Machine Intelligence Research
www.mi-research.net

20(6), December 2023, 822-836
DOI: 10.1007/s11633-023-1466-0

Practical Blind Image Denoising via Swin-Conv-UNet and Data Synthesis

Kai Zhang[1] Yawei Li[1] Jingyun Liang[1] Jiezhang Cao[1] Yulun Zhang[1]
Hao Tang[1] Deng-Ping Fan[1] Radu Timofte[2] Luc Van Gool[1,3]

[1] Computer Vision Lab, ETH Zürich, Zürich 8092, Switzerland

[2] Computer Vision Lab, University of Würzburg, Würzburg 97074, Germany

[3] KU Leuven, Leuven 3000, Belgium

Abstract: While recent years have witnessed a dramatic upsurge of exploiting deep neural networks toward solving image denoising, existing methods mostly rely on simple noise assumptions, such as additive white Gaussian noise (AWGN), JPEG compression noise and camera sensor noise, and a general-purpose blind denoising method for real images remains unsolved. In this paper, we attempt to solve this problem from the perspective of network architecture design and training data synthesis. Specifically, for the network architecture design, we propose a swin-conv block to incorporate the local modeling ability of residual convolutional layer and non-local modeling ability of swin transformer block, and then plug it as the main building block into the widely-used image-to-image translation UNet architecture. For the training data synthesis, we design a practical noise degradation model which takes into consideration different kinds of noise (including Gaussian, Poisson, speckle, JPEG compression, and processed camera sensor noises) and resizing, and also involves a random shuffle strategy and a double degradation strategy. Extensive experiments on AGWN removal and real image denoising demonstrate that the new network architecture design achieves state-of-the-art performance and the new degradation model can help to significantly improve the practicability. We believe our work can provide useful insights into current denoising research. The source code is available at https://github.com/cszn/SCUNet.

Keywords: Blind image denoising, real image denosing data synthesis, Transformer, image signal processing (ISP) pipeline.

Citation: K. Zhang, Y. Li, J. Liang, J. Cao, Y. Zhang, H. Tang, D. P. Fan, R. Timofte, L. V. Gool. Practical blind image denoising via Swin-Conv-UNet and data synthesis. *Machine Intelligence Research*, vol.20, no.6, pp.822–836, 2023. http://doi.org/10.1007/s11633-023-1466-0

1 Introduction

Image denoising, which is the process of recovering a latent clean image $\mathbf{x}$ from its noisy observation $\mathbf{y}$, is perhaps the most fundamental image restoration problem. The reason is at least three-fold. First, it can help to evaluate the effectiveness of different image priors and optimization algorithms[1]. Second, it can be plugged into variable splitting algorithms (e.g., half-quadratic splitting[2] and alternating direction method of multipliers[3]) to solve other problems (e.g., deblurring and super-resolution)[4]. Third, it could be the very first step for other vision tasks.

The degradation model of image denoising can be mathematically formulated by

$$\mathbf{y} = \mathbf{x} + \mathbf{n} \tag{1}$$

where $\mathbf{n}$ is the noise to be removed. Recently, deep neural networks have become the mainstream method for image denoising. To improve deep image denoising performance, researchers mainly focus on two research directions. The first one is to improve the performance under the assumption that $\mathbf{n}$ is additive white Gaussian noise (AWGN). The second one largely focuses on training data or noise modeling. Both directions can contribute to the ultimate goal of improving the practicability for real images.

The common assumptions of $\mathbf{n}$ are AWGN, JPEG compression noise, Poisson noise, and camera sensor noise, among which AWGN is the most widely-used one due to its mathematical convenience. However, it is known that deep image denoising model trained by AWGN performs poorly for most of real images due to noise assumption mismatch[5, 6]. Nevertheless, AWGN removal is fair to test the effectiveness of different network architecture designs. In recent years, various network architecture designs have been proposed. Some representat-

Research Article
Manuscript received on November 18, 2022; accepted on August 7, 2023; published online on September 15, 2023
Recommended by Associate Editor Cheng-Lin Liu
Colored figures are available in the online version at https://link.springer.com/journal/11633

ive ones are DnCNN[7], N3Net[8], NLRN[9], DRUNet[10], and SwinIR[11]. Indeed, network architecture designs can help to capture image prior for better image denoising performance. For example, N3Net[8] and NLRN[9] are specifically designed to capture non-local image prior. Although the PSNR performance on benchmark datasets has been largely improved, e.g., SwinIR[11] outperforms DnCNN[7] by an average PSNR of 0.57dB on Set12 dataset for noise level 25, it is still interesting to raise the first question whether the PSNR performance can be further improved by advanced network architecture design.

In order to facilitate the practicability of deep denoising models, a flurry of work has been devoted to noise modeling. The motivation behind this is to make the noise assumption consistent with the degradation of real images. Plötz and Roth[5] establish a realistic Darmstadt Noise Dataset (DND) with consumer cameras which is composed of different pairs of real noisy and almost noiseless reference images in the RAW domain and sRGB domain. They further show that the model retrained with accurate degradation can significantly outperform that with AWGN on the sRGB DND dataset[8]. By leveraging the physics of digital sensors and the steps of an imaging pipeline, Brooks et al.[6] design a camera sensor noise synthesis method and provide an effective deep raw image denoising model. Although the above attempts have emphasized the importance of degradation models, they mainly focus on camera sensor induced noise removal. Yet, few work has been done on training a deep model for general-purpose blind image denoising. It is interesting to raise the second question of how to improve the training data for blind denoising.

We attempt to answer the above two questions with novel network architecture design and novel training data synthesis. For the network architecture design, motivated by the facts that 1) different methods for image denoising have complementary image prior modeling ability and can be incorporated to boost the performance[12]; 2) DRUNet[10] and SwinIR[11] exploit very different network architecture designs while achieving very promising denoising performance, we propose a swin-conv block to combine the local modeling ability of residual convolutional layer[13] and non-local modeling ability of swin transformer block[14], and then plug it as the main building block into the UNet architecture. In order to test its effectiveness, we evaluate its PNSR performance on benchmark datasets for AWGN removal. Since real image noise could be introduced by other types of noise, such as JPEG compression noise, processed camera sensor noise, and be further affected by resizing, it is too complex to model with a parametric probability distribution. To resolve this problem, we propose a random shuffle of different kinds of noise (including Gaussian, Poisson, speckle, JPEG compression, and processed camera sensor noises) and resizing operations (including the commonly used bilinear and bicubic interpolations) to make a rough approximation of real image noise.

Our contributions are listed as follows:

1) We propose a novel denoising network by plugging novel swin-conv blocks into multiscale UNet to boost the local and non-local modeling ability.

2) We propose a hand-designed noise synthesis model, which can be used to train a general-purpose blind image denoising model.

3) Our blind denoising model trained with the proposed noise synthesis model can significantly improve the practicability for real images.

4) Our work provides a strong baseline for both synthetic Gaussian denoising and practical blind image denoising.

2 Related work

2.1 Deep blind image denoising

Compared to non-blind image denoising, where the noise type and noise level are assumed to be known, blind denoising tackles the case when the noise level of certain noise type is unknown or even the noise type is unknown. During past few years, several attempts have been made to solve the problem of deep blind denoising. Zhang et al.[7] demonstrate that a single deep model can handle Gaussian denoising with various noise levels and can even handle JPEG compression with different quality factors and single image super-resolution with different scale factors. Chen et al.[15] propose to adopt generative adversarial networks (GAN) to generate noise from clean images and then construct the paired training data for subsequent training. Guo et al.[16] propose a convolutional blind denoising network (CBDNet) with a noise estimation subnetwork and then propose to train the model with realistic noise model and real-world noisy-clean image pairs. Krull et al.[17] propose a blind-spot network which can be trained without noisy image pairs or clean target images. Yue et al.[18] propose a variational inference method for blind image denoising which incorporates both noise estimation and image denoising into a unique Bayesian framework. While achieving promising results, the above methods are mainly evaluated on the synthetic Gaussian noise or the processed camera sensor noise such as the DND dataset[5]. Since real noise is far more complex, the above methods can not be readily applied for real applications. It is still unclear how to establish more practical noisy/clean image pairs for training a deep blind model.

2.2 Deep architecture for non-local prior

State-of-the-art model-based image denoising methods mostly exploit non-local self-similarity (NLSS) prior which refers to the fact that a local patch often has many non-local similar patches across the image[19]. Some rep-

resentative methods include BM3D[20], LSSC[21] and WNNM[22]. Inspired by the effectiveness of NLSS prior, some deep learning methods attempt to explicitly model the correlation among non-local patches via the network structure. Sun and Tappen[23] propose a gradient-based discriminative non-local range Markov random field (MRF) method to exploit the advantages of BM3D and non-local means. Inspired by non-local variational methods, Lefkimmiatis[24] designs an unrolled network that can perform non-local processing for better denoising performance. However, the above methods adopt the non-differentiable KNN matching in fixed feature spaces. To resolve this, Plötz and Roth[8] propose a fully end-to-end trainable neural nearest neighbor block to leverage the principle of non-local self-similarity. Liu et al.[9] propose a non-local recurrent network (NLRN) to incorporate non-local operations into a recurrent neural network. Chen et al.[25] propose image processing transformer (IPT) to exploit the non-local modeling of transformer. However, IPT works on fixed image patch size and tends to result in border artifacts. Liang et al.[11] address this issue by adopting the swin transformer as the main building block. It has been shown that transformer-based methods favors more on images with repetitive structures, which verifies the effectiveness of the transformer for non-local modeling ability.

