# 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. In this survey, we aim to give a survey on recent advances of image super-resolution techniques using deep learning approaches in a systematic way. 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], [6]), amongst others. Other than improving image perceptual quality, it also helps to improve other computer vision tasks [7], [8], [9], [10], [11]. 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 [12], [13], [14], edge-based methods [15], [16], statistical methods [17], [18], patch-based methods [15], [19], [20], [21] and sparse representation methods [22], [23], 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 [24], [25]) to recent promising SR approaches using Generative Adversarial Nets (GAN) [26] (e.g., SRGAN [27]). 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 [28], [29], [30], different types of loss functions [10], [31], [32], different types of learning principles and strategies [10], [33], [34], etc.


- *Corresponding author: Steven C.H. Hoi is with the School of Information Systems, Singapore Management University, Singapore. Email: chhoi@smu.edu.sg.*
- *This work was done when Zhihao Wang was a visiting student at the School of Information Systems, Singapore Management University, Singapore. Z. Wang is with the South China University of Technology, China. E-mail: zhwang@smu.edu.sg*
- *J. Chen is with the South China University of Technology, China. E-mail: ellachen@scut.edu.cn.*


In this paper, we give a comprehensive overview of recent advances in image super-resolution with deep learning. Although there are some existing surveys of super-resolution in literature, our work differs in that we are focused in deep learning based SR techniques, while most of the earlier works [35], [36], [37], [38] aim at surveying traditional SR algorithms or some studies mainly concentrate on providing quantitative evaluations based on full-reference metrics or human visual perception [39], [40]. 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 super-resolution 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 super-resolution modularly. Section 4 gives a brief introduction to unsupervised super-resolution methods. Section 5 introduces some popular domain-specific SR applications, and Section 6 discusses future directions and open issues.

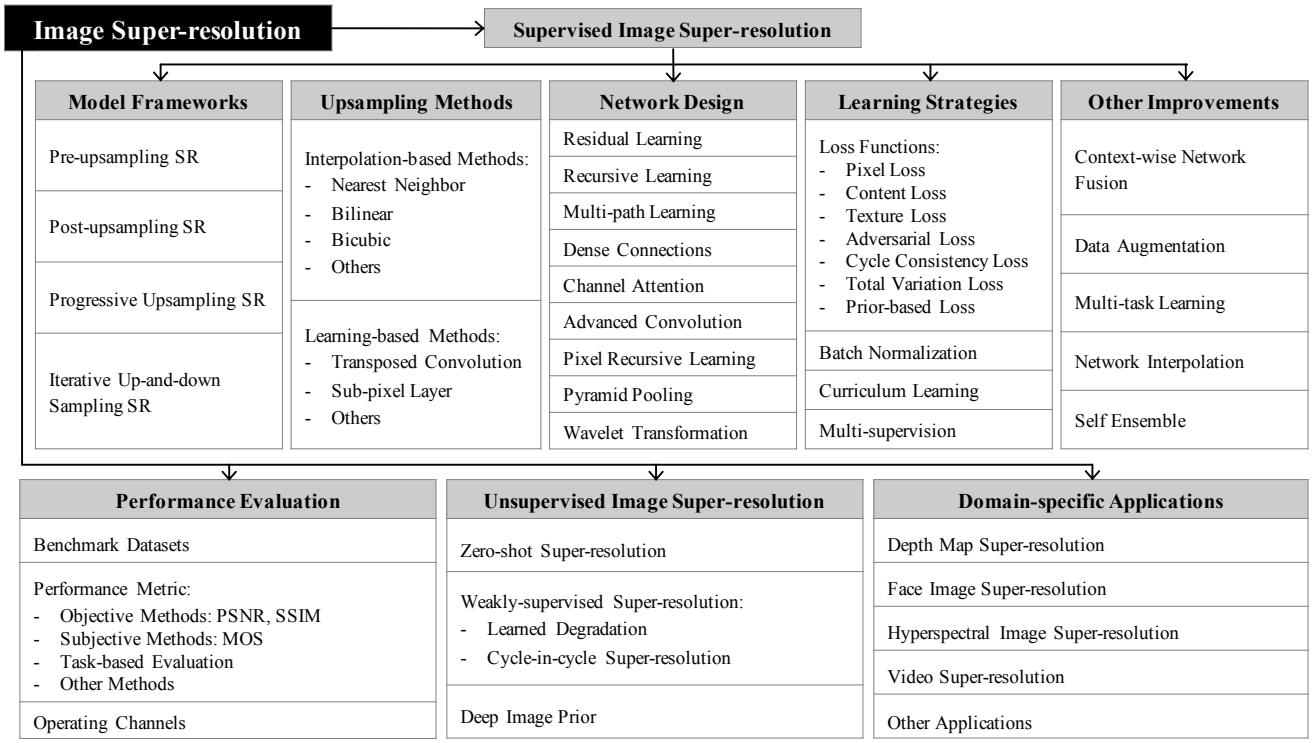


Fig. 1. Hierarchically-structured taxonomy of this survey.

## 2 Problem Setting and Terminology

### 2.1 Problem Definitions

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

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

where $I_y$ is the corresponding HR image, $\mathcal{D}$ represents a degradation mapping function, and $\delta$ denotes the parameters of the degradation process (e.g., the scaling factor or some noise factors). Under general conditions, the degradation process (i.e., $\mathcal{D}$ and $\delta$) is unknown and only LR images are provided. In this case, researchers are required to recover the corresponding HR image $\hat{I}_y$ from the LR image $I_x$, so that $\hat{I}_y$ is identical to the ground truth HR image $I_y$, following the process:

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

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

Although the degradation process is unknown and can be affected by various factors (e.g., defocusing, compression artefacts, anisotropic degradations, sensor noise and speckle noise, etc), 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 super-resolution are built based on this pattern, and the most commonly used downsampling operation is bicubic interpolation with antialiasing. However, there are other works [41] 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 $\kappa$ and the HR image $I_y$, and $n_\varsigma$ is some additive white Gaussian noise with standard deviation $\varsigma$. 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 [41].

To this end, the objective of super-resolution 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 $\lambda$ is the trade-off 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

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

| Dataset | Amount | Avg. Resolution | Avg. Pixels | Format | Category Keywords |
|---|---|---|---|---|---|
| BSDS300 [42] | 300 | (435, 367) | 154, 401 | JPG | animal, building, food, landscape, people, plant, etc |
| BSDS500 [43] | 500 | (432, 370) | 154, 401 | JPG | animal, building, food, landscape, people, plant, etc |
| DIV2K [44] | 1000 | (1972, 1437) | 2, 793, 250 | PNG | environment, flora, fauna, handmade object, people, scenery, etc |
| General-100 [45] | 100 | (435, 381) | 181, 108 | BMP | animal, daily necessity, food, people, plant, texture, etc |
| L20 [46] | 20 | (3843, 2870) | 11, 577, 492 | PNG | animal, building, landscape, people, plant, etc |
| Manga109 [47] | 109 | (826, 1169) | 966, 011 | PNG | manga volume |
| OutdoorScene [48] | 10624 | (553, 440) | 249, 593 | PNG | animal, building, grass, mountain, plant, sky, water |
| PIRM [49] | 200 | (617, 482) | 292, 021 | PNG | environments, flora, natural scenery, objects, people, etc |
| Set5 [50] | 5 | (313, 336) | 113, 491 | PNG | baby, bird, butterfly, head, woman |
| Set14 [51] | 14 | (492, 446) | 230, 203 | PNG | humans, animals, insects, flowers, vegetables, comic, slides, etc |
| T91 [23] | 91 | (264, 204) | 58, 853 | PNG | car, flower, fruit, human face, etc |
| Urban100 [52] | 100 | (984, 797) | 774, 314 | PNG | architecture, city, structure, urban, etc |

of image datasets commonly used by the SR community, 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 in this field, including ImageNet [53], MS-COCO [54], VOC2012 [55], CelebA [56], LSUN [57], WED [58], etc. In addition, combining multiple datasets for training is also popular, such as combining T91 and BSD300 [28], [29], [59], [60], combining DIV2K and Flickr2K [33], [61], etc.

## 2.3 Image Quality Assessment

Image quality refers to visually significant attributes of images and focuses on the perceptual assessments of human viewers. And the process of determining the image quality is called image quality assessment (IQA). In general, IQA methods include subjective methods based on the human observer's perceptual evaluation and objective methods based on computational models automatically predicting the image quality. The subjective methods are more in line with our need but usually inconvenient, time-consuming and expensive, thus the objective methods are currently the mainstream IQA methods. However, subjective and objective methods aren't necessarily consistent between each other, because the latter ones are often unable to capture the human visual perception very accurately, which may lead to large difference in the IQA results [27], [62].

In addition, the objective IQA methods are further divided into three types [62]: full-reference methods performing an assessment using reference images assumed to have perfect quality, reduced-reference methods based on comparisons of extracted features of both images, and no-reference methods (i.e., blind IQA) trying to assess the quality without any reference images. Here we concentrate on full-reference IQA methods since in general cases we often assume that we have perfect ground truth HR images for IQA.

