AIM 2020 Challenge on Real Image Super-Resolution: Methods and Results

Pengxu Wei[1](✉), Hannan Lu[2], Radu Timofte[3], Liang Lin[1], Wangmeng Zuo[2], Zhihong Pan[4], Baopu Li[4], Teng Xi[5], Yanwen Fan[5], Gang Zhang[5], Jingtuo Liu[5], Junyu Han[5], Errui Ding[5], Tangxin Xie[6], Liang Cao[6], Yan Zou[6], Yi Shen[6], Jialiang Zhang[6], Yu Jia[6], Kaihua Cheng[7], Chenhuan Wu[7], Yue Lin[8], Cen Liu[8], Yunbo Peng[9], Xueyi Zou[10], Zhipeng Luo[11], Yuehan Yao[11], Zhenyu Xu[11], Syed Waqas Zamir[12], Aditya Arora[12], Salman Khan[12], Munawar Hayat[12], Fahad Shahbaz Khan[12], Keon-Hee Ahn[13], Jun-Hyuk Kim[13], Jun-Ho Choi[13], Jong-Seok Lee[13], Tongtong Zhao[14], Shanshan Zhao[14], Yoseob Han[15], Byung-Hoon Kim[16], JaeHyun Baek[17], Haoning Wu[18], Dejia Xu[19], Bo Zhou[19], Wei Guan[20], Xiaobo Li[20], Chen Ye[20], Hao Li[21], Haoyu Zhong[21], Yukai Shi[21], Zhijing Yang[21], Xiaojun Yang[21], Haoyu Zhong[21], Xin Li[22], Xin Jin[22], Yaojun Wu[22], Yingxue Pang[22], Sen Liu[22], Zhi-Song Liu[23], Li-Wen Wang[24], Chu-Tak Li[24], Marie-Paule Cani[24], Wan-Chi Siu[24], Yuanbo Zhou[25], Rao Muhammad Umer[26], Christian Micheloni[26], Xiaofeng Cong[26], Rajat Gupta[27], Keon-Hee Ahn[28], Jun-Hyuk Kim[28], Jun-Ho Choi[28], Jong-Seok Lee[28], Feras Almasri[29], Thomas Vandamme[29], and Olivier Debeir[29]

[1] Sun Yat-sen University, Guangzhou, China
weipx3@mail.sysu.edu.cn
[2] Harbin Institute of Technology University, Harbin, China
[3] Computer Vision Lab, ETH Zurich, Zurich, Switzerland
[4] Baidu Research, Silicon Valley, USA
[5] Department of Computer Vision Technology (VIS), Baidu Incorportation, Silicon Valley, USA
[6] China Electronic Technology Cyber Security Co., Ltd., Beijing, China
[7] Guangdong OPPO Mobile Telecommunications Corp., Ltd., Dongguan, China
[8] NetEase Games AI Lab, Beijing, China
[9] Noah's Ark Lab Huawei, Beijing, China
[10] DeepBlue Technology (Shanghai) Co., Ltd., Shanghai, China
[11] Inception Institute of Artificial Intelligence (IIAI), Beijing, China
[12] Yonsei University, Seodaemun-gu, South Korea
[13] Dalian Maritime Univerity, Dalian, China
[14] Loa Alamos National Laboratory (LANL), New Mexico, USA
[15] Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea
[16] Amazon Web Services (AWS), Seattle, USA
[17] Peking University, Beijing, China

(P. Wei, H. Lu, R. Timofte, L. Lin, W. Zuo—Challenge organizers and the other others participated in the challenge. Appendix A contains the authors's teams and affiliations. AIM webpage: https://data.vision.ee.ethz.ch/cvl/aim20/)

A. Bartoli and A. Fusiello (Eds.): ECCV 2020 Workshops, LNCS 12537, pp. 392–422, 2020.
https://doi.org/10.1007/978-3-030-67070-2_24

[18] Jiangnan University, Jiangnan, China
[19] Karlsruher Institut fuer Technologie, Karlsruher, Germany
[20] Tongji University, Tongji, China
[21] Guangdong University of Technology, Tongji, China
[22] University of Science and Technology of China, Hefei, China
[23] LIX - Computer science laboratory at the Ecole polytechnique [Palaiseau], Palaiseau, France
[24] Center of Multimedia Signal Processing, The Hong Kong Polytechnic University, Hong Kong, China
[25] Fuzhou University, Fuzhou, Fujian, China
[26] University of Udine, Udine, Italy
[27] Indian Institute of Technology, Khargapur, India
[28] National University of Defense Technology, Changsha, China
[29] LISA Department, Universie Libre de Bruxelles, Brussels, Belgium

Abstract. This paper introduces the real image Super-Resolution (SR) challenge that was part of the Advances in Image Manipulation (AIM) workshop, held in conjunction with ECCV 2020. This challenge involves three tracks to super-resolve an input image for ×2, ×3 and ×4 scaling factors, respectively. The goal is to attract more attention to realistic image degradation for the SR task, which is much more complicated and challenging, and contributes to real-world image super-resolution applications. 452 participants were registered for three tracks in total, and 24 teams submitted their results. They gauge the state-of-the-art approaches for real image SR in terms of PSNR and SSIM.

1 Introduction

Single image super-resolution (SR) reconstructs high-resolution (HR) images from low-resolution (LR) counterparts with image quality degradations [12,44]. Instead of imposing higher requirements on hardware devices and sensors, it could be applicable to many practical scenarios, such as video surveillance, satellite, medical imaging, *etc.* As a fundamental res earch topic, SR has attracted a long-standing and considerable attention in computer vision community.

With the emergence of deep learning, convolutional neural network (CNN) based SR methods (*e.g.*, SRCNN [8], SRGAN [18], EDSR [20], ESRGAN [38] and RCAN [51]) inherit the powerful capacity of deep learning and have achieved remarkable performance improvements. Nevertheless, so far, the remarkable progress of SR is mainly driven by the supervised learning of models from LR images and their HR counterparts. While the bicubic downsampling is usually adopted to simulate the LR images, the learned deep SR model performs much less effective for real-world SR applications since the image degradation in real-world is much more complicated.

To mitigate this issue, several real SR datasets have been recently built, City 100 [5] and SR-RAW [50]. The images in City100 were captured for the printed postcards in the indoor environment, which are limited in capturing the complicated image and degradation characteristics of natural scenes. The images in SR-RAW were collected in the real world and a contextual bilateral loss was proposed to address the misalignment problem in the dataset. Besides, Cai *et al.* [4] released another real image SR dataset, named RealSR, which was captured from two DSLR cameras. They proposed the LP-KPN method in a Laplacian pyramid framework. Considering the complex image degradation across different scenes and devices, a large-scale diverse real SR dataset, named DRealSR [40], was released to further promote the research on real-world image SR. Images of DRealSR were captured by five different DSLR cameras and posed more challenging image degradation. In [40], the proposed component divide-and-conquer model (CDC) built a baseline, hourglass SR network (HGSR), in a stacked architecture, explored different reconstruction difficulties in terms of three low-level image components inspired by corner point detection, *i.e*, the flat, edges and corner points, and trained the model with a mediate supervision strategy. Besides, its proposed gradient-weighted (GW) loss also drives the model to adapt learning objectives to the reconstruction difficulties of three image components and has a flexibility of the application to any SR model.

Jointly with the Advances in Image Manipulation (AIM) 2020 workshop, we organize the AIM Challenge on Real-world Image Super-Resolution. Specifically, this challenge concerns the real-world SISR, which poses two challenging issues [40]: (1) more complex degradation against bicubic downsampling, and (2) diverse degradation processes among devices, aiming to learn a generic model to super-resolve LR images captured in practical scenarios. To achieve this goal, paired LR and HR images are captured by various DSLR cameras and provided for training. They are randomly selected from the DRealSR dataset. Images for training, validation and testing are captured in the same way with the same set of cameras. The setting is similar to that from the NTIRE 2019 challenge on real image super-resolution [3] employing RealSR dataset [4], and is different from the AIM 2019 [25] and NTIRE 2020 [24] challenges on real-world super-resolution where no LR-HR pairs are available for training, therefore an unsupervised setting defined in [23].

This challenge is one of the AIM 2020 associated challenges on: scene relighting and illumination estimation [10], image extreme inpainting [27], learned image signal processing pipeline [15], rendering realistic bokeh [16], real image super-resolution [39], efficient super-resolution [49], video temporal super-resolution [32] and video extreme super-resolution [11].

Table 1. Details of the dataset for the challenge

Scale	Split	Type	Number	Size (LR)	Evaluation
×2	Train	Cropped patches	19,000	380 × 380	PSNR (on RGB channels), SSIM
	Validation	Aligned images	20	∼2000 × 3000	
	Test	Aligned images	60		
×3	Train	Cropped patches	19,000	272 × 272	
	Validation	Aligned images	20	∼1300 × 2000	
	Test	Aligned images	60		
×4	Train	Cropped patches	19,000	192 × 192	
	Validation	Aligned images	20	∼1000 × 1250	
	Test	Aligned images	60		

2 AIM 2020 Challenge on Real Image Super-Resolution

The objectives of the AIM 2020 challenge on real image super-resolution challenge are: (i) to further explore the researches on real image SR; (ii) to fully evaluate different SR approaches on different scale factors; (iii) to offer an opportunity of communications between academic and industrial participants.

2.1 DRealSR Dataset

DRealSR[1] [40] is a large-scale real-world image super-resolution. Only half of images in DRealSR are randomly selected for this challenge. These images are captured from five DSLR cameras (*i.e.,* Canon, Sony, Nikon, Olympus and Panasonic) in natural scenes and cover indoor and outdoor scenes avoiding moving objects, *e.g.*, advertising posters, plants, offices, buildings, *etc.* These HR-LR image pairs are aligned. To get access to the training and validation data and submit SR results, the registration on Codalab[2] is required. Details of the dataset in this challenge are given in Table 1.

2.2 Track and Competition

Tracks. The challenge uses the newly released DRealSR dataset and has three tracks corresponding to ×2, ×3, ×4 upscaling factors. The aim is to obtain a network design or solution capable to produce high-quality results with the best fidelity to the reference ground truth.

Challenge Phases. *(1) Development phase:* HR images from DRealSR have 4000 × 6000 pixels on average. For the convenience of model training, images are cropped into patches. For ×2 scale factor, LR image patches are 380 × 380; for ×3 scale factor, LR image patches are 272 × 272; for ×4 scale factor, LR image

[1] The dataset is publicly available at https://github.com/xiezw5/Component-Divide-and-Conquer-for-Real-World-Image-Super-Resolution.

