Abstract

This paper reviews the NTIRE 2020 challenge on real image denoising with focus on the newly introduced dataset, the proposed methods and their results. The challenge is a new version of the previous NTIRE 2019 challenge on real image denoising that was based on the SIDD benchmark. This challenge is based on a newly collected validation and testing image datasets, and hence, named SIDD+. This challenge has two tracks for quantitatively evaluating image denoising performance in (1) the Bayer-pattern rawRGB and (2) the standard RGB (sRGB) color spaces. Each track ~250 registered participants. A total of 22 teams, proposing 24 methods, competed in the final phase of the challenge. The proposed methods by the participating teams represent the current state-of-the-art performance in image denoising targeting real noisy images. The newly collected SIDD+ datasets are publicly available at: https://bit.ly/siddplus_data.

1. Introduction

Image denoising is a fundamental and active research area (e.g., [39, 47, 48, 15]) with a long-standing history in computer vision (e.g., [21, 24]). A primary goal of image denoising is to remove or correct for noise in an image, either for aesthetic purposes, or to help improve other downstream tasks. For many years, researchers have primarily relied on synthetic noisy image for developing and evaluating image denoisers, especially the additive white Gaussian noise (AWGN)—e.g., [9, 11, 47]. Recently, more focus has been given to evaluating image denoisers on real noisy images [3, 36, 4]. To this end, we have proposed this challenge as a means to evaluate and benchmark image denoisers on real noisy images.

This challenge is a new version of the Smartphone Image Denoising Dataset (SIDD) benchmark [3] with a newly collected validation and testing datasets, hence named SIDD+. The original SIDD consisted of thousands of real noisy images with estimated ground-truth, in both raw sensor data (rawRGB) and standard RGB (sRGB) color spaces. Hence,
in this challenge, we provide two tracks for benchmarking image denoisers in both rawRGB and sRGB color spaces. We present more details on both tracks in the next section.

2. The Challenge

This challenge is one of the NTIRE 2020 associated challenges on: deblurring [34], nonhomogeneous dehazing [5], perceptual extreme super-resolution [46], video quality mapping [14], real image denoising [2], real-world super-resolution [32], spectral reconstruction from RGB image [7] and demoireing [45].

The NTIRE 2020 Real Image Denoising Challenge is an extension of the previous NTIRE 2019 challenge [4]. Both challenges aimed to gauge and advance the state-of-the-art in image denoising. The focus of the challenge is on evaluating image denoisers on real, rather than synthetic, noisy images. In the following, we present some details about the new dataset used in this version of the challenge and how the challenge is designed.

2.1. Dataset

The SIDD dataset [3] was used for providing training images for the challenge. The SIDD dataset consists of thousands of real noisy images and their corresponding ground truth, from ten different scenes, captured repeatedly with five different smartphone cameras under different lighting conditions and ISO levels. The ISO levels ranged from 50 to 10,000. The images are provided in both rawRGB and sRGB color spaces.

For validation and testing, we collected a new dataset of 2048 images following a similar procedure to the one used in generating the SIDD validation and testing datasets.

2.2. Challenge Design and Tracks

Tracks We provide two tracks to benchmark the proposed image denoisers based on the two different color spaces: the rawRGB and the sRGB. Images in the rawRGB format represent minimally processed images obtained directly from the camera’s sensor. These images are in a sensor-dependent color space where the R, G, and B values are related to the sensor’s color filter array’s spectral sensitivity to incoming visible light. Images in the sRGB format represent the camera’s rawRGB image that have been processed by the in-camera image processing pipeline to map the sensor-dependent RGB colors to a device-independent color space, namely standard RGB (sRGB). Different camera models apply their own proprietary photo-finishing routines, including several nonlinear color manipulations, to modify the rawRGB values to appear visually appealing (see [20] for more details). We note that the provided sRGB images are not compressed and therefore do not exhibit compression artifacts. Denoising a rawRGB would typically represent a denoising module applied within the in-camera image processing pipeline. Denoising an sRGB image would represent a denoising module applied after the in-camera color manipulation. As found in recent works [4, 3, 36], image denoisers tend to perform better in the rawRGB color space than in the sRGB color space. However, rawRGB images are far less common than sRGB images which are easily saved in common formats, such as JPEG and PNG. Since the SIDD dataset contains both rawRGB and sRGB versions of the same image, we found it feasible to provide a separate track for denoising in each color space. Both tracks follow similar data preparation, evaluation, and competition timeline, as discussed next.

Data preparation The provided training data was the SIDD-Medium dataset that consists of 320 noisy images in both rawRGB and sRGB space with corresponding ground truth and metadata. Each noisy or ground truth image is a 2D array of normalized rawRGB values (mosaiced color filter array) in the range $[0, 1]$ in single-precision floating point format saved as Matlab .mat files. The metadata files contained dictionaries of Tiff tags for the rawRGB images, saved as .mat files.

We collected a new validation and testing datasets following a similar procedure to the one used in SIDD [3], and hence, we named the new dataset SIDD+.

