

A Conspectus of Deep Learning Techniques for Single-Image Super-Resolution

Garima Pandey^{a,*} and Umesh Ghanekar^{a,**}

^a Department of Electronics and Communication, National Institute of Technology Kurukshetra, Kurukshetra, 136119 India

* e-mail: garima_6160006@nitkkr.ac.in

** e-mail: ugnitk@nitkkr.ac.in

Abstract—Single image super-resolution (SISR) is one of the contemporary research areas in the field of image restoration that involves solving an ill-posed inverse equation. A rich profusion of techniques has been proposed in the past four decades. However, the expansion of deep learning (DL) in recent years has improved image reconstruction drastically. In this paper, a compendium of DL applications in the SISR field has been provided, mainly focusing on inspection of the latest advancements and categorization. For completeness of the study, different aspects of DL based SISR are included in brief with a synoptic study on the available image datasets. In conclusion, issues existing in DL-based SISR and viable solutions for them are proffered.

Keywords: single image super-resolution, deep learning, image database, convolutional neural network, generative adversarial network

DOI: 10.1134/S1054661822010059

1. INTRODUCTION

SISR attempts to resolve the problem of bad quality images that are generally obtained from the acquisition systems. It intends to create a high-resolution (HR) image from the available low-resolution (LR) image through software processing. Image estimation in SISR comes under the category of mathematically ill-posed inverse problems which necessitates strenuous efforts for its solving and optimization. A plenitude of methods is proposed over the years for SISR [124, 125], which ranges from the involvement of simple interpolation equations to optimization through different machine learning (ML) techniques. The main challenge in learning-based processing is to determine the underlying structure of the HR image by estimating the relationship that can be established between the available LR image to HR image. In the case of large data processing and the requirement of higher dimensional mapping, the performance of simple ML-algorithms deteriorates. In such scenarios, the application of deep learning (DL) for the estimation of HR image is a good option. Therefore, the research community is shifting from simple ML to DL for better performances.

DL [82] is a subset of ML in artificial intelligence (AI) which involves learning of high-level abstraction by using a hierarchical structure of mapping from input to output. In DL, *deep* refers to the number of

layers that are used to transform data from input to output. It is one of the trendy tools for researchers active in the field of image processing, computer vision, gaming, pattern and speech recognition, social media management, language processing, medical analysis, machine translation, etc. DL is better in estimating the mathematical mapping functions from input to output directly using the available data without much involvement of information obtained from human skills. The capacity to handle and process a huge amount of unstructured data efficiently makes DL-based techniques much more suitable for SISR implementation.

In this paper, efforts have been made to recapitulate the DL related techniques used for the SISR process. DL-based SISR techniques mainly consist of three consecutive steps; feature extraction, nonlinear mapping, and HR patch reconstruction which are discussed in detail. A summary of the different image databases that are used for the training process for HR image reconstruction is also provided in the paper. The remaining paper is continued in the following sequence. Section 2 consists of the basic knowledge of SISR, and in Section 3 major architectures of deep networks that are used in DL are provided. Section 4 comprises a systematic exposition of the available literature on DL-based SISR techniques. Section 5 includes a comparison among various DL based techniques that comprise the state-of-the-art in the SISR field. Section 6 discusses the issues related to the existing techniques and some feasible technical ideas to overcome them. Lastly, the paper has been concluded in Section 7.

Received January 30, 2021; revised August 27, 2021;
accepted August 31, 2021

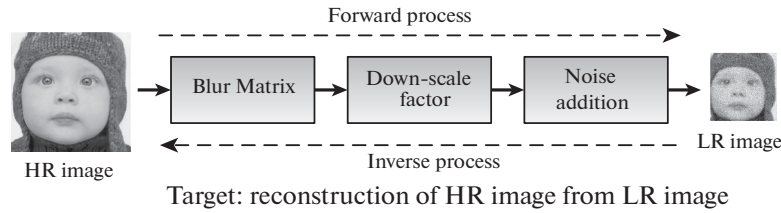


Fig. 1. Degradation model in image capturing process.

2. BACKGROUND OF SINGLE IMAGE SUPER-RESOLUTION

HR images are a prerequisite for fields like biometrics, face and pattern recognition, computer vision, forensic science, archeology, medical diagnosis and applications, surveillance and remote sensing, satellite imaging, entertainment and TV industry, consumer electronics, etc. High-quality cameras and imaging systems are available in the market to ensure the quality of images, even then sometimes quality of the images debase due to environmental disturbance and amateurish handling. Besides this, sometimes high-resolution cameras also fail to produce images of desired resolution due to their inherent physical constraints. The quality of images is generally degraded due to blurring, down-sampling, and noise addition, as shown in Fig. 1. As shown in Fig. 1, it is the case of an inverse problem. The target for researchers is to estimate the HR image from the available LR image. This process is known as super-resolution (SR) and when only one LR image in the input, it is called SISR. SISR process can be mathematically modeled as follows.

$$L = BSH + N, \quad (1)$$

where “ H ” is the HR image and “ L ” is the LR image obtained due to blurring “ B ”, down-scaling “ S ” and noise adding “ N ”.

The high utility of HR images in everyday life has captivated researchers to work in the field of SISR. A surfeit of literature is present in this area with some mainstay concepts such as neighbor embedding [18], sparse coding [178], random forest [63, 108], and convolutional neural network [26] developed over the years. The existing SISR techniques can be classified into interpolation-based, reconstruction-based, and learning-based. In interpolation-based SISR, HR image is obtained by imposing the LR image on the grid of HR image and predicting the missing pixels of the HR image through the neighborhood pixels. Techniques like linear interpolation (bilinear [43], nearest neighbor), non-linear interpolation (cubic [15], bi-cubic, spline [115], fractal [111, 199]), edge directed interpolation [94] (geometric duality, partial differential equation), and adaptive interpolation [179], etc., are used. These techniques are fast in generating HR

images, however, fail in preserving edges and creating new high spectral information. Reconstruction-based SISR techniques use statistical calculations and knowledge of prior information for the construction of the HR image. Different types of image priors such as sparsity prior [29, 92, 95], nonsparsity prior [168], parametric prior [96], non-parametric prior [155], local prior (profile prior [149], spatially adaptive prior [58, 169], generic prior [69], soft-edge smoothness prior [109], textural prior [138]), semi-local prior [78], and nonlocal prior [41] are used for the reconstruction propose. In reconstruction based techniques, performance deteriorates severely with increasing scaling factor and fails in creating new high spectral information, as well.

Learning-based processing have indelible effects in the field of SISR owing to their success in creating new high-frequency details and constructing better HR images in comparison to interpolation and reconstruction based processing. In this process, HR image is either estimated by calculating each pixel separately or estimated by constructing the HR patches from the LR patches and combining them to form the complete HR image. Patch processing involves training of a machine through a dictionary learning algorithm and then using it for approximation of a HR image. Dictionary learning can be performed either using input image or through external sources i.e., set of standard images. In this field, some of the worth mentioning techniques are neighbor embedding based SISR, sparse coding based SISR, regression based SISR, random forest based SISR, DL based SISR, etc.

Neighbor embedding (NE) is one of the elementary technique in the field of SISR. Technique proposed in [18] is based on the concept of forming manifolds between LR and HR patches due to similar inherent geometrical structure. Over the years, different NE-based techniques are proposed such as non-negative NE [12], partially supervised NE [194], NE with self-similar image prior [205], low rank matrix recovery NE [21], sparse NE [37], anchored NE [160], partially supervised anchored NE [150], and rotation expanded dictionary (RED) [89]. These techniques help in lowering the computations that result in fast execution. Also, it requires lesser number of patches as exemplars for reconstruction process. In

case of higher dimensionality and complex geometry, performance of NE-based techniques exacerbate because of their inability to predict the underlying structure.

Sparse coding (SC) entails delineation of image patches in sparse form. It requires over-complete dictionaries for representation of patches and falls under the category of unsupervised learning. Objective function generally used for SC is given by:

$$\tilde{H} = BSH - L_2^2 + \lambda\theta(H), \quad (2)$$

where second part is the constraint term applied to minimize the set of possible solutions. In constraint term θ is the regularization operator and λ is the constant value. One of the prominent algorithm was proposed in [178], which is improved over the years by introducing different variants based on the concept of sparse prior [2, 4, 29, 88, 92, 93, 95, 96, 118, 160, 164].

In regression-based SISR, regression equations [17, 38, 51, 68, 70, 107, 156, 192] are learned with the help of exemplars to map from LR to HR image. It can have supervised or semi-supervised learning models to evaluate regression equations. Some of the variants proposed by researchers are greedy regression [157], support vector regression [121], local regression [57], [188], fast regression [181], kernel ridge regression [24, 155], global regression [22], etc. These techniques sometime produce undesirable ringing effects due to overfitting of regression equations. Random forest (RF) is based on formation of decision trees [62, 87, 133] for training process and helps in overcoming the overfitting problem of simple regression techniques.

The basic ML-based SISR algorithms are mostly computationally complex and generate inferior quality HR images due to involvement of huge amount of raw data and complex geometrical structure. The DL based techniques have an edge over these techniques and are widely used by the researchers, since they utilize the deep network architecture for improving the reconstruction accuracy. A brief description of DL architectures is provided in the upcoming section which has been used for designing of different DL based SISR techniques.

3. DEEP LEARNING AND ITS ARCHITECTURE

DL [8, 9] is based on representation learning [7, 9] through artificial neural network (ANN) [10, 112, 136] that can be performed through anyone of the following methods: (a) supervised learning, (b) semi-supervised learning, and (c) unsupervised learning. It uses multi-layer structure for extracting the higher level features from the available input data. In multilayering, each layer is responsible for converting the input information to a higher level of abstraction and composite representation i.e., it has a substantial credit assignment path depth at each layer. DL is employed by using dif-

ferent neural networks (NNs). NN is defined as sequence of algorithms that are used for representing the underlying relations from input to output by using different artificial nodes (neurons). The best part of using NN is that it can easily adapt to the change in input without changing the complete setup and can produces better results in comparison to others. Based on the depth, i.e., the number of hidden layers, it can be shallow or deep. Shallow NN has only one hidden layer besides an input and an output layer in its framework, while deep NN has more than one hidden layers in its framework. Deep networks are better in extracting features; beside, for same computational power, they are better than shallow networks. But increase in number of hidden layers is directly proportional to computational complexity, which is undesirable. Deep networks [152] are further classified into cyclic and acyclic networks. Cyclic networks have some connecting intermediate states between input and output which can be simple in nature, i.e., one to one, or can be interacting in nature from each other. Acyclic NNs have feed-forward structure and are mostly utilized by researchers for implementing DL algorithms. Techniques [129] under this category are further classified in following groups.

(a) Unsupervised pretrained network: This network typically uses unsupervised learning for processing of data which can be done through three different architectures such as autoencoder, deep belief network (DBN), and generative adversarial network (GAN). Autoencoder learns to represent data through an encoder and a decoder. It is used for dimensionality reduction and feature learning, which is achieved through different regularized and variational encoders. Regularized autoencoders use models such as sparse, denoising, and contractive autoencoders for capturing information, whereas variational autoencoders use generative models for the same. Generally, autoencoders have restrictions during output approximations because of identity model generation. DBN is a generative graphical model used for probabilistic reconstruction of its input consisting of interconnected multilayers from hidden units having no connections between the nodes within the layer. DBNs are generally trained by restricted Boltzmann machines or autoencoders. A restricted Boltzmann machine is generative energy based undirected model involving contrastive divergence for subnetworks modeling that results in better modeling of training data. GANs [39] are used for efficient representation of training data that results in better estimation of output with the help of training data. Beside unsupervised training, GAN can be used for semi-supervised and supervised learning. GANs are good for parallel processing and consist of discriminators and generative networks. In generative networks, deconvolutional layers are used to find various features through different activation functions.

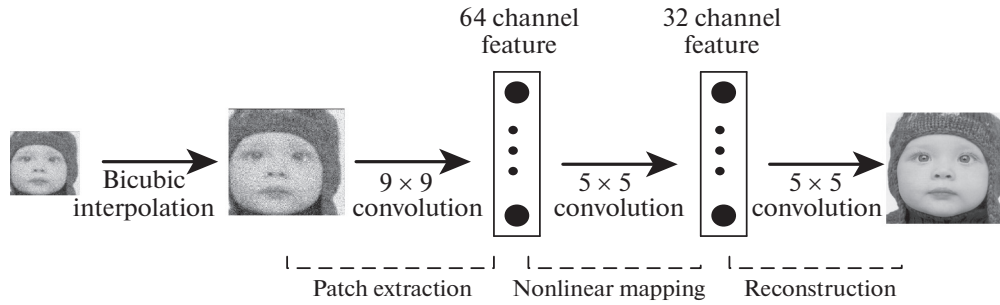


Fig. 2. Basic block diagram of DL based SISR.

(b) Recurrent neural network: This network involves recurrent connections in the nodes on temporal basis and retains the state during modeling of input data. It utilizes back-propagation through time for capturing information related to time. It is used for both sequential and parallel computations.