[2] https://competitions.codalab.org.

patches are 192 × 192. *(2) Testing Phase:* In the final test phase, participants have access to LR images for three tracks, submit their SR results to Codalab evaluation server and email their codes and factsheets to the organizers. The organizers checked all the SR results and the provided codes to obtain the final results.

Evaluation Protocol. The evaluation includes the comparison of the super-resolved images with the reference ground truth images. We use the standard peak signal to noise ratio (PSNR) and, complementary, the structural similarity (SSIM) index as often employed in the literature. PSNR and SSIM implementations are found in most of the image processing toolboxes. For each dataset, we report the average results (i.e. $PSNR_{avg}$ and $SSIM_{avg}$) over all the processed images belonging to it and employ for ranking the weighted value of normalized $PSNR_{avg}$ and $SSIM_{avg}$, which is defined as follows,

$$PSNR_{avg}/50 + (SSIM_{avg} - 0.4)/0.6. \tag{1}$$

3 Challenge Results

There are 174, 128 and 168 registered participants for three tracks, respectively. In total, 24 teams submitted their super-resolution results; 10, 2 and 11 teams submitted results of one, two and three tracks, respectively. Among those submitted results of one track, seven teams are for ×4 scale factor. Details of final testing results are provided in Table 2. It mainly reports the final evaluation results and model training details.

As for the evaluation metric of weighted score claimed in Sect. 2.2, the leading entries for Track 1, 2 and 3 are all from team Baidu. For Track 1 and 2, the CETC-CSKT and the OPPO_CAMERA team win the second and the third places, respectively. For Track 3, ALONG and CETC-CSKT win the second and the third places, respectively. Among those solutions for the challenge, some interesting trends can be observed as follows.

Network Architecture. All the teams utilize deep neural networks for super-resolution. The architecture of the deep network will greatly affect the performance of super-resolution images. Several teams, *e.g.*, TeamInception, construct a network with the residual structure to reduce the difficulty of optimization, While OPPO_CAMERA connected the input to the output with a trainable convolution layer. CETC-CSKT further proposed to pre-train the trainable layer in the skip branch in advance. Several teams, such as DeepBlueAI and SR-IM applied channel attention module in their network, while several others like TeamInception and Noah_TerminalVision employ both spatial attention and channel attention on the feature level.

Data Augmentation. Most solutions conduct the data augmentation by randomly flipping and rotating images by 90°. The newly proposed CutBlur method was employed by ALONG and OPPO_CAMERA and performance improvements are reported by these teams.

Table 2. Evaluation results in the final testing phase. "Score" indicates the weighted score (Eq. 1), *i.e.*, the evaluation metric for the challenge. For "Ensemble", "model" and "self" indicate the model ensemble and the self-ensemble, respectively. "/" indicates that those items are not provided by participants. We also provide results of "EDSR*" for comparison with the same challenge dataset.

Team	PSNR	SSIM	Score	Ensemble	ExtraData	Loss
Track1 (×2)						
Baidu	**33.446**	**0.927**	**0.7736**	Model+Self	False	L_1 + SSIM
CETC-CSKT	33.314	0.925	0.7702	Model+Self	False	L_1
OPPO_CAMERA	33.309	0.924	0.7699	Model+Self	False	L_1 + SSIM + MS-SSIM
AiAiR	33.263	0.924	0.7695	Model+Self	True	Clip L_1
TeamInception	33.232	0.924	0.7690	Model+Self	True	L_1 + MS-SSIM + VGG
Noah_TerminalVision	33.289	0.923	0.7686	Self	False	adaptive robust loss
DeepBlueAI	33.177	0.924	0.7681	Self	False	/
ALONG	33.098	0.924	0.7674	Self	False	L_1 + L_2
LISA-ULB	32.987	0.923	0.7659	/	False	L_1 + SSIM
lyl	32.937	0.921	0.7635	/	False	L_1
GDUT-SL	32.973	0.920	0.7634	Model	False	L_1
MCML-Yonsei	32.903	0.919	0.7612	None	False	L_1
Kailos	32.708	0.920	0.7601	Self	False	L_1 + wavelet loss
qwq	31.640	0.913	0.7436	None	False	L_1 + SSIM
debut_kele	31.236	0.889	0.7196	None	True	/
EDSR*	31.220	0.889	0.7194	/	/	/
RRDN_IITKGP	29.851	0.845	0.6696	None	True	/
Track2 (×3)						
Baidu	**30.950**	**0.876**	**0.7063**	Model+Self	False	L_1 + SSIM
CETC-CSKT	30.765	0.871	0.7005	Model+Self	False	L_1
OPPO_CAMERA	30.537	0.870	0.6966	Model+Self	False	L_1 + SSIM + MS-SSIM
Noah_TerminalVision	30.564	0.866	0.6941	Self	False	adaptive robust loss
MCML-Yonsei	30.477	0.866	0.6931	Self	False	L_1
TeamInception	30.418	0.866	0.6928	Model+Self	True	L_1 + MS-SSIM + VGG
ALONG	30.375	0.866	0.6922	Self	False	L_1 + L_2
DeepBlueAI	30.302	0.867	0.6918	Self	False	/
lyl	30.365	0.864	0.6905	/	False	L_1
Kailos	30.130	0.866	0.6900	Self	False	L_1 + wavelet loss
qwq	29.266	0.852	0.6694	None	False	L_1 + SSIM
EDSR*	28.763	0.821	0.6383	/	/	/
anonymous	18.190	0.825	0.5357	/	False	/
Track3 (×4)						
Baidu	**31.396**	**0.875**	**0.7099**	Model+Self	False	L_1 + SSIM
ALONG	31.237	0.874	0.7075	Self	False	L_1 + L_2
CETC-CSKT	31.123	0.874	0.7066	Model+Self	False	L_1
SR-IM	31.174	0.873	0.7057	Self	False	/
DeepBlueAI	30.964	0.874	0.7044	Self	False	/
JNSR	30.999	0.872	0.7035	Model+Self	True	/
OPPO_CAMERA	30.86	0.874	0.7033	Model+Self	False	L_1 + SSIM + MS-SSIM
Kailos	30.866	0.873	0.7032	Self	False	L_1 + wavelet loss
SR_DLu	30.605	0.866	0.6944	Self	False	/
Noah_TerminalVision	30.587	0.866	0.6944	Self	False	adaptive robust loss
Webbzhou	30.417	0.867	0.6936	None	False	/
TeamInception	30.347	0.868	0.6935	Model+Self	True	L_1 + MS-SSIM + VGG
lyl	30.319	0.866	0.6911	/	False	L_1
MCML-Yonsei	30.420	0.864	0.6906	Self	False	L_1
MoonCloud	30.283	0.864	0.6898	Model + Self	True	/
qwq	29.588	0.855	0.6748	None	False	L_1 + SSIM
SrDance	29.595	0.852	0.6729	/	True	MAE+VGG+GAN loss
MLP_SR	28.619	0.831	0.6457	Self	True	GAN,TV,L_1,SSIM,MS-SSIM,Cycle
EDSR*	28.212	0.824	0.6356	/	/	/
RRDN_IITKGP	27.971	0.809	0.6201	None	True	/
congxiaofeng	26.392	0.826	0.6187	None	False	L_1

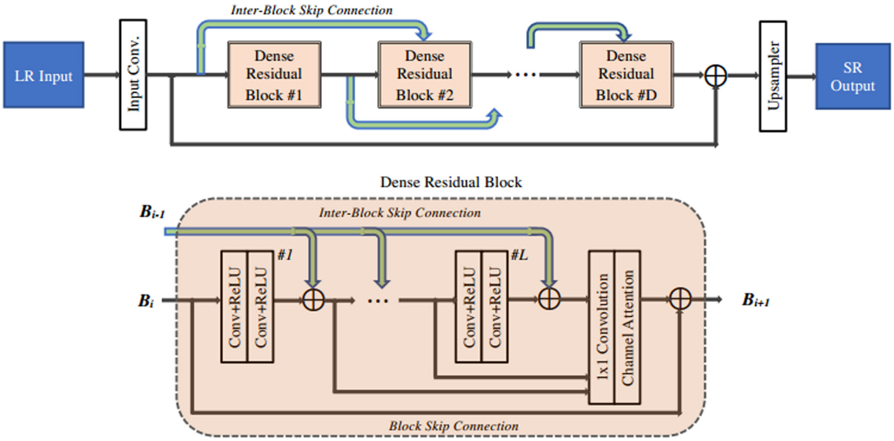


Fig. 1. The dense residual network architecture of the Baidu team for image Super-Resolution

Ensemble Strategy. Most solutions adopted self-ensemble ×8. Some solutions also performed model-ensemble by fusing results from models with different training parameter, or even of different architectures.

Platform. All the teams except one team using Tensorflow utilized PyTorch to conduct their experiments.

4 Challenge Methods and Teams

Baidu

The Baidu team proposed to apply Neural Architecture Search (NAS) approach selecting variations of their previous dense residual model as well as RCAN model [28]. In order to accelerate the searching process, Gaussian Process based Neural Architecture Search (GP-NAS) was applied as in [19]. Specifically, given the hyper-parameters of GP-NAS, they are capable of predicting the performance of any architectures in the search space effectively. Then, the NAS process is converted to hyper-parameters estimation. By mutual information maximization, the Baidu team can efficiently sample networks. Accordingly, based on the performances of sampled networks, the posterior distribution of hyper-parameters can be gradually and efficiently updated. Based on the estimated hyper-parameters, the architecture with the best performance can be obtained.

