华东师范大学

East China Normal University

SEEK TRUTH. FOSTER ORIGINALITY AND
LIVE UP TO THE NAME OF TEACHER

Juncheng LI （李俊诚）

2020.12.21

Name: Juncheng LI (李俊诚)

Research: Computer Vision、Image Processing

Supervisor: Guixu Zhang (张桂戌) & Faming Fang (方发明)

HomePage: junchenglee.com

Application

Low-Level Vision	High-Level Vision
Image Super-Restoration	Image Classification
Image Denoising	Image Segmentation
Image Dehazing	Image Stitching
Image Deblurring	Object Detection
Image Enhancement	Crowd Counting

Theoretical

Learning Strategy
Generative Adversarial Learning
Weakly Supervised Learning
Meta Learning

Exploration and Construction of Lightweight Image Restoration Model

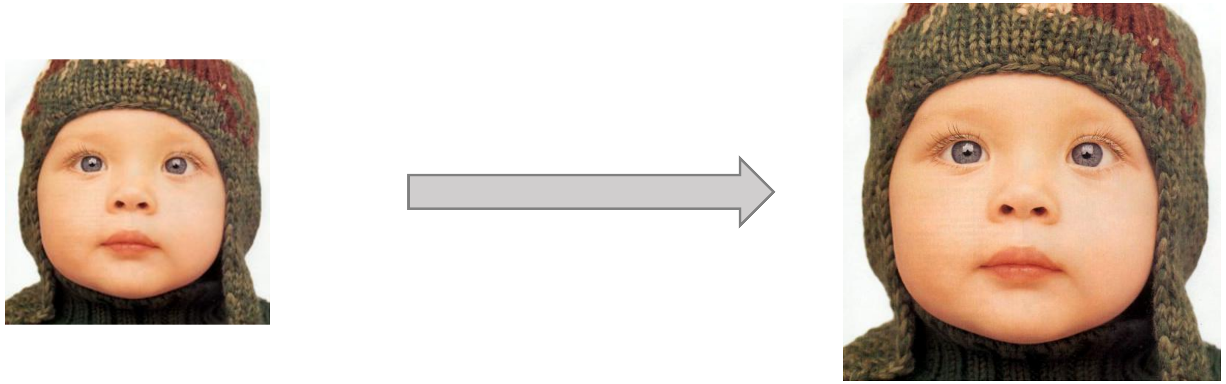
01

Introduction

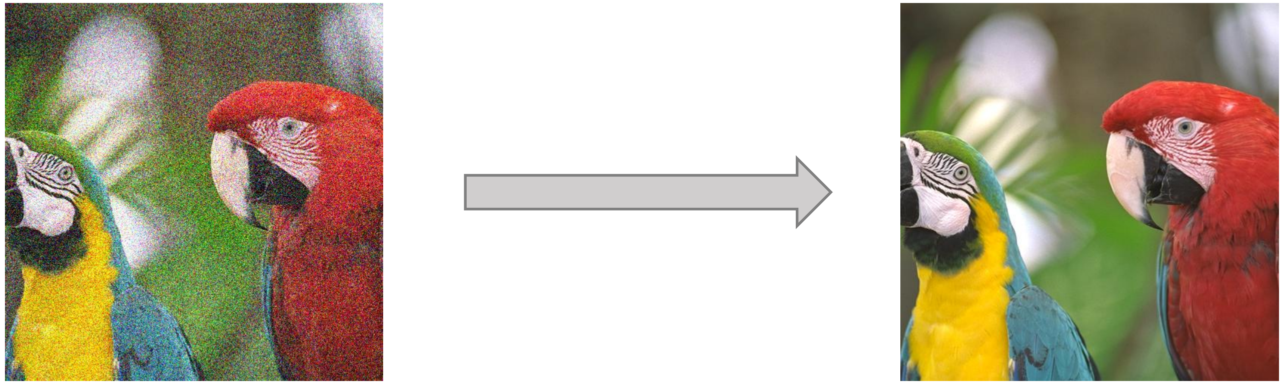
What is image restoration?

Image Restoration (IR) aims to reconstruct visually pleasing high-quality (HQ) images from degraded low-quality (LQ) images (such as low-restoration images, noisy images, compressed images, and blurred images), which is important for high-level computer vision tasks and has been widely used in security surveillance, autonomous driving, and medical image processing.

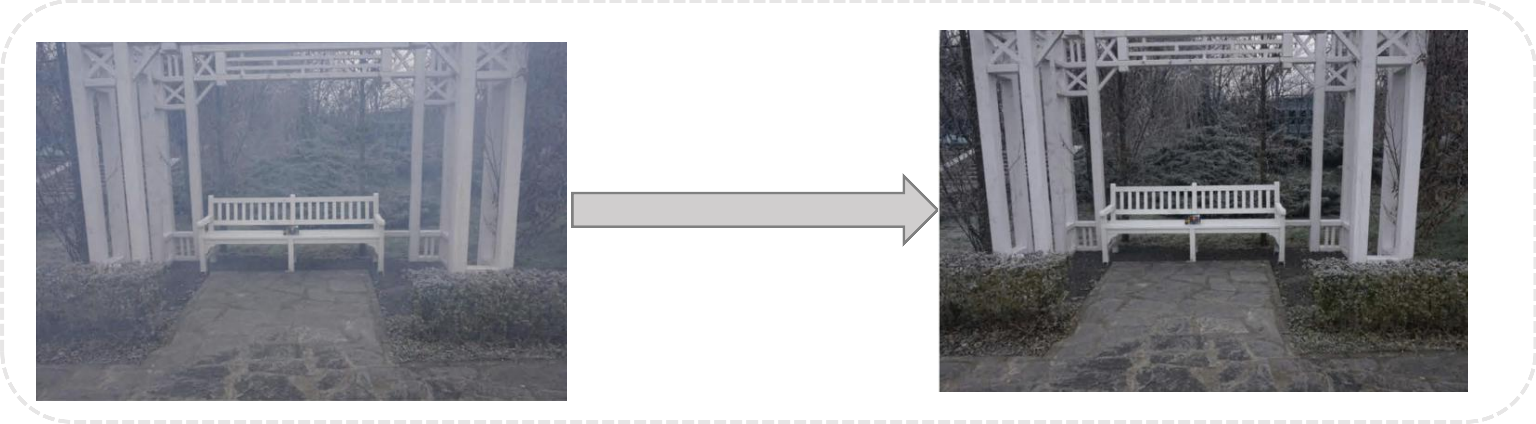
Single Image
Super-Resolution
（图像超分辨率）



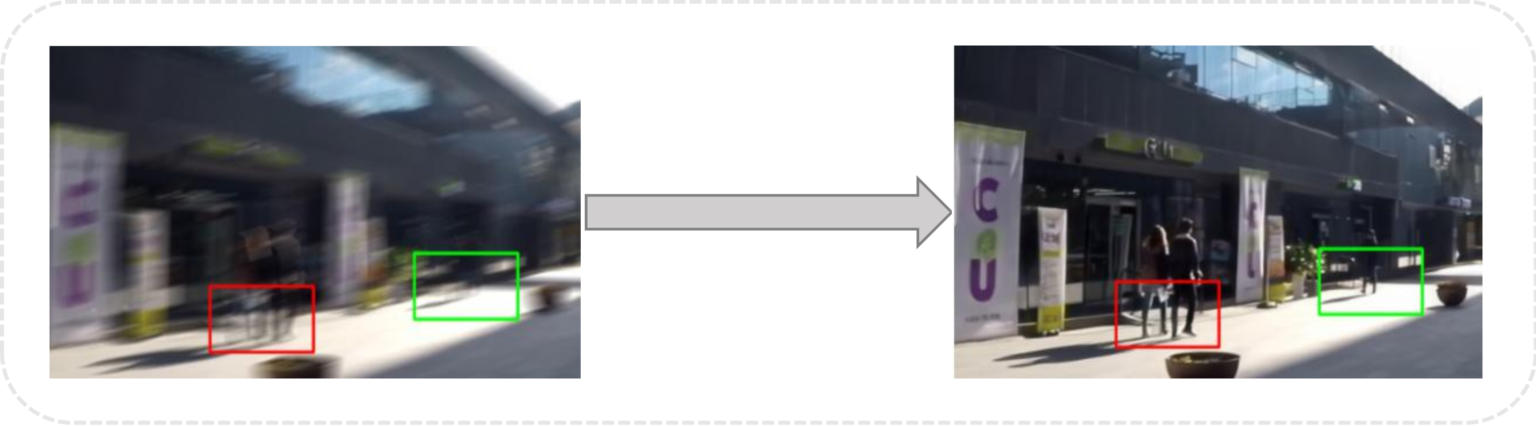
Single Image
Denoising
（图像去噪）



Single Image
Dehazing
（图像去雾）



Single Image
Deblurring
（图像去模糊）



How to reconstruct high-quality images?

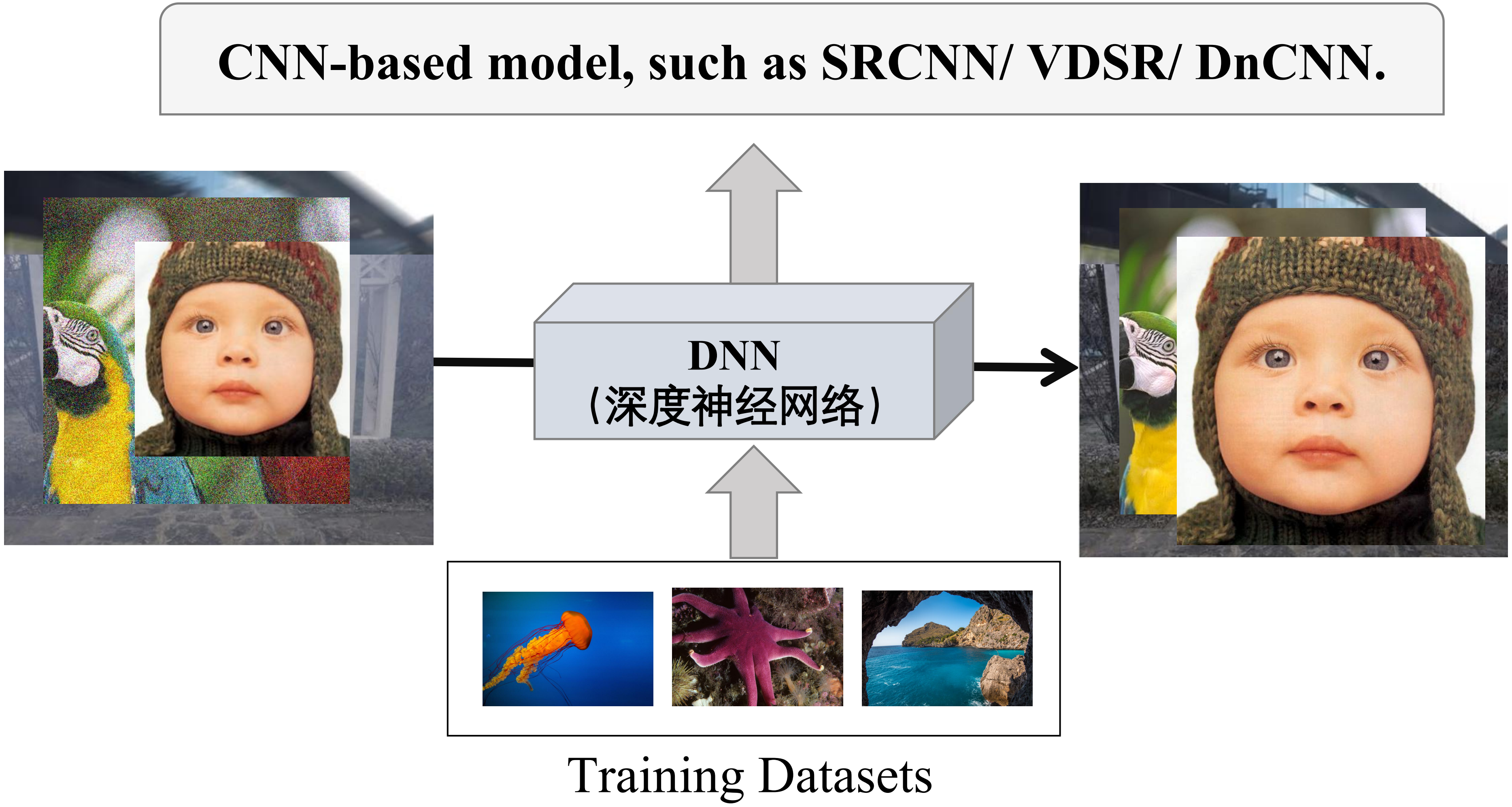
Degraded image

Traditional methods

Deep learning-based methods

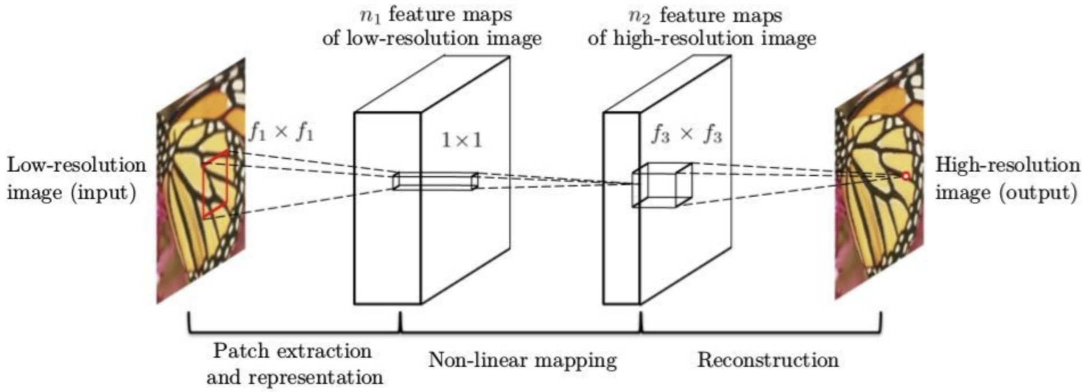
High-quality images

Introduction

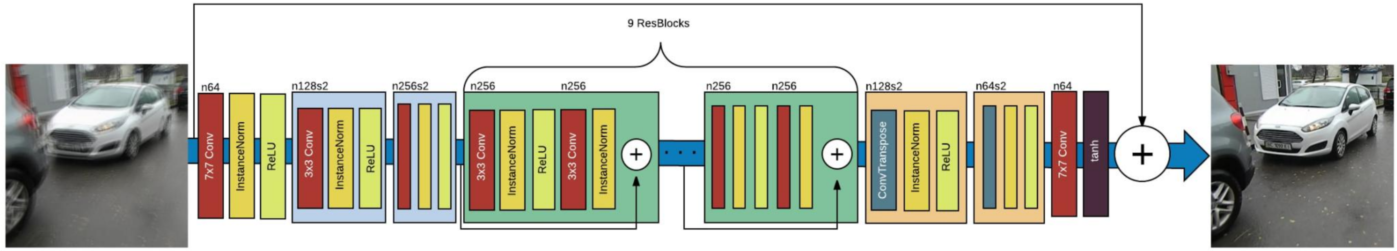


02

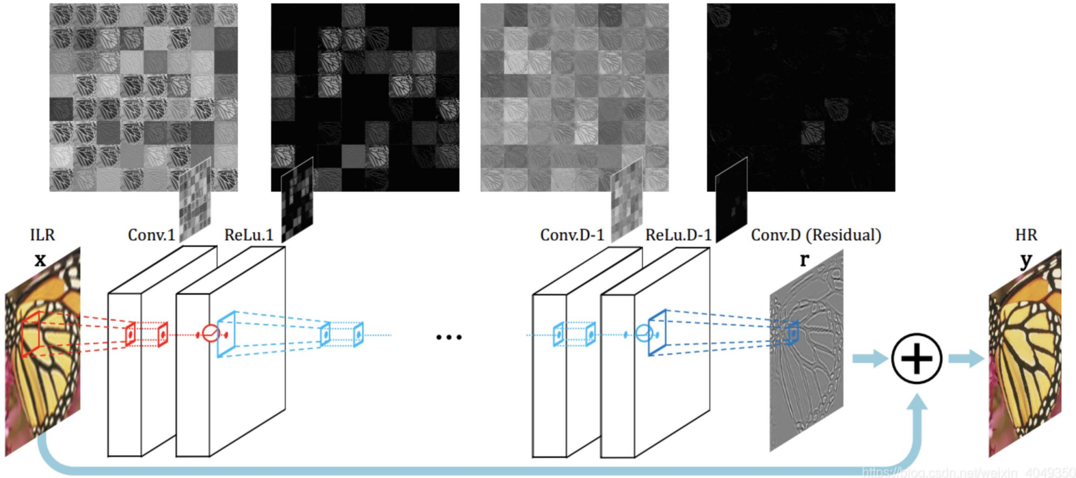
Motivation



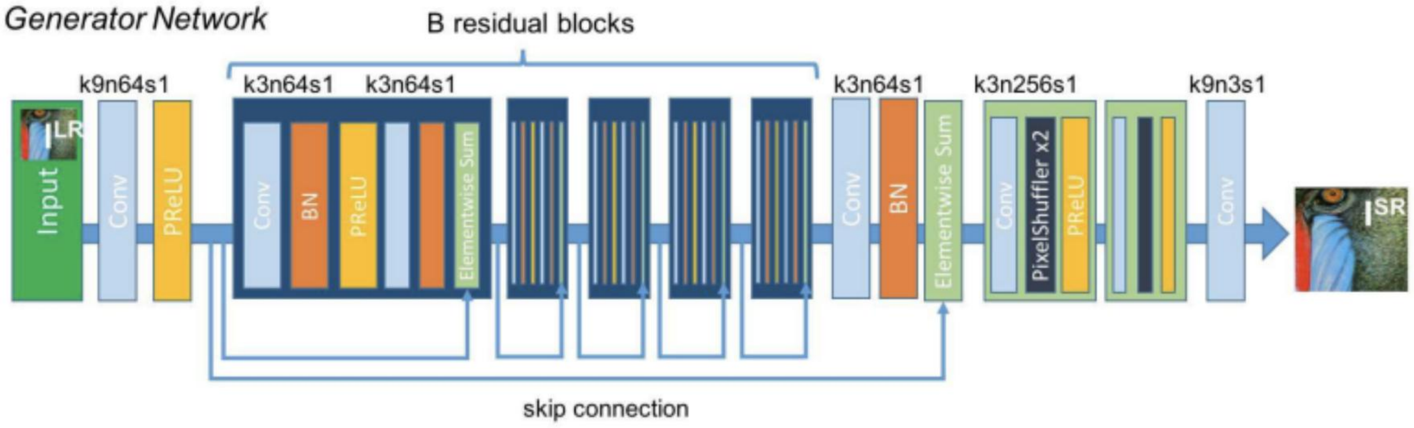
SRCNN



DeblurGAN

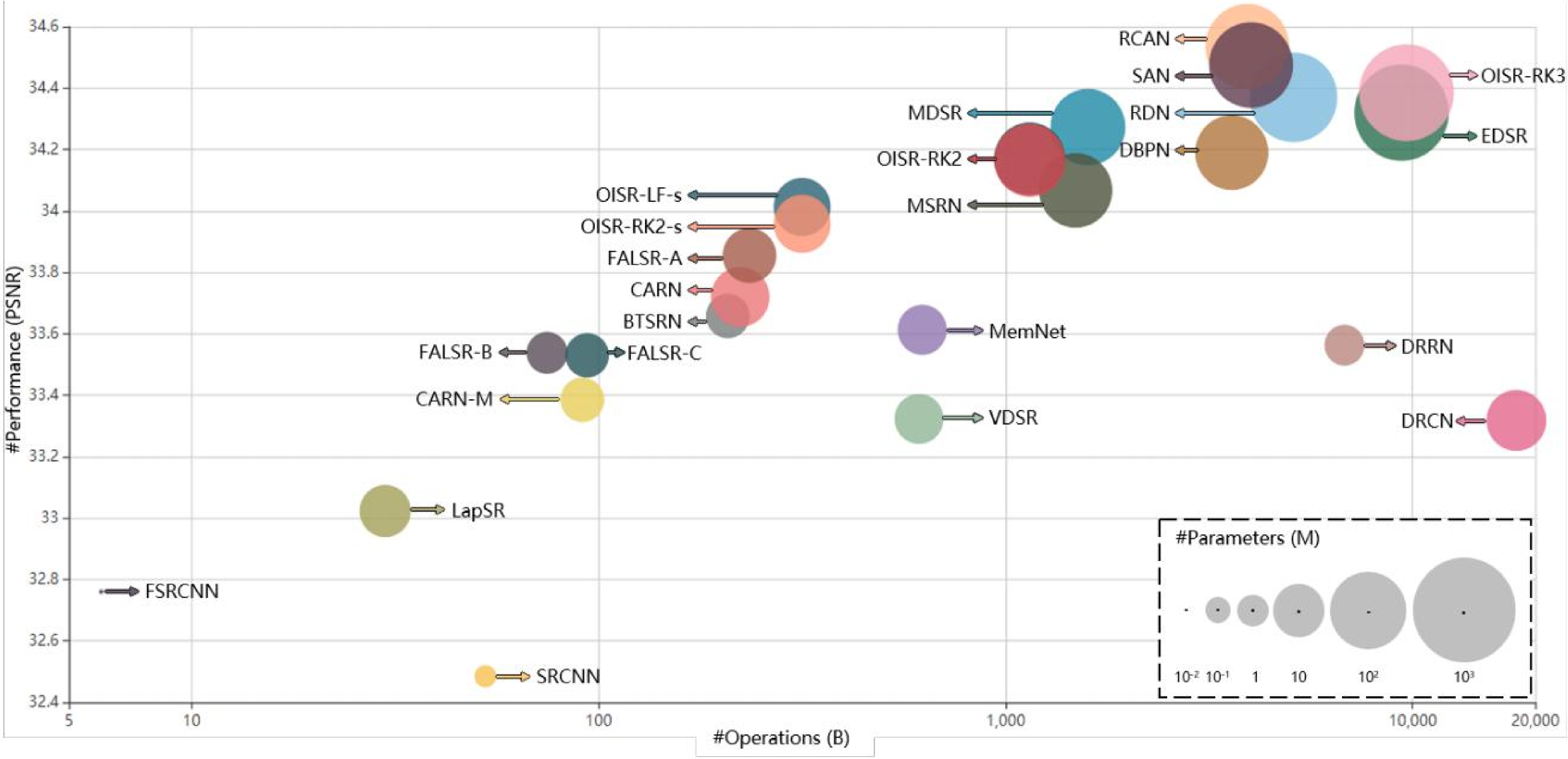


VDSR



SRResNet

Motivation



RCAN
SAN
OISR-RK3
MDSR
RDN
EDSR
DBPN
OISR-RK2
MSRN
OISR-LF-s
OISR-RK2-s
FALSR-A
CARN
BTSRN
MemNet
DRRN
FALSR-B
FALSR-C
CARN-M
VDSR
DRCN
LapSR
FSRCNN
SRCNN
#Performance (PSNR)
34.6
34.4
34.2
34
33.8
33.6
33.4
33.2
33
32.8
32.6
32.4
5
10
100
1,000
10,000
20,000
#Operations (B)
#Parameters (M)
10-2 10-1 1 10 10² 10³

The problems faced by these models:

The size of the model becomes larger and larger, the number of parameters becomes more and more, and the structure of the model becomes more and more complex.

