Deep Learning for Image Super-resolution: A Survey

Zhihao Wang, Jian Chen, Steven C.H. Hoi, Fellow, IEEE

Abstract—Image Super-Resolution (SR) is an important class of image processing techniques to enhance the resolution of images and videos in computer vision. Recent years have witnessed remarkable progress of image super-resolution using deep learning techniques. This article aims to provide a comprehensive survey on recent advances of image super-resolution using deep learning approaches. In general, we can roughly group the existing studies of SR techniques into three major categories: supervised SR, unsupervised SR, and domain-specific SR. In addition, we also cover some other important issues, such as publicly available benchmark datasets and performance evaluation metrics. Finally, we conclude this survey by highlighting several future directions and open issues which should be further addressed by the community in the future.

Index Terms—Image Super-resolution, Deep Learning, Convolutional Neural Networks (CNN), Generative Adversarial Nets (GAN)

1 Introduction

Image super-resolution (SR), which refers to the process of recovering high-resolution (HR) images from low-resolution (LR) images, is an important class of image processing techniques in computer vision and image processing. It enjoys a wide range of real-world applications, such as medical imaging [1], [2], [3], surveillance and security [4], [5]), amongst others. Other than improving image perceptual quality, it also helps to improve other computer vision tasks [6], [7], [8], [9]. In general, this problem is very challenging and inherently ill-posed since there are always multiple HR images corresponding to a single LR image. In literature, a variety of classical SR methods have been proposed, including prediction-based methods [10], [11], [12], edge-based methods [13], [14], statistical methods [15], [16], patch-based methods [13], [17], [18], [19] and sparse representation methods [20], [21], etc.

With the rapid development of deep learning techniques in recent years, deep learning based SR models have been actively explored and often achieve the state-of-the-art performance on various benchmarks of SR. A variety of deep learning methods have been applied to tackle SR tasks, ranging from the early Convolutional Neural Networks (CNN) based method (e.g., SRCNN [22], [23]) to recent promising SR approaches using Generative Adversarial Nets (GAN) [24] (e.g., SRGAN [25]). In general, the family of SR algorithms using deep learning techniques differ from each other in the following major aspects: different types of network architectures [26], [27], [28], different types of loss functions [8], [29], [30], different types of learning principles and strategies [8], [31], [32], etc.

In this paper, we give a comprehensive overview of recent advances in image super-resolution with deep learning. Although there are some existing SR surveys in literature, our work differs in that we are focused in deep learning based SR techniques, while most of the earlier works [33], [34], [35], [36] aim at surveying traditional SR algorithms or some studies mainly concentrate on providing quantitative evaluations based on full-reference metrics or human visual perception [37], [38]. Unlike the existing surveys, this survey takes a unique deep learning based perspective to review the recent advances of SR techniques in a systematic and comprehensive manner.

The main contributions of this survey are three-fold:

1) We give a comprehensive review of image super-resolution techniques based on deep learning, including problem settings, benchmark datasets, performance metrics, a family of SR methods with deep learning, domain-specific SR applications, etc.
2) We provide a systematic overview of recent advances of deep learning based SR techniques in a hierarchical and structural manner, and summarize the advantages and limitations of each component for an effective SR solution.
3) We discuss the challenges and open issues, and identify the new trends and future directions to provide an insightful guidance for the community.

In the following sections, we will cover various aspects of recent advances in image super-resolution with deep learning. Fig. 1 shows the taxonomy of image SR to be covered in this survey in a hierarchically-structured way. Section 2 gives the problem definition and reviews the mainstream datasets and evaluation metrics. Section 3 analyzes main components of supervised SR modularly. Section 4 gives a brief introduction to unsupervised SR methods. Section 5 introduces some popular domain-specific SR applications, and Section 6 discusses future directions and open issues.

- *Corresponding author: Steven C.H. Hoi is currently with Salesforce Research Asia, and also a faculty member (on leave) of the School of Information Systems, Singapore Management University, Singapore. Email: shoi@salesforce.com or chhoi@smu.edu.sg.*
- *Z. Wang is with the South China University of Technology, China. E-mail: ptkin@outlook.com. This work was done when he was a visiting student with Dr Hoi's group at the School of Information Systems, Singapore Management University, Singapore.*
- *J. Chen is with the South China University of Technology, China. E-mail: ellachen@scut.edu.cn.*

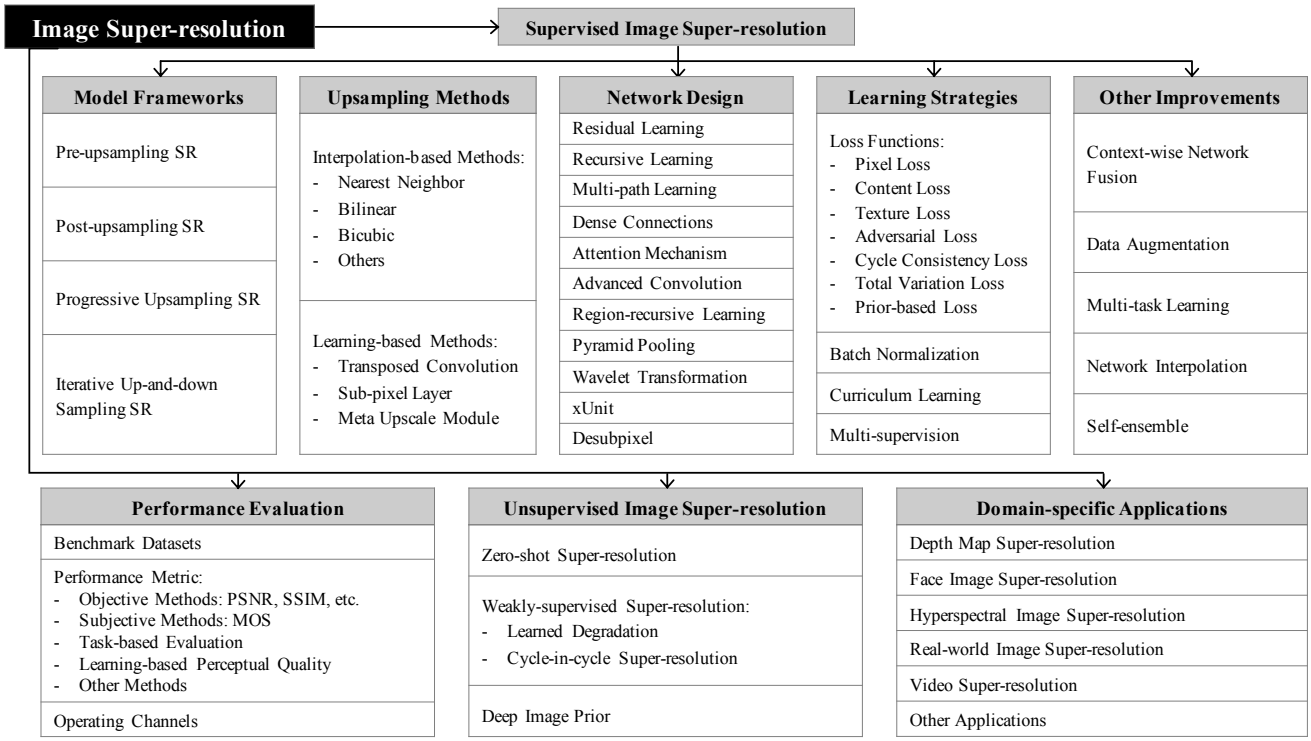


Fig. 1. Hierarchically-structured taxonomy of this survey.

2 Problem Setting and Terminology

2.1 Problem Definitions

Image super-resolution aims at recovering the corresponding HR images from the LR images. Generally, the LR image I_x is modeled as the output of the following degradation:

$$I_x = \mathcal{D}(I_y; \delta), \tag{1}$$

where $\mathcal{D}$ denotes a degradation mapping function, I_y is the corresponding HR image and δ is the parameters of the degradation process (e.g., the scaling factor or noise). Generally, the degradation process (i.e., $\mathcal{D}$ and δ) is unknown and only LR images are provided. In this case, also known as blind SR, researchers are required to recover an HR approximation $\hat{I_y}$ of the ground truth HR image I_y from the LR image I_x, following:

$$\hat{I_y} = \mathcal{F}(I_x; \theta), \tag{2}$$

where $\mathcal{F}$ is the super-resolution model and θ denotes the parameters of $\mathcal{F}$.

Although the degradation process is unknown and can be affected by various factors (e.g., compression artifacts, anisotropic degradations, sensor noise and speckle noise), researchers are trying to model the degradation mapping. Most works directly model the degradation as a single downsampling operation, as follows:

$$\mathcal{D}(I_y; \delta) = (I_y) \downarrow_s, \{s\} \subset \delta, \tag{3}$$

where $\downarrow_s$ is a downsampling operation with the scaling factor s. As a matter of fact, most datasets for generic SR are built based on this pattern, and the most commonly used downsampling operation is bicubic interpolation with anti-aliasing. However, there are other works [39] modelling the degradation as a combination of several operations:

$$\mathcal{D}(I_y; \delta) = (I_y \otimes \kappa) \downarrow_s + n_\varsigma, \{\kappa, s, \varsigma\} \subset \delta, \tag{4}$$

where $I_y \otimes \kappa$ represents the convolution between a blur kernel κ and the HR image I_y, and n_ς is some additive white Gaussian noise with standard deviation ς. Compared to the naive definition of Eq. 3, the combinative degradation pattern of Eq. 4 is closer to real-world cases and has been shown to be more beneficial for SR [39].

To this end, the objective of SR is as follows:

$$\hat{\theta} = \arg\min_{\theta} \mathcal{L}(\hat{I_y}, I_y) + \lambda\Phi(\theta), \tag{5}$$

where $\mathcal{L}(\hat{I_y}, I_y)$ represents the loss function between the generated HR image $\hat{I_y}$ and the ground truth image I_y, $\Phi(\theta)$ is the regularization term and λ is the tradeoff parameter. Although the most popular loss function for SR is pixel-wise mean squared error (i.e., pixel loss), more powerful models tend to use a combination of multiple loss functions, which will be covered in Sec. 3.4.1.

2.2 Datasets for Super-resolution

Today there are a variety of datasets available for image super-resolution, which greatly differ in image amounts, quality, resolution, and diversity, etc. Some of them provide LR-HR image pairs, while others only provide HR images, in which case the LR images are typically obtained by *imresize* function with default settings in MATLAB (i.e., bicubic interpolation with anti-aliasing). In Table 1 we list a number of image datasets commonly used by the SR community,

TABLE 1
List of public image datasets for super-resolution benchmarks.

Dataset	Amount	Avg. Resolution	Avg. Pixels	Format	Category Keywords
BSDS300 [40]	300	$(435, 367)$	$154,401$	JPG	animal, building, food, landscape, people, plant, etc.
BSDS500 [41]	500	$(432, 370)$	$154,401$	JPG	animal, building, food, landscape, people, plant, etc.
DIV2K [42]	1000	$(1972, 1437)$	$2,793,250$	PNG	environment, flora, fauna, handmade object, people, scenery, etc.
General-100 [43]	100	$(435, 381)$	$181,108$	BMP	animal, daily necessity, food, people, plant, texture, etc.
L20 [44]	20	$(3843, 2870)$	$11,577,492$	PNG	animal, building, landscape, people, plant, etc.
Manga109 [45]	109	$(826, 1169)$	$966,011$	PNG	manga volume
OutdoorScene [46]	10624	$(553, 440)$	$249,593$	PNG	animal, building, grass, mountain, plant, sky, water
PIRM [47]	200	$(617, 482)$	$292,021$	PNG	environments, flora, natural scenery, objects, people, etc.
Set5 [48]	5	$(313, 336)$	$113,491$	PNG	baby, bird, butterfly, head, woman
Set14 [49]	14	$(492, 446)$	$230,203$	PNG	humans, animals, insects, flowers, vegetables, comic, slides, etc.
T91 [21]	91	$(264, 204)$	$58,853$	PNG	car, flower, fruit, human face, etc.
Urban100 [50]	100	$(984, 797)$	$774,314$	PNG	architecture, city, structure, urban, etc.

and specifically indicate their amounts of HR images, average resolution, average numbers of pixels, image formats, and category keywords.

Besides these datasets, some datasets widely used for other vision tasks are also employed for SR, such as ImageNet [51], MS-COCO [52], VOC2012 [53], CelebA [54]. In addition, combining multiple datasets for training is also popular, such as combining T91 and BSDS300 [26], [27], [55], [56], combining DIV2K and Flickr2K [31], [57].

2.3 Image Quality Assessment

Image quality refers to visual attributes of images and focuses on the perceptual assessments of viewers. In general, image quality assessment (IQA) methods include subjective methods based on humans' perception (i.e., how realistic the image looks) and objective computational methods. The former is more in line with our need but often time-consuming and expensive, thus the latter is currently the mainstream. However, these methods aren't necessarily consistent between each other, because objective methods are often unable to capture the human visual perception very accurately, which may lead to large difference in IQA results [25], [58].

In addition, the objective IQA methods are further divided into three types [58]: full-reference methods performing assessment using reference images, reduced-reference methods based on comparisons of extracted features, and no-reference methods (i.e., blind IQA) without any reference images. Next we'll introduce several most commonly used IQA methods covering both subjective methods and objective methods.

2.3.1 Peak Signal-to-Noise Ratio

Peak signal-to-noise ratio (PSNR) is one of the most popular reconstruction quality measurement of lossy transformation (e.g., image compression, image inpainting). For image super-resolution, PSNR is defined via the maximum pixel value (denoted as L) and the mean squared error (MSE) between images. Given the ground truth image I with N pixels and the reconstruction $\hat{I}$, the PSNR between I and $\hat{I}$ are defined as follows:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{L^2}{\frac{1}{N}\sum_{i=1}^{N}(I(i) - \hat{I}(i))^2}\right), \tag{6}$$

where L equals to 255 in general cases using 8-bit representations. Since the PSNR is only related to the pixel-level MSE, only caring about the differences between corresponding pixels instead of visual perception, it often leads to poor performance in representing the reconstruction quality in real scenes, where we're usually more concerned with human perceptions. However, due to the necessity to compare with literature works and the lack of completely accurate perceptual metrics, PSNR is still currently the most widely used evaluation criteria for SR models.

2.3.2 Structural Similarity

Considering that the human visual system (HVS) is highly adapted to extract image structures [59], the structural similarity index (SSIM) [58] is proposed for measuring the structural similarity between images, based on independent comparisons in terms of luminance, contrast, and structures. For an image I with N pixels, the luminance μ_I and contrast σ_I are estimated as the mean and standard deviation of the image intensity, respectively, i.e., $\mu_I = \frac{1}{N}\sum_{i=1}^{N} I(i)$ and $\sigma_I = (\frac{1}{N-1}\sum_{i=1}^{N}(I(i) - \mu_I)^2)^{\frac{1}{2}}$, where $I(i)$ represents the intensity of the i-th pixel of image I. And the comparisons on luminance and contrast, denoted as $\mathcal{C}_l(I, \hat{I})$ and $\mathcal{C}_c(I, \hat{I})$ respectively, are given by:

$$\mathcal{C}_l(I, \hat{I}) = \frac{2\mu_I\mu_{\hat{I}} + C_1}{\mu_I^2 + \mu_{\hat{I}}^2 + C_1}, \tag{7}$$

$$\mathcal{C}_c(I, \hat{I}) = \frac{2\sigma_I\sigma_{\hat{I}} + C_2}{\sigma_I^2 + \sigma_{\hat{I}}^2 + C_2}, \tag{8}$$

where $C_1 = (k_1 L)^2$ and $C_2 = (k_2 L)^2$ are constants for avoiding instability, $k_1 \ll 1$ and $k_2 \ll 1$.

