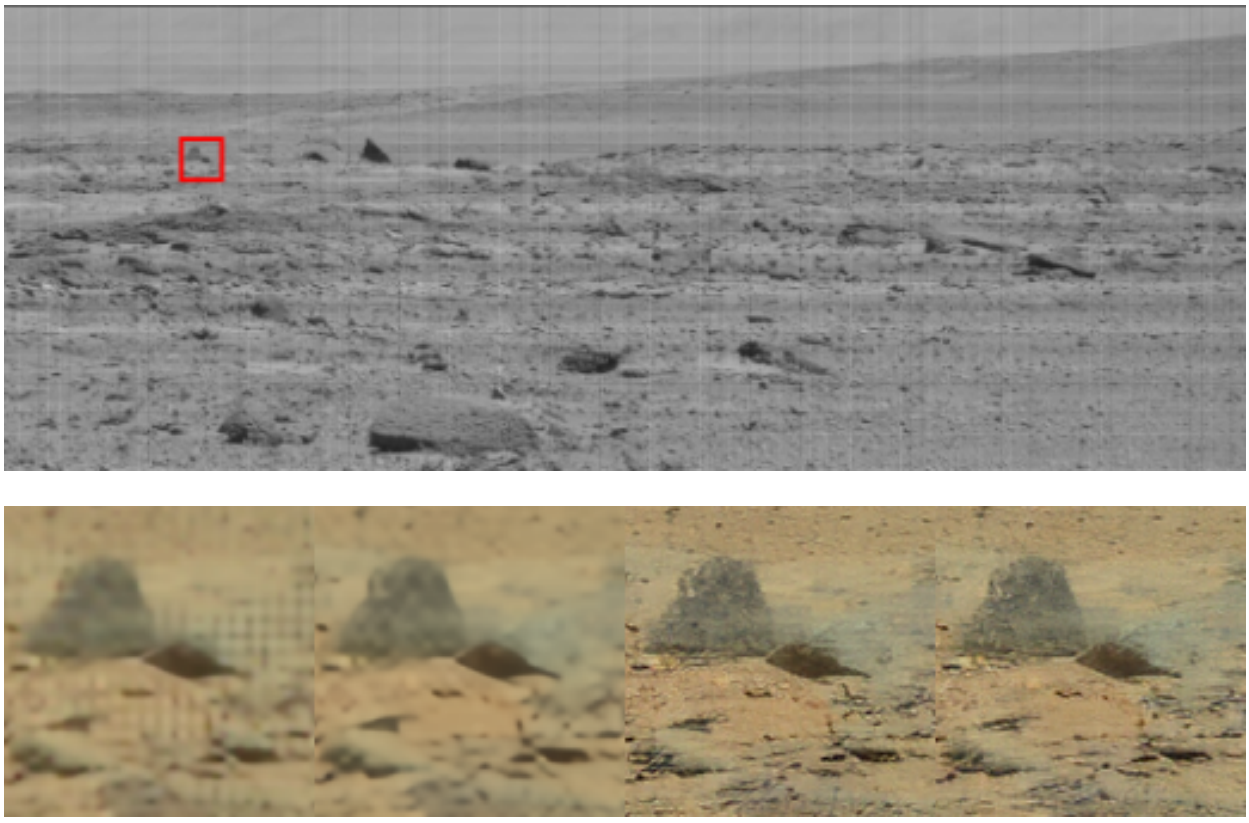

Figure 3.23: More visual quality comparison of JDSR results on real-world Bayer pattern collected by NASA Mars Curiosity.

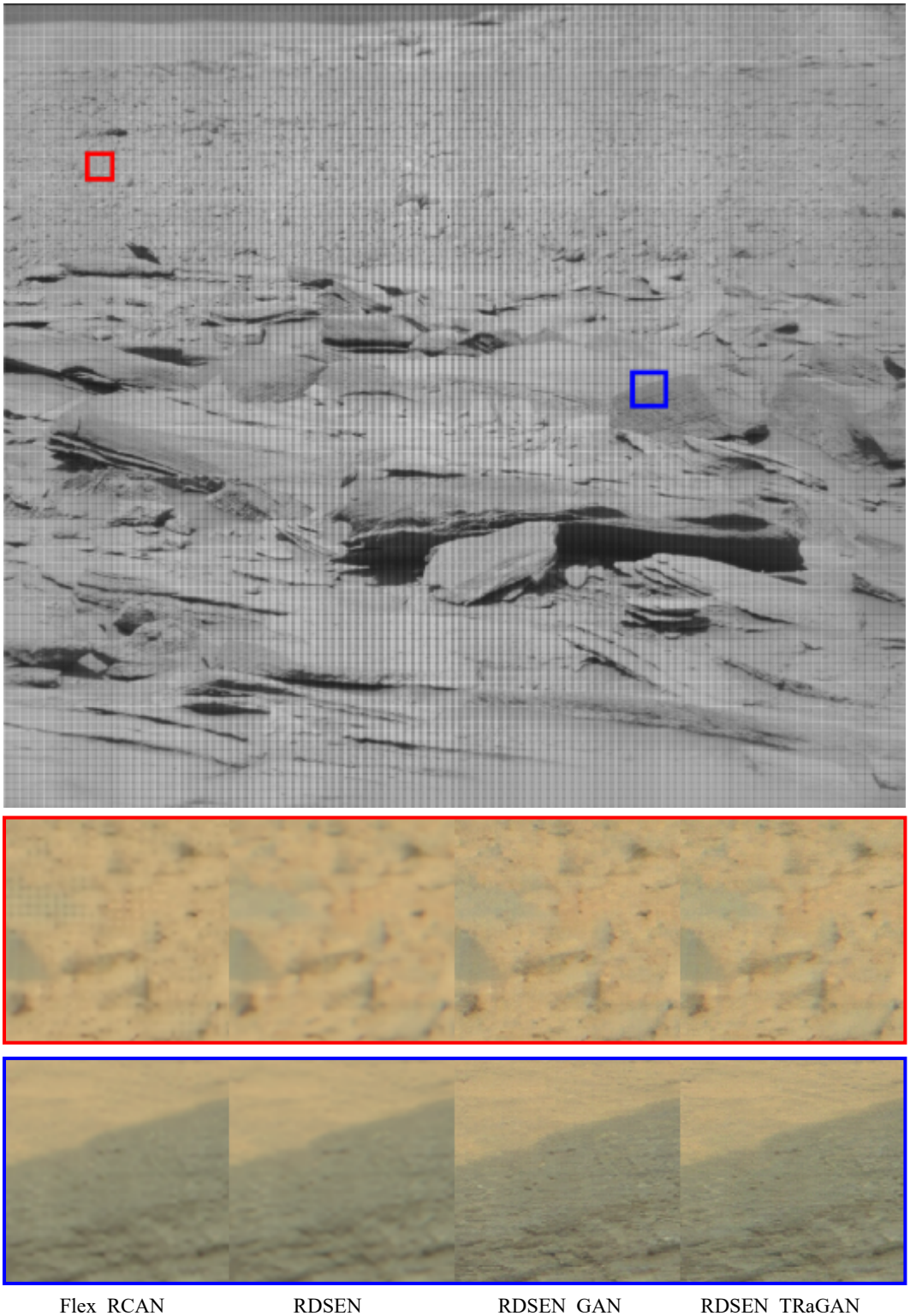

Figure 3.24: More visual quality comparison of JDSR results on real-world Bayer pattern collected by NASA Mars Curiosity.

3.8 Summary

In this chapter, we proposed two networks to study JDSR problem, DSERN and RDSEN. The proposed DSERN includes the Dense Squeeze-and-Excitation (DSE) block that can smooth the information flow by the channel descriptor. As an upgraded version, RDSEN can not only improve the both objective and subjective performance, but also has lower complexity compared with DSERN. Both frameworks can get better subjective and objective results compared with state-of-the-art approaches.

Chapter 4

Video Super-Resolution

4.1 Related Work

Different from image super-resolution which only needs to care about reconstructing spatial domain, video super-resolution regards to not only reconstruct the high-frequency components (spatial domain) but also need to consider the motion offset between the adjacent frames (temporal domain). In this section, we review video super-resolution approaches based on multi-frame such as [53, 56–58, 104, 105], optical flow [50] alignment and deformable convolution [55] alignment.

