Deep Learning for Image/Video Restoration and Super-resolution

A. Murat Tekalp

Koç University, Istanbul, Turkey; mtekalp@ku.edu.tr

ABSTRACT

Recent advances in neural signal processing led to significant improvements in the performance of learned image/video restoration and super-resolution (SR). An important benefit of data-driven deep learning approach to image processing is that neural models can be optimized for any differentiable loss function, including perceptual loss functions, leading to perceptual image/video restoration and SR, which cannot be easily handled by traditional model-based methods.

We start with a brief problem statement and a short discussion on traditional vs. data-driven solutions. We next review recent advances in neural architectures, such as residual blocks, dense connections, residual-in-residual dense blocks, residual blocks with generative neurons, self-attention and visual transformers. We then discuss loss functions and evaluation (assessment) criteria for image/video restoration and SR, including fidelity (distortion) and perceptual criteria, and the relation between them, where we briefly review the perception vs. distortion trade-off.

We can consider learned image/video restoration and SR as learning either a nonlinear regressive mapping from degraded to ideal images based on the universal approximation theorem, or a generative model that captures the probability distribution of ideal images. We first review regressive

inference via residual and/or dense convolutional networks (ConvNet). We also show that using a new architecture with residual blocks based on a generative neuron model can outperform classical residual ConvNets in peak-signal-to-noise ratio (PSNR). We next discuss generative inference based on adversarial training, such as SRGAN and ESRGAN, which can reproduce realistic textures, or based on normalizing flow such as SRFlow by optimizing log-likelihood. We then discuss problems in applying supervised training to real-life restoration and SR, including overfitting image priors and overfitting the degradation model seen in the training set. We introduce multiple-model SR and real-world SR (from unpaired training data) formulations to overcome these problems. Integration of traditional model-based methods and deep learning for non-blind restoration/SR is introduced as another solution to model overfitting in supervised learning.

In learned video restoration and SR (VSR), we first discuss how to best exploit temporal correlations in video, including sliding temporal window vs. recurrent architectures for propagation, and aligning frames in the pixel domain using optical flow vs. in the feature space using deformable convolutions. We next introduce early fusion with feature-space alignment, employed by the EDVR network, which obtains excellent PSNR performance. However, it is well-known that videos with the highest PSNR may not be the most appealing to humans, since minimizing the mean-square error may result in blurring of details. We then address perceptual optimization of VSR models to obtain natural texture and motion. Although perception-distortion tradeoff has been well studied for images, few works address perceptual VSR. In addition to using perceptual losses, such as MS-SSIM, LPIPS, and/or adversarial training, we also discuss explicit loss functions/criteria to enforce and evaluate temporal consistency. We conclude with a discussion of open problems.

1

Introduction

Deep learning has made significant impact not only on computer vision and natural language processing but also on classical signal processing problems such as image/video restoration/super-resolution (SR) and compression. This paper reviews recent advances and the state of the art in image/video restoration and SR using deep learning. It is worth noting that the nonlinear neural signal processing techniques discussed in this paper also apply to other inverse problems in imaging.

This chapter provides an introduction to image restoration and SR problems, including a general overview of classical model-based vs. modern data-driven solutions. We start with the problem statement in Section 1.1, where we pose image restoration/SR as an ill-posed inverse problem. Linear model-based regularization of ill-posed inverse problems is reviewed in Section 1.2. Limitations of linear, shift-invariant (LSI) regularization are discussed in Section 1.3. Next, Section 1.4 provides an overview of classical nonlinear model-based regularized inversion vs. modern data-driven learned approaches. We introduce the three pillars of learned image/video restoration and SR solutions: the architecture, the optimization and evaluation criteria, and training in Section 1.5. Finally, we briefly discuss other related survey articles in Section 1.6.

1.1 Problem Statement

Inverse problems in imaging are those problems, where we want to solve for the ideal image vector $\mathbf{x}$ given a nonlinear observation model

$$\mathbf{y} = \mathcal{D}(\mathbf{x}) + \mathbf{v} \tag{1.1}$$

where $\mathbf{y}$ denotes the observation vector, $\mathcal{D}$ is a nonlinear degradation operator, and $\mathbf{v}$ is the observation noise vector. In the traditional formulation of inverse problems, the degradation (forward) model is assumed to be linear, which can be expressed as

$$\mathbf{y} = \mathbf{DHx} + \mathbf{v} \tag{1.2}$$

where $\mathbf{H}$ denotes a linear degradation operator, and $\mathbf{D}$ is an observation matrix. This linear observation model includes the following image restoration problems as special cases:

- The denoising problem, where $\mathbf{D}=\mathbf{H}=\mathbf{I}$ (identity matrix).
- The deblurring problem, where $\mathbf{D}=\mathbf{I}$ and the matrix $\mathbf{H}$ is determined by the blur point spread function (PSF).
- The super-resolution (SR) problem, where $\mathbf{D}$ and $\mathbf{H}$ represent the sub-sampling operation and the anti-alias filter, respectively.
- The image inpainting problem, where the elements of matrix $\mathbf{D}$ that correspond to missing pixels are set to zero.

1.1.1 Ill-Posed Problems

According to Hadamard, a problem is well-posed if it satisfies the following conditions (Tikhonov and Arsenin, 1977): i) a solution exists, ii) the solution is unique, and iii) small perturbations (noise) in the observations (input) results in small changes in the solution. Problems that are not well-posed in the sense of Hadamard are called ill-posed.

Inverse problems in imaging are often ill-posed because the matrices $\mathbf{D}$ and/or $\mathbf{H}$ may be non-square with more unknowns than the number of equations; hence, the solution either does not exist and/or is not unique, and/or the condition number of matrix $\mathbf{H}$ is large so that the solution is highly sensitive to observation noise.

1.1.2 Non-blind vs. Blind Image Restoration and SR

We can classify inverse problems as non-blind or blind depending on whether the degradation operator and observation noise level in Eqn. 1.1 and Eqn. 1.2 are known or not.

A low resolution (LR) image is modeled as down-sampled version of an ideal high resolution (HR) image. We typically model the anti-alias filtering in the down-sampling operation by a bicubic filter; hence, this process is often referred as bicubic downsampling. In real-world applications, there are additional sources of blur in LR image formation, such as motion blur or camera shake blur, which is represented by a convolution kernel $\mathbf{k}$, given by

$$\mathbf{y} = (\mathbf{k} * \mathbf{x}) \downarrow + \mathbf{v} \tag{1.3}$$

where $\downarrow$ denotes bicubic downsampling. While the blur due to $\downarrow$ is a bicubic filter, the additional source of blur, denoted by $\mathbf{k}$ is usually unknown and image specific.

Non-blind image restoration and SR refers to the case where the blur kernel $\mathbf{k}$ and noise level in Eqn. 1.3 are known or estimated prior to the image restoration process. Most non-blind methods assume that there is no additional source of blur in LR image formation, and only model bicubic anti-alias filtering. Hence, Eqn. 1.3 simplifies as

$$\mathbf{y} = (\mathbf{x}) \downarrow + \mathbf{v} \tag{1.4}$$

Blind image restoration and SR refers to the case where the blur kernel $\mathbf{k}$ and noise level in Eqn. 1.3 are unknown and must be estimated simultaneously with the image restoration and SR process.

1.2 Model-based Regularization of Ill-Posed Inverse Problems

Since the forward model (1.1) or (1.2) is in general not invertible, one can possibly define the ordinary least squares estimate of $\mathbf{x}$ or the pseudo-inverse solution given by

$$\hat{\mathbf{x}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y} \tag{1.5}$$

However, this solution is not regularized in the sense that it is highly sensitive to small perturbations (noise) in the observation vector $\mathbf{y}$.

Finding a solution that is well-behaved in the presence of observation noise is impossible without utilizing some prior information about the ideal signal/image $\mathbf{x}$. This is called regularization of the inverse solution. Traditional model-based regularized inversion methods minimize a cost function subject to some constraints (prior) on the solution. Assuming the observation noise is additive, white Gaussian, and is independent of the signal/image $\mathbf{x}$, the regularized inverse solution can be found as

$$\hat{\mathbf{x}}(\lambda) = arg_{\mathbf{x}}\ min\ \frac{1}{2}||\mathbf{y} - \mathbf{DHx}||^2 + \lambda \mathbf{R}(\mathbf{x}) \tag{1.6}$$

where $\mathbf{R}(\mathbf{x})$ is a regularization operator that imposes some prior on $\mathbf{x}$. Hence, the solution is the minimizer of a data-consistency cost term, which measures how well the restored image matches the observations given the degradation model, and a regularizer term, which imposes some prior knowledge or promotes images with some desirable property.

One of the first regularization methods is Tikhonov regularization, which, in the case $\mathbf{D}=\mathbf{I}$, is given by (Tikhonov and Arsenin, 1977)

$$\hat{\mathbf{x}}(\lambda) = (\mathbf{H}^T\mathbf{H} + \lambda \mathbf{L}^T\mathbf{L})^{-1}\mathbf{H}^T\mathbf{y} \tag{1.7}$$

where $\mathbf{L}$ is a linear regularization operator expressed in matrix form and λ is a parameter that controls the tradeoff between data consistency and regularization, i.e., noise sensitivity. For example, $\mathbf{L}$ can be the Laplacian operator that estimates high frequency image components. In this case, minimizing the energy of high frequency image components can be viewed as imposing a smoothness constraint as an image prior.