Besides, the image structure is represented by the normalized pixel values (i.e., $(I - \mu_I)/\sigma_I$), whose correlations (i.e., inner product) measure the structural similarity, equiv-

alent to the correlation coefficient between I and $\hat{I}$. Thus the structure comparison function $\mathcal{C}_s(I, \hat{I})$ is defined as:

$$\sigma_{I\hat{I}} = \frac{1}{N-1}\sum_{i=1}^{N}(I(i)-\mu_I)(\hat{I}(i)-\mu_{\hat{I}}), \tag{9}$$

$$\mathcal{C}_s(I, \hat{I}) = \frac{\sigma_{I\hat{I}} + C_3}{\sigma_I \sigma_{\hat{I}} + C_3}, \tag{10}$$

where $\sigma_{I,\hat{I}}$ is the covariance between I and $\hat{I}$, and C_3 is a constant for stability.

Finally, the SSIM is given by:

$$\mathrm{SSIM}(I, \hat{I}) = [\mathcal{C}_l(I, \hat{I})]^{\alpha}[\mathcal{C}_c(I, \hat{I})]^{\beta}[\mathcal{C}_s(I, \hat{I})]^{\gamma}, \tag{11}$$

where α, β, γ are control parameters for adjusting the relative importance.

Since the SSIM evaluates the reconstruction quality from the perspective of the HVS, it better meets the requirements of perceptual assessment [60], [61], and is also widely used.

2.3.3 Mean Opinion Score

Mean opinion score (MOS) testing is a commonly used subjective IQA method, where human raters are asked to assign perceptual quality scores to tested images. Typically, the scores are from 1 (bad) to 5 (good). And the final MOS is calculated as the arithmetic mean over all ratings.

Although the MOS testing seems a faithful IQA method, it has some inherent defects, such as non-linearly perceived scales, biases and variance of rating criteria. In reality, there are some SR models performing poorly in common IQA metrics (e.g., PSNR) but far exceeding others in terms of perceptual quality, in which case the MOS testing is the most reliable IQA method for accurately measuring the perceptual quality [8], [25], [46], [62], [63], [64], [65].

2.3.4 Learning-based Perceptual Quality

In order to better assess the image perceptual quality while reducing manual intervention, researchers try to assess the perceptual quality by learning on large datasets. Specifically, Ma *et al.* [66] and Talebi *et al.* [67] propose no-reference Ma and NIMA, respectively, which are learned from visual perceptual scores and directly predict the quality scores without ground-truth images. In contrast, Kim *et al.* [68] propose DeepQA, which predicts visual similarity of images by training on triplets of distorted images, objective error maps, and subjective scores. And Zhang *et al.* [69] collect a large-scale perceptual similarity dataset, evaluate the perceptual image patch similarity (LPIPS) according to the difference in deep features by trained deep networks, and show that the deep features learned by CNNs model perceptual similarity much better than measures without CNNs.

Although these methods exhibit better performance on capturing human visual perception, what kind of perceptual quality we need (e.g., more realistic images, or consistent identity to the original image) remains a question to be explored, thus the objective IQA methods (e.g., PSNR, SSIM) are still the mainstreams currently.

2.3.5 Task-based Evaluation

According to the fact that SR models can often help other vision tasks [6], [7], [8], [9], evaluating reconstruction performance by means of other tasks is another effective way. Specifically, researchers feed the original and the reconstructed HR images into trained models, and evaluate the reconstruction quality by comparing the impacts on the prediction performance. The vision tasks used for evaluation include object recognition [8], [70], face recognition [71], [72], face alignment and parsing [30], [73], etc.

2.3.6 Other IQA Methods

In addition to above IQA methods, there are other less popular SR metrics. The multi-scale structural similarity (MS-SSIM) [74] supplies more flexibility than single-scale SSIM in incorporating the variations of viewing conditions. The feature similarity (FSIM) [75] extracts feature points of human interest based on phase congruency and image gradient magnitude to evaluate image quality. The Natural Image Quality Evaluator (NIQE) [76] makes use of measurable deviations from statistical regularities observed in natural images, without exposure to distorted images.

Recently, Blau *et al.* [77] prove mathematically that distortion (e.g., PSNR, SSIM) and perceptual quality (e.g., MOS) are at odds with each other, and show that as the distortion decreases, the perceptual quality must be worse. Thus how to accurately measure the SR quality is still an urgent problem to be solved.

2.4 Operating Channels

In addition to the commonly used RGB color space, the YCbCr color space is also widely used for SR. In this space, images are represented by Y, Cb, Cr channels, denoting the luminance, blue-difference and red-difference chroma components, respectively. Although currently there is no accepted best practice for performing or evaluating super-resolution on which space, earlier models favor operating on the Y channel of YCbCr space [26], [43], [78], [79], while more recent models tend to operate on RGB channels [28], [31], [57], [70]. It is worth noting that operating (training or evaluation) on different color spaces or channels can make the evaluation results differ greatly (up to 4 dB) [23].

2.5 Super-resolution Challenges

In this section, we will briefly introduce two most popular challenges for image SR, NTIRE [80] and PIRM [47], [81].

NTIRE Challenge. The New Trends in Image Restoration and Enhancement (NTIRE) challenge [80] is in conjunction with CVPR and includes multiple tasks like SR, denoising and colorization. For image SR, the NTIRE challenge is built on the DIV2K [42] dataset and consists of bicubic downscaling tracks and blind tracks with realistic unknown degradation. These tracks differs in degradations and scaling factors, and aim to promote the SR research under both ideal conditions and real-world adverse situations.

PIRM Challenge. The Perceptual Image Restoration and Manipulation (PIRM) challenges are in conjunction with ECCV and also includes multiple tasks. In contrast to NTIRE, one sub-challenge [47] of PIRM focuses on the trade-off between generation accuracy and perceptual quality, and

the other [81] focuses on SR on smartphones. As is well-known [77], the models target for distortion frequently produce visually unpleasing results, while the models target for perceptual quality performs poorly on information fidelity. Specifically, the PIRM divided the perception-distortion plane into three regions according to thresholds on root mean squared error (RMSE). In each region, the winning algorithm is the one that achieves the best perceptual quality [77], evaluated by NIQE [76] and Ma [66]. While in the other sub-challenge [81], SR on smartphones, participants are asked to perform SR with limited smartphone hardwares (including CPU, GPU, RAM, etc.), and the evaluation metrics include PSNR, MS-SSIM and MOS testing. In this way, PIRM encourages advanced research on the perception-distortion tradeoff, and also drives lightweight and efficient image enhancement on smartphones.

3 Supervised Super-resolution

Nowadays researchers have proposed a variety of super-resolution models with deep learning. These models focus on supervised SR, i.e., trained with both LR images and corresponding HR images. Although the differences between these models are very large, they are essentially some combinations of a set of components such as model frameworks, upsampling methods, network design, and learning strategies. From this perspective, researchers combine these components to build an integrated SR model for fitting specific purposes. In this section, we concentrate on modularly analyzing the fundamental components (as Fig. 1 shows) instead of introducing each model in isolation, and summarizing their advantages and limitations.

3.1 Super-resolution Frameworks

Since image super-resolution is an ill-posed problem, how to perform upsampling (i.e., generating HR output from LR input) is the key problem. Although the architectures of existing models vary widely, they can be attributed to four model frameworks (as Fig. 2 shows), based on the employed upsampling operations and their locations in the model.

3.1.1 Pre-upsampling Super-resolution

On account of the difficulty of directly learning the mapping from low-dimensional space to high-dimensional space, utilizing traditional upsampling algorithms to obtain higher-resolution images and then refining them using deep neural networks is a straightforward solution. Thus Dong *et al.* [22], [23] firstly adopt the pre-upsampling SR framework (as Fig. 2a shows) and propose SRCNN to learn an end-to-end mapping from interpolated LR images to HR images. Specifically, the LR images are upsampled to coarse HR images with the desired size using traditional methods (e.g., bicubic interpolation), then deep CNNs are applied on these images for reconstructing high-quality details.

Since the most difficult upsampling operation has been completed, CNNs only need to refine the coarse images, which significantly reduces the learning difficulty. In addition, these models can take interpolated images with arbitrary sizes and scaling factors as input, and give refined results with comparable performance to single-scale

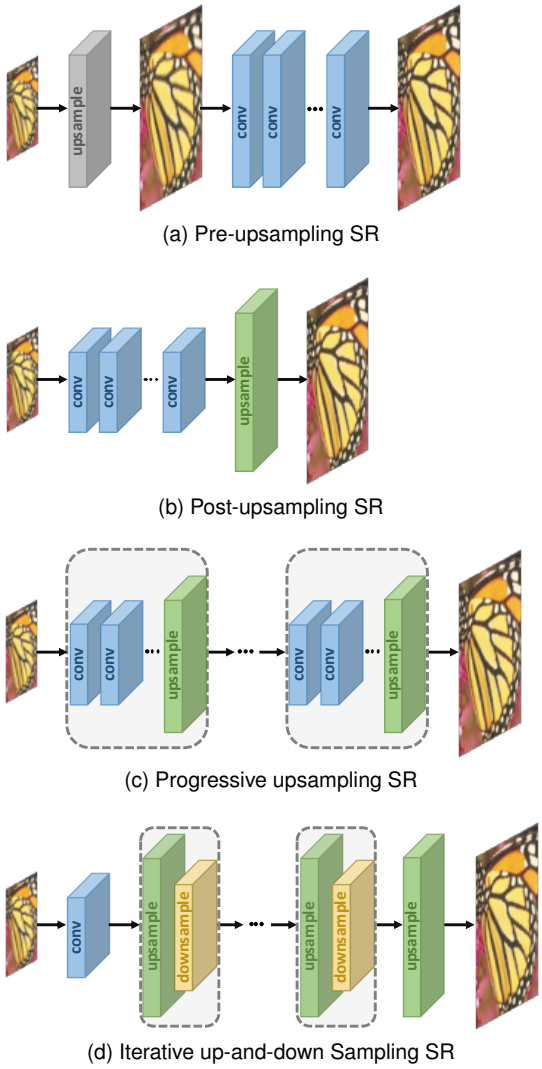


Fig. 2. Super-resolution model frameworks based on deep learning. The cube size represents the output size. The gray ones denote predefined upsampling, while the green, yellow and blue ones indicate learnable upsampling, downsampling and convolutional layers, respectively. And the blocks enclosed by dashed boxes represent stackable modules.

SR models [26]. Thus it has gradually become one of the most popular frameworks [55], [56], [82], [83], and the main differences between these models are the posterior model design (Sec. 3.3) and learning strategies (Sec. 3.4). However, the predefined upsampling often introduce side effects (e.g., noise amplification and blurring), and since most operations are performed in high-dimensional space, the cost of time and space is much higher than other frameworks [43], [84].

3.1.2 Post-upsampling Super-resolution

In order to improve the computational efficiency and make full use of deep learning technology to increase resolution automatically, researchers propose to perform most computation in low-dimensional space by replacing the predefined upsampling with end-to-end learnable layers integrated at the end of the models. In the pioneer works [43], [84] of this framework, namely post-upsampling SR as Fig. 2b shows, the LR input images are fed into deep CNNs without increasing resolution, and end-to-end learnable upsampling layers are applied at the end of the network.

Since the feature extraction process with huge computational cost only occurs in low-dimensional space and the resolution increases only at the end, the computation and spatial complexity are much reduced. Therefore, this framework also has become one of the most mainstream frameworks [25], [31], [79], [85]. These models differ mainly in the learnable upsampling layers (Sec. 3.2), anterior CNN structures (Sec. 3.3) and learning strategies (Sec. 3.4), etc.

3.1.3 Progressive Upsampling Super-resolution

Although post-upsampling SR framework has immensely reduced the computational cost, it still has some shortcomings. On the one hand, the upsampling is performed in only one step, which greatly increases the learning difficulty for large scaling factors (e.g., 4, 8). On the other hand, each scaling factor requires training an individual SR model, which cannot cope with the need for multi-scale SR. To address these drawbacks, a progressive upsampling framework is adopted by Laplacian pyramid SR network (LapSRN) [27], as Fig. 2c shows. Specifically, the models under this framework are based on a cascade of CNNs and progressively reconstruct higher-resolution images. At each stage, the images are upsampled to higher resolution and refined by CNNs. Other works such as MS-LapSRN [65] and progressive SR (ProSR) [32] also adopt this framework and achieve relatively high performance. In contrast to the LapSRN and MS-LapSRN using the intermediate reconstructed images as the "base images" for subsequent modules, the ProSR keeps the main information stream and reconstructs intermediate-resolution images by individual heads.

By decomposing a difficult task into simple tasks, the models under this framework greatly reduce the learning difficulty, especially with large factors, and also cope with the multi-scale SR without introducing overmuch spacial and temporal cost. In addition, some specific learning strategies such as curriculum learning (Sec. 3.4.3) and multi-supervision (Sec. 3.4.4) can be directly integrated to further reduce learning difficulty and improve final performance. However, these models also encounter some problems, such as the complicated model designing for multiple stages and the training stability, and more modelling guidance and more advanced training strategies are needed.

3.1.4 Iterative Up-and-down Sampling Super-resolution

In order to better capture the mutual dependency of LR-HR image pairs, an efficient iterative procedure named back-projection [12] is incorporated into SR [44]. This SR framework, namely iterative up-and-down sampling SR (as Fig. 2d shows), tries to iteratively apply back-projection refinement, i.e., computing the reconstruction error then fusing it back to tune the HR image intensity. Specifically, Haris *et al.* [57] exploit iterative up-and-down sampling layers and propose DBPN, which connects upsampling and downsampling layers alternately and reconstructs the final HR result using all of the intermediately reconstructions. Similarly, the SRFBN [86] employs a iterative up-and-down sampling feedback block with more dense skip connections and learns better representations. And the RBPN [87] for video super-resolution extracts context from continuous video frames and combines these context to produce recurrent output frames by a back-projection module.

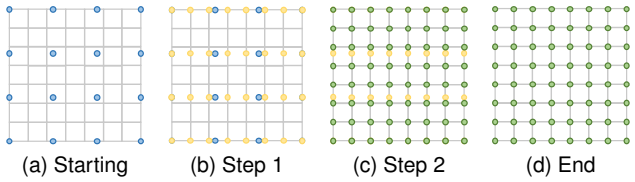


Fig. 3. Interpolation-based upsampling methods. The gray board denotes the coordinates of pixels, and the blue, yellow and green points represent the initial, intermediate and output pixels, respectively.

The models under this framework can better mine the deep relationships between LR-HR image pairs and thus provide higher-quality reconstruction results. Nevertheless, the design criteria of the back-projection modules are still unclear. Since this mechanism has just been introduced into deep learning-based SR, the framework has great potential and needs further exploration.

3.2 Upsampling Methods

In addition to the upsampling positions in the model, how to perform upsampling is of great importance. Although there has been various traditional upsampling methods [20], [21], [88], [89], making use of CNNs to learn end-to-end upsampling has gradually become a trend. In this section, we'll introduce some traditional interpolation-based algorithms and deep learning-based upsampling layers.