3 Method

Since we focus on learning a deep blind model with paired training data, it is necessary to revisit the Maximum A Posteriori (MAP) inference to have a better understanding. Generally, for the problem of blind image denoising, the estimated clean image $\hat{\mathbf{x}}$ can be obtained by solving the following MAP problem with a certain optimization algorithm,

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} D(\mathbf{x}, \mathbf{y}) + \lambda P(\mathbf{x}) \tag{2}$$

where $D(\mathbf{x}, \mathbf{y})$ is the data fidelity term, $P(\mathbf{x})$ is the prior term and λ is the trade-off parameter.

So far, one can see that the key of solving blind denoising lies in modeling the degradation process of noisy image as well as designing the image prior of clean image.

By treating the deep model as a compact unrolled inference of (2), the deep blind denoising generally aims to solve the following bi-level optimization problem[26, 27]

$$\begin{cases} \min\limits_{W} \sum\limits_{i} \mathcal{L}(\hat{\mathbf{x}}_i(\mathbf{y}_i, W), \mathbf{x}_i) & \text{(3a)} \\ \text{s.t.} \quad \hat{\mathbf{x}}_i = \arg\min_{\mathbf{x}} D(\mathbf{x}, \mathbf{y}_i) + \lambda P(\mathbf{x}) & \text{(3b)} \end{cases}$$

where W denotes the network parameters to be learned, $\{\mathbf{y}_i, \mathbf{x}_i\}$ represents the training noisy-clean image pairs, $\mathcal{L}(\cdot)$ is the loss function. In this sense, the deep blind denoising model should capture the knowledge of degradation process and image prior.

On the other hand, the modeling ability of a deep model generally depends on network architecture, model size (or the number of parameters), and training data. It is clear that the degradation process is implicitly defined by the noisy images from the training data, which indicates the noisy images of the training data is responsible for deep blind denoising model to capture the knowledge of degradation process. In order to improve the image prior modeling ability of deep blind denoising model, one should focus on improving the following three factors, including network architecture, model size and clean images of the training data. While the later two factors are easy to improve, how to improve the network architecture remains further study.

From the above discussions and analyses, we can conclude that the network architecture and the training data are two important factors to improve the performance of deep blind denoising model. In the following, we will separately detail our attempts to improve these two factors.

3.1 Swin-Conv-UNet

Fig. 1 illustrates the network architecture of our proposed Swin-Conv-UNet (SCUNet). The main idea of SCUNet is to integrate the complementary network architecture designs of DRUNet and SwinIR. To be specific, SCUNet plugs novel swin-conv (SC) blocks into a UNet[28] backbone. Following DRUNet[10], the UNet backbone of SCUNet has four scales, each of which has a residual connection between 2×2 strided convolution (SConv) based downscaling and 2×2 transposed convolution (TConv) based upscaling. The number of channels in each layer from the first scale to the fourth scale are 64, 128, 256 and 512, respectively. The main difference between SCUNet and DRUNet is that SCUNet adopts four SC blocks rather than four residual convolution blocks in each scale of the downscaling and upscaling.

As shown in the dashed line of Fig. 1, an SC block fuses swin transformer (SwinT) block[14] and residual convolutional (RConv) block[13, 29] via two 1×1 convolutions, split and concatenation operations, and a residual connection. To be specific, for an input feature tensor X, it is first passed through a 1×1 convolution. Subsequently, it is split evenly into two features map groups X_1 and X_2. We formulate such a process as

$$X_1,\ X_2 = \text{Split}(\text{Conv}1\times1(X)). \tag{4}$$

Then, X_1 and X_2 are separately fed into a SwinT block and a RConv block, giving rise to

$$Y_1,\ Y_2 = \text{SwinT}(X_1),\ \text{RConv}(X_2). \tag{5}$$

Finally, Y_1 and Y_2 are concatenated as the input of a 1×1 convolution which has a residual connection with

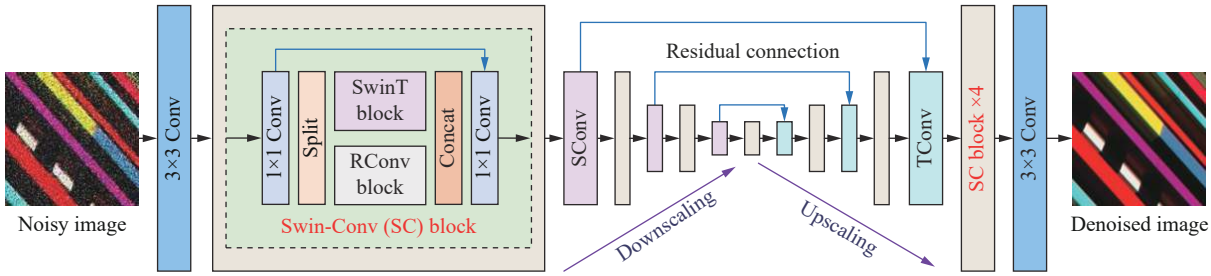


Fig. 1 The architecture of the proposed Swin-Conv-UNet (SCUNet) denoising network. SCUNet exploits the swin-conv (SC) block as the main building block of a UNet backbone. In each SC block, the input is first passed through a 1×1 convolution, and subsequently is split evenly into two feature map groups, each of which is then fed into a swin transformer (SwinT) block and residual 3×3 convolutional (RConv) block, respectively; after that, the outputs of SwinT block and RConv block are concatenated and then passed through a 1×1 convolution to produce the residual of the input. "SConv" and "TConv" denote 2×2 strided convolution with stride 2 and 2×2 transposed convolution with stride 2, respectively.

the input X. As such, the final output of SC block is given by

$$Z = \text{Conv}1\times1(\text{Concat}(Y_1,\ Y_2)) + X. \tag{6}$$

It is worth pointing out that our proposed SCUNet enjoys several merits due to some novel module designs. First, the SC block fuses the local modeling ability of RConv block and non-local modeling ability of SwinT block. Second, the local and non-local modeling ability of SCUNet is further enhanced via the multiscale UNet. Third, the 1×1 convolution can effectively and efficiently facilitate information fusion between SwinT block and RConv block. Fourth, the split and concatenation operations can act as the group convolution with two groups to reduce the computation complexity and the number of parameters. We note that SCUNet essentially functions as a hybrid convolutional neural networks (CNNs)-Transformer network and there also exist several other works that integrate CNNs and Transformer for effective network architecture design. Li et al.[30] propose a fabric-like hybrid CNN-transformer search space, in which each layer can flexibly choose CNN building blocks and transformer building blocks. Yuan et al.[31] introduce a novel Convolution-enhanced Image Transformer (CeiT) that combines the advantages of CNNs in extracting low-level features and strengthening locality with the benefits of Transformers in establishing long-range dependencies. Guo et al.[32] propose a new Transformer-based hybrid network that leverages the strengths of Transformers to capture long-range dependencies and the capabilities of CNNs to extract local information.

It is also worth pointing out the difference between our proposed SCUNet and two recently works including Uformer[33] and Swin-Unet[34]. First, the motivation is different. Our SCUNet is motivated by the fact that state-of-the-art denoising methods DRUNet[10] and SwinIR[11] exploit very different network architecture designs, and thus SCUNet aims to integrate the complementary network architecture designs of DRUNet and SwinIR. By contrast, Uformer and Swin-UNet aim to combine the transformer variants and UNet. Second, the main building blocks are different. Our SCUNet adopts a novel swin-conv block which integrates the local modeling ability of residual convolutional layer[13] and non-local modeling ability of swin transformer block[14] via 1×1 convolution, split and concatenation operations. By contrast, Uformer adopts a new transformer block by combining depth-wise convolution layers[35], while Swin-UNet utilizes the swin transformer block as the main building block.

3.2 Training data synthesis

Instead of establishing a large variety of real noisy/clean image pairs, which is laborious and challenging, we attempt to synthesize realistic noisy/clean image pairs. The main idea is to add different kinds of noise and also include the resizing, as well as incorporating a double degradation strategy and a random shuffle strategy which we will detail next.