4.1.1 Video Super-Resolution

As the pioneers to apply optical-flow to VSR problems in order to utilize temporal and spatial information, Liao *et al.* [106] introduced a draft-ensemble strategy to use two robust optical flow methods, TV-l_1 flow and the motion detail preserving (MDP) flow to overcome large motion variation and then combine SR drafts via the deep convolutional neural network to generate the final SR result. Kappeler *et al.* [107] proposed to use optical-flow to estimate motion compensation of consecutive LR frames and wrapped them as inputs of the CNN to generate SR frames. Those two-stage approaches are not optimal solutions since they separate the motion compensation from frame reconstruction. To explore potential benefits of end-to-end learning architecture for VSR problem, Caballero *et al.* [53] proposed efficient sub-pixel convolutional neural network (ESPCN), a novel end-to-end deep CNN to jointly train the estimation of optical flow and spatio-temporal networks. Tao *et al.* [108] introduced a new layer called sub-pixel motion compensation (SPMC) to

handle inter-frame motion alignment, applied a ConvLSTM [109] architecture for reconstruction and testing. Haris *et al.* [110] proposed a recurrent back-projection network (RBPN) with encoder-decoder mechanism to extract spatial and temporal information. Most recently, Isobe *et al.* [105] proposed a novel temporal group attention (TGA) framework to group the input frames (7 frames) as three groups then generate temporal spatial attention maps to reconstruct the missing details in the reference frame. Xue *et al.* [111] proposed to learn self-supervised motion representation, task-oriented flow (TOFlow), instead of fixing optical flow as the motion compensation module for VSR problem. Jo *et al.* [112] represented a dynamic upsampling filters (DUF) to avoid the use of the explicit motion compensation by computing pixels of local spatio-temporal neighbors of LR frames to learn the implicit motion compensation.

4.1.2 Deformable Convolution

The inherent limitation of traditional CNN is the ability to model geometric transformations because of the fixed kernel shape. Although dilated convolution can alleviate this limitation, it is still difficult for normal kernels to find the key points on the input images or features. To solve this problem, Dai *et al.* [54, 55] first proposed a deformable convolution networks to improve the ability of modeling geometric transformations for traditional CNN by adding learnable offsets to acquire information from other field rather than fixed local area.

Deformable convolution networks have been widely used on high-level vision tasks such as object detection [113] and segmentation [54]. Inspired by [55], Tian *et al.* [57] firstly proposed a temporally-deformable alignment network (TDAN) to adapt deformable convolution to align the consecutive LR input frames at the feature level. Wang *et al.* [56] designed a more aggressive alignment approach, PCD align module, to align the neighboring LR frames at different scale levels; also they proposed a temporal and spatial attention fusion module to future enhance important features. Xiang *et al.* [104] proposed a novel space-time video super-resolution framework to utilize deformable convolution and deformable ConvLSTM module to achieve temporal and spatial super-resolution at the same time. Wang *et al.* [58] introduced another deformable convolution based VSR framework called deformable non-local network (DNLN) with non-local attention module and hierarchical feature fusion block to enhance the global details between neighboring

Method	Optical-flow	DConv.	DKern.	CA	SA	ConvLSTM
Liao *et al.* [106]	✓					
Kappeler *et al.* [107]	✓					
ESPCN [53]	✓					
SPMC [108]						✓
TGA [105]					✓	
TOFlow [111]	✓					
TDAN [57]		✓				
EDVR [56]		✓			✓	
Xiang *et al.* [104]		✓				✓
DNLN [58]		✓				
DKSAN(ours)		✓	✓	✓	✓	

Table 4.1: Previous work comparison on VSR. The "DConv.", "DKern.", "CA", "SA" and "ConvLSTM" respectively represent deformable convolution, deformable kernel, channel attention, spatial attention and convolutional LSTM.

frames and references. Those deformable alignment based methods show better performance than optical-flow based networks.

4.1.3 Summary of VSR approaches

Table. 2.1 presents a summary of the state-of-the-art deep learning methods for the VSR task.

4.2 Dataset Setup

In this work, we use Vid3oC [68] as the training dataset. Vid3oC dataset has a large diversity of contents and become more and more popular for video enhancement and super-resolution tasks. It includes 50 videos in total for training, 16 sequences with 120 frames each for validation and 16 sequences with 120 frames each for testing. Note that the ground-truth validation and testing data are not released. All the videos are 1080p with 30 frames per second. The length of video time is varied. The sample frames are shown in Fig. 4.1 and Fig. 4.2.

Because the authors haven't released the testing dataset, in the meantime, we use IntVID [68]

Figure 4.1: The sample of HR patches from Vid3oC dataset.

Figure 4.2: The sample of low-resolution patches from Vid3oC dataset.

dataset as an alternated test dataset to evaluate the performance of our proposed approach. IntVID dataset is released by the same authors of Vid3oC dataset that includes 60 videos for training, 16 videos for validation and 16 videos for testing. The same as Vid3oC, the ground-truth of validation and test videos are not released. Therefore, we select the last ten training videos and extract 14 consecutive frames from each video as test dataset.

To pre-process training and test data, we first extract the frames from the original videos via FFmpeg, then we crop 4 border pixels on top and bottom of each frame to make sure the cropped frame can be divided by 16. Finally, we downsample the cropped frames by a factor of 16× using Bicubic interpolation to generate LR frames. After the whole pre-process procedure, we have 69,496 LR frames in total and the corresponded 4th frame of HR frames (9,928 in total) for training, 140 frames for testing.

4.3 Approach

The hierarchy of DKSAN network design goes like: DKSAN (Fig. 4.3) → DKC_Align sub-network (Fig. 4.4)→ reconstruction module (Fig. 4.5).

4.3.1 Overview: Deformable Kernel Spatial Attention Networks

For multi-frame based video super-resolution, first, given a $2N+1$ consecutive LR frames $I_T^{LR} = \{I_{r-N}^{LR}, \ldots, I_{r-1}^{LR}, I_r^{LR}, I_{r+1}^{LR}, \ldots, I_{r+N}^{LR}\}$, where I_r^{LR} is denoted as the center frame or reference frame and I_{r-N}^{LR} or I_{r+N}^{LR} are the neighboring frames of I_r^{LR}. The goal of multi-frame based video super-resolution is to reconstruct HR frame $\hat{Y}_r$ from the consecutive sequence of I_T^{LR} by utilizing both the spatial and temporal information. The overall pipeline of proposed networks DKSAN is shown in Fig. 4.3. It mainly includes four parts, feature extractor, DKC_Align module, reconstruct module and upscale module. Different from the traditional deep learning based multi-frame video super-resolution architecture, this work aims to super-resolve the extreme LR videos (with the upscale factor of 16), therefore, it is difficult to upscale the extreme low-resolution feature maps to the target high-resolution one directly (one time upscale such as [56, 57, 105]), undesired blurring and artifacts may be generated in this procedure. To solve this issue, we introduce a cascade upscale solution to super-resolve the low-resolution features several times (four times in this work) to

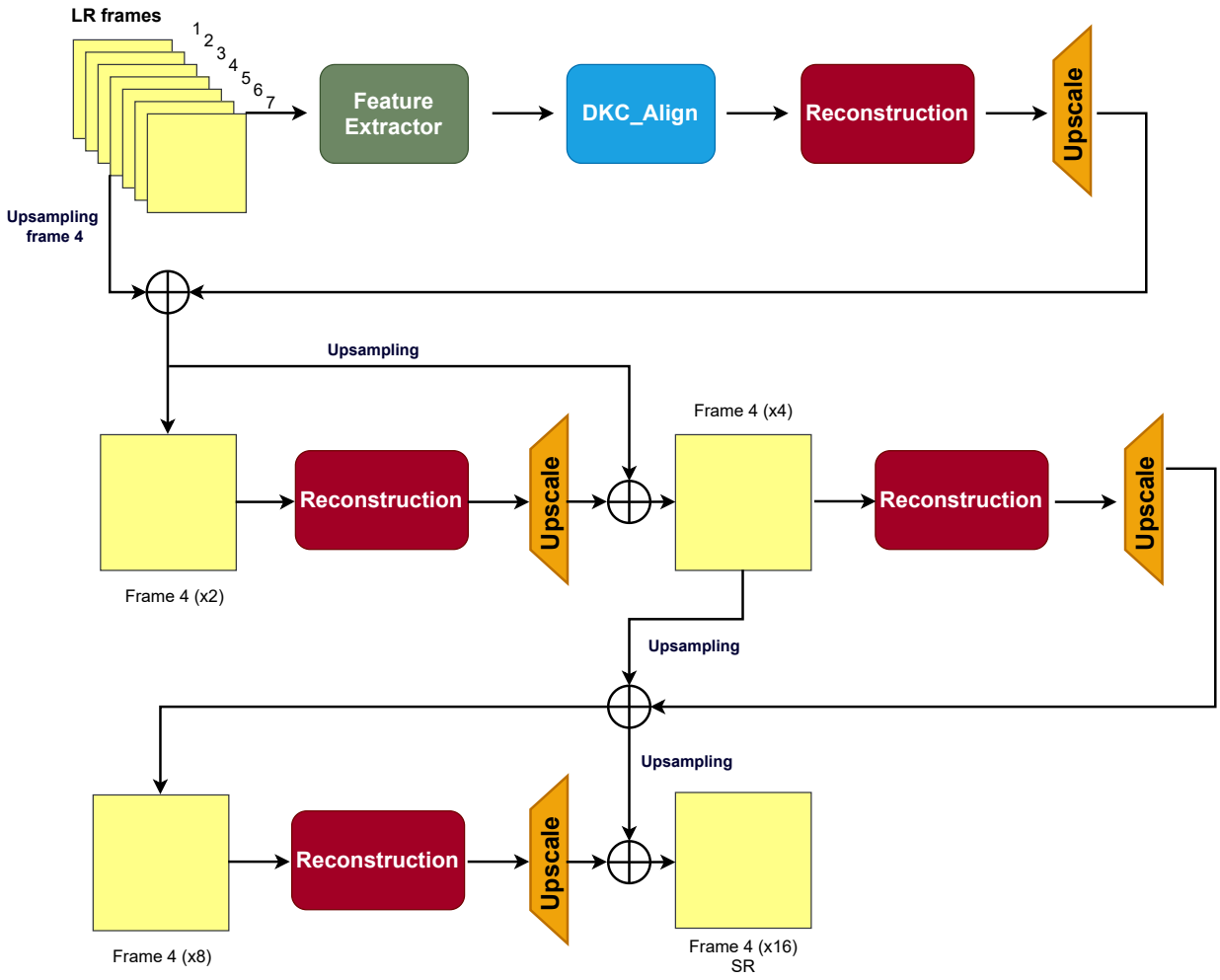

Figure 4.3: Overview of DKSAN.