3.2.1 Interpolation-based Upsampling

Image interpolation, a.k.a. image scaling, refers to resizing digital images and is widely used by image-related applications. The traditional interpolation methods include nearest-neighbor interpolation, bilinear and bicubic interpolation, Sinc and Lanczos resampling, etc. Since these methods are interpretable and easy to implement, some of them are still widely used in CNN-based SR models.

Nearest-neighbor Interpolation. The nearest-neighbor interpolation is a simple and intuitive algorithm. It selects the value of the nearest pixel for each position to be interpolated regardless of any other pixels. Thus this method is very fast but usually produces blocky results of low quality.

Bilinear Interpolation. The bilinear interpolation (BLI) first performs linear interpolation on one axis of the image and then performs on the other axis, as Fig. 3 shows. Since it results in a quadratic interpolation with a receptive field sized 2×2, it shows much better performance than nearest-neighbor interpolation while keeping relatively fast speed.

Bicubic Interpolation. Similarly, the bicubic interpolation (BCI) [10] performs cubic interpolation on each of the two axes, as Fig. 3 shows. Compared to BLI, the BCI takes 4×4 pixels into account, and results in smoother results with fewer artifacts but much lower speed. In fact, the BCI with anti-aliasing is the mainstream method for building SR datasets (i.e., degrading HR images to LR images), and is also widely used in pre-upsampling SR framework (Sec. 3.1.1).

As a matter of fact, the interpolation-based upsampling methods improve the image resolution only based on its own image signals, without bringing any more information.

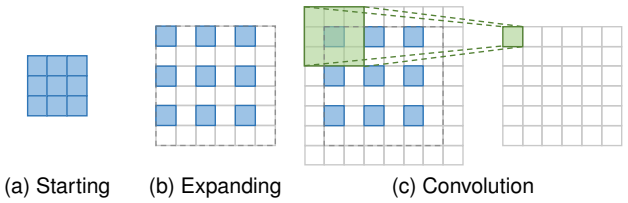


Fig. 4. Transposed convolution layer. The blue boxes denote the input, and the green boxes indicate the kernel and the convolution output.

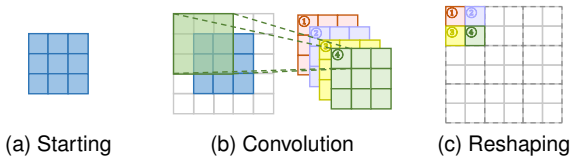


Fig. 5. Sub-pixel layer. The blue boxes denote the input, and the boxes with other colors indicate different convolution operations and different output feature maps.

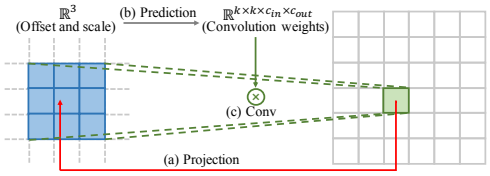


Fig. 6. Meta upscale module. The blue boxes denote the projection patch, and the green boxes and lines indicate the convolution operation with predicted weights.

Instead, they often introduce some side effects, such as computational complexity, noise amplification, blurring results. Therefore, the current trend is to replace the interpolation-based methods with learnable upsampling layers.

3.2.2 Learning-based Upsampling

In order to overcome the shortcomings of interpolation-based methods and learn upsampling in an end-to-end manner, transposed convolution layer and sub-pixel layer are introduced into the SR field.

Transposed Convolution Layer. Transposed convolution layer, a.k.a. deconvolution layer [90], [91], tries to perform transformation opposite a normal convolution, i.e., predicting the possible input based on feature maps sized like convolution output. Specifically, it increases the image resolution by expanding the image by inserting zeros and performing convolution. Taking $2\times$ SR with 3×3 kernel as example (as Fig. 4 shows), the input is firstly expanded twice of the original size, where the added pixel values are set to 0 (Fig. 4b). Then a convolution with kernel sized 3×3, stride 1 and padding 1 is applied (Fig. 4c). In this way, the input is upsampled by a factor of 2, in which case the receptive field is at most 2×2. Since the transposed convolution enlarges the image size in an end-to-end manner while maintaining a connectivity pattern compatible with vanilla convolution, it is widely used as upsampling layers in SR models [57], [78], [79], [85]. However, this layer can easily cause "uneven overlapping" on each axis [92], and the multiplied results on both axes further create a checkerboard-like pattern of varying magnitudes and thus hurt the SR performance.

Sub-pixel Layer. The sub-pixel layer [84], another end-to-end learnable upsampling layer, performs upsampling by generating a plurality of channels by convolution and then reshaping them, as Fig. 5 shows. Within this layer, a convolution is firstly applied for producing outputs with s^2 times channels, where s is the scaling factor (Fig. 5b). Assuming the input size is $h \times w \times c$, the output size will be $h \times w \times s^2c$. After that, the reshaping operation (a.k.a. *shuffle* [84]) is performed to produce outputs with size $sh \times sw \times c$ (Fig. 5c). In this case, the receptive field can be up to 3×3. Due to the end-to-end upsampling manner, this layer is also widely used by SR models [25], [28], [39], [93]. Compared with transposed convolution layer, the sub-pixel layer has a larger receptive field, which provides more contextual information to help generate more realistic details. However, since the distribution of the receptive fields is uneven and blocky regions actually share the same receptive field, it may result in some artifacts near the boundaries of different blocks. On the other hand, independently predicting adjacent pixels in a blocky region may cause unsmooth outputs. Thus Gao *et al.* [94] propose PixelTCL, which replaces the independent prediction to interdependent sequential prediction, and produces smoother and more consistent results.

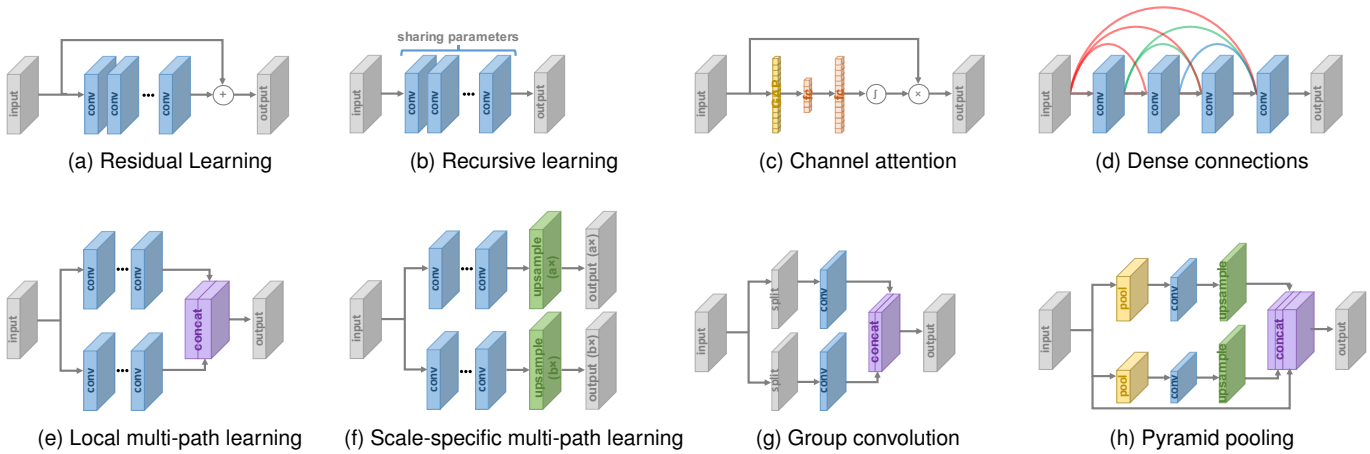
Meta Upscale Module. The previous methods need to predefine the scaling factors, i.e., training different upsampling modules for different factors, which is inefficient and not in line with real needs. So that Hu *et al.* [95] propose meta upscale module (as Fig. 6 shows), which firstly solves SR of arbitrary scaling factors based on meta learning. Specifically, for each target position on the HR images, this module project it to a small patch on the LR feature maps (i.e., $k \times k \times c_{in}$), predicts convolution weights (i.e., $k \times k \times c_{in} \times c_{out}$) according to the projection offsets and the scaling factor by dense layers and perform convolution. In this way, the meta upscale module can continuously zoom in it with arbitrary factors by a single model. And due to the large amount of training data (multiple factors are simultaneously trained), the module can exhibit comparable or even better performance on fixed factors. Although this module needs to predict weights during inference, the execution time of the upsampling module only accounts for about 1% of the time of feature extraction [95]. However, this method predicts a large number of convolution weights for each target pixel based on several values independent of the image contents, so the prediction result may be unstable and less efficient when faced with larger magnifications.

Nowadays, these learning-based layers have become the most widely used upsampling methods. Especially in the post-upsampling framework (Sec. 3.1.2), these layers are usually used in the final upsampling phase for reconstructing HR images based on high-level representations extracted in low-dimensional space, and thus achieve end-to-end SR while avoiding overwhelming operations in high-dimensional space.

3.3 Network Design

Nowadays the network design has been one of the most important parts of deep learning. In the super-resolution

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.2982166, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) Residual Learning (b) Recursive learning (c) Channel attention (d) Dense connections

(e) Local multi-path learning (f) Scale-specific multi-path learning (g) Group convolution (h) Pyramid pooling

Fig. 7. Network design strategies.

field, researchers apply all kinds of network design strategies on top of the four SR frameworks (Sec. 3.1) to construct the final networks. In this section, we decompose these networks to essential principles or strategies for network design, introduce them and analyze the advantages and limitations one by one.

3.3.1 Residual Learning

Before He *et al.* [96] propose ResNet for learning residuals instead of a thorough mapping, residual learning has been widely employed by SR models [48], [88], [97], as Fig. 7a shows. Among them, the residual learning strategies can be roughly divided into global and local residual learning.

Global Residual Learning. Since the image SR is an image-to-image translation task where the input image is highly correlated with the target image, researchers try to learn only the residuals between them, namely global residual learning. In this case, it avoids learning a complicated transformation from a complete image to another, instead only requires learning a residual map to restore the missing high-frequency details. Since the residuals in most regions are close to zero, the model complexity and learning difficulty are greatly reduced. Thus it is widely used by SR models [26], [55], [56], [98].

Local Residual Learning. The local residual learning is similar to the residual learning in ResNet [96] and used to alleviate the degradation problem [96] caused by ever-increasing network depths, reduce training difficulty and improve the learning ability. It is also widely used for SR [70], [78], [85], [99].

In practice, the above methods are both implemented by shortcut connections (often scaled by a small constant) and element-wise addition, while the difference is that the former directly connects the input and output images, while the latter usually adds multiple shortcuts between layers with different depths inside the network.

3.3.2 Recursive Learning

In order to learn higher-level features without introducing overwhelming parameters, recursive learning, which means applying the same modules multiple times in a recursive manner, is introduced into the SR field, as Fig. 7b shows.

Among them, the 16-recursive DRCN [82] employs a single convolutional layer as the recursive unit and reaches a receptive field of 41×41, which is much larger than 13×13 of SRCNN [22], without over many parameters. The DRRN [56] uses a ResBlock [96] as the recursive unit for 25 recursions and obtains even better performance than the 17-ResBlock baseline. Later Tai *et al.* [55] propose MemNet based on the memory block, which is composed of a 6-recursive ResBlock where the outputs of every recursion are concatenated and go through an extra 1×1 convolution for memorizing and forgetting. The cascading residual network (CARN) [28] also adopts a similar recursive unit including several ResBlocks. Recently, Li *et al.* [86] employ iterative up-and-down sampling SR framework, and propose a feedback network based on recursive learning, where the weights of the entire network are shared across all recursions.

Besides, researchers also employ different recursive modules in different parts. Specifically, Han *et al.* [85] propose dual-state recurrent network (DSRN) to exchange signals between the LR and HR states. At each time step (i.e., recursion), the representations of each branch are updated and exchanged for better exploring LR-HR relationships. Similarly, Lai *et al.* [65] employ the embedding and upsampling modules as recursive units, and thus much reduce the model size at the expense of little performance loss.

In general, the recursive learning can indeed learn more advanced representations without introducing excessive parameters, but still can't avoid high computational costs. And it inherently brings vanishing or exploding gradient problems, consequently some techniques such as residual learning (Sec. 3.3.1) and multi-supervision (Sec. 3.4.4) are often integrated with recursive learning for mitigating these problems [55], [56], [82], [85].

3.3.3 Multi-path Learning

Multi-path learning refers to passing features through multiple paths, which perform different operations, and fusing them back for providing better modelling capabilities. Specifically, it could be divided into global, local and scale-specific multi-path learning, as bellow.

Global Multi-path Learning. Global multi-path learning refers to making use of multiple paths to extract features

of different aspects of the images. These paths can cross each other in the propagation and thus greatly enhance the learning ability. Specifically, the LapSRN [27] includes a feature extraction path predicting the sub-band residuals in a coarse-to-fine fashion and another path to reconstruct HR images based on the signals from both paths. Similarly, the DSRN [85] utilizes two paths to extract information in low-dimensional and high-dimensional space, respectively, and continuously exchanges information for further improving learning ability. And the pixel recursive super-resolution [64] adopts a conditioning path to capture the global structure of images, and a prior path to capture the serial dependence of generated pixels. In contrast, Ren *et al.* [100] employ multiple paths with unbalanced structures to perform upsampling and fuse them at the end of the model.

Local Multi-path Learning. Motivated by the inception module [101], the MSRN [99] adopts a new block for multi-scale feature extraction, as Fig. 7e shows. In this block, two convolution layers with kernel size 3×3 and 5×5 are adopted to extract features simultaneously, then the outputs are concatenated and go through the same operations again, and finally an extra 1×1 convolution is applied. A shortcut connects the input and output by element-wise addition. Through such local multi-path learning, the SR models can better extract image features from multiple scales and further improve performance.

Scale-specific Multi-path Learning. Considering that SR models for different scales need to go through similar feature extraction, Lim *et al.* [31] propose scale-specific multi-path learning to cope with multi-scale SR with a single network. To be concrete, they share the principal components of the model (i.e., the intermediate layers for feature extraction), and attach scale-specific pre-processing paths and upsampling paths at the beginning and the end of the network, respectively (as Fig. 7f shows). During training, only the paths corresponding to the selected scale are enabled and updated. In this way, the proposed MDSR [31] greatly reduce the model size by sharing most of the parameters for different scales and exhibits comparable performance as single-scale models. The similar scale-specific multi-path learning is also adopted by CARN [28] and ProSR [32].

3.3.4 Dense Connections

Since Huang *et al.* [102] propose DenseNet based on dense blocks, the dense connections have become more and more popular in vision tasks. For each layer in a dense block, the feature maps of all preceding layers are used as inputs, and its own feature maps are used as inputs into all subsequent layers, so that it leads to $l \cdot (l-1)/2$ connections in a l-layer dense block ($l \geq 2$). The dense connections not only help alleviate gradient vanishing, enhance signal propagation and encourage feature reuse, but also substantially reduce the model size by employing small growth rate (i.e., number of channels in dense blocks) and squeezing channels after concatenating all input feature maps.

For the sake of fusing low-level and high-level features to provide richer information for reconstructing high-quality details, dense connections are introduced into the SR field, as Fig. 7d shows. Tong *et al.* [79] not only adopt dense blocks to construct a 69-layers SRDenseNet, but also insert dense connections between different dense blocks, i.e., for every dense block, the feature maps of all preceding blocks are used as inputs, and its own feature maps are used as inputs into all subsequent blocks. These layer-level and block-level dense connections are also adopted by MemNet [55], CARN [28], RDN [93] and ESRGAN [103]. The DBPN [57] also adopts dense connections extensively, but their dense connections are between all the upsampling units, as are the downsampling units.