Direct computation of (1.7) requires inversion of the large matrix $(\mathbf{H}^T\mathbf{H}+\lambda\mathbf{L}^T\mathbf{L})$. There are two common approaches to avoid inversion of this large matrix: i) employing an iterative solution, ii) diagonalization using the discrete Fourier transform assuming the matrix is circulant. Under certain assumptions, this regularized inverse solution can be obtained by a linear, shift-invariant regularized inverse filter.

1.3 Limitations of Linear Shift-invariant Regularized Inverse Filters

Let's express the observation model (1.2), in the case **D=I**, in scalar form as a convolution

$$y(n_1, n_2) = h(n_1, n_2) * *x(n_1, n_2) + v(n_1, n_2) \tag{1.8}$$

Taking the 2-D discrete Fourier transform of both sides, we obtain

$$Y(e^{j\omega_1}, e^{j\omega_2}) = H(e^{j\omega_1}, e^{j\omega_2})X(e^{j\omega_1}, e^{j\omega_2}) + V(e^{j\omega_1}, e^{j\omega_2}) \tag{1.9}$$

If we process the observed image by a linear, shift-invariant restoration filter $\Phi(e^{j\omega_1}, e^{j\omega_2})$, the estimated image can be expressed as

$$\hat{X}(e^{j\omega_1}, e^{j\omega_2}) = \Phi(e^{j\omega_1}, e^{j\omega_2})Y(e^{j\omega_1}, e^{j\omega_2}) \tag{1.10}$$

If we now substitute Eqn. 1.9 for $Y(e^{j\omega_1}, e^{j\omega_2})$, we get

$$\hat{X}(e^{j\omega_1}, e^{j\omega_2}) = \Phi(e^{j\omega_1}, e^{j\omega_2})[H(e^{j\omega_1}, e^{j\omega_2})X(e^{j\omega_1}, e^{j\omega_2}) + V(e^{j\omega_1}, e^{j\omega_2})] \tag{1.11}$$

In order to analyze the artifacts due to processing with a linear, shift-invariant filter $\Phi(e^{j\omega_1}, e^{j\omega_2})$, we add and subtract $X(e^{j\omega_1}, e^{j\omega_2})$ to the right hand side to obtain (Tekalp and Sezan, 1990):

$$\begin{aligned} \hat{X}(e^{j\omega_1}, e^{j\omega_2}) \quad = \quad & X(e^{j\omega_1}, e^{j\omega_2}) \\ & + [\Phi(e^{j\omega_1}, e^{j\omega_2})H(e^{j\omega_1}, e^{j\omega_2}) - 1]X(e^{j\omega_1}, e^{j\omega_2}) \\ & + \Phi(e^{j\omega_1}, e^{j\omega_2})V(e^{j\omega_1}, e^{j\omega_2}) \end{aligned} \tag{1.12}$$

The second term at the right-hand side is signal-dependent regularization error (ringing artifacts). The third term is filtered noise artifacts. If we let $\Phi(e^{j\omega_1}, e^{j\omega_2}) = H^{-1}(e^{j\omega_1}, e^{j\omega_2})$ (inverse filter) then the second term disappears, but the third term dominates and masks the signal $x(n_1, n_2)$. Hence, the trade-off between the last two terms is a theoretical limitation of LSI regularized solutions (Tekalp and Sezan, 1990).

In order to overcome this theoretical limitation of LSI inverse filters, many adaptive or nonlinear methods have been proposed within the past 30 years. They are briefly discussed in the next section.

1.4 Nonlinear Model-based vs. Data-driven Approaches

Traditional nonlinear model-based regularized inversion methods have been applied to solve image/video restoration and SR problems for over 50 years. We can broadly classify available solutions as: i) iterative methods that impose deterministic constraints or priors about the ideal image, ii) methods based on statistical estimation theory, and iii) example-based methods based on machine learning (but not end-to-end deep learning). Examples of such methods include maximum a posteriori probability (MAP) estimation, sparse modeling (Papyan *et al.*, 2018), adaptive filters (Erdogmus and Principe, 2006), and example-based machine learning (Freeman *et al.*, 2002; Liu *et al.*, 2007).

Iterative methods can be used to impose constraints on the solution. Early iterative regularization methods include nonlinear Landweber iterations, iterative back-projection, or projection onto convex sets (POCS) methods. Iterative solutions to variational optimization formulations, such as the total variation (TV) regularization, have also been proposed. TV regularization suppresses oscillations (noise) in the solution while allowing for discontinuities (edges). Later, iterative solutions based on sparse and redundant image representation have become popular. Sparse redundant representations constrain the signal to the form

$$\mathbf{x} = \mathbf{A}\gamma \tag{1.13}$$

where $\mathbf{x} \in R^n$, $\gamma \in R^m$ such that $m > n$, and the $n \times m$ matrix $\mathbf{A}$ is a dictionary of atoms. The vector γ is sparse with only few (say k) nonzero elements; thus, $\mathbf{x}$ is constrained to be a linear combination of k atoms from a learned dictionary $\mathbf{A}$.

Statistical estimation methods pose image/video restoration and SR as finding the minimum mean square error (MMSE) estimate, given by

$$\hat{\mathbf{x}}_{MMSE} = arg_{\hat{\mathbf{x}}}\ min\ E\{(\mathbf{x} - \hat{\mathbf{x}})^2\} \tag{1.14}$$

or the maximum a posteriori probability (MAP) estimate, given by

$$\hat{\mathbf{x}}_{MAP} = arg_{\mathbf{x}}\ min\ \ln p(\mathbf{x}|\mathbf{y}) = arg_{\mathbf{x}}\ min\ (\ln p(\mathbf{y}|\mathbf{x}) + \ln p(\mathbf{x})) \tag{1.15}$$

Note that when the distributions are Gaussian, the first and second terms in Eqn. (1.15) correspond to those in Eqn. (1.6).

Example-based learning have also been shown to yield good results. Nevertheless, classical model-based solutions require iterations (more computation) during inference and their performance is limited since single-image SR is a severely ill-posed inverse problem.

The latest advance in the state-of-the-art in nonlinear image/video restoration and SR is based on deep learning driven by big data. It only became possible to obtain deep learned SR results that are superior to those of traditional model-based approaches within the last 5-6 years leveraging the recent advances in deep neural network architectures and training methods including optimizers, wide availability of large datasets, and powerful GPU computing.

Learned image restoration and SR tasks can be posed as a nonlinear regression problem or a generative modeling problem. We can gain insight on how deep learning helps to achieve state of the art image restoration and SR results leveraging data-driven regression paradigm by means of the following example. Suppose we want to predict the weight of a person given his/her height and age. Given a dataset with weight, height and ages of people, we can fit a surface in 3-D to given data. If we fit a linear model, this would be a plane in 3-D as depicted in Figure 1.1. A nonlinear regression framework would allow fitting an arbitrary 3-D surface to the given data. Given the height and age of a new person not in the training dataset, we can project the height and age to the 3-D prediction surface to get a reading of the predicted weight. The shape of

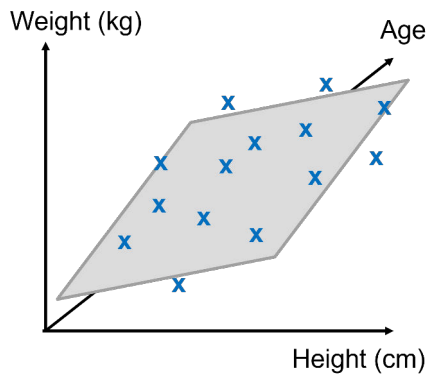


Figure 1.1: Illustration of linear regression in a 3-dimensional space.

the 3-D surface, which determines the accuracy of the predicted weight, depends on the form of the nonlinear predictor, the loss function used in fitting, and of course the goodness of the available training data.

Regressive inference for learned image restoration and SR works similarly, where we have input (LR) and output (HR) image pairs. Each corresponding LR-HR image pair is represented by a point in a very high dimensional space (each pixel is a dimension). For example, if we have 100×100 patches, that would constitute a 10,000 dimensional space. A deep learning model defines a prediction manifold that is fitted to these sample points in the very high-dimensional space. In analogy with the above example, the accuracy of the predicted HR images depends on the architecture of the neural network (the form of the predictor), the optimization criterion, and the available datasets.

Alternatively, generative inference works by first learning a model to represent the distribution of the ideal image conditioned on a given degraded image, and then sampling one or more plausible solutions from this distribution during inference.