Gaussian noise. Additive white Gaussian noise (AWGN) is the most widely-used assumption for denoising. While it can perfectly model read noise of an image sensor, it usually does not match the real noise and would deteriorate the practicability of trained deep denoising models. Nevertheless, it has been shown that deep denoising model (e.g., FFDNet[36]) trained with AWGN can remove non-Gaussian noise by setting a larger Gaussian noise level, with the sacrifice of smoothing the textures and edges. Instead of using the simplified AWGN, we adopt the 3D generalized zero-mean Gaussian noise model[37] with 3×3 covariance matrix to model the noise correlation between R, G and B channels. One of the underlying reasons is that the color image demosaicing step in camera ISP pipeline can correlate the noise across channels. Depending on the cross-channel dependencies, such a generalized Gaussian model has two extreme cases, including the widely-used additive white Gaussian color noise and grayscale Gaussian noise. We uniformly sample their noise levels from $\{2/255, 3/255, \cdots, 50/255\}$. Since in-camera denoising algorithms generally remove the color noise for better perceptual quality, grayscale Gaussian

noise would be a good choice to model the remaining noise. For this reason, we sample the two extreme cases and general case with probabilities 0.4, 0.4 and 0.2, respectively.

Poisson noise. Poisson noise generally refers to the photon shot noise which originates from the discrete nature of electric charge. It occurs severely in low-light conditions, such as night time photography, medical imaging, optical microscopy imaging and astronomy imaging[38]. Different from Gaussian noise which is signal-independent, Poisson noise is signal-dependent. Traditional model-based methods mostly apply the variable-stabilizing transformation (VST) to transfer the noise into approximate signal-independent one, and then tackle the problem with Gaussian denoising methods. However, such methods need to know the noise type beforehand, which is generally impossible for real images. Hence, removing the Poisson noise directly via the deep model would be a good choice. To sample different noise levels for an image, we first multiply the clean image by 10^{α}, where α is uniformly chosen from $[2, 4]$, and then divide back by 10^{α} after adding the signal-dependent Poisson noise. Our Poisson noise can be mathematically modeled as

$$\mathbf{n} \sim \mathcal{P}(10^{\alpha} \times \mathbf{x})/10^{\alpha} - \mathbf{x}; \quad \alpha \sim \mathcal{U}(2, 4). \tag{7}$$

Following the Gaussian noise, we also consider grayscale Poisson noise by converting the clean color image into grayscale image. After that, we add the same grayscale noise to each channel of the given image.

Speckle noise. Speckle noise is multiplicative noise which usually appears in coherent imaging systems, such as synthetic aperture radar (SAR) imaging and medical ultrasonic imaging[39, 40]. It can be modeled by the multiplication between latent clean image and Gaussian noise. We thus simply modify the above Gaussian noise synthesis strategy by multiplying the clean image to generate speckle noise.

JPEG compression noise. Image compression can help to reduce the storage and bandwidth requirements for digital images. Among various image compression standards and formats, JPEG has been the most widely-used one since it is simple and allows for fast encoding and decoding. However, it will reduce the image quality by introducing severe 8×8 blocking artifacts with the increase of compression degree. Such a trade-off is controlled by the quality factor which ranges from 0 to 100. Due to its pervasiveness in Internet and social media usage, we add this kind of noise by uniformly sampling the quality factor from $[20, 95]$.

Processed camera sensor noise. The noise in output RGB image of modern digital cameras is mainly caused by passing the read and shot noise in raw sensor data through an image signal processing (ISP) pipeline. Hence, the distribution of the processed camera sensor noise varies with the read and shot noise model and ISP model. Inspired by [6], we synthesize this kind of noise by generating raw image from clean image via the reverse ISP pipeline, and then processing the noisy raw image via the forward ISP pipeline after adding read and shot noise to raw image. For the read and shot noise model, we exactly borrow the one proposed in [6]. For the ISP model, we adopt the one proposed in [41] which consists of demosaicing, exposure compensation, white balance, camera to XYZ (D50) color space conversion, XYZ (D50) to linear RGB color space conversion, tone mapping and gamma correction. It is still worth pointing out the following details about the adopted ISP model. First, the orders of gamma correction and tone mapping, and and the tone mapping curves are different from these in [6]. Our ISP adopts gamma correction as the final step, whereas [6] uses tone mapping as the final step. While it has been known that the tone mapping curves for different cameras are usually different, Brooks et al.[6] use a fixed tone curve. By contrast, our ISP selects the best fitted tone curves from [42] for each camera based on the error between reconstructed output and the camera ground-truth RGB output. Second, the forward-reverse tone mapping may cause color shift issue with respect to the original image due to the irreversibility, we resolve this by also applying the reverse-forward tone mapping for the clean image.

Resizing. Image resizing is one of the basic digital image editing tools. It can be used to fit the image into a certain space on a screen or be used to downscale the image to reduce the storage size. While resizing does not introduce noise to the clean images, it would affect the noise distribution of the noisy images. For example, upscaling would lead AWGN to be spatially correlated while downscaling would change processed camera sensor noise to be less signal-dependent. To model such resizing induced noise, we uniformly apply the widely-used bilinear and bicubic resizing operations. The scaling factor is uniformly chosen from $[0.5, 2]$. Especially noteworthy is that we apply the same resizing on both noisy image and its clean counterpart since resizing will change the spatial resolution of latent clean image of the noisy image. Hence, it is essentially different from the super-resolution degradation proposed in [41, 43].

In practice, real images might be resized or JPEG compressed several times[44], and JPEG compression might be performed before or after resizing. Inspired by this, our final degradation sequence employs a double degradation strategy and a random shuffle strategy. By doing this, the degradation space is expected to be largely expanded, which can facilitate the generalization ability of the trained deep blind model. Specifically, we perform the above noises and resizing twice. We add Gaussian noise and JPEG compression noise with the probabilities of 1. For the resizing and other noise addition, we set the probabilities to 0.5. Before applying the degradation sequence to a clean image, we first perform a random

shuffle on the degradations. To prevent out-of-range values after each degradation process, we always make sure the image is clipped into the range of 0–1. Due to the introduction of resizing, a large high quality image should be used for the paired training data synthesis. Fig. 2 provides a schematic illustration of the proposed training data synthesis pipeline.

4 Discussions

4.1 Our denoising data synthesis pipeline VS. super-resolution data synthesis pipeline[41, 43]

Our training data synthesis pipeline differs from the ones proposed in [41, 43] in at least three main aspects. First, the applications are different. Our pipeline is used for deep blind image denoising, whereas the ones proposed in [41, 43] are designed for deep blind super-resolution. Second, our pipeline also performs the resizing on the high-quality image to produce the corresponding clean image of the noisy images, whereas the degradation models in [41, 43] do not perform such a procedure. The reason is that denoising does not necessitate removing image blur and enlarging the resolution, which is different from super-resolution. Third, our pipeline adopts more kinds of noise, such as speckle noise. Fig. 3 shows some synthesized noisy/clean patch pairs via our proposed training data synthesis pipeline. It can be seen that our data synthesis pipeline can produce very realistic noisy images. It is worth noting that the noisy/clean patch pairs are from the same high quality image with size 544×544. Since we also perform the resizing operations for clean image patches, we can observe some blurriness from some of the clean image patches.

4.2 Practical blind denoising VS. blind Gaussian denoising and blind camera sensor noise removal for DND and SIDD

Our practical blind denoising is much more difficult than blind Gaussian denoising and blind camera sensor noise removal for DND and SIDD, and is the "true" blind image denoising for practical application. It is widely-

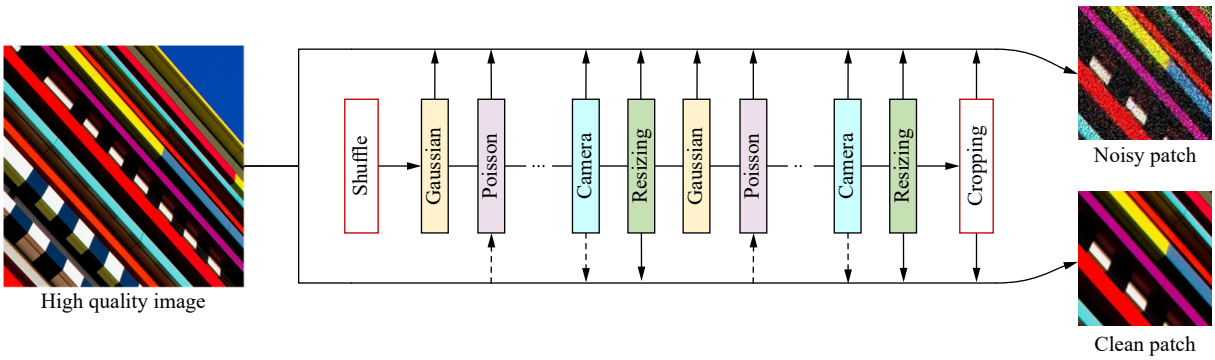


Fig. 2 Schematic illustration of the proposed paired training patches synthesis pipeline. For a high quality image, a randomly shuffled degradation sequence is performed to produce a noisy image. Meanwhile, the resizing and reverse-forward tone mapping are performed to produce a corresponding clean image. Paired noisy/clean training patches are then cropped for training deep blind denoising model. Note that, since Poisson noise is signal-dependent, the dashed arrow for "Poisson" means the clean image is used to generate the Poisson noise. To tackle the color shift issue, the dashed arrow for "Camera Sensor" means the reverse-forward tone mapping is performed on the clean image.