In this section, 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 commonly used to measure the reconstruction quality of lossy transformation (e.g., image compression, image inpainting). For image super-resolution, PSNR is defined via the maximum possible pixel value (denoted as $L$) and the mean squared error (MSE) between images. Given the ground truth image $I$ and reconstructed image $\hat{I}$, both of which are with $N$ pixels, the MSE and the PSNR (in dB) between $I$ and $\hat{I}$ are defined as follows:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(I(i) - \hat{I}(i))^2, \tag{6}$$

$$\mathrm{PSNR} = 10 \cdot \log_{10}(\frac{L^2}{\mathrm{MSE}}). \tag{7}$$

In general cases using 8-bit image representations, $L$ equals to 255 and the typical values for the PSNR vary from 20 to 40, where higher is better. When $L$ is fixed, the PSNR is only related to the pixel-level MSE between images, only caring about the difference between the pixel values at the same positions instead of human visual perception (i.e., how realistic the image looks). This leads to PSNR's poor performance in representing the quality of the super-resolved images in real scenes, in which cases we're usually more concerned with human perception. However, due to the necessity to compare performance with literature works and the lack of completely accurate perceptual metrics, PSNR is 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 structural information from the viewing field [63], the structural similarity index (SSIM) [62] is proposed for measuring the structural similarity between images, based on three relatively independent comparisons, namely luminance, contrast, and structure. For an image $I$ with $N$ pixels, the luminance and contrast are estimated as

the mean and the standard deviation of the image intensity, respectively, as follows:

$$\mu_I = \frac{1}{N}\sum_{i=1}^{N} I(i), \tag{8}$$

$$\sigma_I = (\frac{1}{N-1}\sum_{i=1}^{N}(I(i)-\mu_I)^2)^{\frac{1}{2}}, \tag{9}$$

where $I(i)$ represents the intensity of the $i$-th pixel of image $I$. And the comparison functions on luminance and contrast, denoted as $\mathcal{C}_l(I,\hat{I})$ and $\mathcal{C}_c(I,\hat{I})$ respectively, are given by:

$$\mathcal{C}_l(I,\hat{I}) = \frac{2\mu_I\mu_{\hat{I}} + C_1}{\mu_I^2 + \mu_{\hat{I}}^2 + C_1}, \tag{10}$$

$$\mathcal{C}_c(I,\hat{I}) = \frac{2\sigma_I\sigma_{\hat{I}} + C_2}{\sigma_I^2 + \sigma_{\hat{I}}^2 + C_2}, \tag{11}$$

where $C_1 = (k_1 L)^2$ and $C_2 = (k_2 L)^2$ are constants for avoiding instability, $k_1 \ll 1$ and $k_2 \ll 1$ are small constants, and $L$ is the maximum possible pixel value.

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

$$\sigma_{I\hat{I}} = \frac{1}{N-1}\sum_{i=1}^{N}(I(i)-\mu_I)(\hat{I}(i)-\mu_{\hat{I}}), \tag{12}$$

$$\mathcal{C}_s(I,\hat{I}) = \frac{\sigma_{I\hat{I}} + C_3}{\sigma_I\sigma_{\hat{I}} + C_3}, \tag{13}$$

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:

$$\mathrm{SSIM}(I,\hat{I}) = [\mathcal{C}_l(I,\hat{I})]^\alpha [\mathcal{C}_c(I,\hat{I})]^\beta [\mathcal{C}_s(I,\hat{I})]^\gamma, \tag{14}$$

where $\alpha$, $\beta$, $\gamma$ are control parameters for adjusting the relative importance. In practice, researcher often set $\alpha = \beta = \gamma = 1$ and $C_3 = C_2/2$, so it comes to a specific form of SSIM:

$$\mathrm{SSIM}(I,\hat{I}) = \frac{(2\mu_I\mu_{\hat{I}} + C_1)(\sigma_{I\hat{I}} + C_2)}{(\mu_I^2 + \mu_{\hat{I}}^2 + C_1)(\sigma_I^2 + \sigma_{\hat{I}}^2 + C_2)}. \tag{15}$$

In addition, due to the possible unevenly distribution of image statistical features or distortions, assessing image quality locally is more reliable than applying it globally. Thus mean structural similarity (MSSIM) [62] is proposed for assessing SSIM locally. Specifically, it splits the images into multiple windows, assesses the SSIM of each window, and finally averages them as the final MSSIM.

Since the SSIM evaluates the reconstruction quality from the perspective of the HVS, it better meets the requirements of perceptual assessment [64], [65], and is also widely used by SR models.

#### 2.3.3 Mean Opinion Score

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

The MOS testing has some inherent defects, such as non-linearly perceived scales, biases and variance of rating criteria, and differences between the subjective views of different raters. But when the number of evaluators and evaluations is sufficient, it is still a faithful IQA method, even the one that best fits our needs. In reality,, there are some SR models performing poorly in common IQA metrics such as PSNR and SSIM but far exceeding other models in terms of perceptual quality, in which case the MOS testing is the most reliable IQA method for accurately measuring the perceptual quality [10], [27], [48], [66], [67], [68], [69].

#### 2.3.4 Task-based Evaluation

According to the fact that SR models can often help other vision tasks [7], [8], [9], [10], [11], evaluating reconstruction performance by means of other tasks is another effective way for IQA. Specifically, researchers feed the original and the reconstructed HR images into a trained model, and evaluate the reconstruction quality by comparing the impacts on the prediction performance. The vision tasks used for evaluation include object recognition [10], [70], face recognition [71], [72], face alignment and parsing [32], [73], etc.

However, these methods tend to focus on some specific image attributes which are more favorable to the vision task, instead of the visually perceptual quality. For example, the object recognition models may focus on the high-level semantics while ignoring the image contrast and noise. But on the other hand, they are more in line with the needs of some domain-specific applications (e.g., super-resolving surveillance video for face recognition). In these cases, this evaluation index best reflects the performance of the SR models.

#### 2.3.5 Other IQA Methods

In addition to the above works, there are also other infrequently used metrics for evaluating SR performance. The multi-scale structural similarity (MS-SSIM) [74] supplies more flexibility than single-scale SSIM in incorporating the variations of viewing conditions. Sheikh *et al.* propose information fidelity criterion (IFC) [75] and visual information fidelity (VIF) [76], which treat HVS as a communication channel and predict the subjective image quality by computing the mutual information between the reconstructed images and the reference images. But these two methods don't respond to the structural information of the image explicitly. Besides, the feature similarity (FSIM) [77] extracts feature points of human interest based on phase congruency and image gradient magnitude to evaluate image quality. Although these methods exhibit better performance on capturing human visual perception than PSNR and SSIM, the most widely used SR IQA methods are still PSNR and SSIM due to some historical reasons.

### 2.4 Operating Channels

In addition to the commonly used RGB color space, the YCbCr color space is also widely used for representing images and performing super-resolution. 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 channels, earlier models favor operating on the Y channel of YCbCr space [28], [45], [78], [79], while more recent models tend to operate on RGB channels [30], [33], [61], [70]. It is worth noting that operating (training or evaluation) on different color spaces or channels makes the evaluated performance greatly different (up to 4+ dB) [25].

## 3 Supervised Super-resolution

Nowadays researchers have proposed a variety of super-resolution models with deep learning. These models focus on supervised super-resolution, i.e., trained with both LR images and corresponding ground truth 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, etc. 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 high-resolution output from low-resolution input) is the key problem. Although the architectures of existing SR models vary widely, they can be attributed to four model frameworks (namely pre-upsampling SR, post-upsampling SR, progressive upsampling SR and iterative up-and-down sampling SR, as Fig. 2 shows), based on the employed upsampling operations and their locations in the model. Below we will detail these frameworks.

#### 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. In consideration of this, Dong *et al.* [24], [25] firstly adopt the pre-upsampling SR framework (as depicted in Fig. 2a) 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.

The advantage of this framework is that the difficult upsampling task has been done by predefined traditional algorithms, and deep CNNs only need to refine the coarse images, which significantly reduces the learning difficulty. In addition, these models can take interpolated images with arbitrary size and scaling factors as input, and give refined results with comparable performance to single-scale SR models [28]. Thus it has gradually become one of the most popular frameworks in this field [59], [60], [80], [81], 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 methods often introduce some 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 [45], [82].

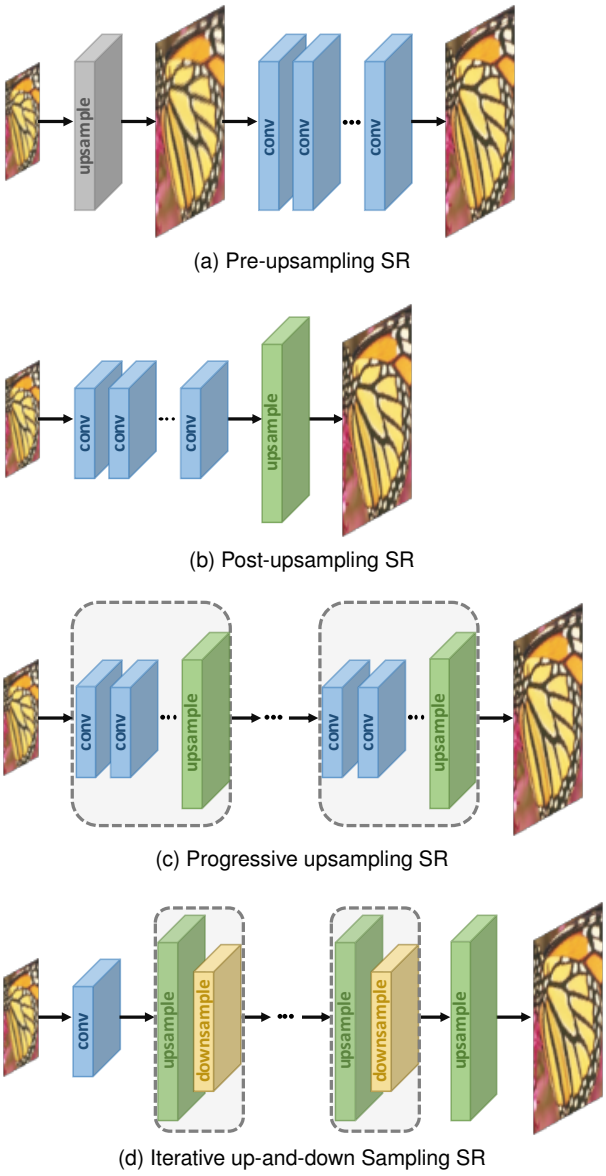


Fig. 2. Super-resolution model frameworks based on deep learning. The trapezoids denote the up-or-down sampling operations, depending on their directions. The gray ones denote predefined upsampling operations, while the green ones and yellow ones indicate learnable upsampling or downsampling layers, respectively. Blue boxes represent convolutional layers, and the blocks enclosed by the dashed box represent some modules that can be stacked in the frameworks.