More Training Datasets.
More Training Skills
More Training Time.
More Storage Space.
More Execution Time.
More Computing Resources.

Fewer Application Scenarios.

Exploring lightweight image restoration model is essential !

- Making full use of the features of the input image.
- Designing more effective feature extraction modules.
- Introducing image priors to guide image reconstruction.
- Exploring more effective coaching and training strategies.

Motivation

- Multi-scale Residual Network for Image Super-Resolution.
  ECCV, 2018 (Top CV Conference， 157 Citations， 213 Star in GitHub)
- Lightweight and Accurate Recursive Fractal Network for Image Super-Resolution.
  ICCV Workshop, 2019 (Oral Presentation)
- HighEr-Resolution Network for Image Demosaicing and Enhancing.
  ICCV Workshop, 2019 (ICCV-AIM2019 Winner)
- Luminance-aware Pyramid Network for Low-light Image Enhancement.
  IEEE Transactions on Multimedia (IEEE TMM), 2020.
- Multi-level Edge Features Guided Network for Image Denoising.
  IEEE Transactions on Neural Networks and Learning Systems (IEEE TNNLS), 2020.
- Soft-edge Assisted Network for Single Image Super-Resolution.
  IEEE Transactions on Image Processing (IEEE TIP), 2020.
- MDCN: Multi-scale Dense Cross Network for Image Super-Resolution.
  IEEE Transactions on Circuits and Systems for Video Technology (IEEE TCSVT), 2020.

Construction of image restoration model based on multi-scale feature fusion:

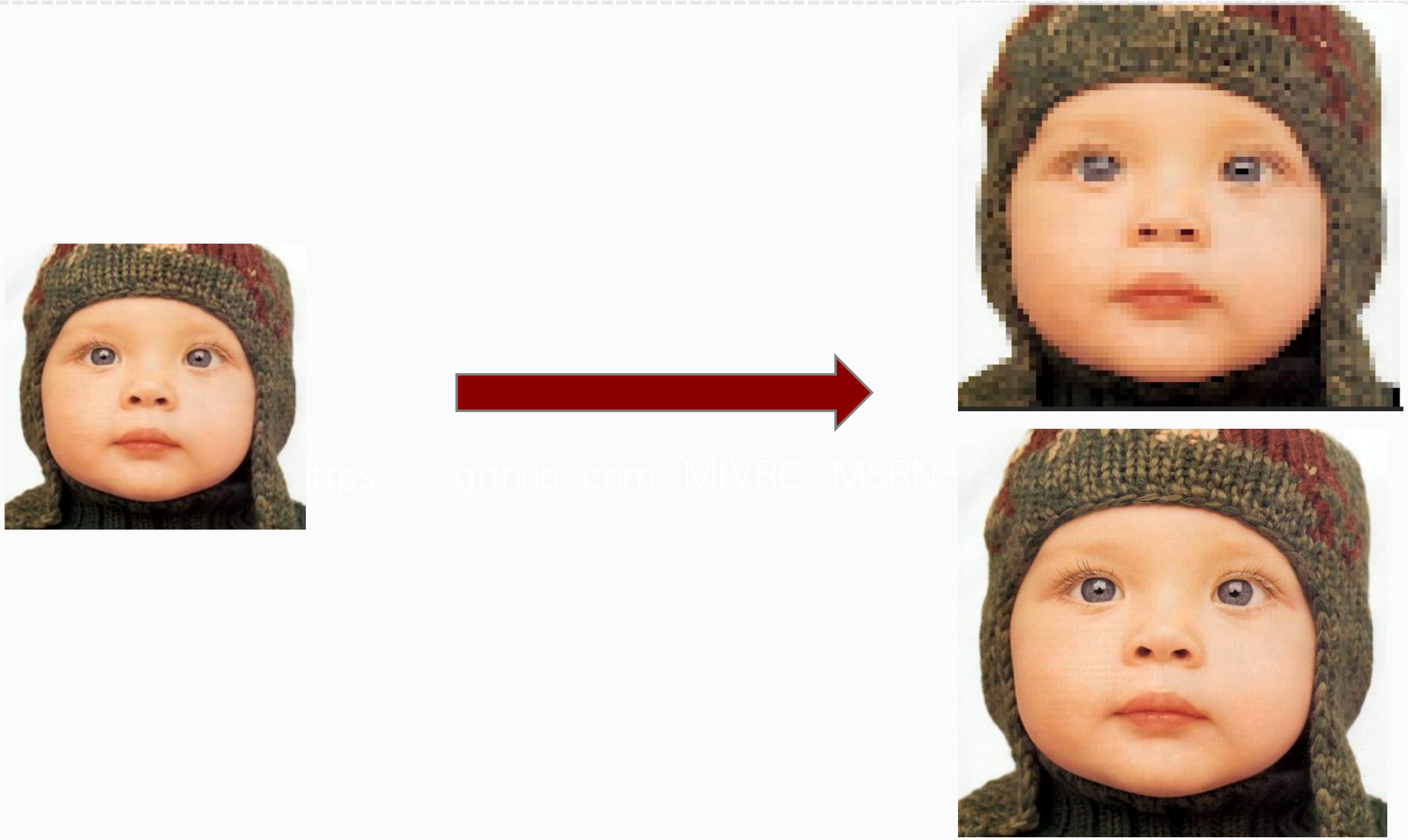
- Multi-scale Residual Network for Image Super-Resolution.
  MSRN, Image Super-resolution.
- MDCN: Multi-scale Dense Cross Network for Image Super-Resolution.
  MDCN, Image Super-resolution, Improved version of MSRN.

Construction of image restoration model based on edge priors guidance:

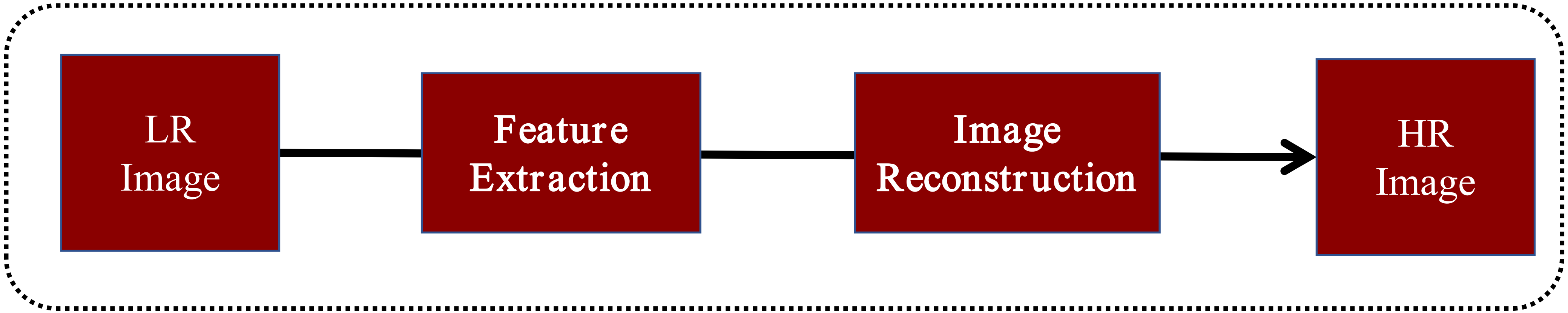
- Soft-edge Assisted Network for Single Image Super-Resolution.
  SeaNet, Image Super-resolution.
- Multi-level Edge Features Guided Network for Image Denoising.
  MLEFGN, Image Denoising.

03

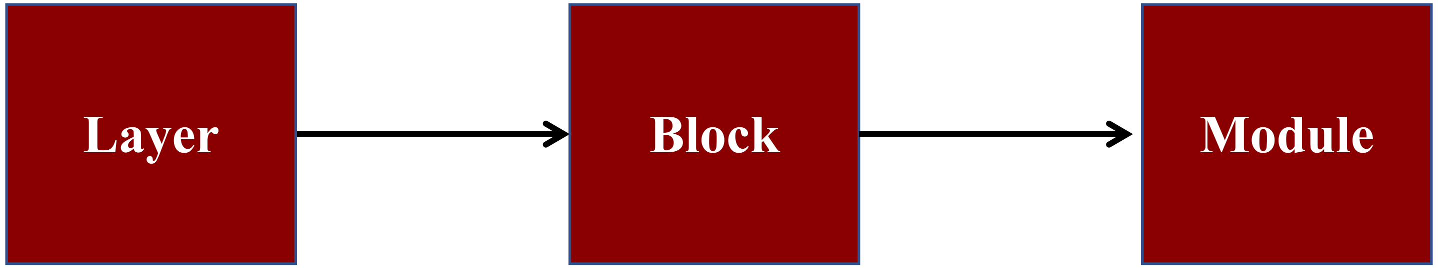
Multi-scale Feature Fusion
MSRN & MDCN



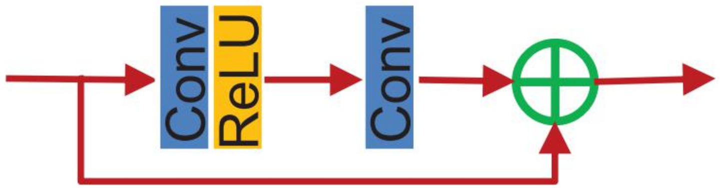
SISR： The task aims to reconstruct a High-Resolution (HR) image from a Low-Resolution (LR) image.



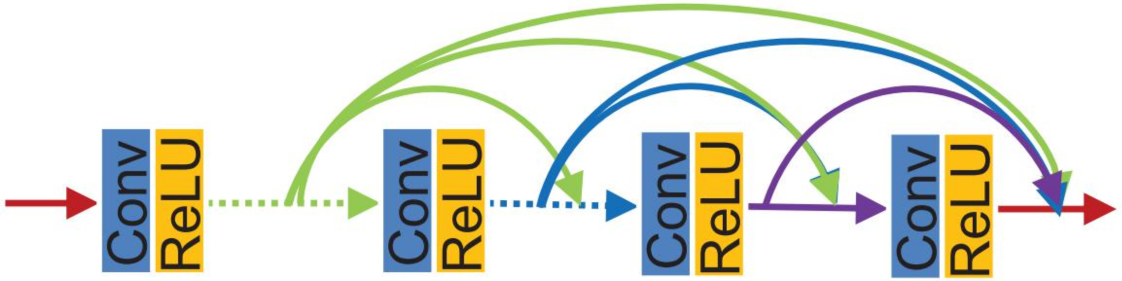
The process of image restoration.



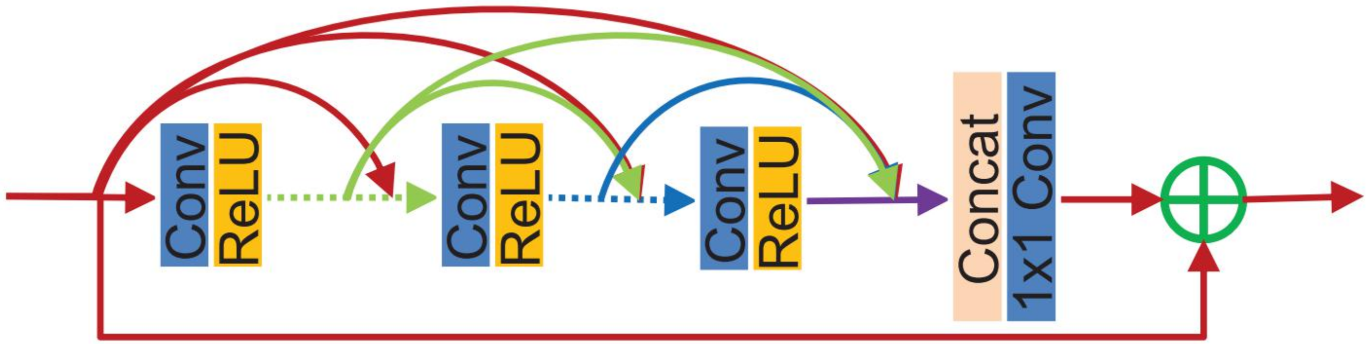
The evolution process of image feature extraction.



(a) Residual block

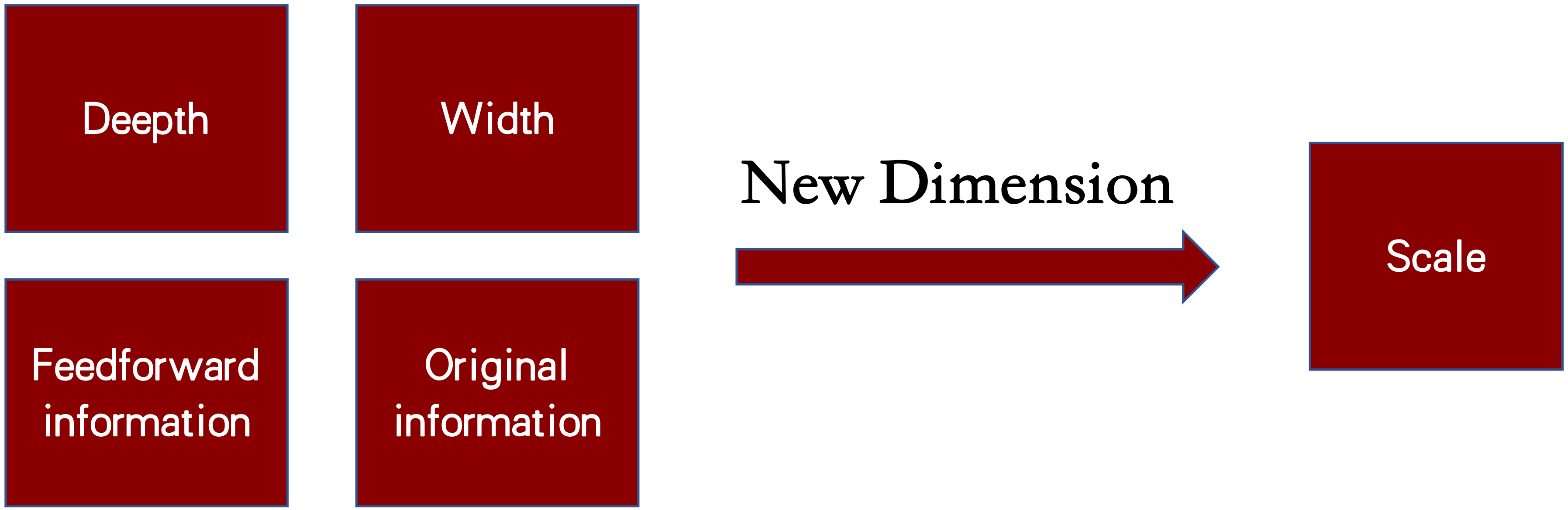


(b) Dense block



(c) Residual dense block

MSRN & MDCN


Deepth
Width
Feedforward information
Original information
New Dimension
Scale

MSRN & MDCN

60×40

12×8

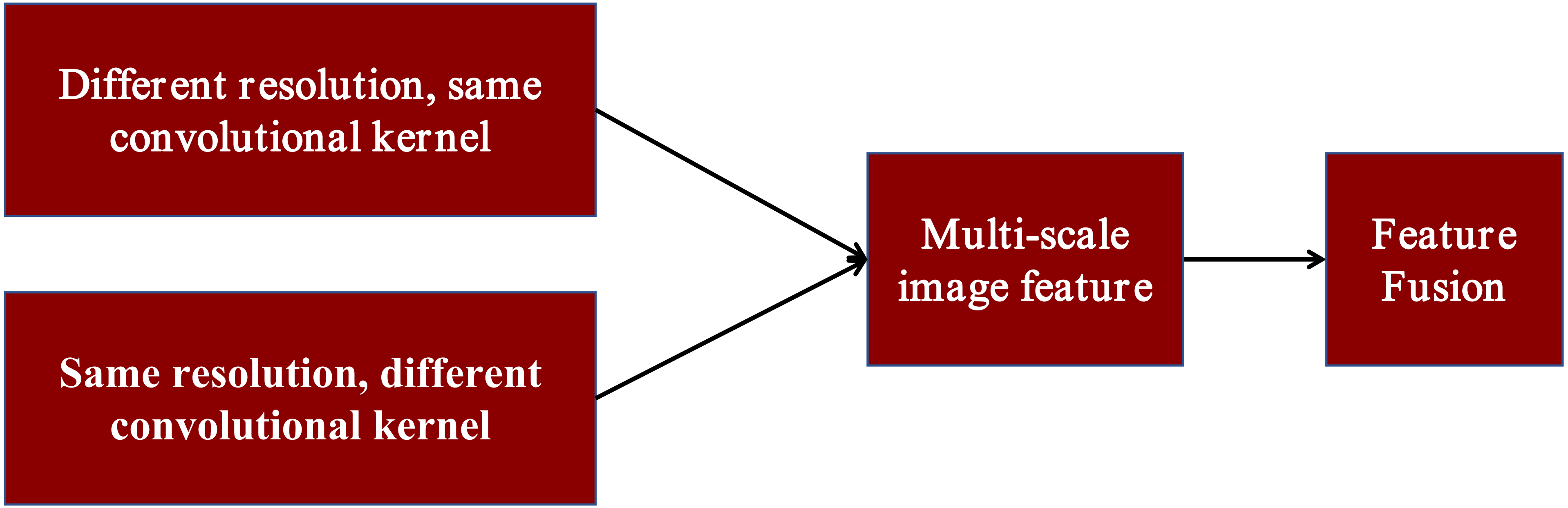
Objects show different details and features in different scale spaces and we can usually observe different features at different scales.

In computer vision, scale is always a big issue, and small objects and large-scale objects often seriously affect performance.

Generally speaking, smaller/dense sampling can see more details, and larger/sparser sampling can see the overall trend.

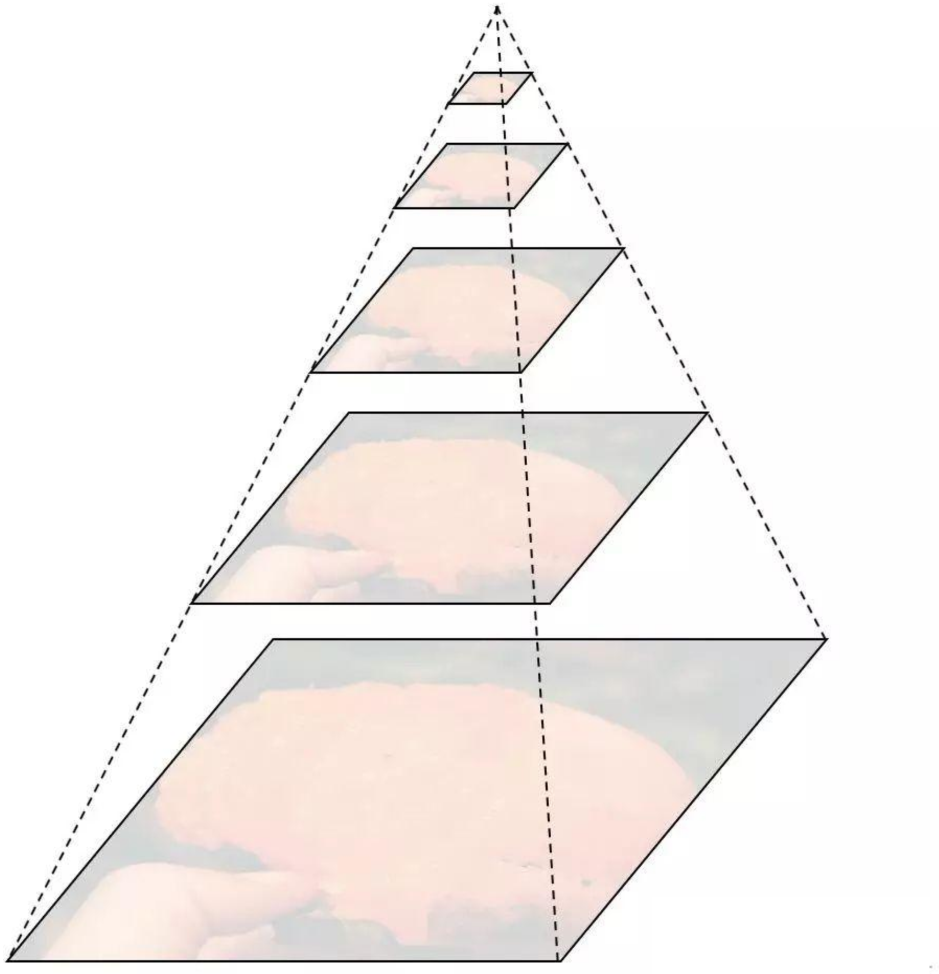
How to extract and utilize multi-scale image features?

--- Change the size of the local receptive field.