smooth the information conduction in upscale module. Benefit by the cascade upscale architecture, the LR frames can be super-resolved smoothly to reconstruct more desirable HR frames compared with other approaches.

The whole problem can be formulated as follows:

$$\hat{Y}_r = \mathbb{F}(I_T^{LR}) \tag{4.1}$$

where I_T^{LR} denotes the consecutive LR frames and $\hat{Y}_r$ denotes the super-resolved reference frame I_r^{LR}. In particular, we extract the preliminary features of all input frames through the feature extractor which is stacked by several resblocks [56] without batch normalization layers. This procedure can be represented as follows:

$$F_{fea} = E_{res}(I_T^{LR}) \tag{4.2}$$

where E_{res} denotes the preliminary feature extractor, the output F_{fea} is the extracted feature maps for all input frames. Let define F_n is the neighboring feature, and F_r is the reference feature separated from F_{fea}. To align the neighboring feature and the reference feature with the proposed DKC_Align module E_{DKC_Align}, we have

$$F_{Align} = E_{DKC\_Align}(F_n, F_r) \tag{4.3}$$

$$F_{fusion} = \mathbf{W}_E(F_{Align}) \tag{4.4}$$

where $n \in [t-N, t+N]$ and $n \neq r$, F_{Align} is the concatenated aligned feature maps for each neighboring frame feature with reference frame feature. The details are demonstrated in section 4.3.2; $\mathbf{W}_E \in \mathbb{R}^{1\times1\times C}$ a 1×1 Conv layers. Then in the reconstruction period, the aligned feature F_{Align} is entered to reconstruction module and upscale module for the first level upscaling operation:

$$\hat{Y}_r^{level1} = U_1(E_{Recon1}(F_{fusion})) + B_{2\times}(I_r^{LR}) \tag{4.5}$$

where E_{Recon1} denotes the first level reconstruction module, U_1 is the first level upscaling module and $B_{2\times}$ stands for the Bicubic interpolation with scale factor of 2; $\hat{Y}_r^{level1}$ is the $2\times$ SR frames. Finally, to get the extreme super-resolved frame $\hat{Y}_r$, we repeat another 3 times of reconstruction

operation:

$$\hat{Y}_r^{level2} = U_2(E_{Recon2}(E_2(\hat{Y}_r^{level1})) + B_{2\times}(\hat{Y}_r^{level1}) \tag{4.6}$$

$$\hat{Y}_r^{level3} = U_3(E_{Recon3}(E_3(\hat{Y}_r^{level2})) + B_{2\times}(\hat{Y}_r^{level2}) \tag{4.7}$$

$$\hat{Y}_r = U_4(E_{Recon4}(E_4(\hat{Y}_r^{level3})) + B_{2\times}(\hat{Y}_r^{level3}) \tag{4.8}$$

where E_2, E_3, E_4 are the preliminary feature extractors for each level; U_2, U_3, U_4 denote the upscaling module for each corresponding level, respectively. The details are described in section 4.3.3 including DKSA module which is not discussed here.

4.3.2 Deformable Kernel Alignment Module

Different from the previous VSR works which applied optical flow to align neighboring frames with reference frame, [57] and [56] introduced to utilize modulated deformable convolution [55] to temporally align the given consecutive frames in order to add temporal information to VSR frameworks.

Deformable Convolution and Deformable Kernel

Inspired by [56, 59], we propose a new alignment module, DKC_Align, to combine the deformable kernel [59] and deformable convolution [55] as shown in Fig. 4.4. First, let F_n and F_n^{align} denote the input and output feature maps (not the reference frame feature), $\mathbf{W}_k$ represents the weight kernel and p_k is the pre-specified offsets for the k-th location (K is the total sampling location), then the modulated deformable convolution can be described as follows:

$$F_n^{align}(p) = \sum_{k \in K} \mathbf{W}_k \cdot F_n(p + p_k + \Delta p_k) \cdot \Delta m_k \tag{4.9}$$

where $F_n^{align}(p)$ and $F_n(p)$ indicates the feature location p from F^{align} and F_n, Δp_k and Δm_k stand for the learnable offset and the modulation scalar, respectively. With Δp_k and Δm_k, the convolution will get the ability to be irregularly dilated. To get Δp_k and Δm_k and align the neighboring feature with reference feature in particular, we first concatenate the neighboring frame feature and the reference frame feature then fuse them with one Conv2D layer and fed them into several deformable kernel layers:

$$\Delta P_n, \Delta M_n = \mathbb{D}(f([F_n, F_r])), n \in [t-N, t+N], n \neq r \tag{4.10}$$

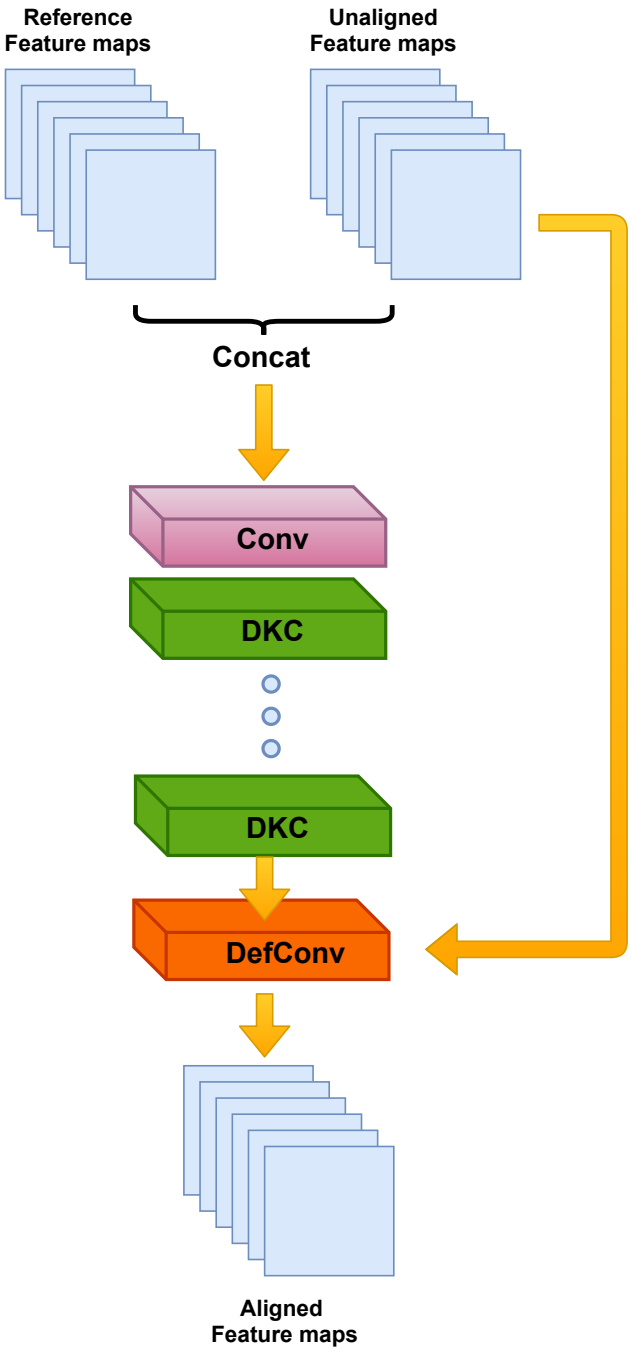

Figure 4.4: Overview of DKC_Align module.