3.3.5 Attention Mechanism

Channel Attention. Considering the interdependence and interaction of the feature representations between different channels, Hu *et al.* [104] propose a "squeeze-and-excitation" block to improve learning ability by explicitly modelling channel interdependence, as Fig. 7c shows. In this block, each input channel is squeezed into a channel descriptor (i.e., a constant) using global average pooling (GAP), then these descriptors are fed into two dense layers to produce channel-wise scaling factors for input channels. Recently, Zhang *et al.* [70] incorporate the channel attention mechanism with SR and propose RCAN, which markedly improves the representation ability of the model and SR performance. In order to better learn the feature correlations, Dai *et al.* [105] further propose a second-order channel attention (SOCA) module. The SOCA adaptively rescales the channel-wise features by using second-order feature statistics instead of GAP, and enables extracting more informative and discriminative representations.

Non-local Attention. Most existing SR models have very limited local receptive fields. However, some distant objects or textures may be very important for local patch generation. So that Zhang *et al.* [106] propose local and non-local attention blocks to extract features that capture the long-range dependencies between pixels. Specifically, they propose a trunk branch for extracting features, and a (non-)local mask branch for adaptively rescaling features of trunk branch. Among them, the local branch employs an encoder-decoder structure to learn the local attention, while the non-local branch uses the embedded Gaussian function to evaluate pairwise relationships between every two position indices in the feature maps to predict the scaling weights. Through this mechanism, the proposed method captures the spatial attention well and further enhances the representation ability. Similarly, Dai *et al.* [105] also incorporate the non-local attention mechanism to capture long-distance spatial contextual information.

3.3.6 Advanced Convolution

Since convolution operations are the basis of deep neural networks, researchers also attempt to improve convolution operations for better performance or greater efficiency.

Dilated Convolution. It is well known that the contextual information facilitates generating realistic details for SR. Thus Zhang *et al.* [107] replace the common convolution by dilated convolution in SR models, increase the receptive field over twice and achieve much better performance.

Group Convolution. Motivated by recent advances on lightweight CNNs [108], [109], Hui *et al.* [98] and Ahn *et al.* [28] propose IDN and CARN-M, respectively, by replacing the vanilla convolution by group convolution. As some previous works have proven, the group convolution much

reduces the number of parameters and operations at the expense of a little performance loss [28], [98].

Depthwise Separable Convolution Since Howard *et al.* [110] propose depthwise separable convolution for efficient convolution, it has been expanded to into various fields. Specifically, it consists of a factorized depthwise convolution and a pointwise convolution (i.e., 1×1 convolution), and thus reduces plenty of parameters and operations at only a small reduction in accuracy [110]. And recently, Nie *et al.* [81] employ the depthwise separable convolution and much accelerate the SR architecture.

3.3.7 Region-recursive Learning

Most SR models treat SR as a pixel-independent task and thus cannot source the interdependence between generated pixels properly. Inspired by PixelCNN [111], Dahl *et al.* [64] firstly propose pixel recursive learning to perform pixel-by-pixel generation, by employing two networks to capture global contextual information and serial generation dependence, respectively. In this way, the proposed method synthesizes realistic hair and skin details on super-resolving very low-resolution face images (e.g., 8×8) and far exceeds the previous methods on MOS testing [64] (Sec. 2.3.3).

Motivated by the human attention shifting mechanism [112], the Attention-FH [113] also adopts this strategy by resorting to a recurrent policy network for sequentially discovering attended patches and performing local enhancement. In this way, it is capable of adaptively personalizing an optimal searching path for each image according to its own characteristic, and thus fully exploits the global intra-dependence of images.

Although these methods show better performance to some extent, the recursive process requiring a long propagation path greatly increases the computational cost and training difficulty, especially for super-resolving HR images.

3.3.8 Pyramid Pooling

Motivated by the spatial pyramid pooling layer [114], Zhao *et al.* [115] propose the pyramid pooling module to better utilize global and local contextual information. Specifically, for feature maps sized $h \times w \times c$, each feature map is divided into $M \times M$ bins, and goes through global average pooling, resulting in $M \times M \times c$ outputs. Then a 1×1 convolution is performed for compressing the outputs to a single channel. After that, the low-dimensional feature map is upsampled to the same size as the original feature map via bilinear interpolation. By using different M, the module integrates global as well as local contextual information effectively. By incorporating this module, the proposed EDSR-PP model [116] further improve the performance over baselines.

3.3.9 Wavelet Transformation

As is well-known, the wavelet transformation (WT) [117], [118] is a highly efficient representation of images by decomposing the image signal into high-frequency sub-bands denoting texture details and low-frequency sub-bands containing global topological information. Bae *et al.* [119] firstly combine WT with deep learning based SR model, take sub-bands of interpolated LR wavelet as input and predict residuals of corresponding HR sub-bands. WT and inverse WT are applied for decomposing the LR input and reconstructing the HR output, respectively. Similarly, the DWSR [120] and Wavelet-SRNet [121] also perform SR in the wavelet domain but with more complicated structures. In contrast to the above works processing each sub-band independently, the MWCNN [122] adopts multi-level WT and takes the concatenated sub-bands as the input to a single CNN for better capturing the dependence between them. Due to the efficient representation by wavelet transformation, the models using this strategy often much reduce the model size and computational cost, while maintain competitive performance [119], [122].

3.3.10 Desubpixel

In order to speed up the inference speed, Vu *et al.* [123] propose to perform the time-consuming feature extraction in a lower-dimensional space, and propose desubpixel, an inverse of the shuffle operation of sub-pixel layer (Sec. 3.2.2). Specifically, the desubpixel operation splits the images spatially, stacks them as extra channels and thus avoids loss of information. In this way, they downsample input images by desubpixel at the beginning of the model, learn representations in a lower-dimensional space, and upsample to the target size at the end. The proposed model achieves the best scores in the PIRM Challenge on Smartphones [81] with very high-speed inference and good performance.

3.3.11 xUnit

In order to combine spatial feature processing and nonlinear activations to learn complex features more efficiently, Kligvasser *et al.* [124] propose xUnit for learning a spatial activation function. Specifically, the ReLU is regarded as determining a weight map to perform element-wise multiplication with the input, while the xUnit directly learn the weight map through convolution and Gaussian gating. Although the xUnit is more computationally demanding, due to its dramatic effect on the performance, it allows greatly reducing the model size while matching the performance with ReLU. In this way, the authors reduce the model size by nearly 50% without any performance degradation.

3.4 Learning Strategies

3.4.1 Loss Functions

In the super-resolution field, loss functions are used to measure reconstruction error and guide the model optimization. In early times, researchers usually employ the pixel-wise L2 loss, but later discover that it cannot measure the reconstruction quality very accurately. Therefore, a variety of loss functions (e.g., content loss [29], adversarial loss [25]) are adopted for better measuring the reconstruction error and producing more realistic and higher-quality results. Nowadays these loss functions have been playing an important role. In this section, we'll take a closer look at the loss functions used widely. The notations in this section follow Sec. 2.1, except that we ignore the subscript y of the target HR image $\hat{I}_y$ and generated HR image I_y for brevity.

Pixel Loss. Pixel loss measures pixel-wise difference between two images and mainly includes L1 loss (i.e., mean absolute error) and L2 loss (i.e., mean square error):

$$\mathcal{L}_{\text{pixel\_l1}}(\hat{I}, I) = \frac{1}{hwc}\sum_{i,j,k}|\hat{I}_{i,j,k} - I_{i,j,k}|, \tag{12}$$

$$\mathcal{L}_{\text{pixel\_l2}}(\hat{I}, I) = \frac{1}{hwc}\sum_{i,j,k}(\hat{I}_{i,j,k} - I_{i,j,k})^2, \tag{13}$$

where h, w and c are the height, width and number of channels of the evaluated images, respectively. In addition, there is a variant of the pixel L1 loss, namely Charbonnier loss [27], [125], given by:.

$$\mathcal{L}_{\text{pixel\_Cha}}(\hat{I}, I) = \frac{1}{hwc}\sum_{i,j,k}\sqrt{(\hat{I}_{i,j,k} - I_{i,j,k})^2 + \epsilon^2}, \tag{14}$$

where ϵ is a constant (e.g., 10^{-3}) for numerical stability.

The pixel loss constrains the generated HR image $\hat{I}$ to be close enough to the ground truth I on the pixel values. Comparing with L1 loss, the L2 loss penalizes larger errors but is more tolerant to small errors, and thus often results in too smooth results. In practice, the L1 loss shows improved performance and convergence over L2 loss [28], [31], [126]. Since the definition of PSNR (Sec. 2.3.1) is highly correlated with pixel-wise difference and minimizing pixel loss directly maximize PSNR, the pixel loss gradual becomes the most widely used loss function. However, since the pixel loss actually doesn't take image quality (e.g., perceptual quality [29], textures [8]) into account, the results often lack high-frequency details and are perceptually unsatisfying with oversmooth textures [25], [29], [58], [74].

Content Loss. In order to evaluate perceptual quality of images, the content loss is introduced into SR [29], [127]. Specifically, it measures the semantic differences between images using a pre-trained image classification network. Denoting this network as ϕ and the extracted high-level representations on l-th layer as $\phi^{(l)}(I)$, the content loss is indicated as the Euclidean distance between high-level representations of two images, as follows:

$$\mathcal{L}_{\text{content}}(\hat{I}, I; \phi, l) = \frac{1}{h_l w_l c_l}\sqrt{\sum_{i,j,k}(\phi^{(l)}_{i,j,k}(\hat{I}) - \phi^{(l)}_{i,j,k}(I))^2}, \tag{15}$$

where h_l, w_l and c_l are the height, width and number of channels of the representations on layer l, respectively.

Essentially the content loss transfers the learned knowledge of hierarchical image features from the classification network ϕ to the SR network. In contrast to the pixel loss, the content loss encourages the output image $\hat{I}$ to be perceptually similar to the target image I instead of forcing them to match pixels exactly. Thus it produces visually more perceptible results and is also widely used in this field [8], [25], [29], [30], [46], [103], where the VGG [128] and ResNet [96] are the most commonly used pre-trained CNNs.

Texture Loss. On account that the reconstructed image should have the same style (e.g., colors, textures, contrast) with the target image, and motivated by the style representation by Gatys *et al.* [129], [130], the texture loss (a.k.a style reconstruction loss) is introduced into SR. Following [129], [130], the image texture is regarded as the correlations between different feature channels and defined as the Gram matrix $G^{(l)} \in \mathbb{R}^{c_l \times c_l}$, where $G^{(l)}_{ij}$ is the inner product between the vectorized feature maps i and j on layer l:

$$G^{(l)}_{ij}(I) = \text{vec}(\phi^{(l)}_i(I)) \cdot \text{vec}(\phi^{(l)}_j(I)), \tag{16}$$

where $\text{vec}(\cdot)$ denotes the vectorization operation, and $\phi^{(l)}_i(I)$ denotes the i-th channel of the feature maps on layer l of image I. Then the texture loss is given by:

$$\mathcal{L}_{\text{texture}}(\hat{I}, I; \phi, l) = \frac{1}{c_l^2}\sqrt{\sum_{i,j}(G^{(l)}_{i,j}(\hat{I}) - G^{(l)}_{i,j}(I))^2}. \tag{17}$$

By employing texture loss, the EnhanceNet [8] proposed by Sajjadi *et al.* creates much more realistic textures and produces visually more satisfactory results. Despite this, determining the patch size to match textures is still empirical. Too small patches lead to artifacts in textured regions, while too large patches lead to artifacts throughout the entire image because texture statistics are averaged over regions of varying textures.

Adversarial Loss. In recent years, due to the powerful learning ability, the GANs [24] receive more and more attention and are introduced to various vision tasks. To be concrete, the GAN consists of a generator performing generation (e.g., text generation, image transformation), and a discriminator which takes the generated results and instances sampled from the target distribution as input and discriminates whether each input comes from the target distribution. During training, two steps are alternately performed: (a) fix the generator and train the discriminator to better discriminate, (b) fix the discriminator and train the generator to fool the discriminator. Through adequate iterative adversarial training, the resulting generator can produce outputs consistent with the distribution of real data, while the discriminator can't distinguish between the generated data and real data.

In terms of super-resolution, it is straightforward to adopt adversarial learning, in which case we only need to treat the SR model as a generator, and define an extra discriminator to judge whether the input image is generated or not. Therefore, Ledig *et al.* [25] firstly propose SRGAN using adversarial loss based on cross entropy, as follows:

$$\mathcal{L}_{\text{gan\_ce\_g}}(\hat{I}; D) = -\log D(\hat{I}), \tag{18}$$

$$\mathcal{L}_{\text{gan\_ce\_d}}(\hat{I}, I_s; D) = -\log D(I_s) - \log(1 - D(\hat{I})), \tag{19}$$

where $\mathcal{L}_{\text{gan\_ce\_g}}$ and $\mathcal{L}_{\text{gan\_ce\_d}}$ denote the adversarial loss of the generator (i.e., the SR model) and the discriminator D (i.e., a binary classifier), respectively, and I_s represents images randomly sampled from the ground truths. Besides, the Enhancenet [8] also adopts the similar adversarial loss.

Besides, Wang *et al.* [32] and Yuan *et al.* [131] use adversarial loss based on least square error for more stable training process and higher quality results [132], given by:

$$\mathcal{L}_{\text{gan\_ls\_g}}(\hat{I}; D) = (D(\hat{I}) - 1)^2, \tag{20}$$

$$\mathcal{L}_{\text{gan\_ls\_d}}(\hat{I}, I_s; D) = (D(\hat{I}))^2 + (D(I_s) - 1)^2. \tag{21}$$

In contrast to the above works focusing on the specific forms of adversarial loss, Park *et al.* [133] argue that the pixel-level discriminator causes generating meaningless high-frequency noise, and attach another feature-level

discriminator to operate on high-level representations extracted by a pre-trained CNN which captures more meaningful attributes of real HR images. Xu *et al.* [63] incorporate a multi-class GAN consisting of a generator and multiple class-specific discriminators. And the ESRGAN [103] employs relativistic GAN [134] to predict the probability that real images are relatively more realistic than fake ones, instead of the probability that input images are real or fake, and thus guide recovering more detailed textures.

Extensive MOS tests (Sec. 2.3.3) show that even though the SR models trained with adversarial loss and content loss achieve lower PSNR compared to those trained with pixel loss, they bring significant gains in perceptual quality [8], [25]. As a matter of fact, the discriminator extracts some difficult-to-learn latent patterns of real HR images, and pushes the generated HR images to conform, thus helps to generate more realistic images. However, currently the training process of GAN is still difficult and unstable. Although there have been some studies on how to stabilize the GAN training [135], [136], [137], how to ensure that the GANs integrated into SR models are trained correctly and play an active role remains a problem.