The backbone model of the proposed method is a deep dense residual network originally developed for raw image demosaicking and denoising. As depicted in Fig. 1, in addition to the shallow feature convolution at the front and the upsampler at the end, the proposed network consists of a total depth of D dense residual blocks (DRB). The input convolution layer converts the 3-channel LR input to a total of F-channel shallow features. For the middle DRB blocks, each one

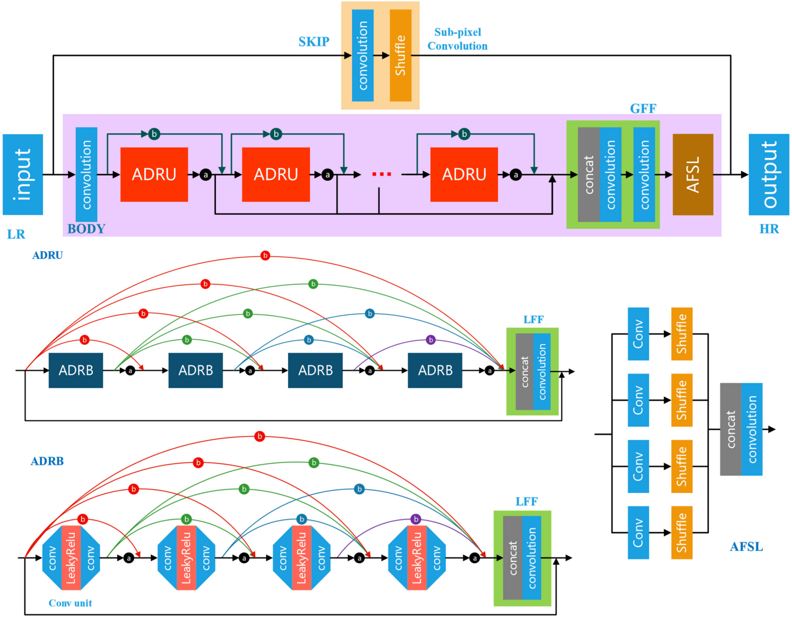


Fig. 2. Framework of Adaptive Dense Connection Super Resolution reconstruction (ADCSR) for the CETC-CSKT team

includes L stages of double layers of convolution and the outputs of all L stages are concatenated together before convoluted from $F \times L$ to F channels. An additional channel-attention layers are included at the end of each block, similar to RCAN [51]. There are two types of skip connections included in each block, the block skip connection (BSC) and inter-block skip connection (IBSC). The BSC is the shortcut between input and output of block B_i, while IBSC includes two shortcuts from the input of block B_{i-1} to the two stages inside block B_i, respectively. The various skip connections, especially IBSC, are included to combine features with a large range of receptive fields. The last block is an enhanced upsampler that transforms all F-channel LR features to the estimated 3-channel SR image. This dense residual network has three main hyper-parameters: F is the number of feature channels, D is the number of DRB layers and L is the number of stages for each DRB. All these three hyper-parameters construct the search space for NAS.

During training, a 120 × 120 patch is randomly cropped and augmented with flipping and transposing from each training image for each epoch. A mixed loss of L_1 and multi-scale structural similarity (MS-SSIM) is taken for training. For the experiment, the new model candidate search scheme using GP-NAS was implemented in PaddlePaddle [26] and the final-training of searched models were conducted using PyTorch. A multi-level ensemble scheme is proposed in testing, including self-ensemble for patches, as well as patch-ensemble and model-ensemble for full-size images. The proposed method is validated to be highly effective, generating impressive testing results on all three tracks of AIM2020 Real Image Super-resolution Challenge.

CETC-CSKT

The CETC-CSKT team proposed Adaptive Dense Connection Super Resolution reconstruction(ADCSR) [42,43]. The algorithm is divided into BODY and SKIP. The BODY part improves the utilization of convolution features through adaptive dense connection. An adaptive sub-pixel reconstruction module (AFSC) is also proposed to reconstruct the features of BODY output. By pre-training SKIP in advance, the BODY part focuses on high-frequency feature learning. for track 1 (×2), spatial attention is added after each residual block. The architecture is shown in Fig. 2. Self-ensemble is used in EDSR [20]. The test image is divided into 80×80 pixel blocks for reconstruction. Finally, only 60×60 input is used for splicing to reduce the edge difference of blocks.

The proposed ADCSR uses the first 18900 training data sets for training, and the last 100 as the test set for training. The input image block size is 80×80. SKIP is trained separately, and then the entire network is trained at the same time. The initial learning rate is 1×10^{-4}. When the learning rate drops to 5×10^{-7}, the training stops. L_1 loss is utilized to optimize the proposed model. The model is trained with NVIDIA RTX2080Ti * 4. Pytorch1.1.0 + Cuda10.0 + cudnn7.5.0 is selected as the deep learning environment.

OPPO_CAMERA

The OPPO_CAMERA team proposed Self-Calibrated Attention Neural Network for Real-World Super Resolution [6]. As shown in Fig. 3, the proposed model is constituted of four integral components, *i.e.*, feature extraction, residual in residual deep feature extraction, upsampling and reconstruction. It employs the same residual structure and dense connections to DRLN [1]. A longer skip connection is also added to connect the input to the output with a trainable parameter, which can greatly reduces the difficulty of optimization and thus, the network would pay more attention to the learning of the high frequency parts in images. As shown in Fig. 4, three Basic Residual Block (BRB) forms a Large Residual Block (LRB) with dense connection. Self-Calibration convolution (SCC) [22], shown at top of Fig. 4, is adopted as a basic unit in order to expand receptive field. Unlike conventional convolution, SCC enables each point in space to have interactive information from nearby regions and channels. Dense connections are established between the Self-Calibration convolution block (SCCB), each densely connected residual block has three SCCB. To incorporate channel information efficiently, an attention block with multi-scale feature integration is added in every basic residual block as DRLN [1]. For the network optimization, L_1 loss function was introduced as pixel-wise loss. In order to improve the fidelity, SSIM and MS-SSIM loss were also used as structure loss. With pixel loss and structure loss, the total loss is formulated as follows,

$$\mathcal{L}_{total} = \mathcal{L}^{L_1} + 0.2 \cdot \mathcal{L}^{MS-SSIM} + 0.2 \cdot \mathcal{L}^{SSIM}$$

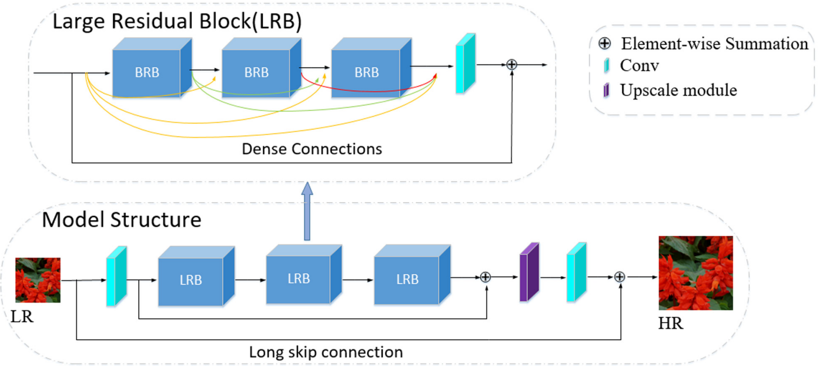


Fig. 3. The detailed network architecture of the proposed network for the OPPO_CAMERA team

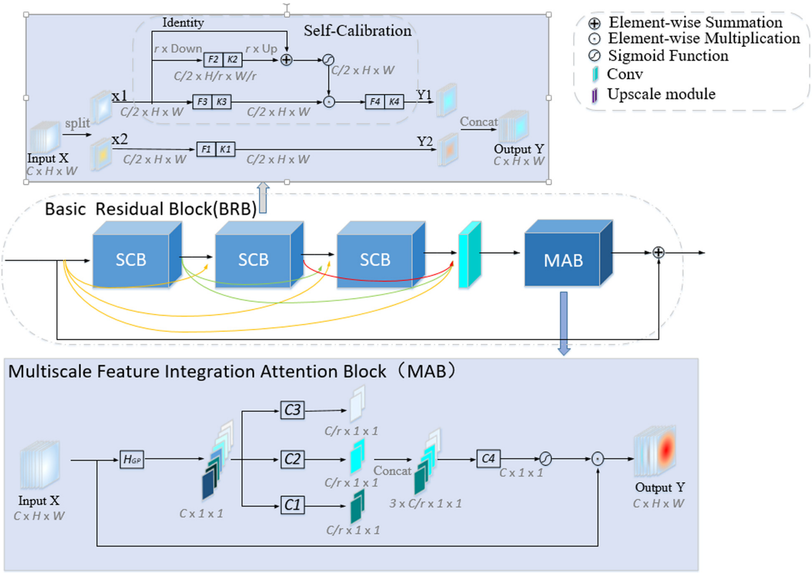


Fig. 4. The proposed BRB and MAB for the OPPO_CAMERA team. The top of the figure shows the basic convolution structure of the proposed network with the dense connection. The middle of the figure shows the basic residual block. The bottom of the figure presents the channel attention mechanism of the network.

For the training, the proposed method splits the training data randomly into two parts, *i.e.*, training set and validation set, with the ratio of 18500:500. Considering its significant improvement in the Real World SR task, CutBlur [45] is applied to augment training images. Self-ensemble and Parameter-fusion strategy would obviously improve the fidelity index(PSNR and SSIM), and meanwhile, less noise in result images. The strategy of self-ensembles (×8) was used as explained in RCAN [52], and all the corresponding parameters of last 3 models are fused to derive a fused model G_{fused}, as described in [30]. Experiments are conducted with Tesla V100 GPU.

AiAiR

The AiAiR team proposes that orientation-aware convolutions meet dual path enhancement network (OADDet). Their method consists of four basic models (model ensemble): OADDet, Deep-OADDet, original EDSR [21] and original DRLN [1]. The core modules of OADDet, illustrated in Fig. 5, are borrowed from DDet [31], Inception [33] and OANet [9] with minor improvements, such as less attention modules, removing skip connections and replacing ReLU with LeakyReLU. Overall architectures are similar to DDet [31]. It is found that redundant attention modules will damage the performance and slow down the training process. Therefore, attention modules are only applied to the last few blocks of the backbone network and the last layer of the shallow network. Similar to RealSR [4], PixelConv is also utilized, which contributes to ~0.15 dB improvement on the validation set.

- The training process generally consists of four stages on three different datasets. The total training time is about 2000 GPU hours on V100.
- OADDet models are trained from scratch and download DIV2K pre-trained EDSR/DRLN from official links.
- DIV2K dataset is used to pre-train our OADDet models and use manually washed AIM2020 datasets to fine-tune all models (further details in GitHub README).
- Four models are trained using three different strategies:
  1) For OADDet: Pre-training on DIV2K (300 epochs) then fine-tuning on original AIM2020 ×2 dataset (600 epochs) and AIM2020 washed ×2 dataset (100 epochs).
  2) For Deep-OADDet: Pre-training on DIV2K (30 epochs) then fine-tuning on AIM2020 washed ×2 + ×3 dataset (350 epochs), AIM2020 washed ×2 dataset (350 epochs) and AIM2020 washed ×2 dataset (100 epochs).
  3) For EDSR/DRLN: Using DIV2K well-trained models then fine-tuning on washed AIM2020 ×2 dataset (1000 epochs).
- Self-ensemble (×8), model-ensemble (four models) and proposed "crop-ensemble" are conducted (further details in GitHub README Reproduce ×2 test dataset results).
- OADDet enjoys a more stable and faster training process than OANet, which introduces too many attention modules at the early stage of the networks. DDet proposes to use dynamic PixelConv with kernelsize = 5,7,9; however, it is proved that kernelsize = 3,5,7 works better during training and testing time.