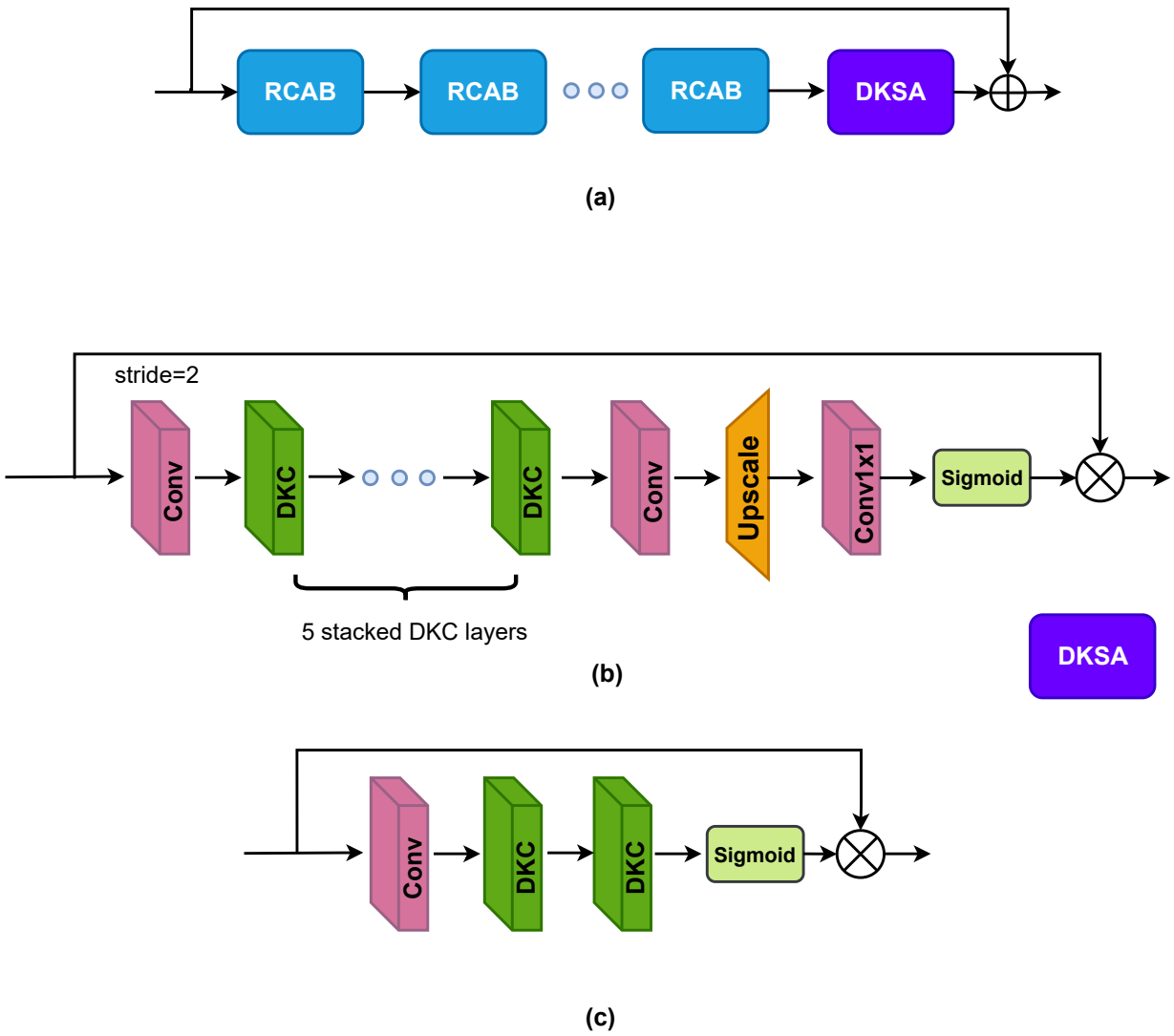

Figure 4.5: Overview of reconstruction module; DKCA is deformable kernel spatial attention module shown in (b); (c) is a light version of DKCA which is applied to the first level reconstruction.

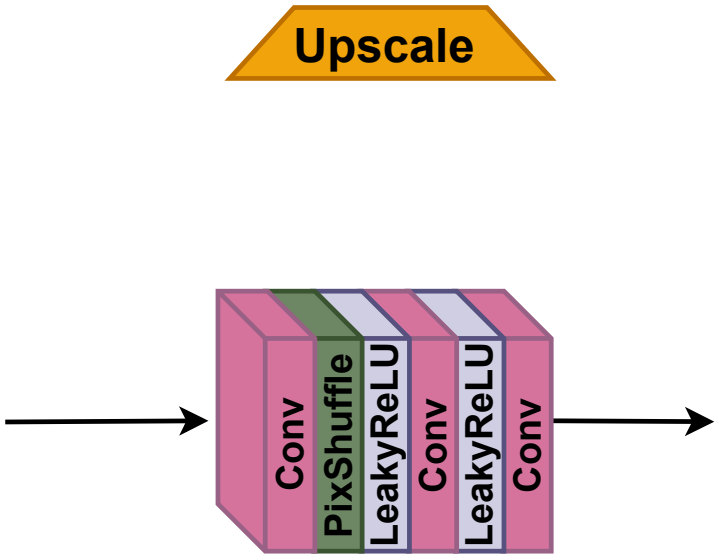

Figure 4.6: The details of upscale module, the last Conv layer has only 3 feature maps output in order to generate RGB color frame.

where f denotes the one Conv2d layer to fuse F_n and F_r, $\mathbb{D}$ represents the deformable kernel convolution layer. To express deformable kernel convolution layer $\mathbb{D}$, let Δk denote a learnable offset of the kernel $\mathbf{W}$, then deformable kernel convolution layer can be formulated as:

$$\mathbb{D} = \sum_{k \in K} \mathbf{W}_{k+\Delta k} \cdot f([F_n, F_r])(p + p_k + \Delta p_k), n \in [t-N, t+N], n \neq r \tag{4.11}$$

The newly designed DKC_Align module can better adapt both the theoretical and effective receptive fields though deformable convolution and deformable kernel to explore the important offset feature to improve the alignment accuracy.

4.3.3 Reconstruction Module

To get the super-resolved frame $\hat{Y}_r$, the output F_{fusion} from DKC_Align module is fed into the reconstruction module. The reconstruction module includes several stacked RCAB blocks and the DKSA module (see Fig. 4.5 (a)):

$$F_{recon} = E_{DKSA}(E_{RCABs}(F_{fusion})) + F_{fusion} \tag{4.12}$$

where F_{recon} is the final reconstruction features to be fed into upscale module (the architecture of upscale module is shown in Fig. 4.6 which includes several Conv layers, PixelShuffle and

LeakyReLU), E_{RCABs} and E_{DKSA} denote the RCAB blocks and DKSA module. As discussed in Chapter 3, attention mechanism has been proven as an efficient way to help high and low-level vision problems. In our proposed reconstruction module (see Fig. 4.5), we implemented both channel attention and spatial attention (Fig. 4.5 (b)) modules. Please refer Chapter 2 and Chapter 3 for more details of RCAB block and channel attention mechanism.

Deformable Kernel Spatial Attention Module

In order to further calibrate output feature maps, we proposed to use Deformable Kernel based Spatial Attention (DKSA) module instead of traditional spatial attention mechanism. As shown in Fig. 4.5 (b), in DKSA, we first use one Conv layer to extract the output of the stacked RCAB blocks, then we use a couple of stacked Deformable Kernel Convolution (DKC) layers to further extract key features from the naive feature map. As discussed in section 4.3.2, deformable kernel can better measure the effective receptive field than normal convolution kernel. Therefore, DKSA can generate better spatial attention map to let networks pay more attention on important features such as edges and textures than the traditional one. Note that Fig. 4.5 (c) shows a light version of DKSA which is used in the level 1 reconstruction module.

4.4 Experimental Results

In this section, we demonstrate the network setting, training details, ablation study and experimental results of proposed video extreme super-resolution problem.

4.4.1 Implementation Details

In the proposed DKSAN networks, to compare with EDVR, we set the kernel size as 3×3 for most of Conv layers, all deformable kernel layers and all deformable convolution layers with 128 filters. The kernel size of feature fusion layers is 1×1. The reduction ratio for channel attention module is still $r = 16$. The upscale module is PixelShuffle layer [100]. The last layer filter is set to 3 in order to output color frames.

In particular, we randomly crop the 7 LR frames as small patches with the size of 32×32, and crop the corresponding 4th HR frames with the size of 512×512. The batch size is 16. We

augment the training set by random flips and rotations. The optimizer we used is ADAM [83] with $\beta_1 = 0.9$, $\beta_2 = 0.999$. The initial learning rate is set to 4×10^{-4}. The total training step is $115k$. The loss function we used is adapted Charbonnier penalty function [69]. The loss can be defined as 4.13 shown as follows:

$$Loss = \sqrt{||\hat{Y}_r - Y_r||^2 + \xi^2} \tag{4.13}$$

where $\xi = 1e-3$, $\hat{Y}_r$ is super-resolved frame and Y_r is target frame (ground-truth). All experiments are trained on 4 NVIDIA Titan Xp GPUs with PyTorch framework Implementation.