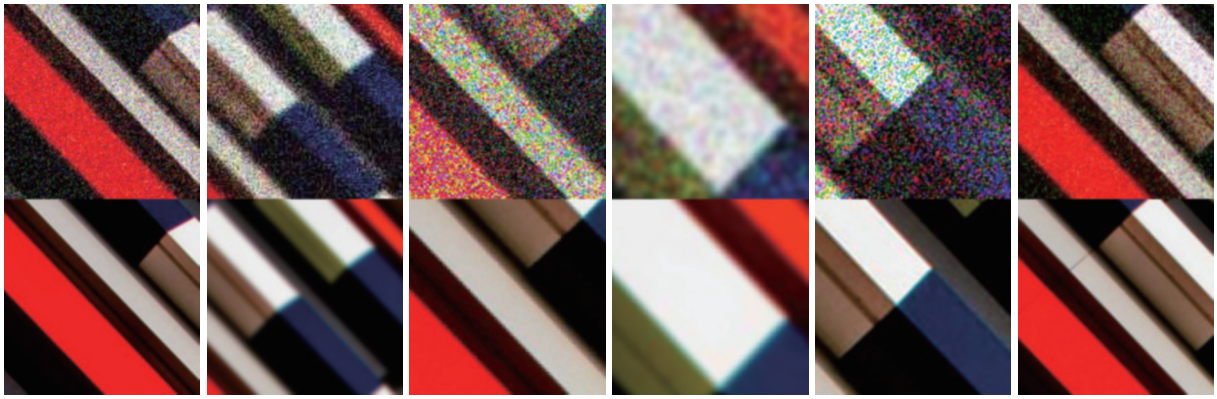


Fig. 3 Synthesized noisy/clean patch pairs via our proposed training data synthesis pipeline. The size of the high quality image patch is 544×544. The size of the noisy/clean patches is 128×128.

known that the deep model trained for blind Gaussian denoising does not perform well for real images due to noise assumption mismatch. For this reason, DND and SIDD are established by capturing noisy and clean images pairs from different cameras. Although these two datasets help researchers shift to real image denoising, however, they focus on camera sensor noise which also deviates significantly from the noise from the Internet in our daily life. Moreover, as shown in Fig. 4, the state-of-the-art DeamNet for these datasets even has a worse result than Noise Clinic for noisy images from a different kind of camera, which indicates that deep models trained for these two datasets do not generalize well for unseen noise, thus having very limited applications. In contrast, our model is trained on a much more complex degradation model whose the degradation space is large enough to cover a large variety of different noise combinations, and thus can significantly improve the practicability. As far as we know, the existing "true" blind denoising is the work entitled "The noise clinic: a blind image denoising algorithm". Our model can significantly outperform Noise Clinic and is the first deep model that can be readily applied for real applications.

5 Experiments

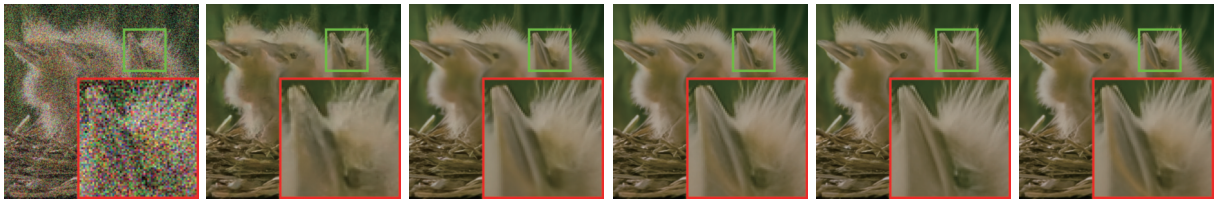
As discussed in Section 3, the network architecture and the training data are two important factors to improve the performance of deep blind denoising model. For the sake of fairness, we first evaluate our SCUNet on synthetic Gaussian denoising. We then evaluate our training data synthesis pipeline with our SCUNet on practical blind image denoising.

5.1 Synthetic Gaussian denoising

Implementation details.

For the high quality image dataset, we use the same training dataset consisting of Waterloo Exploration Database[45], DIV2K[46], and Flick2K[29] for training. The settings of SwinT and Rconv blocks are the same to those in SwinIR and DRUNet, respectively. Following the common setting, we generate the noisy image by adding AWGN with a certain noise level and separately learn a denoising model for each noise level. The parameters are optimized by minimizing the L1 loss with Adam optimizer[47]. The learning rate starts from 1×10^{-4} and and decays by a factor of 0.5 every 200 000 iterations and finally ends with 3.125×10^{-6}. The patch size and batch size are set to 128×128 and 24, respectively. We first train the model with noise level 25 and then finetune the model for other noise levels. All experiments are implemented by PyTorch 1.7.1. It takes about three days to train a denoising model on four NVIDIA RTX 2 080 Ti GPUs.

Grayscale Gaussian denoising. Table 1 reports the PSNR results of different methods on the widely-used Set12[7], BSD68[48, 49], Urban100[50] datasets for noise



PSNR(dB)/SSIM 28.68/0.779 4 29.06/0.797 4 29.28/0.810 3 29.28/0.809 5 29.37/0.813 5

(a) Noisy (b) DnCNN (c) RNAN (d) DRUNet (e) SwinIR (f) SCUNet

Fig. 4 Color image denoising results of different methods on image "163 085" from CBSD68 dataset. The noisy image is corrupted by AWGN with noise level 50.

Table 1 Average PSNR(dB) results of different methods for grayscale image denoising with noise levels 15, 25 and 50 on the widely-used Set12, BSD68 and Urban100 datasets. The best and second best results are highlighted in red and blue colors, respectively.

Datasets	Noise level	DnCNN	FFDNet	N^3Net	NLRN	RNAN	FOCNet	DAGL	DRUNet	SwinIR	Restormer	**SCUNet**
Set12	15	32.86	32.75	–	33.16	–	33.07	33.28	33.25	33.36	33.42	33.43
	25	30.44	30.43	30.55	30.80	–	30.73	30.93	30.94	31.01	31.08	31.09
	50	27.18	27.32	27.43	27.64	27.70	27.68	27.81	27.90	27.91	28.00	28.04
BSD68	15	31.73	31.63	–	31.88	–	31.83	31.93	31.91	31.97	31.96	31.99
	25	29.23	29.19	29.30	29.41	–	29.38	29.46	29.48	29.50	29.52	29.55
	50	26.23	26.29	26.39	26.47	26.48	26.50	26.51	26.59	26.58	26.62	26.67
Urban100	15	32.64	32.40	–	33.45	–	33.15	33.79	33.44	33.70	33.79	33.88
	25	29.95	29.90	30.19	30.94	–	30.64	31.39	31.11	31.30	31.46	31.58
	50	26.23	26.50	26.26	27.49	27.65	27.40	27.97	27.96	27.98	28.29	28.56

levels 15, 25 and 50. The compared methods include DnCNN[7], FFDNet[36], N^3Net[8], NLRN[9], RNAN[51], FOCNet[52], DAGL[53], DRUNet[10], SwinIR[11] and Restormer[54]. We note that N^3Net, NLRN, RNAN and SwinIR explicitly employ non-local module design in order to capture non-local image prior for better denoising performance. It can be seen that our SCUNet achieves significantly better PSNR results than other methods for all the noise levels on the three datasets. Specifically, SCUNet surpasses DnCNN and FFDNet by an average PSNR of 0.6 dB on Set12, 0.3 dB on BSD68 and 1.6 dB on Urban100, and produces a substantial PSNR gain over state-of-the-art DAGL, DRUNet, SwinIR and Restormer. Since images from Urban100 are rich in repetitive structures, such a large improvement on Urban100 over BSD68 indicates that SCUNet has an advantage of modeling non-local image prior.

To qualitatively evaluate the proposed SCUNet, we provide the denoising results of different methods on classical image "Barbara" from Set12 dataset with noise level 50 in Fig. 5. Note that we also include the traditional model-based methods BM3D[20] and WNNM[22] for comparison since they are based on non-local priors. We have the following observations. First, WNNM produces much better visual results than some of the deep denoising methods such as DnCNN, FFDNet, RNAN and FOCNet. Second, while DAGL, DRUNet and SwinIR have better PSNR results than WNNM, they fail to recover some of the repetitive lines which indicates they still have limits in non-local prior modeling. Third, our SCUNet produces more visually pleasant results than others which further verifies the effectiveness of SCUNet for modeling image non-locality.