The SIDD+ validation set consists of 1024 noisy image blocks (i.e., croppings) form both rawRGB and sRGB images, each block is $256 \times 256$ pixels. The blocks are taken from 32 images, 32 blocks from each image ($32 \times 32 = 1024$). All image blocks are combined in a single 4D array of shape $[1024, 256, 256]$ where each consecutive 32 images belong to the same image, for example, the first 32 images belong to the first image, and so on. The blocks have the same number format as the training data. Similarly, the SIDD+ testing set consists of 1024 noisy image blocks from a different set of images, but following the same format as the validation set. Image metadata files were also provided for all 64 images from which the validation and testing data were extracted. All newly created validation and testing datasets are publicly available.

We also provided the simulated camera pipeline used to render rawRGB images into sRGB for the SIDD dataset $^1$. The provided pipeline offers a set of processing stages similar to an on-board camera pipeline. Such stages include: black level subtraction, active area cropping, white balance, color space transformation, and global tone mapping.

Evaluation The evaluation is based on the comparison of the restored clean (denoised) images with the ground-truth images. For this we use the standard peak signal-to-noise

$^1$https://github.com/AbdoKamel/simple-camera-pipeline
ratio (PSNR) and, complementary, the structural similarity (SSIM) index [41] as often employed in the literature. Implementations are found in most of the image processing toolboxes. We report the average results over all image blocks provided.

For submitting the results, participants were asked to provide the denoised image blocks in a multidimensional array shaped in the same way as the input data (i.e., [1024, 256, 256]). In addition, participants were asked to provide additional information: the algorithm’s runtime per mega pixel (in seconds); whether the algorithm employs CPU or GPU at runtime; and whether extra metadata is used as inputs to the algorithm.

At the final stage of the challenge, the participants were asked to submit fact sheets to provide information about the teams and to describe their methods.

Timeline The challenge timeline was performed in two stages. The validation stage started on December 20, 2019. The final testing stage started on March 16, 2020. Each participant was allowed a maximum of 20 and 3 submissions during the validation and testing phases, respectively. The challenge ended on March 26, 2020.

3. Challenge Results

From approximately 250 registered participants in each track, 22 teams entered in the final phase and submitted results, codes/executables, and fact sheets. Tables 1 and 2 report the final test results, in terms of peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) index [41], for the rawRGB and sRGB tracks, respectively. The tables show the method ranks based on each measure in subscripts. We present the self-reported runtime and major details provided in the factsheets submitted by participants. Figures 1 and 2 show a 2D visualization of PSNR and SSIM values for all methods in both rawRGB and sRGB tracks, respectively. For combined visualization, both figures are overlaid in Figure 3. The methods are briefly described in section 4 and team members are listed in Appendix A.

Main ideas All of the proposed methods are based on deep learning. Specifically, all methods employ convolutional neural networks (CNNs) based on various architectures. Most of adapted architectures are based on widely-used networks, such as U-Net [37], ResNet [16], and DenseNet [19]. The main ideas included re-structuring existing networks, introducing skip connections, introducing residual connections, and using densely connected components. Other strategies have been used such as feature attention for image denoising [6], atrous spatial pyramid pooling (ASPP) [10], and neural architectural search (NAS) [13].

Most teams used $L_1$ loss as the optimization function while some teams used $L_2$ loss or adopted a mixed
Table 1: Results and rankings of methods submitted to the rawRGB denoising track.

| Team                        | Username        | PSNR  | SSIM  | Runtime (s/Mpixel) | CPU/GPU (at runtime) | Platform | Ensemble | Loss       |
|-----------------------------|-----------------|-------|-------|--------------------|----------------------|----------|----------|------------|
| Baidu Research Vision 1    | zhichongp       | 57.44 | 0.99785 | 5.76               | Tesla V100           | PaddlePaddle, PyTorch | flip/transpose (×8) | L₂         |
| HITVPC&HUAWEI 1             | hitvpc_huawei   | 57.43 | 0.99785 | ?                  | GTX 1080 Ti          | PyTorch  | flip/rotate (×8) | L₂         |
| Eraser 1                    | Songsaris       | 57.34 | 0.99785 | 36.50              | TITAN V              | PyTorch  | flip/transpose (×8) | L₂         |
| Samsung_SLSI_MSL            | Samsung_SLSI_MSL-2 | 57.29 | 0.99790 | 50                 | Tesla V100           | PyTorch  | flip/rotate (×8) | L₂         |
| Tyan 1                      | Tyan            | 57.23 | 0.99786 | 0.38               | GTX 1080 Ti          | TensorFlow | flip/rotate (×8), models (×3) | L₂         |
| NIU-ITIJ                    | Sora            | 57.22 | 0.99784 | 3.5                | Tesla V100           | PyTorch  | models (×8) | L₁         |
| Panda                       | panda_yun       | 57.20 | 0.99788 | 2.72               | GTX 2080 Ti          | TensorFlow | flip/rotate (×8), models (×3 – 5) | L₁         |
| BOE-IOT-AIBD                | eastworld       | 57.19 | 0.99758 | 0.61               | Tesla P100           | TensorFlow | flip/rotate (×8), models (×3 – 5) | L₁         |
| TCL Research Europe 1       | tcl-research-team | 57.11 | 0.99758 | ?                  | RTX 2080 Ti          | TensorFlow | flip/rotate (×8), models (×3 – 5) | L₁         |
| Eraser 3                    | BumjunPark      | 57.03 | 0.99779 | 0.31               | ?                    | PyTorch  | ?         | L₁         |
| EWAH-AIBI 1                 | jaeyeon         | 57.01 | 0.99781 | 55                 | Tesla V100           | PyTorch  | flip/rotate (×8) | L₁         |
| ZHU231                      | quishizai       | 56.72 | 0.99752 | 0.17               | GTX 1080 Ti          | PyTorch  | self ensemble | L₁, DCT    |
| NoahDn                      | matteomaggioni  | 56.47 | 0.99749 | 3.54               | Tesla V100           | TensorFlow | flip/rotate (×8) | L₁         |
| Dahua-jsp                   | -               | 56.20 | 0.99749 | ?                  | GTX 2080             | PyTorch  | ?         | ?          |