Note that for fair comparison, we retrain EDVR with same dataset (Vid3oC) and keep most of EDVR setting as the same as the original implementation except setting the upscale module from factor 4 to factor 16 in order to make sure EDVR can generate extreme super-resolved frames.

4.4.2 Comparisons

Because few existing works related to video extreme super-resolution (with a scale factor of 16), in this work, we have compared our proposed network against with Bicubic interpolation and state-of-the-art EDVR.

Table 4.2 shows PSNR comparison results of our approach with the competing methods, Bicubic interpolation and EDVR with the scaling factor of 16. It can be seen that our DKSAN method has the best PSNR scores for all ten testing videos. We observe that significant PSNR gains (up to $4dB$) over previous state-of-the-art method EDVR. As discussed in previous Chapters, PSNR metrics sometimes cannot reflect the real visual quality of images, therefore, we also demonstrate the subjective quality comparison. In Fig. 4.7, "050_0010", we can easily observe that our proposed DKSAN can better reconstruct the structure of car tail; the results of Bicubic and EDVR are suffering from severe artifacts and blurring. Taking another example, "054_0007" in Fig. 4.8, our DKSAN can recover rich details of the face compared with Bicubic and EDVR which suffering from blurring and distortion. More subjective results can be find in Fig. 4.74.84.9, please zoom in for a better detailed comparison.

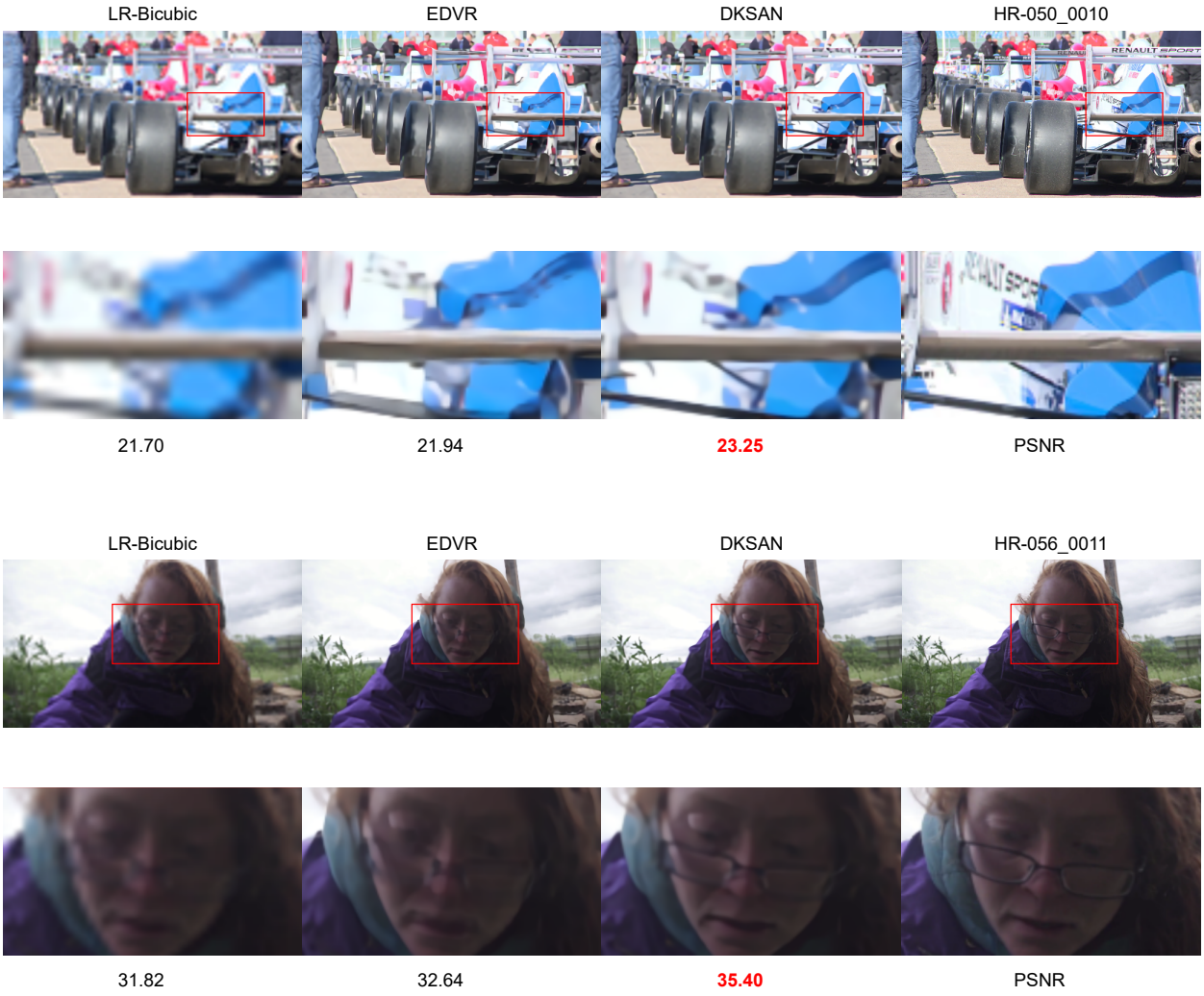

Figure 4.7: Visual comparison results among competing approaches for IntVID dataset (video 050, 051, 052) at a scaling factor of 16.

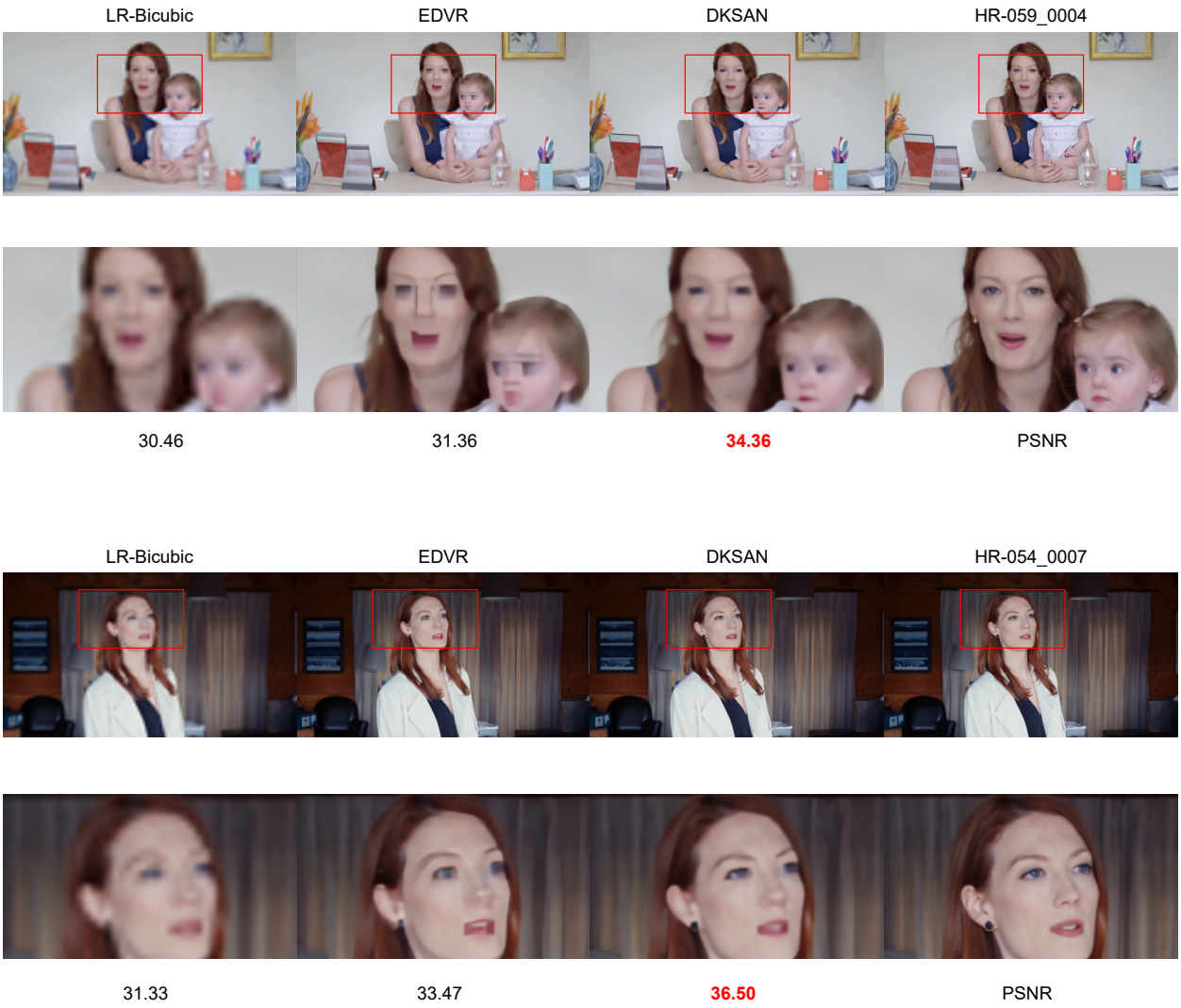

Figure 4.8: Visual comparison results among competing approaches for IntVID dataset (video 053, 054, 055) at a scaling factor of 16.