Different resolution, same convolutional kernel

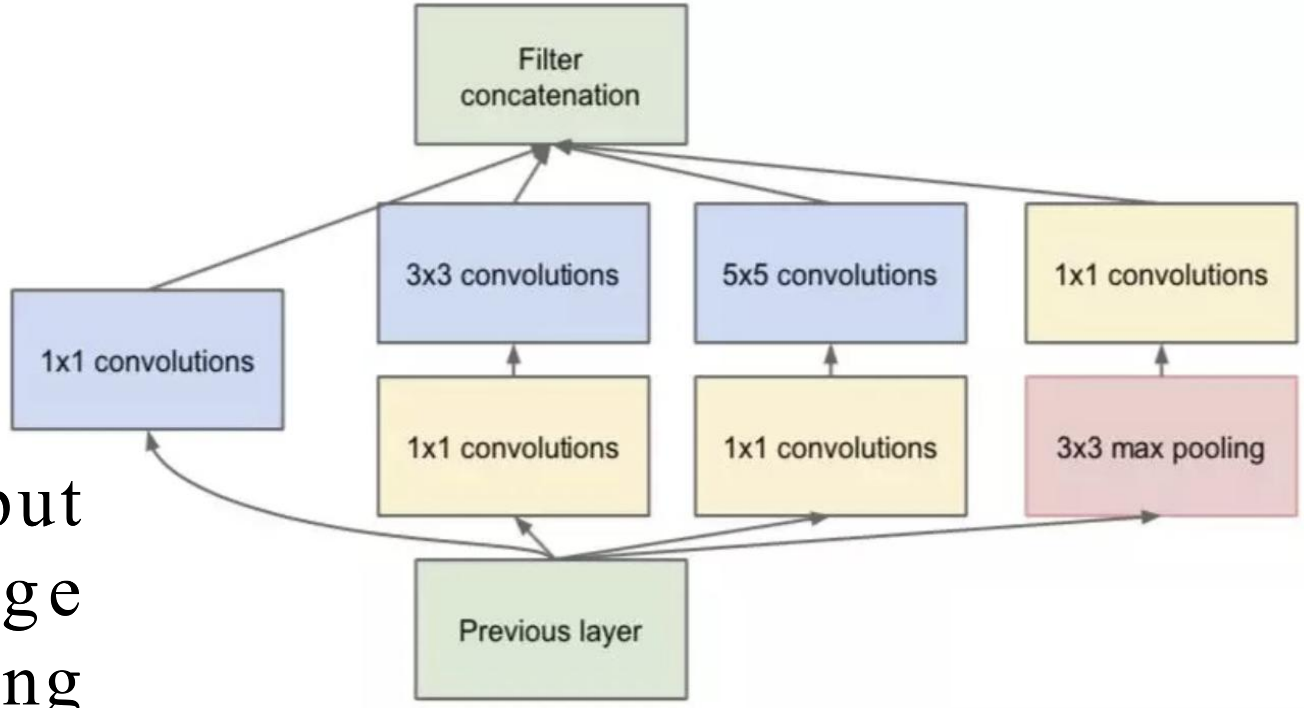
By changing the resolution of the input image to obtain different scale image and multi-scale image features are obtained by applying the same convolutional kernel on the image with different resolutions.



Feature Pyramid

Same resolution, different convolutional kernel

Fixed the resolution of the input image and multi-scale image features are obtained by applying different convolutional kernels on the same resolution image.



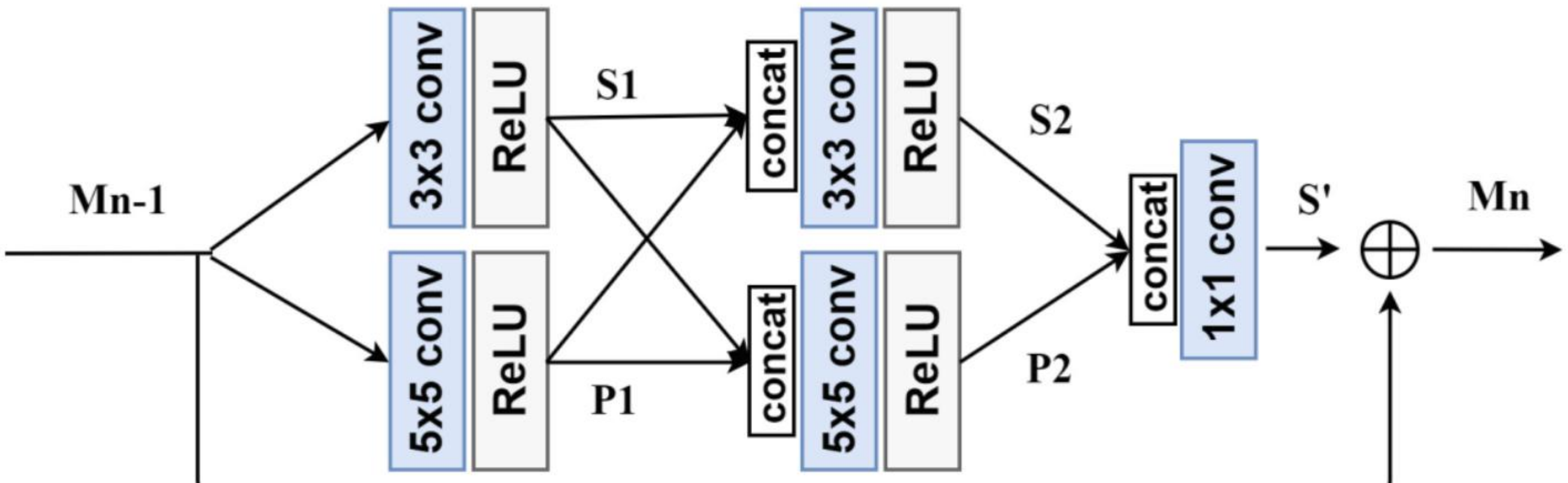
Inception Network

Multi-scale Residual Network for Image Super-Resolution

Juncheng Li[1] Faming Fang[1] Kangfu Mei[2] Guixu Zhang[1]

[1] East China Normal University [2] Jiangxi Normal University

Contact us: *cvjunchengli@gmail.con*



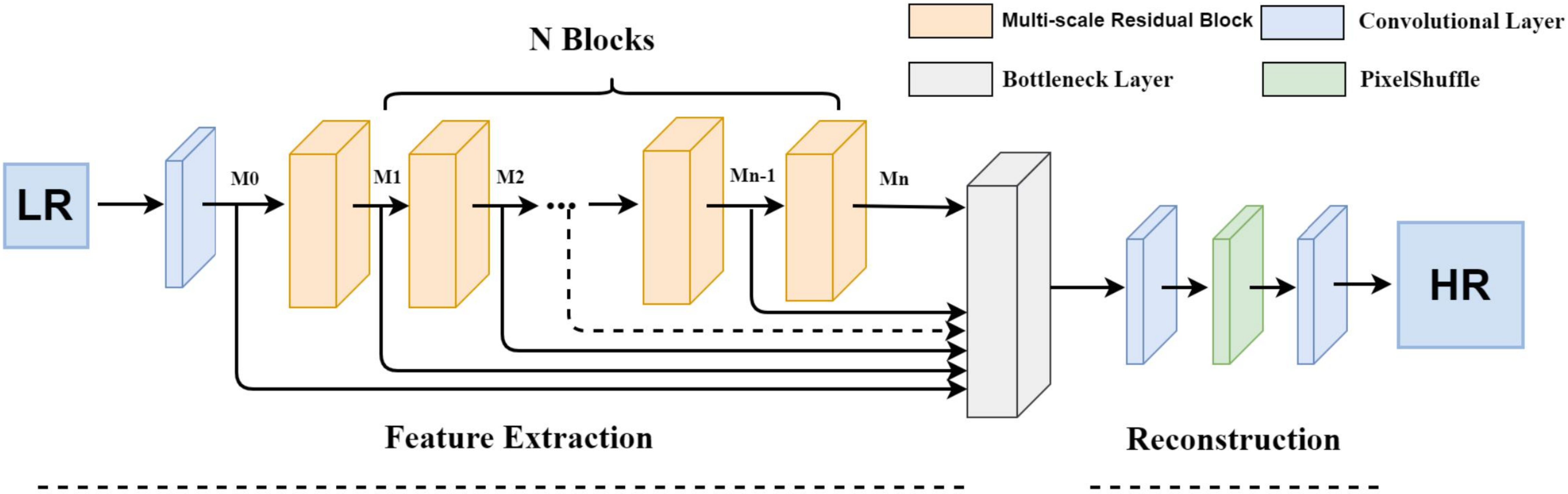
Multi-scale residual block （MSRB）

Contribution

- We propose a novel Multi-scale residual block (MSRB), which can not only adaptively detect the image features, but also achieve feature fusion at different scales. This is the first multi-scale module based on the residual structure. Besides, MSRB can be used for feature extraction in other restoration tasks which show promising results.

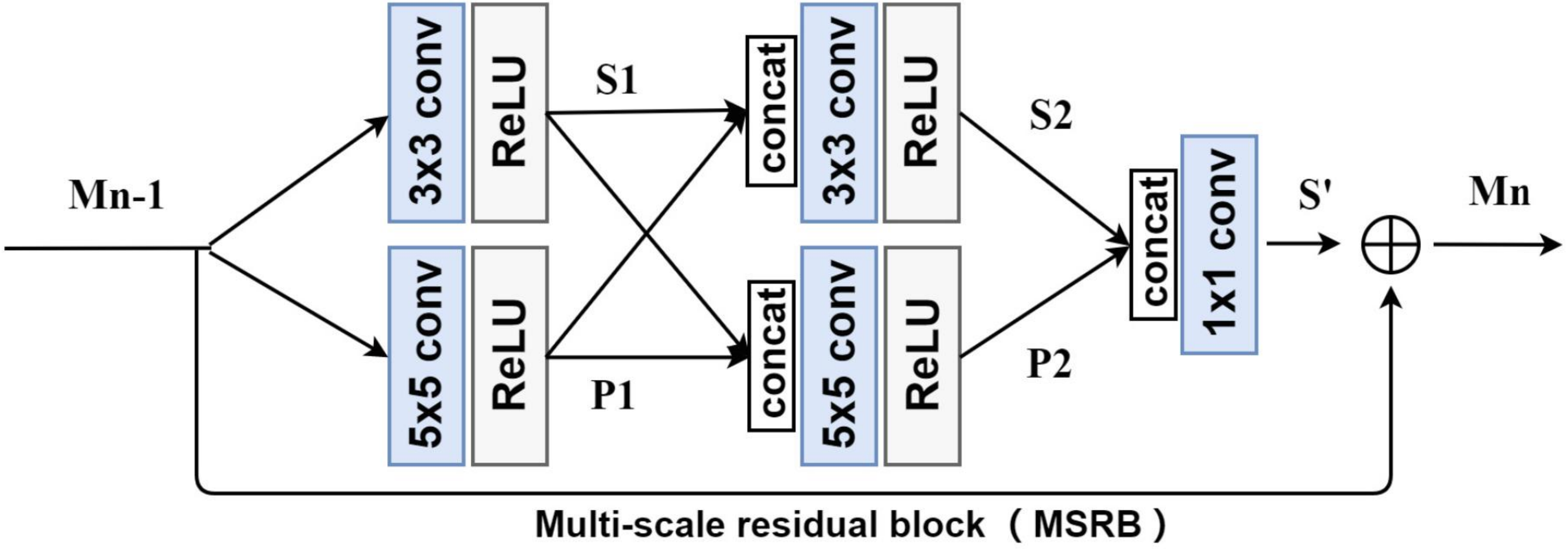
- We propose a simple architecture for hierarchical features fusion (HFFS) and image reconstruction. It can be easily extended to any upscaling factors.

- We propose a Multi-scale residual Network (MSRN) for SISR, which exceeds most of the state-of-the-art methods without deep network structure.



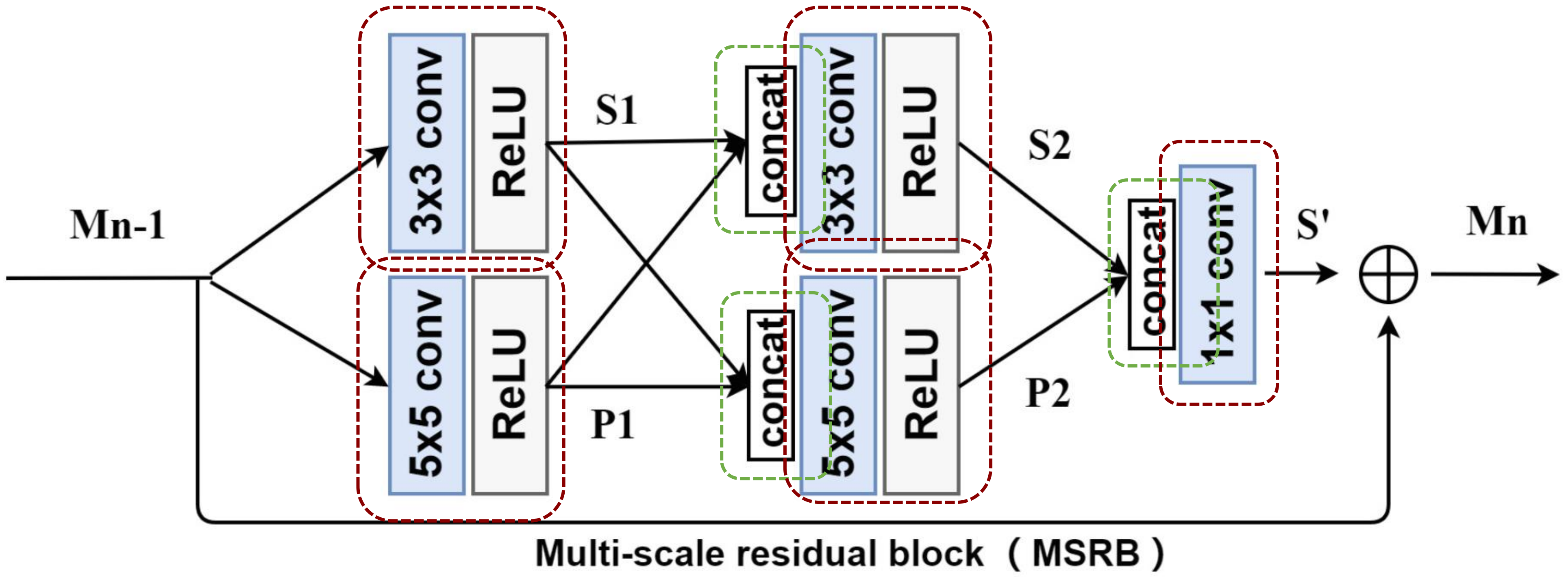
$$\hat{\theta} = arg \min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}^{SR}(F_{\theta}(I_i^{LR}), I_i^{HR})$$

MSRB： Multi-scale feature extraction and fusion.



The structure of multi-scale residual block (MSRB).

MSRB： Multi-scale feature extraction and fusion.



The structure of multi-scale residual block (MSRB).

- **Multi-scale Features Extraction**：

$$S_1 = \sigma(w^1_{3\times3} * M_{n-1} + b^1),$$

$$P_1 = \sigma(w^1_{5\times5} * M_{n-1} + b^1),$$

$$S_2 = \sigma(w^2_{3\times3} * [S_1, P_1] + b^2),$$

- **Multi-scale Features Fusion**：

$$P_2 = \sigma(w^2_{5\times5} * [P_1, S_1] + b^2),$$

$$S^{'} = w^3_{1\times1} * [S_2, P_2] + b^3,$$

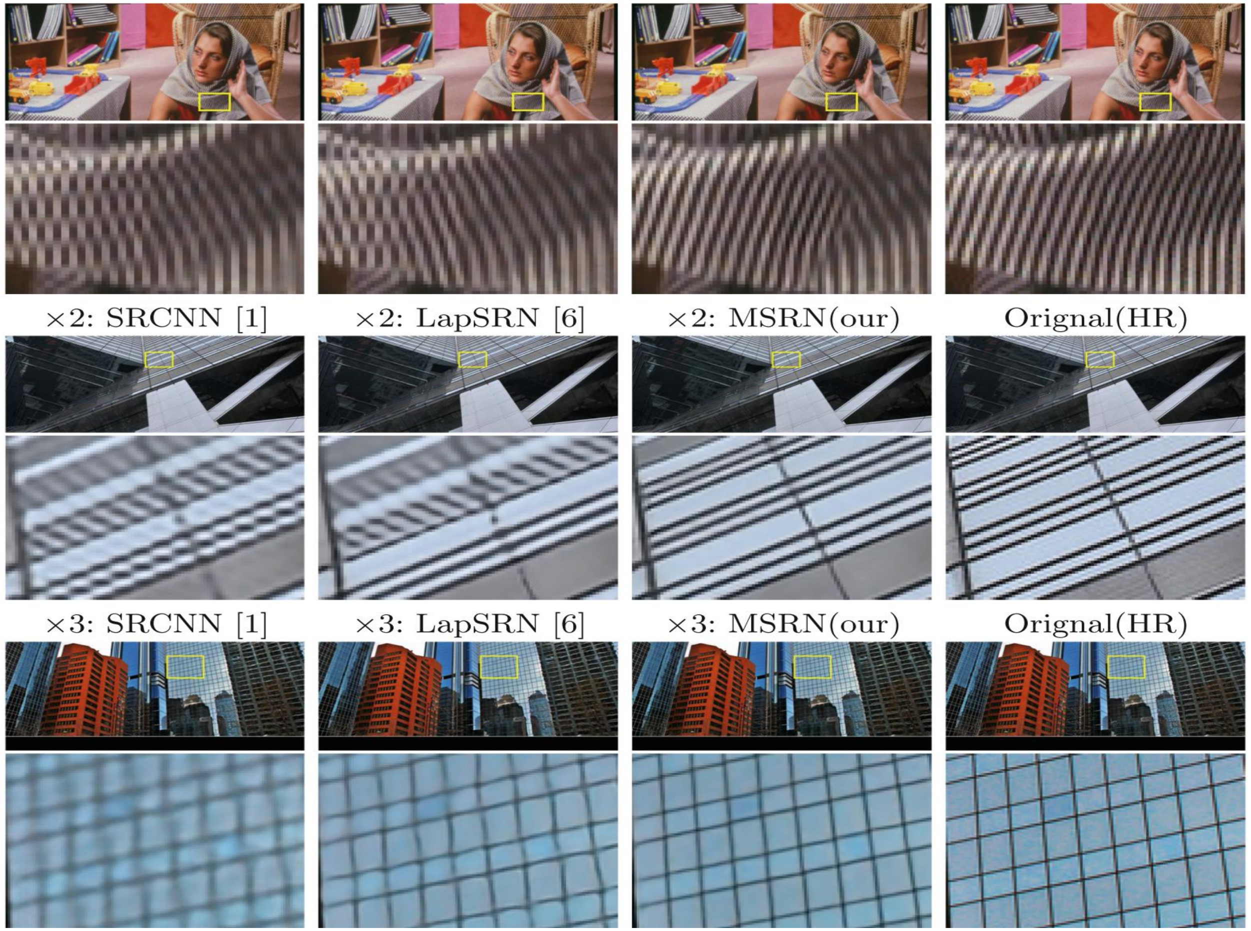
- **Local Residual Learning**：

$$M_n = S^{'} + M_{n-1},$$

MSRN & MDCN

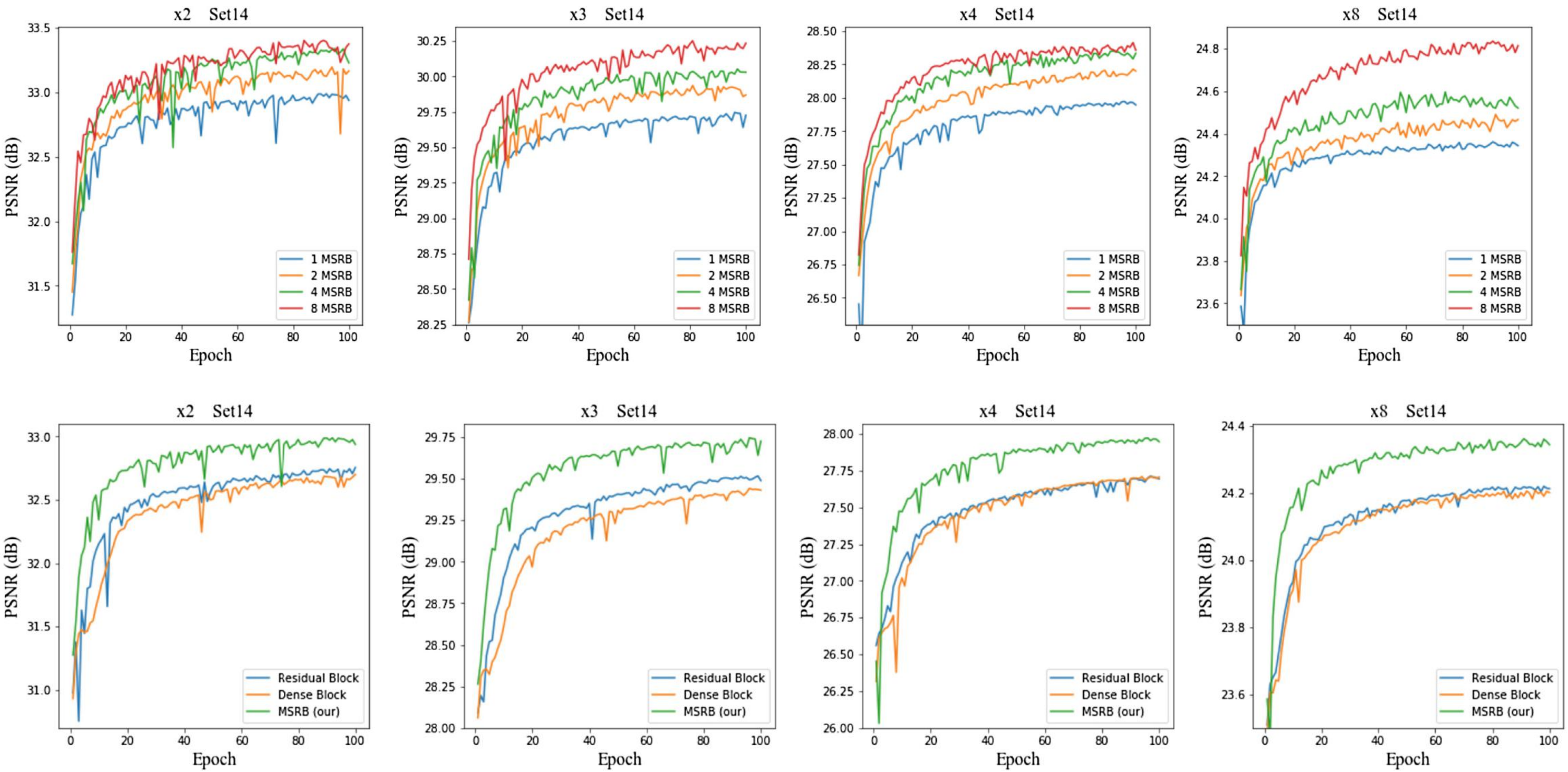
Algorithm	Scale	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSDS100 PSN	Urban100	Manga109
Bicubic	x2	33.69/0.9284	30.34/0.8675	29.5		
A+ [23]	x2	36.60/0.9542	32.42/0.9059	31.2		
SelfExSR [20]	x2	36.60/0.9537	32.46/0.9051	31.2		
SRCNN [1]	x2	36.71/0.9536	32.32/0.9052	31.3		
ESPCN [2]	x2	37.00/0.9559	32.75/0.9098	31.5		
FSRCNN [3]	x2	37.06/0.9554	32.76/0.9078	31.5		
VDSR [4]	x2	37.53/0.9583	33.05/0.9107	31.9		
DRCN [5]	x2	37.63/0.9584	33.06/0.9108	31.8		
LapSRN [6]	x2	37.52/0.9581	33.08/0.9109	31.8		
EDSR [9]	x2	38.11/0.9601	33.92/0.9195	32.3		
MSRN(our)	x2	38.08/0.9605	33.74/0.9170	32.2		
Bicubic	x3	30.41/0.8655	27.64/0.7722	27.2		
A+ [23]	x3	32.63/0.9085	29.25/0.8194	28.3		
SelfExSR [20]	x3	32.66/0.9089	29.34/0.8222	28.3		
SRCNN [1]	x3	32.47/0.9067	29.23/0.8201	28.3		
ESPCN [2]	x3	33.02/0.9135	29.49/0.8271	28.5		
FSRCNN [3]	x3	33.20/0.9149	29.54/0.8277	28.5		
VDSR [4]	x3	33.68/0.9201	29.86/0.8312	28.8		
DRCN [5]	x3	33.85/0.9215	29.89/0.8317	28.8		
LapSRN [6]	x3	33.82/0.9207	29.89/0.8304	28.8		
EDSR [9]	x3	34.65/0.9282	30.52/0.8462	29.2		
MSRN(our)	x3	34.38/0.9262	30.34/0.8395	29.0		