(c) Recursive neural network: In [47], a set of weights are calculated recursively for a structured variable-size input to obtain a structured or scalar estimation of the output in a topological order. Architecture of recursive neural network consists of a binary tree and a shared-weight matrix for allowing network to compute the weights through back-propagation. It is generally used for sequential learning and decision trees. It is good in handling granular and hierarchal structures and as GANs can also be used for semi-supervised and supervised learning.

(d) Convolutional neural network (CNN): CNN is one of the best among all existing DL techniques for image related processing such as reconstruction, segmentation, retrieval, detection, and classification. A simple architecture that is used in CNN, generally, consists of a set of hidden layers that transform the information through nonlinear feature extraction process. Feature extraction process involves (i) convolution operation, (ii) activation function that is generally taken as rectified linear unit (ReLU), and (iii) pooling, i.e., automatic learning of features by constructing higher order features with the help of a set of attributes obtained from the images provided in the input layer. Ability of automatic feature extraction in CNN [76] reduces the requirement of separate network for this purpose. Features extracted through CNN are mostly rotation invariant [197], which is beneficial in reconstruction process.

4. DL BASED SISR

DL based SISR algorithms basically consist of three main steps: (a) patch extraction and representation, (b) nonlinear mapping, and (c) reconstruction. In patch extraction and representation, overlapping patches are formed from the LR image and are represented by feature maps that consist of high-dimensional vectors. Nonlinear mapping involves conver-

sion of one set of feature map to another set of feature map for extracting better information. This process occurs through a set of hidden layers. At last, all the reconstructed HR patches are combined to form the final HR image. A generalized block diagram depicting this process is shown in Fig. 2.

As shown in Fig. 2, generally, NN architectures that are used for image reconstruction have images of same size in the input and output. Thus, some pre-processing or upscaling is performed on the images before feeding them in the neural models. A brief note on upscaling is provided in Subsection 4.1.

4.1. Upscaling Step

Upscaling is done through different approaches as shown in Fig. 3.

Upscaling is either performed by using NN or without NN. Generally, in upscaling without NN, various variants of interpolation techniques are used such as bicubic and Lanczos interpolation. However, these techniques suffer from smoothing of images [31]. To overcome this, researchers try to design a layer in the NN architecture. It involves either upscaling in one-step or gradual. Mostly, gradual upscaling is performed through deconvolution [190] or through transpose convolution [30].

Techniques proposed in [28, 142] use deconvolution that involves either pixel mapping or subpixel mapping followed by convolutional operator for upscaling. Deconvolution layer [143] helps in reducing computations and is used in the end of the NN architecture. In [28], nearest neighbor interpolation is used for pixel mapping. But nearest neighbor interpolation results in repetition of points in feature map in both the directions, which is minimized by using subpixel mapping [142]. In this, feature map is expanded for increasing resolution by rearranging the points in a specific manner. Besides deconvolution, transpose convolution [182] is also used for gradual upscaling. This consists of automatic upscaling kernels and is used for alleviating the artifacts.

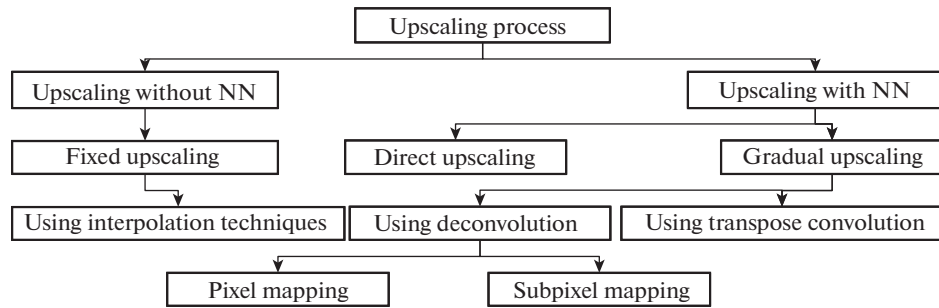


Fig. 3. Different process of upscaling.

For implementing upscaling and different NN architectures, different on-line packages are used. A list of available on-line toolbox packages that are used in SISR are provided in Subsection 4.2.

4.2. Toolbox Packages

Various toolboxes are used by researchers for implementing the architecture that is used for DL based SISR. Some of the available toolboxes are listed below.

(a) MatConvNet deep learning toolbox [166]: It has many CNN architectures useful for various image processing applications. It is simple and efficient in implementation and provides a user friendly environment through deep integration. Algorithms proposed, e.g., in [19, 180], use this toolbox for execution of SISR techniques.

(b) Caffe [65]: It provides framework for DL, specially focusing on expressions, speed, and modularity, and is widely used for startup prototypes, research projects, large-scale industrial applications in speech, multimedia, and vision. In this, optimization and models do not require hard-coding for configuration. Algorithms proposed, e.g., in [20, 26, 28, 135, 137, 182, 183], use this toolbox for execution of SISR techniques.

(c) TensorFlow [1]: It is an open source and end-to-end programmable software library for ML. It provides differentiable programming for data-flow and has flexible architecture that allows ease in computations on different platforms such as tensor processing unit, graphics processing unit, and central processing unit. It also allows communication among desktops to servers such as mobiles and edge devices. Algorithms proposed, e.g., in [36, 105, 131, 139, 176], use this toolbox for execution of SISR techniques.

(d) Compute unified device architecture (CUDA) convnet package [76]: It is used for convolutional neural network implementations and is capable of performing parallel operations through hardware features for general purpose functioning on graphics process-

ing units. It is also capable of implementing the dropout regularization used in neural networks. Algorithms proposed, e.g., in [26], use this toolbox for execution of SISR techniques.

(e) PyTorch [128]: It provides a framework for ML and cloud platforms and also provides different tools and libraries for NN that helps the researcher from research prototyping to its production deployment. It facilitates scaling and seamless transition from graph mode and provides scalable distributed network training for back-end programming. The entire package available in PyTorch can only be used on mini-batches of the input samples. Algorithms proposed, e.g., in [86, 97, 98, 139], use this toolbox for execution of SISR techniques.

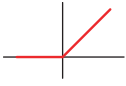
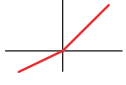
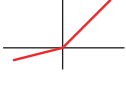
In these toolboxes, for implementing the NNs different activation functions are used for transferring information in encoded pixels of different layers. For DL based SISR, mostly rectified linear unit (ReLU) and its variants, given in Table 1, are used as activation functions because of their simplicity in implementation and better performance.

In DL based SISR algorithms, NNs are initially trained through different big datasets, and, then, their performances are validated on some new test images. A brief of image datasets used by different researchers for this purpose is provided in Subsection 4.3.

4.3. Image Datasets

Different types of image datasets used in SISR can be classified into various categories as shown in Fig. 4. These sets are used for training and testing of the algorithms. In training process, images from the datasets are used to calculate the values of various parameters of NNs and during testing images are used to verify the correctness of the estimated parameters to find the accuracy of the proposed algorithm. As shown in Fig. 4, datasets are formed by either having same type of images, i.e., solitary dataset, or amalgamation of different varieties of images, i.e., hybrid dataset. Images like natural, text, faces, or artificial are used to create the datasets. These images are used in different forms for formation of datasets such as silhouettes,

Table 1. Activation functions used in DL based SISR algorithms

Sr. no.	Activation function	Formula	Features
1.	ReLU [55]	$f(x) = \begin{cases} 0 & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases}$ 	1. Activated sparsely that helps in better prediction and less overfitting. 2. Does not suffer from vanishing gradient like tanh or sigmoid. 3. Economical to use, since it has less mathematical complexity. Takes less time for training and converges faster. 4. Used in [19, 48, 54, 66, 71, 134, 137, 142, 153, 183, 201].
2.	Leaky ReLU [159]	$f(x) = \begin{cases} 0.01x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$ 	1. Reduces “dying problem” of ReLU by having a small value of slope for negative value. 2. It learns faster than ReLU and fastens up the training process. 3. Used, e.g., in [79].
3.	Parametric ReLU [52]	$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$ 	1. It is a variant of leaky ReLU with adaptive slope, which is estimated by the network as a parameter. 2. In PReLU $\alpha \leq 1$ and is equivalent to $f(x) = \max(x, \alpha x)$ [53]. 3. Used in algorithms proposed, e.g., in [28, 49, 97, 129, 182].

segmented or complete images. Count of images in a dataset can vary from a single digit to seven digits number. Some of the available datasets in the literature are provided as follows.

(1) Set5 [13]: Consist of 5 images: baby, bird, butterfly, women, and head.

(2) Set14 [191]: Consist of 14 images: zebra, ppt3, pepper, monarch, man, lenna, foreman, flowers, face, comic, coastguard, bridge, barbara, and baboon.

(3) Berkeley Segmentation Dataset (BSD) [114]: Consist of BSD300, BSD500, and BSD200. It is the collection of different segmented images.

(4) Diverse 2K (DIV2K) dataset [159]: Contains 1000 (800 for training set and 200 for test set) RGB images of 2K resolution.

(5) LabelMe [132]: Contains partially and fully annotated images in training set and fully labeled

images in test set. It has images of objects, cars, trees, sky, buildings, persons, etc.

(6) Canadian Institute for Advanced Research dataset (CIFAR10/CIFAR100) [75]: Contains natural images of size 32×32 divided into 10/100 categories.

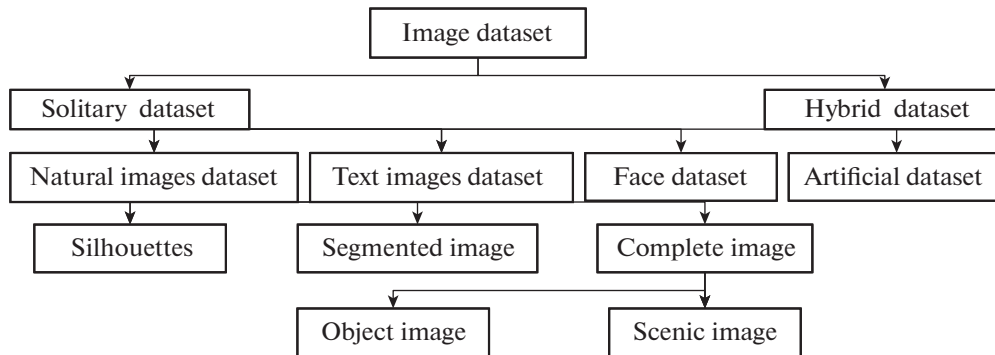
(7) Handwritten digit MNIST dataset [81]: Contains handwritten digits in form of 60000 examples in training set and 10000 examples in test set.

(8) Common Objects in Context (COCO) dataset [91]: Contains 2.5 million labeled object instances in 328k images.

(9) ILSVRC 2013 ImageNet [25]: Contains 3.2 million of images.

(10) Urban scenes [61]: Contains 100 real-world images having self-similarity.

(11) Sun-Hays80 [148]: Contains six million internet images with detailed statistical models of images.

**Fig. 4.** Types of image datasets.

(12) Facial Recognition Technology (FERET) face data set [44]: Consists of 1564 sets of face images (365 duplicate and 1199 individuals).

(13) Manga 109 [35]: Contains 109 variety of Japanese comic books.

(14) eBDtheque [45]: Contains comic dataset having detailed metadata.

(15) Lsun [186]: Consists of more than one million images of scenes and object categories.

(16) Coil100 [119]: Contains around 7200 color images having 100 objects (72 images for each object) comprising complex reflectance and geometric characteristics.

(17) Visual Genome [74]: Is a portal containing around 100000 images that include buildings, people, etc.

(18) Open Images Dataset [73]: Contains around 9 million images in thousands of groups.

(19) Corel image database [103]: Contains 10000 images in JPEG format of 100 categories.

(20) Labelled Faces in the Wild [59]: Contains around 13000 images of face photographs from the web.

(21) Stanford Dogs Dataset [67]: Consists images of different breeds (120) of dogs.

(22) Indoor Scene Recognition [130]: Contains around 15000 images having different indoor scenes under 67 indoor.

(23) Kodak Photo dataset [34]: Contains 24 lossless color images.

(24) Google's Open Images [126]: Contains 600 classes of objects with around 1000000 training images, 41000 validation images, and 125000 test images.

(25) Caltech 101 [33]: Contains images of objects in 101 categories.

(26) NORB dataset [80]: Contains around 190000 stereo paired images.

(27) Laboratory for Image and Video Engineering (LIVE) database [140]: Contains distorted images.

(28) Yahoo Flickr Creative Commons 100 Million Dataset (YFCC100M) [158]: Contains around 100 million objects (images and videos) for media.

(29) Extended Yale Face Database [84]: Consists of around 160 gray-scale images of human faces in different mood.

(30) Texture database [170]: Contains 136 images of different textures.

The dataset listed above is either used directly or created by selecting images from different datasets such as in [26] images from Set5, Set14, and BSD200 are used as test images, and images from ImageNet and Timofte dataset [164] are used as training images.