Color Gaussian denoising. Table 2 reports the color image denoising results of different methods on CBSD68[48, 49], Kodak24[55], McMaster[56] and Urban100[50] datasets. The compared methods include DnCNN, FFDNet, DSNet[57], BRDNet[58], RNAN, RDN[59], IPT, DRUNet, SwinIR and Restormer. As one can see, our SCUNet produces the best overall performance. Specifically, SCUNet surpasses DnCNN, FFDNet and DSNet by an average PSNR of 0.5 dB on CBSD68, 0.7 dB on Kodak24, 1.1 dB on McMaster and 1.6 dB on Urban100. Interestingly, while SCUNet has a similar PSNR gain over DRUNet for different noise levels, it achieves a larger PSNR gain than SwinIR with the increase of noise level. The possible reason is that SwinIR tends to lack the ability to model the long range dependency for heavy noise removal.

Results. Fig. 6 provides the visual results of different blind denoising methods for real image denoising. The testing images includes "Palace" from [60], "Building" form Internet, and "Stars" from [7]. The compared methods include CBDNet[16], DeamNet[61] and Noise Clinic[60]. We also report the results of no-reference image quality assessment (IQA) metrics NIQE[62], NRQM[63] and PIQE[64].

Fig. 4 provides the visual results of different methods on image "163 085" from CBSD68 with noise level 50. It can be seen that SwinIR fails to recover the yellow structure along the beak of the bird while DnCNN, RNAN and DRUNet introduce some smoothness. By contrast, SCUNet recovers fine structures and preserves image sharpness.

FLOPs, runtime and #Params. We report FLOPs, runtime and #Params comparisons among DRUNet, SwinIR and SCUNet in Table 3. We can see that our SCUNet achieves the lowest FLOPs due to the combination of UNet and SC block. Since SwinIR does not use any downscaling operations, it suffers from high FLOPs and long runtime. In comparison, SCUNet achieves the best trade-off between FLOPs, runtime and

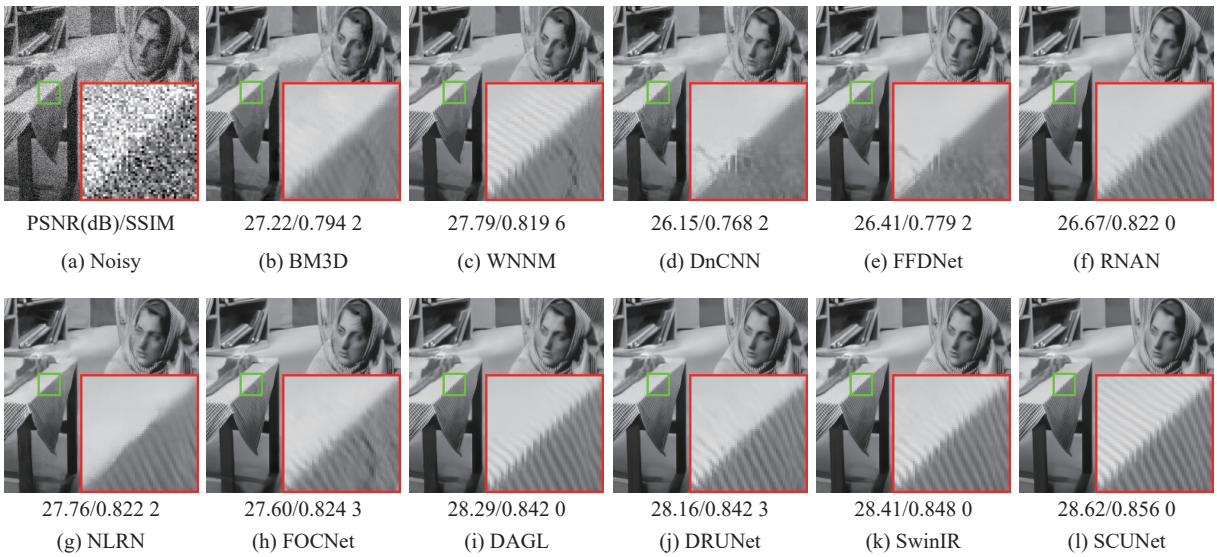


Fig. 5 Grayscale image denoising results of different methods "Barbara" from Set12 dataset. The noisy image is corrupted by AWGN with noise level 50.

Table 2 Average PSNR(dB) results of different methods for color image denoising with noise levels 15, 25 and 50 on the CBSD68, Kodak24, McMaster and Urban100 datasets. The best and second best results are highlighted in red and blue colors, respectively

Dataset	Noise Level	DnCNN	FFDNet	DSNet	BRDNet	RNAN	RDN	IPT	DRUNet	SwinIR	Restormer	**SCUNet**
CBSD68	15	33.90	33.87	33.91	34.10	–	–	–	34.30	34.42	34.40	34.40
	25	31.24	31.21	31.28	31.43	–	–	–	31.69	31.78	31.79	31.79
	50	27.95	27.96	28.05	28.16	28.27	28.31	28.39	28.51	28.56	28.60	28.61
Kodak24	15	34.60	34.63	34.63	34.88	–	–	–	35.31	35.34	35.47	35.34
	25	32.14	32.13	32.16	32.41	–	–	–	32.89	32.89	33.04	32.92
	50	28.95	28.98	29.05	29.22	29.58	29.66	29.64	29.86	29.79	30.01	29.87
McMaster	15	33.45	34.66	34.67	35.08	–	–	–	35.40	35.61	35.61	35.60
	25	31.52	32.35	32.40	32.75	–	–	–	33.14	33.20	33.34	33.34
	50	28.62	29.18	29.28	29.52	29.72	–	29.98	30.08	30.22	30.30	30.29
Urban100	15	32.98	33.83	–	34.42	–	–	–	34.81	35.13	35.13	35.18
	25	30.81	31.40	–	31.99	–	–	–	32.60	32.90	32.96	33.03
	50	27.59	28.05	–	28.56	29.08	29.38	29.71	29.61	29.82	30.02	30.14

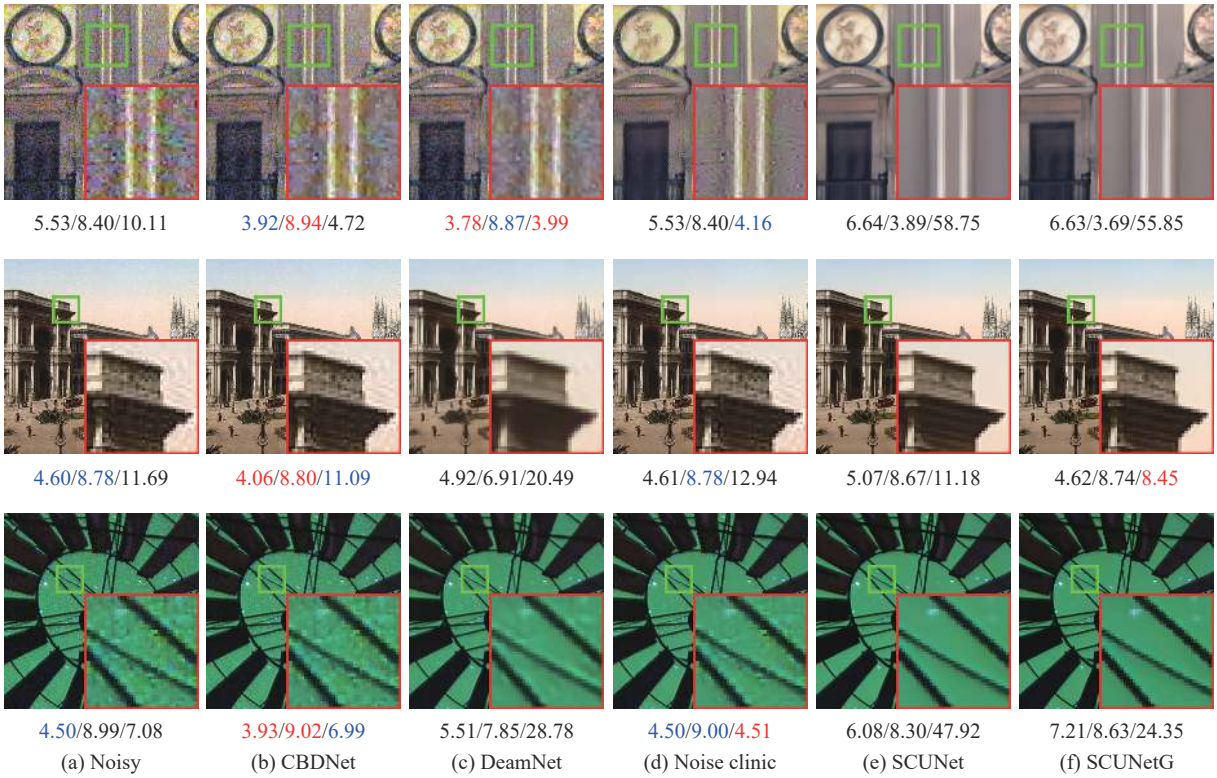


Fig. 6 Visual results and no-reference image quality assessment metrics (NIQE↓/NRQM↑/PIQE↓) results of different methods for real image denoising. The images in each row from top to bottom are "Palace", "Building" and "Stars", respectively.