Cycle Consistency Loss. Motivated by the CycleGAN proposed by Zhu *et al.* [138], Yuan *et al.* [131] present a cycle-in-cycle approach for super-resolution. Concretely speaking, they not only super-resolve the LR image I to the HR image $\hat{I}$ but also downsample $\hat{I}$ back to another LR image I' through another CNN. The regenerated I' is required to be identical to the input I, thus the cycle consistency loss is introduced for constraining their pixel-level consistency:

$$\mathcal{L}_{\text{cycle}}(I', I) = \frac{1}{hwc}\sqrt{\sum_{i,j,k}(I'_{i,j,k} - I_{i,j,k})^2}. \tag{22}$$

Total Variation Loss. In order to suppress noise in generated images, the total variation (TV) loss [139] is introduced into SR by Aly *et al.* [140]. It is defined as the sum of the absolute differences between neighboring pixels and measures how much noise is in the images, as follows:

$$\mathcal{L}_{\text{TV}}(\hat{I}) = \frac{1}{hwc}\sum_{i,j,k}\sqrt{(\hat{I}_{i,j+1,k} - \hat{I}_{i,j,k})^2 + (\hat{I}_{i+1,j,k} - \hat{I}_{i,j,k})^2}. \tag{23}$$

Lai *et al.* [25] and Yuan *et al.* [131] also adopt the TV loss for imposing spatial smoothness.

Prior-Based Loss. In addition to the above loss functions, external prior knowledge is also introduced to constrain the generation. Specifically, Bulat *et al.* [30] focus on face image SR and introduce a face alignment network (FAN) to constrain the consistency of facial landmarks. The FAN is pre-trained and integrated for providing face alignment priors, and then trained jointly with the SR. In this way, the proposed Super-FAN improves performance both on LR face alignment and face image SR.

As a matter of fact, the content loss and the texture loss, both of which introduce a classification network, essentially provide prior knowledge of hierarchical image features for SR. By introducing more prior knowledge, the SR performance can be further improved.

In this section, we introduce various loss functions for SR. In practice, researchers often combine multiple loss functions by weighted average [8], [25], [27], [46], [141] for constraining different aspects of the generation process, especially for distortion-perception tradeoff [25], [103], [142], [143], [144]. However, the weights of different loss functions require a lot of empirical exploration, and how to combine reasonably and effectively remains a problem.

3.4.2 Batch Normalization

In order to accelerate and stabilize training of deep CNNs, Sergey *et al.* [145] propose batch normalization (BN) to reduce internal covariate shift of networks. Specifically, they perform normalization for each mini-batch and train two extra transformation parameters for each channel to preserve the representation ability. Since the BN calibrates the intermediate feature distribution and mitigates vanishing gradients, it allows using higher learning rates and being less careful about initialization. Thus this technique is widely used by SR models [25], [39], [55], [56], [122], [146].

However, Lim *et al.* [31] argue that the BN loses the scale information of each image and gets rid of range flexibility from networks. So they remove BN and use the saved memory cost (up to 40%) to develop a much larger model, and thus increase the performance substantially. Some other models [32], [103], [147] also adopt this experience and achieve performance improvements.

3.4.3 Curriculum Learning

Curriculum learning [148] refers to starting from an easier task and gradually increasing the difficulty. Since super-resolution is an ill-posed problem and always suffers adverse conditions such as large scaling factors, noise and blurring, the curriculum training is incorporated for reducing learning difficulty.

In order to reduce the difficulty of SR with large scaling factors, Wang *et al.* [32], Bei *et al.* [149] and Ahn *et al.* [150] propose ProSR, ADRSR and progressive CARN, respectively, which are progressive not only on architectures (Sec. 3.1.3) but also on training procedure. The training starts with the $2\times$ upsampling, and after finishing training, the portions with $4\times$ or larger scaling factors are gradually mounted and blended with the previous portions. Specifically, the ProSR blends by linearly combining the output of this level and the upsampled output of previous levels following [151], the ADRSR concatenates them and attaches another convolutional layer, while the progressive CARN replace the previous reconstruction block with the one that produces the image in double resolution.

In addition, Park *et al.* [116] divide the $8\times$ SR problem to three sub-problems (i.e., $1\times$ to $2\times$, $2\times$ to $4\times$, $4\times$ to $8\times$) and train independent networks for each problem. Then two of them are concatenated and fine-tuned, and then with the third one. Besides, they also decompose the $4\times$ SR under difficult conditions into $1\times$ to $2\times$, $2\times$ to $4\times$ and denoising or deblurring sub-problems. In contrast, the SRFBN [86] uses this strategy for SR under adverse conditions, i.e., starting from easy degradation and gradually increasing degradation complexity.

Compared to common training procedure, the curriculum learning greatly reduces the training difficulty and shortens the total training time, especially for large factors.

3.4.4 Multi-supervision

Multi-supervision refers to adding multiple supervision signals within the model for enhancing the gradient propagation and avoiding vanishing and exploding gradients. In order to prevent the gradient problems introduced by recursive learning (Sec. 3.3.2), the DRCN [82] incorporates multi-supervision with recursive units. Specifically, they feed each output of recursive units into a reconstruction module to generate an HR image, and build the final prediction by incorporating all the intermediate reconstructions. Similar strategies are also taken by MemNet [55] and DSRN [85], which are also based on recursive learning.

Besides, since the LapSRN [27], [65] under the progressive upsampling framework (Sec. 3.1.3) generates intermediate results of different scales during propagation, it is straightforward to adopt multi-supervision strategy. Specifically, the intermediate results are forced to be the same as the intermediate images downsampled from the ground truth HR images.

In practice, this multi-supervision technique is often implemented by adding some terms in the loss function, and in this way, the supervision signals are back-propagated more effectively, and thus reduce the training difficulty and enhance the model training.

3.5 Other Improvements

In addition to the network design and learning strategies, there are other techniques further improving SR models.

3.5.1 Context-wise Network Fusion

Context-wise network fusion (CNF) [100] refers to a stacking technique fusing predictions from multiple SR networks (i.e., a special case of multi-path learning in Sec. 3.3.3). To be concrete, they train individual SR models with different architectures separately, feed the prediction of each model into individual convolutional layers, and finally sum the outputs up to be the final prediction result. Within this CNF framework, the final model constructed by three lightweight SRCNNs [22], [23] achieves comparable performance with state-of-the-art models with acceptable efficiency [100].

3.5.2 Data Augmentation

Data augmentation is one of the most widely used techniques for boosting performance with deep learning. For image super-resolution, some useful augmentation options include cropping, flipping, scaling, rotation, color jittering, etc. [27], [31], [44], [56], [85], [98]. In addition, Bei *et al.* [149] also randomly shuffle RGB channels, which not only augments data, but also alleviates color bias caused by the dataset with color unbalance.

3.5.3 Multi-task Learning

Multi-task learning [152] refers to improving generalization ability by leveraging domain-specific information contained in training signals of related tasks, such as object detection and semantic segmentation [153], head pose estimation and facial attribute inference [154]. In the SR field, Wang *et al.* [46] incorporate a semantic segmentation network for providing semantic knowledge and generating semantic-specific details. Specifically, they propose spatial feature transformation to take semantic maps as input and predict spatial-wise parameters of affine transformation performed on the intermediate feature maps. The proposed SFT-GAN thus generates more realistic and visually pleasing textures on images with rich semantic regions. Besides, considering that directly super-resolving noisy images may cause noise amplification, the DNSR [149] proposes to train a denoising network and an SR network separately, then concatenates them and fine-tunes together. Similarly, the cycle-in-cycle GAN (CinCGAN) [131] combines a cycle-in-cycle denoising framework and a cycle-in-cycle SR model to joint perform noise reduction and super-resolution. Since different tasks tend to focus on different aspects of the data, combining related tasks with SR models usually improves the SR performance by providing extra information and knowledge.

3.5.4 Network Interpolation

PSNR-based models produce images closer to ground truths but introduce blurring problems, while GAN-based models bring better perceptual quality but introduce unpleasant artifacts (e.g., meaningless noise making images more "realistic"). In order to better balance the distortion and perception, Wang *et al.* [103], [155] propose a network interpolation strategy. Specifically, they train a PSNR-based model and train a GAN-based model by fine-tuning, then interpolate all the corresponding parameters of both networks to derive intermediate models. By tuning the interpolation weights without retraining networks, they produce meaningful results with much less artifacts.

3.5.5 Self-Ensemble

Self-ensemble, a.k.a. enhanced prediction [44], is an inference technique commonly used by SR models. Specifically, rotations with different angles (0°, 90°, 180°, 270°) and horizontal flipping are applied on the LR images to get a set of 8 images. Then these images are fed into the SR model and the corresponding inverse transformation is applied to the reconstructed HR images to get the outputs. The final prediction result is conducted by the mean [31], [32], [44], [70], [78], [93] or the median [83] of these outputs. In this way, these models further improve performance.

3.6 State-of-the-art Super-resolution Models

In recent years, image super-resolution models based on deep learning have received more and more attention and achieved state-of-the-art performance. In previous sections, we decompose SR models into specific components, including model frameworks (Sec. 3.1), upsampling methods (Sec. 3.2), network design (Sec. 3.3) and learning strategies (Sec. 3.4), analyze these components hierarchically and identify their advantages and limitations. As a matter of fact, most of the state-of-the-art SR models today can basically be attributed to a combination of multiple strategies we summarize above. For example, the biggest contribution of the RCAN [70] comes from the channel attention mechanism (Sec. 3.3.5), and it also employs other strategies like sub-pixel upsampling (Sec. 3.2.2), residual learning (Sec. 3.3.1), pixel L1 loss (Sec. 3.4.1), and self-ensemble (Sec. 3.5.5). In similar manners, we summarize some representative models and their key strategies, as Table 2 shows.

TABLE 2
Super-resolution methodology employed by some representative models. The "Fw.", "Up.", "Rec.", "Res.", "Dense.", "Att." represent SR frameworks, upsampling methods, recursive learning, residual learning, dense connections, attention mechanism, respectively.

Method	Publication	Fw.	Up.	Rec.	Res.	Dense	Att.	$\mathcal{L}_{L1}$	$\mathcal{L}_{L2}$	Keywords
SRCNN [22]	2014, ECCV	Pre.	Bicubic						✓	
DRCN [82]	2016, CVPR	Pre.	Bicubic	✓	✓				✓	Recursive layers
FSRCNN [43]	2016, ECCV	Post.	Deconv						✓	Lightweight design
ESPCN [156]	2017, CVPR	Pre.	Sub-Pixel						✓	Sub-pixel
LapSRN [27]	2017, CVPR	Pro.	Bicubic		✓			✓		$\mathcal{L}_{\text{pixel\_Cha}}$
DRRN [56]	2017, CVPR	Pre.	Bicubic	✓	✓				✓	Recursive blocks
SRResNet [25]	2017, CVPR	Post.	Sub-Pixel		✓				✓	$\mathcal{L}_{\text{Con.}}$, $\mathcal{L}_{\text{TV}}$
SRGAN [25]	2017, CVPR	Post.	Sub-Pixel		✓					$\mathcal{L}_{\text{Con.}}$, $\mathcal{L}_{\text{GAN}}$
EDSR [31]	2017, CVPRW	Post.	Sub-Pixel		✓			✓		Compact and large-size design
EnhanceNet [8]	2017, ICCV	Pre.	Bicubic		✓					$\mathcal{L}_{\text{Con.}}$, $\mathcal{L}_{\text{GAN}}$, $\mathcal{L}_{\text{texture}}$
MemNet [55]	2017, ICCV	Pre.	Bicubic	✓	✓	✓			✓	Memory block
SRDenseNet [79]	2017, ICCV	Post.	Deconv		✓	✓			✓	Dense connections
DBPN [57]	2018, CVPR	Iter.	Deconv		✓	✓			✓	Back-projection
DSRN [85]	2018, CVPR	Pre.	Deconv	✓	✓				✓	Dual state
RDN [93]	2018, CVPR	Post.	Sub-Pixel		✓	✓		✓		Residual dense block
CARN [28]	2018, ECCV	Post.	Sub-Pixel	✓	✓	✓		✓		Cascading
MSRN [99]	2018, ECCV	Post.	Sub-Pixel		✓			✓		Multi-path
RCAN [70]	2018, ECCV	Post.	Sub-Pixel		✓		✓	✓		Channel attention
ESRGAN [103]	2018, ECCVW	Post.	Sub-Pixel		✓	✓		✓		$\mathcal{L}_{\text{Con.}}$, $\mathcal{L}_{\text{GAN}}$
RNAN [106]	2019, ICLR	Post.	Sub-Pixel		✓		✓	✓		Non-local attention
Meta-RDN [95]	2019, CVPR	Post.	Meta Upscale		✓	✓		✓		Magnification-arbitrary
SAN [105]	2019, CVPR	Post.	Sub-Pixel		✓		✓	✓		Second-order attention
SRFBN [86]	2019, CVPR	Post.	Deconv	✓	✓	✓		✓		Feedback mechanism

In addition to SR accuracy, the efficiency is another very important aspect and different strategies have more or less impact on efficiency. So in the previous sections, we not only analyze the accuracy of the presented strategies, but also indicate the concrete impacts on efficiency for the ones with a greater impact on efficiency, such as the post-upsampling (Sec. 3.1.2), recursive learning (Sec. 3.3.2), dense connections (Sec. 3.3.4), xUnit (Sec. 3.3.11). And we also benchmark some representative SR models on the SR accuracy (i.e., PSNR), model size (i.e., number of parameters) and computation cost (i.e., number of Multi-Adds), as shown in Fig. 8. The accuracy is measured by the mean of the PSNR on 4 benchmark datasets (i.e., Set5 [48], Set14 [49], B100 [40] and Urban100 [50]). And the model size and computational cost are calculated with PyTorch-OpCounter [157], where the output resolution is 720p (i.e., 1080×720). All statistics are derived from the original papers or calculated on official models, with a scaling factor of 2. For better viewing and comparison, we also provide an interactive online version[1].

4 Unsupervised Super-resolution

Existing super-resolution works mostly focus on supervised learning, i.e., learning with matched LR-HR image pairs. However, since it is difficult to collect images of the same scene but with different resolutions, the LR images in SR datasets are often obtained by performing predefined degradation on HR images. Thus the trained SR models actually learn a reverse process of the predefined degradation. In order to learn the real-world LR-HR mapping without introducing manual degradation priors, researchers pay more and more attention to unsupervised SR, in which case only unpaired LR-HR images are provided for training, so that the resulting models are more likely to cope with the SR problems in real-world scenarios. Next we'll briefly introduce several existing unsupervised SR models with deep learning, and more methods are yet to be explored.

4.1 Zero-shot Super-resolution

Considering that the internal image statistics inside a single image have provided sufficient information for SR, Shocher *et al.* [83] propose zero-shot super-resolution (ZSSR) to cope with unsupervised SR by training image-specific SR networks at test time rather than training a generic model on large external datasets. Specifically, they estimate the degradation kernel from a single image using [158] and use this kernel to build a small dataset by performing degradation with different scaling factors and augmentation on this image. Then a small CNN for SR is trained on this dataset and used for the final prediction.

In this way, the ZSSR leverages on the cross-scale internal recurrence inside every image, and thus outperforms previous approaches by a large margin (1 dB for estimated kernels and 2 dB for known kernels) on images under non-ideal conditions (i.e., images obtained by non-bicubic degradation and suffered effects like blurring, noise, compression artifacts), which is closer to the real-world scenes, while give competitive results under ideal conditions (i.e., images obtained by bicubic degradation). However, since it needs to train different networks for different images during testing, the inference time is much longer than others.