The inference process in model-based methods and learned methods are in stark contrast. In traditional model-based methods, there is no training process, but we need to solve a different optimization problem for each test image. While this requires significantly more computation during inference, it provides flexibility to use a different degradation model for each test image. In learned methods, we typically assume all training and test images are subject to the same degradation process, and the training step requires significant computation, but the inference process is very fast. Hence, classical model-based and deep learning approaches have different strengths and weaknesses.

1.5 Three Pillars of Learned Image Restoration and SR

The three pillars of learned image restoration and SR are the network architecture, the optimization criteria, and training methodology and data. We provide a brief introduction to each of these pillars, depicted in Figure 1.2, in the following subsections.

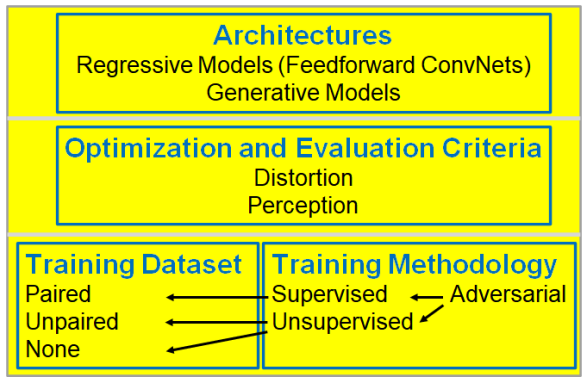


Figure 1.2: Three pillars of learned image restoration and SR.

1.5.1 Network Architectures: Regressive vs. Generative Models

In a very broad way, we can classify deep SR network architectures as regressive models and generative models. Regressive models are feedforward networks that learn a nonlinear mapping from the space of LR images to the space of HR images. They include residual networks, dense networks, and their variations. On the other hand, generative models learn the probability distribution of HR images conditioned on LR images. Thus, generative SR models enable sampling one or more HR images from the estimated conditional distribution of HR images. We provide an overview of recent advances in deep neural network architectures that contribute to achieving the state-of-the-art results in image/video restoration and SR in Chapter 2.

1.5.2 Optimization Criteria: Distortion vs. Perception

Unlike classical model-based methods, that optimize either l_2 or l_1 distortion subject to some regularization prior, learned image restoration and SR allows optimization with respect to any differentiable loss function. Parameters of the network can be optimized purely for distortion (fidelity) or a combination of fidelity and perceptual criteria. Blau and Michaeli, 2018 show that distortion and perceptual quality are at odds with each other leading to perception-distortion tradeoff. Specifically, they study the optimal probability for correctly discriminating the out-

puts of an image restoration algorithm from real images and show that as the mean distortion decreases, this probability increases indicating worse perceptual quality. Achieving the best trade-off between highest fidelity and perceptual quality is an interesting research problem. Fidelity and perceptual optimization criteria and perception-distortion tradeoff are reviewed in more detail in Chapter 3.

1.5.3 Training Methods and Data: Supervised vs. Unsupervised

A vast majority of published literature on learned image restoration and SR perform supervised training from a synthetically generated LR, HR paired image dataset. This dataset depends on a particular blur kernel and noise level that is used to generate LR images from corresponding HR images. SR models obtained this way perform incredibly well, outperforming conventional model-based methods by a large margin, when the test set of images are also generated using the same degradation process. However, if the degradation in the test set of images differ from those in the training set, then SR performance deteriorates. We call this dependence of SR performance on the degradation model used in the training set as model overfitting.

When it comes to real-world problems, this approach of training SR models based on synthetically generated LR-HR image pairs is of limited use due to model overfitting, because real LR images are degraded by blur and noise, which are unknown in the practical setting. Furthermore, in the real-world SR setting, there is no ground-truth; hence there is no paired data available for training. Hence, in the real-world setting we have blind image restoration/SR problem without ground-truth data.

Recently, more researchers have started working on blind image restoration/SR methods that require no training, or can be trained without an external training set, or can be trained by unpaired datasets. These methods can be classified as: i) two-step approaches, where the blur kernel is estimated first and then used in a non-blind SR model, or ii) methods that iteratively correct the blur kernel estimate based on the LR image and the most recent estimate of the SR image. Both supervised and unsupevised training of image and video SR models are discussed in Chapter 4 and Chapter 5, respectively.

1.6 Related Recent Survey Articles

Other survey articles have appeared in the literature while we are working on this manuscript. Some of them introduce a taxonomy for deep learned SR models grouping them into categories, some benchmark SR algorithms, and some are in preprint.

Wang *et al.*, 2021 provide a nice overview of the SISR literature; however, their paper does not cover transformer-based architectures, and touches upon video SR and real-world SR issues very briefly.

In deep journey into SR (Anwar *et al.*, 2021), the authors introduce a new taxonomy of the SR algorithms based on their architectures. They also provide a systematic evaluation of more than 30 SISR algorithms on six publicly available datasets given LR-HR image pairs. However, the assessment of results was only performed in terms of PSNR and SSIM; they do not discuss perception-distortion tradeoff, and they do not address real-world SR or video SR.

Liu *et al.*, 2020 propose a taxonomy and classify video SR methods into six sub-categories according to the ways they utilize inter-frame information in a preprint article. They also compare more than 30 video SR algorithms. Blind image SR (Liu *et al.*, 2021a) is another preprint article that surveys image SR methods that can deal with an unknown degradation. The authors propose a taxonomy to categorize existing methods into three different classes according to the ways they model the degradation process.

Unlike these surveys, we do not benchmark a set of algorithms or propose a new taxonomy, but we focus on the understanding of foundational ideas and provide a comprehensive overview of basic principles of regressive (predictive) and generative SR network architectures, approaches to enforce temporal consistency in video SR, full-reference and no-reference image/video quality assessment (QA) measures, and differentiable QA measures that can be used as optimization loss functions. We also discuss the real-world SR problem and survey how to deal with the cases of known degradation model and blind SR as well as unsupervised learning approaches for real-world SR in detail. We believe this article can be used as reference material in an advanced image processing class.

2

Modern Network Architectures

This chapter reviews recent advances in deep neural network architectures that led to significant performance improvements in learned image restoration and SR. Section 2.1 introduces convolutional networks (ConvNet), and discusses ConvNet architectures that are more easily trainable and have more expressive power. Section 2.2 presents self-organizing neural networks that learn the best nonlinearity for the task at hand. Section 2.3 introduces self-attention mechanism and visual transformers. Finally, Section 2.4 discusses the differences between regressive and generative neural models.

The origin of today's deep neural networks can be traced back to the binary neuron model (McCulloch and Pitts, 1943) and its extension to the perceptron (Rosenblatt, 1958). The McCulloch-Pitts neuron only allowed for binary inputs (with no weighting) and outputs, using the threshold step activation function. Later, Rosenblatt modified this binary neuron to the perceptron, which weighs different inputs with "learnable" coefficients, while still using the binary threshold (the Heaviside step function) activation. Limitations of the perceptron model was pointed out by (Minsky and Papert, 1969). In particular, they showed the perceptron could not implement the XOR and NXOR functions and

it could only classify linearly separable classes. As a result, few people continued to work in the area until the 1980's.

These difficulties were resolved by the multi-layer perceptron (MLP), which is a neural network consisting of multiple layers of perceptrons. Hornik *et al.*, 1989 have proven that MLPs are universal approximators. An MLP has an input layer, at least one hidden layer, and an output layer. Let's assume a particular layer of MLP has N input neurons (perceptrons) corresponding to $N = N_1 \times N_2$ pixels with K_1 channels represented by $N \times K_1$ matrix $\mathbf{x}$ and M outputs each with K_2 channels denoted by the $M \times K_2$ matrix $\mathbf{y}$. For each output channel $k = 1, \cdots, K_2$, we first compute an affine combination $\mathbf{z}$ of inputs

$$\mathbf{z}_k = \mathbf{W}_k \mathbf{x} + \mathbf{b}_k \tag{2.1}$$

where $\mathbf{W}_k$ is a tensor of weights with the shape $M \times N \times K_1$ for channel k of the layer, b_k is an $M \times 1$ bias vector. If the network is fully-connected, then the matrix $\mathbf{W}_k$ is fully populated. The output $\mathbf{y}_k$ for channel k of the layer is a nonlinear pointwise function $f(\cdot)$ of $\mathbf{z}_k$ given by

$$\mathbf{y}_k = f(\mathbf{z}_k) \tag{2.2}$$

The function $f(\cdot)$ is called an activation. The elements of tensor $\mathbf{W}_k$ and vector $\mathbf{b}_k$ are learnable parameters. End-to-end training of MLPs was made possible by the introduction of the back propagation algorithm (Rumelhart *et al.*, 1986). However, if we do not impose a structure on the tensor $\mathbf{W}_k$, then there are an enormous number of parameters for a typical image processing problem, which makes training impractical.

It is interesting to note that the first GPU implementation of an MLP, which resulted in a speed up by a factor of 20 compared to CPU implementation, was proposed by Oh and Jung, 2004.

2.1 Convolutional Networks (ConvNet)

2.1.1 Early ConvNets

The first neural network, which imposed a structure on the tensor $\mathbf{W}$ was the neocognitron (Fukushima, 1980) that was proposed for handwritten character recognition using simple and complex cells. The simple

cells perform a convolution and complex cells perform average pooling. However, the backpropagation algorithm was not known at the time, and success of the method was limited due to difficulty of training it.