Timofte dataset is formed by the combination of all the images from Set5, Set14 and 91 training images available in [178]. Datasets used by some of the algorithms constituting state-of-the-art are provided in Section 5.

Different NN parameters are learned through these datasets by optimizing different objective functions. To achieve this, various loss functions are used for minimizing difference between the estimated image, H , and corresponding HR image, \tilde{H} . A brief note on loss functions and their types are explained in Subsection 4.4.

4.4. Loss Functions

A machine is trained through minimization of loss function. It helps in better fitting of the equation for modeling the image reconstruction task. Based on the uses, loss functions can be divided into two types: (a) loss function for classification and (b) loss function for regression. Former is used for categorization of the class of a specific sample in the output, whereas later is used for predicting a continuous value. In SISR, regression loss function and its variants are used. Types of regression loss functions that are used in DL based SISR are shown in Fig. 5 and explained further.

(a) L_2 loss function: It is calculated by averaging the squared differences between estimated values and original values, as given by Eq. (3).

$$L(\Theta) = \frac{1}{n} \sum_{i=1}^n F(L_i; \Theta) - H_{i2}^2. \quad (3)$$

Here, $F(L_i; \Theta)$ are the reconstructed images at different levels for corresponding original image H and n is the number of samples. It is widely used in SISR because of simple implementation and ease in computation of gradients. However, it poorly correlates with complex characteristics of human visual systems (HVSs), since it considers noise to be independent of the characteristics and decreases with deterioration in perceptual quality [14]. Also, because of squaring during error calculations, large differences are heavily penalized in comparison to a small difference, which causes distortion in local structures and underlying textures [173, 202].

(b) L_1 loss function: It is calculated by averaging the sum of absolute differences between estimated values and original values, as given by Eq. (4).

$$L(\Theta) = \frac{1}{n} \sum_{i=1}^n F(L_i; \Theta) - H_{i1}. \quad (4)$$

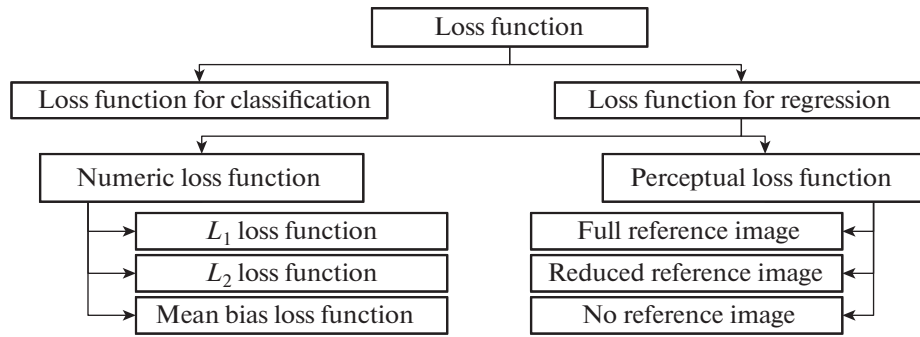


Fig. 5. Types of loss functions.

Like L_2 , L_1 also does not consider directions during calculations, but, in comparison to former, latter gives better results in presence of outliers. Besides, it provides better minimum value than former and thus results in higher convergence rate [90]. But, unlike L_2 , implementation of L_1 requires more complicated software and tools for calculations of gradients.

(c) Mean bias loss function: It is calculated by averaging the sum of differences between estimated values and original values, as given by Eq. (5).

$$L(\Theta) = \frac{1}{n} \sum_{i=1}^n (F(L_i; \Theta) - H_i). \quad (5)$$

Compared to previous two loss functions, it is rarely used in DL domain. However, this loss function is used for determining the bias required, i.e., positive or negative, by the network at different nodes.

(d) Perceptual loss: Above discussed loss functions significantly improve the performance in terms of PSNR, but sometimes fail in improving the perceptual qualities of the images [184]. To overcome this, perceptual loss functions are used for optimization. It can be of three types: (a) full reference image, (b) reduced reference image, and (c) no reference image. Full reference image perceptual loss function [141, 145, 180] requires complete information about the reference image, whereas no reference image perceptual loss function [5, 32, 113] estimates the quality of distorted image without any reference image. No reference image perceptual loss functions are more practical in real scenario than full reference image perceptual loss functions, but are also comparatively more challenging to implement. To balance the trade-off between them, reduced reference image perceptual loss function [110, 175] is used. It requires partial knowledge about the reference image.

In this section, till here, different aspects of DL based SISR algorithms such as different tools and activation functions used for implementing NNs, various databases used for training and testing of the algo-

rithms, and loss functions used for optimization, have been discussed. Now, a broad categorization of existing algorithms is done in Subsection 4.5.

4.5. Classification of DL Based SISR

Figure 6 presents a general classification of the techniques proposed so far in the field of DL based SISR.

Based on mode of operation, techniques can be classified in frequency domain or spatial domain. In former, frequency components obtained from the images are used partially or completely for reconstruction of images, whereas in latter, complete process is carried out in the spatial domain only.

In frequency domain [46, 60, 104, 176, 204] data input to the neural networks are generally obtained by applying different transforms, such as discrete cosine transform (DCT), Haar, and wavelet, on the actual images. These transforms are used to decompose the input image into different levels of the frequency components for getting better details like texture and edge information of the image. In [46] and [176], wavelet transform is combined with residual network [54] for predicting residual wavelet subbands to produce sparsity in the output. This leads to stable training process and better convergence rate. In [104], pooling is replaced by wavelet transform for availing the advantage of its inverse property. Wavelet transform is completely invertible, i.e., no information is lost during the inversion process. Frequency domain operations are good in processing the high and low frequency components of the image separately, but results in excessive number of decompositions of features and computational cost. Also, it causes impeding of gradient back-propagation in the process of training because of excessive formation of redundant channels.

Although frequency domain NN processing has many advantages, time domain operations are more favored by the researchers because of their simplicity and better control over various parameters. In this, NNs are either trained internally or externally. In internal training [116, 144, 162], network is trained by

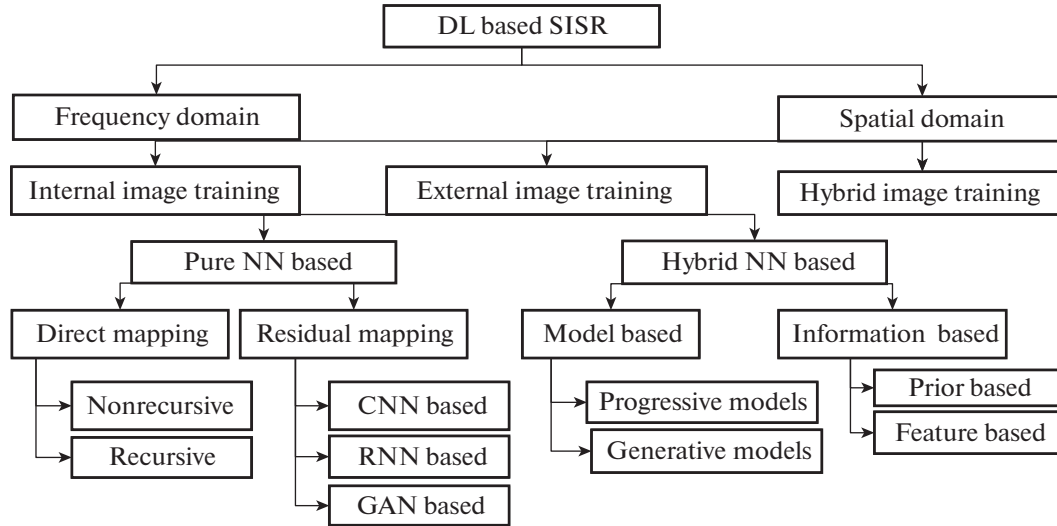


Fig. 6. Classification of DL based SISR algorithms.

the input image only obtained by scaling at different levels. However, in external training, firstly, networks are trained by a set of images and then are validated on different images. Internal training is based on the notion that the desired information can be acquired by scaling the image at different factors. One of the specialty of the internal training [193] is that search process is not constrained to certain regions, and it is also capable of handling specific features that sometimes are not available in external training. Besides, it is robust to repetitive features present in the image and requires small-scale NN architecture for its implementation. However, internal training sometimes causes distortion in fine detailing of the image, mostly in the edge regions [162]. Thus, some researchers tried to combine the internal training with external training to avail the benefits of both [19, 162].

Compared to internal training, external training gives bigger receptive field for utilizing the deep NN architectures. External training facilitates better estimation of nonlinear mapping. Here, simple NN architecture can be directly applied for image reconstruction through end-to-end mapping, or it can be merged with other representative models for image reconstruction. Both the orientations of HR image reconstruction are explained further.

4.5.1. Pure NN based DL for SISR. In simple NN based DL, different NN networks such as CNNs and GANs are used for reconstruction of HR image. In this, training of the network is either achieved by direct mapping from inputs to outputs or through training of the residuals. A brief note of this is provided below.

(a) Direct mapping: HR image in direct mapping [26–28, 77, 97] is obtained by learning an end-to-end mapping with the help of LR and HR images available

for training. It can be done either by using simple (nonrecursive blocks) or recursive blocks in the NN architectures.

(i) Nonrecursive blocks: In this, simple feed-forward networks are used for direct mapping. Algorithm proposed in [26] is one of the pioneer and pertinent work in this field. It uses three-layer CNN for nonlinear mapping, where filter size of first layer is $64 \times 1 \times 9 \times 9$, second layer is $32 \times 64 \times 5 \times 5$, and last layer is $1 \times 32 \times 5 \times 5$. Despite bearing a simple architecture, it proves its superiority over traditional algorithms, but requires huge amount of time for training. To make the training process faster without sacrificing the performance, in [77], instead of taking bicubic interpolated LR image as input, image obtained through zero component transform [75] is taken as input. This helps in enhancing the optimization process and also improves the edges and textures in the output HR image. In this algorithm polynomial NN (PNN) [120] is also used for learning the mapping in hierarchical and nonlinear form, which helps in reducing the required training samples. Algorithm proposed in [28] improvised the technique proposed in [142] by having a deconvolutional layer at the end. Nonrecursive architectures are comparatively simple to implement but require bulky architectures for having performances on par with others.

(ii) Recursive blocks: In SISR, reconstructing finer details are very significant, and in nonrecursive process it is generally achieved by increasing either the width or depth of the networks. This causes increase in computational complexity by introducing more and more parameters for calculations. Thus, possibility of overfitting increases and, simultaneously, expansion in network results in storage and retrieval problems. To subdue these problems, recursive blocks are used in the networks. In [97], up-down sampling blocks are

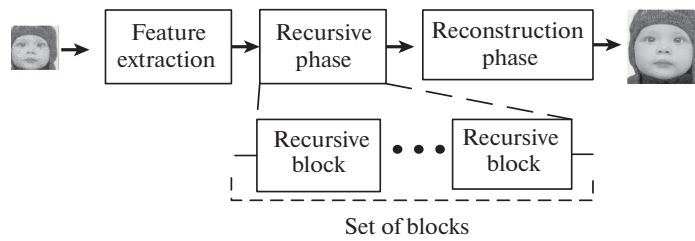


Fig. 7. Basic block diagram of recursive network for SISR.

recursively used for better modeling. A generalized block diagram of recursive network is shown in Fig. 7. In general performance of recursive networks are enhanced by increasing its depth through stacking of recursive blocks. Recursive blocks can be used in architectures such as CNN and GAN, and besides direct mapping, it is also used for residual mapping.

Direct mapping is easier to implement but requires huge storage and computations. It also suffers from exploding/vanishing gradient problems [11, 40], resulting into slow convergence. Thus, researchers give more emphasis on residual learning.

(b) Residual mapping: In SISR, the inherent geometrical structures of the LR image and the HR image are basically similar; thus, instead of using complete image for training, networks are trained by a residual image, i.e., matrix obtained by subtracting the LR and the estimated HR image. In residual learning [54], along with input and original image, residual estimates are required at each node for optimization. It is one of the groundbreaking research in DL group and is also employed in SISR [16, 36, 72, 75, 83, 85, 86, 90, 99, 131, 153, 167–170, 183]. Residual learning is practically effectuated in SISR by making use of variants of CNNs, recurrent NNs (RNNs) and GANs.

(i) CNN based: One of the remarkable basic algorithm utilizing residual learning through CNN architecture was proposed in [72]. Inspired by deep CNN architecture [106, 151] and network proposed in [71], it uses 20 convolution layers including input and output. In [64], two types of features are mixed through compression unit for reducing the number of filters for each layer that results in faster execution. Some researchers also tried to pre-activate the network by introducing ReLU followed by convolutional layer in every residual unit [153, 188]. In [153], a network consists of approximately 52 layers is proposed that learns the residuals for both local and global features extraction. To improve the learning capacity of the network through residues, in [183] two-phase residual network is proposed. In this, firstly, hierarchical features are learned and, then, these features are used to learn high-frequency components in the next phase. Some other developments in CNN architectures include denser network [177], network in network concept [163], residual channel attention network

[200], widely activated deep network [187], cascading residual network [3], information distillation network [56] and so on. In spite of having promising results, CNNs mostly suffer from overfitting problems in case of weak database and also stuck to local minima during optimization process.