Bicubic	x4	28.43/0.8022	26.10/0.6936	25.97/0.6517	23.14/0.6599	24.91/0.7826
A+ [23]	x4	30.33/0.8565	27.44/0.7450	26.83/0.6999	24.34/0.7211	27.03/0.8439
SelfExSR [20]	x4	30.34/0.8593	27.55/0.7511	26.84/0.7032	24.83/0.7403	27.83/0.8598
SRCNN [1]	x4	30.50/0.8573	27.62/0.7453	26.91/0.6994	24.53/0.7236	27.66/0.8505
ESPCN [2]	x4	30.66/0.8646	27.71/0.7562	26.98/0.7124	24.60/0.7360	27.70/0.8560
FSRCNN [3]	x4	30.73/0.8601	27.71/0.7488	26.98/0.7029	24.62/0.7272	27.90/0.8517
VDSR [4]	x4	31.36/0.8796	28.11/0.7624	27.29/0.7167	25.18/0.7543	28.83/0.8809
DRCN [5]	x4	31.56/0.8810	28.15/0.7627	27.24/0.7150	25.15/0.7530	28.98/0.8816
LapSRN [6]	x4	31.54/0.8811	28.19/0.7635	27.32/0.7162	25.21/0.7564	29.09/0.8845
EDSR [9]	x4	32.46/0.8968	28.80/0.7876	27.71/0.7420	-/-	-/-
MSRN(our)	x4	32.07/0.8903	28.60/0.7751	27.52/0.7273	26.04/0.7896	30.17/0.9034
Bicubic	x8	24.40/0.6045	23.19/0.5110	23.67/0.4808	20.74/0.4841	21.46/0.6138
A+ [23]	x8	25.53/0.6548	23.99/0.5535	24.21/0.5156	21.37/0.5193	22.39/0.6454
SelfExSR [20]	x8	25.49/0.6733	24.02/0.5650	24.19/0.5146	21.81/0.5536	22.99/0.6907
SRCNN [1]	x8	25.34/0.6471	23.86/0.5443	24.14/0.5043	21.29/0.5133	22.46/0.6606
ESPCN [2]	x8	25.75/0.6738	24.21/0.5109	24.37/0.5277	21.59/0.5420	22.83/0.6715
FSRCNN [3]	x8	25.42/0.6440	23.94/0.5482	24.21/0.5112	21.32/0.5090	22.39/0.6357
VDSR [4]	x8	25.73/0.6743	23.20/0.5110	24.34/0.5169	21.48/0.5289	22.73/0.6688
DRCN [5]	x8	25.93/0.6743	24.25/0.5510	24.49/0.5168	21.71/0.5289	23.20/0.6686
LapSRN [6]	x8	26.15/0.7028	24.45/0.5792	24.54/0.5293	21.81/0.5555	23.39/0.7068
MSRN(our)	x8	26.59/0.7254	24.88/0.5961	24.70/0.5410	22.37/0.5977	24.28/0.7517



Algorithm	Feature extraction	Filters	Layers	Depth	Parameters	Updates	Channel
EDSR [9]	**32 blocks**	**256**	**69**	**69**	**43M**	1×10^6	RGB
MSRN (our)	8 blocks	64	44	28	6.3M	4×10^5	Y

MSRN & MDCN

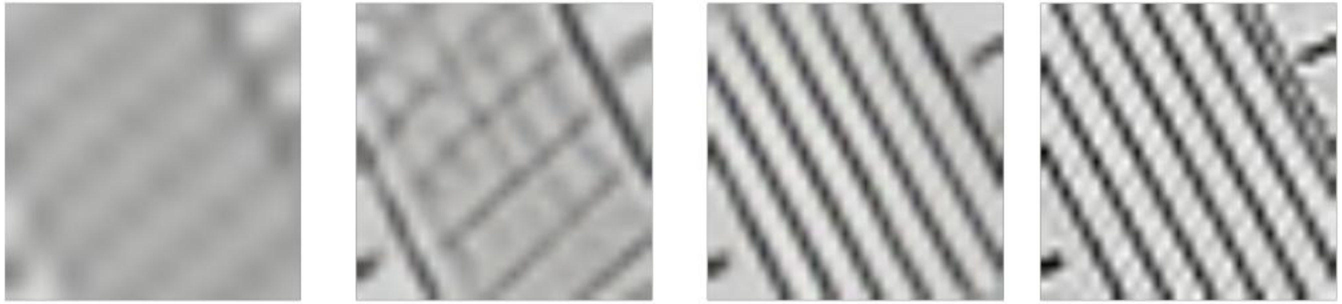

x2 Set14
x3 Set14
x4 Set14
x8 Set14
PSNR (dB)
Epoch
1 MSRB
2 MSRB
4 MSRB
8 MSRB
Residual Block
Dense Block
MSRB (our)

Remaining Problems:

- MSRB cannot obtain feedforward information.
- HFFS will cause a large number of parameters, and this hierarchical feature utilization method will generate a lot of redundant features, which will make the model difficult to train.
- A single model cannot handle different upsampling factors.

MDCN: Multi-scale Dense Cross Network for Image Super-Resolution

Juncheng Li, Faming Fang, Jiaqian Li, Kangfu Mei, and Guixu Zhang



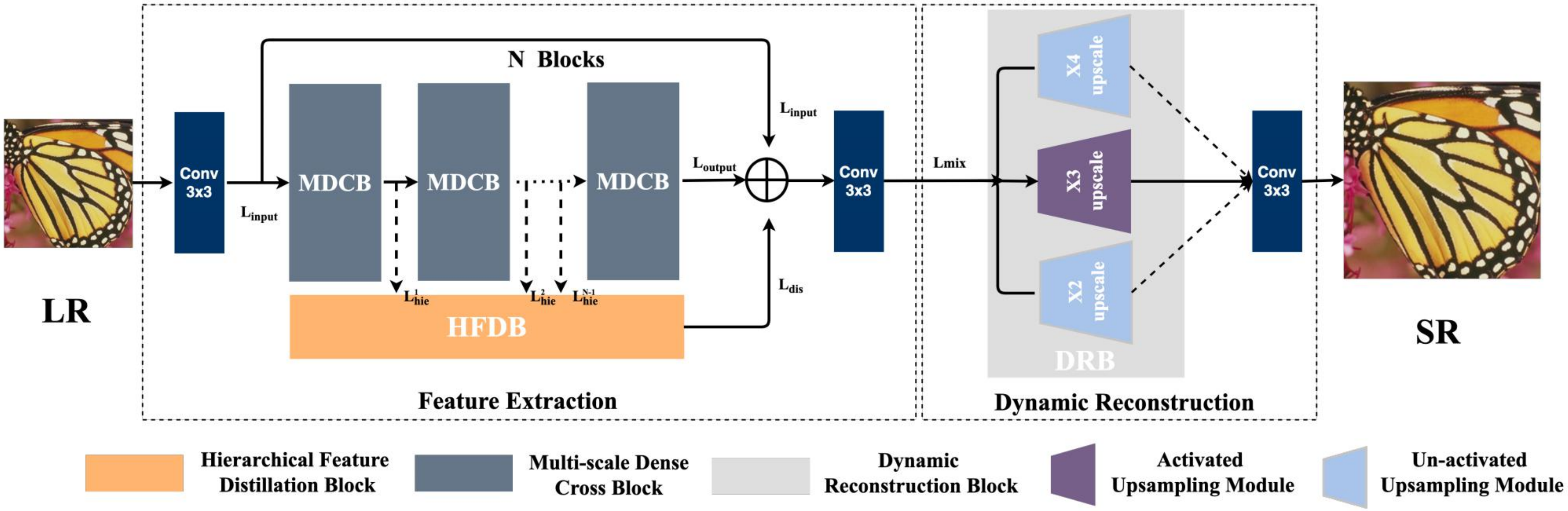
Bicubic MSRN MDCN(Ours) GT

Contribution

- We devise a Multi-scale Dense Cross Block (MDCB) for feature extraction, which essentially a dual-path dense network that can effectively detect local and multi-scale features.

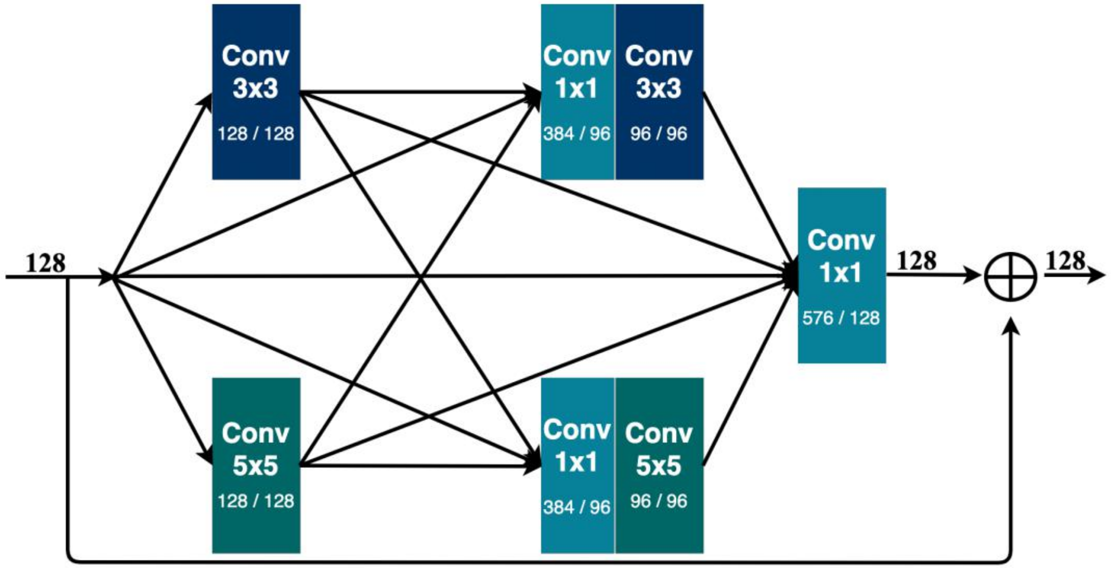
- We design a Hierarchical Feature Distillation Block (HFDB) to maximize the use of hierarchical features. It is the first CNN module specially designed for hierarchical feature learning.

- We introduce a Dynamic Reconstruction Block (DRB) to learn the inter-scale correlation between different upsampling factors, which makes MDCN can reconstruct SR images with different upsampling factors in a single model.



The complete architecture of our proposed Multi-scale Dense Cross Network (MDCN), which consists of two stages: feature extraction and dynamic reconstruction. The dark blue block, orange block, and gray block denote the MDCB, HFDB, and DRB, respectively.

MDCB： Multi-scale feature extraction and fusion.



$$L_{22} = C^1_{3\times3}(L_{11}), H_{22} = C^1_{5\times5}(H_{11}),$$

$$L_{33} = C^2_{3\times3}\left(C^2_{1\times1}([L_{12}, L_{22}, M_{53}])\right),$$

$$H_{33} = C^2_{5\times5}\left(C^2_{1\times1}([H_{12}, H_{22}, M_{35}])\right),$$

$$L_{out} = C^3_{1\times1}\left([L_{23}, L_{33}, H_{23}, H_{33}, L_{in}]\right).$$

MDCB： Multi-scale feature extraction and fusion.

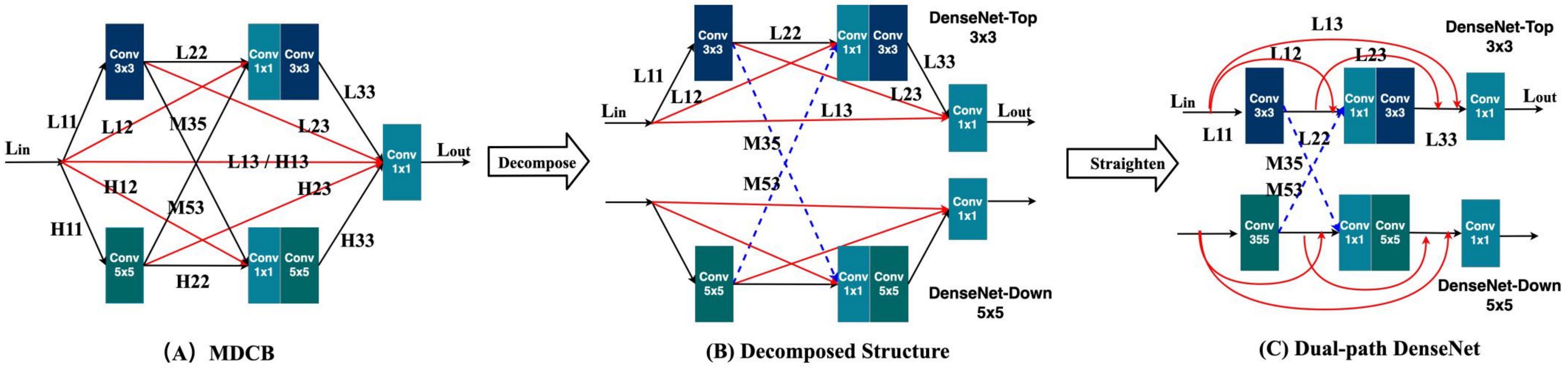
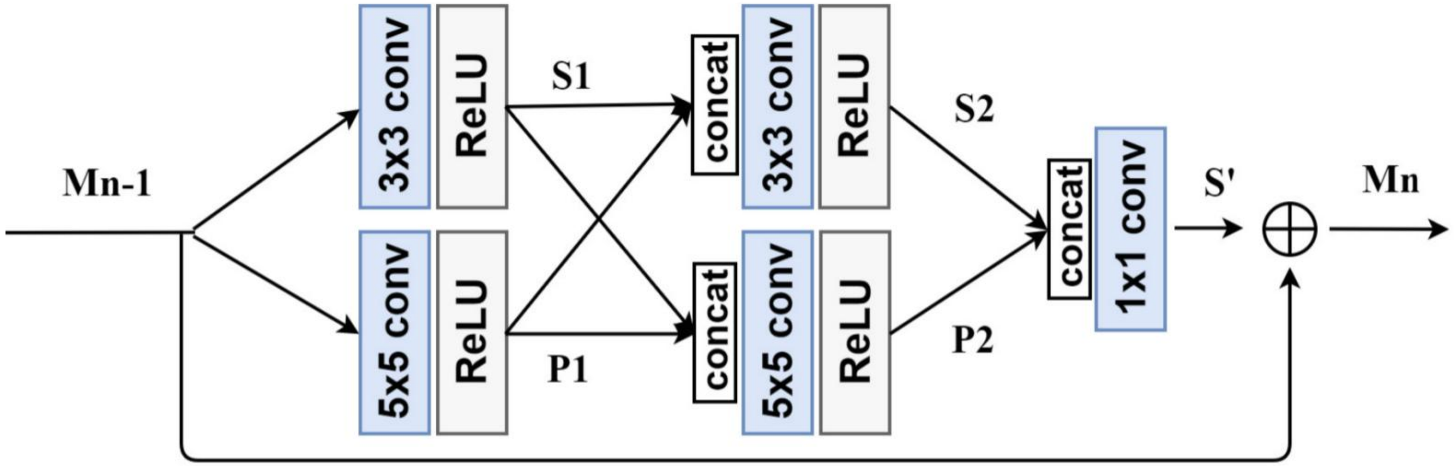


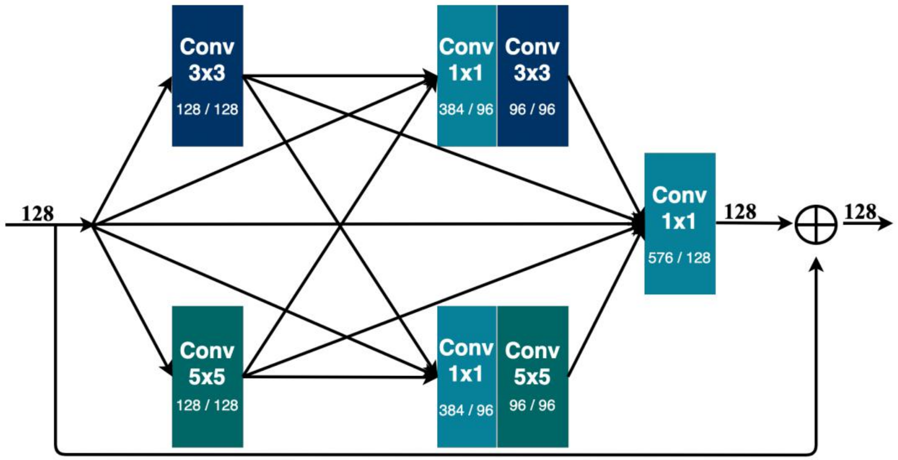
Fig. 6. The decomposed structure of MDCB, which remove the residual learning for better representation. (A) is the MDCB structure that removes residual learning, (B) is the decomposed structure of MDCB, and (C) is the equivalent structure after straightening (B).

The difference between MSRB and MDCB：

MSRB



MDCB



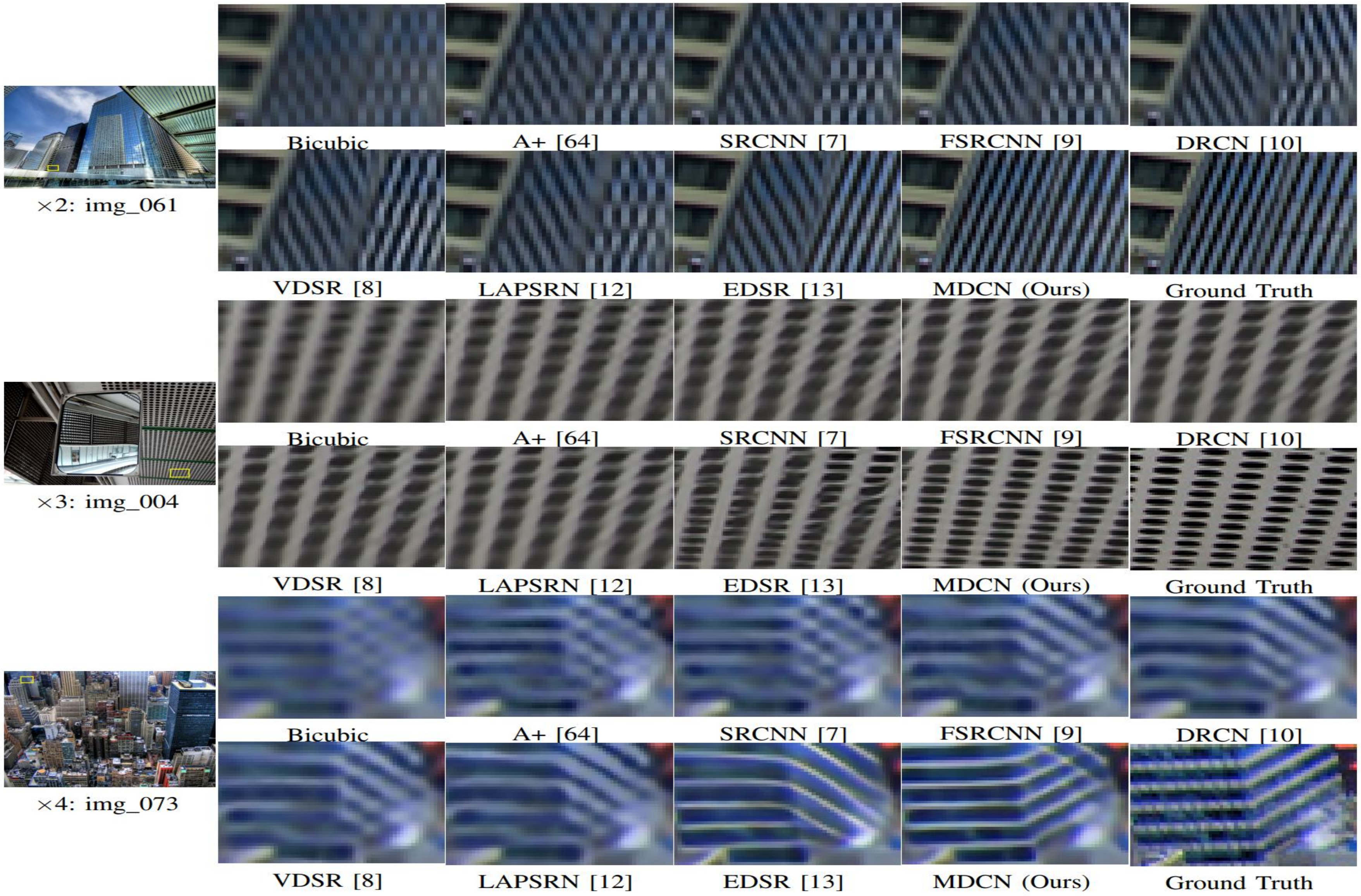
MDCB can obtain more feedforward information.
Fewer parameters, extract richer features, and achieve better performance.