TeamInception

The TeamInception team proposes learning Enriched Features for Real Image Restoration and Enhancement. MIRNet, recently introduced in [47], is utilized

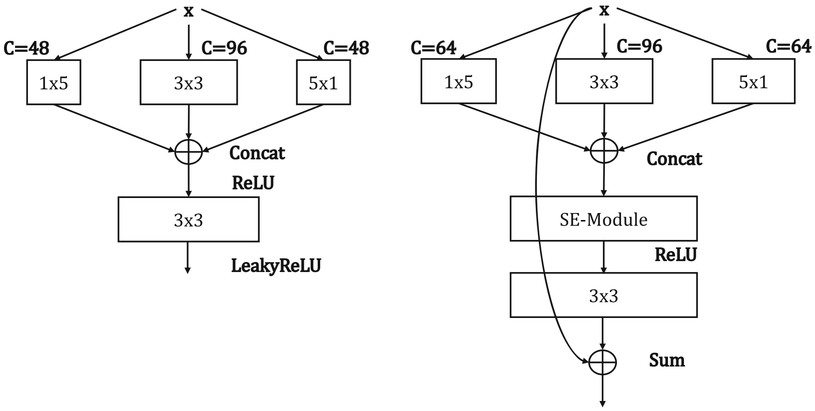


Fig. 5. OADDet and Deep-OADDet for the AiAiR team.

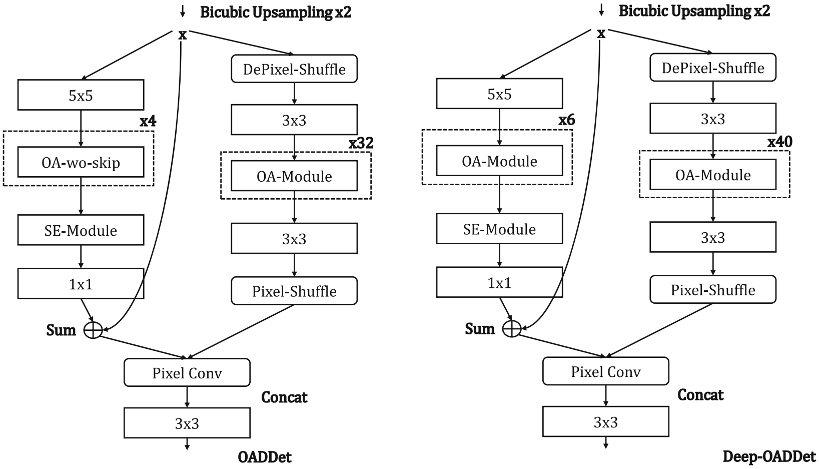


Fig. 6. Overall architectures of OADDet and Deep-OADDet for the AiAiR team.

with the collective goals of maintaining spatially-precise high-resolution representations through the entire network and receiving strong contextual information from the low-resolution representations. In Fig. 7. MIRNet[3] has a multi-scale residual block (MRB) containing several key elements: **(a)** parallel multi-resolution convolution streams for extracting (fine-to-coarse) semantically-richer and (coarse-to-fine) spatially-precise feature representations, **(b)** information exchange across multi-resolution streams, **(c)** attention-based aggregation of features arriving from multiple streams, and **(d)** dual-attention units to capture contextual information in both spatial and channel dimensions.

[3] The code is publicly available at https://github.com/swz30/MIRNet.

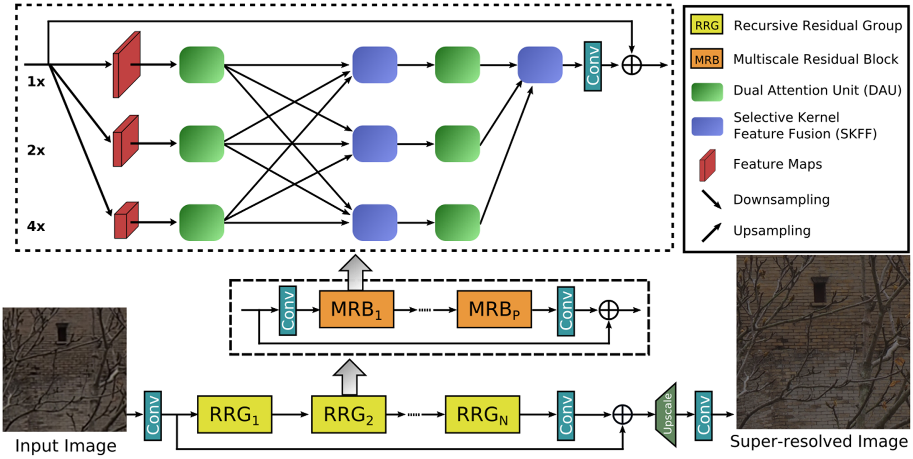

RRG Recursive Residual Group
MRB Multiscale Residual Block
Dual Attention Unit (DAU)
Selective Kernel Feature Fusion (SKFF)
Feature Maps
Downsampling
Upsampling
1x
2x
4x
Conv
MRB$_1$
MRB$_P$
RRG$_1$
RRG$_2$
RRG$_N$
Upscale
Input Image
Super-resolved Image

Fig. 7. Framework of the network MIRNet (recently introduced in [47]) for the Team-Inception team.

The MRB consists of multiple (three in this work) fully-convolutional streams connected in parallel. It allows information exchange across parallel streams in order to consolidate the high-resolution features with the help of low-resolution features, and vice versa. Each component of MRB is described as follows.

Selective Kernel Feature Fusion (SKFF). The SKFF module performs dynamic adjustment of receptive fields via two operations –*Fuse* and *Select*, as illustrated in Fig. 8. The *fuse* operator generates global feature descriptors by combining the information from multi-resolution streams. The *select* operator uses these descriptors to recalibrate the feature maps (of different streams) followed by their aggregation. Details of both operators for the three-stream case are elaborated as follows. **(1) Fuse:** SKFF receives inputs from three parallel convolution streams carrying different scales of information. We first combine these multi-scale features using an element-wise sum as: $\mathbf{L} = \mathbf{L_1} + \mathbf{L_2} + \mathbf{L_3}$. We then apply global average pooling (GAP) across the spatial dimension of $\mathbf{L} \in \mathbb{R}^{H\times W\times C}$ to compute channel-wise statistics $\mathbf{s} \in \mathbb{R}^{1\times 1\times C}$. Next, a channel-downscaling convolution layer is used to generate a compact feature representation $\mathbf{z} \in \mathbb{R}^{1\times 1\times r}$, where $r = \frac{C}{8}$ for our experiments. Finally, the feature vector $\mathbf{z}$ passes through three parallel channel-upscaling convolution layers (one for each resolution stream) and provides us with three feature descriptors $\mathbf{v_1}, \mathbf{v_2}$ and $\mathbf{v_3}$, each with dimensions $1 \times 1 \times C$. **(2) Select:** this operator applies the softmax function to $\mathbf{v_1}, \mathbf{v_2}$ and $\mathbf{v_3}$, yielding attention activations $\mathbf{s_1}, \mathbf{s_2}$ and $\mathbf{s_3}$ that we use to adaptively recalibrate multi-scale feature maps $\mathbf{L_1}, \mathbf{L_2}$ and $\mathbf{L_3}$, respectively. The overall process of feature recalibration and aggregation is defined as: $\mathbf{U} = \mathbf{s_1} \cdot \mathbf{L_1} + \mathbf{s_2} \cdot \mathbf{L_2} + \mathbf{s_3} \cdot \mathbf{L_3}$. Note that the SKFF uses ∼6× fewer parameters than aggregation with the concatenation but generates more favorable results.

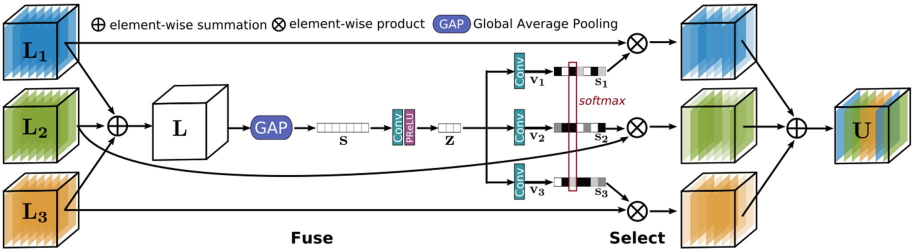


Fig. 8. Schematic for selective kernel feature fusion (SKFF) for the TeamInception team. It operates on features from multiple convolutional.

Dual Attention Unit (DAU). While the SKFF block fuses information across multi-resolution branches, we also need a mechanism to share information within a feature tensor, both along the spatial and the channel dimensions. The dual attention unit (DAU) is proposed to extract features in the convolutional streams. The schematic of DAU is shown in Fig. 9. The DAU suppresses less useful features and only allows more informative ones to pass further. This feature recalibration is achieved by using channel attention [14] and spatial attention [41] mechanisms. **(1) Channel attention (CA)** branch exploits the inter-channel relationships of the convolutional feature maps by applying *squeeze* and *excitation* operations [14]. Given a feature map $\mathbf{M} \in \mathbb{R}^{H \times W \times C}$, the squeeze operation applies global average pooling across spatial dimensions to encode global context, thus yielding a feature descriptor $\mathbf{d} \in \mathbb{R}^{1 \times 1 \times C}$. The excitation operator passes $\mathbf{d}$ through two convolutional layers followed by the sigmoid gating and generates activations $\hat{\mathbf{d}} \in \mathbb{R}^{1 \times 1 \times C}$. Finally, the output of CA branch is obtained by rescaling $\mathbf{M}$ with the activations $\hat{\mathbf{d}}$. **(2) Spatial attention (SA)** branch is designed to exploit the inter-spatial dependencies of convolutional features. The goal of SA is to generate a spatial attention map and use it to recalibrate the incoming features $\mathbf{M}$. To generate the spatial attention map, the SA branch first independently applies global average pooling and max pooling operations on features $\mathbf{M}$ along the channel dimensions and concatenates the outputs to form a feature map $\mathbf{f} \in \mathbb{R}^{H \times W \times 2}$. The map $\mathbf{f}$ is passed through a convolution and sigmoid activation to obtain the spatial attention map $\hat{\mathbf{f}} \in \mathbb{R}^{H \times W \times 1}$, which is used to rescale $\mathbf{M}$.