1. https://github.com/ptkin/Awesome-Super-Resolution

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.2982166, IEEE Transactions on Pattern Analysis and Machine Intelligence

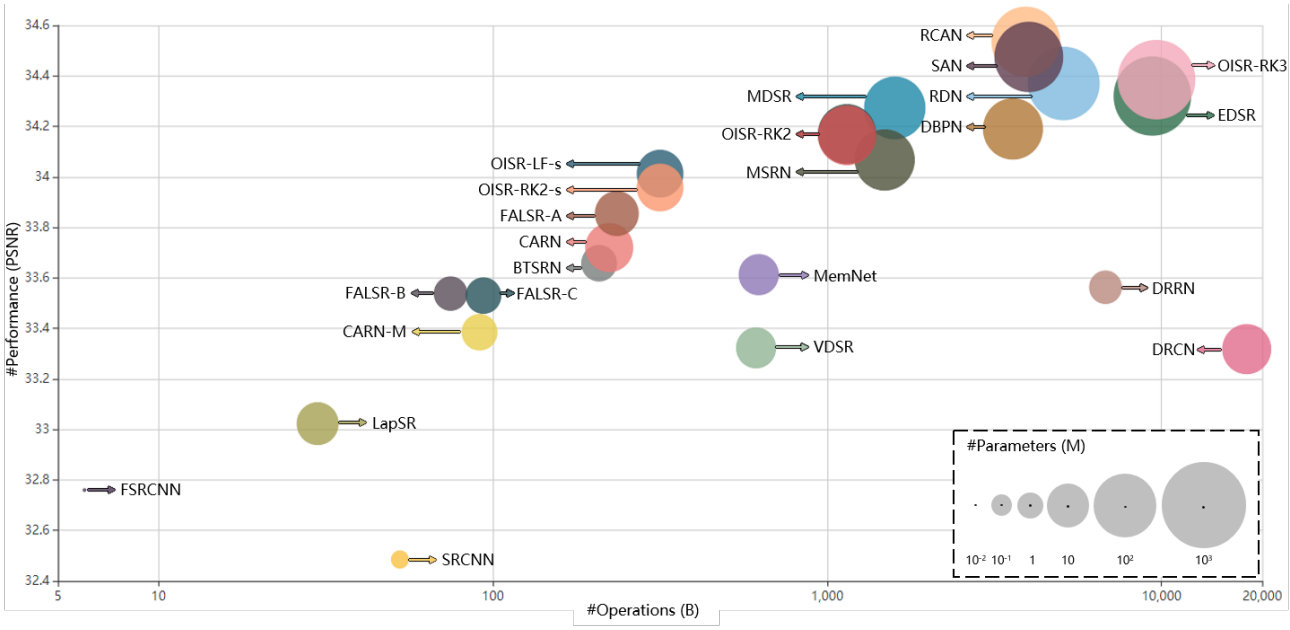


Fig. 8. Super-resolution benchmarking. The x-axis and the y-axis denote the Multi-Adds and PSNR, respectively, and the circle size represents the number of parameters.

4.2 Weakly-supervised Super-resolution

To cope with super-resolution without introducing predefined degradation, researchers attempt to learn SR models with weakly-supervised learning, i.e., using unpaired LR-HR images. Among them, some researchers first learn the HR-to-LR degradation and use it to construct datasets for training the SR model, while others design cycle-in-cycle networks to learn the LR-to-HR and HR-to-LR mappings simultaneously. Next we'll detail these models.

Learned Degradation. Since the predefined degradation is suboptimal, learning the degradation from unpaired LR-HR datasets is a feasible direction. Bulat *et al.* [159] propose a two-stage process which firstly trains an HR-to-LR GAN to learn degradation using unpaired LR-HR images and then trains an LR-to-HR GAN for SR using paired LR-HR images conducted base on the first GAN. Specifically, for the HR-to-LR GAN, HR images are fed into the generator to produce LR outputs, which are required to match not only the LR images obtained by downscaling the HR images (by average pooling) but also the distribution of real LR images. After finishing training, the generator is used as a degradation model to generate LR-HR image pairs. Then for the LR-to-HR GAN, the generator (i.e., the SR model) takes the generated LR images as input and predicts HR outputs, which are required to match not only the corresponding HR images but also the distribution of the HR images.

By applying this two-stage process, the proposed unsupervised model effectively increases the quality of super-resolving real-world LR images and obtains large improvement over previous state-of-the-art works.

Cycle-in-cycle Super-resolution. Another approach for unsupervised super-resolution is to treat the LR space and the HR space as two domains, and use a cycle-in-cycle structure to learn the mappings between each other. In this case, the training objectives include pushing the mapped results to match the target domain distribution and making the images recoverable through round-trip mappings.

Motivated by CycleGAN [138], Yuan *et al.* [131] propose a cycle-in-cycle SR network (CinCGAN) composed of 4 generators and 2 discriminators, making up two CycleGANs for *noisy LR* $\rightleftharpoons$ *clean LR* and *clean LR* $\rightleftharpoons$ *clean HR* mappings, respectively. Specifically, in the first CycleGAN, the noisy LR image is fed into a generator, and the output is required to be consistent with the distribution of real clean LR images. Then it's fed into another generator and required to recover the original input. Several loss functions (e.g., adversarial loss, cycle consistency loss, identity loss) are employed for guaranteeing the cycle consistency, distribution consistency, and mapping validity. The other CycleGAN is similarly designed, except that the mapping domains are different.

Because of avoiding the predefined degradation, the unsupervised CinCGAN not only achieves comparable performance to supervised methods, but also is applicable to various cases even under very harsh conditions. However, due to the ill-posed essence of SR problem and the complicated architecture of CinCGAN, some advanced strategies are needed for reducing the training difficulty and instability.

4.3 Deep Image Prior

Considering that the CNN structure is sufficient to capture a great deal of low-level image statistics prior for inverse problems, Ulyanov *et al.* [160] employ a randomly-initialized CNN as handcrafted prior to perform SR. Specifically, they define a generator network which takes a random vector z as input and tries to generate the target HR image I_y. The goal is to train the network to find an $\hat{I}_y$ that the downsampled $\hat{I}_y$ is identical to the LR image I_x. Since the network is randomly initialized and never trained, the only prior is the CNN structure itself. Although the performance of this method is still worse than the supervised methods

(2 dB), it outperforms traditional bicubic upsampling considerably (1 dB). Besides, it shows the rationality of the CNN architectures itself, and prompts us to improve SR by combining the deep learning methodology with handcrafted priors such as CNN structures or self-similarity.

5 Domain-Specific Applications

5.1 Depth Map Super-resolution

Depth maps record the depth (i.e., distance) between the viewpoint and objects in the scene, and plays important roles in many tasks like pose estimation [161], [162] and semantic segmentation [163], [164]. However, due to economic and production constraints, the depth maps produced by depth sensors are often low-resolution and suffer degradation effects such as noise, quantization and missing values. Thus super-resolution is introduced for increasing the spatial resolution of depth maps.

Nowadays one of the most popular practices for depth map SR is to use another economical RGB camera to obtain HR images of the same scenes for guiding super-resolving the LR depth maps. Specifically, Song *et al.* [165] exploit the depth field statistics and local correlations between depth maps and RGB images to constrain the global statistics and local structures. Hui *et al.* [166] utilize two CNNs to simultaneously upsample LR depth maps and downsample HR RGB images, then use RGB features as the guidance for upsampling depth maps with the same resolution. And Haefner *et al.* [167] further exploit the color information and guide SR by resorting to the shape-from-shading technique. In contrast, Riegler *et al.* [168] combine CNNs with an energy minimization model in the form of a powerful variational model to recover HR depth maps without other reference images.

5.2 Face Image Super-resolution

Face image super-resolution, a.k.a. face hallucination (FH), can often help other face-related tasks [72], [73], [169]. Compared to generic images, face images have more face-related structured information, so incorporating facial prior knowledge (e.g., landmarks, parsing maps, identities) into FH is a very popular and promising approach.

One of the most straightforward way is to constrain the generated images to have the identical face-related attributes to ground truth. Specifically, the CBN [170] utilizes the facial prior by alternately optimizing FH and dense correspondence field estimation. The Super-FAN [30] and MTUN [171] both introduce FAN to guarantee the consistency of facial landmarks by end-to-end multi-task learning. And the FSRNet [73] uses not only facial landmark heatmaps but also face parsing maps as prior constraints. The SICNN [72], which aims at recovering the real identity, adopts a super-identity loss function and a domain-integrated training approach to stable the joint training.

Besides explicitly using facial prior, the implicit methods are also widely studied. The TDN [172] incorporates spatial transformer networks [173] for automatic spatial transformations and thus solves the face unalignment problem. Based on TDN, the TDAE [174] adopts a decoder-encoder-decoder framework, where the first decoder learns to upsample and denoise, the encoder projects it back to aligned and noise-free LR faces, and the last decoder generates hallucinated HR images. In contrast, the LCGE [175] employs component-specific CNNs to perform SR on five facial components, uses k-NN search on an HR facial component dataset to find corresponding patches, synthesizes finer-grained components and finally fuses them to FH results. Similarly, Yang *et al.* [176] decompose deblocked face images into facial components and background, use the component landmarks to retrieve adequate HR exemplars in external datasets, perform generic SR on the background, and finally fuse them to complete HR faces.

In addition, researchers also improve FH from other perspectives. Motivated by the human attention shifting mechanism [112], the Attention-FH [113] resorts to a recurrent policy network for sequentially discovering attended face patches and performing local enhancement, and thus fully exploits the global interdependency of face images. The UR-DGN [177] adopts a network similar to SRGAN [25] with adversarial learning. And Xu *et al.* [63] propose a multi-class GAN-based FH model composed of a generic generator and class-specific discriminators. Both Lee *et al.* [178] and Yu *et al.* [179] utilize additional facial attribute information to perform FH with the specified attributes, based on the conditional GAN [180].

5.3 Hyperspectral Image Super-resolution

Compared to panchromatic images (PANs, i.e., RGB images with 3 bands), hyperspectral images (HSIs) containing hundreds of bands provide abundant spectral features and help various vision tasks [181], [182], [183]. However, due to hardware limitations, collecting high-quality HSIs is much more difficult than PANs and the resolution is also lower. Thus super-resolution is introduced into this field, and researchers tend to combine HR PANs and LR HSIs to predict HR HSIs. Among them, Masi *et al.* [184] employ SRCNN [22] and incorporate several maps of nonlinear radiometric indices for boosting performance. Qu *et al.* [185] jointly train two encoder-decoder networks to perform SR on PANs and HSIs, respectively, and transfer the SR knowledge from PAN to HSI by sharing the decoder and applying constraints such as angle similarity loss and reconstruction loss. Recently, Fu *et al.* [186] evaluate the effect of camera spectral response (CSR) functions for HSI SR and propose a CSR optimization layer which can automatically select or design the optimal CSR, and outperform the state-of-the-arts.

5.4 Real-world Image Super-resolution

Generally, the LR images for training SR models are generated by downsampling RGB images manually (e.g., by bicubic downsampling). However, real-world cameras actually capture 12-bit or 14-bit RAW images, and performs a series of operations (e.g., demosaicing, denoising and compression) through camera ISPs (image signal processors) and finally produce 8-bit RGB images. Through this process, the RGB images have lost lots of original signals and are very different from the original images taken by the camera. Therefore, it is suboptimal to directly use the manually downsampled RGB image for SR.

To solve this problem, researchers study how to use real-world images for SR. Among them, Chen *et al.* [187] analyze

the relationships between image resolution (R) and field-of-view (V) in imaging systems (namely R-V degradation), propose data acquisition strategies to conduct a real-world dataset City100, and experimentally demonstrate the superiority of the proposed image synthesis model. Zhang *et al.* [188] build another real-world image dataset SR-RAW (i.e., paired HR RAW images and LR RGB images) through optical zoom of cameras, and propose contextual bilateral loss to solve the misalignment problem. In contrast, Xu *et al.* [189] propose a pipeline to generate realistic training data by simulating the imaging process and develop a dual CNN to exploit the originally captured radiance information in RAW images. They also propose to learn a spatially-variant color transformation for effective color corrections and generalization to other sensors.

5.5 Video Super-resolution

For video super-resolution, multiple frames provide much more scene information, and there are not only intra-frame spatial dependency but also inter-frame temporal dependency (e.g., motions, brightness and color changes). Thus the existing works mainly focus on making better use of spatio-temporal dependency, including explicit motion compensation (e.g., optical flow-based, learning-based) and recurrent methods, etc.

Among the optical flow-based methods, Liao *et al.* [190] employ optical flow methods to generate HR candidates and ensemble them by CNNs. VSRnet [191] and CVSRnet [192] deal with motion compensation by Druleas algorithm [193], and uses CNNs to take successive frames as input and predict HR frames. While Liu *et al.* [194], [195] perform rectified optical flow alignment, and propose a temporal adaptive net to generate HR frames in various temporal scales and aggregate them adaptively.

Besides, others also try to directly learn the motion compensation. The VESPCN [156] utilizes a trainable spatial transformer [173] to learn motion compensation based on adjacent frames, and enters multiple frames into a spatio-temporal ESPCN [84] for end-to-end prediction. And Tao *et al.* [196] root from accurate LR imaging model and propose a sub-pixel-like module to simultaneously achieve motion compensation and super-resolution, and thus fuse the aligned frames more effectively.

Another trend is to use recurrent methods to capture the spatial-temporal dependency without explicit motion compensation. Specifically, the BRCN [197], [198] employs a bidirectional framework, and uses CNN, RNN, and conditional CNN to model the spatial, temporal and spatial-temporal dependency, respectively. Similarly, STCN [199] uses a deep CNN and a bidirectional LSTM [200] to extract spatial and temporal information. And FRVSR [201] uses previously inferred HR estimates to reconstruct the subsequent HR frames by two deep CNNs in a recurrent manner. Recently the FSTRN [202] employs two much smaller 3D convolution filters to replace the original large filter, and thus enhances the performance through deeper CNNs while maintaining low computational cost. While the RBPN [87] extracts spatial and temporal contexts by a recurrent encoder-decoder, and combines them with an iterative refinement framework based on the back-projection mechanism (Sec. 3.1.4).

In addition, the FAST [203] exploits compact descriptions of the structure and pixel correlations extracted by compression algorithms, transfers the SR results from one frame to adjacent frames, and much accelerates the state-of-the-art SR algorithms with little performance loss. And Jo *et al.* [204] generate dynamic upsampling filters and the HR residual image based on the local spatio-temporal neighborhoods of each pixel, and also avoid explicit motion compensation.

5.6 Other Applications

Deep learning based super-resolution is also adopted to other domain-specific applications and shows great performance. Specifically, the Perceptual GAN [205] addresses the small object detection problem by super-resolving representations of small objects to have similar characteristics as large objects and be more discriminative for detection. Similarly, the FSR-GAN [206] super-resolves small-size images in the feature space instead of the pixel space, and thus transforms the raw poor features to highly discriminative ones, which greatly benefits image retrieval. Besides, Jeon *et al.* [207] utilize a parallax prior in stereo images to reconstruct HR images with sub-pixel accuracy in registration. Wang *et al.* [208] propose a parallax-attention model to tackle the stereo image super-resolution problem. Li *et al.* [209] incorporate the 3D geometric information and super-resolve 3D object texture maps. And Zhang *et al.* [210] separate view images in one light field into groups, learn inherent mapping for every group and finally combine the residuals in every group to reconstruct higher-resolution light fields. All in all, super-resolution technology can play an important role in all kinds of applications, especially when we can deal with large objects well but cannot handle small objects.