Gradient based learning using the backpropagation algorithm to train a neocognitron-like architecture, called LeNet-5, for handwritten character and zip code recognition was first demonstrated by LeCun *et al.*, 1989. LeNet-5 architecture consists of an input layer, 2 convolutional layers with 5×5 kernels, 2 subsampling layers and 2 fully connected layers. The input layer takes 28×28 images and pads them to 32×32. The first convolutional layer outputs 6 feature channels 28×28 each, which are then subsampled to 14×14. The second convolutional layer outputs 16 channels 10×10 each, which are then subsampled to 5×5. Classification of these features is performed by 2 fully connected layers.

There are two factors that lead to reduction of the number of parameters in ConvNets: i) The tensor $\mathbf{W}$ consists of sparse blocks meaning neurons are connected only to their local neighbors (i.e., finite impulse response filtering), which is motivated by the fact that correlations in natural images are confined to local neighborhoods. ii) Parameter sharing across neurons, which means all sparse blocks in $\mathbf{W}$ have the same parameters, assuming that texture in natural images is homogeneous. Combining "sparsity" and "parameter sharing" constraints on $\mathbf{W}$, matrix multiplication in (2.1) can be implemented by convolutional filtering.

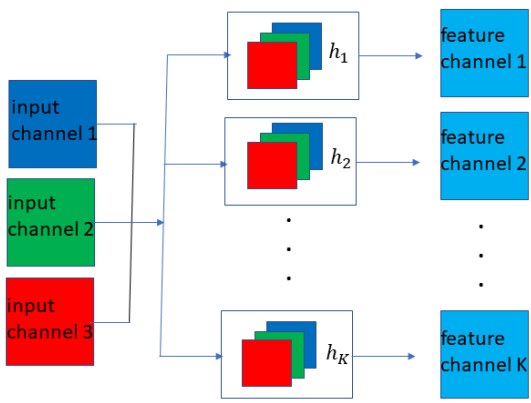


Figure 2.1: Illustration of a convolutional layer with 3 input and K output channels.

Implementation of a convolutional layer with three input channels and K output channels is illustrated in Figure 2.1. We have K convolutions with the filters $h_1, \cdots, h_K$, where the impulse response (kernel) of each filter is $L_1 \times L_2 \times 3$. We denote a convolutional layer with K_1 input channels, $L_1 \times L_2$ kernel for each input channel, and K_2 output channels with the notation "$K_1, L_1 \times L_2, K_2$"

We note that although each convolution layer has a small receptive field, e.g., 3×3 pixels using a 3×3 kernel, as we stack multiple convolution layers, the receptive field of later layers grows, e.g.,stacking 2 layers with 3×3 kernels, we get a 5×5 receptive field, and so on.

Following the work of Oh and Jung, 2004, several GPU implementations of ConvNets were proposed (Chellapilla *et al.*, 2006; Ciresan *et al.*, 2011; Krizhevsky *et al.*, 2012) for different image processing and computer vision applications. Among these, the success of AlexNet (Krizhevsky *et al.*, 2012) in the ImageNet Large Scale Visual Recognition Challenge 2012 has made a significant impact in ConvNets gaining popularity in the image processing/computer vision community.

2.1.2 Residual Networks (ResNet)

Early ConvNets, such as LeNet-5 (LeCun *et al.*, 1989), DanNet (Ciresan *et al.*, 2011), AlexNet (Krizhevsky *et al.*, 2012), were not very deep because it was observed that as we stack more and more layers, both the test and training performances of networks do not improve but rather first saturate and then start to degrade. The main reason for this phenomenon is the vanishing/exploding gradients problem. While normalizing inputs and intermediate layer outputs alleviates this problem for networks with 10-20 layers, it does not help with deeper networks.

The motivation for residual networks (He *et al.*, 2016a) is as follows: Consider two networks, one with L layers and a deeper network with $M > L$ layers. We expect the deeper network to perform at least as well as the shallower network. Clearly, the deeper model could achieve the performance of shallower model by replacing the first L layers of the deep network with the trained layers of the shallower network, and the remaining $M - L$ layers with an identity mapping. But this does not happen in practice. Residual networks address this problem.

We note that highway networks (Srivastava *et al.*, 2015), which feature shortcut connections with learnable gating functions, were concurrently proposed to address issues with training very deep networks.

Residual Blocks

ResNet is composed of residual blocks, where each residual block features a skip (or shortcut) connection to allow implementation of an identity mapping. Formally, a residual blocks approximates a nonlinear mapping $H(x) = F(x) + x$, where the second term (identity mapping) is implemented by a short-cut connection as depicted in Figure 2.2.

The original residual block design (He *et al.*, 2016b) is depicted in Figure 2.2(a). When multiple residual blocks are stacked together to form deeper networks, the gradients can flow directly through the skip connections backwards from later to initial filters to overcome the vanishing gradients problem. However, it was observed that the performance of a thousand layer ResNet is still worse than a hundred-layer ResNet. After testing different combinations, it was shown the configuration batchnorm-RELU-convolution depicted in Figure 2.2(b), called full pre-activation residual block, can alleviate the vanishing gradients problem better. ResNet formed by such blocks is referred to as pre-ResNet.

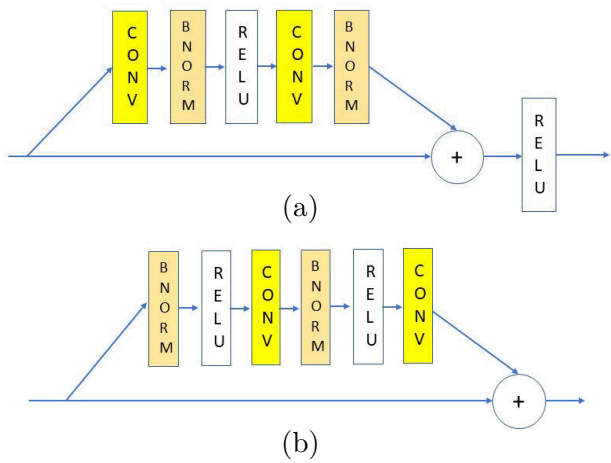


Figure 2.2: Illustration of residual block: (a) original, (b) full pre-activation.

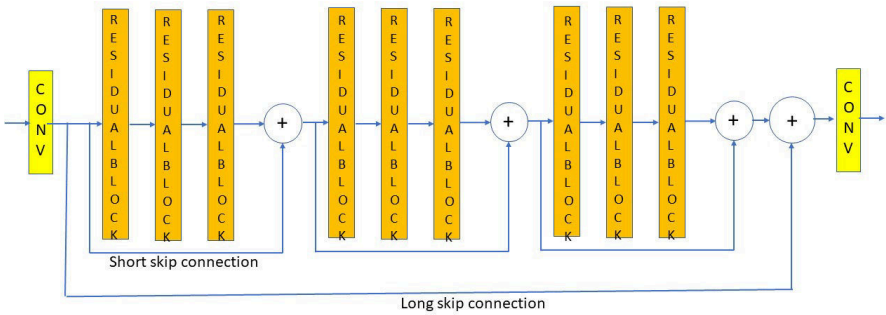


Figure 2.3: Illustration of residual networks of residual networks. The first convolution layer generates the desired number of feature channels from the input image. The last convolution layer generates the output image from the feature channels.

Residual Networks of Residual Networks

Many variants of the original ResNet have been proposed for improved performance. Among them, multi-level residual networks (Zhang *et al.*, 2018c), also known as residual networks of residual networks (RoR), is based on the hypothesis that the residual mapping of residual mapping is easier to learn than the original residual mapping. In particular, RoR adds level-wise shortcut connections upon original residual networks to improve the learning capability of the overall network. An example of RoR networks is depicted in Figure 2.3, where short-skip connections have been added as another level of skip connections for every three regular residual blocks. ResNet variants are among the most popular architectures used for image processing and computer vision tasks.

2.1.3 Densely Connected Networks (DenseNet)

A DenseNet is a type of ConvNet, which consists of multiple dense blocks that utilize dense connections between layers, where the output of a layer is concatenated to inputs of all later layers with matching feature-channel sizes. Whereas a traditional convolutional network with L layers have L connections, one between each layer and its subsequent layer, a DenseNet with L layers has $L(L+1)/2$ direct connections between layers. Since different layers have different receptive fields, this allows for a multi-resolution representation of feature channels.

Dense Blocks

A dense block is composed of multiple dense layers that concatenates the output of a layer to the inputs of all later layers with matching feature-map sizes. For each dense layer, the feature-maps of all preceding dense layers are used as inputs. The composition of a dense layer and a dense block are depicted in Figure 2.4(a) and (b), respectively. Figure 2.4(a) shows a dense layer with 64 channel input and 96 channel output, i.e., with the growth rate 32 channels. Note that the layer generates 32 new feature channels and concatenates them with the 64 input channels. The dense block shown in Figure 2.4(b) has 4 such dense layers, where each dense layer generates 32 new feature channels for a total of 192 feature channels at the output of the dense block.