#### 3.1.2 Post-upsampling Super-resolution

In order to solve the problem of computational efficiency and make full use of deep learning technology to improve image resolution automatically, researchers propose to per-

form most of the mappings in low-dimensional space by replacing the predefined upsampling operations with end-to-end learnable upsampling layers integrated at the end of SR models. In the pioneer works [45], [82] of this framework, namely post-upsampling SR as Fig. 2b shows, the LR input images are fed into deep CNNs without increasing the resolution, and end-to-end learnable upsampling layers are applied at the end of the network.

Due to the fact that the feature extraction process through nonlinear convolutions with huge computational cost only occurs in low-dimensional space and the resolution increases only at the very end of the network, the computation complexity and spatial complexity is much reduced, and it also brings considerably faster training speed and inference speed. Therefore, this framework also has become one of the most mainstream frameworks in the super-resolution field [27], [33], [79], [83]. 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 models under post-upsampling SR framework have reduced the immensely large computational and runtime cost, it still has some shortcomings. On the one hand, the upsampling operation 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 an individual SR model, which cannot cope with the need for multi-scale SR. To address these drawbacks, a progressive upsampling SR framework is adopted by Laplacian pyramid SR network (LapSRN) [29], 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. Some other works such as MS-LapSRN [69] and progressive SR (ProSR) [34] also adopt this framework and achieve relatively high performance. In contrast to the LapSRN and MS-LapSRN which use the intermediate reconstructed images as the "base images" for subsequent modules, the ProSR only 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 not only greatly reduce the learning difficulty and obtain better performance, especially with large factors, but also cope with the multi-scale super-resolution problem without introducing overmuch spacial and temporal cost. In addition, because of the specific multi-stage design of the framework, some specific learning strategies such as curriculum learning (Sec. 3.4.3) and multi-supervision (Sec. 3.4.4) can be 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 difficulty, so more instructional structure designing 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 [14] is incorporated into SR for better mining the deep LR-HR relationships [46]. This SR framework, namely iterative up-and-down sampling SR (Fig. 2d), tries to iteratively apply back-projection refinement, i.e., computing the reconstruction error then fusing it back to tune the HR image intensity. However, the previous studies based on back-projection are mostly not deep learning based and involve some unlearnable operations [84], [85]. To make better use of this mechanism, Haris *et al.* [61] exploit iterative up-and-down sampling layers and propose deep back-projection network (DBPN) to mutually connect upsampling layers and downsampling layers alternately and reconstruct the final HR result using concatenation of all the intermediately reconstructed HR feature maps. Coupled with other techniques (e.g., dense connections [86]), the DBPN wins the championship on the classical track of NTIRE 2018 [87].

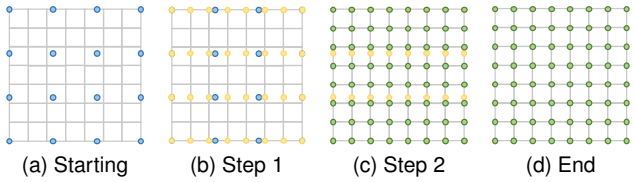


Fig. 3. Interpolation-based upsampling methods. The gray board denotes the coordinates of pixels, and the blue, yellow and green points represent the initial, intermediate and final pixels, respectively.

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. In fact, the back-projection units used in DBPN have a very complicated structure and require heavy manual design. Since this mechanism has just been introduced into super-resolution based on deep learning, the framework has great potential and needs further exploration.

### 3.2 Upsampling Methods

In addition to where to apply the upsampling operations in the model, how to implement them is also of great importance. Although there has been a variety of traditional upsampling algorithms [22], [23], [88], [89], [90], making use of neural networks to directly learn an end-to-end upsampling process has gradually become a trend. In this section, we'll introduce several commonly used 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 used by almost all image-related applications. The traditional interpolation methods include nearest-neighbor interpolation, bilinear and bicubic methods, Sinc and Lanczos resampling, etc. Since these methods are interpretable and easy to implement, some of them are still widely used in super-resolution based on deep learning.

**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.

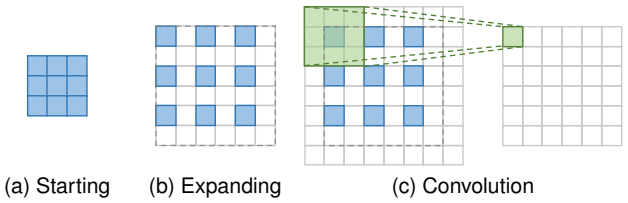


Fig. 4. Transposed convolution layer. The blue boxes denote the input, and the green boxes indicate the kernel and the output of the convolution operation.

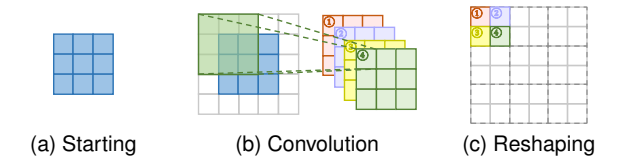


Fig. 5. Sub-pixel layer. The blue boxes denote the input, and the boxes with other colors indicate different convolution operations and different output feature maps.

**Bilinear Interpolation.** The bilinear interpolation first performs linear interpolation on one axis of the image and then performs it again on the other axis. This two-step interpolation process is shown in Fig. 3. Although each step is linear in the sampled values and positions, it results in a quadratic interpolation with a receptive field sized $2 \times 2$, and shows much better performance than nearest-neighbor interpolation while keeping relatively fast speed.

**Bicubic Interpolation.** Similarly, the bicubic interpolation [12] performs a cubic interpolation on each of the two dimensions of the image, as Fig. 3 shows. Compared to bilinear interpolation, the bicubic interpolation takes $4 \times 4$ pixels into count, and thus generates smoother results with fewer interpolation artefacts and lower speed. In fact, the bicubic interpolation with anti-aliasing is currently the mainstream method for constructing SR datasets (i.e., degrading HR images to corresponding 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 contents, which doesn't bring any more information. Instead, they often introduce some side effects into the SR models, such as computational complexity, noise amplification, blurring results, etc.

#### 3.2.2 Learning-based Upsampling

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

**Transposed Convolution Layer.** Transposed convolution layer, a.k.a. deconvolution layer [91], [92], tries to perform a transformation opposite a normal convolution, i.e., predicting the possible input based on feature maps sized like the output of convolutional layers. Specifically, it improves the image resolution by expanding the image by inserting zero values and performing convolution. In the interest of brevity, we show how to perform $2\times$ upsampling with a $3 \times 3$ kernel, as Fig. 4 shows. At first, the input is expanded twice of the original size, where the newly added pixel values are set to 0 (Fig. 4b). Then a convolution with kernel sized $3 \times 3$, stride 1 and padding 1 is applied (Fig. 4c). Through such an operation, the input feature map is upsampled by a factor 2, in which case the receptive field is at most $2 \times 2$.

Since the transposed convolution layer can enlarge the image size in an end-to-end manner while maintaining a connectivity pattern compatible with vanilla convolution, it is widely used as the upsampling layer in SR models [61], [78], [79], [83]. However, this layer can easily cause "uneven overlapping" on each axis [93], and the multiplied results on both axes further create a characteristic checkerboard-like pattern of varying magnitudes and thus hurt the SR performance.

**Sub-pixel Layer.** The sub-pixel layer [82], which is also an 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 normal convolution is firstly applied for producing outputs with $s^2$ times channels, where $s$ is the upsampling 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 (named *shuffle* in [82]) is performed to produce outputs with size $sh \times sw \times c$ (Fig. 5c). In this case, the receptive field can be up to $3 \times 3$.

Due to the end-to-end upsampling manner, the sub-pixel layer is also widely used by SR models [27], [30], [41], [94]. Compared with transposed convolution layer, the greatest advantage of sub-pixel layer is the larger receptive field, which provides more contextual information to help generate more accurate details. Nevertheless, the distribution of the receptive fields of sub-pixel layers is uneven, blocky regions actually share the same receptive field, which may result in some artefacts near the boundaries of different blocks.

Nowadays, these two 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 features 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 field, researchers apply various network design strategies (e.g., residual learning, dense connections, etc) on top of the four SR frameworks (Sec. 3.1) to build the final SR networks. In this section, we decompose these networks to the essential principles or strategies for network design and introduce them one by one.