Table 2: Results and rankings of methods submitted to the sRGB denoising track.

| Team                        | Username        | PSNR  | SSIM  | Runtime (s/Mpixel) | CPU/GPU (at runtime) | Platform | Ensemble | Loss       |
|-----------------------------|-----------------|-------|-------|--------------------|----------------------|----------|----------|------------|
| Eraser 2                    | Songsaris       | 33.22 | 0.9596 | 103.92             | TITAN V              | PyTorch  | flip/rotate/RGB shuffle (×8) | L₁         |
| Alpha                       | q935970314      | 33.12 | 0.9578 | 6.72               | RTX 2080 Ti          | PyTorch  | flip/rotate (×8) | L₁         |
| HITVPC&HUAWEI 2             | hitvpc_huawei   | 33.04 | 0.9580 | ?                  | GTX 1080 Ti          | PyTorch  | flip/rotate (×8) | L₁         |
| ADDBlock                    | BONG            | 32.80 | 0.9565 | 76.80              | Titan XP             | PyTorch  | flip/rotate (×8), models (×4) | L₁         |
| UIUC_JFP                    | Self-Worker     | 32.05 | 0.9572 | 0.61               | Tesla V100           | PyTorch  | flip/rotate (×8), models (×3) | L₁         |
| Baidu Research Vision 2    | zhichongp       | 32.30 | 0.9532 | 9.28               | Tesla V100           | PaddlePaddle, PyTorch | flip/transpose (×8) | L₁         |
| Rainbow                     | JiKun63         | 32.24 | 0.9410 | 2.41               | RTX 2080 Ti          | PyTorch  | flip/rotate (×8) | L₁         |
| TCL Research Europe 2       | tcl-research-team | 32.23 | 0.9467 | ?                  | RTX 2080 Ti          | TensorFlow | flip/rotate (×8), models (×3 – 5) | L₁         |
| LDResNet                    | SKjim           | 32.09 | 0.9507 | 17.85              | GTX 1080             | PyTorch  | flip/rotate (×8) | L₁         |
| Eraser 4                    | BumjunPark      | 32.06 | 0.9484 | ?                  | ?                    | PyTorch  | ?         | L₁         |
| STAIR                       | dark_1m1less    | 31.67 | 0.9281 | 0                  | 1.86                 | Titan TITAN | ?         | L₁         |
| Cougar AI 2                 | priyakansal     | 31.61 | 0.9383 | 0.23               | GTX 1080             | Keras/Tensorflow | None | MSE/SSIM |
| EWAH-AIBI 1                 | jaeyeon         | 31.38 | 0.9417 | ?                  | Tesla V100           | PyTorch  | flip/rotate (×8) | L₁         |
| NCIA-Lab                    | Han-Soo-Choi    | 31.37 | 0.9269 | 2.92               | TITAN RTX            | PyTorch  | None | MS-SSIM/L₁ |
| Cougar AI 1                 | sabarinathana   | 31.34 | 0.9296 | 0.23               | GTX 1080             | Keras/Tensorflow | None | MSE/SSIM |
| Visionaries                 | rajatguptakgp   | 19.97 | 0.6794 | ?                  | GTX 1050 Ti          | PyTorch  | None | MSE       |

loss between $L_1$, $L_2$, multi-scale structural similarity (MS-SSIM) [42], and/or Laplace gradients.

**Top results** The top methods achieved very close performances, in terms of PSNR and SSIM. In the rawRGB track, the top two methods are 0.01 dB apart in terms of PSNR, whereas in the sRGB track, the top three methods have ~ 0.1 dB difference in terms of PSNR, as shown in Figures 1 and 2. The differences in SSIM values were similarly close. In terms of PSNR, the main performance metric used in the challenge, the best two methods for rawRGB denoising are proposed by teams Baidu Research Vision and HITVPC&HUAWEI, and achieved 57.44 and 57.43 dB PSNR, respectively, while the best method for sRGB denoising is proposed by team Eraser and achieved 33.22 dB PSNR. In terms of SSIM, as a complementary performance metric, the best method for rawRGB denoising is proposed by the team Samsung_SLSI_MSL and achieved a SSIM index of 0.9979, while the best SSIM index for sRGB denoising is achieved by the Eraser team.
Ensembles To boost performance, most of the methods applied different flavors of ensemble techniques. Specifically, most teams used a self-ensemble [40] technique where the results from eight flipped/rotated versions of the same image are averaged together. Some teams applied additional model-ensemble techniques.