A dense block differs from a residual block in the following ways: i) the number of feature channels in the input and output of a residual block are the same, whereas the output of a dense block has more channels than its input, ii) the input and output of a residual block are combined by addition, whereas the input and output of a dense block are combined by concatenation, iii) a residual block passes its outputs to only the next residual block as input, whereas a dense block passes its outputs to all later dense blocks as inputs.

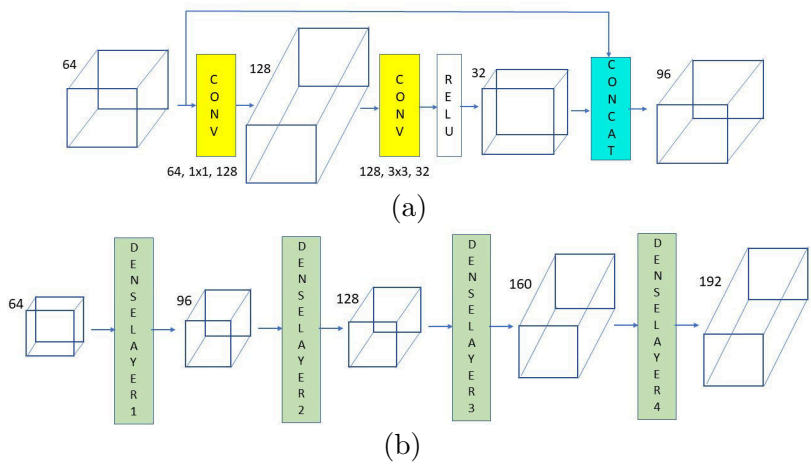


Figure 2.4: Illustration of (a) dense layer, (b) dense block with 4 dense layers.

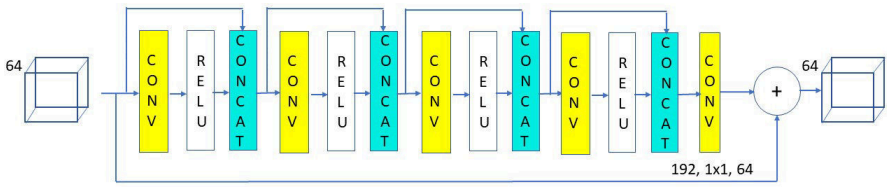


Figure 2.5: Illustration of a residual dense block.

Residual Dense Blocks

A residual dense block (Huang *et al.*, 2017) combines the best features of a residual block and a dense block. Its architecture is depicted in Figure 2.5. The output of each convolutional layer is concatenated with the outputs of previous layers. The 1×1 convolution layer reduces the number of output channels of the upper branch down to the number of input channels. This is because of the addition of the output of the upper branch with the input passed through the short-cut connection. Hence, the input and output of a residual dense block have the same number of feature channels.

Residual-in-Residual Dense Blocks (RRDB)

Inspired by the architecture of residual networks of residual networks, the residual-in-residual dense block (RRDB) (Wang *et al.*, 2018b) extends the residual dense block architecture with multiple levels of short-cuts for improved performance. The RRDB block is comprised of three residual dense blocks (RDBs) stacked back to back and an end-to-end shortcut as depicted in Figure 2.6.

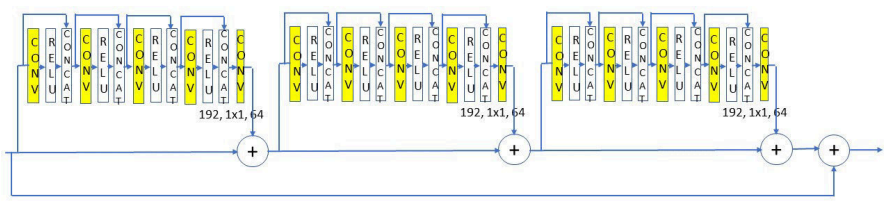


Figure 2.6: Illustration of a residual-in-residual dense block (RRDB).

2.2 Generative Neurons and Self-organized Residual Blocks

Various forms of ConvNets discussed so far are all built by stacking multiple layers of multi-channel perceptron neurons with sparse weight tensors. Perceptrons are limited in their expressive power because they employ linear combination of inputs. The only nonlinearity is the point-wise nonlinearity of the activation function. Recently, operational neural networks (ONN) and self-organized ONNs (Self-ONN) based on new generalized neuron models have been proposed (Kiranyaz *et al.*, 2021).

ONNs employ the generalized operational perceptrons (GOP) as their basic neuron model. A GOP is formed by a particular set of nodal, pool and activation operators from a pre-determined operator set library. The classical perceptron is a special case, where the nodal, pool and activation operators are multiplication, addition, and RELU, respectively. An optimal operator set per network layer can iteratively be searched during several short back-propagation (BP) training sessions.

A Self-ONN layer is formed by more expressive generative neurons, which are explained in detail in Section 2.2.1. They can learn to approximate any nonlinear function, without the limitation of an operator set library and are computationally more efficient. It has been shown that Self-ONNs can learn highly complex and multi-modal functions using few layers of generative neurons with minimal network complexity and training data because generative neurons have superior expressive power. While generative neurons can be employed to replace the perceptrons in any ConvNet architecture that is discussed above, the fact that Self-ONNs with few layers have excellent expressive power reduces the need for very deep networks; hence, the need for more complex architectures such as residual networks of residual networks and residual-in-residual dense blocks.

2.2.1 Generative Neurons

A generative neuron approximates a non-linear function $f(x)$ by a Taylor series expansion

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n \tag{2.3}$$

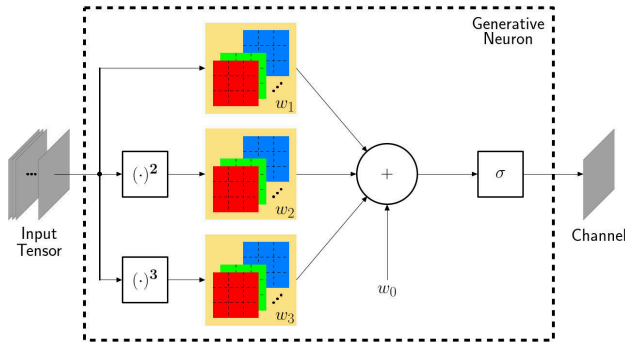


Figure 2.7: Illustration of a one-channel generative neuron for $q = 3$.

around the point a. If we truncate the series to q terms, we have an approximation $g(\mathbf{w}, x, a)$ given by

$$g(\mathbf{w}, x, a) \quad = w_0 + w_1(x - a) + \cdots + w_q(x - a)^q \tag{2.4}$$

where

$$w_n = \frac{f^{(n)}(a)}{n!}. \tag{2.5}$$

For a c-channel input tensor, the parameters $w_n, n = 1, \ldots, q$ denote q banks of c-channel convolution kernels and w_0 denotes a bias. These parameters can be learned by the classical back-propagation algorithm.

A generative neuron with 3×3 kernels, $a = 0$, $q = 3$, and activation function $\sigma()$ is illustrated in Figure 2.7. Each neuron takes c-channels as input and outputs a single channel. The activation function limits outputs within a range about the value a before they are input to the next neuron, since the Taylor series is expanded around a. So, for $a = 0.5$, $\sigma()$ can be taken as sigmoid that bounds the output in the range $[0\ 1]$, or if $a = 0$, $\sigma()$ can be $\tanh(x)$ to bound the outputs in the range $[-1\ 1]$. Note that if we choose $q = 1$ and $a = 0$, the generative neuron model reduces to the classic convolutional perceptron.

2.2.2 Self-Organized Residual Blocks

A self-organized residual (SOR) block can be obtained by replacing all regular convolutional layers in a residual block with self-organized layers (SOL) formed by generative neurons without the activation function $\sigma()$.

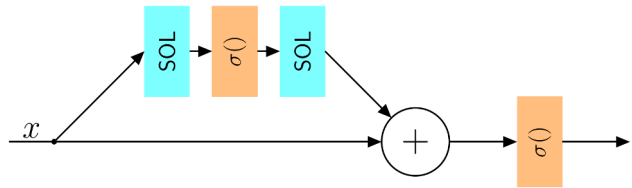


Figure 2.8: Illustration of SOR block (Keleş *et al.*, 2021a), where SOL stands for a layer formed by generative neurons without an activation function.

Figure 2.8 depicts a SOR block consisting of SOL, activation layer, and SOL, where we define the activation function $\sigma()$ as a separate layer, and another activation layer after the summation with the short-cut connection to soft-limit the output of the SOR block.

Using SOR blocks in place of regular convolutional residual blocks, any ConvNet architecture with residual blocks can be transformed into a self-organized residual network. The main advantage of SOR blocks over standard residual blocks is that we can obtain better performance with a fewer number of blocks, eliminating the need for very deep networks.