(ii) RNN based: To increase the performance of CNNs, mostly the depth of the networks are increased [117], but deeper networks contain large number of parameters, and calculating these parameters with limited data during training results in noise generation [146, 147] and increase in complexity. To deprecate this problem, researchers tried to utilize the RNN with CNN architecture [42, 54, 85] for reconstruction of HR images [48, 50, 71]. In RNNs, downsampling operator sometimes leads to information/memory loss, which is minimized in [154] by using memory networks for reconstruction of the image. In [100], non-local blocks are introduced in RNN to avail the benefits of nonlocal property, but it results in higher computations. In RNN, performance can be enhanced just by increasing the recursion depth for same number of hidden layers. Training process in RNN is simplified by introducing skip connections and recursive supervisions [71]. CNNs and RNNs almost share same training process and their architectures can be used in different combinations. These networks perform well in terms of PSNR, performs poorly in terms of perceptual qualities.

(iii) GAN based: Adversarial networks have significant results in producing realistic images in comparison to CNNs and RNNs, whereas have more complexity in their training process. It consists of two parts: (i) a generator that learns to estimate the values of new pixels and (ii) a discriminator that compares between the estimated and the real values of the pixels [39]. In [83], GAN was introduced for SISR implementation, in which generative network scales up the input image to actual size of HR image, and discriminative network distinguishes between the original and the estimated HR image through pixel-wise assessment metric. Some of the GAN based algorithms are proposed, e.g., in [66, 134, 139, 172, 189, 198]. Performance of GANs is improved by introducing additional loss functions such as perceptual loss [66], pixel-wise loss [134], adversarial loss [198], and texture transfer loss [66]. Some researchers have also

tried to upgrade the process by employing additional discriminator [127] or by training through two-phases of pre- and adversarial training [198]. Training process in GAN based SISR algorithms, generally, face the vanishing gradient problem and mode collapse issue [189, 203].

4.5.2. Hybrid NN based DL for SISR. For better interpretation and exploitation of image information and models, many researchers have also tried to take advantage of neural architecture along with the representative methods used for SISR. Representative methods utilize different type of information that can be drawn from images to tackle specific problems of SISR. Brief notes on its types are discussed below.

(a) Information based: In this, along with the information that are extracted by the NN networks itself, some additional information is also provided for better reconstruction of HR image. This additional information can be based on (a) prior and (b) features.

(i) Prior based: Trailblazing of priors in DL leads to desired output by providing flexibility in handling different cases. Some of the recently used priors in NN architecture are sparse prior, semantic prior, statistical prior, etc. In [101, 174], sparse prior is introduced in the network through sparse coding and is trained in cascaded form. Sparse prior implementation facilitates efficient training and also reduces the size of model. In [171], semantic prior [161] is used for better texture reconstruction. In [195], denoiser prior is used in training for effective and fast CNN through model based optimization process. This is mostly used for different low level image applications. In [165], statistical prior are learned from huge number of exemplars from low level images. NNs combined with suitable priors perform better in qualitative, as well as, quantitative measures, however choice of appropriate priors is highly image dependent which acts as obstacles for developing generalized SISR algorithms.

(ii) Feature based: Different features of the images are directly used for better edge preservation and image reconstruction. In [185], edge map of LR image is used along with the LR image as an input for sharp edge. In [171], spatial feature transform is used to generate affine transformation parameters for better texture recovery. In [196] dimensionality stretching is used for multiple and spatial degradations. Training the NNs with ‘extra’ external information lead to improve their performances, but like prior based these systems also suffer from image dependency.

(b) Model based: Here, different models that are specifically utilizing various patterns of images are incorporated with different NN architectures for SISR implementation. It can be further divided into (i) Progressive model, and (ii) Generative model.

(i) Progressive models: Progressive models [23, 79, 102, 167, 185] are used for gradually generating the HR image. Algorithm proposed in [185] uses residual network combined with sub-bands for progressive recon-

struction whereas algorithm proposed in [79] uses Laplacian pyramid for generating HR image at different hierarchical scales. In [102, 167], ensemble learning is used for expanding CNN through adding an extra model with adaptive weights, resulting into greater flexibility of design. Structure tensor is another choice for to make weights adaptive as shown in [102]. Progressive models are good for adaptive scaling i.e. same network can be used for different scaling factors, however suffer from higher complexity.

(ii) Generative models: Deep generative models [122, 123] are good choice in case scaling factor is very high. Compared to traditional networks, generative networks use conditional maximum likelihood prediction through directional graphic model for generating HR image.

5. COMPARISONS BETWEEN DIFFERENT SISR TECHNIQUES

Precedence among different SISR algorithms are shown through measurable parametric indexes such as mean square error (MSE), peak signal to noise ratio (PSNR), structural similarity index (SSIM), feature similarity index (FSIM), etc. Pictorial display is also used for comparing different algorithms. Here, only two parametric indexes, i.e., PSNR and SSIM given by Eqs. (6) and (8), respectively, are used for comparison.

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) dB, \quad (6)$$

where MSE is defined as

$$MSE = \frac{1}{k_1 k_2} \sum_{x=1}^{k_1} \sum_{y=1}^{k_2} (H(x, y) - \tilde{H}(x, y))^2. \quad (7)$$

In this, H is the actual HR image and \tilde{H} is the estimated HR image of dimension $k_1 \times k_2$ each.

$$SSIM(H, \tilde{H}) = \frac{(2\mu_H \mu_{\tilde{H}} + \rho_1)(\sigma_{H\tilde{H}} + \rho_2)}{(\mu_H^2 + \mu_{\tilde{H}}^2 + \rho_1)(\sigma_H^2 + \sigma_{\tilde{H}}^2 + \rho_2)}. \quad (8)$$

Here, μ_H and $\mu_{\tilde{H}}$ are the mean value of image H and \tilde{H} , respectively. Similarly, σ_H and $\sigma_{\tilde{H}}$ are the variance value of image H and \tilde{H} , respectively. In this equation, ρ_1 and ρ_2 are constant terms.

In Table 2, algorithms proposed in [26, 49, 71, 79, 142, 153, 172, 201] are compared for images available in datasets Set5 and Set14. These algorithms involve different NN architectures as shown in Fig. 8 for implementing DL in SISR and constitute the state of art. Stimulations are performed only on luminance (Y) component that is obtained by transforming RGB images to YCbCr images. The other two components (Cb and Cr) are resized through bicubic interpolations. LR test image is obtained through applying blur-

Table 2. Comparison among DL based SISR techniques for $S = 2$ and $S = 4$ in terms of PSNR(dB) and SSIM (PSNR(dB)/SSIM)

Techniques	Scale	Test Images		Training Set	Loss function
		Set 5	Set 14		
SRCNN [26]	$\times 2$	36.65/0.954	32.29/0.903	ImageNet subset	L_2
	$\times 4$	30.49/0.862	27.61/0.754		
VDSR [71]	$\times 2$	37.63/0.958	33.04/0.911	BSD200 + 91 images [178]	L_2
	$\times 4$	31.53/0.885	28.02/0.767		
ESPCN [142]	$\times 2$	37.34/0.943	32.98/0.909	ImageNet subset	L_2
	$\times 4$	30.90/0.868	27.89/0.756		
DRRN [153]	$\times 2$	37.74/0.959	33.23/0.913	BSD200 + 91 images [178]	L_2
	$\times 4$	31.68/0.888	28.21/0.772		
LapSRN [79]	$\times 2$	37.52/0.959	33.23/0.913	BSD200 + 91 images [178]	Charbonnie [6]
	$\times 4$	31.54/0.885	28.19/0.772		
ESRGAN [172]	$\times 2$	37.90/0.972	32.12/0.941	DIV2K	L_1 , Perceptual loss
	$\times 4$	32.10/0.913	27.51/0.800		
RDN [201]	$\times 2$	38.30/0.961	34.10/0.921	DIV2K	L_1
	$\times 4$	32.61/0.900	28.92/0.789		
DBPN [49]	$\times 2$	38.15/0.956	33.99/0.919	DIV2K + ImageNet	L_2
	$\times 4$	32.47/0.898	29.32/0.782		

ring operation on corresponding HR image with average mask of size 5×5 , followed by downscaling with a factor S . Here, experimental results are obtained for S equal to 2 and 4.

For qualitative analysis various algorithms given in Table 2 are performed on a test image “Girl” and their outcomes is shown in Fig. 9. From the figure it can be observed that image obtained from algorithm proposed in [172] is best for visual perceptions; however, its PSNR is comparatively less than for algorithms proposed in [49, 201].

6. CHALLENGES IN EXISTING DL BASED SISR

Even though DL based SISR algorithms have gained popularity in research community due to their promising results, still there remain many unresolved issues which are of major concern for researchers. Some of them are listed as follows.

(a) The depth of neural architecture is one of the key factor for enhancing its performance. Disproportionate increase in its depth results in enormous calculations and intolerable vanishing gradient problem. Hence, optimization of depth in NN is desirable.

(b) Customization of DL architectures are desirable in accordance with priors used for training.

(c) Finding an appropriate loss function which can improve perceptual quality and low distortions, simultaneously, is also required.

Efforts that can be accomplished in future for mollifying the existing shortcomings for improved estimation of HR image are described in next part.

6.1. Future Trends

Shortcomings of the existing algorithms can be alleviated either by implementing different tweaks in existing techniques or by proposing new techniques. Some of the concepts that can be incorporated in the SISR algorithms in future are as follows.

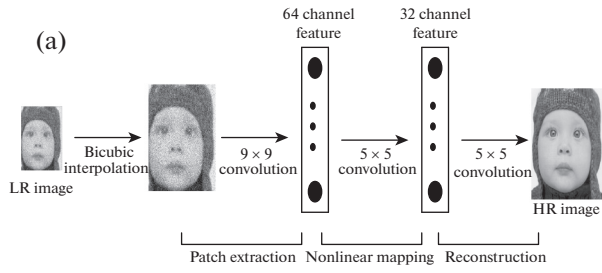
(a) Efforts can be made for forming lighter structured architectures and concepts such as dropout [146] can be introduced in architectures used in SISR for lowering the computations.

(b) Instead of shallow networks with higher number of nodes in each layer, researchers may try networks with more layers but lesser number of nodes in each layer. This may help in reducing the number of unknown parameters in the networks without compromising their performance.

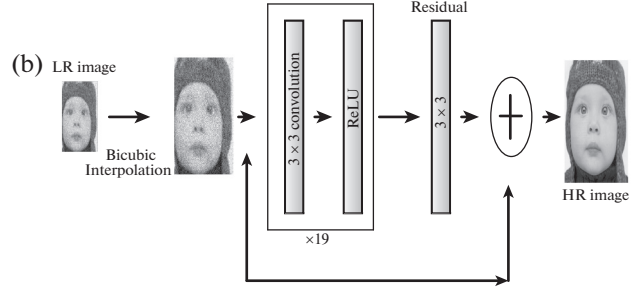
(c) DL networks are just like a magical boxes in the research community, thus, it should be explored further to grab benefits of the other prospects of these techniques. Other networks that are shown in Fig. 3 can also be used in SISR which are not investigated, yet.

(d) Objective functions incorporating different priors (used in traditional SISR [125]) can be framed for training of the neural networks.