#### 3.3.1 Residual Learning

Before He *et al.* [95] propose ResNet for learning residuals instead of a thorough mapping, residual learning has been widely employed by SR models [50], [88], [96], as Fig. 6a shows. Among them, the residual learning strategies can

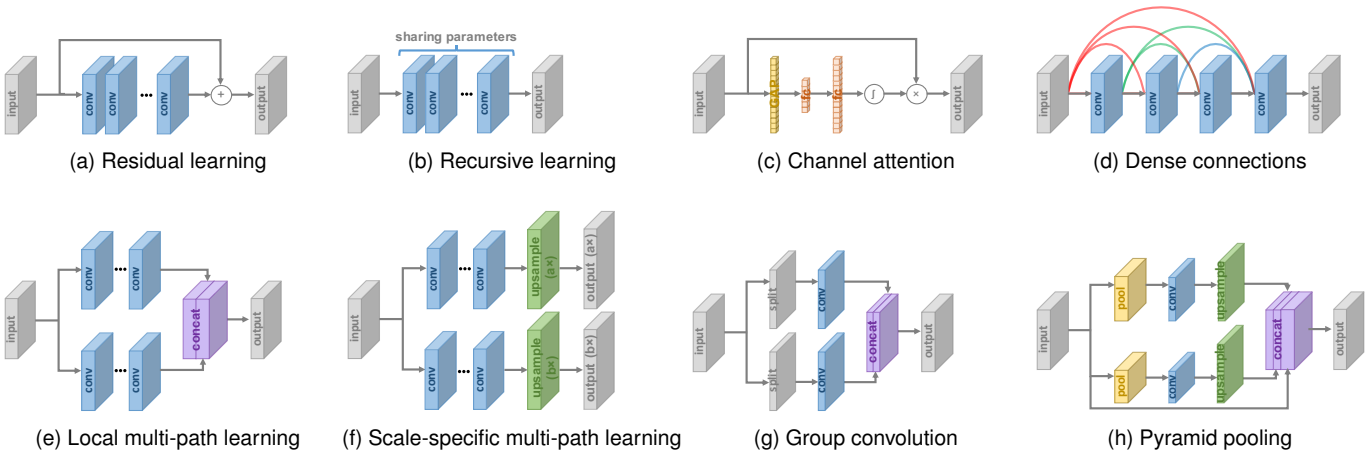


Fig. 6. Network design strategies.

be roughly divided into two types, i.e., global and local residual learning.

**Global Residual Learning.** Since super-resolution 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 two images, 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. And because 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 [28], [59], [60], [97], especially under the pre-upsampling framework (Sec. 3.1.1).

**Local Residual Learning.** The local residual learning is similar to the residual learning in ResNet and used to alleviate the degradation problem [95] caused by ever-increasing network depths and improve the learning ability. It is also widely used in the SR field [70], [78], [83], [98].

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

### 3.3.2 Recursive Learning

In order to achieve larger receptive field and learn higher-level features without introducing overwhelming parameters, recursive learning, which refers to applying the same modules multiple times in a recursive manner, is introduced into the super-resolution field, as Fig. 6b shows.

Among them, the 16-recursion DRCN [80] employs a single convolutional layer as the recursive unit and reaches a receptive field of $41 \times 41$, which is much larger than $13 \times 13$ of SRCNN [24], without over many parameters. The DRRN [60] uses a residual block [95] as the recursive unit for 25 recursions and obtains even better performance than a non-recursive baseline with 17 residual blocks. Later Tai *et al.* [59] propose MemNet based on the memory block, which is composed of a 6-recursive residual block where the outputs of every recursion are concatenated and go through an extra $1 \times 1$ convolution for memorizing and forgetting. Recently the cascading residual network (CARN) [30] also adopts a similar recursive unit including several residual blocks.

Different from above works, some researchers decompose super-resolution with large scaling factors into several sub-problems with small factors, and use recursive structure to solve multiple sub-problems simultaneously. Specifically, Han *et al.* [83] propose dual-state recurrent network (DSRN) to exchange signals between the HR state and the LR state. At each time step (i.e., recursion), they update the LR state based on the current LR state and HR state, and then transmit it to the HR state for updating. By means of the dual-state recursive learning (up to 7 recursions), the deep relationships between LR-HR image pairs are better explored. In contrast, Lai *et al.* [69] not only use a convolutional layer as a recursive layer, but also employ the feature embedding module, feature upsampling module and image upsampling module as recursive modules, whose parameters are shared for each sub-problem. By this way the amount of model parameters is much reduced (up to 8 times) at the expense of a little performance loss.

In practice, recursive learning 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 combined together with recursive learning for mitigating these problems [59], [60], [80], [83].

### 3.3.3 Multi-path Learning

Multi-path learning refers to passing features through multiple paths of the model, which perform different operations, for providing better modelling capabilities. Specifically, it could be divided into three types, as detailed below.

**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 ability of feature extraction. Specifically, the LapSRN [29] includes a feature extraction path predicting the sub-band residuals in a coarse-to-fine fashion, and an image reconstruction path to reconstruct visible HR images based

on the information streams of both paths. Similarly, the DSRN [83] utilizes an LR path and an HR path to extract information in low-dimensional space and high-dimensional space, respectively. These two paths continuously exchanging information for further improving learning ability. The pixel recursive super-resolution [68] adopts a conditioning path to capture the global structure of images, and a prior path to capture the serial dependence of the generated pixels. In contrast, Ren *et al.* [99] employ multiple paths with unbalanced structures to perform upsampling and fuse them at the very end of the model.

**Local Multi-path Learning.** Motivated by inception module [100], the MSRN [98] adopts a new block for multi-scale feature extraction, as Fig. 6e shows. In this block, two convolution operations with kernel size $3 \times 3$ and $5 \times 5$ are adopted to extract features simultaneously, then the outputs are concatenated and go through the same operations again, and finally an extra $1 \times 1$ convolution is applied. A shortcut connects the outputs and the inputs of this block 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 actually need to go through the similar feature extraction process, Lim *et al.* [33] propose a scale-specific multi-path learning strategy to cope with multi-scale SR problems with a single network. To be concrete, they share the principal part of the model (i.e., the intermediate part for feature extraction), and attach scale-specific pre-processing paths and upsampling paths at the beginning and end of the network, respectively (as Fig. 6f shows). During training, only the paths that correspond to the selected scale are enabled and updated. In this way, most of the parameters are shared across different scales, and the proposed MDSR exhibits comparable performance as single-scale models. The similar scale-specific multi-path learning is also adopted by CARN [30] and ProSR [34].

#### 3.3.4 Dense Connections

Since Huang *et al.* [86] 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. The dense connections not only help alleviate gradient vanishing, enhance signal propagation and encourage feature reuse, but also substantially reduce the number of parameters by employing small growth rate (i.e., number of channels in dense blocks) and squeezing channels after concatenation.

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. 6d 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 [59], CARN [30], RDN [94] and ESRGAN [101]. The DBPN [61] also adopts dense connections extensively, but their dense connections are between all of the upsampling units, as are the downsampling units.

#### 3.3.5 Channel Attention

Considering the interdependence and interaction of the feature representations between different channels, Hu *et al.* [102] propose a "squeeze-and-excitation" block to improve representation ability by explicitly modelling channel interdependence, as Fig. 6c shows. In this block, each input channel is squeezed into a channel descriptor (i.e., a constant) using global average pooling, and then these descriptors are fed into two fully-connected layers to produce channel-wise scaling factors. The final output is obtained by rescaling the input channels with the scaling factors using channel-wise multiplication. Using this channel attention mechanism, the proposed SENet won the first place in ILSVRC 2017 [103]. Recently, Zhang *et al.* [70] firstly incorporate it into super-resolution and propose RCAN, which markedly improves the representation ability of the model and advances the SR performance.

#### 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 faster speeds.

**Dilated Convolution.** It is well known that the contextual information facilitates generating realistic details in image super-resolution. Thus Zhang *et al.* [104] replace the common convolution by dilated convolution in SR models, increase the receptive field over twice and finally achieve much better performance.

**Group Convolution.** Motivated by recent advances on lightweight CNNs [105], Ahn *et al.* [30] propose CARN-M by replacing the common convolution by group convolution. As some previous works have proven that the group convolution can reduce plenty of parameters and operations at the expense of little performance [105], [106], [107], the CARN-M reduces the number of parameters by $5$ times and operations by $4$ times with only a little performance loss.

#### 3.3.7 Pixel 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 [108], Dahl *et al.* [68] 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., $8 \times 8$) and far exceeds the previous methods on MOS testing (Sec. 2.3.3).

Motivated by the human attention shifting mechanism [109], the Attention-FH [110] 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 [111], Zhao *et al.* [112] 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 $1 \times 1$ convolution is performed for compressing the outputs to one 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 can integrate global as well as local contextual information effectively. By incorporating this module, the proposed EDSR-PP model [113] further improve the performance.

#### 3.3.9 Wavelet Transformation

As is well-known, the wavelet transformation (WT) [114], [115] is a highly efficient representation of images by decomposing the image signal into high-frequency wavelets denoting texture details and low-frequency wavelets containing global topological information. Bae *et al.* [116] 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 [117] and Wavelet-SRNet [118] 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 [119] adopts multi-level WT and takes the concatenated sub-bands as the input to a single CNN for better capturing the dependence between them.