2.3 Self-Attention and Visual Transformers

The strong inductive bias of ConvNets has been the main motivation for adopting them as the backbone for image processing tasks. However, convolutions are not effective in capturing long range correlations in images because of their limited receptive fields. Given the improvements that can be obtained by non-local image processing, e.g. by using non-local means (Buades *et al.*, 2005), and the success of self-attention to effectively exploit long range interactions in sequence modeling tasks, there is growing interest in using self-attention layers either to augment ConvNets or as stand-alone primitives for image processing tasks.

We note that the objective and scope of self-attention layer is different from those of scaling attention mechanisms, such as channel-wise attention and spatial attention, which simply scale feature maps. Self-attention aims to capture long range interactions, and is defined as attention applied to a single context instead of across multiple contexts, i.e., the query, keys, and values are all extracted from the same content.

2.3.1 Global vs. Local Self-Attention

Global self-attention can be viewed as a form of non-local means (Buades *et al.*, 2005), which is a powerful tool in classical image/video processing. Non-local neural networks (Wang *et al.*, 2018a) relates self-attention to the more general class of non-local filtering operations in image and video processing. The proposed non-local module, an adaptation of dot-product attention, demonstrates significant improvements in several computer vision and image processing tasks. Attention augmented convolutional networks (Bello *et al.*, 2019) propose combining both convolutions and self-attention by concatenating convolutional feature maps with feature maps produced via self-attention. Experiments show that attention augmentation leads to consistent improvements in image classification and object detection across different models and scales.

We note that the memory requirement and computational complexity of global dependency modeling by dot-product attention is quadratic in image size. This prohibits its application to high-resolution images and large videos. In many applications, input images to self-attention layers need are downsampled for computational feasibility. Recently, an efficient implementation of self-attention with linear complexity has been proposed (Shen *et al.*, 2021). When scaling normalization is used, the efficient attention mechanism is mathematically equivalent to dot-product attention. On the other hand, when softmax normalization is used, the two mechanisms are approximately equivalent.

While above works show augmenting convolutional models with global self-attention achieves gains on different vision tasks, others investigate whether local self-attention can be used as a stand-alone primitive for vision models. Hu *et al.*, 2019 introduce a form of pixel-wise sliding window self-attention as a new image feature extractor, called the local relation layer, that adaptively determines aggregation weights based on the compositional relationship of local pixel pairs. Ramachandran *et al.*, 2019 propose a fully self-attentional model by replacing all instances of spatial convolutions in the ResNet with local relative self-attention. Their model outperforms the baseline on ImageNet classification with 12% fewer FLOPS and 29% fewer parameters. Ablation study shows self-attention has more impact at later layers.

2.3.2 Vision Transformers

Vision Transformer (ViT) (Dosovitskiy *et al.*, 2021) is an encoder network, with multiple layers of multi-head self-attention, which takes flattened image patches as inputs and treats them the same way as tokens (words) in an NLP application. A summary of the operation of ViT is as follows: i) Split an image into non-overlapping patches; ii) Flatten the patches into vectors; iii) Produce lower-dimensional linear embeddings from the flattened patches; iv) Add positional embeddings; v) Feed the sequence as an input to a standard transformer encoder; vi) Perform fully supervised pretraining of the model on a huge dataset; vii) Finetune the model on the smaller application dataset. A key insight of this work is that, like transformer models in NLP, ViT needs a sufficient amount of labelled training data to realize its potential.

ViT for dense prediction (Ranftl *et al.*, 2021) is an encoder-decoder network that uses ViT (Dosovitskiy *et al.*, 2021) as a backbone encoder. They reassemble the bag-of-tokens representation that is provided by ViT into image-like feature representations at various resolutions and progressively combine the feature representations into the final dense prediction using a convolutional decoder. Unlike fully-convolutional networks, the ViT backbone does not perform explicit downsampling after an initial image embedding has been computed; hence, it maintains a representation, which has constant dimensionality and a global receptive field throughout all stages.

Unlike ViT that produces feature maps of a single low resolution and have quadratic computation complexity in input image size due to computing self-attention globally, Shifted windows (Swin) Transformer (Liu *et al.*, 2021b) proposes a hierarchical representation with shifted windows at various scales. The shifted windowing scheme has linear computational complexity with respect to image size due to computing self attention over non-overlapping local windows, while also allowing for cross-window connections. The shift and scale variables of the Swin transformer representation resembles the same properties of the wavelet transform.

Swin Transformer V2 (Liu *et al.*, 2022) addresses difficulties with scaling up model capacity and input image resolution of the original

model. The authors observe instability issues in training large models. In addition, effective transfer of models pre-trained at low resolution to higher resolution images requires some modifications. Another problem is the GPU memory consumption when the image resolution is high. To address these issues, the authors propose: 1) a post normalization technique and a scaled cosine attention approach to improve stability of training large vision models; 2) a log-spaced continuous position bias technique to effectively transfer models pre-trained with low-resolution images and windows to their higher-resolution counterparts. In addition, they also share important implementation details that lead to significant savings of GPU memory consumption and thus make it feasible to train large vision models with regular GPUs.

2.4 Regressive Models vs. Generative Models

From a deterministic perspective, multi-layer feedforward networks learn to approximate any continuous nonlinear mapping between two spaces by a composition of simpler functions (affine maps). Universal approximation theorem (UAT) (Hornik *et al.*, 1989) states that multi-layer neural networks can represent a wide variety of nonlinear functions with desired accuracy when given appropriate weights. Note that UAT is an existence theorem, i.e., it does not provide a construction for the weights. Finding a set of good weights is the subject of appropriate training procedures. In the particular case of image restoration and SR problem, feedforward networks learn a nonlinear regressive mapping from the space of LR images to the space of HR images.

Alternatively, from a probabilistic perspective, we can consider the observed data, S, as a finite set of samples from an underlying distribution, $p_S(s)$. Real world images are highly structured and are contained in a low dimensional manifold of a very high dimensional space. Recall that a 100×100 image patch is a point in 10,000 dimensional space. Discovering the underlying structure of this low-dimensional manifold is key to learning generative models. The goal of a generative model is to learn the data distribution $p_S(s)$ given the dataset S. We would be seeking a parametric approximation to the actual data distribution, which minimizes some notion of distance between the model

distribution and the actual data distribution. In the particular case of image restoration and SR problem, we would be learning the distribution of HR images conditioned on given LR images. Then, an estimate of the SR image is computed by sampling from this learned distribution conditioned on the given LR image. Different types of generative models include autoregressive (AR) models, variational autoencoders (VAE), generative adversarial networks (GAN), normalizing flow (NF) models, and diffusion models. Among these GANs and NF models are extensively used for image restoration and SR.

GAN was proposed as a powerful framework for synthesizing natural images with high perceptual quality (Goodfellow *et al.*, 2014). The GAN framework simultaneously trains two models as adversaries: a generator G and a discriminator D, where G learns to synthesize real-looking images and D learns to estimate the probability that its input sample is a synthetic (generated) vs. a real (ground-truth from the training set) image. This framework corresponds to a minimax two-player game, where G learns the distribution of training data using the outputs of D as a penalty term added to its loss and D yields $\frac{1}{2}$ for all inputs at equilibrium. The entire system can be trained using backpropagation when G and D are defined by neural networks.

NF models provide a general methodology for constructing arbitrary probability distributions over continuous random variables. Let $\mathbf{x}$ be a D-dimensional real vector, and suppose we would like to define a joint distribution over $\mathbf{x}$. The main idea is to express $\mathbf{x}$ as a transformation T of a real vector $\mathbf{z}_0$ sampled from a simple base distribution $p_{Z_0}(\mathbf{z}_0)$, i.e., $\mathbf{x} = T(\mathbf{z}_0)$. Flow methods construct arbitrarily complex densities by composing several simple transformations, i.e., $T = T_K \circ \cdots \circ T_1$ and applying the change of variables formula successively. NF models exhibit several key advantages over GAN-based generative models, such as monotonic converge and stable training. In addition, flow-based generative models learn to produce a diverse set of sample images, where the diversity of solutions increases with the temperature τ of latent variables.

We discuss the application of GAN and NF models to image SR problem in Sections 4.4.2 and 4.4.4, respectively.

3

Optimization and Evaluation Criteria

There exist a dilemma in the evaluation of the output of image restoration and SR systems: Whether the output should be as close as possible to a pristine original (i.e., fidelity), or it should be pleasing to a human observer (i.e., perceptual quality). In general, these two requirements are in conflict (Blau and Michaeli, 2018). We need to choose the correct optimization criterion for the application at hand since an optimized system is only as good as the optimization criterion used to design it. If the goal of restoration/SR is to recover information, e.g., whether a number is 3 or 8, a fidelity criterion such l_1 or l_2 norm should be appropriate. On the other hand, if the goal is to restore a visually pleasing image/video, then a perceptual criterion should be used.

Traditionally, the l_2 norm of pixel-wise error or the mean-square error (MSE) has been used for both optimization and evaluation of image processing systems. The MSE has been commonly used as an optimization criterion (or loss function) because it leads to closed-form linear estimators when signal and noise are independent and Gaussian distributed. Peak-signal to noise ratio (PSNR), which is a logarithmic function of the MSE normalized for signal dynamic range, is used as a measure of fidelity to evaluate the results. However, it is well-known

that simple measures of fidelity, such as l_1 or l_2 error, do not correlate well with human evaluation of perceptual quality.