#Params. Note that the runtime of SCUNet can be reduced by efficient implementation.

5.2 Practical blind image denoising

Implementation details.

We use the same training implementations as in synthetic Gaussian denoising except the following: First, each high quality image is first cropped into a size of 544×544 before processing it into a pair of noisy/clean images. Second, the learning rate is fixed to 1×10^{-4} as it tends to enhance the generalization ability. Third, we also train a perceptual quality-oriented blind model, namely SCUNetG, by minimizing a weighted combination of L1 loss, VGG perceptual loss on five convolution layers and UNetGAN loss[43] with weights 1, 1 and 1, respectively.

Results.

From Fig. 6, we can observe that our SCUNet and

Table 3 FLOPs, runtime and #Params comparisons on images of size 256 × 256 on a PC with an Nvidia Titan Xp GPU

Metrics	DRUNet	SwinIR	**SCUNet**
FLOPs	143.5 G	787.9 G	67.1 G
Runtime	0.020 s	0.525 s	0.072 s
#Params	32.64 M	11.49 M	17.94 M

SCUNetG achieve the best visual results for noise removal and details preserving. For example, both CBDNet and DeamNet fail to removal the processed camera sensor noise for "Palace" while ours can remove such low-frequency noise and recover the underlying edges. However, our results do not show promising no-reference IQA results. As pointed out in [41], such a phenomenon further indicates that no-reference IQA methods should update with degradation types. Fig. 7 provides more blind denoising results of our SCUNet and SCUNetG on real images from RNI15 dataset[36]. Note that we do not know the ground-truth noise type and noise levels of these real images. For example, the "Boy", "Dog" and "Glass" are likely to be corrupted by processed camera sensor noise with unknown camera type and the "Flowers" is corrupted by Gaussian-like noise. Surprisingly, our models effectively handle these images, which could be due to the fact that they have been trained to manage a wide range of degradation scenarios created by various types of noise, resizing, and a random shuffle strategy. According to the above results, we can conclude that the proposed training data synthesis pipeline is suitable for training deep blind denoising model for real applications.

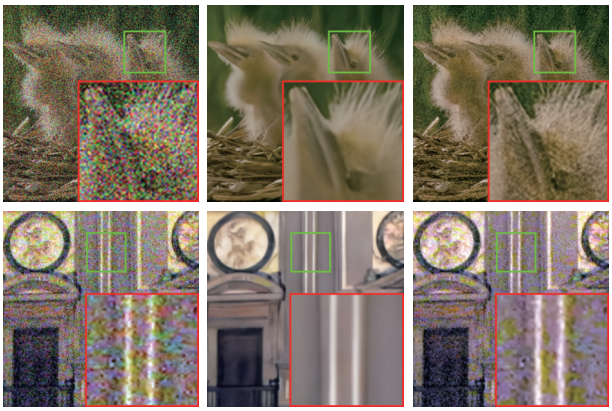
Impact of the resizing for data synthesis. Since one of the main differences between our proposed noisy image synthesis from others is that we adopt resizing to diversify the noise distribution, it is interesting to investigate the performance of the trained model without using resizing in the training data synthesis. Fig. 8 provides the visual comparisons on two upsampled noisy image by bicubic resizing with a scale factor of 2. The first noisy image is corrupted by Gaussian noise with noise level 50 while the second one is corrupted by unknown processed camera sensor noise. It can be seen that the trained model without using resizing in the training data synthesis fails to completely remove the noise. Thus, we can conclude that the resizing can help to improve the generalization ability.

6 Conclusions

In this paper, we focus on the problem of practical blind image denoising. Inspired by the Maximum A Posteriori (MAP) inference which indicates prior modeling and degradation modeling are essential for the success of deep blind denoising, we propose a new network architecture for better prior modeling and a novel data synthesis method for better practical usage. Specifically, we design a new swin-conv block which incorporates the local modeling ability of residual convolution block and non-local modeling ability of swin transformer block, and plug it as the main building block into a UNet to further enhance the local and non-local modeling ability. Moreover, we design a data synthesis pipeline which considers different kinds of noise and also involves a random shuffle strategy and a double degradation strategy. Extensive experimental results demonstrated the effectiveness of the new architecture design for Gaussian denoising and practicability of the trained deep blind model for real noisy images.

(a) Boy (b) Dog (c) Movie (d) Window (e) Glass (f) Flowers

Fig. 7 More blind denoising results of our SCUNet and SCUNetG on real images from RNI15 dataset. From top row to bottom row: noisy images, results of SCUNet, results of SCUNetG. Please zoom in for better view.



(a) (b) (c)

Fig. 8 Comparison between SCUNet and its variant without using resizing in the training data synthesis for denoising a resized noisy image: (a) Upsampled noisy image by bicubic resizing with a scale factor of 2; (b) Denoising result of SCUNet; (c) Denoising results of SCUNet with using resizing in the training data synthesis.

Acknowledgements

This work was partly supported by the ETH Zürich Fund (OK), and by Huawei grants. Open Access funding provided by Swiss Federal Institute of Technology Zurich.

Declarations of conflict of interest

The authors declared that they have no conflicts of interest to this work.

Open Access

References

[1] P. Chatterjee, P. Milanfar. Is denoising dead? *IEEE Transactions on Image processing*, vol. 19, no. 4, pp. 895–911, 2009. DOI: 10.1109/TIP.2009.2037087.

[2] M. V. Afonso, J. M. Bioucas-Dias, M. A. T. Figueiredo. Fast image recovery using variable splitting and constrained optimization. *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2345–2356, 2010. DOI: 10.1109/TIP.2010.2047910.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011. DOI: 10.1561/2200000016.

[4] U. S. Kamilov, H. Mansour, B. Wohlberg. A plug-and-play priors approach for solving nonlinear imaging inverse problems. *IEEE Signal Processing Letters*, vol. 24, no. 12, pp. 1872–1876, 2017. DOI: 10.1109/LSP.2017.2763583.

[5] T. Plötz, S. Roth. Benchmarking denoising algorithms with real photographs. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Honolulu, USA, pp. 2750–2759, 2017. DOI: 10.1109/CVPR.2017.294.

[6] T. Brooks, B. Mildenhall, T. F. Xue, J. W. Chen, D. Sharlet, J. T. Barron. Unprocessing images for learned raw denoising. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 11028–11037, 2019. DOI: 10.1109/CVPR.2019.01129.

[7] K. Zhang, W. M. Zuo, Y. J. Chen, D. Y. Meng, L. Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image processing*, vol. 26, no. 7, pp. 3142–3155, 2017. DOI: 10.1109/TIP.2017.2662206.

[8] T. Plötz, S. Roth. Neural nearest neighbors networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal Canada, pp. 1095–1106, 2018.

[9] D. Liu, B. H. Wen, Y. C. Fan, C. C. Loy, T. S. Huang. Non-local recurrent network for image restoration. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal, Canada, pp. 1680–1689, 2018.

[10] K. Zhang, Y. W. Li, W. M. Zuo, L. Zhang, L. Van Gool, R. Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6360–6376, 2022. DOI: 10.1109/TPAMI.2021.3088914.

[11] J. Y. Liang, J. Z. Cao, G. L. Sun, K. Zhang, L. Van Gool, R. Timofte. SwinIR: Image restoration using swin transformer. In *Proceedings of IEEE/CVF International Conference on Computer Vision Workshops*, IEEE, Montreal, Canada, pp. 1833–1844, 2021. DOI: 10.1109/ICCVW54120.2021.00210.

[12] H. C. Burger, C. Schuler, S. Harmeling. Learning how to combine internal and external denoising methods. In *Proceedings of the 35th German Conference on Pattern Recognition*, Springer, Saarbrücken, Germany, pp. 121–130, 2013. DOI: 10.1007/978-3-642-40602-7_13.

[13] K. M. He, X. Y. Zhang, S. Q. Ren, J. Sun. Identity mappings in deep residual networks. In *Proceedings of the 14th European Conference on Computer Vision*, Springer, Amsterdam, The Netherlands, pp. 630–645, 2016. DOI: 10.1007/978-3-319-46493-0_38.

[14] Z. Liu, Y. T. Lin, Y. Cao, H. Hu, Y. X. Wei, Z. Zhang, S. Lin, B. N. Guo. Swin transformer: Hierarchical vision

transformer using shifted windows. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 9992–10002, 2021. DOI: 10.1109/ICCV48922.2021.00986.

[15] J. W. Chen, J. W. Chen, H. Y. Chao, M. Yang. Image blind denoising with generative adversarial network based noise modeling. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 3155–3164, 2018. DOI: 10.1109/CVPR.2018.00333.

[16] S. Guo, Z. F. Yan, K. Zhang, W. M. Zuo, L. Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 1712–1722, 2019. DOI: 10.1109/CVPR.2019.00181.