Conclusion From the analysis of the presented results, we can conclude that the proposed methods achieve state-of-the-art performance in real image denoising on the SIDD+ benchmark. The top methods proposed by the top ranking teams (i.e., HITVPC&HUAWEI, Baidu Research Vision, Eraser, and Alpha) achieve consistent performance across both color spaces—that is, rawRGB and sRGB (see Figure 3).

4. Methods and Teams

4.1. HITVPC&HUAWEI

Distilling Knowledge from Original Network and Siamese Network for Real Image Denoising The team used distilling knowledge and NAS (Neural Architecture Search technology) to improve the denoising performance. The proposed network is based on both MWCNN [26] and ResNet [16] to propose the mwresnet (multi-level wavelet-resnet). The team used the proposed network to design the Siamese Network by use of NAS technology. Both networks can complement each other to improve denoising performance in distilling knowledge stage. Only the Siamese Network is used in the final denoising stage. The network architecture proposed by the team is shown in Figure 4.

4.2. Baidu Research Vision

Neural Architecture Search (NAS) based Dense Residual Network for Image Denoising The Baidu Research Vision team first proposed a dense residual network that includes multiple types of skip connections to learn features at different resolutions. A new NAS based scheme is further implemented in PaddlePaddle [1] to search for the number of dense residual blocks, the block size and the number of features, respectively. The proposed network achieves good denoising performance in the sRGB track, and the added NAS scheme achieves impressive performance in the rawRGB track. The architectures of the neural network and the distributed SA-NAS proposed by the team are illustrated in Figure 5.

4.3. Eraser

Iterative U-in-U network for image denoising (UinUNet) The team modified the down-up module and connections in the DIDN [44]. Down-sampling and up-sampling layers are inserted between two modules to construct more hierarchical block connections. Several three-level down-up units (DUU) are included in a two-level down-up module (UUB). The UinUNet architecture proposed by the team is shown in Figure 6.
Kernel Attention CNN for Image Denoising (KADN) is inspired by Selective Kernel Networks (SKNet) [22], DeepLab V3 [10], and Densely Connected Hierarchical Network for Image Denoising (DHDN) [35]. The DCR blocks of DHDN are replaced with Kernel Attention (KA) blocks. KA blocks apply the concept of atrous spatial pyramid pooling (ASPP) of DeepLab V3 to apply the idea of SKNet that dynamically selects features from different convolution kernels. The KADN architecture proposed by the team is shown in Figure 7.

4.4. Alpha

Enhanced Asymmetric Convolution Block (EACB) for image restoration tasks [28] Based on ACB[12], the team added two additional diagonal convolutions to further strengthen the kernel skeleton. For image restoration tasks, they removed the batch normalization layers and the bias parameters for better performance, and used the cosine annealing learning rate scheduler [31] to prevent gradient explosions. They used a simple version of RCAN [49] as the backbone. The specific modifications are: (1) Remove all channel attention (CA) modules. Too many CAs lead to an increase in training and testing time, and bring little performance improvement. (2) Remove the upsampling module to keep all the features of the same size. (3) Add a global residual to enhance the stability of the network and make the network reach higher performance in the early stages of training.

4.5. Samsung_SLSI_MSL

Real Image Denoising based on Multi-scale Residual Dense Block and Cascaded U-Net with Block-connection [8] The team used three networks: residual dense network (RDN) [51], multi-scale residual dense network (MRDN), and Cascaded U-Net [37] with residual dense block (RDB) connections (CU-Net). Inspired by Atrous Spatial Pyramid Pooling (ASPP) [10] and RDB, the team designed multi-scale RDB (MRDB) to utilize the multi-scale features within component blocks and built MRDN. Instead of skip-connection, the team designed U-Net with block-connection (U-Net-B) to utilize an additional neural module (i.e., RDB) to connect the encoder and the decoder. They also proposed and used noise permutation for data augmentation to avoid model overfitting. The network architecture of MRDN proposed by the team is shown in Figure 8, and CU-Net is detailed in [8].

4.6. ADDBlock

PolyU-Net (PUNet) for Real Image Denoising The team utilized the idea of Composite Backbone Network (CBNet) architecture [30] used for object detection. They used a U-net architecture [35] as the backbone of their PolyU-Net (PUNet). They constructed recurrent connections between backbones with only addition and without upsampling operation contrast to CBNet to prevent distortion of the original information of backbones. Additionally, contrary to CBNet, a slight performance gain was obtained by sharing weights in the backbone networks. The network architecture proposed by the team is shown in Figure 9.
4.4. UIUC IFP
Using U-Nets as ResNet blocks for Real Image Denoising The team concatenated multiple of the U-Net models proposed in [25, 53]. Each U-Net model is treated as a residual block. The team used eight residual blocks in their model. Model ensemble was used to improve the performance. Specifically, the team trained ten separate models and deployed the top three models that achieves the best results on the validation set. In the testing phase, the team first applied rotating and flipping operations to augment each testing image. Then, a fusion operation is applied to the results obtained from the three high-performance models.

4.5. NJU-IITJ
Learning RAW Image Denoising with Color Correction
The team adapted scaling the Bayer pattern channels based on each channel’s maximum. They used Bayer unification [25] for data augmentation and selected Deep iterative down-up CNN network (DIDN) [44] as their base model for denoising.