For training, L_1, multi-scale SSIM and VGG loss functions are considered in the model, defined as follows

$$\mathcal{L}_f = \alpha \mathcal{L}_1(\hat{\mathbf{y}}, \mathbf{y}) + \beta \mathcal{L}_{\text{MS-SSIM}}(\hat{\mathbf{y}}, \mathbf{y}) + \gamma \mathcal{L}_{\text{VGG}}(\hat{\mathbf{y}}, \mathbf{y}) \tag{2}$$

$\mathcal{L}_{\text{VGG}}$ uses the features of *conv2* layer after ReLU in the pre-trained VGG-16 network. Three RRGs are utilized, each of which contains 2 MRBs. MRB consists of 3 parallel streams with channel dimensions of $64, 128, 256$ at resolutions $1, \frac{1}{2}, \frac{1}{4}$, respectively. Each stream has 2 DAUs. Patches with the size of 128×128 are cropped. Horizontal and vertical flips are employed for data augmentation. The model is trained from scratch with the Adam optimizer ($\beta_1 = 0.9$, and $\beta_2 = 0.999$) for 7×10^5 iterations. The initial learning rate is 2×10^{-4} and the batch

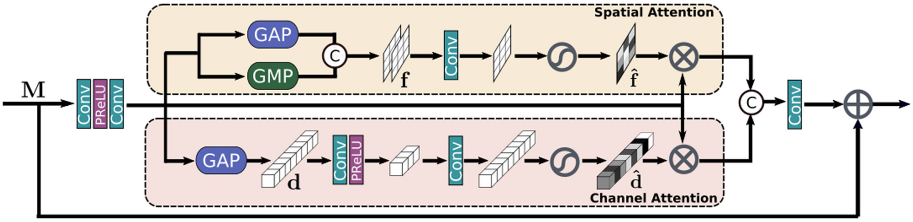


Fig. 9. Dual attention unit incorporating spatial and channel attention mechanisms for the TeamInception team.

size is 16. The cosine annealing strategy is employed to steadily decrease the learning rate from the initial value to 10^{-6} during training.

At inference time, the self-ensemble strategy [2] is employed. For each test image, a set of following 8 images are created: original, flipped, rotated 90°, rotated 180°, rotated 270°, 90° & flipped, 180° & flipped, and 270° & flipped. Next, these transformed images are passed through our model and obtain super-resolved outputs. Then we undo the transformations and perform averaging to obtain the final image. To fuse results, three different variants of the proposed networks are trained with different loss functions (Eq. 2): **(1)** only the first term, **(2)** the first two terms (i.e., $\alpha\mathcal{L}_1 + \beta\mathcal{L}_{\text{MS-SSIM}}$), and **(3)** all the terms. For the variant 2, $\alpha = 0.16$ and $\beta = 0.84$; for the variant 3, $\alpha = 0.01$ and $\beta = 0.84$, $\gamma = 0.15$.

Given an image, the generated self-ensembled results with each of these three networks are averaged to obtain the final image. Results with self-ensemble strategy and fusion are reported in Table 3. With 4 Tesla-V100 GPUs, it takes ~3 days to train the network. The time required to process a test image of size 3780 × 5780 is 2 s (single method), 30 s (self-ensemble) and 87 s (fusion).

Noah_TerminalVision

The Noah_TerminalVision team proposed Super Resolution with weakly-paired data using an Adaptive Robust Loss. The network is based on RRDBNet with 23 Residual in Residual Denseblocks. Only training pairs with a high PSNR score were used for training. To further alleviate the bad effect of miss-alignment of training data, the adaptive robust loss function proposed by Jon Barron was used. For track 3, it additionally used a spatial attention module and an efficient channel attention module. The spatial attention module is borrowed from EDVR [37] and the efficient attention module is borrowed from ECA-Net [36]. Considering that the training data are not perfectly aligned, Adaptive Robust Loss Function [2] for super resolution tasks is utilized to solve the weakly-paired training problem. The self-ensemble strategy is to run inference on the combination of the 90/180/270-degree rotated images of the original/flipped input and then to average the results.

Only training pairs with a high PSNR score (29) were used for training. The learning rate is 2e−4, the patch size of inputs is 80 × 80 and the batchsize is 4. CosineAnnealingLR_Restart learning rate scheme is employed and the restart

Table 3. Results of validation set for the scale factor ×4 for the TeamInception team. Comparison of using single method (SM), self-ensemble (SE) and Fusion (F) on validation set.

	$\mathcal{L}_1$	$\mathcal{L}_1 + \mathcal{L}_{\text{MS-SSIM}}$	$\mathcal{L}_1 + \mathcal{L}_{\text{MS-SSIM}} + \mathcal{L}_{\text{VGG}}$	PSNR
SM	√			29.72
SM		√		29.83
SM			√	29.89
SM + F	√	√	√	30.08
SE + F	√	√	√	30.25

period is 250,000 steps. For each input, due to GPU memory constraint, images are tested patch-wisely. The crop window is of size 120 × 120, and a stride of 110 × 110 was used to collect patches.

DeepBlueAI

The DeepBlueAI team proposed a solution based on RCAN [51], which was implemented with PyTorch. In each RG, the RCAB number is 20, G = 10 and C = 128 in the RIR structure. The model is trained from scratch, which costs about 4 days with 4 × 32G Tesla V100 GPU. For training, all the training images are augmented by random horizontal flips and 90 rotations. In each training batch, LR color patches with the size of 64 × 64 are extracted as inputs. The initial leaning rate is set to 2.0×10^{-4} and learning rate of each parameter group use a cosine annealing schedule with total 1.0×10^{5} iterations and without restart. For testing, each low resolution image is flipped and rotated to generate seven augmented inputs; with the trained RCAN model, the corresponding super-resolved images are generated. An inverse transform is applied to those output images to get the original geometry. The transformed outputs are averaged all together to yield the self-ensemble result.

ALONG

The ALONG team proposed Dual Path Network with high frequency guided for real-world image Super-Resolution. The proposed method follows the main structure of RCAN [51] and utilizes the guild filter to decompose the detail layer and to restore high-frequency details. As illustrated in Fig. 10, a lot of share-source skip connections in the original feature extraction path with channel attention. Due to share-source skip connections, the abundant low-frequency information can be bypassed and facilitate to train deeper network. Compared with the previous simulated datasets, the image degradation process for real SR is much more complicated. Low-resolution images lose more high-frequency information and look blurry. Inspired by other image deblurring tasks [37,54,55], a pre-deblur module is used before the residual groups to pre-process blurry inputs and improve super-resolution accuracy. Specifically, the input image is first down-sampled with strided convolution layers; then the upsampling layer at the end will resize the features back to the original input resolution. The

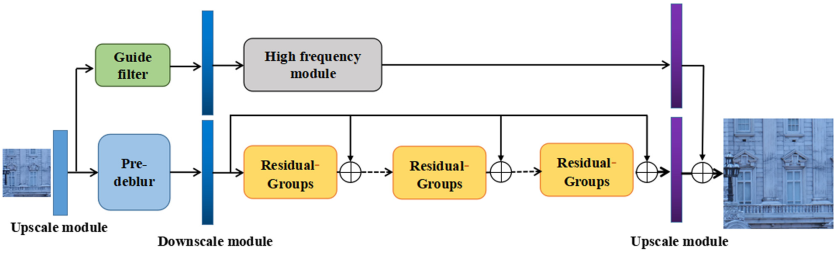


Fig. 10. RCAN for the Real Image Super-Resolution (RCANv2) for the ALONG team.

proposed dual path network restores fine details by decomposing the input image and focusing on the detail layers. An additional branch focuses on the high-frequency reconstruction. The input LR image is decomposed into the detail layer using the guided filter, an edge-preserving low-pass filter [13]. Then a high-frequency module is adopted on the detail layer, so the output result can focus on restoring high-frequency details.

Besides, a variety of data augmentation strategies are combined to achieve competitive results in different tracks, including Cutout [7], CutMix [46], Mixup [48], CutMixup, RGB permutation, Blend. In addition, inspired by [45], CutBlur, unlike Cutout, can utilize the entire image information while it enjoys the regularization effect due to the varied samples of random HR ratios and locations. The experimental results also show that a reasonable combination of data enhancement can improve the model performance without additional computation cost in the test phase. The model is trained with 8 2080Ti, 11G memory each GPU. Pseudo ensemble is also employed. The inputs are flipped/rotated and the HR results are aligned and averaged for enhanced prediction.

LISA-ULB

The LISA-ULB team proposed VCycles BackProjection networks generation two (VCBPv2), which utilized an iterative error correcting feedback mechanism to guide the reconstruction of the final SR output. As shown in Fig. 11, the proposed network is composed of an outer loop of 10 cycles and an inner loop of 3 cycles. The input of the proposed VCBPv2 is the LR image and the upsampled counterpart. The upsample and downsample modules iteratively transform features between high- and low-resolution space as residual for error correction. The decoder in the end reconstructs the corrected feature to SR image.

The model is trained using AdamW optimizer with learning rate of 1×10^{-4} and halved at every 400 epochs, then the training is followed by SGDM optimizer. Equally weighted ℓ_1 and SSIM loss is adopted for training.

lyl

The lyl team proposed a coarse to fine network for progressive super-resolution. As shown in Fig. 12, based on the Laplacian pyramid framework, the proposed model takes an LR image as input and progressively predicts residual images at $S_1, S_2 ... S_n$ levels. S is the scale factor, $S = S_1 \times S_2 ... \times S_n$, where $n = log_2^S$.