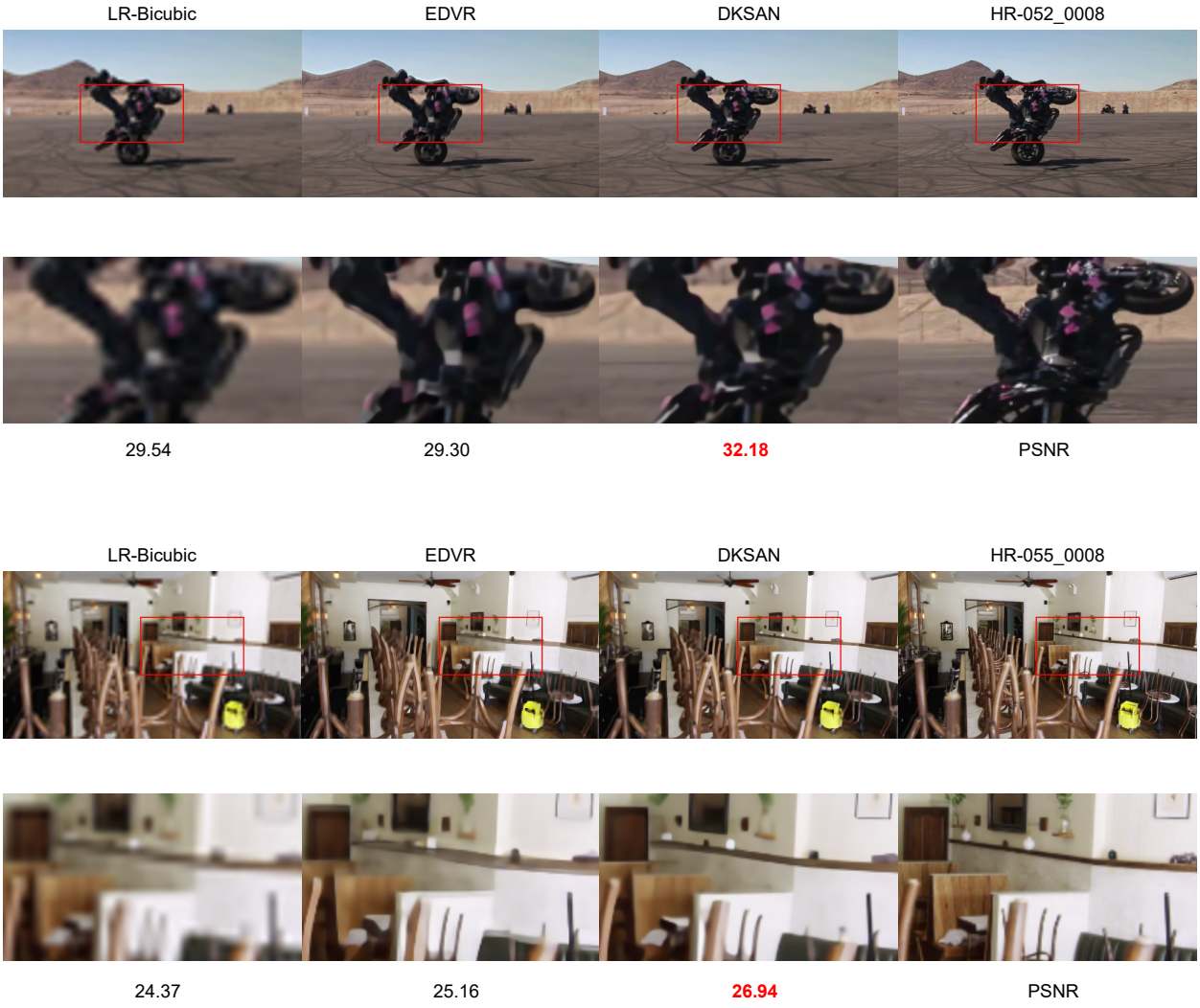

Figure 4.9: Visual comparison results among competing approaches for IntVID dataset (video 057, 058, 059) at a scaling factor of 16.

Video Name	Scale	Bicubic	EDVR	DKSAN (ours)
		PSNR (dB)	PSNR (dB)	PSNR (dB)
050	x16	21.56	21.81	**23.06**
051	x16	23.02	24.13	**24.92**
052	x16	29.56	29.33	**31.87**
053	x16	24.05	24.51	**25.09**
054	x16	31.34	33.15	**36.18**
055	x16	24.39	25.01	**26.88**
056	x16	31.16	31.93	**34.22**
057	x16	34.35	35.20	**39.75**
058	x16	36.00	37.36	**38.15**
059	x16	30.49	31.37	**34.17**
Average	x16	28.59	29.38	**31.43**

Table 4.2: Quantitative comparisons among Bicubic interpolation, EDVR and proposed DKSA on IntVID dataset (10 videos) for scaling factor of 16. **Bold** font indicates the best result.

4.4.3 Ablation Studies

To investigate the effect of proposed DKC_Align module and DKSA module, we have conducted different strategies to remove the certain components from the final framework DKSAN. In particular, we have implemented three competing models for our ablation studies: 1) training with only resblocks, without channel attention, alignment and DKSA; 2) training without DKC_Align and DKSA module; 3) training with DKC_Align module but without DKSA module; 4) training with all modules (proposed DKSAN). Note that all experiments are trained under same dataset and conditions for fair comparison.

Table. 4.3 shows the results of all four strategies mentioned previously with PSNR scores of each video and average. The backbone result is running only based on resblock, no channel attention, alignment and DKSA applied. From the results, we can see that the backbone has the worst performance; adding channel attention module but without DKC_Align and DKSA modules, the result is only 31.27 dB; after adding DKC_Align module, the result is improved to 31.32 dB; finally, we observe that after adding DKSA module (the full version of DKSAN), the result is further improved to 31.43 dB (**0.4dB** and **0.16dB** gained when compared with backbone and w/o

Video Name	Backbone	w/o Alignment & DKSA	w/o DKSA Module	DKSAN (ours)
	PSNR (dB)	PSNR (dB)	PSNR (dB)	PSNR (dB)
050	22.80	22.98	22.87	**23.06**
051	24.72	24.89	24.89	**24.92**
052	31.65	31.75	31.85	**31.87**
053	25.05	25.07	**25.18**	25.09
054	35.30	35.73	35.86	**36.18**
055	26.52	**26.89**	26.69	26.88
056	33.79	33.92	**34.33**	34.22
057	38.97	39.13	39.27	**39.75**
058	38.09	38.21	**38.30**	38.15
059	33.38	34.15	34.16	**34.17**
Average	31.03	31.27	31.32	**31.43**

Table 4.3: Ablation Studies for DKSAN on IntVID dataset for scaling factor of 16. Backbone means only resblocks are applied, channel-attention, alignment and DKSA modules are not applied; w/o Alignment & DKSA means DKC_Align and DKSA Module are not applied; w/o DKSA means only the DKSA module is not applied. **Bold** font indicates the best result.

Alignment & DKSA respectively) because of the effective module DKSA. Note that, as the PSNR scores shown in Table. 4.3, we can find that for some cases such as video "058", the DKSAN cannot deliver a convincing result. The main reason is video "058" has very small motions compared with other subjects, which our designed DKSA and alignment modules cannot reconstruct them very well.

4.5 Summary

In this Chapter, we reviewed the recent state-of-the-art approaches related to multi-frame video super-resolution, and proposed a novel Deformable Kernel Spatial Attention Network (DKSAN) to solve the video extreme super-resolution task (with the scale factor of 16). Our experimental results indicate that our method can achieve both the higher PSNR scores and better visual quality compared with previous state-of-the-art method EDVR [56].

Chapter 5

Conclusions and Future Work

In this dissertation, we have mainly studied three low-level vision problems: 1) single image super-resolution; 2) joint demosaicing and super-resolution; 3) multi-frame video super-resolution, to demonstrate the efficient of proposed attention mechanisms (spatial-spectral attention, color attention and deformable kernel spatial attention).

5.1 Single Image Super-Resolution

In this work, we developed a spatial color attention networks (SCAN) to tackle the problem of single image super-resolution based on real-world image dataset from NTIRE2019 challenge. The newly designed spatial color attention module (SCAM) can enable the networks to learn the joint representations across spectral channels and better calibrate the feature maps with R,G,B spatial color attention maps. When compared with start-of-the-art RCAN, our method SCAN can significantly improve both objective (including PSNR/SSIM/PI) and subjective results. Meantime, the designed SCAM module can easily be integrated with other existing super-resolution networks. Under the framework of NTIRE challenge, one issue that remains to be addressed is the modeling/learning of real-world degradation (the forward process). We expect that exploiting a priori

information about the degradation process can offer new insight to the problem of real SISR.