4.6. Panda
Pyramid Real Image Denoising Network The team proposed a pyramid real image denoising network (PRIDNet), which contains three stages: (1) noise estimation stage that uses channel attention mechanism to recalibrate the channel importance of input noise; (2) at the multi-scale denoising stage, pyramid pooling is utilized to extract multi-scale features; and (3) the feature fusion stage adopts a kernel selecting operation to adaptively fuse multi-scale features. The PRIDNet architecture proposed by the team is shown in Figure 11.

4.7. Tyan
Parallel U-net for Real Image Denoising The team proposed parallel U-net for considering global and pixel-wise denoising at the same time. Two kinds of U-net were combined in a parallel way: one traditional U-net for global denoising due to its great receptive field, and another U-net with dilated convolutions replacing the pooling operations; to preserve the feature map size. Both U-nets take the same input noisy image separately and their outputs are concatenated and followed by a 1x1 convolution to produce the final clean image. The network architecture proposed by the team is shown in Figure 10.

4.8. UIUC IFP
Using U-Nets as ResNet blocks for Real Image Denoising The team concatenated multiple of the U-Net models proposed in [25, 53]. Each U-Net model is treated as a residual block. The team used eight residual blocks in their model. Model ensemble was used to improve the performance. Specifically, the team trained ten separate models and deployed the top three models that achieves the best results on the validation set. In the testing phase, the team first applied rotating and flipping operations to augment each testing image. Then, a fusion operation is applied to the results obtained from the three high-performance models.

4.9. NJU-IITJ
Learning RAW Image Denoising with Color Correction
The team adapted scaling the Bayer pattern channels based on each channel’s maximum. They used Bayer unification [25] for data augmentation and selected Deep iterative down-up CNN network (DIDN) [44] as their base model for denoising.

4.10. Panda
Pyramid Real Image Denoising Network The team proposed a pyramid real image denoising network (PRIDNet), which contains three stages: (1) noise estimation stage that uses channel attention mechanism to recalibrate the channel importance of input noise; (2) at the multi-scale denoising stage, pyramid pooling is utilized to extract multi-scale features; and (3) the feature fusion stage adopts a kernel selecting operation to adaptively fuse multi-scale features. The PRIDNet architecture proposed by the team is shown in Figure 11.

4.11. Rainbow
Densely Self-Guided Wavelet Network for Image Denoising The team proposed a top-down self-guidance architecture for exploiting image multi-scale information. The low-resolution information is extracted and gradually propagated into the higher resolution sub-networks to guide the feature extraction processes. Instead of pixel-shuffling/unshuffling, the team used the discrete wavelet transform (DWT) and the inverse discrete wavelet transform (IDWT) for upsampling and downsampling, respectively. The used loss was a combination between the L1 and the Laplace gradient losses. The network architecture proposed by the team is shown in Figure 12.

4.12. TCL Research Europe
Neural Architecture Search for image denoising The team proposed an ensemble model consisting of 3 - 5 sub-networks. Two types of sub-networks are proposed: (1) the Superkernel-based Multi Attentional Residual U-Net and (2) the Superkernel SkipInit Residual U-Net. The superkernel method used by the team is based on [38].
4.13. BOE-IOT-AIBD

Raw Image Denoising with Unified Bayer Pattern and Multiscale Strategies The team utilized a pyramid denoising network [52] and Bayer pattern unification techniques [25] where all input noisy rawRGB images are unified to RGGB bayer pattern according to the metadata information. Then inputs are passed into Squeeze-and-Excitation blocks [18] to extract features and assign weights to different channels. Multiscale denoising blocks and selective kernel blocks [22] were applied. The network architecture proposed by the team is shown in Figure 13.

4.14. LDResNet

Mixed Dilated Residual Network for Image Denoising The team designed a deep and wide network by piling up the dilated and residual (DR) blocks equipped with multiple dilated convolutions and skip connections. In addition to the given noisy-clean image pairs, the team utilized extra undesired-clean image pairs as a way to add some noise on the ground truth images of the training data of the SIDD dataset. The network architecture proposed by the team is shown in Figure 14.

4.15. EWHA-AIBI

Denoising with wavelet domain loss The team used an enhanced deep residual network EDSR [23] architecture with global residual skip and input that is decomposed with stationary wavelet transform and used loss in wavelet transform domain. To accelerate the performance of networks, channel attention [43] is added every fourth res block.

4.16. STAIR

Down-Up Scaling Second-Order Attention Network for Real Image Denoising The team proposes a Down-Up scaling RNAN (RNAN-DU) method to deal with real noise that may not be statistically independent. Accordingly, the team used the residual non-local attention network (RNAN) [50] as a backbone of the proposed RNAN-DU method. The down-up sampling blocks are used to suppress the noise, while non-local attention modules are focused on dealing with more severe, non-uniformly distributed real noise. The network architecture proposed by the team is shown in Figure 15.

4.17. Couger AI

Lightweight Residual Dense net for Image Denoising The team proposed a U-net like model with stacked residual dense blocks along with simple convolution/convolution transpose. The input image is first processed by a coordinate convolutional layer [27] aiming to improve the learning of spatial features in the input image. Moreover, the team used the modified dense block to learn the global hierarchical features and then fused these features to the output of decoder in a more holistic way.