### 3.4 Learning Strategies

#### 3.4.1 Loss Functions

In the super-resolution field, loss functions are used to measure the difference between generated HR images and ground truth HR images, 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 [31], adversarial loss [27], etc.) are adopted to better measure the reconstruction error. Nowadays these loss functions have been playing an important role in this field. In this section, we'll take a closer look at the loss functions used widely in SR models. 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):

$$\mathcal{L}_{\text{pixel\_l1}}(\hat{I}, I) = \frac{1}{hwc} \sum_{i,j,k} |\hat{I}_{i,j,k} - I_{i,j,k}|, \tag{16}$$

$$\mathcal{L}_{\text{pixel\_l2}}(\hat{I}, I) = \frac{1}{hwc} \sum_{i,j,k} (\hat{I}_{i,j,k} - I_{i,j,k})^2, \tag{17}$$

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 [29], [120], given by:.

$$\mathcal{L}_{\text{pixel\_Cha}}(\hat{I}, I) = \frac{1}{hwc} \sum_{i,j,k} \sqrt{(\hat{I}_{i,j,k} - I_{i,j,k})^2 + \epsilon^2}, \tag{18}$$

where $\epsilon$ is a small constant (e.g., $1e-3$) for numerical stability.

The pixel loss constrains the generated HR image $\hat{I}$ to be close enough to the ground truth HR image $I$ on the pixel values. Comparing with L1 loss, the L2 loss penalizes larger errors but is more tolerant to small errors. In practice, the L1 loss shows improved performance and convergence over L2 loss [30], [33], [121]. 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 has become the most widely used loss function in this field. However, since the pixel loss actually doesn't take image quality (e.g., perceptual quality [31], textures [10]) into account, it often lacks high-frequency details and produces perceptually unsatisfying results with overly smooth textures [27], [31], [62], [74].

**Content Loss.** To evaluate image quality based on the perceptual quality, the content loss is introduced into super-resolution [31], [122]. Specifically, it measures the semantic differences between images using a pre-trained image classification network. Denoting this network as $\phi$ 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 between two images, as follows:

$$\mathcal{L}_{\text{content}}(\hat{I}, I; \phi, l) = \frac{1}{h_l w_l c_l} \sqrt{\sum_{i,j,k} (\phi^{(l)}_{i,j,k}(\hat{I}) - \phi^{(l)}_{i,j,k}(I))^2}, \tag{19}$$

where $h_l$, $w_l$ and $c_l$ are the height, width and number of channels of the extracted feature maps on layer $l$, respectively.

Essentially the content loss transfers the learned knowledge of hierarchical image features from the classification network $\phi$ 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 [10], [27], [31], [32], [48], [101], where the VGG [123] and ResNet [95] 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.* [124], [125], the texture loss (a.k.a. style

reconstruction loss) is introduced into super-resolution. Following [124], [125], the texture of an image is regarded as the correlations between different feature channels and defined as the Gram matrix $G^{(l)} \in \mathcal{R}^{c_l \times c_l}$, where $G^{(l)}_{ij}$ is the inner product between the vectorized feature maps $i$ and $j$ on layer $l$:

$$G^{(l)}_{ij}(I) = \text{vec}(\phi^{(l)}_i(I)) \cdot \text{vec}(\phi^{(l)}_j(I)), \tag{20}$$

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

$$\mathcal{L}_{\text{texture}}(\hat{I}, I; \phi, l) = \frac{1}{c_l^2} \sqrt{\sum_{i,j} (G^{(l)}_{i,j}(\hat{I}) - G^{(l)}_{i,j}(I))^2}. \tag{21}$$

By employing texture loss, the SR model can create realistic textures and produce visually more satisfactory results [10]. Despite this, determining the size of the patch to match textures is still empirical. Too small patches lead to artefacts in textured regions, while too large patches lead to artefacts throughout the entire image because texture statistics are averaged over regions of varying textures.

**Adversarial Loss.** In recent years, the GANs [26] have been more and more popular and 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 output 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 iterative adversarial training and after the model eventually converges, 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 the super-resolution field, it is straightforward to adopt adversarial learning, in which case we only need to treat the SR model as a generator, and additionally define a discriminator to judge whether the input image is generated or not. Ledig *et al.* [27] firstly introduce SRGAN using adversarial loss based on cross entropy, as follows:

$$\mathcal{L}_{\text{gan\_ce\_g}}(\hat{I}; D) = -\log D(\hat{I}), \tag{22}$$

$$\mathcal{L}_{\text{gan\_ce\_d}}(\hat{I}, I_s; D) = -\log D(I_s) - \log(1 - D(\hat{I})), \tag{23}$$

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. $I_s$ represents randomly sampled data from ground truth HR images. Besides, the Enhancenet [10] also adopts the similar adversarial loss.

Furthermore, Wang *et al.* [34] and Yuan *et al.* [126] use adversarial loss based on least square error for more stable training process and higher quality results [127], given by:

$$\mathcal{L}_{\text{gan\_ls\_g}}(\hat{I}; D) = (D(\hat{I}) - 1)^2, \tag{24}$$

$$\mathcal{L}_{\text{gan\_ls\_d}}(\hat{I}, I_s; D) = (D(\hat{I}))^2 + (D(I_s) - 1)^2. \tag{25}$$

And Bulat *et al.* [128] adopt the hinge-format adversarial loss [129], as follows:

$$\mathcal{L}_{\text{gan\_hi\_g}}(\hat{I}; D) = -D(\hat{I}), \tag{26}$$

$$\mathcal{L}_{\text{gan\_hi\_d}}(\hat{I}, I_s; D) = \min(0, D(\hat{I}) - 1) + \min(0, -D(I_s) - 1). \tag{27}$$

In contrast to the above works focusing on the specific form of adversarial loss, Park *et al.* [130] argue that the pixel-level discriminator only causes the generator to generate meaningless high-frequency noise (which cannot be learned by pixel loss), and attach an additional feature-level discriminator to operate on high-level representations extracted by a pre-trained CNN for capturing more meaningful potential attributes of real HR images. Xu *et al.* [67] incorporate a multi-class GAN including a single generator and class-specific discriminators. And the ESRGAN [101] employs relativistic GAN [131] to predict the probability that real images are relatively more realistic than fake ones, instead of predicting the probability that input images are real or generated.

Extensive MOS tests [10], [27] 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. 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 [129], [132], [133], 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.* [134] for image-to-image translation tasks, Yuan *et al.* [126] 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 a 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:

$$\mathcal{L}_{\text{cycle}}(I', I) = \frac{1}{hwc} \sqrt{\sum_{i,j,k} (I'_{i,j,k} - I_{i,j,k})^2}. \tag{28}$$

**Total Variation Loss.** In order to suppress noise in generated images, the total variation (TV) loss [135] is introduced into the SR field by Aly *et al.* [136]. It is defined as the sum of the absolute differences between neighboring pixels and measures how much noise is in the images. For the generated HR image $\hat{I}$, the TV loss is define as:

$$\mathcal{L}_{\text{TV}}(\hat{I}) = \frac{1}{hwc} \sum_{i,j,k} \sqrt{(\hat{I}_{i,j+1,k} - \hat{I}_{i,j,k})^2 + (\hat{I}_{i+1,j,k} - \hat{I}_{i,j,k})^2}. \tag{29}$$

Lai *et al.* [27] and Yuan *et al.* [126] also adopt this 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 process. Bulat *et al.* [32] focus on face image SR and introduce a face alignment network (FAN) to constrain the consistency of facial landmarks detected from the original and generated images. The FAN is pre-trained and integrated for providing face alignment knowledge. In this way, the proposed Super-FAN improves performance both on low-resolution face alignment and face image super-resolution. 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 performance of super-resolution can be further improved.

In this section, we introduce various loss functions widely used in the super-resolution field. In practice, researchers often combine multiple loss functions by weighted average [10], [27], [29], [48], [126] for constraining different aspects of the generation process, especially for distortion-perception tradeoff [27], [101], [137], [138], [139]. However, how to determine the weights of different loss functions requires a lot of empirical exploration. How to combine these loss functions reasonably and effectively remains a problem.

#### 3.4.2 Batch Normalization

In order to accelerate training of deep CNNs, Sergey *et al.* [140] propose batch normalization (BN) to reduce internal covariate shift of networks. To be concrete, they perform the 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 the vanishing gradient problem, it allows us to use much higher learning rates and be less careful about initialization. Thus this technique is widely used by SR models [27], [41], [59], [60], [119], [141].

However, Lim *et al.* [33] argue that the BN loses the scale information of each image and gets rid of range flexibility from networks. So they remove BN layers, use the saved memory (up to 40%) to employ a much larger model, and thus increase the performance substantially. Some other models [34], [101], [142] also adopt this experience and achieve performance improvements.

#### 3.4.3 Curriculum Learning

Curriculum learning [143] refers to starting from an easier subtask and gradually increasing the task difficulty. Since super-resolution is essentially an ill-posed problem and some adverse conditions such as large scaling factors, noise or blurring further increase the learning difficulty, the curriculum training strategy helps a lot on this problem.

Considering that performing SR with large factors in one step is a very difficult task, Wang *et al.* [34] and Bei *et al.* [144] propose ProSR and ADRSR, 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 portion, and after finishing training current portions, the portions with $4\times$ or larger scaling factors are gradually mounted and blended with the previous portions. Specifically, the ProSR blends two portions by linearly combining the output of this level and the upsampled output of previous levels following [145], while the ADRSR concatenates them and attaches another convolutional layer.

In contrast, Park *et al.* [113] divide the $8\times$ SR problem to three sub-problems (i.e., $1\times$ to $2\times$ SR, $2\times$ to $4\times$ SR, $4\times$ to $8\times$ SR) and train an individual network for each problem. Then two of them is concatenated and fine-tuned jointly, and then with the other one. In addition, they also decompose the $4\times$ SR under difficult conditions into three sub-problems (i.e., denoising/deblurring, $1\times$ to $2\times$ SR, $2\times$ to $4\times$ SR) and adopt a similar training strategy.