6 Conclusion and Future Directions

In this paper, we have given an extensive survey on recent advances in image super-resolution with deep learning. We mainly discussed the improvement of supervised and unsupervised SR, and also introduced some domain-specific applications. Despite great success, there are still many unsolved problems. Thus in this section, we will point out these problems explicitly and introduce some promising trends for future evolution. We hope that this survey not only provides a better understanding of image SR for researchers but also facilitates future research activities and application developments in this field.

6.1 Network Design

Good network design not only determines a hypothesis space with great performance upper bound, but also helps efficiently learn representations without excessive spatial and computational redundancy. Below we will introduce some promising directions for network improvements.

Combining Local and Global Information. Large receptive field provides more contextual information and helps generate more realistic results. Thus it is promising to combine local and global information for providing contextual information of different scales for image SR.

Combining Low- and High-level Information. Shallow layers in CNNs tend to extract low-level features like colors and

edges, while deeper layers learn higher-level representations like object identities. Thus combining low-level details with high-level semantics can be of great help for HR reconstruction.

Context-specific Attention. In different contexts, people tend to care about different aspects of the images. For example, for the grass area people may be more concerned with local colors and textures, while in the animal body area people may care more about the species and corresponding hair details. Therefore, incorporating attention mechanism to enhance the attention to key features facilitates the generation of realistic details.

More Efficient Architectures. Existing SR modes tend to pursue ultimate performance, while ignoring the model size and inference speed. For example, the EDSR [31] takes 20s per image for $4\times$ SR on DIV2K [42] with a Titan GTX GPU [80], and DBPN [57] takes 35s for $8\times$ SR [211]. Such long prediction time is unacceptable in practical applications, thus more efficient architectures are imperative. How to reduce model sizes and speed up prediction while maintaining performance remains a problem.

Upsampling Methods. Existing upsampling methods (Sec. 3.2) have more or less disadvantages: interpolation methods result in expensive computation and cannot be end-to-end learned, the transposed convolution produces checkerboard artifacts, the sub-pixel layer brings uneven distribution of receptive fields, and the meta upscale module may cause instability or inefficiency and have further room for improvement. How to perform effective and efficient upsampling still needs to be studied, especially with high scaling factors.

Recently, the neural architecture search (NAS) technique for deep learning has become more and more popular, greatly improving the performance or efficiency with little artificial intervention [212], [213], [214]. For the SR field, combining the exploration of the above directions with NAS is of great potential.

6.2 Learning Strategies

Besides good hypothesis spaces, robust learning strategies are also needed for achieving satisfactory results. Next we'll introduce some promising directions of learning strategies.

Loss Functions. Existing loss functions can be regarded as establishing constraints among LR/HR/SR images, and guide optimization based on whether these constraints are met. In practice, these loss functions are often weighted combined and the best loss function for SR is still unclear. Therefore, one of the most promising directions is to explore the potential correlations between these images and seek more accurate loss functions.

Normalization. Although BN is widely used in vision tasks, which greatly speeds up training and improves performance, it is proven to be sub-optimal for super-resolution [31], [32], [147]. Thus other effective normalization techniques for SR are needed to be studied.

6.3 Evaluation Metrics

Evaluation metrics are one of the most fundamental components for machine learning. If the performance cannot be measured accurately, researchers will have great difficulty verifying improvements. Metrics for super-resolution face such challenges and need more exploration.

More Accurate Metrics. Nowadays the PSNR and SSIM have been the most widely used metrics for SR. However, the PSNR tends to result in excessive smoothness and the results can vary wildly between almost indistinguishable images. The SSIM [58] performs evaluation in terms of brightness, contrast and structure, but still cannot measure perceptual quality accurately [8], [25]. Besides, the MOS is the closest to human visual response, but needs to take a lot of efforts and is non-reproducible. Although researchers have proposed various metrics (Sec. 2.3), but currently there is no unified and admitted evaluation metrics for SR quality. Thus more accurate metrics for evaluating reconstruction quality are urgently needed.

Blind IQA Methods. Today most metrics used for SR are all-reference methods, i.e., assuming that we have paired LR-HR images with perfect quality. But since it's difficult to obtain such datasets, the commonly used datasets for evaluation are often conducted by manual degradation. In this case, the task we perform evaluation on is actually the inverse process of the predefined degradation. Therefore, developing blind IQA methods also has great demands.

6.4 Unsupervised Super-resolution

As mentioned in Sec. 4, it is often difficult to collect images with different resolutions of the same scene, so bicubic interpolation is widely used for constructing SR datasets. However, the SR models trained on these datasets may only learn the inverse process of the predefined degradation. Therefore, how to perform unsupervised super-resolution (i.e., trained on datasets without paired LR-HR images) is a promising direction for future development.

6.5 Towards Real-world Scenarios

Image super-resolution is greatly limited in real-world scenarios, such as suffering unknown degradation, missing paired LR-HR images. Below we'll introduce some directions towards real-world scenarios.

Dealing with Various Degradation. Real-world images tend to suffer degradation like blurring, additive noise and compression artifacts. Thus the models trained on datasets conducted manually often perform poorly in real-world scenes. Some works have been proposed for solving this [39], [131], [149], [159], but these methods have some inherent drawbacks, such as great training difficulty and over-perfect assumptions. This issue is urgently needed to be resolved.

Domain-specific Applications. Super-resolution can not only be used in domain-specific data and scenes directly, but also help other vision tasks greatly (Sec. 5). Therefore, it is also a promising direction to apply SR to more specific domains, such as video surveillance, object tracking, medical imaging and scene rendering.

Acknowledgment

Prof. Jian Chen is supported by the Guangdong special branch plans young talent with scientific and technological innovation (Grant No. 2016TQ03X445), the Guangzhou science and technology planning project (Grant No. 201904010197) and Natural Science Foundation of Guangdong Province, China (2016A030313437).

REFERENCES

[1] H. Greenspan, "Super-resolution in medical imaging," *The Computer Journal*, vol. 52, 2008.

[2] J. S. Isaac and R. Kulkarni, "Super resolution techniques for medical image processing," in *ICTSD*, 2015.

[3] Y. Huang, L. Shao, and A. F. Frangi, "Simultaneous super-resolution and cross-modality synthesis of 3d medical images using weakly-supervised joint convolutional sparse coding," in *CVPR*, 2017.

[4] L. Zhang, H. Zhang, H. Shen, and P. Li, "A super-resolution reconstruction algorithm for surveillance images," *Elsevier Signal Processing*, vol. 90, 2010.

[5] P. Rasti, T. Uiboupin, S. Escalera, and G. Anbarjafari, "Convolutional neural network super resolution for face recognition in surveillance monitoring," in *AMDO*, 2016.

[6] D. Dai, Y. Wang, Y. Chen, and L. Van Gool, "Is image super-resolution helpful for other vision tasks?" in *WACV*, 2016.

[7] M. Haris, G. Shakhnarovich, and N. Ukita, "Task-driven super resolution: Object detection in low-resolution images," *Arxiv:1803.11316*, 2018.

[8] M. S. Sajjadi, B. Schölkopf, and M. Hirsch, "Enhancenet: Single image super-resolution through automated texture synthesis," in *ICCV*, 2017.

[9] Y. Zhang, Y. Bai, M. Ding, and B. Ghanem, "Sod-mtgan: Small object detection via multi-task generative adversarial network," in *ECCV*, 2018.

[10] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, 1981.

[11] C. E. Duchon, "Lanczos filtering in one and two dimensions," *Journal of Applied Meteorology*, vol. 18, 1979.

[12] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models and Image Processing*, vol. 53, 1991.

[13] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," *TOG*, vol. 30, 2011.

[14] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *CVPR*, 2008.

[15] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *TPAMI*, vol. 32, 2010.

[16] Z. Xiong, X. Sun, and F. Wu, "Robust web image/video super-resolution," *IEEE Transactions on Image Processing*, vol. 19, 2010.

[17] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, vol. 22, 2002.

[18] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *CVPR*, 2004.

[19] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *ICCV*, 2009.

[20] Y. Jianchao, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *CVPR*, 2008.

[21] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, 2010.

[22] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *ECCV*, 2014.

[23] ——, "Image super-resolution using deep convolutional networks," *TPAMI*, vol. 38, 2016.

[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014.

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

[26] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *CVPR*, 2016.

[27] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate superresolution," in *CVPR*, 2017.

[28] N. Ahn, B. Kang, and K.-A. Sohn, "Fast, accurate, and lightweight super-resolution with cascading residual network," in *ECCV*, 2018.

[29] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *ECCV*, 2016.

[30] A. Bulat and G. Tzimiropoulos, "Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans," in *CVPR*, 2018.

[31] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *CVPRW*, 2017.

[32] Y. Wang, F. Perazzi, B. McWilliams, A. Sorkine-Hornung, O. Sorkine-Hornung, and C. Schroers, "A fully progressive approach to single-image super-resolution," in *CVPRW*, 2018.

[33] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Processing Magazine*, vol. 20, 2003.

[34] K. Nasrollahi and T. B. Moeslund, "Super-resolution: A comprehensive survey," *Machine Vision and Applications*, vol. 25, 2014.

[35] J. Tian and K.-K. Ma, "A survey on super-resolution imaging," *Signal, Image and Video Processing*, vol. 5, 2011.

[36] J. Van Ouwerkerk, "Image super-resolution survey," *Image and Vision Computing*, vol. 24, 2006.

[37] C.-Y. Yang, C. Ma, and M.-H. Yang, "Single-image super-resolution: A benchmark," in *ECCV*, 2014.

[38] D. Thapa, K. Raahemifar, W. R. Bobier, and V. Lakshminarayanan, "A performance comparison among different super-resolution techniques," *Computers & Electrical Engineering*, vol. 54, 2016.

[39] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *CVPR*, 2018.

[40] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *ICCV*, 2001.

[41] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *TPAMI*, vol. 33, 2011.

[42] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *CVPRW*, 2017.

[43] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *ECCV*, 2016.

[44] R. Timofte, R. Rothe, and L. Van Gool, "Seven ways to improve example-based single image super resolution," in *CVPR*, 2016.

[45] A. Fujimoto, T. Ogawa, K. Yamamoto, Y. Matsui, T. Yamasaki, and K. Aizawa, "Manga109 dataset and creation of metadata," in *MANPU*, 2016.

[46] X. Wang, K. Yu, C. Dong, and C. C. Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," 2018.

[47] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor, "2018 pirm challenge on perceptual image super-resolution," in *ECCV Workshop*, 2018.

[48] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on non-negative neighbor embedding," in *BMVC*, 2012.

[49] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International Conference on Curves and Surfaces*, 2010.

[50] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *CVPR*, 2015.

[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[52] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[53] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *IJCV*, vol. 111, 2015.

[54] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *ICCV*, 2015.

[55] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: A persistent memory network for image restoration," in *ICCV*, 2017.

[56] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *CVPR*, 2017.

[57] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep backp-rojection networks for super-resolution," in *CVPR*, 2018.

[58] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, 2004.

[59] Z. Wang, A. C. Bovik, and L. Lu, "Why is image quality assessment so difficult?" in *ICASSP*, 2002.

[60] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, vol. 15, 2006.

[61] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *IEEE Signal Processing Magazine*, vol. 26, 2009.

[62] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *ICCV*, 2015.

[63] X. Xu, D. Sun, J. Pan, Y. Zhang, H. Pfister, and M.-H. Yang, "Learning to super-resolve blurry face and text images," in *ICCV*, 2017.

[64] R. Dahl, M. Norouzi, and J. Shlens, "Pixel recursive super resolution," in *ICCV*, 2017.

[65] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Fast and accurate image super-resolution with deep laplacian pyramid networks," *TPAMI*, 2018.

[66] C. Ma, C.-Y. Yang, X. Yang, and M.-H. Yang, "Learning a no-reference quality metric for single-image super-resolution," *Computer Vision and Image Understanding*, 2017.

[67] H. Talebi and P. Milanfar, "Nima: Neural image assessment," *IEEE Transactions on Image Processing*, vol. 27, 2018.

[68] J. Kim and S. Lee, "Deep learning of human visual sensitivity in image quality assessment framework," in *CVPR*, 2017.

[69] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018.

[70] 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*, 2018.

[71] C. Fookes, F. Lin, V. Chandran, and S. Sridharan, "Evaluation of image resolution and super-resolution on face recognition performance," *Journal of Visual Communication and Image Representation*, vol. 23, 2012.

[72] K. Zhang, Z. ZHANG, C.-W. Cheng, W. Hsu, Y. Qiao, W. Liu, and T. Zhang, "Super-identity convolutional neural network for face hallucination," in *ECCV*, 2018.

[73] Y. Chen, Y. Tai, X. Liu, C. Shen, and J. Yang, "Fsrnet: End-to-end learning face super-resolution with facial priors," in *CVPR*, 2018.

[74] Z. Wang, E. Simoncelli, A. Bovik *et al.*, "Multi-scale structural similarity for image quality assessment," in *Asilomar Conference on Signals, Systems, and Computers*, 2003.

[75] L. Zhang, L. Zhang, X. Mou, D. Zhang *et al.*, "Fsim: a feature similarity index for image quality assessment," *IEEE transactions on Image Processing*, vol. 20, 2011.

[76] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a completely blind image quality analyzer," *IEEE Signal Processing Letters*, 2013.

[77] Y. Blau and T. Michaeli, "The perception-distortion tradeoff," in *CVPR*, 2018.

[78] X. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in *NIPS*, 2016.

[79] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *ICCV*, 2017.

[80] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee *et al.*, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *CVPRW*, 2017.

[81] A. Ignatov, R. Timofte, T. Van Vu, T. Minh Luu, T. X Pham, C. Van Nguyen, Y. Kim, J.-S. Choi, M. Kim, J. Huang *et al.*, "Pirm challenge on perceptual image enhancement on smartphones: report," in *ECCV Workshop*, 2018.

[82] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *CVPR*, 2016.

[83] A. Shocher, N. Cohen, and M. Irani, "zero-shot super-resolution using deep internal learning," in *CVPR*, 2018.

[84] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *CVPR*, 2016.

[85] W. Han, S. Chang, D. Liu, M. Yu, M. Witbrock, and T. S. Huang, "Image super-resolution via dual-state recurrent networks," in *CVPR*, 2018.

[86] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, "Feedback network for image super-resolution," in *CVPR*, 2019.

[87] M. Haris, G. Shakhnarovich, and N. Ukita, "Recurrent back-projection network for video super-resolution," in *CVPR*, 2019.

[88] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *ACCV*, 2014.

[89] S. Schulter, C. Leistner, and H. Bischof, "Fast and accurate image upscaling with super-resolution forests," in *CVPR*, 2015.

[90] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *CVPRW*, 2010.

[91] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *ECCV*, 2014.

[92] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, 2016.

[93] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *CVPR*, 2018.

[94] H. Gao, H. Yuan, Z. Wang, and S. Ji, "Pixel transposed convolutional networks," *TPAMI*, 2019.

[95] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, "Meta-sr: A magnification-arbitrary network for super-resolution," in *CVPR*, 2019.

[96] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[97] R. Timofte, V. De Smet, and L. Van Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *ICCV*, 2013.

[98] Z. Hui, X. Wang, and X. Gao, "Fast and accurate single image super-resolution via information distillation network," in *CVPR*, 2018.

[99] J. Li, F. Fang, K. Mei, and G. Zhang, "Multi-scale residual network for image super-resolution," in *ECCV*, 2018.