With the recent advances in deep neural networks, we can now easily implement nonlinear (neural) signal processing algorithms that are optimized for any desired perceptual evaluation criterion, which is differentiable in the unknowns. Hence, the long standing goal of perceptual image/video processing can now be realized by nonlinear neural signal processing; i.e., by means of deep networks optimized for perceptual criteria. This is the most promising aspect of employing neural methods for image/video restoration and SR. Hence, the grand challenge is to find image quality assessment (IQA) and video quality assessment (VQA) measures that correlate well with human judgement.

The safest way of evaluating perceptual quality is to solicit the opinion of human observers. However, conducting subjective evaluation experiments is not only difficult and expensive, but also the results cannot be incorporated into restoration/SR systems as an optimization criterion. Hence, it is desirable to design objective measures of perceptual image/video quality in a way that is consistent with subjective human evaluation. Although there exist many proposals, no universally accepted objective measure of perceptual image quality currently exists that can robustly evaluate images/video like humans. Various proposals for IQA/VQA measure differ in their use of knowledge about the reference (original) image, a model of the human visual system (HVS), and the type of distortion. Prior art include *hand-coded models*, which typically fail to model the complexity of the HVS, *machine learning models* that are trained on human-labeled datasets, which are specific to distorted types and prone to human labeling errors, and more recent *deep learning models*, which yield more promising results.

This chapter reviews recent advances in IQA/VQA measures. We classify IQA measures as full-reference (FR) distortion measures, which are reviewed in Section 3.1 and no-reference (NR) perceptual measures, which are reviewed in Section 3.2. Section 3.3 reviews recent advances in VQA measures. We discuss IQA measures that can be used as optimization criteria in learned image/video restoration and SR methods in Section 3.4. We conclude this chapter with a discussion of perception-distortion trade-off in image/video restoration and SR in Section 3.5.

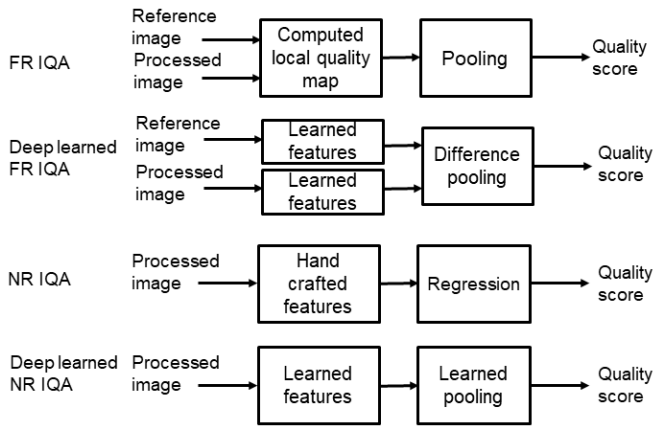


Figure 3.1: Classification of IQA measures.

3.1 Full-Reference Image Quality Assessment Measures

The goal of a FR IQA measure is to compare two images, where one of them is a pristine original and the other is distorted, by means of an objective score that evaluates the degree of fidelity/similarity or, conversely, the amount of distortion/dissimilarity between them. FR IQA measures are classified as hand-crafted vs. deep-learned as depicted in Figure 3.1. The simplest FR fidelity/distortion measure is the pixel-wise l_p norm that is reviewed in Section 3.1.1. We next discuss some hand-crafted perceptually-inspired FR IQA measures: Those based on modeling the HVS are discussed in Section 3.1.2, structural similarity index measure (SSIM) is presented in Section 3.1.3, and visual information fidelity (VIF) is explained in Section 3.1.4. Finally, deep learned FR IQA measures are discussed in Section 3.1.5.

3.1.1 l_p Norm and PSNR

The simplest fidelity/distortion measure is the l_p norm between two image vectors $x_i, i = 1, \cdots, N$ and $y_i, i = 1, \cdots, N$ that are formed by lexicographical ordering of pixels. The l_p norm is defined

$$D_p = \left(\frac{1}{N} \sum_{i=1}^{N} |x_i - y_i|^p \right)^{1/p} \tag{3.1}$$

where N is the number of pixels. When $p = 2$, D_p becomes the root mean square error (RMSE). However, it is more common to work with the MSE given by

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2 \tag{3.2}$$

The PSNR is a logarithmic function of the MSE that is normalized with respect to the dynamic range of the image. For images with L level dynamic range, e.g., $L = 255$ for 8-bit images, we have

$$PSNR = 10\ log_{10}\frac{L^2}{MSE} \tag{3.3}$$

MSE/PSNR can be computed in the RGB or YCrCb (luminance-chrominance) space. Different ways for computing the PSNR for sets of images and video are discussed in (Keleş *et al.*, 2021b).

The MSE defines the average energy of the difference/error image/frames, which is preserved under any orthogonal (or unitary) linear transformation, such as the Fourier transform (see Parseval's theorem). The MSE possesses nice properties, such as convexity, symmetry, and differentiability, which are desirable in the context of optimization. Minimum-MSE (MMSE) optimization problems often have closed-form analytical solutions, and when they don't, iterative numerical optimization procedures are often easy to formulate, since the gradient and the Hessian matrix of the MSE are easy to compute. The MSE is also a desirable measure in the statistics and estimation framework (where the sample average in Eqn. 3.2 is replaced by statistical expectation).

The limitations of MSE/PSNR as a fidelity measure are discussed in the paper by Wang and Bovik, 2009. In summary: 1) MSE is independent of spatial relations between pixels; i.e., if the original and distorted image pixels are randomly shuffled in the same way, then the MSE between them is unchanged. 2) MSE does not consider any relation between the original and error images; i.e., for a given error image, the MSE remains unchanged, regardless of which original image it is added to. 3) MSE is independent of the signs of the error samples. 4) MSE assumes all pixels are equally important to image fidelity.

Many perceptual FR IQA measures, both hand-crafted and machine learning based, have been proposed to address these limitations. They

can be classified as HVS-based, structure-based, statistics-based, and learning-based measures, which are reviewed in the following subsections.

3.1.2 Measures based on Modeling Human Visual System

There has been considerable progress in mathematical modeling of the HVS in the past 50 years. Examples of HVS effects include the just noticeable difference (JND), brightness adaptation, and spatial/temporal masking. In a pioneering work on finding a distortion measure that is in agreement with subjective evaluation of compressed images, Mannos and Sakrison, 1974 expressed the sensitivity of human observers to gray-scale errors at different spatial frequencies by means of a contrast sensitivity function (CSF). In another influential work, Daly, 1993 proposed a visible differences predictor incorporating multiple visual effects. Traditional HVS-based models involve 1) a preprocessing step including a point-wise nonlinear transform, simulation of eye optics by low-pass filtering, and color space transformation, 2) a channel decomposition step that transforms images into different spatial frequency/orientation selective subbands, 3) an error normalization step that weighs the error in each subband to incorporate the error-sensitivity of HVS for different subbands and between-coefficient error contrast masking, and 4) an error pooling step that combines the errors in different subbands into a single measure.

More recently, researchers proposed modified PSNR measures that take HVS models into account. These modified PSNR measures have been based on wavelet or DCT domain HVS models or by simple block-wise weighting of MSE. Visual signal-to-noise ratio (VSNR) (Chandler and Hemami, 2007) is a wavelet-based measure that exploits *contrast detection threshold* property of the HVS. It is a two-stage approach, where the first stage determines whether the distortions are below the threshold of visual detection, and the second stage quantifies the distortions that exceed the threshold. VSNR is a low-complexity measure that appears to be competitive with other IQA algorithms. Alternatively, Ponomarenko *et al.*, 2007 develop a model of between-coefficient contrast masking of DCT basis functions. For each DCT coefficient of 8x8 image blocks, the model allows calculation of the maximal distortion that is

not visible due to the between-coefficient masking. A modification of the PSNR, called PSNR-HVS-M, that takes into account the calculated between-coefficient masking and the contrast sensitivity function is proposed. Another approach that proved effective is simple block-wise weighting of the MSE based on spatial activity of each block, resulting in WPSNR and XPSNR measures (Helmrich *et al.*, 2020).

These methods are general purpose, in the sense that they do not assume any specific distortion type or viewing conditions.

3.1.3 Structural Similarity Index Measure (SSIM)

The structural similarity index measure (SSIM) (Wang *et al.*, 2004) is based on the assumption that the HVS is highly adapted to extract structural information from the viewing field and a measure of change in structural information can provide a good approximation to perceived image distortion/quality. SSIM quantifies the similarity of three properties of local image patches, the luminance (brightness) values $l(x, y)$, the contrast $c(x, y)$, and the structure $s(x, y)$, given by

$$\begin{aligned} SSIM(x,y) &= l(x,y)c(x,y)s(x,y) \\ &= \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \cdot \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \cdot \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \end{aligned} \tag{3.4}$$

where μ_x and μ_y are the local sample means of x and y, and σ_x and σ_y are the local sample standard deviations of x and y, respectively, and σ_{xy} is the sample cross correlation of x and y after removing their means. The constants C_1, C_2, and C_3 are small positive values so that near-zero sample means, variances, or correlations do not lead to numerical instability. SSIM usually works well even if we set $C_1 = C_2 = C_3 = 0$.