[17] A. Krull, T. O. Buchholz, F. Jug. Noise2Void-learning denoising from single noisy images. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 2124–2132, 2019. DOI: 10.1109/CVPR.2019.00223.

[18] Z. S. Yue, H. W. Yong, Q. Zhao, L. Zhang, D. Y. Meng. Variational denoising network: Toward blind noise modeling and removal. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Vancouver, Canada, pp. 151, 2019.

[19] A. Buades, B. Coll, J. M. Morel. A non-local algorithm for image denoising. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, San Diego, USA, pp. 60–65, 2005. DOI: 10.1109/CVPR.2005.38.

[20] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007. DOI: 10.1109/TIP.2007.901238.

[21] J. Mairal, F. Bach, J. Ponce, G. Sapiro, A. Zisserman. Non-local sparse models for image restoration. In *Proceedings of the IEEE 12th International Conference on Computer Vision*, IEEE, Kyoto, Japan, pp. 2272–2279, 2009. DOI: 10.1109/ICCV.2009.5459452.

[22] S. H. Gu, L. Zhang, W. M Zuo, X. C. Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, USA, pp. 2862–2869, 2014. DOI: 10.1109/CVPR.2014.366.

[23] J. Sun, M. F. Tappen. Learning non-local range markov random field for image restoration. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Colorado Springs, USA, pp. 2745–2752, 2011. DOI: 10.1109/CVPR.2011.5995520.

[24] S. Lefkimmiatis. Non-local color image denoising with convolutional neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Honolulu, USA, pp. 5882–5891, 2017. DOI: 10.1109/CVPR.2017.623.

[25] H. T. Chen, Y. H. Wang, T. Y. Guo, C. Xu, Y. P. Deng, Z. H. Liu, S. W. Ma, C. J. Xu, C. Xu, W. Gao. Pre-trained image processing transformer. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Nashville, USA, pp. 12294–12305, 2021. DOI: 10.1109/CVPR46437.2021.01212.

[26] U. Schmidt, S. Roth. Shrinkage fields for effective image restoration. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, USA, pp. 2774–2781, 2014. DOI: 10.1109/CVPR.2014.349.

[27] Y. J. Chen, T. Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1256–1272, 2017. DOI: 10.1109/TPAMI.2016.2596743.

[28] O. Ronneberger, P. Fischer, T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the 18th International Conference on Medical Image computing and Computer-assisted Intervention*, Springer, Munich, Germany, pp. 234–241, 2015. DOI: 10.1007/978-3-319-24574-4_28.

[29] B. Lim, S. Son, H. Kim, S. Nah, K. Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, Honolulu, USA, pp. 1132–1140, 2017. DOI: 10.1109/CVPRW.2017.151.

[30] C. L Li, T. Tang, G. R. Wang, J. F. Peng, B. Wang, X. D. Liang, X. J. Chang. Bossnas: Exploring hybrid CNN-transformers with block-wisely self-supervised neural architecture search. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 12261–12271, 2021. DOI: 10.1109/ICCV48922.2021.01206.

[31] K. Yuan, S. P. Guo, Z. W. Liu, A. J. Zhou, F. W. Yu, W. Wu. Incorporating convolution designs into visual transformers. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 559–568, 2021. DOI: 10.1109/ICCV48922.2021.00062.

[32] J. Y. Guo, K. Han, H. Wu, Y. H. Tang, X. H. Chen, Y. H. Wang, C. Xu. CMT: Convolutional neural networks meet vision transformers. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, New Orleans, USA, pp. 12165–12175, 2022. DOI: 10.1109/CVPR52688.2022.01186.

[33] Z. D. Wang, X. D. Cun, J. M. Bao, W. G. Zhou, J. Z. Liu, H. Q. Li. Uformer: A general U-shaped transformer for image restoration. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, New Orleans, USA, pp. 17662–17672, 2022. DOI: 10.1109/CVPR52688.2022.01716.

[34] H. Cao, Y. Y. Wang, J. Chen, D. S. Jiang, X. P. Zhang, Q. Tian, M. N. Wang. Swin-unet: Unet-like pure transformer for medical image segmentation. In *Proceedings of the 17th European Conference on Computer Vision*, Springer, Tel Aviv, Israel, pp. 205–218, 2023. DOI: 10.1007/978-3-031-25066-8_9.

[35] Y. W. Li, K. Zhang, J. Z. Cao, R. Timofte, L. Van Gool. LocalViT: Bringing locality to vision transformers, [Online], Available: https://arxiv.org/abs/2104.05707, 2021.

[36] K. Zhang, W. M. Zuo, L. Zhang. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, 2018. DOI: 10.1109/TIP.2018.2839891.

[37] S. Nam, Y. Hwang, Y. Matsushita, S. J. Kim. A holistic approach to cross-channel image noise modeling and its application to image denoising. In *Proceedings of IEEE*

Conference on Computer Vision and Pattern Recognition, IEEE, Las Vegas, USA, pp. 1683–1691, 2016. DOI: 10.1109/CVPR.2016.186.

[38] S. W. Hasinoff. Photon, poisson noise. *Computer Vision: A Reference Guide*, K. Ikeuchi, Ed., New York, USA: Springer, pp. 608–610, 2014. DOI: 10.1007/978-0-387-31439-6_482.

[39] M. Tur, K. C. Chin, J. W. Goodman. When is speckle noise multiplicative? *Applied Optics*, vol. 21, no. 7, pp. 1157–1159, 1982. DOI: 10.1364/AO.21.001157.

[40] R. Racine, G. A. H. Walker, D. Nadeau, R. Doyon, C. Marois. Speckle noise and the detection of faint companions. *Publications of the Astronomical Society of the Pacific*, vol. 111, no. 759, pp. 587–594, 1999. DOI: 10.1086/316367.

[41] K. Zhang, J. Y. Liang, L. Van Gool, R. Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 4771–4780, 2021. DOI: 10.1109/ICCV48922.2021.00475.

[42] M. D. Grossberg, S. K. Nayar. What is the space of camera response functions? In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Madison, USA, pp. II–602, 2003. DOI: 10.1109/CVPR.2003.1211522.

[43] X. T. Wang, L. B. Xie, C. Dong, Y. Shan. Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of IEEE/CVF International Conference on Computer Vision Workshops*, IEEE, Montreal, Canada, pp. 1905–1914, 2021. DOI: 10.1109/ICCVW54120.2021.00217.

[44] J. X. Jiang, K. Zhang, R. Timofte. Towards flexible blind JPEG artifacts removal. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 4977–4986, 2021. DOI: 10.1109/ICCV48922.2021.00495.

[45] K. D. Ma, Z. F. Duanmu, Q. B. Wu, Z. Wang, H. W. Yong, H. L. Li, L. Zhang. Waterloo exploration database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 1004–1016, 2017. DOI: 10.1109/TIP.2016.2631888.

[46] E. Agustsson, R. Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, Honolulu, USA, pp. 1122–1131, 2017. DOI: 10.1109/CVPRW.2017.150.

[47] D. P. Kingma, J. Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, USA, 2015.

[48] D. Martin, C. Fowlkes, D. Tal, J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the 8th IEEE International Conference on Computer Vision*, IEEE, Vancouver, Canada, pp. 416–423, 2001. DOI: 10.1109/ICCV.2001.937655.

[49] S. Roth, M. J. Black. Fields of experts. *International Journal of Computer Vision*, vol. 82, no. 2, pp. 205–229, 2009. DOI: 10.1007/s11263-008-0197-6.

[50] J. B. Huang, A. Singh, N. Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Boston, USA, pp. 5197–5206, 2015. DOI: 10.1109/CVPR.2015.7299156.

[51] Y. L. Zhang, K. P. Li, K. Li, B. N. Zhong, Y. Fu. Residual non-local attention networks for image restoration. In *Proceedings of the 7th International Conference on Learning Representations*, New Orleans, USA, 2019.

[52] X. X. Jia, S. Y. Liu, X. C. Feng, L. Zhang. FOCNet: A fractional optimal control network for image denoising. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 6047–6056, 2019. DOI: 10.1109/CVPR.2019.00621.

[53] C. Mou, J. Zhang, Z. Y. Wu. Dynamic attentive graph learning for image restoration. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 4308–4317, 2021. DOI: 10.1109/ICCV48922.2021.00429.

[54] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. khan, M. H. Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, New Orleans, USA, pp. 5718–5729, 2022. DOI: 10.1109/CVPR52688.2022.00564.

[55] R. Franzen. Kodak lossless true color image suite, [Online], Available: https://r0k.us/graphics/kodak, November 15, 1999.

[56] L. Zhang, X. L. Wu, A. Buades, X. Li. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic Imaging*, vol. 20, no. 2, Article number 023016, 2011. DOI: 10.1117/1.3600632.