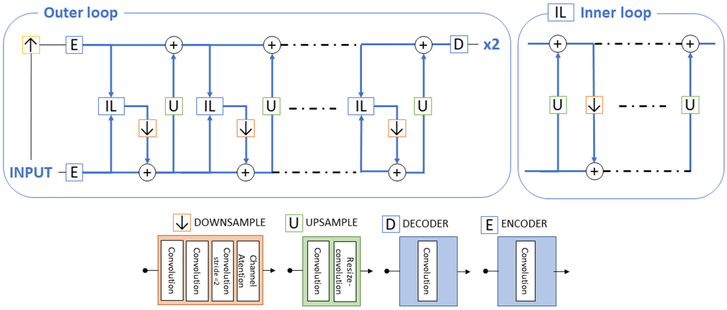
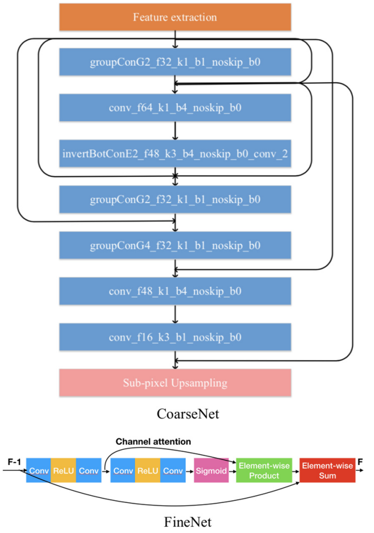


Fig. 11. The architecture of the proposed network by the LISA-ULB team.



CoarseNet

FineNet

Fig. 12. The architecture of the proposed network by the lyl team.

ℓ_1 was adopted to optimize the proposed network. Each level of the proposed CFN was supervised by different scales of HR images.

GDUT-SL

The GDUT-SL team used the RRDBNet of ESRGAN [38] to perform super-resolution. Typical RRDB block has 3 Dense blocks, which including 5 Conv layers with Leaky-ReLU and remove BN layers. The RRDB number was set to 23. Two UpConv layer is used for upsampling. Different from ESRGAN, the GDUT-SL team replaced the activation function with ReLU to obtain better PSNR results.

Residual scaling and smaller initialization were adopted to facilitate training a deep architecture. In training phase, the mini-batch size was set to 16, with image size of 96 × 96. 20 promising models were selected for model-ensemble.

MCML-Yonsei

As shown in Fig. 13, the MCML-Yonsei team proposed an attention based multi-scale deep residual network based on MDSR [20], which shares most of the parameters across different scales. In order to utilize various features in each real image adaptively, the MCML-Yonsei team added an attention module in the existing Resblock. As shown in Fig. 14, the attention module is based on MAMNet [17] where the global variance pooling was replaced with total variation pooling.

They initialized all parameters except the attention module with the pre-trained MDSR, which was optimized for bicubic downsampling based training data. The mini-batch size was set to 16 and the patch size was set to 48. They subtracted the mean of each R, G, B channel of the train set for data normalization. The learning rate was initially set to $1e-4$, and it decayed at the 15k steps. The total training step was 20k.

kailos

The kailos team proposed RRBD Network with Attention mechanism using Wavelet loss for Single Image Super-Resolution. The loss function consisted of conventional L_1 loss $\mathcal{L}_{L_1}$ and novel wavelet loss $\mathcal{L}_{wavelet}$. The conventional L_1 loss $\mathcal{L}_{L_1}$ is given as $\mathcal{L}_{L_1} = \sum \mid x - y \mid_1$, where x is reconstructed image and y is ground truth image.

A wavelet transform can separate the signal features along the low and high frequency components. Most of the energy distribution in the signal, such as global structure and color distribution, is concentrated in the low frequency components. On the other hand, the high frequency components include signal patterns and image textures. Since both frequency components have different characteristics, a different loss function must be applied to each component. Therefore, the proposed novel wavelet loss $\mathcal{L}_{wavelet}$ is the sum of L_1 loss for high frequency components and L_2 loss for low frequency components given as $\mathcal{L}_{high} = \sum_{i=1}^{N} \mid \Psi_H^i(x) - \Psi_H^i(y) \mid_1$, $\mathcal{L}_{low} = \sum_{i=1}^{N} \parallel \Psi_L^i(x) - \Psi_L^i(y) \parallel_2^2$, and $\mathcal{L}_{wavelet} = \mathcal{L}_{low} + \mathcal{L}_{high}$, where N denotes the stage of wavelet transform and Ψ_H and Ψ_L are high and low frequency decomposition filters, respectively.

In the experiment, N is 2 and Haar wavelet filters are used as wavelet decomposition filters. Therefore, a total loss is defined by $\mathcal{L}_{total} = \mathcal{L}_{L_1} + \lambda\ \mathcal{L}_{wavelet}$, where λ denotes the regularization parameter and $\lambda = 1$ was used in the proposed method. Figure 16 shows an overview of the proposed method. Adam optimizer was used in training process, and the size of image patch was the quarter size of training data.

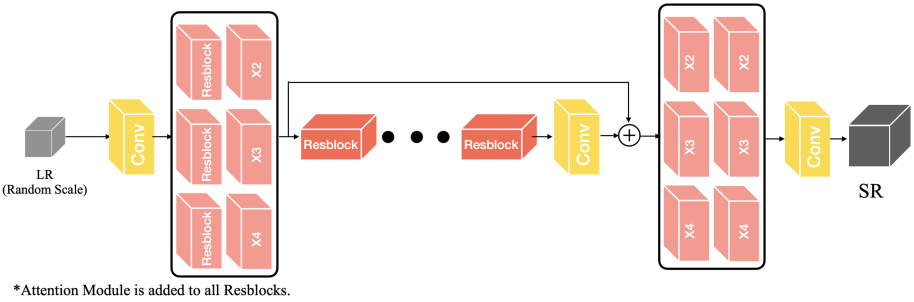


Fig. 13. Overview of the network for the MCML-Yonsei team.

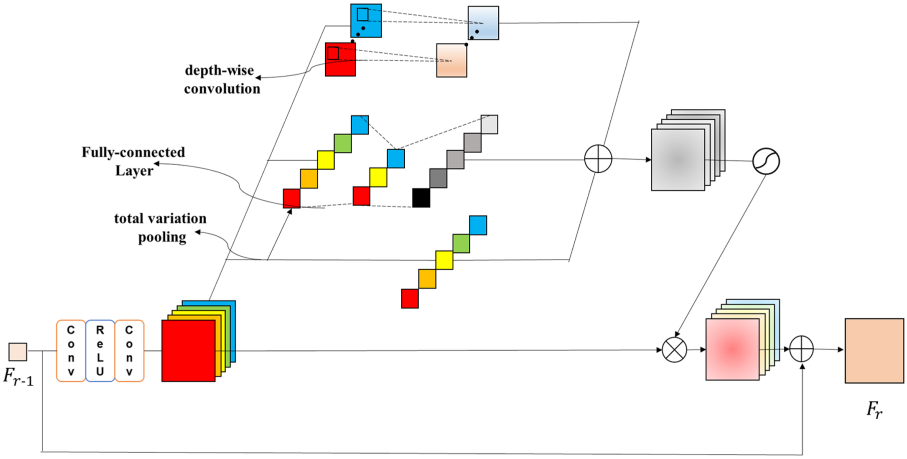


Fig. 14. Resblock with attention module for the MCML-Yonsei team.

qwq

The qwq team proposed a Multi-Scale Network based on RCAN [51]. As shown in Fig[1], the multi-scale mechanism was integrated into the base block of RCAN in order to enlarge the receptive field. Dual Loss was adopted for training. Mix-Corrupt augmentation was conducted, for it allowed the network to learn from robust SR results from different degradations, which is specially designed for the real-world scenario.

RRDN_IITKGP

The RRDN_IITKGP used a GAN based Residual in Residual Dense Network [38], where the model is pre-trained on other dataset and evaluated on the challenge dataset.

SR-IM

The SR-IM team proposed frequency-aware network [29], as shown in Fig. 17. A hierarchical feature extractor (HFE) is utilized to extract the high representation,

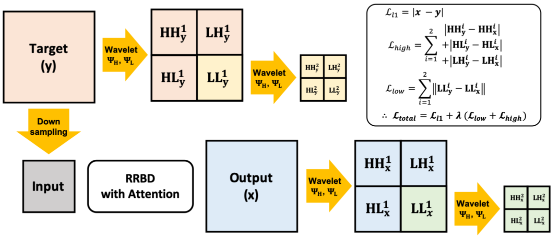

Target (y)
Wavelet Ψ_H, Ψ_L
HH_y^1
LH_y^1
HL_y^1
LL_y^1
Wavelet Ψ_H, Ψ_L
HH_y^2
LH_y^2
HL_y^2
LL_y^2
$\mathcal{L}_{l1} = |x - y|$
$\mathcal{L}_{high} = \sum_{i=1}^{2} |HH_y^i - HH_x^i| + |HL_y^i - HL_x^i| + |LH_y^i - LH_x^i|$
$\mathcal{L}_{low} = \sum_{i=1}^{2} \|LL_y^i - LL_x^i\|$
$\therefore \mathcal{L}_{total} = \mathcal{L}_{l1} + \lambda (\mathcal{L}_{low} + \mathcal{L}_{high})$
Down sampling
Input
RRBD with Attention
Output (x)
Wavelet Ψ_H, Ψ_L
HH_x^1
LH_x^1
HL_x^1
LL_x^1
Wavelet Ψ_H, Ψ_L
HH_x^2
LH_x^2
HL_x^2
LL_x^2

Fig. 15. Overview of the proposed method of for the kailos team.

middle representation and low representation. The basic unit of the body consists of residual dense block and channel attention module. Finally, the three branches are fused into one super-resolved image by the gate and fusion module.

The mini-batch size was set to 8 and the patch size was set to 160 during training. They used Adam optimizer with an initial learning rate of 0.0001. The learning rate decayed by a factor of 0.5 every 30 epochs. The entire training time is about 48 h.

JNSR

The JNSR team utilized EDSR [20] and DRLN [1] to perform model ensemble. The EDSR and DRLN were trained on AIM2020 dataset, the best models were chosen for model ensemble.