Compared to common training procedure, this curriculum learning strategy not only greatly reduces the training difficulty and improves the performance with all scaling factors, especially for large factors, but also significantly shortens the total training time.

#### 3.4.4 Multi-supervision

Multi-supervision refers to adding multiple extra supervision signals within the model for enhancing the gradient propagation and avoiding vanishing and exploding gradient. In order to prevent the gradient problems introduced by recursive learning (Sec. 3.3.2), the DRCN [80] incorporates multi-supervision to recursion units. Specifically, they feed each output of recursive units into a reconstruction module to generate an HR image, and construct the final prediction by weighted averaging all these intermediate HR images, where the weights are learned during training. Similar multi-supervision approaches are taken by MemNet [59] and DSRN [83], which are also based on recursive learning.

Since the LapSRN [29], [69] under the progressive upsampling SR framework (Sec. 3.1.3) generates intermediate upsampling results of different scales during the feedforward propagation, it is straightforward to adopt multi-supervision. Specifically, the intermediate results are forced to be the same as the intermediate ground truth images downsampled from the original HR image.

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 backpropagated effectively and thus much enhance the model training.

### 3.5 Other Improvements

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

#### 3.5.1 Context-wise Network Fusion

Context-wise network fusion (CNF) [99] 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 [24], [25] achieves comparable performance with state-of-the-art models with acceptable efficiency [99].

#### 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 randomly cropping, flipping, scaling, rotation, color jittering, etc [29], [33], [46], [60], [83], [97]. In addition, Bei *et al.* [144] also randomly shuffle RGB channels, which not only augments data, but also alleviates the biased color problem caused by the dataset with unbalanced colors. With the help of data augmentation, the SR models boost the performance a lot.

#### 3.5.3 Multi-task Learning

Multi-task learning [146] refers to improving generalization ability by using domain-specific information contained in training signals of related tasks, such as object detection and semantic segmentation [147], head pose estimation and facial attribute inference [148]. In the super-resolution field, Wang *et al.* [48] incorporate a pre-trained semantic segmentation network for providing semantic knowledge which enables generating semantic-specific details. Specifically, they introduce a spatial feature transform layer that takes semantic maps as input and outputs spatial-wise parameters for affine transformation performed on the intermediate feature maps. The proposed SFT-GAN thus generates much more realistic and visually pleasing textures on images with rich semantic regions, and obtains comparable performance on other images. Besides, considering that directly super-resolving noisy images may cause noise amplification, the DNSR [144] proposes to train a denoising network and an SR network separately, then concatenates them and fine-tunes together. Similarly, the cycle-in-cycle GAN (CinC-GAN) [126] 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 tend to produce images closer to ground truth but introduce blurring and noise amplifying, while GAN-based models bring better perceptual quality but introduce unpleasant artefacts (e.g., meaningless noise making images more "realistic"). In order to balance the visual quality and image fidelity, Wang *et al.* [101] 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 these two networks to derive intermediate models. By tuning the interpolation weights without retraining networks, they produce meaningful results with much less artefacts.

#### 3.5.5 Self Ensemble

Self ensemble, a.k.a. enhanced prediction [46], is an inference technique commonly used by SR models. Specifically, rotations with different angles (i.e., 0°, 90°, 180°, 270°) and flipping are applied on the LR images to get a set of 8 LR images. Then these images are fed into the SR model and the corresponding reverse transformation is applied to the reconstructed HR images to get the outputs. The final prediction result is conducted by the mean [33], [34], [46], [70], [78], [94] or the median [81] of these outputs.

## 4 Unsupervised Super-resolution

Existing super-resolution works mostly focus on supervised learning, i.e., learning the LR-to-HR mapping using matched LR-HR image pairs. However, since it is difficult to collect images of the same scene of different resolutions, the LR images in SR datasets are often obtained by performing predefined degradation on HR images. Thus the SR models trained on these datasets are more likely to learn a reverse version of the predefined process. In order to prevent the adverse effects brought by the predefined degradation, researchers pay more and more attention to unsupervised super-resolution, in which case only unpaired images (HR or LR) are provided for training, so the resulting models are actually more likely to be able 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 is sufficient to provide the information needed for super-resolution, Shocher *et al.* [81] propose zero-shot super-resolution (ZSSR) to cope with unsupervised SR by training small image-specific SR networks at test time rather than training a generic model on large external datasets. Specifically, they use a kernel estimation method [85] to directly estimate the degradation kernel from a single test image, and use this kernel to construct a small dataset by performing degradation with different scaling factors on the test image. Then a small CNN for super-resolution is trained on this dataset and used for the final prediction.

In this way, the ZSSR leverages on the power of the cross-scale internal recurrence of image-specific information, and thus outperforms previous state-of-the-art approaches by a large margin (+1dB for estimated kernels, +2dB for known kernels) on images under non-ideal conditions (i.e., images obtained by non-bicubic degradation kernels and suffered some effects like blurring, noise, compression artefacts, etc), which is closer to the real-world scenes, while give competitive results under ideal conditions (i.e., images constructed by bicubic interpolation). However, since this method needs to train a single network for each image during testing, which makes its testing time much longer than other SR models with deep learning.

### 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.* [128] 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 image super-resolution 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 [134], Yuan *et al.* [126] 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. Concretely speaking, 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 structure of a CNN is sufficient to capture a great deal of low-level image statistics prior for inverse problems, Ulyanov *et al.* [149] 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 $I_x$. Because the network is randomly initialized and never trained on datasets, the only prior is the CNN structure itself. Although the performance of this method is still much worse than the supervised methods (+2dB), it outperforms traditional bicubic upsampling considerably (+1dB). Besides, it shows the rationality of the CNN architectures itself, and prompts us to improve super-resolution 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 distance between the viewpoint and the objects in the scene, and the depth information plays important roles in many tasks such as pose estimation [150], [151], [152], semantic segmentation [153], [154], etc. However, due to productive and economic limitations, the depth maps produced by depth sensors are often low-resolution and suffer degeneration effects such as noise, quantization, missing values, etc. Thus super-resolution is introduced for increasing the spatial resolution of depth maps.

Today 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.* [155] exploit the depth field statistics and the local correlation between depth maps and RGB images to constrain the global statistics and local structure. Hui *et al.* [156] utilize two CNNs to simultaneously upsample LR depth maps and downsample HR RGB images, then use RGB features as the guidance of the upsampling process at the same resolution. Similarly, Ni *et al.* [157] and Zhou *et al.* [158] use HR RGB images as guidance by extracting HR edge map and predicting missing high-frequency components, respectively. While Xiao *et al.* [159] use the pyramid network to enlarge the receptive field, extract features from LR depth maps and HR RGB images, respectively, and fuse these features to predict HR depth maps. And Haefner *et al.* [160] fully exploit the color information in order to guide super-resolution by resorting to the shape-from-shading technique.

In contrast to the above works, Riegler *et al.* [161] 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 [6], [72], [73], [162]. Compared to generic images, face images have much 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.

The most straightforward way to exploit facial prior is to constrain the generated HR images to have the identical face-related information to ground truth HR images. Specifically, the CBN [163] utilizes the facial prior by alternately optimizing FH and dense correspondence field estimation. The Super-FAN [32] and MTUN [164] 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 the facial prior constraints. The SICNN [72], which aims at recovering the real identity information, 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 [165] incorporates spatial transformer networks [166] for automatic spatial transformations and thus solves the face unalignment problem. Based on TDN, the TDAE [167] 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 [168] 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. Yang *et al.* [169] also decompose deblocked face images into facial components, whose landmarks are used to retrieve adequate HR component exemplars in an external dataset, the background is fed into a generic SR network, and finally fuse them to complete HR face images.

In addition to the above works, researchers also improve FH from other perspectives. Motivated by the human attention shifting mechanism [109], the Attention-FH [110] 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 [170] adopts a network similar to SRGAN [27] with adversarial learning. And Xu *et al.* [67] propose a multi-class GAN-based FH model composed of a generic generator and class-specific discriminators. Both Lee *et al.* [171] and Yu *et al.* [172] utilize additional facial attribute information to perform FH with the specified attributes, based on the conditional GAN [173].

### 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 a variety of vision tasks [174], [175], [176], [177]. Nevertheless, due to hardware limitations, not only collecting high-quality HSIs is much more difficult than collecting PANs, but also the resolution of collected HSIs is much 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, Huang *et al.* [178] present a sparse denoising autoencoder to learn LR-to-HR mappings with PANs and transfer it to HSIs. Masi *et al.* [179] employ the SRCNN [24] and incorporate several maps of nonlinear radiometric indices for boosting performance. Wei *et al.* [180] propose a much deeper DRPNN based on residual learning [95] and achieve higher spatial-spectral unified accuracy. Recently, Qu *et al.* [181] jointly train two encoder-decoder networks to perform SR on PANs and HSIs, respectively, and transfer the SR knowledge in the PAN domain to the HSI domain by sharing the decoder and applying constraints such as angle similarity loss and reconstruction loss.

### 5.4 Video Super-resolution

In terms of 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 the spatio-temporal dependency, including explicit motion compensation (e.g., optical flow algorithms, learning-based methods) and recurrent methods, etc.