5.2 Joint Image Demosaicing and Super-Resolution

In this work, we proposed to study the problem of joint demosaicing and super-resolution (JDSR) - a topic has been underexplored in the literature of deep learning. Our solution includes two networks design: 1) DSERN; 2) RDSEN. DSERN consists of a new densely-connected squeeze-and-excitation residual network for image reconstruction. Compared with naive network designs, our proposed network can stack more layers and be trained deeper by newly designed DSE block. This is because DSE makes multiple expansions on a reduced channel descriptor to allow more faithful information flow. As an updated version, RDSEN has newly designed PDNet and RDSEB block to reduce the computational complexity of DSERN and further improve both objective and subjective performance.

Additionally, we have studied the problem of perceptual optimization for JDSR. Our experimental results have verified that TRaGAN can generate more realistically-looking images (especially around textured regions) and achieve lower PI scores than standard GAN. Finally, we have evaluated our proposed method (RDSEN_TRaGAN) on real-world Bayer patterns collected by the Mastcam of NASA Mars Curiosity Rover, which supports its superiority to naive network design (e.g., Flex+RCAN) and the effectiveness of perceptual optimization. Another potential application of JDSR in practice is the digital zoom feature in smartphone cameras.

5.3 Multi-Frame Video Super-Resolution

In this work, we proposed a multi-frame based cascade VSR network DKSAN for extreme low-resolution videos. The novel temporal alignment module, DKC_Align, can help the networks to better learn and align the detailed features under both local and global fields between reference frame and its neighboring frames. Furthermore, the DKSA module calibrated the reconstructed complementary features to further enhance the edges and textures at the spatial domain. Thanks to the newly designed DKC_Align and DKSA modules, the proposed architecture can reconstruct high-quality HR frames from extreme LR frames and significantly improve both objective and

subjective results when compared with state-of-the-art approach EDVR.

5.4 Future Works

Although in Chapter 2, we proposed a possible solution for real world image super-resolution, it still requests paired high and low resolution images as training data. The limitations of this data collection way are that: 1) request accurate alignment method to align paired images; 2) because of the difficulty of collection data, it's almost impossible to collect a large dataset to improve the network generalization ability. To solve these limitations, an unpaired approach for SISR is worth to be explored.

Furthermore, Graph Convolutional Networks (GCN) recently has been noticed by computer vision society, and it has been applied for many low-level vision tasks such as depth completion [114] and super-resolution [115]. Inspired by [115], we consider studying a more efficient solution for JDSR tasks to better trade-off network complexity and reconstruction quality in order to apply it in practice.

Finally, in modern life, people are looking not only for video clarity such as 1080p, 2K, 4K even 8K, but also for the fluency of video (from $30fps$ to $60fps$ even $90fps$), such the demand like this will result in insufficient bandwidth. Therefore, the Joint Video Spatial and Temporal Super-Resolution (JSTSR) for extreme low-resolution videos is a valuable open topic to study to help video coding, compression etc. In the future work, we plan to explore an efficient solution to utilize Conv3D layer and Recurrent Neural Networks (RNN) for JSTSR.

Bibliography

[1] F. Heide, M. Steinberger, Y.-T. Tsai, M. Rouf, D. Pajak, D. Reddy, O. Gallo, J. Liu, W. Heidrich, K. Egiazarian, *et al.*, "FlexISP: A flexible camera image processing framework," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 6, p. 231, 2014.

[2] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[3] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European conference on computer vision*, pp. 184–199, Springer, 2014.

[4] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1646–1654, 2016.

[5] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network.," in *CVPR*, vol. 2, p. 4, 2017.

[6] M. Gharbi, G. Chaurasia, S. Paris, and F. Durand, "Deep joint demosaicking and denoising," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 191, 2016.

[7] "Mast camera (mastcam)n." `https://msl-scicorner.jpl.nasa.gov/Instruments/Mastcam/`.

[8] X. Xu and X. Li, "SCAN: Spatial color attention networks for real single image super-resolution," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2024–2032, 2019.

[9] X. Xu, Y. Ye, and X. Li, "Joint demosaicing and super-resolution (JDSR): Network design and perceptual optimization," *IEEE Transactions on Computational Imaging*, pp. 1–1, 2020.

[10] H. Greenspan, "Super-resolution in medical imaging," *The computer journal*, vol. 52, no. 1, pp. 43–63, 2009.

[11] L. Zhang, H. Zhang, H. Shen, and P. Li, "A super-resolution reconstruction algorithm for surveillance images," *Signal Processing*, vol. 90, no. 3, pp. 848–859, 2010.

[12] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 11, pp. 1254–1259, 1998.

[13] L. Itti and C. Koch, "Computational modelling of visual attention," *Nature reviews neuroscience*, vol. 2, no. 3, p. 194, 2001.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

[15] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5659–5667, 2017.

[16] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, pp. 2048–2057, 2015.

[17] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Lip reading sentences in the wild," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3444–3453, IEEE, 2017.

[18] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3156–3164, 2017.

[19] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[20] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, 2018.

[21] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu, *et al.*, "Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2956–2964, 2015.

[22] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, "Spatial transformer networks," in *Advances in neural information processing systems*, pp. 2017–2025, 2015.

[23] Y. Hu, J. Li, Y. Huang, and X. Gao, "Channel-wise and spatial feature modulation network for single image super-resolution," *arXiv preprint arXiv:1809.11130*, 2018.

[24] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, "Residual non-local attention networks for image restoration," *arXiv preprint arXiv:1903.10082*, 2019.

[25] F. S. Khan, J. Van De Weijer, and M. Vanrell, "Top-down color attention for object recognition," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 979–986, IEEE, 2009.

[26] F. S. Khan, J. Van de Weijer, and M. Vanrell, "Modulating shape features by color attention for object recognition," *International Journal of Computer Vision*, vol. 98, no. 1, pp. 49–64, 2012.

[27] J.-M. Geusebroek, R. Van den Boomgaard, A. W. M. Smeulders, and H. Geerts, "Color invariance," *IEEE Transactions on Pattern analysis and machine intelligence*, vol. 23, no. 12, pp. 1338–1350, 2001.

[28] L. Zhang and X. Wu, "Color demosaicking via directional linear minimum mean square-error estimation," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2167–2178, 2005.

[29] X. Li, B. Gunturk, and L. Zhang, "Image demosaicing: A systematic survey," in *Visual Communications and Image Processing 2008*, vol. 6822, p. 68221J, International Society for Optics and Photonics, 2008.

[30] X. Li, "Demosaicing by successive approximation," *IEEE Transactions on Image Processing*, vol. 14, no. 3, pp. 370–379, 2005.

[31] W. Ye and K.-K. Ma, "Color image demosaicing using iterative residual interpolation," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5879–5891, 2015.

[32] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," 2012.

[33] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1, pp. I–I, IEEE, 2004.

[34] R. Timofte, V. De Smet, and L. Van Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *Proceedings of the IEEE international conference on computer vision*, pp. 1920–1927, 2013.

[35] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE transactions on image processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[36] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*, pp. 711–730, Springer, 2010.

[37] F.-L. He, Y.-C. F. Wang, and K.-L. Hua, "Self-learning approach to color demosaicking via support vector regression," in *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pp. 2765–2768, IEEE, 2012.

[38] O. Kapah and H. Z. Hel-Or, "Demosaicking using artificial neural networks," in *Applications of Artificial Neural Networks in Image Processing V*, vol. 3962, pp. 112–121, International Society for Optics and Photonics, 2000.

[39] F. Kokkinos and S. Lefkimmiatis, "Deep image demosaicking using a cascade of convolutional residual denoising networks," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[40] J. Sun and M. F. Tappen, "Separable markov random field model and its applications in low level vision," *IEEE transactions on image processing*, vol. 22, no. 1, pp. 402–407, 2013.

[41] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "ESRGAN: Enhanced super-resolution generative adversarial networks," in *The European Conference on Computer Vision Workshops (ECCVW)*, September 2018.

[42] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[43] R. Zhou, R. Achanta, and S. Süsstrunk, "Deep residual network for joint demosaicing and super-resolution," in *Color and Imaging Conference*, vol. 2018, pp. 75–80, Society for Imaging Science and Technology, 2018.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[45] H. Seibel, S. Goldenstein, and A. Rocha, "Eyes on the target: Super-resolution and license-plate recognition in low-quality surveillance videos," *IEEE access*, vol. 5, pp. 20020–20035, 2017.

[46] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE transactions on image processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[47] C. Liu and D. Sun, "On bayesian adaptive video super resolution," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 2, pp. 346–360, 2013.

[48] Z. Ma, R. Liao, X. Tao, L. Xu, J. Jia, and E. Wu, "Handling motion blur in multi-frame super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5224–5232, 2015.