[57] Y. L. Peng, L. Zhang, S. G. Liu, X. J. Wu, Y. Zhang, X. L. Wang. Dilated residual networks with symmetric skip connection for image denoising. *Neurocomputing*, vol. 345, pp. 67–76, 2019. DOI: 10.1016/j.neucom.2018.12.075.

[58] C. W. Tian, Y. Xu, W. M. Zuo. Image denoising using deep CNN with batch renormalization. *Neural Networks*, vol. 121, pp. 461–473, 2020. DOI: 10.1016/j.neunet.2019.08.022.

[59] Y. L. Zhang, Y. P. Tian, Y. Kong, B. N. Zhong, Y. Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 7, pp. 2480–2495, 2021. DOI: 10.1109/TPAMI.2020.2968521.

[60] M. Lebrun, M. Colom, J. M. Morel. The noise clinic: A blind image denoising algorithm. *Image Processing on Line*, vol. 5, pp. 1–54, 2015. DOI: 10.5201/ipol.2015.125.

[61] C. Ren, X. H. He, C. C. Wang, Z. B. Zhao. Adaptive consistency prior based deep network for image denoising. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Nashville, USA, pp. 8592–8602, 2021. DOI: 10.1109/CVPR46437.2021.00849.

[62] A. Mittal, R. Soundararajan, A. C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 209–212, 2013. DOI: 10.1109/LSP.2012.2227726.

[63] C. Ma, C. Y. Yang, X. K. Yang, M. H. Yang. Learning a no-reference quality metric for single-image super-resolution. *Computer Vision and Image Understanding*, vol. 158, pp. 1–16, 2017. DOI: 10.1016/j.cviu.2016.12.009.

[64] N. Venkatanath, D. Praneeth, M. C. Bh, S. S. Channappayya, S. S. Medasani. Blind image quality evaluation using perception based features. In *Proceedings of the 21th First National Conference on communications*, IEEE, Mumbai, India, pp. 1–6, 2015. DOI: 10.1109/NCC.2015.7084843.

Kai Zhang received the Ph. D. degree in computer science from School of Computer Science and Technology, Harbin Institute of Technology, China in 2019. He is currently a postdoctoral researcher at Computer Vision Lab, ETH Zürich, Switzerland. He was a research assistant from July, 2015 to July, 2017 and from July, 2018 to April, 2019 in the Department of Computing, The Hong Kong Polytechnic University, China. He has proposed several popular image restoration methods, such as DnCNN, SRMD, USRNet, DPIR, SwinIR, and BSRGAN, which have received more than 10 000 citations.

His research interest mainly focuses on developing flexible, effective, efficient, and interpretable deep learning techniques for inverse problems in low-level computer vision.

E-mail: cskaizhang@gmail.com (Corresponding author)
ORCID iD: 0000-0002-6319-3722

Yawei Li received the B. Eng. and B. Econ. degrees in computer science in 2014 and the M. Eng. degree in computer science in 2017 both from the University of Electronic Science and Technology of China. He is a Ph. D. degree candidate at Computer Vision Laboratory, ETH Zürich, Switzerland, supervised by Prof. Luc Van Gool and Prof. Radu Timofte.

His research interests include image restoration and enhancement, model acceleration, and network compression.

E-mail: yawei.li@vision.ee.ethz.ch
ORCID iD: 0000-0002-8948-7892

Jingyun Liang received the B. Sc. degree in information system engineering from the National University of Defense Technology, China in 2016, where he also received the M. Sc. degree in control science and engineering in 2019. He is a Ph.D. degree candidate at the Computer Vision Lab, Department of Information Technology and Electrical Engineering, ETH Zürich, Switzeland since September 2019.

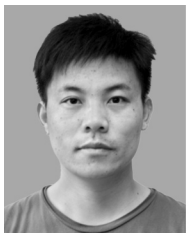
His research interest is low-level vision, especially image and video restoration, such as image/video super-resolution, deblurring and denoising.

E-mail: jinliang@vison.ee.ethz.ch

Jiezhang Cao received the B. Sc. degree in statistics from Guangdong University of Technology, China in 2017, and the M. Eng. degree in School of Software Engineering, South China University of Technology, China in 2020. He is a Ph. D. degree candidate at Department of Information Technology and Electrical Engineering, ETH Zürich, Switzerland since November 2020.

His research interests include machine learning and image/video restoration.

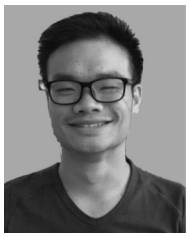
E-mail: jiezhang.cao@vision.ee.ethz.ch
ORCID iD: 0000-0002-0050-4714



Yulun Zhang received the B. Eng. degree in intelligence science and technology from the School of Electronic Engineering, Xidian University, China in 2013, and the M. Eng. degree in control engineering from the Department of Automation, Tsinghua University, China in 2017. He received the Ph. D. degree in computer engineering from Department of ECE, Northeastern University, USA in 2021. He is a postdoctoral researcher at Computer Vision Lab, ETH Zürich, Switzerland. He also worked as a research fellow in Harvard University, USA. He was the recipitents of the Best Student Paper Award at VCIP in 2015 and the Best Paper Award at ICCV RLQ Workshop in 2019.

His research interests include image/video restoration and synthesis, biomedical image analysis, model compression, and computational imaging.

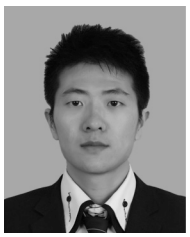
E-mail: yulun100@gmail.com
ORCID iD: 0000-0002-2288-5079



Hao Tang received the M. Eng. degree in computer science from School of Electronics and Computer Engineering, Peking University, China in 2016, and the Ph. D. degree in computer science from the Multimedia and Human Understanding Group, University of Trento, Italy in 2021. He is currently a postdoctoral with Computer Vision Lab, ETH Zürich, Switzerland. He was a visiting scholar in Department of Engineering Science, University of Oxford, UK.

His research interests include deep learning, machine learning, and their applications to computer vision.

E-mail: hao.tang@vision.ee.ethz.ch



Deng-Ping Fan received the Ph. D. degree from Nankai University, China in 2019. He joined the Inception Institute of Artificial Intelligence (IIAI), UAE in 2019. He has published approximately 50 top journal and conference papers such as TPAMI, CVPR, ICCV, ECCV, etc. He won the Best Paper Finalist Award at IEEE CVPR 2019, and the Best Paper Award Nominee at IEEE CVPR 2020. He was recognized as the CVPR 2019 outstanding reviewer with a special mention award, the CVPR 2020 outstanding reviewer, the ECCV 2020 high-quality reviewer, and the CVPR 2021 outstanding reviewer. He served as a program committee board (PCB) member of IJCAI 2022–2024, a senior program committee (SPC) member of IJCAI 2021, a committee member of China Society of Image and Graphics (CSIG), area chair in NeurIPS 2021 Datasets and Benchmarks Track, area chair in MICCAI2020 Workshop (OMIA7), Associate Editor of *Machine Intelligence Research*.

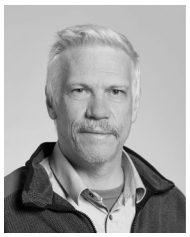
His research interests include computer vision, deep learning, and visual attention, especially the human vision on co-salient object detection, RGB salient object detection, RGB-D salient object detection, and video salient object detection.

E-mail: dengpfan@gmail.com
ORCID iD: 0000-0002-5245-7518

Radu Timofte received the Ph. D. degree in electrical engineering from the KU Leuven, Belgium in 2013. From 2013 to 2016, he was postdoc in the Computer Vision Lab, ETH Zürich, Switzerland. He is currently group leader and lecturer in the same lab. He is an editorial board member of top journals such as *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *Elsevier Neurocomputing*, *Elsevier Computer Vision and Image Understanding*, *SIAM Journal on Imaging Sciences and served(s)* as an area chair for top conferences such as CVPR 2021, IJCAI 2021, ECCV 2020, ACCV 2020, ICCV 2019. His work received several awards. He is a co-founder of Merantix and a co-organizer of NTIRE, CLIC, AIM, and PIRM events.

His research interests include deep learning, implicit models, compression, tracking, restoration and enhancement.

E-mail: radu.timofte@uni-wuerzburg.de
ORCID iD: 0000-0002-1478-0402



Luc Van Gool received the B. Eng. degree in electromechanical engineering from the Katholieke Universiteit Leuven in 1981. Currently, he is a professor at the Katholieke Universiteit Leuven in Belgium and the ETH Zürich, Switzerland. He leads computer vision research at both places, and also teaches at both. He has been a program committee member of several major computer vision conferences. He received several Best Paper awards, won a David Marr Prize and a Koenderink Award, and was nominated Distinguished Researcher by the IEEE Computer Science committee. He is a co-founder of 10 spin-off companies.

His research interests include 3D reconstruction and modelling, object recognition, tracking, and gesture analysis, and the combination of those.

E-mail: vangool@vision.ee.ethz.ch
ORCID iD: 0000-0002-3445-5711