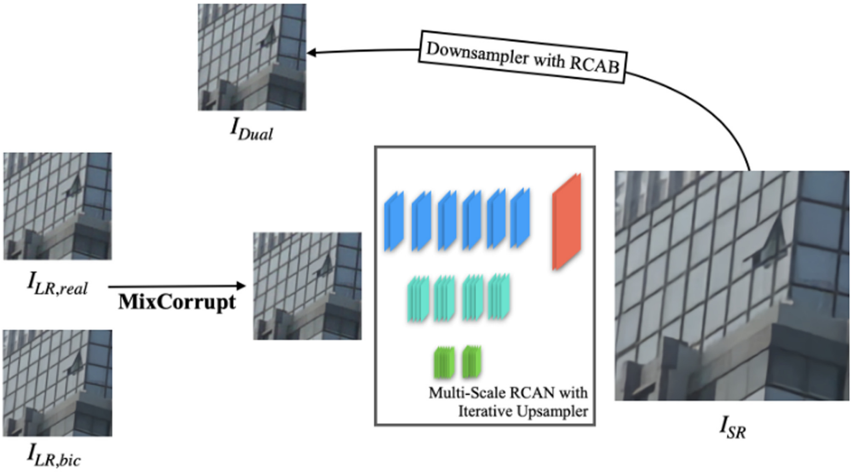
SR_DL

The SR_DL team proposed attention back projection network (ABPN++), as shown in Fig. 18. The proposed ABPN++ network first conducts feature extraction to expand the feature space of the input LR image. Then the densely connected enhanced down- and up-sampling back projection blocks perform up- and down-sampling the feature maps. The Cross-scale Attention Block (CAB) takes the outputs from down-sampling back projection blocks to compute the cross-correlation for feature fusion. Finally, the Refined Back Projection Block works as a final refinement that estimates the feature residuals between input LR and predicted LR images for update. The complete network includes 10 down- and up-sampling back projection block, 2 feature extraction blocks and 1 refined back projection block. Each back projection block is made of 5 convolutional layers. The kernel number is 32 for all convolution and deconvolution layers. For down- and up-sampling convolution layer, the kernel size is 6, stride is 4 and padding is 1.

The mini-batch size was set to 16 and the LR patch size was set to 48 during training. The learning rate is fixed to 1e-4 for all layers for 2×10^5 iterations in total as the first stage. Then the batch size increases to 32 for 1×10^5 iterations as fine-tuning.

Webbzhou

The Webbzhou team fine-tuned the pre-trained RRDB [38] on the challenge dataset.



$$L_{total} = \|I_{SR} - I_{GT}\|_1 + 0.01 * SSIM(I_{SR}, I_{GT}) + 0.001 * \|I_{Dual} - I_{LR}\|_2$$

Fig. 16. The total learning diagram of for the qwq team. In upsample network, they used features from 0.25×, 1×, 2× and 4×(HR) five scales.

MoonCloud

The MoonCloud team utilized RCAN [51] for the challenge. Totally 6 models were used for model ensemble. Three of them were trained on challenge dataset with scale of 4. The other three were trained on the challenge dataset with scale of 3, which were fine-tuned on the dataset with scale of 4 after. The final outputs were obtained by averaging the outputs of these six models.

SrDance

The SrDance team utilized RRDB [38]. A new training strategy was adopted for model optimization. The model was firstly pre-trained on DIV2K dataset. Then they trained their model by randomly picking one image in dataset and randomly crop a few 40×40 patches, which is alike stochastic gradient descent. Second, when model stepped, they trained on 10 pics, one 40×40 patch from each picture and fed to the model.

MLP_SR

The MLP_SR team proposed Deep Cyclic Generative Adversarial Residual Convolutional Networks for Real Image Super-Resolution [35], as shown in Fig. 19. The SR generator [34] network G_{SR} was trained in a GAN framework by using the LR ($\mathbf{y}$) images with their corresponding HR images with pixel-wise supervision in the clean HR target domain ($\mathbf{x}$), while maintaining the cyclic consistency between the LR and HR domain.

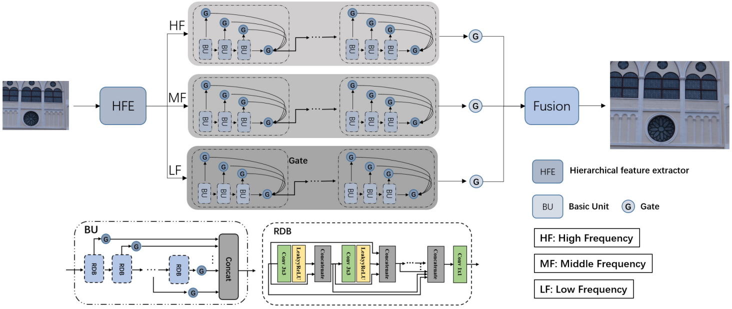


Fig. 17. Structure of Frequency-aware Network (FAN) for the SR-IM team. There are three branches, representing the high frequency, middle frequency and low frequency components. The gate attention is used to adaptively select the required frequency components.

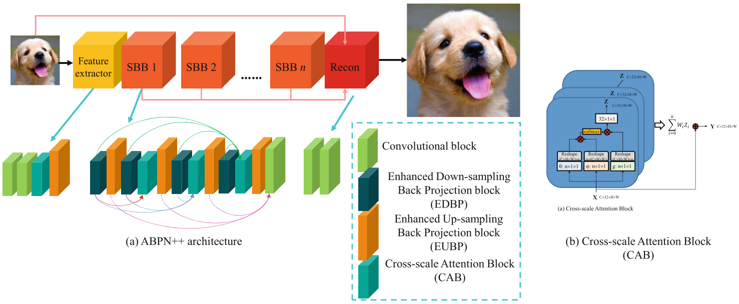

Feature extractor
SBB 1
SBB 2
......
SBB n
Recon
(a) ABPN++ architecture
Convolutional block
Enhanced Down-sampling Back Projection block (EDBP)
Enhanced Up-sampling Back Projection block (EUBP)
Cross-scale Attention Block (CAB)
softmax
Reshape
(a) Cross-scale Attention Block
(b) Cross-scale Attention Block (CAB)

Fig. 18. (a): ABPN++: Attention based Back Projection Network for image super-resolution. (b): the proposed Cross-scale Attention Block by the SR_DL team.

congxiaofeng

The congxiaofeng team proposed RDB-P SRNet, which contains several residual-dense blocks with pixel shuffle for upsampling. The network was inspired by RDN [53].

debut_kele

The debut_kele team proposed Enhanced Deep Residual Networks for real image super-resolution.

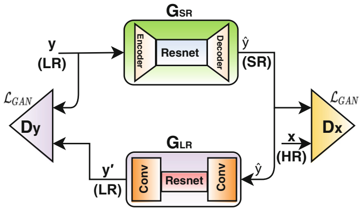


Fig. 19. Illustration of the structure of SR approach setup proposed by the MLP_SR team.

Acknowledgements. We thank the AIM 2020 sponsors: Huawei, MediaTek, NVIDIA, Qualcomm AI Research, Google and Computer Vision Lab (CVL) ETH Zurich.

A. Teams and Affiliations

AIM2020 team

Title: AIM 2020 Challenge on Real Image Super-Resolution
Members:
Pengxu Wei[1] (`weipx3@mail.sysu.edu.cn`),
Hannan Lu[2] (`hannanlu@hit.edu.cn`),
Radu Timofte[3] (`radu.timofte@vision.ee.ethz.ch`),
Liang Lin[1] (`linliang@ieee.org`),
Wangmeng Zuo[2] (`cswmzuo@gmail.com`)
Affiliations:
[1] Sun Yat-sen University, China
[2] Harbin Institute of Technology University, China
[3] Computer Vision Lab, ETH Zurich, Switzerland

Baidu

Title: Real Image Super Resolution via Heterogeneous Model Ensemble using GP-NAS
Members: Zhihong Pan[1] (`zhihongpan@baidu.com`), Baopu Li[1] Teng Xi[2], Yanwen Fan[2], Gang Zhang[2], Jingtuo Liu[2], Junyu Han[2], Errui Ding[2]
Affiliation:
[1] Baidu Research (USA)
[2] Department of Computer Vision Technology (VIS), Baidu Incorportation

CETC-CSKT

Title: Adaptive dense connection super resolution reconstruction
Members: Tangxin Xie (`xxh96@outlook.com`), Yi Shen, Jialiang Zhang, Yu Jia, Liang Cao, Yan Zou
Affiliation: China Electronic Technology Cyber Security Co., Ltd.

OPPO_CAMERA
Title: Self-Calibrated Attention Neural Network for Real-World Super Resolution
Members: Kaihua Cheng (chengkaihua@oppo.com), Chenhuan Wu
Affiliation: Guangdong OPPO Mobile Telecommunications Corp., Ltd.

ALONG
Title: Dual Path Network with High Frequency Guided for Real World Image Super-Resolution
Members: Yue Lin (gzlinyue@corp.netease.com), Cen Liu, Yunbo Peng
Affiliation: NetEase Games AI Lab

Noah_TerminalVision
Title: Super Resolution with weakly-paired data using an Adaptive Robust Loss
Members: Xueyi Zou (zouxueyi@huawei.com),
Affiliation: Noah's Ark Lab, Huawei

DeepBlueAI
Title: A solution based on RCAN
Members: Zhipeng Luo, Yuehan Yao (yaoyh@deepblueai.com), Zhenyu Xu
Affiliation: DeepBlue Technology (Shanghai) Co., Ltd

TeamInception
Title: Learning Enriched Features for Real Image Restoration and Enhancement
Members: Syed Waqas Zamir (waqas.zamir@inceptioniai.org), Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan
Affiliation: Inception Institute of Artificial Intelligence (IIAI)

MCML-Yonsei
Title: Multi-scale Dynamic Residual Network Using Total Variation for Real Image Super-Resolution
Members: Keon-Hee Ahn (khahn196@gmail.com), Jun-Hyuk Kim, Jun-Ho Choi, Jong-Seok Lee
Affiliation: Yonsei University

lyl
Title: Coarse to Fine Pyramid Networks for Progressive image super-resolution
Members: Tongtong Zhao (daitoutiere@gmail.com), Shanshan Zhao
Affiliation: Dalian Maritime Univerity

kailos
Title: RRDB Network with Attention mechanism using Wavelet loss for Single Image Super-Resolution

Members: Yoseob Han[1] (yoseobhan@lanl.gov), Byung-Hoon Kim[2], JaeHyun Baek[3]
Affiliation:
[1] Loa Alamos National Laboratory (LANL)
[2] Korea Advanced Institute of Science and Technology (KAIST)
[3] Amazon Web Services (AWS)

qwq
Title: Dual Learning for SR using Multi-Scale Network
Members: Haoning Wu, Dejia Xu *Affiliation:* Peking University

AiAiR
Title: OADDet: Orientation-aware Convolutions Meet Dual Path Enhancement Network
Members: Bo Zhou[1] (1826356001@qq.com),
Haodong Yu[2] (haodong.yu@outlook.com)
Affiliation:
[1] Jiangnan University
[2] Karlsruher Institut fuer Technologie