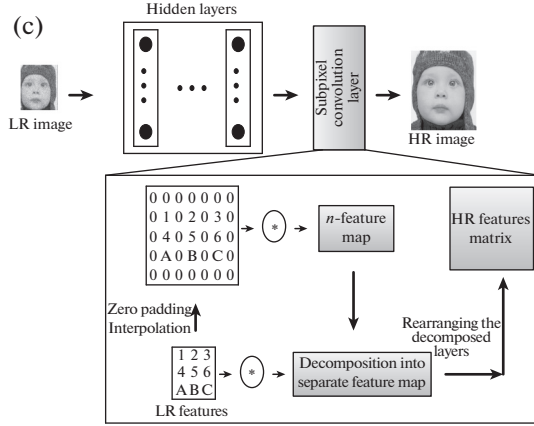
Super-resolution convolutional neural network (SRCNN) [26]



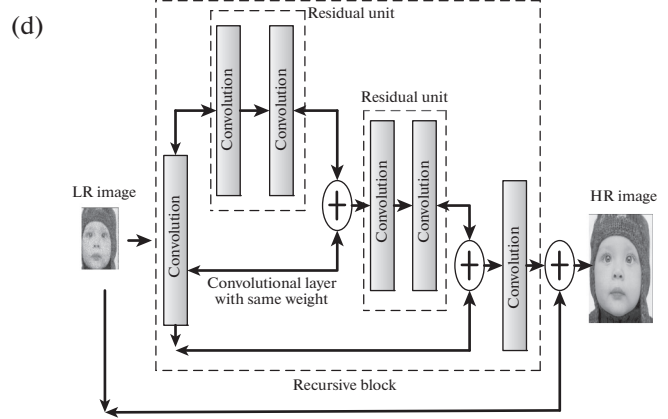
Very deep super-resolution (VDSR) [71]



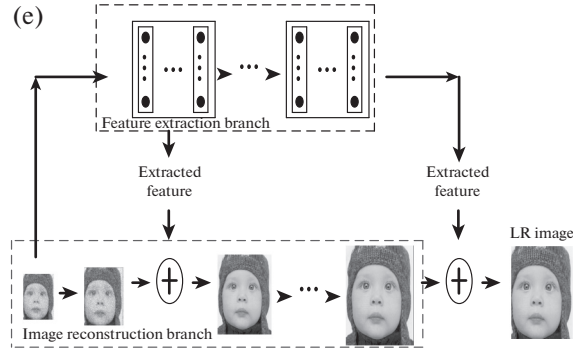
Efficient subpixel convolutional neural network (ESPCN)



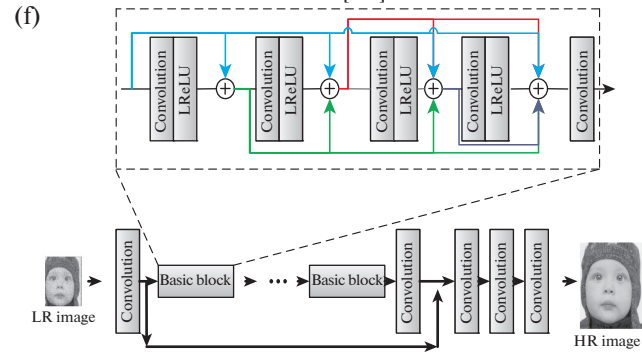
Deep recursive residual network (DRRN) [153]



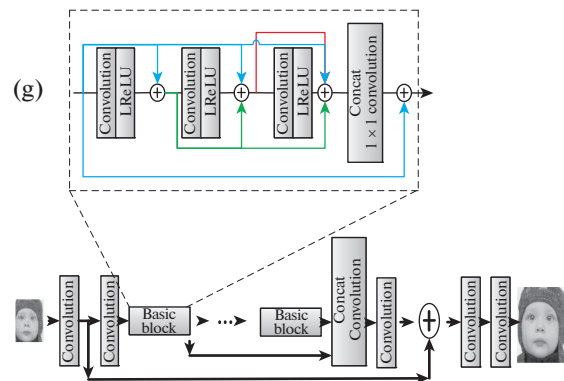
Laplacian pyramid super-resolution network (LapSRN) [79]



Enhanced super-resolution generative adversarial network (ESRGAN) [172]



Residual dense network (RDN) [201]



Deep back-projection networks (DBPN) [49]

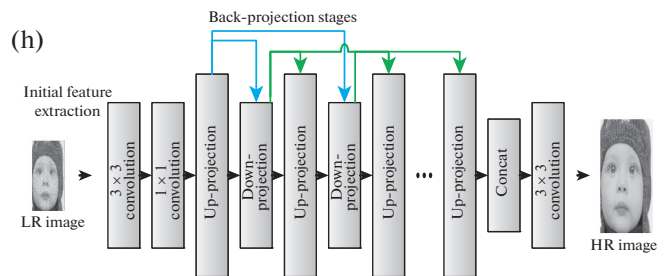


Fig. 8. Architecture of different techniques.

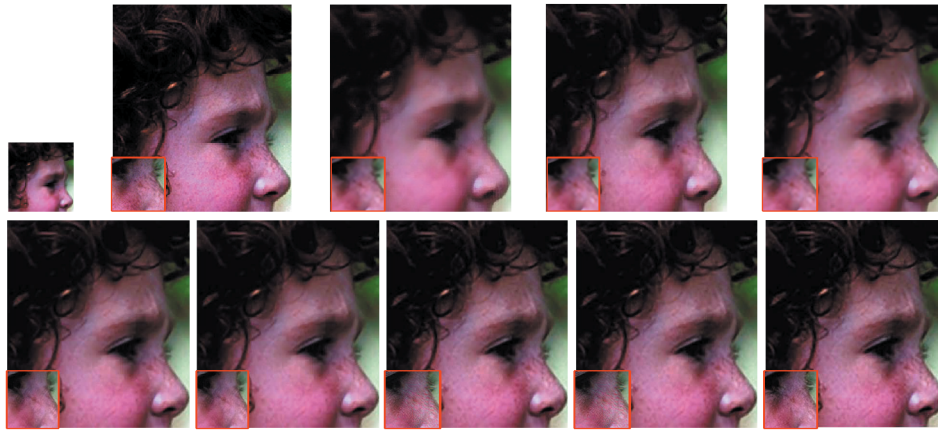


Fig. 9. Images obtained from different DL based SISR techniques on “Girl” image ($S = 4$); 1st row (left to right): LR input image, original HR image, SRCNN [26], VDSR [71], ESPCN [142]; 2nd row (left to right): DRRN [153], LapSRN [79], ESRGAN [172], RDN [201], DBPN [49].

(e) Loss functions can be redesigned by including extra parameters for perceptual and numeric assessment.

7. CONCLUSIONS

In this survey paper, a critical study has been made for existing DL based SISR algorithms, summing up their various aspects such as different benchmarks of image dataset, architectures involved in DL and the loss functions used for optimizations. Before exploring the DL based SISR algorithms, a short note on traditional techniques is also provided for better understanding of the area. In existing DL based SISR techniques, researchers have mostly used CNN and GAN architectures for estimation of HR images. In these architectures, residual learning is preferred over direct mapping, since it leads to lesser number of computations and storage. Performance of CNNs and RNNs are better in terms of MSE and PSNR, whereas GANs are better in term of perceptual quality. Despite the propitious performances of existing techniques, reconstruction of finer details such as edges and textures, are still substantial issues and require further improvements in edge preserving techniques. In implementation of DL based SISR techniques, researchers are also concerned regarding selection of better architecture, modeling of priors and features, forming balance among the accuracy of HR image reconstruction, computational complexity and time efficiency, etc.

The paper denouement with the desire that agglomeration and assessment of existing DL based techniques provided here will facilitate the researchers to get enlightened in this area and will encourage to work in future.

COMPLIANCE WITH ETHICAL STANDARDS

This article is a completely original work of its authors; it has not been published before and will not be sent to other publications until the *PRIA* Editorial Board decides not to accept it for publication.

Conflict of Interest

The authors declare that they have no conflicts of interest.

REFERENCES

1. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. J. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. G. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. A. Tucker, V. Vanhoucke, V. Vasudevan, F. B. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous distributed systems,” arXiv:1603.04467
2. R. Abiantun, F. J. Xu, U. Prabhu, and M. Savvides, “SSR2: Sparse signal recovery for single-image super-resolution on faces with extreme low resolutions,” *Pattern Recognit.* **90**, 308–324 (2019). <https://doi.org/10.1016/j.patcog.2019.01.032>
3. N. Ahn, B. Kang, and K. Sohn, “Fast, accurate, and, lightweight superresolution with cascading residual network,” arXiv:1803.08664
4. S. Ayas and M. Ekinici, “Single image super resolution using dictionary learning and sparse coding with multi-scale and multi-directional Gabor feature representation,” *Inf. Sci.* **512**, 1264–1278 (2020). <https://doi.org/10.1016/j.ins.2019.10.040>
5. B. Bare, K. Li, B. Yan, B. Feng, and C. Yao, “A deep learning based no-reference image quality assessment model for single-image super-resolution,” in *IEEE Int.*

- Conf. on Acoustics, Speech and Signal Processing (ICASSP), Calgary, Canada, 2018* (IEEE, 2018), pp. 1223–1227.
<https://doi.org/10.1109/icassp.2018.8461931>
6. J. T. Barron, “A more general robust loss function,” arXiv:1701.03077
 7. Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” *IEEE Trans. Pattern Anal. Mach. Intell.* **35**, 1798–1828 (2013). <https://doi.org/10.1109/TPAMI.2013.50>
 8. Y. Bengio, “Deep learning of representations for unsupervised and transfer learning,” in *Proc. of the 2011 Int. Conf. on Unsupervised and Transfer Learning Workshop, Washington, 2011*, Ed. by I. Guyon, G. Dror, V. LeMaître, G. Taylor, and D. Silver (JMLR.org, 2011), pp. 17–37.
 9. Y. Bengio, A. C. Courville, and P. Vincent, “Unsupervised feature learning and deep learning: A review and new perspectives,” arXiv:1206.5538
 10. Y. Bengio, “Learning deep architectures for AI,” *Found. Trends Mach. Learn.* **2** (1), 1–127 (2009). <https://doi.org/10.1561/22000000006>
 11. Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Trans. Neural Networks* **5**, 157–166 (1994). <https://doi.org/10.1109/72.279181>
 12. M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel, “Neighbor embedding based single-image super-resolution using semi-nonnegative matrix factorization,” in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, 2012* (IEEE, 2012). <https://doi.org/10.1109/icassp.2012.6288125>
 13. M. Bevilacqua, A. Roumy, C. Guillemot, and M. A. Morel, “Low complexity single-image super-resolution based on nonnegative neighbor embedding,” in *Proc. British Machine Vision Conf., 2012*, Ed. by R. Bowden, J. Collomosse, and K. Mikolajczyk (BMVA Press, 2012). <https://doi.org/10.5244/c.26.135>
 14. Y. Blau and T. Michaeli, “The perception-distortion tradeoff,” in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition, Salt Lake City, Utah, 2018* (IEEE, 2018), pp. 6228–6237. <https://doi.org/10.1109/CVPR.2018.00652>
 15. Z. Caiming, Z. Xin, L. Xuemei, and C. Fuhua, “Cubic surface fitting to image with edges as constraints,” in *IEEE Int. Conf. on Image Processing, Melbourne, 2013* (IEEE, 2013). <https://doi.org/10.1109/icip.2013.6738216>
 16. F. Cao and H. Liu, “Single image super-resolution via multi-scale residual channel attention network,” *Neurocomputing* **358**, 424–436 (2019). <https://doi.org/10.1016/j.neucom.2019.05.066>
 17. Y. Cao, Z. He, Z. Ye, X. Li, Y. Cao, and J. Yang, “Fast and accurate single image super-resolution via an energy-aware improved deep residual network,” *Signal Process.* **162**, 115–125 (2019). <https://doi.org/10.1016/j.sigpro.2019.03.018>
 18. H. Chang, D.-Y. Yeung, and Y. Xiong, “Super-resolution through neighbor embedding,” in *Proc. of the 2004 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, Washington, 2004* (IEEE, 2004). <https://doi.org/10.1109/cvpr.2004.1315043>
 19. J. Y. Cheong and I. K. Park, “Deep CNN-based super-resolution using external and internal examples,” *IEEE Signal Process. Lett.* **24**, 1252–1256 (2017). <https://doi.org/10.1109/lsp.2017.2721104>
 20. J. Chen, X. He, H. Chen, Q. Teng, and L. Qing, “Single image super-resolution based on deep learning and gradient transformation,” in *IEEE 13th Int. Conf. on Signal Processing (ICSP), Chengdu, China, 2016* (IEEE, 2016), pp. 663–667. <https://doi.org/10.1109/icsp.2016.7877915>
 21. X. Chen and C. Qi, “Low-rank neighbor embedding for single image super resolution,” *IEEE Signal Process. Lett.* **21**, 79–82 (2014). <https://doi.org/10.1109/lsp.2013.2286417>
 22. J. S. Choi and M. Kim, “Single image super-resolution using global regression based on multiple local linear mappings,” *IEEE Trans. Image Process.* **26**, 1300–1314 (2017). <https://doi.org/10.1109/tip.2017.2651411>
 23. R. Dahl, M. Norouzi, and J. Shlens, “Pixel recursive super resolution,” arXiv:1702.00783
 24. D. Dai, R. Timofte, and L. Van Gool, “Jointly optimized regressors for image super-resolution,” *Comput. Graphics Forum* **34**, 95–104 (2015). <https://doi.org/10.1111/cgf.12544>
 25. J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *IEEE Conf. on Computer Vision and Pattern Recognition, Miami, 2009* (IEEE, 2009). <https://doi.org/10.1109/cvpr.2009.5206848>
 26. C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” *IEEE Trans. Pattern Anal. Mach. Intell.* **38**, 295–307 (2016). <https://doi.org/10.1109/tpami.2015.2439281>
 27. C. Dong, C. C. Loy, K. He, and X. Tang, “Learning a deep convolutional network for image super-resolution,” in *Computer Vision – ECCV 2014*, Ed. by D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, *Lecture Notes in Computer Science*, vol. 8692 (Springer, Cham, 2014), pp. 184–199. https://doi.org/10.1007/978-3-319-10593-2_13
 28. C. Dong, C. C. Loy, and X. Tang, “Accelerating the super-resolution convolutional neural network,” arXiv:1608.00367
 29. W. Dong, G. Shi, Y. Ma, and X. Li, “Image restoration via simultaneous sparse coding: Where structured sparsity meets gaussian scale mixture,” *Int. J. Comput. Vision* **114**, 217–232 (2015). <https://doi.org/10.1007/s11263-015-0808-y>
 30. V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” arxiv:1603.07285
 31. N. Efrat, D. Glasner, A. Apartsin, B. Nadler, and A. Levin, “Accurate blur models vs. image priors in single image super-resolution,” in *IEEE Int. Conf. on Computer Vision, Sydney, 2013* (IEEE, 2013), pp. 2832–2839. <https://doi.org/10.1109/iccv.2013.352>