MSRN & MDCN

Algorithm	Scale	Set5 [57] PSNR / SSIM	Set14 [58] PSNR / SSIM	BSDS100 [62] PSNR / SSIM	Urban100 [63] PSNR / SSIM	Manga109 [61] PSNR / SSIM	Average PSNR / SSIM
SRCNN [7] (2014)	×2	36.66 / 0.9542	32.45 / 0.9067	31.36 / 0.8879	29.50 / 0.8946	35.60 / 0.9663	33.11 / 0.9219
ESPCN [11] (2016)	×2	37.00 / 0.9559	32.75 / 0.9098	31.51 / 0.8939	29.87 / 0.9065	36.21 / 0.9694	33.47 / 0.9271
VDSR [8] (2016)	×2	37.53 / 0.9590	33.05 / 0.9130	31.90 / 0.8960	30.77 / 0.9140	37.22 / 0.9750	34.09 / 0.9314
DRCN [10](2016)	×2	37.63 / 0.9584	33.06 / 0.9108	31.85 / 0.8947	30.76 / 0.9147	37.63 / 0.9723	34.19 / 0.9302
LapSRN [12] (2017)	×2	37.52 / 0.9591	33.08 / 0.9130	31.80 / 0.8950	30.41 / 0.9101	37.27 / 0.9740	34.02 / 0.9302
EDSR [13] (2017)	×2	38.11 / 0.9602	33.92 / 0.9195	32.32 / 0.9013	32.93 / 0.9351	39.10 / 0.9773	35.27 / 0.9387
SRMDNF [19] (2018)	×2	37.79 / 0.9					
MSRN [17] (2018)	×2	38.07 / 0.9					
RAN [24] (2019)	×2	37.58 / 0.9					
DNCL [25] (2019)	×2	37.65 / 0.9					
FilterNet [22] (2019)	×2	37.86 / 0.9					
MRFN [21] (2019)	×2	37.98 / 0.9					
SeaNet [26] (2020)	×2	38.08 / 0.9					
MDCN (Ours)	×2	38.19 / 0.9					
MDCN+ (Ours)	×2	**38.25 / 0.9**					
SRCNN [7] (2014)	×3	32.75 / 0.9					
ESPCN [11] (2016)	×3	33.02 / 0.9					
VDSR [8] (2016)	×3	33.67 / 0.9					
DRCN [10] (2016)	×3	33.85 / 0.9					
LapSRN [12] (2017)	×3	33.82 / 0.9					
EDSR [13] (2017)	×3	34.65 / 0.9					
SRMDNF [19] (2018)	×3	34.12 / 0.9					
MSRN [17] (2018)	×3	34.48 / 0.9276	30.40 / 0.8436	29.13 / 0.8061	28.31 / 0.8560	33.56 / 0.9451	31.18 / 0.8757
RAN [24] (2019)	×3	33.71 / 0.9223	29.84 / 0.8326	28.84 / 0.7981	N / A	N / A	N / A
DNCL [25] (2019)	×3	33.95 / 0.9232	29.93 / 0.8340	28.91 / 0.7995	27.27 / 0.8326	N / A	N / A
FilterNet [22] (2019)	×3	34.08 / 0.9250	30.03 / 0.8370	28.95 / 0.8030	27.55 / 0.8380	N / A	N / A
MRFN [21] (2019)	×3	34.21 / 0.9267	30.03 / 0.8363	28.99 / 0.8029	27.53 / 0.8389	32.82 / 0.9396	30.72 / 0.8689
SeaNet [26] (2020)	×3	34.55 / 0.9282	30.42 / 0.8444	29.17 / 0.8071	28.50 / 0.8594	33.73 / 0.9463	31.27 / 0.8771
MDCN (Ours)	×3	34.69 / 0.9294	30.54 / 0.8470	29.26 / 0.8095	28.83 / 0.8662	34.17 / 0.9485	31.50 / 0.8801
MDCN+ (Ours)	×3	**34.76 / 0.9299**	**30.63 / 0.8480**	**29.31 / 0.8103**	**29.00 / 0.8687**	**34.43 / 0.9497**	**31.63 / 0.9913**

SRCNN [7] (2014)	×4	30.48 / 0.8628	27.50 / 0.7513	26.90 / 0.7101	24.52 / 0.7221	27.58 / 0.8555	27.40 / 0.7804
ESPCN [11] (2016)	×4	30.66 / 0.8646	27.71 / 0.7562	26.98 / 0.7124	24.60 / 0.7360	27.70 / 0.8560	27.53 / 0.7850
VDSR [8] (2016)	×4	31.35 / 0.8830	28.02 / 0.7680	27.29 / 0.7267	25.18 / 0.7540	28.83 / 0.8870	28.13 / 0.8037
DRCN [10] (2016)	×4	31.56 / 0.8810	28.15 / 0.7627	27.24 / 0.7150	25.15 / 0.7530	28.98 / 0.8816	28.22 / 0.7987
LapSRN [12] (2017)	×4	31.54 / 0.8850	28.19 / 0.7720	27.32 / 0.7270	25.21 / 0.7560	29.09 / 0.8900	28.27 / 0.8060
EDSR [13] (2017)	×4	32.46 / 0.8968	28.80 / 0.7876	27.71 / 0.7420	26.64 / 0.8033	31.02 / 0.9148	29.33 / 0.8289
SRMDNF [19] (2018)	×4	31.96 / 0.8925	28.35 / 0.7787	27.49 / 0.7337	25.68 / 0.7731	30.09 / 0.9024	28.71 / 0.8161
MSRN [17] (2018)	×4	32.25 / 0.8958	28.63 / 0.7833	27.61 / 0.7377	26.22 / 0.7905	30.57 / 0.9103	29.05 / 0.8235
RAN [24] (2019)	×4	31.43 / 0.8847	28.09 / 0.7691	27.31 / 0.7260	N / A	N / A	N / A
DNCL [25] (2019)	×4	31.66 / 0.8871	28.23 / 0.7717	27.39 / 0.7282	25.36 / 0.7606	N / A	N / A
FilterNet [22] (2019)	×4	31.74 / 0.8900	28.27 / 0.7730	27.39 / 0.7290	25.53 / 0.7680	N / A	N / A
MRFN [21] (2019)	×4	31.90 / 0.8916	28.31 / 0.7746	27.43 / 0.7309	25.46 / 0.7654	29.57 / 0.8962	28.53 / 0.8117
SeaNet [26] (2020)	×4	32.33 / 0.8970	28.72 / 0.7855	27.65 / 0.7388	26.32 / 0.7942	30.74 / 0.9129	29.13 / 0.8257
MDCN (Ours)	×4	32.48 / 0.8985	28.83 / 0.7879	27.74 / 0.7423	26.69 / 0.8049	31.10 / 0.9163	29.37 / 0.8300
MDCN+ (Ours)	×4	**32.61 / 0.9000**	**28.90 / 0.7893**	**27.79 / 0.7434**	**26.86 / 0.8083**	**31.40 / 0.9188**	**29.51 / 0.8320**

MSRN & MDCN


×2: img_061
Bicubic
A+ [64]
SRCNN [7]
FSRCNN [9]
DRCN [10]
VDSR [8]
LAPSRN [12]
EDSR [13]
MDCN (Ours)
Ground Truth
×3: img_004
Bicubic
A+ [64]
SRCNN [7]
FSRCNN [9]
DRCN [10]
VDSR [8]
LAPSRN [12]
EDSR [13]
MDCN (Ours)
Ground Truth
×4: img_073
Bicubic
A+ [64]
SRCNN [7]
FSRCNN [9]
DRCN [10]
VDSR [8]
LAPSRN [12]
EDSR [13]
MDCN (Ours)
Ground Truth

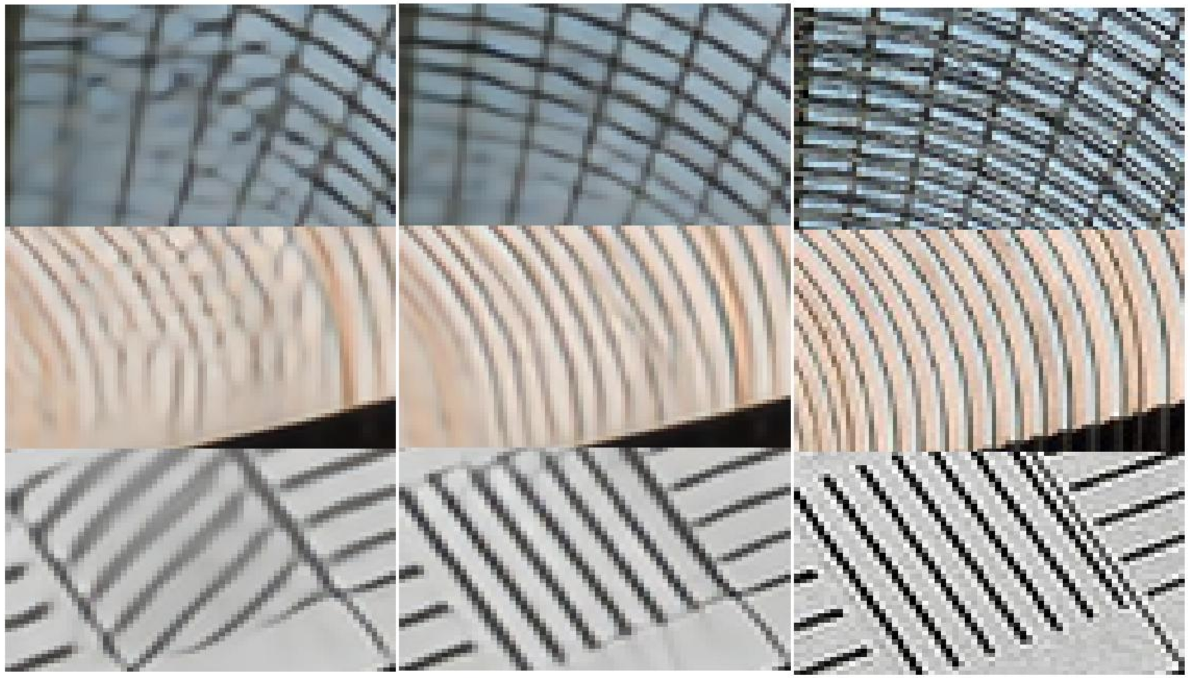
MSRN VS MDCN： MDCN achieves better results with fewer parameters

TABLE VI
REPLACE THE MSRB IN THE MSRN WITH MDCB TO GET THE MDCN'. MDCN' ACHIEVES BETTER RESULTS WITH FEWER PARAMETERS.

Methods	MSRN			MDCN'(Ours)		
Scale	x2	x3	x4	x2	x3	x4
Parameters	5.92M	6.11M	6.07M	**4.34M** ↓	**4.52M** ↓	**4.48M** ↓
Set5	38.07/0.9608	34.48/0.9276	32.25/0.8958	**38.10/0.9608**	**34.52/0.9278**	**32.30/0.8965**
Set14	33.68/0.9184	30.40/0.8436	28.63/0.7833	**33.74/0.9186**	**30.45/0.8444**	**28.68/0.7844**
BSD100	32.22/0.9002	29.13/0.8061	27.61/0.7377	**32.23/0.9003**	**29.16/0.8067**	**27.63/0.7383**
Urban100	32.32/0.9304	28.31/0.8560	26.20/0.7905	**32.34/0.9304**	**28.39/0.8575**	**26.26/0.7918**
Manga109	38.64/0.9771	33.56/0.9451	30.57/0.9103	**38.73/0.9774**	**33.77/0.9462**	**30.68/0.9119**
Average	34.99/0.9374	31.18/0.8754	29.05/0.8235	**35.03/0.9375 (0.04/0.0001 ↑)**	**31.26/0.8765 (0.08/0.0011 ↑)**	**29.11/0.8246 (0.06/0.0011 ↑)**

TABLE VII
REPLACE THE MDCB IN THE MDCN WITH MSRB TO GET THE MSRN'. MDCN ACHIEVES BETTER RESULTS WITH FEWER PARAMETERS.

Methods	MSRN'(x2,x3,x4)			MDCN (x2,x3,x4, Ours)		
Scale	x2	x3	x4	x2	x3	x4
Parameters	16.77M			**15.62M** ↓		
Set5	38.07/0.9608	34.51/0.9279	32.32/0.8967	**38.19/0.9612**	**34.69/0.9294**	**32.48/0.8985**
Set14	33.78/0.9192	30.45/0.8445	28.71/0.7850	**33.86/0.9202**	**30.54/0.8470**	**28.83/0.7988**
BSD100	32.23/0.9001	29.17/0.8070	27.66/0.7391	**32.32/0.9014**	**29.26/0.8095**	**27.74/0.7423**
Urban100	32.43/0.9314	28.43/0.8584	26.35/0.7946	**32.92/0.9355**	**28.83/0.8662**	**26.69/0.8049**
Manga109	38.55/0.9779	33.72/0.9461	30.75/0.9120	**39.09/0.9780**	**34.17/0.9485**	**31.10/0.9163**
Average	35.01/0.9379	31.26/0.8768	29.16/0.8255	**35.28/0.9393 (0.27/0.0014 ↑)**	**31.40/0.8801 (0.14/0.0033 ↑)**	**29.37/0.8322 (0.21/0.0067 ↑)**

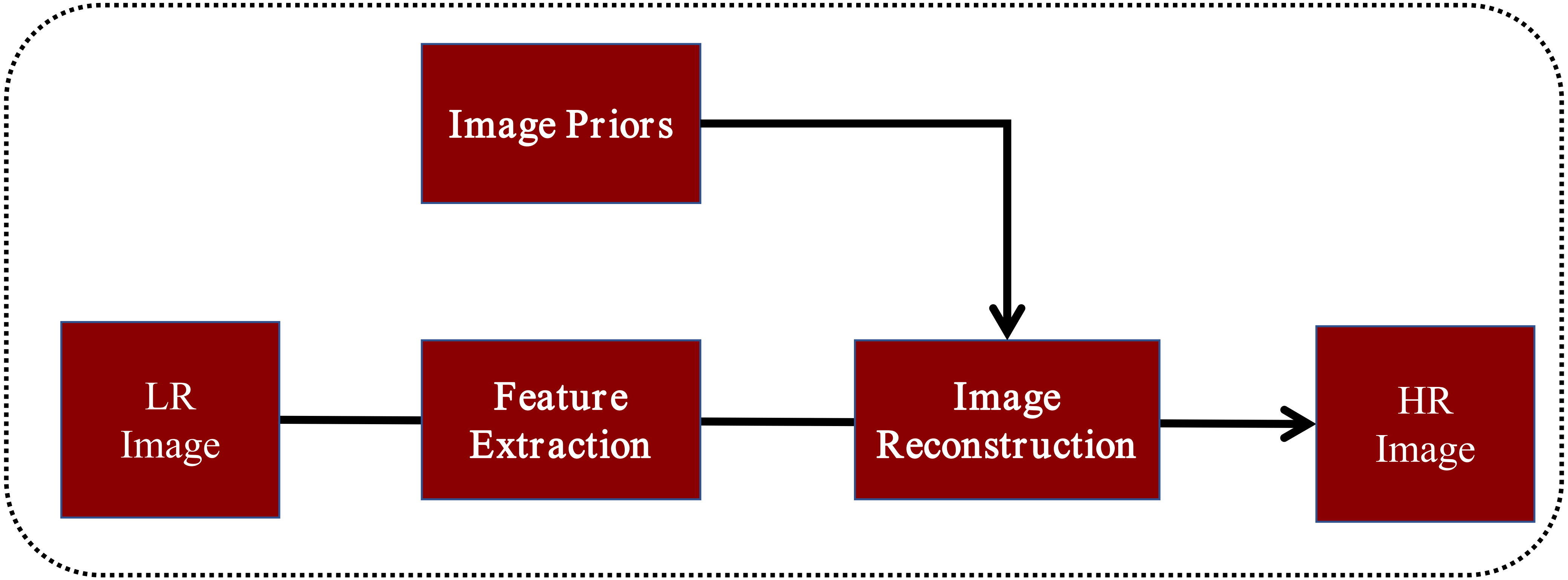


MSRN [17] MDCN (Ours) Ground Truth

Fig. 11. Visual comparison between MSRN and MDCN (×4).

04

Edge Priors Guidance SeaNet & MLEFGN



The process of image restoration.

Image restoration guided by image priors

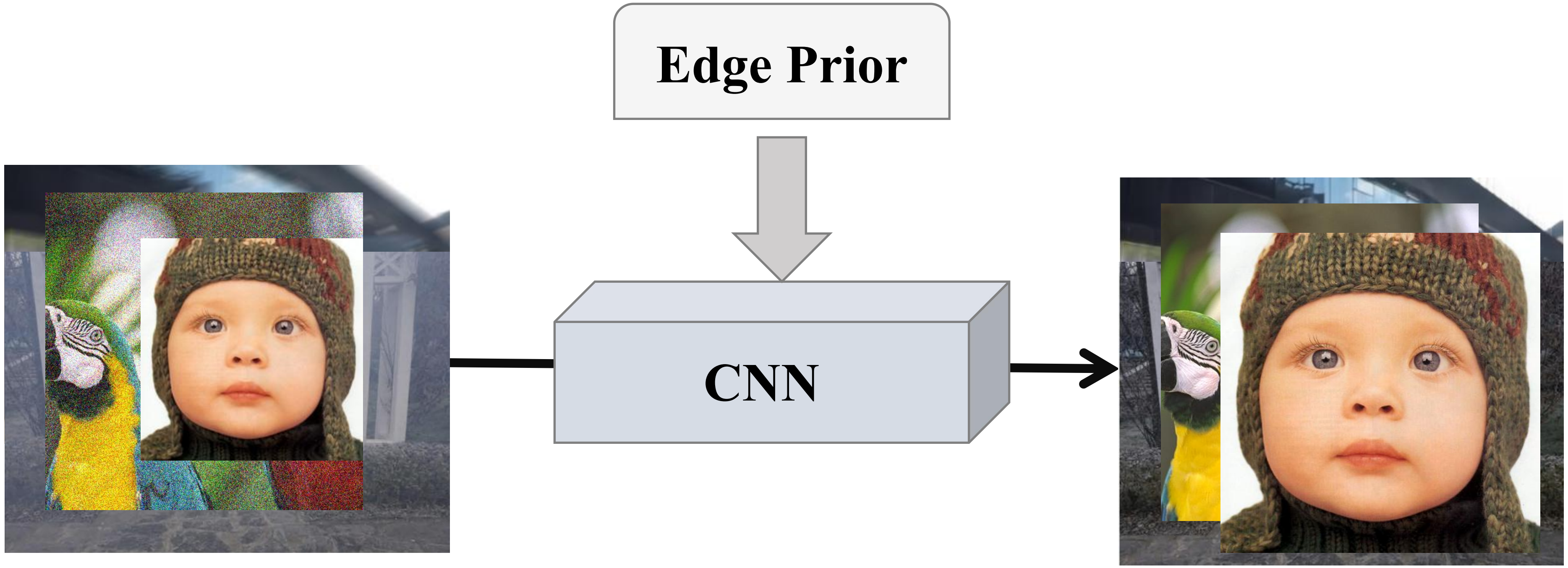
Plenty of works have pointed out that **prior knowledge can effectively assist image restoration**. Accordingly, many image priors have been proposed and used, such as :

- Total Variation Prior
- Sparse Prior
- Edge Prior

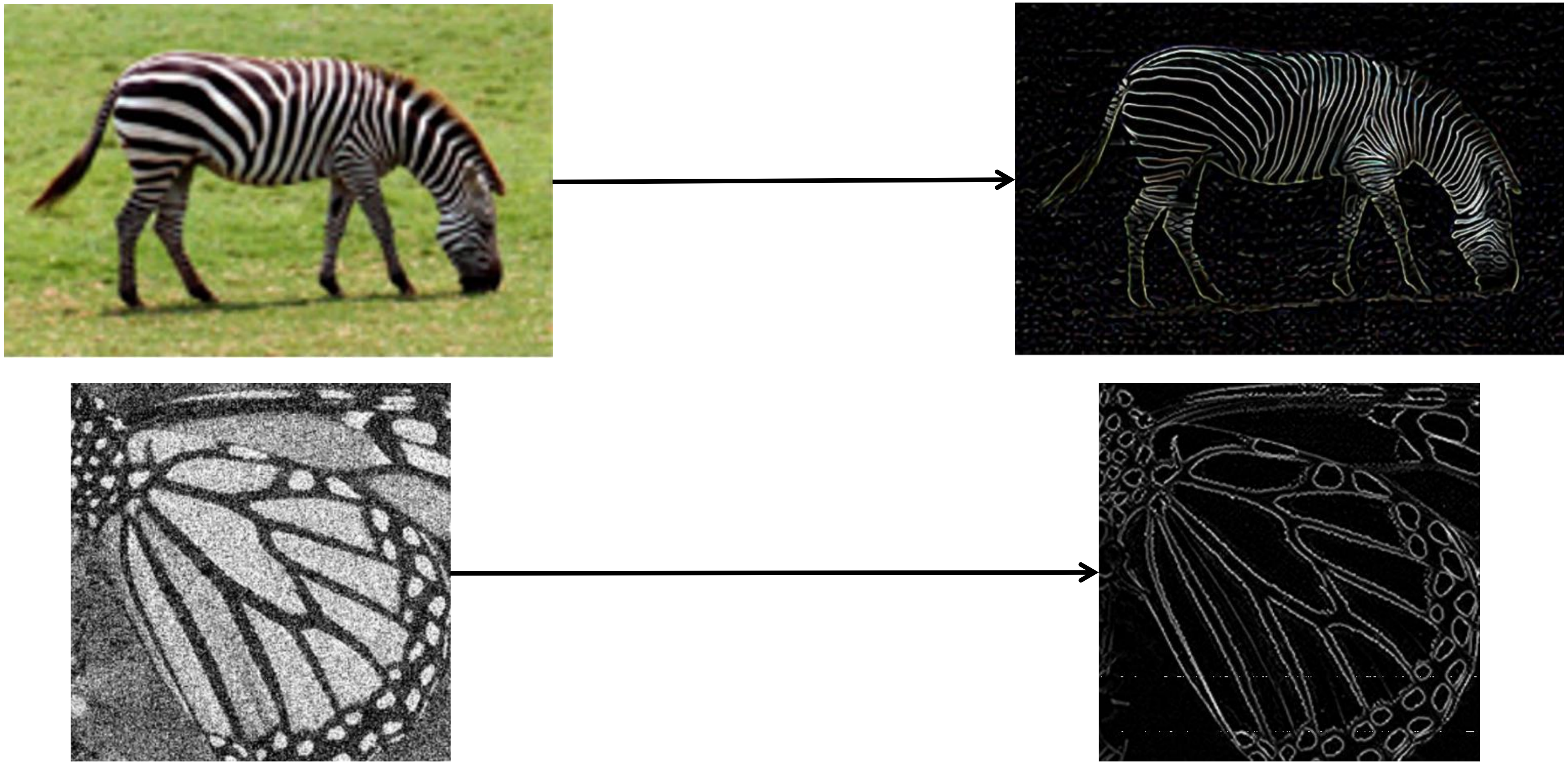
Image restoration guided by image priors

- TV prior will smooth texture details.
- Sparse prior is difficult to model because it requires other domain knowledge.
- Edge prior is one of the most effective priors since image edges are important high-frequency features.