4.18. ZJU231

Deep Prior Fusion Network (DPFNet) for Real Image Denoising The team presented DPFNet based on U-net [37]. They utilize the DPF block and the Residual block, which are both modified versions of the standard residual block in ResNet [16], for feature extraction and image reconstruction. Compared with the Residual block, the DPF block introduces an extract $1 \times 1$ convolutional layer to enhance the cross-channel exchange of feature maps during the feature extraction stage.

4.19. NoahDn

Learnable Nonlocal Image Denoising The team proposed a method that explicitly use the nonlocal image prior
within a fully differentiable framework. In particular, the image is processed in a block-wise fashion, and after a shallow feature extraction, self-similar blocks are extracted within a search window and then jointly denoised by exploiting their nonlocal redundancy. The final image estimate is obtained by returning each block in its original position and adaptively aggregating the overcomplete block estimates within the overlapping regions.

4.20. NCIA-Lab

SAID: Symmetric Architecture for Image Denoising
The team proposed a two-branch bi-directional correction model. The first branch was designed to estimate positive values in the final residual layer, while the second branch was used to estimate negative values in the final residual layer. In particular, the team built their model on top of the DHDN architecture [35] by adapting two models of the DHDN architecture.

4.21. Dahua.jsp

Dense Residual Attention Network for Image Denoising
The team optimized the RIDNet [6] with several modifications: using long dense connection to avoid gradients vanishing; and adding depth-wise separable convolution [17] as the transition.

4.22. Visionaries

Image Denoising through Stacked AutoEncoders
The team used a stacked autoencoder to attempt denoising the images. While training, under each epoch, image pairs (original and noisy) were randomly shuffled and a Gaussian noise with mean and standard deviation of the difference between the original and noisy image for all 160 image pairs was added.

Acknowledgements

We thank the NTIRE 2020 sponsors: Huawei, Oppo, Voyage81, MediaTek, DisneyResearch|Studios, and Computer Vision Lab (CVL) ETH Zurich.

A. Teams and Affiliations

NTIRE 2020 Team

Title: NTIRE 2020 Challenge on Real Image Denoising: Dataset, Methods and Results
Members:
Abdelrahman Abdelhamed1 (kamel@eecs.yorku.ca),
Mahmoud Afifi1 (mafifi@eecs.yorku.ca),
Radu Timofte2 (radu.timofte@vision.ee.ethz.ch),
Michael S. Brown1 (mbrown@eecs.yorku.ca)
Affiliations:
1 York University, Canada
2 ETH Zurich, Switzerland

HITVPC&HUAWEI

Title: Distillating Knowledge from Original Network and Siamese Network for Real Image Denoising
Members: Yue Cao1 (hitvpc_huawei@163.com), Zhilu Zhang1, Wangmeng Zuo1, Xiaoling Zhang2, Jiye Liu2, Wendong Chen2, Changyuan Wen2, Meng Liu2, Shuailin Lv2, Yuncao Zhang2
Affiliations: 1 Harbin Institute of Technology, China 2 Huawei, China

Baidu Research Vision

Title: Neural Architecture Search (NAS) based Dense Residual Network for Image Denoising
Members: Zhihong Pan1 (zhihongpan@baidu.com), Baopu Li1, Teng Xi2, Yanwen Fan2, Xiyu Yu2, Gang Zhang2, Jingtu Liu2, Junyu Han2, Errui Ding2
Affiliations: 1 Baidu Research (USA), 2 Department of Computer Vision Technology (VIS), Baidu Incorporation

Eraser

Title: Iterative U-in-U network for image denoising, Kernel Attention CNN for Image Denoising
Members: Songhyun Yu (3069song@naver.com), Bumjun Park, Jechang Jeong
Affiliations: Hanyang University, Seoul, Korea

Alpha

Title: Enhanced Asymmetric Convolution Block (EACB) for image restoration tasks
Members: Shuai Liu1 (18601200232@163.com), Ziyao Zong1, Nan Nan1, Chenghua Li2
Affiliations: 1 North China University of Technology, 2 Institute of Automation, Chinese Academy of Sciences

Samsung_SLSI_MSL

Title: Real Image Denoising based on Multi-scale Residual Dense Block and Cascaded U-Net with Block-connection
Members: Zengli Yang (zengli.y@samsung.com), Long Bao, Shuangquan Wang, Dongwoon Bai, Jungwon Lee
Affiliations: Samsung Semiconductor, Inc.

ADDBlock

Title: PolyU-Net (PUNet) for Real Image Denoising
Members: Youngjung Kim (read12300@add.re.kr), Kyeongha Rho, Changyep Shin, Sungho Kim
Affiliations: Agency for Defense Development

Tyan

Title: Parallel U-Net for Real Image Denoising
Members: Pengliang Tang (tl21200@outlook.com), Yiyun Zhao
Affiliations: Beijing University of Posts and Telecommunications
UIUC IFP
Title: Using U-Nets as ResNet blocks for Real Image Denoising
Members: Yuqian Zhou (zhouyuqian133@gmail.com), Yuchen Fan, Thomas Huang
Affiliations: University of Illinois at Urbana Champaign

NJU-IITJ
Title: Learning RAW Image Denoising with Color Correction
Members: Zhihao Li¹ (lizhihao6@outlook.com), Nisarg A. Shah²
Affiliations: ¹ Nanjing University, Nanjing, China, ² Indian Institute of Technology, Jodhpur, Rajasthan, India