[49] W. Wang, C. Ren, X. He, H. Chen, and L. Qing, "Video super-resolution via residual learning," *IEEE Access*, vol. 6, pp. 23767–23777, 2018.

[50] B. K. Horn and B. G. Schunck, "Determining optical flow," in *Techniques and Applications of Image Understanding*, vol. 281, pp. 319–331, International Society for Optics and Photonics, 1981.

[51] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang, "Robust video super-resolution with learned temporal dynamics," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2507–2515, 2017.

[52] M. S. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-recurrent video super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6626–6634, 2018.

[53] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4778–4787, 2017.

[54] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 764–773, 2017.

[55] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9308–9316, 2019.

[56] X. Wang, K. C. Chan, K. Yu, C. Dong, and C. Change Loy, "EDVR: Video restoration with enhanced deformable convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0, 2019.

[57] Y. Tian, Y. Zhang, Y. Fu, and C. Xu, "TDAN: Temporally-deformable alignment network for video super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3360–3369, 2020.

[58] H. Wang, D. Su, C. Liu, L. Jin, X. Sun, and X. Peng, "Deformable non-local network for video super-resolution," *IEEE Access*, vol. 7, pp. 177734–177744, 2019.

[59] H. Gao, X. Zhu, S. Lin, and J. Dai, "Deformable kernels: Adapting effective receptive fields for object deformation," in *International Conference on Learning Representations*, 2020.

[60] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images.," in *ICCV*, vol. 98, p. 2, 1998.

[61] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[62] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 4809–4817, IEEE, 2017.

[63] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[64] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[65] A. Jolicoeur-Martineau, "The relativistic discriminator: a key element missing from standard GAN," in *International Conference on Learning Representations*, 2019.

[66] T. Vu, T. M. Luu, and C. D. Yoo, "Perception-enhanced image super-resolution via relativistic generative adversarial networks," in *European Conference on Computer Vision*, pp. 98–113, Springer, 2018.

[67] W. Ye and K.-K. Ma, "Color image demosaicing using iterative residual interpolation," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5879–5891, 2015.

[68] S. Kim, G. Li, D. Fuoli, M. Danelljan, Z. Huang, S. Gu, and R. Timofte, "The Vid3oC and IntVID datasets for video super resolution and quality mapping," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 3609–3616, IEEE, 2019.

[69] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate superresolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, p. 5, 2017.

[70] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[71] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "StackGAN++: Realistic image synthesis with stacked generative adversarial networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 1947–1962, 2018.

[72] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

[73] J. Chen, J. Chen, H. Chao, and M. Yang, "Image blind denoising with generative adversarial network based noise modeling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3155–3164, 2018.

[74] H. Zhu, X. Peng, V. Chandrasekhar, L. Li, and J.-H. Lim, "DehazeGAN: When image dehazing meets differential programming.," in *IJCAI*, pp. 1234–1240, 2018.

[75] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[76] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[77] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *European Conference on Computer Vision*, pp. 391–407, Springer, 2016.

[78] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 5, 2017.

[79] M. S. Sajjadi, B. Schölkopf, and M. Hirsch, "Enhancenet: Single image super-resolution through automated texture synthesis," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 4501–4510, IEEE, 2017.

[80] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1664–1673, 2018.

[81] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[82] Y. Blau and T. Michaeli, "The perception-distortion tradeoff," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[83] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[84] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.

[85] R. Timofte, R. Rothe, and L. Van Gool, "Seven ways to improve example-based single image super resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1865–1873, 2016.

[86] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor, "2018 PIRM challenge on perceptual image super-resolution," *arXiv preprint arXiv:1809.07517*, 2018.

[87] C. Ma, C.-Y. Yang, X. Yang, and M.-H. Yang, "Learning a no-reference quality metric for single-image super-resolution," *Computer Vision and Image Understanding*, vol. 158, pp. 1–16, 2017.

[88] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a" completely blind" image quality analyzer.," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 209–212, 2013.

[89] N.-S. Syu, Y.-S. Chen, and Y.-Y. Chuang, "Learning deep convolutional networks for demosaicing," *arXiv preprint arXiv:1802.03769*, 2018.

[90] R. Tan, K. Zhang, W. Zuo, and L. Zhang, "Color image demosaicking via deep residual learning," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 793–798, IEEE, 2017.

[91] W. Dong, M. Yuan, X. Li, and G. Shi, "Joint demosaicing and denoising with perceptual optimization on a generative adversarial network," *arXiv preprint arXiv:1802.04723*, 2018.

[92] Q. Wang and G. Guo, "LS-CNN: Characterizing local patches at multiple scales for face recognition," *IEEE Transactions on Information Forensics and Security*, 2019.

[93] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[94] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*, pp. 711–730, Springer, 2010.

[95] 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 *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 2, pp. 416–423, IEEE, 2001.

[96] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5197–5206, 2015.

[97] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, "Sketch-based manga retrieval using manga109 dataset," *Multimedia Tools and Applications*, vol. 76, no. 20, pp. 21811–21838, 2017.

[98] L. Zhang, X. Wu, A. Buades, and X. Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *Journal of Electronic imaging*, vol. 20, no. 2, p. 023016, 2011.

[99] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, "Demosaicking: color filter array interpolation," *IEEE Signal processing magazine*, vol. 22, no. 1, pp. 44–54, 2005.

[100] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "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*, pp. 1874–1883, 2016.

[101] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1712–1722, 2019.

[102] V. Singh, K. Ramnath, S. Arunachalam, and A. Mittal, "Going much wider with deep networks for image super-resolution," in *The IEEE Winter Conference on Applications of Computer Vision*, pp. 2343–2354, 2020.

[103] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*, pp. 694–711, Springer, 2016.

[104] X. Xiang, Y. Tian, Y. Zhang, Y. Fu, J. P. Allebach, and C. Xu, "Zooming Slow-Mo: Fast and accurate one-stage space-time video super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3370–3379, 2020.

[105] T. Isobe, S. Li, X. Jia, S. Yuan, G. Slabaugh, C. Xu, Y.-L. Li, S. Wang, and Q. Tian, "Video super-resolution with temporal group attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8008–8017, 2020.

[106] R. Liao, X. Tao, R. Li, Z. Ma, and J. Jia, "Video super-resolution via deep draft-ensemble learning," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 531–539, 2015.

[107] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, 2016.

[108] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4472–4480, 2017.

[109] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in neural information processing systems*, pp. 802–810, 2015.

[110] M. Haris, G. Shakhnarovich, and N. Ukita, "Recurrent back-projection network for video super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3897–3906, 2019.

[111] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video enhancement with task-oriented flow," *International Journal of Computer Vision*, vol. 127, no. 8, pp. 1106–1125, 2019.

[112] Y. Jo, S. Wug Oh, J. Kang, and S. Joo Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3224–3232, 2018.

[113] G. Bertasius, L. Torresani, and J. Shi, "Object detection in video with spatiotemporal sampling networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 331–346, 2018.

[114] X. Xiong, H. Xiong, K. Xian, C. Zhao, Z. Cao, and X. Li, "Sparse-to-dense depth completion revisited: Sampling strategy and graph construction," in *European Conference on Computer Vision (ECCV)*, 2020.

[115] S. Zhou, J. Zhang, W. Zuo, and C. C. Loy, "Cross-scale internal graph neural network for image super-resolution," *arXiv preprint arXiv:2006.16673*, 2020.

List of Publications

[1] X. Xu and X. Li, "SCAN: Spatial color attention networks for real single image super-resolution," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2024–2032, 2019.

[2] X. Xu, Y. Ye, and X. Li, "Joint demosaicing and super-resolution (JDSR): Network design and perceptual optimization," *IEEE Transactions on Computational Imaging*, pp. 1–1, 2020.

[3] J. Cai, S. Gu, R. Timofte, L. Zhang, X. Xu, and X. Li, "NTIRE 2019 challenge on real image super-resolution: Methods and results," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[4] U. Sanzhaeva, X. Xu, P. Guggilapu, M. Tseytlin, V. V. Khramtsov, and B. Driesschaert, "Imaging of enzyme activity by electron paramagnetic resonance: Concept and experiment using a paramagnetic substrate of alkaline phosphatase," *Angewandte Chemie*, vol. 130, no. 36, pp. 11875–11879, 2018.

[5] O. Tseytlin, P. Guggilapu, A. A. Bobko, H. AlAhmad, X. Xu, B. Epel, R. O'Connell, E. H. Hoblitzell, T. D. Eubank, V. V. Khramtsov, *et al.*, "Modular imaging system: Rapid scan epr at 800 mhz," *Journal of Magnetic Resonance*, 2019.

[6] X. Xu, M. Martin, and T. Bourlai, "Automatic tattoo image registration system," in *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 1238–1243, IEEE, 2016.

[7] M. Martin, X. Xu, and T. Bourlai, "A multimedia application for location-based semantic retrieval of tattoos," in *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pp. 1–8, IEEE, 2016.