32. Y. Fang, C. Zhang, W. Yang, J. Liu, and Z. Guo, "Blind visual quality assessment for image super-resolution by convolutional neural network," *Multimedia Tools Appl.* **77**, 29829–29846 (2018). <https://doi.org/10.1007/s11042-018-5805-z>
33. L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**, 594–611 (2006). <https://doi.org/10.1109/tpami.2006.79>
34. R. Franzen, "True color Kodak images," <http://r0k.us/graphics/kodak/>.
35. A. Fujimoto, T. Ogawa, K. Yamamoto, Y. Matsui, T. Yamasaki, and K. Aizawa, "Manga109 dataset and creation of metadata," in *Proc. of the 1st Int. Workshop on coMics Analysis, Processing and Understanding, Cancun, Mexico, 2016* (Association for Computing Machinery, New York, 2016), p. 2. <https://doi.org/10.1145/3011549.3011551>
36. M. Gao, X. H. Han, J. Li, H. Ji, H. Zhang, and J. Sun, "Image super-resolution based on two-level residual learning CNN," *Multimedia Tools Appl.* **79**, 4831–4846. <https://doi.org/10.1007/s11042-018-6751-5>
37. X. Gao, K. Zhang, D. Tao, and X. Li, "Image super-resolution with sparse neighbor embedding," *IEEE Trans. Image Process.* **21**, 3194–3205 (2012). <https://doi.org/10.1109/tip.2012.2190080>
38. X. Gao, K. Zhang, D. Tao, and X. Li, "Joint learning for single-image super-resolution via a coupled constraint," *IEEE Trans. Image Process.* **21**, 469–480 (2012). <https://doi.org/10.1109/tip.2011.2161482>
39. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. of the 27th Int. Conf. on Neural Information Processing Systems, Montreal, 2014*, Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (MIT Press, Cambridge, Mass., 2014), pp. 2672–2680.
40. X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feed-forward neural networks," *Proc. Mach. Learn. Res.* **9**, 249–256 (2010).
41. W. Gong, Y. Tang, X. Chen, Q. Yi, and W. Li, "Combining edge difference with nonlocal self-similarity constraints for single image super-resolution," *Neurocomputing* **249**, 157–170 (2017). <https://doi.org/10.1016/j.neucom.2017.03.067>
42. A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Studies in Computational Intelligence (Springer, Berlin, 2011). <https://doi.org/10.1007/978-3-642-24797-2>
43. K. T. Gribbon and D. G. Bailey, "A novel approach to real-time bilinear interpolation," in *Proc. DELTA 2004. Second IEEE Int. Workshop on Electronic Design, Test and Applications, Perth, Australia, 2004* (IEEE, 2004), pp. 126–131. <https://doi.org/10.1109/delta.2004.10055>
44. R. Gross, "Face databases," in *Handbook of Face Recognition*, Ed. by S. Z. Li and A. K. Jain (Springer, New York, 2005), pp. 301–327. https://doi.org/10.1007/0-387-27257-7_14
45. C. Guérin, C. Rigaud, A. Mercier, F. Ammar-Boudjelal, K. Bertet, A. Bouju, J.-C. Burie, G. Louis, J. M. Ogier, and A. Revel, "eBDtheque: A representative database of comics," in *12th Int. Conf. on Document Analysis and Recognition, Washington, 2013* (IEEE, 2013), pp. 1145–1149. <https://doi.org/10.1109/icdar.2013.232>
46. T. Guo, H. S. Mousavi, T. H. Vu, and V. Monga, "Deep wavelet prediction for image super-resolution," in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, 2017* (IEEE, 2017), pp. 1100–1109. <https://doi.org/10.1109/cvprw.2017.148>
47. B. Hammer, A. Micheli, A. Sperduti, and M. Strickert, "Recursive self-organizing network models," *Neural Networks* **17**, 1061–1085 (2004). <https://doi.org/10.1016/j.neunet.2004.06.009>
48. W. Han, S. Chang, D. Liu, M. Yu, M. J. Witbrock, and T. S. Huang, "Image super-resolution via dual-state recurrent networks," arXiv:1805.02704
49. M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," arXiv:1803.02735
50. M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition, Salt Lake City, Utah, 2018* (IEEE, 2018), pp. 1664–1673. <https://doi.org/10.1109/cvpr.2018.00179>
51. H. He and W.-C. Siu, "Single image super-resolution using Gaussian process regression," in *CVPR 2011* (IEEE, 2011), pp. 449–456. <https://doi.org/10.1109/cvpr.2011.5995713>
52. K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," arXiv:1502.01852
53. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 2016* (IEEE, 2016), pp. 770–778. <https://doi.org/10.1109/cvpr.2016.90>
54. V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. of the 27th Int. Conf. on Machine Learning, Haifa, Israel, 2010*.
55. Z. Hui, X. Wang, and X. Gao, "Fast and accurate single image super-resolution via information distillation network," arXiv:1803.09454
56. J. Hu and Y. Luo, "Single-image superresolution based on local regression and nonlocal self-similarity," *J. Electron. Imaging* **23**, 033014 (2014). <https://doi.org/10.1117/1.jei.23.3.033014>
57. X. Hu, S. Peng, and W.-L. Hwang, "Learning adaptive interpolation kernels for fast single-image super resolution," *Signal, Image Video Process.* **8**, 1077–1086 (2014). <https://doi.org/10.1007/s11760-014-0634-7>
58. G. B. Huang, M. Ramesh, T. Berg, and E. L. Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Tech. Rep. 07–49, (University of Massachusetts, Amherst, 2007).
59. H. Huang, R. He, Z. Sun, and T. Tan, "Wavelet-SR-Net: A wavelet-based CNN for multi-scale face super resolution," in *IEEE Int. Conf. on Computer Vision (ICCV), Venice, 2017* (IEEE, 2017), pp. 1698–1706. <https://doi.org/10.1109/iccv.2017.187>
60. J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in

- IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, 2015* (IEEE, 2015), pp. 5197–5206. <https://doi.org/10.1109/cvpr.2015.7299156>
61. J.-J. Huang, T. Liu, P. L. Dragotti, and T. Stathaki, “SRHRF+: Self-example enhanced single image super-resolution using hierarchical random forests,” in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, Hawaii, 2017* (IEEE, 2017), pp. 1067–1075. <https://doi.org/10.1109/cvprw.2017.144>
 62. J. J. Huang and W. C. Siu, “Learning hierarchical decision trees for single image super-resolution,” *IEEE Trans. Circuits Syst. Video Technol.* **27**, 937–950 (2017). <https://doi.org/10.1109/tcsvt.2015.2513661>
 63. Z. Hui, X. Wang, and X. Gao, “Fast and accurate single image super-resolution via information distillation network,” in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, 2018* (IEEE, 2018), pp. 723–731. <https://doi.org/10.1109/CVPR.2018.00082>
 64. Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” arXiv:1408.5093
 65. J. Johnson, A. Alahi, and F. Li, “Perceptual losses for real-time style transfer and super-resolution,” arXiv:1603.08155
 66. A. Khosla, N. Jayadevaprakash, B. Yao, and L. Fei-Fei, “Novel dataset for fine-grained image categorization,” in *First Workshop on Fine-Grained Visual Categorization, IEEE Conf. on Computer Vision and Pattern Recognition, Colorado Springs, Colo., 2011*.
 67. J. Kim and C. Kim, “Discrete feature transform for low-complexity single image super-resolution,” in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conf. (APSIPA), Jeju, Korea, 2016* (IEEE, 2016), pp. 1–2. <https://doi.org/10.1109/apsipa.2016.7820852>
 68. K. I. Kim and Y. Kwon, “Single-image super-resolution using sparse regression and natural image prior,” *IEEE Trans. Pattern Anal. Mach. Intell.* **32**, 1127–1133 (2010). <https://doi.org/10.1109/tpami.2010.25>
 69. K. I. Kim and Y. Kwon, “Example-based learning for single-image super-resolution,” in *Pattern Recognition. DAGM 2008*, Ed. by G. Rigoll, Lecture Notes in Computer Science, vol. 5096 (Springer, Berlin, 2008), pp. 456–465. https://doi.org/10.1007/978-3-540-69321-5_46
 70. J. Kim, J. K. Lee, and K. M. Lee, “Deeply-recursive convolutional network for image super-resolution,” arXiv:1511.04491
 71. J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 2016* (IEEE, 2016), pp. 1646–1654. <https://doi.org/10.1109/cvpr.2016.182>
 72. I. Krasin, T. Duerig, N. Alldrin, V. Ferrari, S. Abu-El-Haija, A. Kuznetsova, H. Rom, J. Uijlings, S. Popov, A. Veit, S. Belongie, V. Gomes, A. Gupta, C. Sun, G. Chechik, D. Cai, Z. Feng, D. Narayanan, and K. Murphy, “Openimages: A public dataset for large-scale multi-label and multi-class image classification,” <https://github.com/openimages>.
 73. R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei, “Visual genome: Connecting language and vision using crowd sourced dense image annotations,” arXiv:1602.07332 [cs.CV]
 74. A. Krizhevsky, “Learning multiple layers of features from tiny images,” Tech. Rep. (2009).
 75. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM* **60** (6), 84–90 (2017). <https://doi.org/10.1145/3065386>
 76. N. Kumar and A. Sethi, “Fast learning-based single image super-resolution,” *IEEE Trans. Multimedia* **18**, 1504–1515 (2016). <https://doi.org/10.1109/tmm.2016.2571625>
 77. Y. Kwon, K. I. Kim, J. Tompkin, J. H. Kim, and C. Theobalt, “Efficient learning of image super-resolution and compression artifact removal with semi-local Gaussian processes,” *IEEE Trans. Pattern Anal. Mach. Intell.* **37**, 1792–1805 (2015). <https://doi.org/10.1109/tpami.2015.2389797>
 78. W. Lai, J. Huang, N. Ahuja, and M. Yang, “Deep Laplacian pyramid networks for fast and accurate super-resolution,” arXiv:1704.03915
 79. Y. LeCun, F. J. Huang, and L. Bottou, “Learning methods for generic object recognition with invariance to pose and lighting,” in *Proc. of the 2004 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, CVPR 2004, Washington, 2004* (IEEE, 2004), vol. 2. <https://doi.org/10.1109/cvpr.2004.1315150>
 80. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE* **86**, 2278–2324 (1998). <https://doi.org/10.1109/5.726791>
 81. Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature* **521**, 436–444 (2015). <https://doi.org/10.1038/nature14539>
 82. C. Ledig, L. Theis, F. Huszar, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network,” arXiv:1609.04802
 83. K. C. Lee, J. Ho, and D. Kriegman, “Acquiring linear subspaces for face recognition under variable lighting,” *IEEE Trans. Pattern Anal. Mach. Intell.* **27**, 684–698 (2005). <https://doi.org/10.1109/tpami.2005.92>
 84. Q. Liao and T. A. Poggio, “Bridging the gaps between residual learning, recurrent neural networks and visual cortex,” arXiv:1604.03640
 85. F. Li, H. Bai, and Y. Zhao, “Detail-preserving image super-resolution via recursively dilated residual network,” *Neurocomputing* **358**, 285–293 (2019). <https://doi.org/10.1016/j.neucom.2019.05.042>
 86. H. Li, K. M. Lam, and M. Wang, “Image super-resolution via featureaugmented random forest,” *Signal Process.: Image Commun.* **72**, 25–34 (2019). <https://doi.org/10.1016/j.image.2018.12.001>
 87. J. Li and W. Guan, “Adaptive l_q -norm constrained general nonlocal self-similarity regularizer based sparse representation for single image super resolu-