Panda
Title: Pyramid Real Image Denoising Network
Members: Yiyun Zhao (yiyunzhao@bupt.edu.cn), Pengliang Tang
Affiliations: Beijing University of Posts and Telecommunications

Rainbow
Title: Densely Self-guided Wavelet Network for Image Denoising
Members: Wei Liu (liujikun@hit.edu.cn), Qiong Yan, Yuzhi Zhao
Affiliations: SenseTime Research; Harbin Institute of Technology

TCL Research Europe
Title: Neural Architecture Search for image denoising
Members: Marcin Możejko (marcin.mozejko@tcl.com), Tomasz LATkowski, Lukasz Tresczotko, Michal Szfraniuk, Krzysztof Trojanowski
Affiliations: TCL Research Europe

ST Unitas AI Research (STAIR)
Title: Down-Up Scaling Second-Order Attention Network for Real Image Denoising
Members: Magauiya Zhussip (magauiya@stunitas.com), Azamat Khassenov, Jong Hyun Kim, Hwechul Cho
Affiliations: ST Unitas

EWHA-AIBI
Title: Denoising with wavelet domain loss
Members: Wonjin Kim (onejean81@gmail.com), Jaayeon Lee, Jang-Hwan Choi
Affiliations: Ewha Womans University

Couger AI
Title: Lightweight Residual Dense net for Image Denoising, Lightweight Deep Convolutional Model for Image Denoising
Members: Priya Kansal (priya@couger.co.jp), Sabari Nathan (sabari@couger.co.jp)
Affiliations: Couger Inc.

ZJU231
Title: Deep Prior Fusion Network for Real Image Denoising
Members: Zhangyu Ye¹ (qiushizai@zju.edu.cn), Xiwen Lu², Yaqi Wu³, Jiangxin Yang¹, Yanlong Cao¹, Siliang Tang¹, Yanpeng Cao¹
Affiliations: ¹ Zhejiang University, Hangzhou, China, ² Nanjing University of Aeronautics and Astronautics, Nanjing, China, ³ Harbin Institute of Technology Shenzhen, Shenzhen, China

NOAH-Dn
Title: Learnable Nonlocal Image Denoising
Members: Matteo Maggioni (matteo.maggioni@huawei.com), Ioannis Marras, Thomas Tanay, Gregory Slabaugh, Youliang Yan
Affiliations: Huawei Technologies Research and Development (UK) Ltd, Noah’s Ark Lab London

NCIA-Lab
Title: SAID: Symmetric Architecture for Image Denoising
Members: Myungjoo Kang (mkkang@snu.ac.kr), Han-Soo Choi, Sujein Kim, Kyungmin Song
Affiliations: Seoul National University

Dahua isp
Title: Dense Residual Attention Network for Image Denoising
Members: Shusong Xu (13821177832@163.com), Xiaomu Lu, Tingnian Wang, Chunxia Lei, Bin Liu
Affiliations: Dahua Technology
Visionaries

Title: Image Denoising through Stacked AutoEncoders

Members: Rajat Gupta
            (rajatgba2021@email.imical.ac.in), Vineet Kumar

Affiliations: Indian Institute of Technology Kharagpur

References

[1] Parallel distributed deep learning platform. https://github.com/PaddlePaddle/, 2019. 5

[2] A. Abdelhamed, M. Afifi, R. Timofte, M. Brown, et al. Ntire 2020 challenge on real image denoising: Dataset, methods and results. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020. 2

[3] A. Abdelhamed, S. Lin, and M. S. Brown. A high-quality denoising dataset for smartphone cameras. In CVPR, 2018. 1, 2

[4] A. Abdelhamed, R. Timofte, and M. S. Brown. Ntire 2019 challenge on real image denoising: Methods and results. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2019. 1, 2

[5] C. O. Ancuti, C. Ancuti, F.-A. Vasluianu, R. Timofte, et al. Ntire 2020 challenge on nonhomogeneous dehazing. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020. 2

[6] S. Anwar and N. Barnes. Real image denoising with feature attention. In Proceedings of the IEEE International Conference on Computer Vision, pages 3155–3164, 2019. 3, 9

[7] B. Arad, R. Timofte, Y.-T. Lin, G. Finlayson, O. Ben-Shahar, et al. Ntire 2020 challenge on spectral reconstruction from an rgb image. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020. 2

[8] L. Bao, Z. Yang, S. Wang, D. Bai, and J. Lee. Real image denoising based on multi-scale residual dense block and cascaded U-Net with block-connection. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2020. 6

[9] A. Buades, B. Coll, and J. Morel. A non-local algorithm for image denoising. In CVPR, 2005. 1

[10] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European conference on computer vision (ECCV), pages 801–818, 2018. 3, 5, 6

[11] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3D transform-domain collaborative filtering. IEEE TIP, 16(8):2080–2095, 2007. 1

[12] X. Ding, Y. Guo, G. Ding, and J. Han. Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks. In Proceedings of the IEEE International Conference on Computer Vision, pages 1911–1920, 2019. 6

[13] T. Elsken, J. H. Metzen, and F. Hutter. Neural architecture search: A survey. arXiv preprint arXiv:1808.05377, 2018. 3

[14] D. Fuoli, Z. Huang, M. Danelljan, R. Timofte, et al. Ntire 2020 challenge on video quality mapping: Methods and results. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020. 2