The focus of our research.



Therefore, we aim to introduce edge priors to guide image restoration.

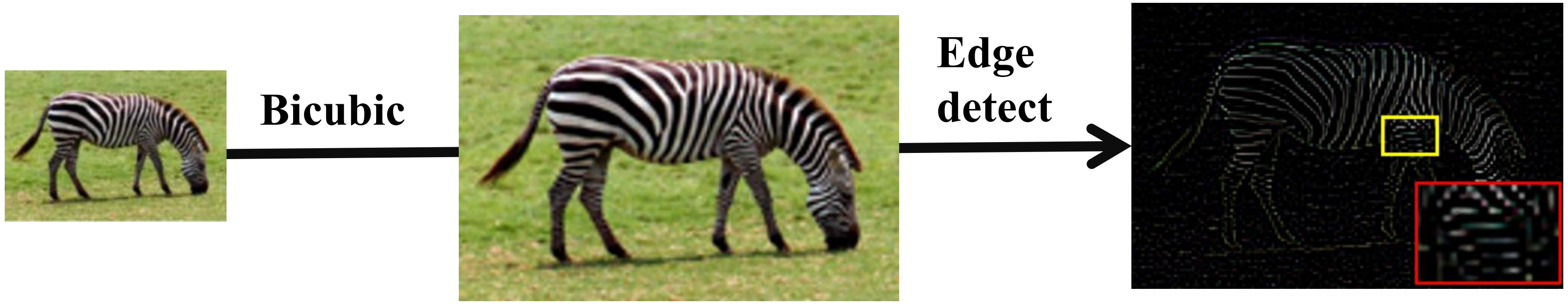


Original Image Image Edge

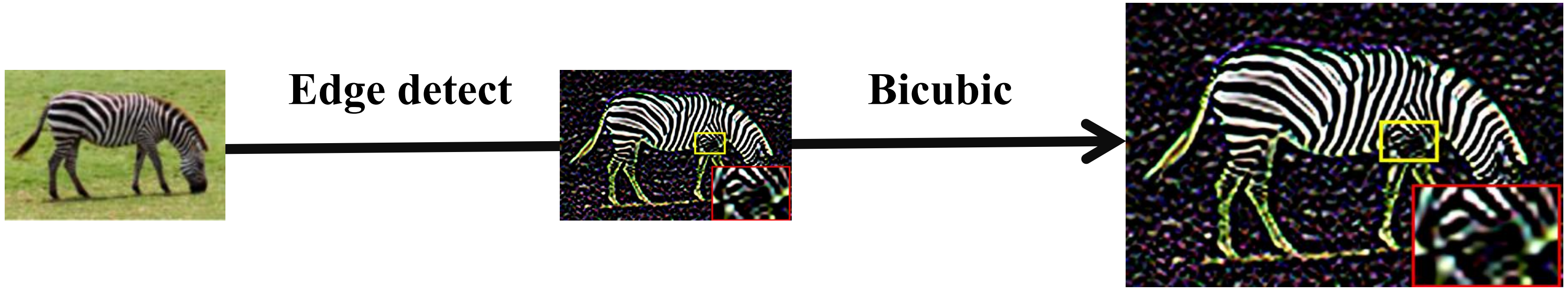
The points where the brightness of an image changes drastically are usually organized into a set of curve segments called **image edges**.

How to obtain imge edges?

The most widely used method is to apply off-the-shelf edge detectors on the degraded image to obtain image edges.

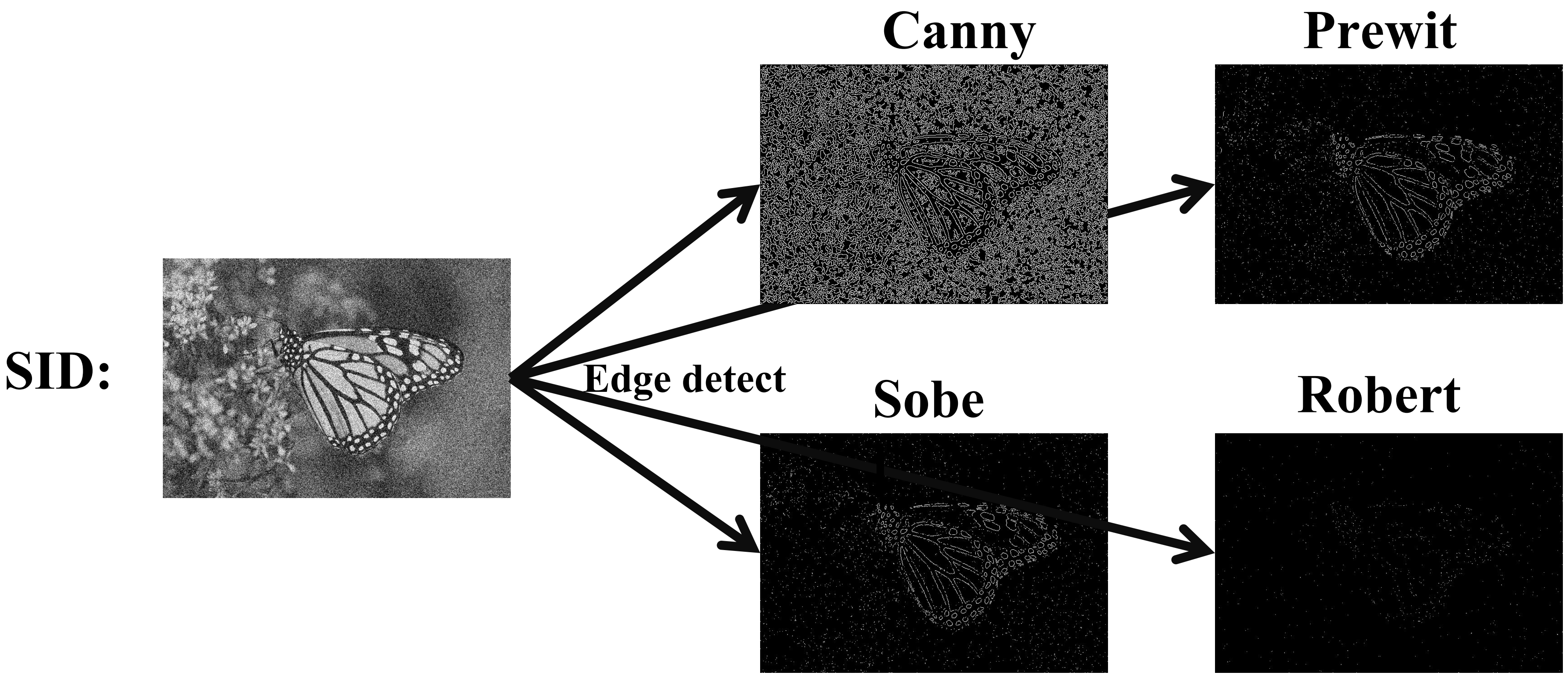


SISR:



How to obtain imge edges?

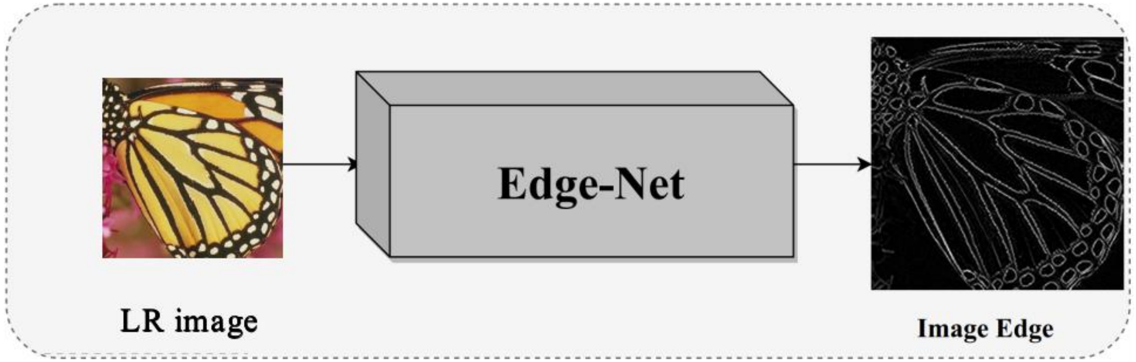
The most widely used method is to apply off-the-shelf edge detectors on the degraded image to obtain image edges.



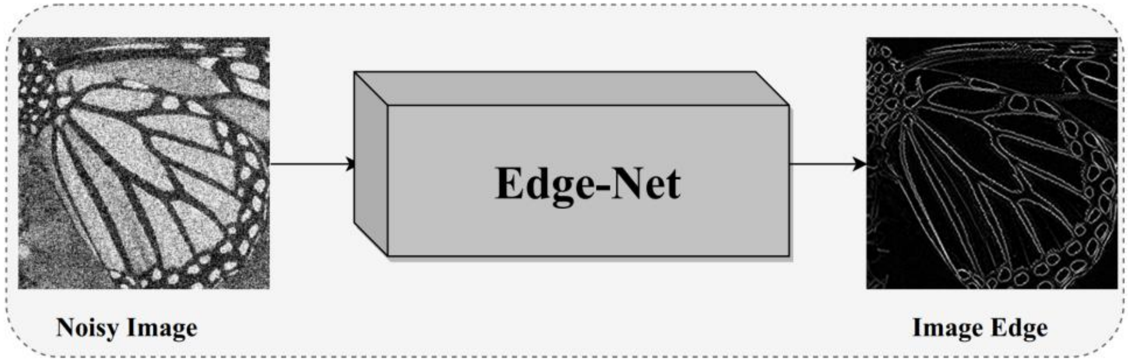
Remaining Problems:

- Existing edge extractors are **extremely sensitive to noise** or other interference factors.
- It is **extremely difficult to obtain clear and accurate edges** from the degraded image using **off-the-shelf edge operators**.
- **Inaccurate edges** will **interfere** with the quality of the reconstructed images.
- Existing edge detectors use the binarization measurement to convert all the values of the edges to 0 and 1, which results in **the loss of a great number of image features** and the appearance of **false edges**.

Solution: Edge-Net



SISR



SID

We aim to explore a CNN model that can reconstruct clear and accurate soft-edges from the degraded image, thus it can use for image restoration.

Solution: Soft-edge

$$I_{Edge} = div(u_x, u_y),$$

$$u_i = \frac{\nabla_i I_{HR}}{\sqrt{1 + |\nabla I_{HR}|^2}}, i \in \{x, y\}$$

We suggest using image soft-edge instead of image edge since it can **retain more accurate image edge information.**

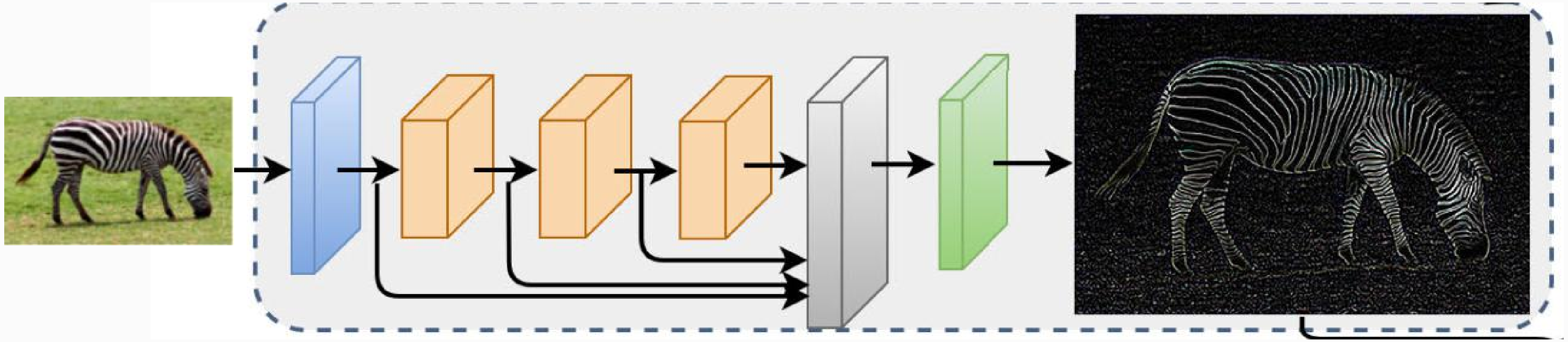
Soft-Edge Assisted Network for Single Image Super-Resolution

Faming Fang, Juncheng Li, and Tieyong Zeng

Contribution

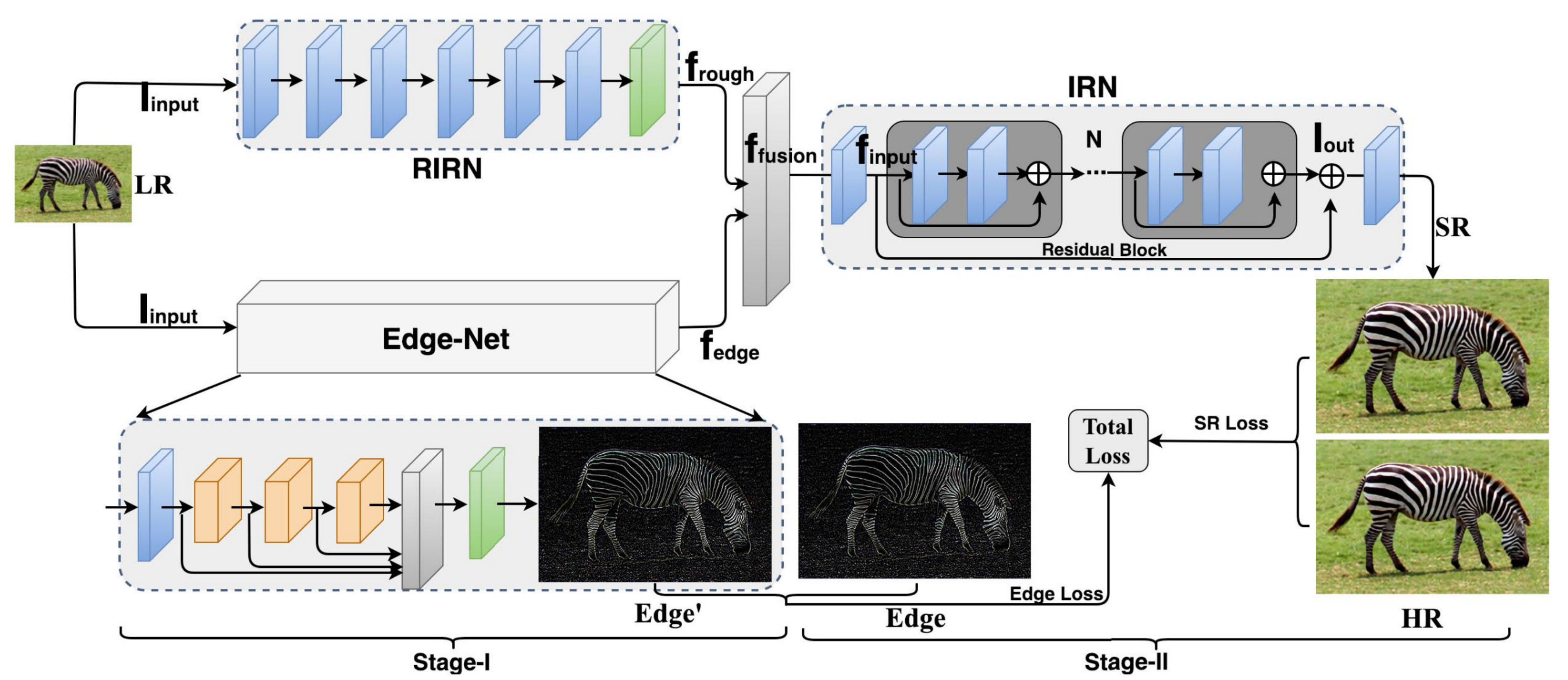
- We verified the importance and effectiveness of edge prior for SISR, and suggested using image soft-edge instead of image edge to obtain more information.
- We propose a soft-edge reconstruction network (EdgeNet), which is the first CNN model used to reconstruct the image soft-edge directly from the LR image.
- We propose an efficient and accurate Soft-edge assisted Network (SeaNet), which is a well-designed network that introduces the Edge-Net to provide image soft-edge prior.

Edge-Net:



Edge-Net can be used as part of any SR model to provide image soft-edge or works independently to reconstruct a super-resolution image soft-edge from the LR image directly. We define the edge loss as

$$\mathcal{L}_{edge} = \left\| E(I_{LR}) - I_{Edge} \right\|_1$$



$$\hat{\theta} = \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \left\| F(I_{LR}^{i}) - I_{HR}^{i} \right| + \lambda \left\| E(I_{LR}^{i}) - I_{Edge}^{i} \right\|_1$$

TABLE I

QUANTITATIVE COMPARISONS OF THE STATE-OF-THE-ART SR METHODS. ALL OF THESE METHODS ARE TRADITIONAL MATHEMATICAL MODELS OR MODELS THAT INTRODUCE IMAGE PRIORS INTO CNN FOR SR IMAGE RECONSTRUCTION. NOTICE THAT, DEGREE-MV USES THE MULTI-VIEW TESTING STRATEGY TO IMPROVE PERFORMANCE. BEST RESULTS ARE **HIGHLIGHTED**

Algorithm	Scale	Set5 [37] PSNR / SSIM	Set14 [38] PSNR / SSIM	BSDS100 [39] PSNR / SSIM	Image Priors
Bicubic	$\times 2$	33.69 / 0.9284	30.34 / 0.8675	29.57 / 0.8434	-
A+ [32]	$\times 2$	36.60 / 0.9542	32.42 / 0.9059	31.24 / 0.8870	Neighbor embedding
SelfExSR [40]	$\times 2$	36.60 / 0.9537	32.46 / 0.9051	31.20 / 0.8863	Transformed self-exemplars
CSCN-MV [27]	$\times 2$	37.14 / 0.9567	32.56 / 0.9074	31.40 / 0.8840	Sparse prior
DEGREE-MV [29]	$\times 2$	37.61 / 0.9589	33.11 / 0.9129	31.84 / 0.8951	Edge prior
SeaNet (Ours)	$\times 2$	**38.08 / 0.9609**	**33.75 / 0.9190**	**32.27 / 0.9008**	Edge prior
Bicubic	$\times 3$	30.41 / 0.8655	27.64 / 0.7722	27.21 / 0.7344	-
A+ [32]	$\times 3$	32.63 / 0.9085	29.25 / 0.8194	28.31 / 0.7828	Neighbor embedding
SelfExSR [40]	$\times 3$	32.66 / 0.9089	29.34 / 0.8222	28.30 / 0.7839	Transformed self-exemplars
CSCN-MV [27]	$\times 3$	33.26 / 0.9167	29.55 / 0.8271	28.50 / 0.7885	Sparse prior
DEGREE-MV [29]	$\times 3$	33.70 / 0.9212	29.77 / 0.8309	28.76 / 0.7956	Edge prior
SeaNet (Ours)	$\times 3$	**34.55 / 0.9282**	**30.42 / 0.8445**	**29.17 / 0.8071**	Edge prior
Bicubic	$\times 4$	28.43 / 0.8022	26.10 / 0.6936	25.97 / 0.6517	-
A+ [32]	$\times 4$	30.33 / 0.8565	27.44 / 0.7450	26.83 / 0.6999	Neighbor embedding
SelfExSR [40]	$\times 4$	30.34 / 0.8593	27.55 / 0.7511	26.84 / 0.7032	Transformed self-exemplars
CSCN-MV [27]	$\times 4$	31.04 / 0.8775	27.76 / 0.7620	27.11 / 0.7191	Sparse prior
DEGREE-MV [29]	$\times 4$	31.30 / 0.8968	27.92 / 0.7637	27.18 / 0.7207	Edge prior
SeaNet (Ours)	$\times 4$	**32.33 / 0.8970**	**28.72 / 0.7855**	**27.65 / 0.7388**	Edge prior

SeaNet & MLEFGN

QUANTITATIVE COMPARISONS OF THE STATE-OF-THE-ART SR METHODS. ALL OF THESE METHODS ARE BASED ON CNN WITHOUT IMAGE PRIORS. BEST RESULTS ARE **HIGHLIGHTED** AND SECOND BEST RESULTS ARE <u>UNDERLINED</u>