Among the methods based on optical flow algorithms, Liao *et al.* [182] employ various optical flow methods to generate HR candidates and ensemble them by CNNs. VSRnet [183] and CVSRnet [184] implement motion compensation by Druleas algorithm [185], and uses CNNs to take successive frames as input and predict HR frames. While Liu *et al.* [186], [187] 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 [188] utilizes a trainable spatial transformer [166] to learn motion compensation based on adjacent frames, and enters multiple frames into a spatio-temporal ESPCN [82] for end-to-end prediction. And Tao *et al.* [189] 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 [190], [191] employs a bidirectional framework, and uses CNN, RNN, and conditional CNN to model the spatial, temporal and spatial-temporal dependency, respectively. Similarly, STCN [192] uses a deep CNN and a bidirectional LSTM [193] to extract spatial and temporal information. And FRVSR [194] uses previously inferred HR estimates to reconstruct the subsequent HR frame by two deep CNNs in a recurrent manner.

In addition to the above works, the FAST [195] exploits the compact description of the structure and pixel correlations extracted by compression algorithms, transfers the SR result from one frame to adjacent frames, and accelerates the state-of-the-art SR algorithms by 15 times with little performance loss (0.2dB). And Jo *et al.* [196] generate dynamic upsampling filters and the HR residual image based on the local spatio-temporal neighborhood of each pixel, and also avoid explicit motion compensation.

### 5.5 Other Applications

Deep learning based super-resolution is also adopted to other domain-specific applications and shows great performance. Specifically, the RACNN [197] employs SR models for enhancing the discriminability of LR image details for fine-grained classification. Similarly, the Perceptual GAN [198] addresses the small object detection problem by super-resolving representations of small objects, achieving similar characteristics as large objects and more discriminative for detection. And the FSR-GAN [199] 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, Dai *et al.* [7] verify the effectiveness and usefulness of SR technology in several vision applications, including edge detection, semantic segmentation, digit and scene recognition. Huang *et al.* [200] develop RS-DRL specifically for

super-resolving remote sensing images. And Jeon *et al.* [201] utilize a parallax prior in stereo images to reconstruct HR images with sub-pixel accuracy in registration.

## 6 Conclusion and Future Directions

Image super-resolution based on deep learning have made breakthroughs in recent years. In this paper, we have given a extensive survey on recent advances in image super-resolution with deep learning. We mainly discussed the improvement of supervised super-resolution and unsupervised super-resolution, 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 research trends for the future evolution. We hope that this survey not only provides a better understanding of image super-resolution 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 data 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 HR images. It is promising to combine local and global information for providing contextual information of different scales for super-resolution.

*Combining Low- and High-level Information.* Shallow layers in deep CNNs tend to extract low-level features such as colors and edges, while deeper layers extract higher-level representations like the object identities. Thus combining low-level details with high-level abstract semantics can be of great help for HR reconstruction.

*Context-specific Attention.* Different contexts focus on different information for SR. For example, the grass area may be more concerned with colors and textures, while the animal body area may focus more on the hair details. Therefore, incorporating attention mechanism to exploit contextual information to enhance the attention to key features facilitates the generation of realistic details.

*Lightweight Architectures.* Existing SR modes tend to pursue ultimate performance, while ignoring the model size and inference speed. For example, the EDSR [33] takes 20s for $4\times$ SR per image of DIV2K [44] on a Titan GTX [202], and DBPN [61] takes 35s for $8\times$ SR [87]. Such long prediction time is unacceptable in practical applications, thus lightweight architectures are imperative. How to reduce model sizes and speed up prediction while maintaining performance remains a problem.

*Upsampling Layers.* Although upsampling operations play a very important role for super-resolution, existing methods (Sec. 3.2) have more or less disadvantages: the interpolation-based methods result in expensive computation and cannot be end-to-end learned, the transposed convolution produces checkerboard artefacts, and the sub-pixel layer brings uneven distribution of receptive fields. Hence, how to perform effective and efficient upsampling still needs to be studied, especially with high scaling factors.

### 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 [33], [34], [142]. 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 metrics cannot accurately measure model performance, researchers will have great difficulty verifying improvements. Metrics for super-resolution face such challenges and need more exploration.

*More Accurate Metrics.* The most widely used metrics for super-resolution are PSNR and SSIM. However, the PSNR tends to result in excessive smoothing, and the results often vary wildly between almost indistinguishable images. The SSIM [62] performs evaluation in terms of brightness, contrast and structure, but still cannot measure image perceptual quality accurately [10], [27]. Besides, the MOS is closest to human visual response, but takes a lot of manpower and effort and is non-reproducible. 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 of 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 factors, missing paired LR-HR images, etc. Below we'll introduce some directions towards real-world scenarios.

*Dealing with Various Degradation.* Real-world images tend to suffer unknown degradation, such as additive noise, compression artefacts and blurring, etc. Thus the models trained on datasets conducted by manual degradation often perform poorly in real-world scenarios. Some works have been proposed for solving this problem [41], [126], [128], [144], 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, face recognition, object tracking, medical imaging, scene rendering, etc.

*Multi-scale Super-resolution.* Most existing SR models perform SR with a fixed scaling factor, but in real-world scenes we often need to perform SR with arbitrary scaling factors. Hence, it is also a potential direction to develop a single model performing multi-scale super-resolution.

## 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] F. Lin, C. Fookes, V. Chandran, and S. Sridharan, "Super-resolved faces for improved face recognition from surveillance video," in *ICB*, 2007.

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

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

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

[8] H. Zhang, D. Liu, and Z. Xiong, "Convolutional neural network-based video super-resolution for action recognition," in *FG*, 2018.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

[26] 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.

[27] 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.

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

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

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

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

[32] 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.

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

[34] 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.

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

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

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

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

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

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

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

[42] 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.

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

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

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

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

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

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

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

[50] 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.

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

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

[53] 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.

[54] 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.

[55] 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.

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

[57] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop," *Arxiv:1506.03365*, 2015.

[58] K. Ma, Z. Duanmu, Q. Wu, Z. Wang, H. Yong, H. Li, and L. Zhang, "Waterloo exploration database: New challenges for image quality assessment models," *IEEE Transactions on Image Processing*, vol. 26, 2017.

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

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

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

[62] 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.

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

[64] 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.

[65] 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.

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

[67] 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.

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

[69] 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.

[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] H. R. Sheikh, A. C. Bovik, and G. De Veciana, "An information fidelity criterion for image quality assessment using natural scene statistics," *IEEE Transactions on Image Processing*, vol. 14, 2005.

[76] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," in *ICASSP*, 2004.

[77] 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.

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

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

[82] 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.

[83] 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.

[84] W. Dong, L. Zhang, G. Shi, and X. Wu, "Nonlocal back-projection for adaptive image enlargement," in *ICIP*, 2009.

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

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

[87] 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.

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

[89] C.-Y. Yang and M.-H. Yang, "Fast direct super-resolution by simple functions," in *ICCV*, 2013.

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

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

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

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

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

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

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

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

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

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

[100] 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.

[101] 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.

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

[103] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, 2015.

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

[105] 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.

[106] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *CVPR*, 2018.

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

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

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

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

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

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

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

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

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

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

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

[118] 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.

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

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

[121] 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.

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

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

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

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

[126] 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.

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

[128] 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.

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

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

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

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

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

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

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

[136] 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.

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

[138] 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.

[139] 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.

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

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

[142] 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.

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

[144] 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.

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

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

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

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

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

[150] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon, "Efficient regression of general-activity human poses from depth images," in *ICCV*, 2011.

[151] 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.

[152] 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.

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

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

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

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

[157] M. Ni, J. Lei, R. Cong, K. Zheng, B. Peng, and X. Fan, "Color-guided depth map super resolution using convolutional neural network," *IEEE Access*, vol. 5, 2017.

[158] W. Zhou, X. Li, and D. Reynolds, "Guided deep network for depth map super-resolution: How much can color help?" in *ICASSP*, 2017.

[159] Y. Xiao, X. Cao, X. Zhu, R. Yang, and Y. Zheng, "Joint convolutional neural pyramid for depth map super-resolution," *Arxiv:1801.00968*, 2018.

[160] 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.

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

[162] 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.