[100] H. Ren, M. El-Khamy, and J. Lee, "Image super resolution based on fusing multiple convolution neural networks," in *CVPRW*, 2017.

[101] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015.

[102] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *CVPR*, 2017.

[103] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang, "Esrgan: Enhanced super-resolution generative adversarial networks," in *ECCV Workshop*, 2018.

[104] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018.

[105] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *CVPR*, 2019.

[106] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, "Residual non-local attention networks for image restoration," *ICLR*, 2019.

[107] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *CVPR*, 2017.

[108] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *CVPR*, 2017.

[109] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *CVPR*, 2017.

[110] 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:1704.04861*, 2017.

[111] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves *et al.*, "Conditional image generation with pixelcnn decoders," in *NIPS*, 2016.

[112] J. Najemnik and W. S. Geisler, "Optimal eye movement strategies in visual search," *Nature*, vol. 434, 2005.

[113] Q. Cao, L. Lin, Y. Shi, X. Liang, and G. Li, "Attention-aware face hallucination via deep reinforcement learning," in *CVPR*, 2017.

[114] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *ECCV*, 2014.

[115] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *CVPR*, 2017.

[116] D. Park, K. Kim, and S. Y. Chun, "Efficient module based single image super resolution for multiple problems," in *CVPRW*, 2018.

[117] I. Daubechies, *Ten lectures on wavelets*. SIAM, 1992.

[118] S. Mallat, *A wavelet tour of signal processing*. Elsevier, 1999.

[119] W. Bae, J. J. Yoo, and J. C. Ye, "Beyond deep residual learning for image restoration: Persistent homology-guided manifold simplification," in *CVPRW*, 2017.

[120] T. Guo, H. S. Mousavi, T. H. Vu, and V. Monga, "Deep wavelet prediction for image super-resolution," in *CVPRW*, 2017.

[121] H. Huang, R. He, Z. Sun, T. Tan *et al.*, "Wavelet-srnet: A wavelet-based cnn for multi-scale face super resolution," in *ICCV*, 2017.

[122] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level wavelet-cnn for image restoration," in *CVPRW*, 2018.

[123] T. Vu, C. Van Nguyen, T. X. Pham, T. M. Luu, and C. D. Yoo, "Fast and efficient image quality enhancement via desubpixel convolutional neural networks," in *ECCV Workshop*, 2018.

[124] I. Kligvasser, T. Rott Shaham, and T. Michaeli, "xunit: Learning a spatial activation function for efficient image restoration," in *CVPR*, 2018.

[125] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/kanade meets horn/schunck: Combining local and global optic flow methods," *IJCV*, vol. 61, 2005.

[126] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Transactions on Computational Imaging*, vol. 3, 2017.

[127] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," in *NIPS*, 2016.

[128] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[129] L. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," in *NIPS*, 2015.

[130] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *CVPR*, 2016.

[131] Y. Yuan, S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. Lin, "Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks," in *CVPRW*, 2018.

[132] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *ICCV*, 2017.

[133] S.-J. Park, H. Son, S. Cho, K.-S. Hong, and S. Lee, "Srfeat: Single image super resolution with feature discrimination," in *ECCV*, 2018.

[134] A. Jolicoeur-Martineau, "The relativistic discriminator: a key element missing from standard gan," *Arxiv:1807.00734*, 2018.

[135] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *ICML*, 2017.

[136] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *NIPS*, 2017.

[137] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," in *ICLR*, 2018.

[138] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *ICCV*, 2017.

[139] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, 1992.

[140] H. A. Aly and E. Dubois, "Image up-sampling using total-variation regularization with a new observation model," *IEEE Transactions on Image Processing*, vol. 14, 2005.

[141] Y. Guo, Q. Chen, J. Chen, J. Huang, Y. Xu, J. Cao, P. Zhao, and M. Tan, "Dual reconstruction nets for image super-resolution with gradient sensitive loss," *arXiv:1809.07099*, 2018.

[142] S. Vasu, N. T. Madam *et al.*, "Analyzing perception-distortion tradeoff using enhanced perceptual super-resolution network," in *ECCV Workshop*, 2018.

[143] M. Cheon, J.-H. Kim, J.-H. Choi, and J.-S. Lee, "Generative adversarial network-based image super-resolution using perceptual content losses," in *ECCV Workshop*, 2018.

[144] J.-H. Choi, J.-H. Kim, M. Cheon, and J.-S. Lee, "Deep learning-based image super-resolution considering quantitative and perceptual quality," in *ECCV Workshop*, 2018.

[145] I. Sergey and S. Christian, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[146] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, "Amortised map inference for image super-resolution," in *ICLR*, 2017.

[147] R. Chen, Y. Qu, K. Zeng, J. Guo, C. Li, and Y. Xie, "Persistent memory residual network for single image super resolution," in *CVPRW*, 2018.

[148] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *ICML*, 2009.

[149] Y. Bei, A. Damian, S. Hu, S. Menon, N. Ravi, and C. Rudin, "New techniques for preserving global structure and denoising with low information loss in single-image super-resolution," in *CVPRW*, 2018.

[150] N. Ahn, B. Kang, and K.-A. Sohn, "Image super-resolution via progressive cascading residual network," in *CVPRW*, 2018.

[151] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *ICLR*, 2018.

[152] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, 1997.

[153] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

[154] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *ECCV*, 2014.

[155] X. Wang, K. Yu, C. Dong, X. Tang, and C. C. Loy, "Deep network interpolation for continuous imagery effect transition," in *CVPR*, 2019.

[156] J. Caballero, C. Ledig, A. P. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *CVPR*, 2017.

[157] L. Zhu, "pytorch-opcounter," https://github.com/Lyken17/pytorch-OpCounter, 2019.

[158] T. Michaeli and M. Irani, "Nonparametric blind super-resolution," in *ICCV*, 2013.

[159] A. Bulat, J. Yang, and G. Tzimiropoulos, "To learn image super-resolution, use a gan to learn how to do image degradation first," in *ECCV*, 2018.

[160] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *CVPR*, 2018.

[161] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *CVPR*, 2011.

[162] G. Moon, J. Yong Chang, and K. Mu Lee, "V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map," in *CVPR*, 2018.

[163] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from rgb-d images for object detection and segmentation," in *ECCV*, 2014.

[164] W. Wang and U. Neumann, "Depth-aware cnn for rgb-d segmentation," in *ECCV*, 2018.

[165] X. Song, Y. Dai, and X. Qin, "Deep depth super-resolution: Learning depth super-resolution using deep convolutional neural network," in *ACCV*, 2016.

[166] T.-W. Hui, C. C. Loy, and X. Tang, "Depth map super-resolution by deep multi-scale guidance," in *ECCV*, 2016.

[167] B. Haefner, Y. Quéau, T. Möllenhoff, and D. Cremers, "Fight ill-posedness with ill-posedness: Single-shot variational depth super-resolution from shading," in *CVPR*, 2018.

[168] G. Riegler, M. Rüther, and H. Bischof, "Atgv-net: Accurate depth super-resolution," in *ECCV*, 2016.

[169] J.-S. Park and S.-W. Lee, "An example-based face hallucination method for single-frame, low-resolution facial images," *IEEE Transactions on Image Processing*, vol. 17, 2008.

[170] S. Zhu, S. Liu, C. C. Loy, and X. Tang, "Deep cascaded bi-network for face hallucination," in *ECCV*, 2016.

[171] X. Yu, B. Fernando, B. Ghanem, F. Porikli, and R. Hartley, "Face super-resolution guided by facial component heatmaps," in *ECCV*, 2018.

[172] X. Yu and F. Porikli, "Face hallucination with tiny unaligned images by transformative discriminative neural networks," in *AAAI*, 2017.

[173] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *NIPS*, 2015.

[174] X. Yu and F. Porikli, "Hallucinating very low-resolution unaligned and noisy face images by transformative discriminative autoencoders," in *CVPR*, 2017.

[175] Y. Song, J. Zhang, S. He, L. Bao, and Q. Yang, "Learning to hallucinate face images via component generation and enhancement," in *IJCAI*, 2017.

[176] C.-Y. Yang, S. Liu, and M.-H. Yang, "Hallucinating compressed face images," *IJCV*, vol. 126, 2018.

[177] X. Yu and F. Porikli, "Ultra-resolving face images by discriminative generative networks," in *ECCV*, 2016.

[178] C.-H. Lee, K. Zhang, H.-C. Lee, C.-W. Cheng, and W. Hsu, "Attribute augmented convolutional neural network for face hallucination," in *CVPRW*, 2018.

[179] X. Yu, B. Fernando, R. Hartley, and F. Porikli, "Super-resolving very low-resolution face images with supplementary attributes," in *CVPR*, 2018.

[180] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *Arxiv:1411.1784*, 2014.

[181] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proceedings of the IEEE*, vol. 101, 2013.

[182] Y. Fu, Y. Zheng, I. Sato, and Y. Sato, "Exploiting spectral-spatial correlation for coded hyperspectral image restoration," in *CVPR*, 2016.

[183] B. Uzkent, A. Rangnekar, and M. J. Hoffman, "Aerial vehicle tracking by adaptive fusion of hyperspectral likelihood maps," in *CVPRW*, 2017.

[184] G. Masi, D. Cozzolino, L. Verdoliva, and G. Scarpa, "Pansharpening by convolutional neural networks," *Remote Sensing*, vol. 8, 2016.

[185] Y. Qu, H. Qi, and C. Kwan, "Unsupervised sparse dirichlet-net for hyperspectral image super-resolution," in *CVPR*, 2018.

[186] Y. Fu, T. Zhang, Y. Zheng, D. Zhang, and H. Huang, "Hyperspectral image super-resolution with optimized rgb guidance," in *CVPR*, 2019.

[187] C. Chen, Z. Xiong, X. Tian, Z.-J. Zha, and F. Wu, "Camera lens super-resolution," in *CVPR*, 2019.

[188] X. Zhang, Q. Chen, R. Ng, and V. Koltun, "Zoom to learn, learn to zoom," in *CVPR*, 2019.

[189] X. Xu, Y. Ma, and W. Sun, "Towards real scene super-resolution with raw images," in *CVPR*, 2019.

[190] R. Liao, X. Tao, R. Li, Z. Ma, and J. Jia, "Video super-resolution via deep draft-ensemble learning," in *ICCV*, 2015.

[191] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, 2016.

[192] ——, "Super-resolution of compressed videos using convolutional neural networks," in *ICIP*, 2016.

[193] M. Drulea and S. Nedevschi, "Total variation regularization of local-global optical flow," in *ITSC*, 2011.

[194] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang, "Robust video super-resolution with learned temporal dynamics," in *ICCV*, 2017.

[195] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, X. Wang, and T. S. Huang, "Learning temporal dynamics for video super-resolution: A deep learning approach," *IEEE Transactions on Image Processing*, vol. 27, 2018.

[196] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *ICCV*, 2017.

[197] Y. Huang, W. Wang, and L. Wang, "Bidirectional recurrent convolutional networks for multi-frame super-resolution," in *NIPS*, 2015.

[198] ——, "Video super-resolution via bidirectional recurrent convolutional networks," *TPAMI*, vol. 40, 2018.

[199] J. Guo and H. Chao, "Building an end-to-end spatial-temporal convolutional network for video super-resolution," in *AAAI*, 2017.

[200] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional lstm networks for improved phoneme classification and recognition," in *ICANN*, 2005.

[201] M. S. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-recurrent video super-resolution," in *CVPR*, 2018.

[202] S. Li, F. He, B. Du, L. Zhang, Y. Xu, and D. Tao, "Fast spatio-temporal residual network for video super-resolution," in *CVPR*, 2019.

[203] Z. Zhang and V. Sze, "Fast: A framework to accelerate super-resolution processing on compressed videos," in *CVPRW*, 2017.

[204] Y. Jo, S. W. Oh, J. Kang, and S. J. Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *CVPR*, 2018.

[205] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *CVPR*, 2017.

[206] W. Tan, B. Yan, and B. Bare, "Feature super-resolution: Make machine see more clearly," in *CVPR*, 2018.

[207] D. S. Jeon, S.-H. Baek, I. Choi, and M. H. Kim, "Enhancing the spatial resolution of stereo images using a parallax prior," in *CVPR*, 2018.

[208] L. Wang, Y. Wang, Z. Liang, Z. Lin, J. Yang, W. An, and Y. Guo, "Learning parallax attention for stereo image super-resolution," in *CVPR*, 2019.

[209] Y. Li, V. Tsiminaki, R. Timofte, M. Pollefeys, and L. V. Gool, "3d appearance super-resolution with deep learning," in *CVPR*, 2019.

[210] S. Zhang, Y. Lin, and H. Sheng, "Residual networks for light field image super-resolution," in *CVPR*, 2019.

[211] C. Ancuti, C. O. Ancuti, R. Timofte, L. Van Gool, L. Zhang, M.-H. Yang, V. M. Patel, H. Zhang, V. A. Sindagi, R. Zhao *et al.*, "Ntire 2018 challenge on image dehazing: Methods and results," in *CVPRW*, 2018.

[212] H. Pham, M. Y. Guan, B. Zoph, Q. V. Le, and J. Dean, "Efficient neural architecture search via parameter sharing," in *ICML*, 2018.

[213] H. Liu, K. Simonyan, and Y. Yang, "Darts: Differentiable architecture search," *ICLR*, 2019.

[214] Y. Guo, Y. Zheng, M. Tan, Q. Chen, J. Chen, P. Zhao, and J. Huang, "Nat: Neural architecture transformer for accurate and compact architectures," in *NIPS*, 2019, pp. 735–747.

Zhihao Wang received the BE degree in South China University of Technology (SCUT), China, in 2017, and is working toward the ME degree at the School of Software Engineering, SCUT. Now he is as a visiting student at the School of Information Systems, Singapore Management University, Singapore. His research interests are computer vision based on deep learning, including visual recognition and image super-resolution.

Jian Chen is currently a Professor of the School of Software Engineering at South China University of Technology where she started as an Assistant Professor in 2005. She received her B.S. and Ph.D. degrees, both in Computer Science, from Sun Yat-Sen University, China, in 2000 and 2005 respectively. Her research interests can be summarized as developing effective and efficient data analysis techniques for complex data and the related applications.

Steven C. H. Hoi is currently the Managing Director of Salesforce Research Asia, and an Associate Professor (on leave) of the School of Information Systems, Singapore Management University, Singapore. Prior to joining SMU, he was an Associate Professor with Nanyang Technological University, Singapore. He received his Bachelor degree from Tsinghua University, P.R. China, in 2002, and his Ph.D degree in computer science and engineering from The Chinese University of Hong Kong, in 2006. His research interests are machine learning and data mining and their applications to multimedia information retrieval (image and video retrieval), social media and web mining, and computational finance, etc., and he has published over 150 refereed papers in top conferences and journals in these related areas. He has served as the Editor-in-Chief for Neurocomputing Journal, general co-chair for ACM SIGMM Workshops on Social Media (WSM'09, WSM'10, WSM'11), program co-chair for the fourth Asian Conference on Machine Learning (ACML'12), book editor for "Social Media Modeling and Computing", guest editor for ACM Transactions on Intelligent Systems and Technology (ACM TIST), technical PC member for many international conferences, and external reviewer for many top journals and worldwide funding agencies, including NSF in US and RGC in Hong Kong. He is an IEEE Fellow and ACM Distinguished Member.