JNSR
Title: Dual Path Enhancement Network
Members: Bo Zhou (jeasonzhou1@gmail.com)
Affiliation: Jiangnan University

SrDance
Title: Training Strategy Optimization
Members: Wei Guan (missanswer@163.com), Xiaobo Li, Chen Ye
Affiliation: Tongji University

GDUT-SL
Title: Ensemble of RRDB for Image Restoration
Members: Hao Li (2111903004@mail2.gdut.edu.cn), Haoyu Zhong, Yukai Shi, Zhijing Yang, Xiaojun Yang
Affiliation: Guangdong University of Technology

MoonCloud
Title: Mixed Residual Channel Attention
Members: Haoyu Zhong (hy0421@outlook.com), Yukai Shi, Xiaojun Yang, Zhijing Yang,
Affiliation: Guangdong University of Technology,

SR-IM
Title: FAN: Frequency-aware network for image super-resolution

Members: Xin Li (lixin666@mail.ustc.edu.cn), Xin Jin, Yaojun Wu, Yingxue Pang, Sen Liu
Affiliation: University of Science and Technology of China

SR_DL
Title: ABPN++: Attention based Back Projection Network for image super-resolution
Members: Zhi-Song Liu[1], Li-Wen Wang[2], Chu-Tak Li[2], Marie-Paule Cani[1], Wan-Chi Siu[2]
Affiliation:
[1] LIX - Computer science laboratory at the Ecole polytechnique [Palaiseau]
[2] Center of Multimedia Signal Processing, The Hong Kong Polytechnic University

Webbzhou
Title: RRDB for Real World Super-Resolution
*Members:*Yuanbo Zhou (webbozhou@gmail.com),
Affiliation: Fuzhou University, Fujian Province, China

MLP SR
Title: Deep Cyclic Generative Adversarial Residual Convolutional Networks for Real Image Super-Resolution
Members: Rao Muhammad Umer (engr.raoumer943@gmail.com), Christian Micheloni
Affiliation: University Of Udine, Italy

congxiaofeng
Title: RDB-P SRNet: Residual-dense block with pixel shuffle
Members: Xiaofeng Cong (1752808219@qq.com)
Affiliation: (Not provided)

RRDN_IITKGP
Title: A GAN based Residual in Residual Dense Network
Members: Rajat Gupta (rajatgba2021@email.iimcal.ac.in)
Affiliation: Indian Institute of Technology

debut_kele
Title: Self-supervised Learning for Pretext Training
Members: Kele Xu (kelele.xu@gmail.com), Hengxing Cai, Yuzhong Liu
Affiliation: National University of Defense Technology

Team-24
Title: VCBPv2 - VCycles Backprojection Upscaling Network
Members: Feras Almasri, Thomas Vandamme, Olivier Debeir
Affiliation: Universié Libre de Bruxelles, LISA department

References

1. Anwar, S., Barnes, N.: Densely residual laplacian super-resolution. arXiv preprint arXiv:1906.12021 (2019)
2. Barron, J.T.: A general and adaptive robust loss function. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4331–4339 (2019)
3. Cai, J., Gu, S., Timofte, R., Zhang, L.: Ntire 2019 challenge on real image super-resolution: methods and results. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2019
4. Cai, J., Zeng, H., Yong, H., Cao, Z., Zhang, L.: Toward real-world single image super-resolution: a new benchmark and a new model. In: International Conference on Computer Vision (2019)
5. Chen, C., Xiong, Z., Tian, X., Zha, Z., Wu, F.: Camera lens super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1652–1660 (2019)
6. Cheng, K., Wu, C.: Self-calibrated attention neural network for real-world super resolution. In: European Conference on Computer Vision Workshops (2020)
7. DeVries, T., Taylor, G.W.: Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552 (2017)
8. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: European Conference on Computer Vision, pp. 184–199 (2014)
9. Du, C., et al.: Orientation-aware deep neural network for real image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2019
10. El Helou, M., Zhou, R., Süsstrunk, S., Timofte, R., et al.: AIM 2020: scene relighting and illumination estimation challenge. In: European Conference on Computer Vision Workshops (2020)
11. Fuoli, D., Huang, Z., Gu, S., Timofte, R., et al.: AIM 2020 challenge on video extreme super-resolution: Methods and results. In: European Conference on Computer Vision Workshops (2020)
12. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 349–356 (2009)
13. He, K., Sun, J., Tang, X.: Guided image filtering. IEEE Trans. Pattern Anal. Machine Intell. **35**(6), 1397–1409 (2012)
14. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: CVPR (2018)
15. Ignatov, A., Timofte, R., et al.: AIM 2020 challenge on learned image signal processing pipeline. In: European Conference on Computer Vision Workshops (2020)
16. Ignatov, A., Timofte, R., et al.: AIM 2020 challenge on rendering realistic bokeh. In: European Conference on Computer Vision Workshops (2020)
17. Kim, J.H., Choi, J.H., Cheon, M., Lee, J.S.: Mamnet: multi-path adaptive modulation network for image super-resolution. Neurocomputing **402**, 38–49 (2020)

18. Ledig, C., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 105–114 (2017)
19. Li, Z., Xi, T., Deng, J., Zhang, G., Wen, S., He, R.: Gp-nas: gaussian process based neural architecture search. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2020
20. Lim, B., Son, S., Kim, H., Nah, S., Lee, K.M.: Enhanced deep residual networks for single image super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1132–1140 (2017)
21. Lim, B., Son, S., Kim, H., Nah, S., Lee, K.M.: Enhanced deep residual networks for single image super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (2017)
22. Liu, J.J., Hou, Q., Cheng, M.M., Wang, C., Feng, J.: Improving convolutional networks with self-calibrated convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 10096–10105 (2020)
23. Lugmayr, A., Danelljan, M., Timofte, R.: Unsupervised learning for real-world super-resolution. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3408–3416. IEEE (2019)
24. Lugmayr, A., Danelljan, M., Timofte, R.: Ntire 2020 challenge on real-world image super-resolution: methods and results. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020
25. Lugmayr, A., et al.: Aim 2019 challenge on real-world image super-resolution: methods and results. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3575–3583. IEEE (2019)
26. Ma, Y., Yu, D., Wu, T., Wang, H.: Paddlepaddle: an open-source deep learning platform from industrial practice. Front. Data Comput. **1**(1), 105–115 (2019)
27. Ntavelis, E., Romero, A., Bigdeli, S.A., Timofte, R., et al.: AIM 2020 challenge on image extreme inpainting. In: European Conference on Computer Vision Workshops (2020)
28. Pan, Z., Li, B., Xi, T., Fan, Y., Zhang, G., Liu, J., Han, J., Ding, E.: Real image super resolution via heterogeneous model ensemble using gp-nas. In: European Conference on Computer Vision Workshop (2020)
29. Pang, Y., Li, X., Jin, X., Wu, Y., Liu, J., Liu, S., Chen, Z.: FAN: frequency aggregation network for real image super-resolution. In: European Conference on Computer Vision Workshops (2020)
30. Shang, T., Dai, Q., Zhu, S., Yang, T., Guo, Y.: Perceptual extreme super-resolution network with receptive field block. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 440–441 (2020)
31. Shi, Y., Zhong, H., Yang, Z., Yang, X., Lin, L.: Ddet: Dual-path dynamic enhancement network for real-world image super-resolution. arXiv preprint arXiv:2002.11079 (2020)
32. Son, S., Lee, J., Nah, S., Timofte, R., Lee, K.M., et al.: AIM 2020 challenge on video temporal super-resolution. In: European Conference on Computer Vision Workshops (2020)
33. Szegedy, C., et al.: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
34. Umer, R.M., Foresti, G.L., Micheloni, C.: Deep generative adversarial residual convolutional networks for real-world super-resolution, pp. 1769–1777 (2020)
35. Umer, R.M., Micheloni, C.: Deep cyclic generative adversarial residual convolutional networks for real image super-resolution. In: European Conference on Computer Vision Workshops (2020)

36. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: Eca-net: efficient channel attention for deep convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 11534–11542 (2020)
37. Wang, X., Chan, K.C., Yu, K., Dong, C., Change Loy, C.: Edvr: video restoration with enhanced deformable convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (2019)
38. Wang, X., et al.: Esrgan: enhanced super-resolution generative adversarial networks. In: Proceedings of the European Conference on Computer Vision (2018)
39. Wei, P., Lu, H., Timofte, R., Lin, L., Zuo, W., et al.: AIM 2020 challenge on real image super-resolution. In: European Conference on Computer Vision Workshops (2020)
40. Wei, P., Xie, Z., Lu, H., Zhan, Z., Ye, Q., Zuo, W., Lin, L.: Component divide-and-conquer for real-world image super-resolution. In: European Conference on Computer Vision (2020)
41. Woo, S., Park, J., Lee, J.Y., So Kweon, I.: CBAM: Convolutional block attention module. In: ECCV (2018)
42. Xie, T., Li, J., Shen, Y., Jia, Y., Zhang, J., Zeng, B.: Enhanced adaptive dense connection single image super-resolution. In: European Conference on Computer Vision Workshops (2020)
43. Xie, T., Yang, X., Jia, Y., Zhu, C., Xiaochuan, L.: Adaptive densely connected single image super-resolution. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3432–3440. IEEE (2019)
44. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE Trans. Image Process. **19**(11), 2861–2873 (2010)
45. Yoo, J., Ahn, N., Sohn, K.A.: Rethinking data augmentation for image super-resolution: a comprehensive analysis and a new strategy. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8375–8384 (2020)
46. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 6023–6032 (2019)
47. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L.: Learning enriched features for real image restoration and enhancement. In: ECCV (2020)
48. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017)
49. Zhang, K., Danelljan, M., Li, Y., Timofte, R., et al.: AIM 2020 challenge on efficient super-resolution: methods and results. In: European Conference on Computer Vision Workshops (2020)
50. Zhang, X., Chen, Q., Ng, R., Koltun, V.: Zoom to learn, learn to zoom. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3762–3770 (2019)
51. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision, pp. 286–301 (2018)
52. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 286–301 (2018)
53. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision (2018)

54. Zhou, S., Zhang, J., Pan, J., Xie, H., Zuo, W., Ren, J.: Spatio-temporal filter adaptive network for video deblurring. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2482–2491 (2019)
55. Zhou, S., Zhang, J., Zuo, W., Xie, H., Pan, J., Ren, J.S.: Davanet: stereo deblurring with view aggregation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 10996–11005 (2019)