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

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

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

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

[167] X. Yu and F. Porikli, "Hallucinating very low-resolution unaligned and noisy face images by transformative discriminative autoencoders," in *CVPR*, 2017.
[168] Y. Song, J. Zhang, S. He, L. Bao, and Q. Yang, "Learning to hallucinate face images via component generation and enhancement," in *IJCAI*, 2017.
[169] C.-Y. Yang, S. Liu, and M.-H. Yang, "Hallucinating compressed face images," *IJCV*, vol. 126, 2018.
[170] X. Yu and F. Porikli, "Ultra-resolving face images by discriminative generative networks," in *ECCV*, 2016.
[171] 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.
[172] X. Yu, B. Fernando, R. Hartley, and F. Porikli, "Super-resolving very low-resolution face images with supplementary attributes," in *CVPR*, 2018.
[173] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *Arxiv:1411.1784*, 2014.
[174] 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.
[175] Y. Fu, Y. Zheng, I. Sato, and Y. Sato, "Exploiting spectral-spatial correlation for coded hyperspectral image restoration," in *CVPR*, 2016.
[176] B. Uzkent, M. J. Hoffman, and A. Vodacek, "Real-time vehicle tracking in aerial video using hyperspectral features," in *CVPRW*, 2016.
[177] B. Uzkent, A. Rangnekar, and M. J. Hoffman, "Aerial vehicle tracking by adaptive fusion of hyperspectral likelihood maps," in *CVPRW*, 2017.
[178] W. Huang, L. Xiao, Z. Wei, H. Liu, and S. Tang, "A new pan-sharpening method with deep neural networks," *GRSL*, vol. 12, 2015.
[179] G. Masi, D. Cozzolino, L. Verdoliva, and G. Scarpa, "Pansharpening by convolutional neural networks," *Remote Sensing*, vol. 8, 2016.
[180] Y. Wei, Q. Yuan, H. Shen, and L. Zhang, "Boosting the accuracy of multispectral image pansharpening by learning a deep residual network," *GRSL*, vol. 14, 2017.
[181] Y. Qu, H. Qi, and C. Kwan, "Unsupervised sparse dirichlet-net for hyperspectral image super-resolution," in *CVPR*, 2018.
[182] R. Liao, X. Tao, R. Li, Z. Ma, and J. Jia, "Video super-resolution via deep draft-ensemble learning," in *ICCV*, 2015.
[183] 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.
[184] ——, "Super-resolution of compressed videos using convolutional neural networks," in *ICIP*, 2016.
[185] M. Drulea and S. Nedevschi, "Total variation regularization of local-global optical flow," in *ITSC*, 2011.
[186] 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.
[187] 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.
[188] 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.
[189] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *ICCV*, 2017.
[190] Y. Huang, W. Wang, and L. Wang, "Bidirectional recurrent convolutional networks for multi-frame super-resolution," in *NIPS*, 2015.
[191] ——, "Video super-resolution via bidirectional recurrent convolutional networks," *TPAMI*, vol. 40, 2018.
[192] J. Guo and H. Chao, "Building an end-to-end spatial-temporal convolutional network for video super-resolution," in *AAAI*, 2017.
[193] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional lstm networks for improved phoneme classification and recognition," in *ICANN*, 2005.
[194] M. S. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-recurrent video super-resolution," in *CVPR*, 2018.
[195] Z. Zhang and V. Sze, "Fast: A framework to accelerate super-resolution processing on compressed videos," in *CVPRW*, 2017.
[196] 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.
[197] D. Cai, K. Chen, Y. Qian, and J.-K. Kämäräinen, "Convolutional low-resolution fine-grained classification," *Pattern Recognition Letters*, 2017.
[198] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *CVPR*, 2017.
[199] W. Tan, B. Yan, and B. Bare, "Feature super-resolution: Make machine see more clearly," in *CVPR*, 2018.
[200] N. Huang, Y. Yang, J. Liu, X. Gu, and H. Cai, "Single-image super-resolution for remote sensing data using deep residual-learning neural network," in *ICONIP*, 2017.
[201] 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.
[202] 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.

**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 an Associate Professor of the School of Information Sytems, Singapore Management Unviersity, 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.

## APPENDIX A

For better reading of this survey, we provide all the notations used in this survey and their detailed definitions in Table 2. And we also list the full text of all the abbreviations used herein in Table 3.

TABLE 2
Notations.

| Notation | Description |
|---|---|
| $I_x$ | LR image |
| $I_y$ | ground truth HR image, abbreviated as $I$ |
| $\hat{I}_y$ | reconstructed HR image, abbreviated as $\hat{I}$ |
| $I_s$ | randomly sampled HR image from the real HR images |
| $I(i)$ | intensity of the $i$-th pixel of image $I$ |
| $D$ | discriminator network of GAN |
| $\phi$ | image classification network |
| $\phi^{(l)}$ | extracted representations on $l$-th layer by $\phi$ |
| vec | vectorization operation |
| $G^{(l)}$ | Gram matrix of representations on $l$-th layer |
| $l$ | layer of CNN |
| $h$, $w$, $c$ | width, height and number of channels of feature maps |
| $h_l$, $w_l$, $c_l$ | width, height and number of channels of feature maps in $l$-th layer |
| $\mathcal{D}$ | degradation process |
| $\delta$ | parameters of $\mathcal{D}$ |
| $\mathcal{F}$ | super-resolution process |
| $\theta$ | parameters of $\mathcal{F}$ |
| $\otimes$ | convolution operation |
| $\kappa$ | convolution kernel |
| $\downarrow$ | downsampling operation |
| $s$ | scaling factor |
| $n$ | Gaussian noise |
| $\varsigma$ | standard deviation of $n$ |
| $z$ | a random vector |
| $\mathcal{L}$ | loss function |
| $\mathcal{L}_{\text{content}}$ | content loss |
| $\mathcal{L}_{\text{cycle}}$ | content consistency loss |
| $\mathcal{L}_{\text{pixel\_l1}}$ | pixel L1 loss |
| $\mathcal{L}_{\text{pixel\_l2}}$ | pixel L2 loss |
| $\mathcal{L}_{\text{pixel\_Cha}}$ | pixel Charbonnier loss |
| $\mathcal{L}_{\text{gan\_ce\_g}}$, $\mathcal{L}_{\text{gan\_ce\_d}}$ | adversarial loss of the generator and discriminator based on cross entropy |
| $\mathcal{L}_{\text{gan\_hi\_g}}$, $\mathcal{L}_{\text{gan\_hi\_d}}$ | adversarial loss of the generator and discriminator based on hinge error |
| $\mathcal{L}_{\text{gan\_ls\_g}}$, $\mathcal{L}_{\text{gan\_ls\_g}}$ | adversarial loss of the generator and discriminator based on least square error |
| $\mathcal{L}_{\text{TV}}$ | total variation loss |
| $\Phi$ | regularization term |
| $\lambda$ | trade-off parameter of $\Phi$ |
| $\epsilon$ | small instant for stability |
| $\mu_I$ | luminance of image $I$, i.e., mean of intensity |
| $\sigma_I$ | contrast of image $I$, i.e., standard deviation of intensity |
| $\sigma_{I,\hat{I}}$ | covariance between images $I$ and $\hat{I}$ |
| $\mathcal{C}_l$, $\mathcal{C}_c$, $\mathcal{C}_s$ | comparison function of luminance, contrast, structure |
| $\alpha$, $\beta$, $\gamma$ | weights of $\mathcal{C}_l$, $\mathcal{C}_c$, $\mathcal{C}_s$ |
| $C_1$, $C_2$, $C_3$ | constants |
| $k_1$, $k_2$ | constants |
| $L$ | maximum possible pixel value |
| $N$ | number of pixels |
| $M$ | number of bins |

TABLE 3
Abbreviations.

| Abbreviation | Full name | Abbreviation | Full name |
|---|---|---|---|
| FH | face hallucination | PAN | panchromatic image |
| HR | high-resolution | SR | super-resolution |
| HSI | hyperspectral image | TV | total variation |
| HVS | human visual system | WT | wavelet transformation |
| LR | low-resolution | | |
| FSIM [77] | feature similarity | MS-SSIM [74] | multi-scale SSIM |
| IFC [75] | information fidelity criterion | PSNR | peak signal-to-noise ratio |
| IQA | image quality assessment | SSIM [62] | structural similarity |
| MOS | mean opinion score | VIF [76] | visual information fidelity |
| MSSIM [62] | mean SSIM | | |
| BN [140] | batch normalization | GAN [26] | generative adversarial net |
| CNN | convolutional neural network | LSTM | long short term memory network |
| CycleGAN [134] | cycle-in-cycle GAN | ResNet [95] | residual network |
| DenseNet [86] | densely connected CNN | SENet [102] | squeeze-and-excitation network |
| FAN | face alignment network | SPMC [189] | sub-pixel motion compensation |
| ADRSR [144] | automated decomposition and reconstruction | LapSRN [29], [69] | Laplacian pyramid SR network |
| Attention-FH [110] | attention-aware FH | LCGE [168] | learn FH via component generation and enhancement |
| BRCN [190], [191] | bidirectional recurrent CNN | MDSR [33] | multi-scale deep SR system |
| CARN [30] | cascading residual network | MemNet [59] | memory network |
| CARN-M [30] | CARM based on MobileNet | MS-LapSRN [69] | multi-scale LapSRN |
| CBN [163] | cascaded bi-network | MTUN [164] | multi-task upsampling network |
| CinCGAN [126] | cycle-in-cycle GAN | MWCNN [119] | multi-level wavelet CNN |
| CNF [99] | context-wise network fusion | ProSR [34] | progressive SR |
| CVSRnet [184] | compressed VSRnet | RACNN [197] | resolution-aware CNN |
| DBPN [61] | deep back-projection network | RCAN [70] | residual channel attention networks |
| DNSR [144] | denoising for SR | RDN [94] | residual dense network |
| DRCN [80] | deeply-recursive CNN | RS-DRL [200] | remote sensing deep residual-learning network |
| DRPNN [180] | deep residual PAN-sharpening network | SFT-GAN [48] | GAN with spatial feature transformation |
| DRRN [60] | deep recursive residual network | SICNN [72] | super-identity CNN |
| DSRN [83] | dual-state recurrent network | SRCNN [24], [25] | SR CNN |
| DWSR [117] | deep wavelet prediction for SR | SRGAN [27] | SR GAN |
| EDSR [33] | enhanced deep SR network | SRDenseNet [79] | SR DenseNet |
| EDSR-PP [113] | EDSR with pyramid pooling | STCN [192] | spatial-temporal CNN |
| ESPCN [188] | efficient sub-pixel CNN | TDAE [167] | transformative discriminative auto-encoder |
| ESRGAN [101] | enhanced SRGAN | TDN [165] | transformative discriminative network |
| FAST [195] | free adaptive SR via transfer | Super-FAN [32] | SR with FAN |
| FRVSR [194] | frame-recurrent video SR | UR-DGN [170] | ultra-resolving by discriminative generative networks |
| FSRCNN [45] | fast SRCNN | VESPCN [188] | video ESPCN |
| FSR-GAN [199] | feature SRGAN | VSRnet [183] | video SR network |
| FSRNet [73] | face SR network | ZSSR [81] | zero-shot SR |