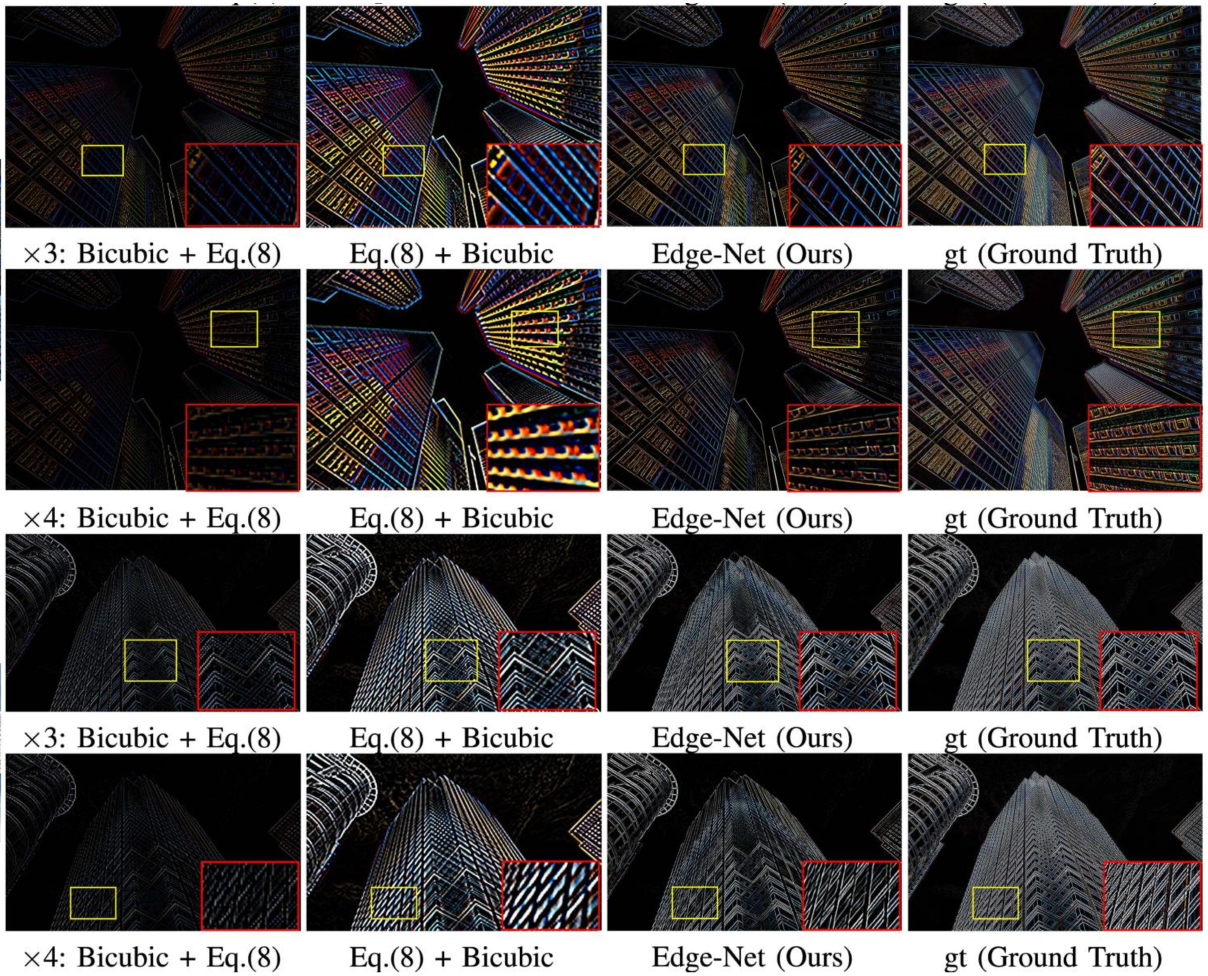
Algorithm	Scale	Set5 [37] PSNR / SSIM	Set14 [38]	BSDS100 [39]	Urban100 [40]	Manga109 [41]
SRCNN [7]	×2	36.71 / 0.9536				
ESPCN [9]	×2	37.00 / 0.9559				
FSRCNN [8]	×2	37.06 / 0.9554				
VDSR [10]	×2	37.53 / 0.9583				
DRCN [11]	×2	37.63 / 0.9584				
LapSRN [13]	×2	37.52 / 0.9581				
DRRN [12]	×2	37.74 / 0.9590				
SeaNet (Ours)	×2	<u>38.08 / 0.9609</u>				
SeaNet+ (Ours)	×2	**38.15 / 0.9611**				
SRCNN [7]	×3	32.47 / 0.9067				
ESPCN [9]	×3	33.02 / 0.9135				
FSRCNN [8]	×3	33.20 / 0.9149				
VDSR [10]	×3	33.68 / 0.9201				
DRCN [11]	×3	33.85 / 0.9215				
LapSRN [13]	×3	33.82 / 0.9207				
DRRN [12]	×3	34.03 / 0.9240				
SeaNet (Ours)	×3	<u>34.55 / 0.9282</u>				
SeaNet+ (Ours)	×3	**34.65 / 0.9290**				
SRCNN [7]	×4	30.50 / 0.8573				
ESPCN [9]	×4	30.66 / 0.8646				
FSRCNN [8]	×4	30.73 / 0.8601				
VDSR [10]	×4	31.36 / 0.8796				
DRCN [11]	×4	31.56 / 0.8810				
LapSRN [13]	×4	31.54 / 0.8811				
DRRN [12]	×4	31.68 / 0.8888	28.21 / 0.7722	27.38 / 0.7240	25.44 / 0.7640	29.46 / 0.8960
SeaNet (Ours)	×4	<u>32.33 / 0.8970</u>	<u>28.72 / 0.7855</u>	<u>27.65 / 0.7388</u>	<u>26.32 / 0.7942</u>	<u>30.74 / 0.9129</u>
SeaNet+ (Ours)	×4	**32.44 / 0.8981**	**28.81 / 0.7872**	**27.70 / 0.7399**	**26.50 / 0.7976**	**31.05 / 0.9154**

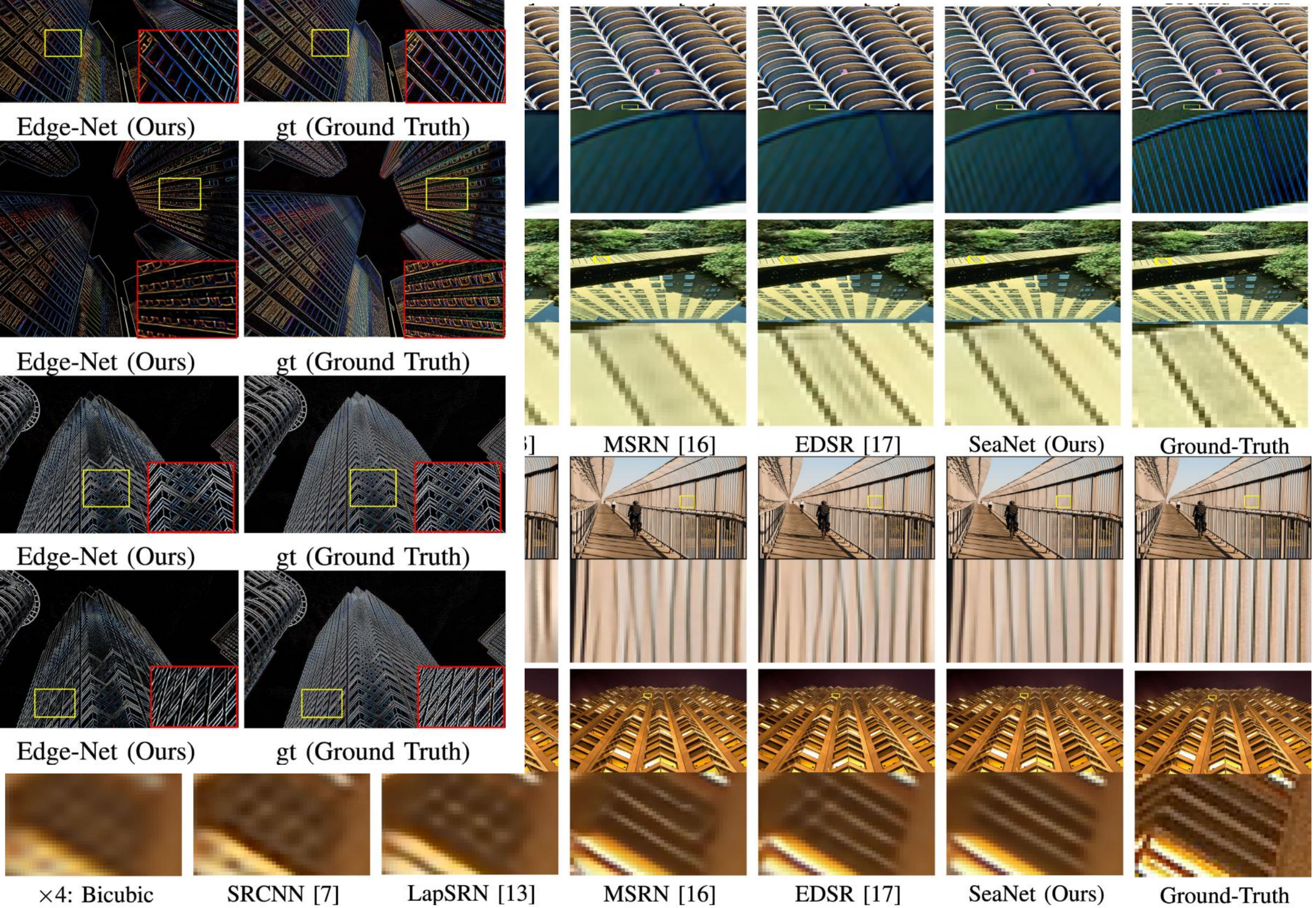
TABLE III

QUANTITATIVE COMPARISONS OF MSRN, EDSR, EDSR+, SEANET (BASELINE), SEANET (FINAL), AND SEANET+ (FINAL)

Algorithm	Scale	Parameters	Set5 [37] PSNR/SSIM	Set14 [38] PSNR/SSIM	BSDS100 [39] PSNR/SSIM	Urban100 [40] PSNR/SSIM	Manga109 [41] PSNR/SSIM
MSRN [16]	×2	5926k	38.08/0.9605	33.74/0.9170	32.23/0.9013	32.22/0.9326	38.82/0.9868
SeaNet (Baseline, Ours)	×2	4194k	37.99/0.9607	33.60/0.9174	32.18/0.8995	32.08/0.9276	38.48/0.9768
EDSR [17]	×2	**40121k**	38.11/0.9602	33.92/0.9195	32.32/0.9013	32.93/0.9351	39.10/0.9773
EDSR+ [17]	×2	**40121k**	38.20/0.9606	34.02/0.9204	32.37/0.9018	33.10/0.9363	– / –
SeaNet (Final, Ours)	×2	7102k	38.08/0.9609	33.75/0.9190	32.27/0.9008	32.50/0.9318	38.76/0.9774
SeaNet+ (Final, Ours)	×2	7102k	38.15/0.9611	33.86/0.9198	32.31/0.9013	32.68/0.9332	38.97/0.9779
MSRN [16]	×3	6110k	34.38/0.9262	30.34/0.8395	29.08/0.8041	28.08/0.8554	33.44/0.9427
SeaNet (Baseline, Ours)	×3	4563k	34.36/0.9280	30.34/0.8428	29.09/0.8053	28.17/0.8527	33.40/0.9444
EDSR [17]	×3	**42481k**	34.65/0.9280	30.52/0.8462	29.25/0.8093	28.80/0.8653	34.17/0.9476
EDSR+ [17]	×3	**42481k**	34.76/0.9290	30.66/0.8481	29.32/0.8104	29.02/0.8685	– / –
SeaNet (Final, Ours)	×3	7471k	34.55/0.9282	30.42/0.8444	29.17/0.8071	28.50/0.8594	33.73/0.9463
SeaNet+ (Final, Ours)	×3	7471k	34.65/0.9290	30.53/0.8461	29.23/0.8081	28.68/0.8620	34.02/0.9478
MSRN [16]	×4	6073k	32.07/0.8903	28.60/0.7751	27.52/0.7273	26.04/0.7896	30.17/0.9034
SeaNet (Baseline, Ours)	×4	4224k	32.18/0.8948	28.61/0.7822	27.57/0.7359	26.05/0.7896	30.44/0.9088
EDSR [17]	×4	**45430k**	32.46/0.8968	28.80/0.7876	27.71/0.7420	26.64/0.8033	31.02/0.9148
EDSR+ [17]	×4	**45430k**	32.62/0.8984	28.94/0.7901	27.79/0.7437	26.67/0.8041	– / –
SeaNet (Final, Ours)	×4	7397k	32.33/0.8970	28.72/0.7855	27.65/0.7388	26.32/0.7942	30.74/0.9129
SeaNet+ (Final, Ours)	×4	7397k	32.44/0.8981	28.81/0.7872	27.70/0.7399	26.50/0.7976	31.05/0.9154

SeaNet & MLEFGN





QUANTITATIVE COMPARISONS OF SRN AND SEANET (BASELINE)

Dataset	Scale	SRN	SeaNet (Baseline)
Set5 [37] PSNR/SSIM	×2	37.78/0.9597	**37.99/0.9607**
	×3	34.11/0.9249	**34.36/0.9280**
	×4	32.01/0.8919	**32.18/0.8948**
Set14 [38] PSNR/SSIM	×2	33.42/0.9158	**33.60/0.9174**
	×3	30.12/0.8378	**30.34/0.8428**
	×4	28.42/0.7771	**28.61/0.7822**
BSDS100 [39] PSNR/SSIM	×2	32.04/0.8974	**32.18/0.8995**
	×3	28.95/0.8006	**29.09/0.8053**
	×4	27.43/0.7304	**27.57/0.7359**
Urban100 [40] PSNR/SSIM	×2	31.56/0.9223	**32.08/0.9276**
	×3	27.74/0.8415	**28.17/0.8527**
	×4	25.74/0.7718	**26.05/0.7896**
Manga109 [41] PSNR/SSIM	×2	37.98/0.9756	**38.48/0.9768**
	×3	32.98/0.9405	**33.40/0.9444**
	×4	30.00/0.9022	**30.44/0.9088**

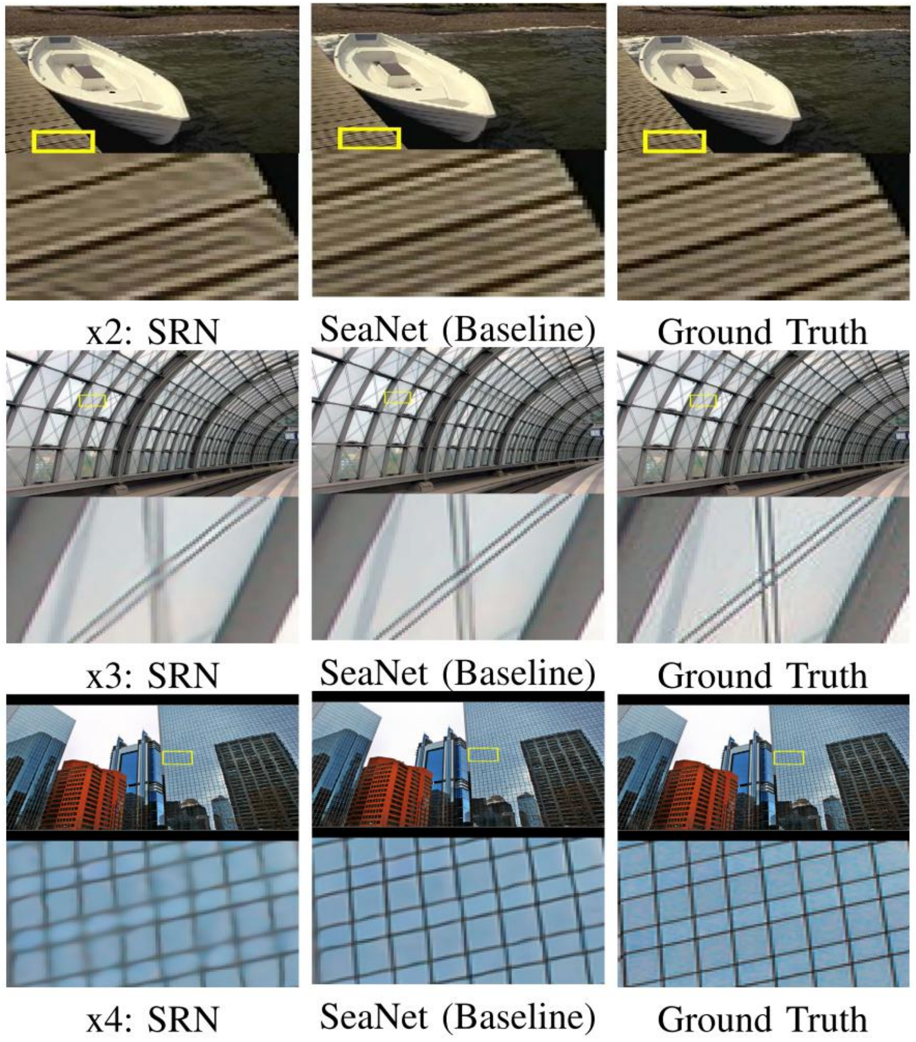


Fig. 6. Visual comparison of SRN and SeaNet (baseline) for x2, x3, and x4 SR images. SRN is a simplified model obtained by removing the Edge-Net from the SeaNet (baseline).

QUANTITATIVE COMPARISONS OF ESPCN [9] AND 'ESPCN+ISE'

Dataset	Scale	ESPCN	ESPCN+ISE
Set5 [37] PSNR/SSIM	×2	37.00/0.9559	**37.50/0.9580**
	×3	33.02/0.9135	**33.63/0.9190**
	×4	30.66/0.8646	**31.32/0.8780**
Set14 [38] PSNR/SSIM	×2	32.75/0.9098	**33.01/0.9100**
	×3	29.49/0.8271	**29.78/0.8291**
	×4	27.71/0.7562	**28.07/0.7600**
BSDS100 [39] PSNR/SSIM	×2	31.51/0.8939	**31.88/0.8956**
	×3	28.50/0.7937	**28.77/0.7960**
	×4	26.98/0.7124	**27.20/0.7156**
Urban100 [40] PSNR/SSIM	×2	29.87/0.9065	**30.66/0.9145**
	×3	26.41/0.8161	**27.09/0.8299**
	×4	24.60/0.7360	**25.08/0.7430**
Manga109 [41] PSNR/SSIM	×2	36.21/0.9694	**37.18/0.9711**
	×3	30.79/0.9181	**31.89/0.9288**
	×4	27.70/0.8560	**28.78/0.8756**

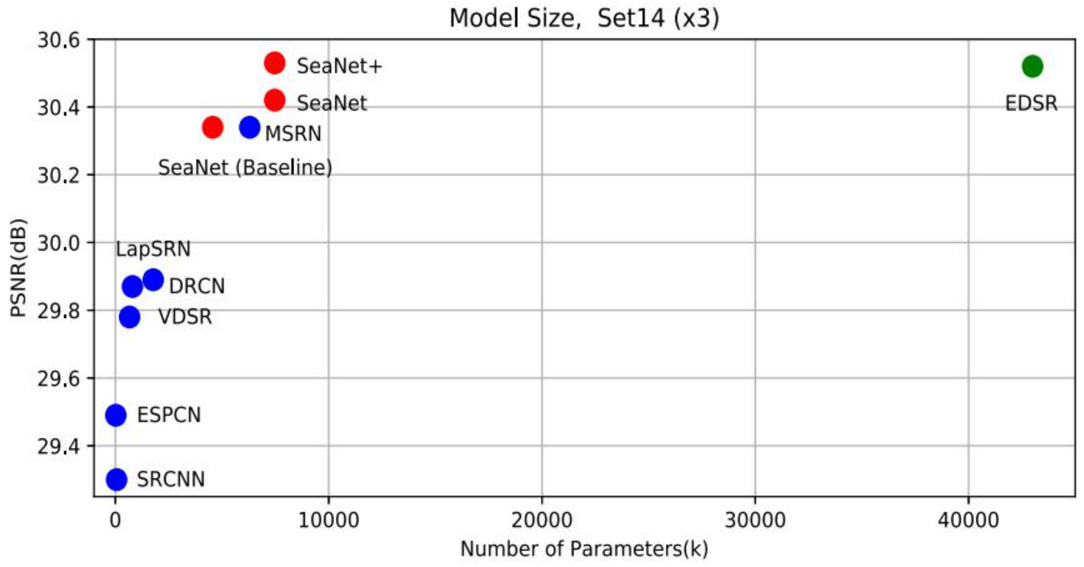


Fig. 8. Study of model size on the test dataset Set14 (x3). SeaNet strikes a good balance between model size and performance.

Multilevel Edge Features Guided Network for Image Denoising

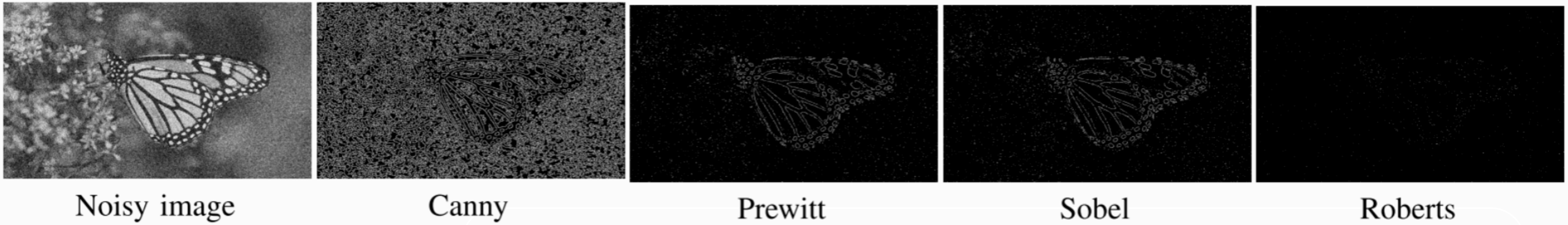
Faming Fang, Juncheng Li, Yiting Yuan, Tieyong Zeng, and Guixu Zhang

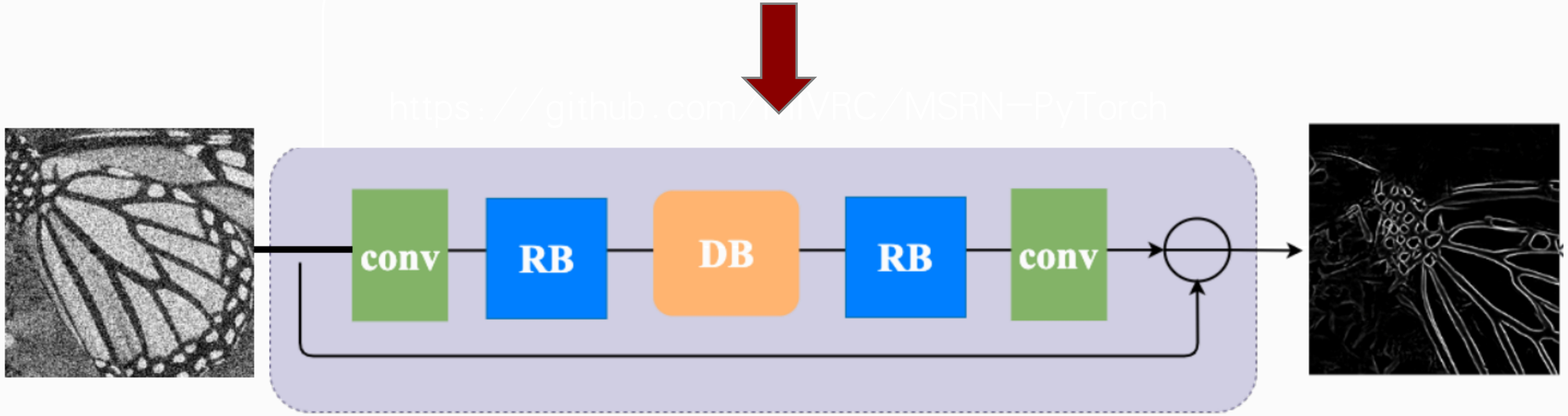
Noisy image MLEFGN (Ours) Original

Contribution

- We verified the importance and effectiveness of edge prior for SID.
- We propose a new edge guidance framework for image denoising, which integrates edge detection, edge guidance, and image denoising in an end-to-end model.

- We propose a new Edge-Net for SID. Edge-Net is the first CNN model that can directly reconstruct clear edges from the noisy observation.
- We propose a Multi-level Edge Features Guided Network (MLEFGN). MLEFGN is a well-designed model that can make full use of edges predicted by the Edge-Net to reconstruct high-quality noise-free images.