- tion,” *Inf. Fusion* **53**, 88–102 (2020).
<https://doi.org/10.1016/j.inffus.2019.06.010>
88. T. Li, X. He, Q. Teng, and X. Wu, “Rotation expanded dictionary-based single image super-resolution,” *Neurocomputing* **216**, 1–17 (2016).
<https://doi.org/10.1016/j.neucom.2016.06.066>
89. B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, “Enhanced deep residual networks for single image super-resolution,” arXiv:1707.02921
90. T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: common objects in context,” arXiv:1405.0312
91. X. Li, G. Cao, Y. Zhang, and B. Wang, “Single image super-resolution via adaptive sparse representation and low-rank constraint,” *J. Visual Commun. Image Representation* **55**, 319–330 (2018).
<https://doi.org/10.1016/j.jvcir.2018.06.012>
92. X. Li, G. Cao, Y. Zhang, A. Shafique, and P. Fu, “Combining synthesis sparse with analysis sparse for single image super-resolution,” *Signal Process.: Image Commun.* **83**, 115805 (2020).
<https://doi.org/10.1016/j.image.2020.115805>
93. X. Li and M. Orchard, “New edge-directed interpolation,” *IEEE Trans. Image Process.* **10**, 1521–1527 (2001). <https://doi.org/10.1109/83.951537>
94. Y. Li, W. Dong, X. Xie, G. Shi, J. Wu, and X. Li, “Image super-resolution with parametric sparse model learning,” *IEEE Trans. Image Process.* **27**, 4638–4650 (2018). <https://doi.org/10.1109/tip.2018.2837865>
95. Y. Li, W. Dong, G. Shi, and X. Xie, “Learning parametric distributions for image super-resolution: Where patch matching meets sparse coding,” in *IEEE Int. Conf. on Computer Vision (ICCV), Santiago, 2015* (IEEE, 2015), pp. 450–458.
<https://doi.org/10.1109/iccv.2015.59>
96. Z. Li, Q. Li, W. Wu, J. Yang, Z. Li, and X. Yang, “Deep recursive up-down sampling networks for single image super-resolution,” *Neurocomputing* **398**, 377–388 (2020).
<https://doi.org/10.1016/j.neucom.2019.04.004>
97. D. Lin, G. Xu, W. Xu, Y. Wang, X. Sun, and K. Fu, “SCRSR: An efficient recursive convolutional neural network for fast and accurate image super-resolution,” *Neurocomputing* **398**, 399–407 (2020).
<https://doi.org/10.1016/j.neucom.2019.02.067>
98. C. Liu, X. Sun, C. Chen, P. L. Rosin, Y. Yan, L. Jin, and X. Peng, “Multi-scale residual hierarchical dense networks for single image super-resolution,” *IEEE Access* **7**, 60572–60583 (2019).
<https://doi.org/10.1109/access.2019.2915943>
99. D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang, “Non-local recurrent network for image restoration,” arXiv:1806.02919
100. D. Liu, Z. Wang, B. Wen, J. Yang, W. Han, and T. S. Huang, “Robust single image super-resolution via deep networks with sparse prior,” *IEEE Trans. Image Process.* **25**, 3194–3207 (2016).
<https://doi.org/10.1109/tip.2016.2564643>
101. D. Liu, Z. Wang, N. M. Nasrabadi, and T. S. Huang, “Learning a mixture of deep networks for single image super-resolution,” arXiv:1701.00823
102. G. H. Liu, J. Y. Yang, and Z. Li, “Content-based image retrieval using computational visual attention model,” *Pattern Recognit.* **48**, 2554–2566 (2015).
<https://doi.org/10.1016/j.patcog.2015.02.005>
103. P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, “Multi-level wavelet-CNN for image restoration,” arXiv:1805.07071
104. P. Liu, Y. Hong, and Y. Liu, “Deep differential convolutional network for single image super-resolution,” *IEEE Access* **7**, 37555–37564 (2019).
<https://doi.org/10.1109/access.2019.2903528>
105. S. Liu and W. Deng, “Very deep convolutional neural network based image classification using small training sample size,” in *3rd IAPR Asian Conf. on Pattern Recognition (ACPR), Kuala Lumpur, 2015* (IEEE, 2015), pp. 730–734.
<https://doi.org/10.1109/acpr.2015.7486599>
106. X. Liu, D. Zhao, R. Xiong, S. Ma, W. Gao, and H. Sun, “Image interpolation via regularized local linear regression,” *IEEE Trans. Image Process.* **20**, 3455–3469 (2011). <https://doi.org/10.1109/tip.2011.2150234>
107. Z.-S. Liu, W.-C. Siu, and J.-J. Huang, “Image super-resolution via weighted random forest,” in *IEEE Int. Conf. on Industrial Technology (ICIT), Toronto, 2017* (IEEE, 2017), pp. 1019–1023.
<https://doi.org/10.1109/icit.2017.7915501>
108. X. Liu, D. Zhai, R. Chen, X. Ji, D. Zhao, and W. Gao, “Depth super resolution via joint color-guided internal and external regularizations,” *IEEE Trans. Image Process.* **28**, 1636–1645 (2019).
<https://doi.org/10.1109/tip.2018.2875506>
109. Y. Liu, G. Zhai, K. Gu, X. Liu, D. Zhao, and W. Gao, “Reduced-reference image quality assessment in free-energy principle and sparse representation,” *IEEE Trans. Multimedia* **20**, 379–391 (2018).
<https://doi.org/10.1109/tmm.2017.2729020>
110. Y. Liu, Y. Zhang, Q. Guo, and C. Zhang, “Image interpolation based on weighted and blended rational function,” in *Computer Vision – ACCV 2014 Workshops*, Ed. by C. Jawahar and S. Shan, Lecture Notes in Computer Science, vol. 9009 (Springer, Cham, 2015), pp. 78–88.
<https://doi.org/10.1007/978-3-319-16631-536>
111. A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier nonlinearities improve neural network acoustic models,” in *Proc. of the 30th Int. Conf. on Machine Learning, Atlanta, Ga., 2013*.
112. C. Ma, C. Yang, X. Yang, and M. Yang, “Learning a no-reference quality metric for single-image super-resolution,” arXiv:1612.05890
113. 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 *Proc. Eighth IEEE Int. Conf. on Computer Vision. ICCV 2001, Vancouver, 2001* (IEEE, 2001), vol. 2, pp. 416–423.
<https://doi.org/10.1109/iccv.2001.937655>
114. S. Matsumoto, M. Kamada, and R.-O. Mijiddorj, “Adaptive image interpolation by cardinal splines in piecewise constant tension,” *Optim. Lett.* **6**, 1265–1280 (2011). <https://doi.org/10.1007/s11590-011-0371-6>

115. T. Michaeli and M. Irani, "Nonparametric blind super-resolution," in *IEEE Int. Conf. on Computer Vision, Sydney, 2013*, (IEEE, 2013), pp. 945–952. <https://doi.org/10.1109/iccv.2013.121>
116. G. F. Montufar, R. Pascanu, K. Cho, and Y. Bengio, "On the number of linear regions of deep neural networks," in *Proc. of the 27th Int. Conf. on Neural Information Processing Systems, 2014*, Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. O. Weinberger (MIT Press, Cambridge, Mass., 2014), vol. 2, pp. 2924–2932. <https://papers.nips.cc/paper/5422-on-the-number-of-linear-regions-of-deep-neural-networks.pdf>
117. H. Nasrollahi, K. Farajzadeh, V. Hosseini, E. Zareza-deh, and M. Abdollahzadeh, "Deep artifact-free residual network for single-image super-resolution," *Signal, Image Video Process.* **14**, 407–415 (2020). <https://doi.org/10.1007/s11760-019-01569-3>
118. S. A. Nene, S. K. Nayar, and H. Murase, Columbia object image library (COIL-20), Tech. Rep. No. CUCS-006-96 (Columbia Univ. New York, 1996).
119. N. Nikolaev and H. Iba, "Learning polynomial feed-forward neural networks by genetic programming and backpropagation," *IEEE Trans. Neural Networks* **14**, 337–350 (2003). <https://doi.org/10.1109/tnn.2003.809405>
120. K. S. Ni and T. Q. Nguyen, "Image superresolution using support vector regression, *IEEE Trans. Image Process.* **16**, 1596–1610 (2007). <https://doi.org/10.1109/tip.2007.896644>
121. A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," arXiv:1601.06759
122. A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu, "Conditional image generation with PixelCNN decoders," arXiv:1606.05328
123. G. Pandey and U. Ghanekar, "A compendious study of super-resolution techniques by single image," *Optik* **166**, 147–160 (2018). <https://doi.org/10.1016/j.ijleo.2018.03.103>
124. G. Pandey and U. Ghanekar, "Classification of priors and regularization techniques appurtenant to single image super-resolution," *Visual Comput.* **36**, 1291–1304 (2020). doi <https://doi.org/10.1007/s00371-019-01729-z>
125. D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, and V. Ferrari, "We don't need no bounding-boxes: Training object class detectors using only human verification," arXiv:1602.08405
126. S. J. Park, H. Son, S. Cho, K.-S. Hong, and S. Lee, "SRFeat: Single image super-resolution with feature discrimination," in *Computer Vision – ECCV 2018*, Ed. by V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Lecture Notes in Computer Science, vol. 11220 (Springer, Cham, 2018), pp. 455–471. https://doi.org/10.1007/978-3-030-01270-0_27
127. A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *31st Conf. on Neural Information Processing Systems (NIPS 2017), Long Beach, Calif., 2017*. <https://openreview.net/forum?id=BJJsrmlfCZ>
128. J. Patterson and A. Gibson, *Deep Learning: A Practitioner's Approach*, (O'Reilly, Beijing, 2017). <https://www.safaribooksonline.com/library/view/deep-learning/9781491924570/>
129. A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *IEEE Conf. on Computer Vision and Pattern Recognition, Miami, Fla., 2009* (IEEE, 2009), pp. 413–420. <https://doi.org/10.1109/cvpr.2009.5206537>
130. S. Ren, D. K. Jain, K. Guo, T. Xu, and T. Chi, "Towards efficient medical lesion image super-resolution based on deep residual networks," *Signal Process.: Image Commun.* **75**, 1–10 (2019). <https://doi.org/10.1016/j.image.2019.03.008>
131. B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image annotation," *Int. J. Computer Vision* **77**, 157–173 (2008). <https://doi.org/10.1007/s11263-007-0090-8>
132. J. Salvador and E. Pérez-Pellitero, "Naive bayes super-resolution forest," in *IEEE Int. Conf. on Computer Vision (ICCV), Santiago, 2015* (IEEE, 2015). <https://doi.org/10.1109/iccv.2015.45>
133. M. S. M. Sajjadi, B. Schölkopf, and M. Hirsch, "EnhanceNet: Single image super-resolution through automated texture synthesis," arXiv:1612.07919
134. F. Sha, S. M. Zandavi, and Y.-Y. Chung, "Fast deep parallel residual network for accurate super resolution image processing," *Expert Syst. Appl.* **128**, 157–168 (2019). <https://doi.org/10.1016/j.eswa.2019.03.032>
135. J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks* **61**, 85–117 (2015). <https://doi.org/10.1016/j.neunet.2014.09.003>
136. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large scale image recognition," arXiv:1409.1556 [cs.CV]
137. A. Singh and N. Ahuja, "Sub-band energy constraints for self-similarity based super-resolution," in *22nd Int. Conf. on Pattern Recognition, Stockholm, 2014* (IEEE, 2014). <https://doi.org/10.1109/icpr.2014.761>
138. P. Shamsolmoali, M. Zareapoor, R. Wang, D. K. Jain, and J. Yang, "GGANISR: Gradual generative adversarial network for image super resolution," *Neurocomputing* **366**, 140–153 (2019). <https://doi.org/10.1016/j.neucom.2019.07.094>
139. L. C. H.R. Sheikh, Z. Wang, and A. Bovik, "Live image quality assessment database release 2," <http://live.ece.utexas.edu/research/quality>
140. H. Sheikh, A. Bovik, and G. de Veciana, "An information fidelity criterion for image quality assessment using natural scene statistics," *IEEE Trans. Image Process.* **14**, 2117–2128 (2005). <https://doi.org/10.1109/tip.2005.859389>
141. W. Shi, J. Caballero, F. Huszar, 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," arXiv:1609.05158
142. W. Shi, J. Caballero, L. Theis, F. Huszar, A. P. Aitken, C. Ledig, and Z. Wang, "Is the deconvolution layer the same as a convolutional layer?," arXiv:1609.07009