[15] S. Gu and R. Timofte. A brief review of image denoising algorithms and beyond. In Springer series on Challenges in Machine Learning, 2019. 1

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR, pages 770–778, 2016. 3, 5, 8

[17] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017. 9

[18] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7132–7141, 2018. 8

[19] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In CVPR, pages 4700–4708, 2017. 3

[20] H. Karaimer and M. S. Brown. A software platform for manipulating the camera imaging pipeline. In ECCV, 2016. 2

[21] D. T. Kuan, A. A. Sawchuk, T. C. Strand, and P. Chavel. Adaptive noise smoothing filter for images with signal-dependent noise. IEEE TPAMI, (2):165–177, 1985. 1

[22] X. Li, W. Wang, X. Hu, and J. Yang. Selective kernel networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 510–519, 2019. 5, 8

[23] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee. Enhanced deep residual networks for single image super-resolution. In CVPR Workshops, pages 136–144, 2017. 8

[24] C. Liu, R. Szeliski, S. B. Kang, C. L. Zitnick, and W. T. Freeman. Automatic estimation and removal of noise from a single image. IEEE TPAMI, 30(2):299–314, 2008. 1

[25] J. Liu, C.-H. Wu, Y. Wang, Q. Xu, Y. Zhou, H. Huang, C. Wang, S. Cai, Y. Ding, H. Fan, and J. Wang. Learning raw image denoising with bayer pattern normalization and bayer preserving augmentation. In CVPR Workshops, 2019. 7

[26] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo. Multi-level wavelet-cnn for image restoration. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 773–782, 2018. 5

[27] R. Liu, J. Lehman, P. Molino, F. P. Such, E. Frank, A. Sergeev, and J. Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. In NeurIPS, pages 9605–9616, 2018. 8

[28] S. Liu, C. Li, N. Nan, Z. Zong, and R. Song. MMDM: Multi-frame and multi-scale for image denoising. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2020. 6

[29] W. Liu, Q. Yan, and Y. Zhao. Densely self-guided wavelet network for image denoising. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2020. 7

[30] Y. Liu, Y. Wang, S. Wang, T. Liang, Q. Zhao, Z. Tang, and H. Ling. Chnet: A novel composite backbone network architecture for object detection. arXiv preprint arXiv:1909.03625, 2019. 6

[31] I. Loshchilov and F. Hutter. SGDR: Stochastic gradient descent with warm restarts. 2016. 6
A. Lugmayr, M. Danelljan, R. Timofte, et al. Ntire 2020 challenge on real-world image super-resolution: Methods and results. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020.

M. Mozejko, T. Latkowski, L. Tresczotko, M. Szafraniuk, and K. Trojanowski. Superkernel Neural Architecture Search for Image Denoising. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2020.

S. Nah, S. Son, R. Timofte, K. M. Lee, et al. Ntire 2020 challenge on image and video deblurring. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020.

B. Park, S. Yu, and J. Jeong. Densely connected hierarchical network for image denoising. In CVPR Workshops, 2019.

T. Plötz and S. Roth. Benchmarking denoising algorithms with real photographs. In CVPR, 2017.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), pages 234–241. Springer, 2015.

D. Stamoulis, R. Ding, D. Wang, D. Lymberopoulos, B. Priyantha, J. Liu, and D. Marculescu. Single-path nas: Designing hardware-efficient convnets in less than 4 hours. arXiv preprint arXiv:1904.02877, 2019.

Y. Tai, J. Yang, X. Liu, and C. Xu. Memnet: A persistent memory network for image restoration. In CVPR, 2017.

R. Timofte, R. Rothe, and L. Van Gool. Seven ways to improve example-based single image super resolution. In CVPR, pages 1865–1873, 2016.

Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE TIP, 13(4):600–612, 2004.

Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multiscale structural similarity for image quality assessment. In Asilomar Conference on Signals, Systems & Computers, volume 2, pages 1398–1402. IEEE, 2003.

S. Woo, J. Park, J.-Y. Lee, and I. So Kweon. CBAM: Convolutional block attention module. In ECCV, pages 3–19, 2018.

S. Yu, B. Park, and J. Jeong. Deep iterative down-up CNN for image denoising. In CVPR Workshops, 2019.

S. Yuan, R. Timofte, A. Leonardis, G. Slabaugh, et al. Ntire 2020 challenge on image demoireing: Methods and results. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020.

K. Zhang, S. Gu, R. Timofte, et al. Ntire 2020 challenge on perceptual extreme super-resolution: Methods and results. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020.

K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. IEEE TIP, 2017.

K. Zhang, W. Zuo, and L. Zhang. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. IEEE Transactions on Image Processing, 27(9):4608–4622, 2018.

Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu. Image super-resolution using very deep residual channel attention networks. In ECCV, pages 286–301, 2018.

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

Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu. Residual dense network for image super-resolution. In CVPR, pages 2472–2481, 2018.

Y. Zhao, Z. Jiang, A. Men, and G. Ju. Pyramid real image denoising network. arXiv preprint arXiv:1908.00273, 2019.

Y. Zhou, D. Ren, N. Emerton, S. Lim, and T. Large. Image restoration for under-display camera. arXiv preprint arXiv:2003.04857, 2020.