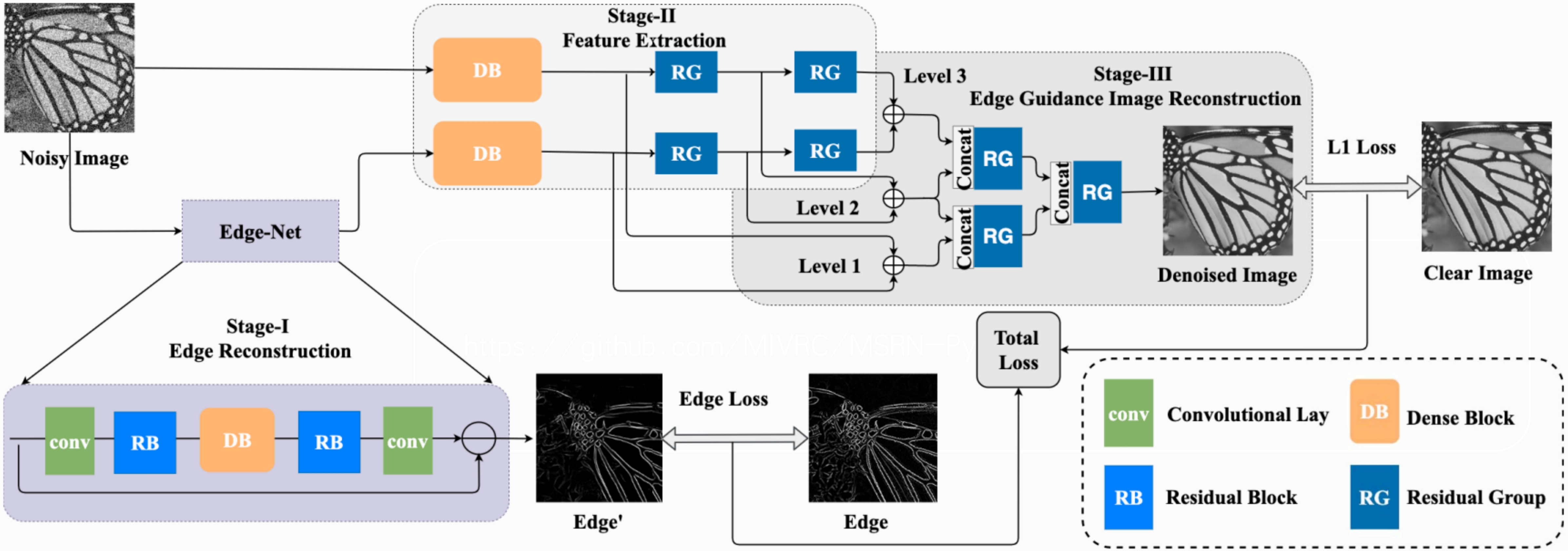




$$I'_{\text{edge}} = E(I_{\text{noisy}}) = I_{\text{noisy}} - R(I_{\text{noisy}})$$

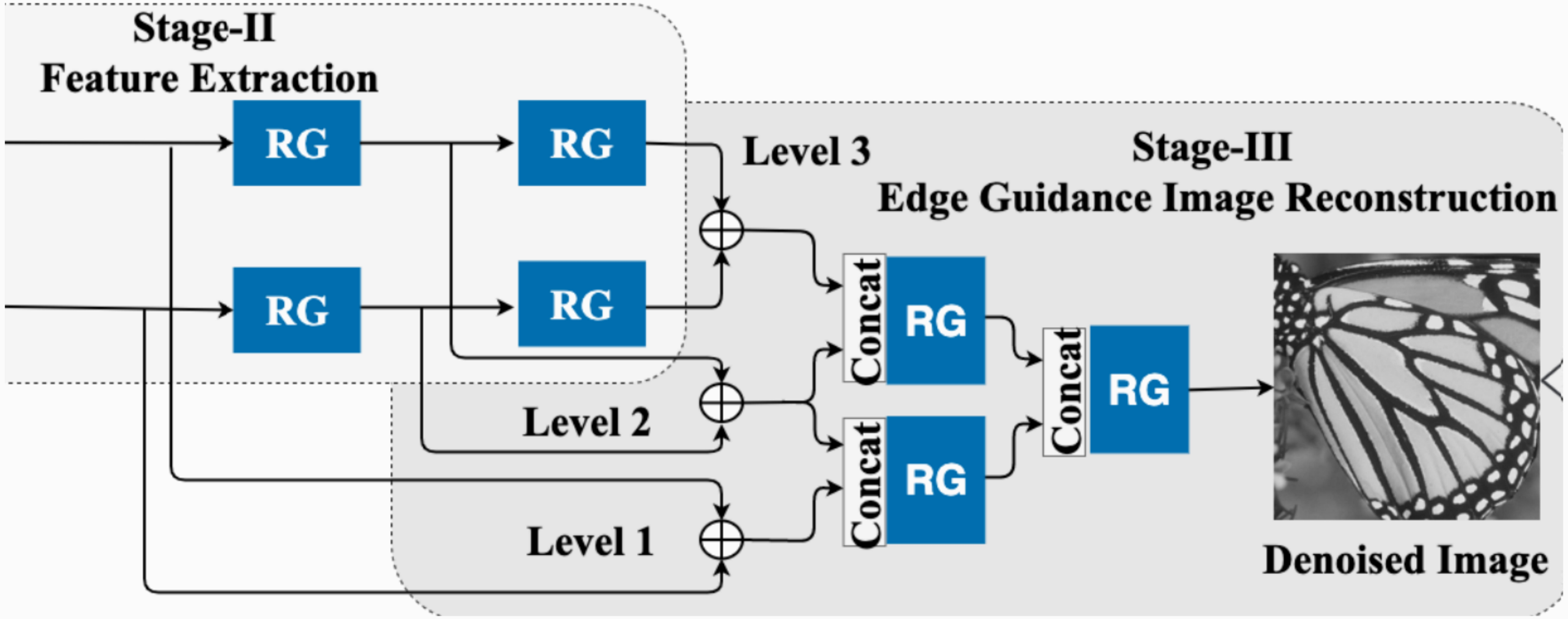
$$\mathcal{L}_{\text{edge}} = \left\| I'_{\text{edge}} - I_{\text{edge}} \right\|_1$$

SeaNet & MLEFGN



$$\hat{\theta} = \arg\min_{\theta} \frac{1}{M} \sum_{i=1}^{M} \| F(I^i_{\text{noisy}}, E(I^i_{\text{noisy}})) - I^i_{\text{clear}} \|_1 + \lambda \| E(I^i_{\text{noisy}}) - I^i_{\text{edge}} \|_1$$

Multi-level Guidance Mechanism:



$$f_{\text{fuse}}^{i} = f_{\text{image}}^{i} + f_{\text{edge}}^{i}$$

$$I_{\text{IPM}}^{1} = F_{\text{RG}}^{1}\left(\left[f_{\text{fuse}}^{1}, f_{\text{fuse}}^{2}\right]\right)$$
$$I_{\text{IPM}}^{2} = F_{\text{RG}}^{2}\left(\left[f_{\text{fuse}}^{2}, f_{\text{fuse}}^{3}\right]\right)$$
$$I_{\text{IPM}}^{3} = F_{\text{RG}}^{3}\left(\left[I_{\text{IPM}}^{1}, I_{\text{IPM}}^{2}\right]\right)$$

PSNR (dB) RESULTS OF DIFFERENT IMAGE DENOISING METHODS ON **GRAY-SCALE** IMAGES (**SET12**) WITH NOISE LEVELS σ = 15, 25, 35, AND 50. "AVERAGE" REPRESENTS THE AVERAGE RESULT OF THE DATA SET, AND THE BEST RESULTS ARE HIGHLIGHTED IN **RED**

Images	C.ma	House	Peppers	Starfish	Monarch	Airplane	Parror	Lena	Barbara	Boat	Man	Couple	**Average**
Noise Level	σ=15												
BM3D [15]	31.91	34.93	32.69	31.14	31.85	31.07	31.37	34.26	33.10	32.13	31.92	32.10	32.37
TNRD [43]	32.19	34.53	33.04	31.75	32.56	31.46	31.63	34.24	32.13	32.14	32.23	32.11	32.50
NLED$^{6}_{7\times7}$ [44]	32.28	34.76	33.10	31.75	32.71	31.59	31.70	34.35	32.53	32.16	32.22	32.13	32.61
WNNM [16]	32.17	35.13	32.99	31.82	32.71	31.39	31.62	34.27	33.60	32.27	32.11	32.17	32.70
IRCNN [45]	32.55	34.89	33.31	32.02	32.82	31.70	31.84	34.53	32.43	32.34	32.40	32.40	32.77
DnCNN [26]	32.61	34.97	33.30	32.20	33.09	31.70	31.83	34.62	32.64	32.42	32.46	32.47	32.86
FFDNet [28]	32.42	35.01	33.10	32.02	32.77	31.58	31.77	34.63	32.50	32.35	32.40	32.45	32.75
ADNet [46]	**32.81**	35.22	**33.49**	32.17	33.17	**31.86**	**31.96**	34.71	32.80	32.57	32.47	32.58	32.98
MLEFGN (Ours)	32.56	**35.41**	33.42	**32.29**	**33.44**	31.82	31.90	**34.80**	**33.05**	**32.60**	**32.51**	**32.67**	**33.04**
Noise Level	σ=25												
BM3D [15]	29.45	32.85	30.16	28.56	29.25	28.42	28.93	32.07	30.71	29.90	29.61	29.71	29.97
TNRD [43]	29.72	32.53	30.57	29.02	29.85	28.88	29.18	32.00	29.41	29.91	29.87	29.71	30.06
NLED$^{6}_{7\times7}$ [44]	29.75	32.81	30.66	29.09	30.03	28.99	29.29	32.18	30.11	29.90	29.86	29.74	30.18
WNNM [16]	29.64	32.22	30.42	29.03	29.84	28.69	29.15	32.24	**31.24**	30.03	29.76	29.82	30.26
IRCNN [45]	30.08	33.06	30.88	29.27	30.09	29.12	29.47	32.43	29.92	30.17	30.04	30.08	30.38
DnCNN [26]	30.18	33.06	30.87	29.41	30.28	29.13	29.43	32.44	30.00	30.21	30.10	30.12	30.43
FFDNet [28]	30.06	33.27	30.79	29.33	30.14	29.05	29.43	32.59	29.98	30.23	30.10	30.18	30.43
ADNet [46]	**30.34**	33.41	**31.14**	29.41	30.39	29.17	**29.49**	32.61	30.25	30.37	30.08	30.24	30.58
MLEFGN (Ours)	30.29	**33.61**	30.98	**29.66**	**30.52**	**29.25**	29.46	**32.76**	30.57	**30.40**	**30.13**	**30.30**	**30.66**
Noise Level	σ=35												
BM3D [15]	27.92	31.36	28.51	26.86	27.58	26.83	27.40	30.56	28.98	28.43	28.22	28.15	28.40
MLP [20]	28.08	31.18	28.54	27.12	27.97	27.22	27.72	30.82	27.62	28.53	28.47	28.24	28.46
WNNM [16]	**28.80**	31.92	28.75	27.27	28.13	27.10	27.69	30.73	**29.48**	28.54	28.33	28.24	28.69
DnCNN [26]	28.61	31.61	29.14	27.53	28.51	27.52	27.94	30.91	28.09	28.72	28.66	28.52	28.82
FFDNet [28]	28.54	31.99	29.18	27.58	28.54	27.47	28.02	31.20	28.29	28.82	28.70	28.68	28.92
MLEFGN (Ours)	28.78	**32.47**	**29.37**	**27.77**	**28.70**	**27.62**	**28.03**	**31.36**	29.09	**29.00**	**28.79**	**28.88**	**29.15**
Noise Level	σ=50												
BM3D [15]	26.13	29.69	26.68	25.04	25.82	25.10	25.90	29.05	27.22	26.78	26.81	26.46	26.72
MLP [20]	26.37	29.64	26.68	25.43	26.26	25.56	26.12	29.32	25.24	27.03	27.06	26.67	26.78
TNRD [43]	26.62	29.48	27.10	25.42	26.31	25.59	26.16	28.93	25.70	26.94	26.98	26.50	26.81
WNNM [16]	26.45	30.33	26.95	25.44	26.32	25.42	26.14	29.25	27.79	26.97	26.94	26.64	27.05
IRCNN [45]	26.88	29.96	27.33	25.57	26.61	25.89	26.55	29.40	26.24	27.17	27.17	26.88	27.14
DnCNN [26]	27.03	30.00	27.32	25.70	26.78	25.87	26.48	29.39	26.22	27.20	27.24	26.90	27.18
FFDNet [28]	27.03	30.43	27.43	**25.77**	26.88	25.90	**26.58**	29.68	26.48	27.32	27.30	27.07	27.32
ADNet [46]	27.31	30.59	**27.69**	25.70	26.90	25.88	26.56	29.59	26.64	27.35	27.17	27.07	27.37
MLEFGN (Ours)	27.15	**31.00**	27.63	**25.77**	**27.01**	**26.05**	26.56	**29.85**	**27.37**	**27.40**	**27.32**	**27.35**	**27.54**

AVERAGE PSNR (dB) RESULTS OF DIFFERENT IMAGE DENOISING METHODS ON **GRAY-SCALE** IMAGES (**BSD68**) WITH NOISE LEVELS $\sigma = 15, 25, 35,$ AND 50

Method	σ=15	σ=25	σ=35	σ=50
BM3D [15]	31.08	28.57	27.08	25.62
EPLL [54]	31.21	28.68	27.16	25.67
WNNM [16]	31.37	28.83	27.30	25.87
TNRD [43]	31.42	28.92	N/A	25.97
$\text{NLED}^{6}_{7\times7}$ [44]	31.43	28.93	N/A	N/A
MLP [20]	31.50	28.96	27.50	26.03
DnCNN [26]	31.73	29.23	27.69	26.23
FFDNet [28]	31.63	29.19	27.73	26.29
ADNet [46]	31.74	29.25	N/A	26.29
N^3Net [55]	N/A	29.30	N/A	**26.39**
MLEFGN (Ours)	**31.81**	**29.34**	**27.85**	**26.39**

AVERAGE PSNR (dB) RESULTS OF DIFFERENT IMAGE DENOISING METHODS ON **GRAY-SCALE** IMAGES (**URBAN100**) WITH NOISE LEVELS $\sigma = 15, 25,$ AND 50

Method	σ=15	σ=25	σ=50
TNRD [43]	31.98	29.29	25.71
BM3D [15]	32.34	29.70	25.94
IRCNN [45]	32.49	29.82	26.14
DnCNN [26]	32.68	29.97	26.28
NN3D [58]	N/A	30.09	26.47
FFDNet [28]	32.42	29.92	26.52
N^3Net [55]	N/A	30.19	26.82
WNNM [16]	32.97	30.39	26.83
MLEFGN (Ours)	**33.21**	**30.64**	**27.22**

AVERAGE PSNR (dB) RESULTS OF DIFFERENT IMAGE DENOISING METHODS ON **COLOR** IMAGES (**KODAK24 [49], CBSD68 [50], AND URBAN100 [51]**) WITH NOISE LEVELS $\sigma = 10, 30, 50,$ AND 70. BEST RESULTS ARE HIGHLIGHTED IN **RED** COLOR

Method	Kodak24 [49]				CBSD68 [50]				Urban100 [51]			
Noise Level	σ=10	σ=30	σ=50	σ=70	σ=10	σ=30	σ=50	σ=70	σ=10	σ=30	σ=50	σ=70
TNRD [43]	34.33	28.83	27.17	24.94	33.36	27.64	25.96	23.83	33.60	27.40	25.52	22.63
RED [56]	34.91	29.71	27.62	26.36	33.89	28.46	26.35	25.08	34.59	29.02	26.40	24.74
MemNet [57]	N/A	29.67	27.65	26.40	N/A	28.39	26.33	25.08	N/A	28.93	26.53	24.93
CBM3D [15]	36.57	30.89	28.63	27.27	35.91	29.73	27.38	26.00	36.00	30.36	27.94	26.31
IRCNN [45]	36.70	31.24	28.93	N/A	36.06	30.22	27.86	N/A	26.53	30.28	27.69	N/A
DnCNN [26]	36.98	31.39	29.16	27.64	36.31	30.40	28.01	26.56	36.21	30.28	28.16	26.17
FFDNet [28]	36.81	31.39	29.10	27.68	36.14	30.31	27.96	26.53	35.77	30.53	28.05	26.39
MLEFGN (Ours)	**37.04**	**31.67**	**29.38**	**27.94**	**36.37**	**30.56**	**28.21**	**26.75**	**36.42**	**31.32**	**28.92**	**27.28**

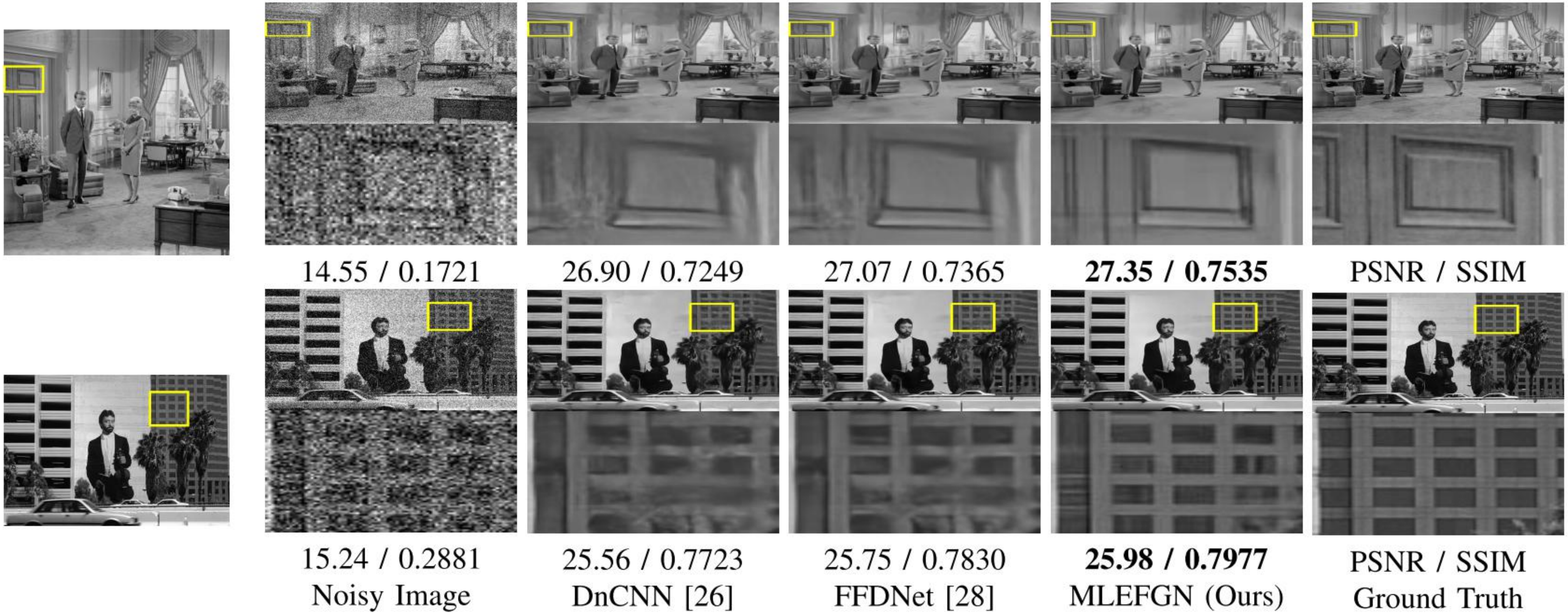


Fig. 8. Visual comparison of MLEFGN with DnCNN [26] and FFDNet [28] on gray-scale images (noise level: $\sigma = 50$).

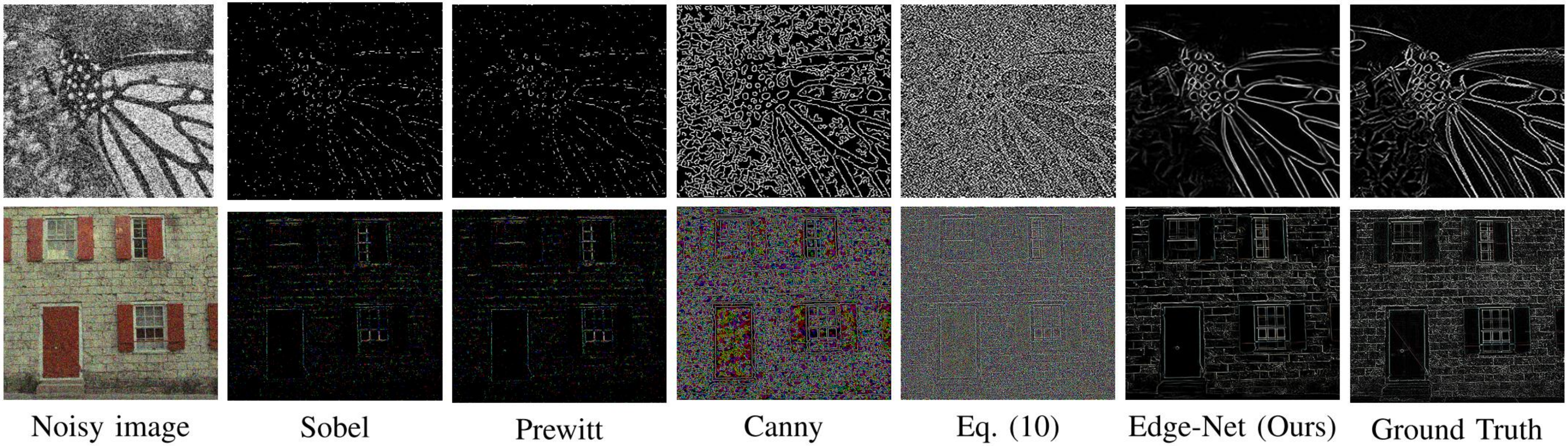


Fig. 12. Visual comparison of image edges extracted by different methods. Noise levels of gray-scale and color images are set to $\sigma = 75$ and 50, respectively.

05

Summary

Summary

- We verified the importance and effectiveness of multi-scale image featurs for image restoration.
- We verified the importance and effectiveness of image edge priors for image restoration.
- According to the above strategies, we have designed a series of lightweight image restoration models that can achieve better performance with fewer parameters.

06

Discussion

Discussion

Lightweight model exploration. → Knowledge distillation

Lack of real training datasets. → Few-shot or zero-shot learning
Weakly supervised or unsupervised learning

Generalization ability needs to be further improved. → Ensemble Learning
Uncertainty measurement

A model is difficult to suitable for multiple different degradation modes. → ?

THANKS