143. A. Shocher, N. Cohen, and M. Irani, "Zero-shot" super-resolution using deep internal learning," arXiv:1712.06087
144. Y.-Z. Song, W.-Y. Liu, J.-C. Chen, and K. W. Lin, "Single image super-resolution with vision loss function," in *Intelligent Information and Database Systems. ACIIDS 2019*, Ed. by N. Nguyen, F. Gaol, T. P. Hong, and B. Trawiński, Lecture Notes in Computer Science, vol. 11432 (Springer, Cham, 2019), pp. 173–179. https://doi.org/10.1007/978-3-030-14802-7_15
145. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.* **15**, 1929–1958 (2014). <http://jmlr.org/papers/v15/srivastava14a.html>
146. R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," arXiv:1507.06228
147. L. Sun and J. Hays, "Super-resolution from internet-scale scene matching," in *IEEE Int. Conf. on Computational Photography (ICCP), Seattle, Wash., 2012* (IEEE, 2012), pp. 1–12. <https://doi.org/10.1109/iccp.2012.6215221>
148. J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *IEEE Conf. on Computer Vision and Pattern Recognition, Anchorage, Alaska, 2008*, (IEEE, 2008), pp. 1–8. <https://doi.org/10.1109/cvpr.2008.4587659>
149. L. Sun, T. Zhan, Z. Wu, and B. Jeon, "A novel 3D anisotropic total variation regularized low rank method for hyperspectral image mixed denoising," *ISPRS Int. J. Geo-Inf.* **7**, 412 (2018). <https://doi.org/10.3390/ijgi7100412>
150. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," arXiv:1409.4842
151. V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE* **105**, 2295–2329 (2017). <https://doi.org/10.1109/jproc.2017.2761740>
152. Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Honolulu, 2017* (IEEE, 2017). <https://doi.org/10.1109/cvpr.2017.298>
153. Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: A persistent memory network for image restoration," arXiv:1708.02209
154. H. Takeda, S. Farsiu, and P. Milanfar, "Robust kernel regression for restoration and reconstruction of images from sparse noisy data," in *Int. Conf. on Image Processing, Atlanta, 2006* (IEEE, 2006). <https://doi.org/10.1109/icip.2006.312573>
155. Y. Tang and L. Shao, "Pairwise operator learning for patch-based single-image super-resolution," *IEEE Trans. Image Process.* **26**, 994–1003 (2017). <https://doi.org/10.1109/tip.2016.2639440>
156. Y. Tang, Y. Yuan, P. Yan, and X. Li, "Greedy regression in sparse coding space for single-image super-resolution," *J. Visual Commun. Image Representation* **24**, 148–159 (2013). <https://doi.org/10.1016/j.jvcir.2012.02.003>
157. B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L. Li, "The new data and new challenges in multimedia research," arXiv:1503.01817
158. R. Timofte, E. Agustsson, L. V. Gool, M.-H. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee, X. Wang, Y. Tian, K. Yu, Y. Zhang, S. Wu, C. Dong, L. Lin, Y. Qiao, C. C. Loy, W. Bae, J. Yoo, Y. Han, J. C. Ye, J.-S. Choi, M. Kim, Y. Fan, J. Yu, W. Han, D. Liu, H. Yu, Z. Wang, H. Shi, X. Wang, T. S. Huang, Y. Chen, K. Zhang, W. Zuo, Z. Tang, L. Luo, S. Li, M. Fu, L. Cao, W. Heng, G. Bui, T. Le, Y. Duan, D. Tao, R. Wang, X. Lin, J. Pang, J. Xu, Y. Zhao, X. Xu, J. Pan, D. Sun, Y. Zhang, X. Song, Y. Dai, X. Qin, X.-P. Huynh, T. Guo, H. S. Mousavi, T. H. Vu, V. Monga, C. Cruz, K. Egiazarian, V. Katkovnik, R. Mehta, A. K. Jain, A. Agarwalla, C. V. S. Praveen, R. Zhou, H. Wen, C. Zhu, Z. Xia, Z. Wang, and Q. Guo, "NTIRE 2017 challenge on single image super-resolution: Methods and results," in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, 2017*, (IEEE, 2017), pp. 1110–1121. <https://doi.org/10.1109/cvprw.2017.149>
159. R. Timofte, V. De, and L. Van Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *IEEE Int. Conf. on Computer Vision, Sydney, 2013* (IEEE, 2013), pp. 1920–1927. <https://doi.org/10.1109/iccv.2013.241>
160. R. Timofte, V. D. Smet, and L. Van Gool, "Semantic super-resolution: When and where is it useful?," *Comput. Vision Image Understanding* **142**, 1–12 (2016). <https://doi.org/10.1016/j.cviu.2015.09.008>
161. T. Tirer and R. Giryes, "Super-resolution via image-adapted denoising CNNs: Incorporating external and internal learning," *IEEE Signal Process. Lett.* **26**, 1080–1084 (2019). <https://doi.org/10.1109/lsp.2019.2920250>
162. T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *IEEE Int. Conf. on Computer Vision (ICCV), Venice, 2017* (IEEE, 2017), pp. 4809–4817. <https://doi.org/10.1109/iccv.2017.514>
163. V. D. S. R. Timofte, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Computer Vision – ACCV 2014*, Ed. by D. Cremers, I. Reid, H. Saito, and M. H. Yang, Lecture Notes in Computer Science, vol. 9006 (Springer, Cham, 2015), pp. 111–126. https://doi.org/10.1007/978-3-319-16817-3_8
164. D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, "Deep image prior," arXiv:1711.10925
165. A. Vedaldi and K. Lenc, "Matconvnet - convolutional neural networks for MATLAB," arXiv:1412.4564 [cs.CV]
166. A. Veit, M. J. Wilber, and S. J. Belongie, "Residual networks are exponential ensembles of relatively shallow networks," arXiv:1605.06431
167. S. Villena, M. Vega, S. Babacan, R. Molina, and A. Katsaggelos, "Bayesian combination of sparse and non-sparse priors in image super resolution," *Digital Signal Process.* **23**, 530–541 (2013). <https://doi.org/10.1016/j.dsp.2012.10.002>

168. S. Villena, M. Vega, R. Molina, and A. Katsaggelos, "A non-stationary image prior combination in super-resolution," *Digital Signal Process.* **32**, 1–10 (2014). <https://doi.org/10.1016/j.dsp.2014.05.017>
169. F.-A. Vasluianu, A. Romero, L. Van Gool, and R. Timofte, "Shadow removal with paired and unpaired learning," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Nashville, Tenn., 2021 (IEEE, 2021), pp. 826–835. <https://doi.org/10.1109/CVPRW53098.2021.00092>
170. X. Wang, K. Yu, C. Dong, and C. C. Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," arXiv:1804.02815
171. 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," arXiv:1809.00219
172. Z. Wang and A. Bovik, "Mean squared error: Love it or leave it? A new look at signal fidelity measures," *IEEE Signal Process. Mag.* **26**, 98–117 (2009). <https://doi.org/10.1109/msp.2008.930649>
173. Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *IEEE Int. Conf. on Computer Vision (ICCV)*, Santiago, 2015, (IEEE, 2015). <https://doi.org/10.1109/iccv.2015.50>
174. J. Wu, W. Lin, G. Shi, and A. Liu, "Reduced-reference image quality assessment with visual information fidelity," *IEEE Trans. Multimedia* **15**, 1700–1705 (2013). <https://doi.org/10.1109/tmm.2013.2266093>
175. S. Xue, W. Qiu, F. Liu, and X. Jin, "Wavelet-based residual attention network for image super-resolution," *Neurocomputing* **382**, 116–126 (2020). <https://doi.org/10.1016/j.neucom.2019.11.044>
176. J. Yamanaka, S. Kuwashima, and T. Kurita, "Fast and accurate image super resolution by deep CNN with skip connection and network in network," arXiv:1707.05425
177. J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.* **19**, 2861–2873 (2010). <https://doi.org/10.1109/tip.2010.2050625>
178. X. Yao, Q. Wu, P. Zhang, and F. Bao, "Adaptive rational fractal interpolation function for image super-resolution via local fractal analysis," *Image Vision Comput.* **82**, 39–49 (2019). <https://doi.org/10.1016/j.imavis.2019.02.002>
179. B. Yan, B. Bare, C. Ma, K. Li, and W. Tan, "Deep objective quality assessment driven single image super-resolution," *IEEE Trans. Multimedia* **21**, 2957–2971 (2019). <https://doi.org/10.1109/tmm.2019.2914883>
180. J. Yang, Z. Lin, and S. Cohen, "Fast image super-resolution based on in-place example regression," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Portland, 2013, (IEEE, 2013), pp. 1059–1066. <https://doi.org/10.1109/cvpr.2013.141>
181. X. Yang, H. Mei, J. Zhang, K. Xu, B. Yin, Q. Zhang, and X. Wei, DRFN: Deep recurrent fusion network for single-image super-resolution with large factors, *IEEE Trans. Multimedia* **21**, 328–337 (2019). <https://doi.org/10.1109/tmm.2018.2863602>
182. J. Yang, W. Li, R. Wang, L. Xue, and M. Hu, "Enhanced two-phase residual network for single image super-resolution," *J. Visual Commun. Image Representation* **61**, 188–197 (2019). <https://doi.org/10.1016/j.jvcir.2019.04.002>
183. C.-Y. Yang, C. Ma, and M. H. Yang, "Single-image super-resolution: A benchmark," in *Computer Vision – ECCV 2014*, Ed. by D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Lecture Notes in Computer Science, vol. 8692 (Springer, Cham, 2014), pp. 372–386. doi https://doi.org/10.1007/978-3-319-10593-2_25
184. W. Yang, J. Feng, J. Yang, F. Zhao, J. Liu, Z. Guo, and S. Yan, "Deep edge guided recurrent residual learning for image super-resolution," *IEEE Trans. Image Process.* **26**, 5895–5907 (2017). <https://doi.org/10.1109/tip.2017.2750403>
185. F. Yu, Y. Zhang, S. Song, A. Seff, and J. Xiao, "LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop," arXiv:1506.03365
186. J. Yu, Y. Fan, J. Yang, N. Xu, Z. Wang, X. Wang, and T. S. Huang, "Wide activation for efficient and accurate image super-resolution," arXiv:1808.08718
187. S. Yu, W. Kang, S. Ko, and J. Paik, "Single image super-resolution using locally adaptive multiple linear regression," *J. Opt. Soc. Am. A* **32**, 2264 (2015). <https://doi.org/10.1364/josaa.32.002264>
188. M. Zareapoor, M. E. Celebi, and J. Yang, "Diverse adversarial network for image super-resolution," *Signal Process.: Image Commun.* **74**, 191–200 (2019). <https://doi.org/10.1016/j.image.2019.02.008>
189. M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Int. Conf. on Computer Vision, Barcelona, 2011* (IEEE, 2011), pp. 2018–2025. doi <https://doi.org/10.1109/iccv.2011.6126474>
190. R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse representations," in *Curves and Surfaces*, Ed. by J. D. Boissonnat, P. Chenin, A. Cohen, C. Gout, and T. Lyche, Lecture Notes in Computer Science, vol. 6920 (Springer, Berlin, 2012), pp. 711–730. https://doi.org/10.1007/978-3-642-27413-8_47
191. C. Zhang, W. Liu, J. Liu, C. Liu, and C. Shi, "Sparse representation and adaptive mixed samples regression for single image super-resolution," *Signal Process.: Image Commun.* **67**, 79–89 (2018). <https://doi.org/10.1016/j.image.2018.06.001>
192. K. Zhang, X. Gao, D. Tao, and X. Li, "Single image super-resolution with multiscale similarity learning," *IEEE Trans. Neural Networks Learn. Syst.* **24**, 1648–1659 (2013). <https://doi.org/10.1109/tnnls.2013.2262001>
193. K. Zhang, X. Gao, X. Li, and D. Tao, "Partially supervised neighbor embedding for example-based image super-resolution," *IEEE J. Sel. Top. Signal Process.* **5**, 230–239 (2011). <https://doi.org/10.1109/jstsp.2010.2048606>
194. K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," arXiv:1704.03264

195. K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," arXiv:1712.06116
196. W. Zhang, K. Itoh, J. Tanida, and Y. Ichioka, "Parallel distributed processing model with local space-invariant interconnections and its optical architecture," *Appl. Opt.* **29**, 4790 (1990).
<https://doi.org/10.1364/ao.29.004790>
197. W. Zhang, Y. Liu, C. Dong, and Y. Qiao, "Ranksrgan: Generative adversarial networks with ranker for image super-resolution," arXiv:1908.06382
198. Y. Zhang, Q. Fan, F. Bao, Y. Liu, and C. Zhang, "Single-image super-resolution based on rational fractal interpolation," *IEEE Trans. Image Process.* **27**, 3782–3797 (2018).
<https://doi.org/10.1109/tip.2018.2826139>
199. Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Computer Vision – ECCV 2018*, Ed. by V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Lecture Notes in Computer Science, vol. 11211 (Springer, Cham, 2018), pp. 294–310.
https://doi.org/10.1007/978-3-030-01234-2_18
200. Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," arXiv:1802.08797
201. H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imaging* **3**, 47–57 (2017).
<https://doi.org/10.1109/tci.2016.2644865>
202. J. J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," arXiv:1609.03126
203. Z. Zhong, T. Shen, Y. Yang, Z. Lin, and C. Zhang, "Joint sub-bands learning with clique structures for wavelet domain super-resolution," arXiv:1809.04508
204. Y. Zhu, Y. Zhang, and A. L. Yuille, "Single image super-resolution using deformable patches," in *IEEE Conf. on Computer Vision and Pattern Recognition, Columbus, Ohio, 2014* (IEEE, 2014), pp. 2917–2924.
<https://doi.org/10.1109/cvpr.2014.373>



Garima Pandey completed her BTech in electronics and communication engineering and MTech in communication engineering. Presently doing PhD from National Institute of Technology Kurukshetra, Kurukshetra, Haryana, India. Active in field of image processing and super-resolution.



Umesh Ghanekar completed his MTech degree in Electronics and Communication Engineering in 1988 from Indian Institute of Technology, Roorkee, India and PhD in computer engineering in 2013 from National Institute of Technology Kurukshetra, Kurukshetra, Haryana, India. Presently, he is a Professor in the Department of Electronics and Communication Engineering at N.I.T Kurukshetra. His research interests include signal and image processing.