Multi-scale SSIM (MS-SSIM) (Wang *et al.*, 2003) is an extension of SSIM that incorporates image details at different resolutions to account for the facts that the perceivability of image details depends on the sampling density of the image and the viewing distance.

3.1.4 Visual Information Fidelity (VIF)

Natural scenes form an extremely small subspace (manifold) of the space of all possible images. Many researchers have attempted to characterize

this subspace of natural images by statistical modeling. Measurement of image fidelity has also been formulated within an information communication framework based on statistics of natural images, where the transmitter involves a scene, light source(s), atmosphere conditions, and sensing/recording devices; the channel is any transmission/storage processing that degrade the image; and the receiver includes display devices and the HVS (Wang and Bovik, 2009).

Sheikh *et al.*, 2005 proposed an information-theoretic approach to quantifying visual fidelity by means of an Information Fidelity Criterion (IFC) derived based on natural scene statistics. Given an original and distorted image, the visual fidelity of the distorted image can be quantified based on the amount of information it provides about the original. The images are modeled as realizations of a mixture of marginal Gaussian densities chosen for wavelet subband coefficients, and visual fidelity is quantified based on the mutual information between the coefficients of the original and distorted images.

Visual Information Fidelity (VIF) (Sheikh and Bovik, 2006) is an extension of this model that assumes the reference (original) image is the output of a stochastic source, which passes through the HVS channel and is processed by the brain. The information content of the original image is quantified as the mutual information between the input and output of the HVS channel. The same information measure is then calculated for the distorted test image. The VIF measure then computes the ratio of these two information values calculated over wavelet subbands to form a visual information fidelity measure that relates visual quality to relative image information. Similar to the SSIM, the VIF measure has been shown to perform well for a variety of suprathreshold distortions.

3.1.5 Deep-Learned FR Perceptual IQA Measures

There are some FR IQA measures that apply learning methods to handcrafted features computed from the reference and test images. Instead we focus on deep learning based methods whose inputs are raw images.

DeepSim (Gao *et al.*, 2017) is one of the first to relate perceived image quality to similarity in a feature space. It measures local similarities between the features of the reference and test images computed by

a ConvNet model. Various pooling strategies, such as deviation pooling, per-centile pooling, and average pooling, are then explored to integrate the local quality indices into an overall image quality score. Learned perceptual image patch similarity (LPIPS) (Zhang *et al.*, 2018d) shows that deep features, trained on supervised, self-supervised, and unsupervised objectives, model low-level perceptual similarity surprisingly well, outperforming widely-used metrics on a large-scale perceptual similarity dataset containing various distortions and 484k human judgments.

Perceptual image-error Assessment through Pairwise Preference (PieAPP) (Prashnani *et al.*, 2018) is based on the observation that it is much easier for humans to compare two given images to identify the one that is more similar to a reference than to assign quality scores to each. Their training dataset is formed as follows: Given two distorted versions (A and B) of a reference image R, subjects are asked to select the one that looks more similar to R. They store the percentage of people who selected image A over B as the ground-truth label for this pair, which is called the probability of preference of A over B. They train a deep-learning model using the proposed pairwise-learning framework to predict the preference of one distorted image over the other. They show that perceptual error estimated by PieAPP is well-correlated with human opinion, while also generalizing to new kinds of distortions.

In Ding *et al.*, 2020, the authors construct an injective and differentiable function that transforms images to multi-scale overcomplete representations using a convolutional neural network. After transforming the original and corrupted images, they construct the Deep image structure and texture similarity (DISTS) measure combining two terms over all feature maps: one that compares the spatial averages (and thus, the texture properties) of the two images, and a second that compares the structural details. The final distortion score is computed as a weighted sum of these two terms, with the weights adjusted to match human perception of image quality and invariance to resampled texture patches. Experiments show that the optimized method explains human perceptual scores, both on conventional image quality databases, as well as on texture databases. The method is relatively insensitive to geometric transformations (e.g., translation and dilation), without use of any specialized training or data augmentation.

NTIRE 2021 challenge on perceptual image quality assessment (PIQA) (Gu *et al.*, 2021) aims at benchmarking PIQA methods. Perceptual image processing algorithms based on the GAN framework produce images with more realistic textures. Hence, the training and testing datasets in this challenge include outputs of selected perceptual image processing algorithms and the corresponding subjective scores, in order to develop and evaluate IQA methods on GAN-based distortions. LPIPS, PIEAPP and DISTS measures have been chosen as baseline in this challenge. LIPT team was the winner. They develop an image quality transformer (IQT), introduced in (Cheon *et al.*, 2021), that applies a transformer architecture to the perceptual IQA task.

3.2 No-Reference Perceptual Image Quality Assessment

In real-world image restoration and SR, a ground-truth (reference) image is not available. Hence, one needs a no-reference (NR) IQA method to predict the perceptual quality of images without referring to an undistorted original.

Early NR-IQA methods focused on specific known distortion types, such as blocking and blurring, extracted distortion-specific features based on a model of assumed distortion type, and achieved successful results (Ferzli and Karam, 2009). Clearly, the application scope of these methods are limited. This section focuses on generic NR-IQA with unknown complex distortions, which is a more challenging problem.

NR-IQA methods can be classified as traditional vs. deep learned as shown in Figure 3.1. The traditional methods derive a score on the basis of hand-crafted features, whereas deep-learned methods take raw images as input and generate a score based on learned features.

The methods can also be opinion-aware or opinion-unaware. The goal of an opinion-aware IQA method is to learn to predict scores that correlate well with human judgement of image quality. To this effect, they depend on databases of distorted images annotated with the average mean opinion score (MOS) or differential MOS (DMOS) of human evaluators. Opinion-unaware methods only use natural scene statistics of pristine (undegraded) images without any need for degraded training images. We discuss some example approaches in the following.

3.2.1 Opinion-Aware Traditional NR-IQA Measures

In the training stage of opinion-aware methods, hand-crafted feature vectors are extracted from distorted images, and then, a mapping is learned from feature space to quality scores using a regression module, e.g., a support vector machine regressor (SVR) based on an annotated set of training images. In the inference stage, a feature vector is extracted from each test image, and then fed into the learned regression model to predict the quality score. In these methods, handcrafted features are usually selected to model natural scene statistics.

No-reference image quality assessment using visual codebook (Ye and Doermann, 2011) extracts a visual codebook consisting of Gabor-filter-based local features from local image patches to capture complex statistics of a natural image. The codebook encodes statistics by quantizing the feature space and accumulating histograms of patch appearances. Blind referenceless image spatial quality evaluator (BRISQUE) (Mittal *et al.*, 2012) is based on the observation that the mean and variance normalized luminance coefficients of natural images follow a Gaussian distribution, while the distribution of distorted images not.

These NR-IQA models do not assume any specific types of distortions; however, they do not generalize well to degradations unseen in training samples, and hence require the types of distortions in the inference samples to match those in the training examples. As a result, they are not very effective to evaluate the quality of image restoration and SR results unless they are trained on such datasets.

3.2.2 Opinion-Unaware Traditional NR-IQA Measures

Opinion-unaware NR IQA models only make use of measurable deviations of statistics of a given image from statistical regularities observed in natural images, without training on human-rated distorted images or any exposure to distorted images. These methods are based on the hypothesis that natural images possess statistical regularities, which are altered in the presence of distortions (e.g., blur and noise). Hence, perceptual quality of images are measured in terms of quantifiable deviation from natural image statistics. The performance of opinion-unaware methods can exceed that of opinion-aware ones for complex distortions.

The Natural Image Quality Evaluator (NIQE) (Mittal *et al.*, 2013) constructs a 'quality aware' collection of statistical features based on a simple space domain natural scene statistic (NSS) model. Specifically, they use locally mean subtracted and contrast normalized (MSCN) luminance and products of pairs of adjacent MSCN values as features. These features are derived from a corpus of natural, undistorted images. Experimental results show that the NIQE delivers performance comparable to top performing NR IQA models that require training on large databases of human opinions of distorted images.

An extension of NIQE, called the Integrated Local NIQE (IL-NIQE) (Zhang *et al.*, 2015), introduces three additional quality-aware features and fits the feature vector of each patch of the test image to a multi-variate Gaussian (MVG) model, and compute a pooling of local quality scores instead of using a single global MVG model to describe the whole image. This improved NIQE model captures local distortion artifacts more comprehensively.

3.2.3 Deep-Learned NR-IQA Measures

Neural networks are very powerful nonlinear regressors that can be trained to predict scores directly from raw images; however, training them requires huge amounts of annotated data. Unfortunately, it is difficult to create large annonated image quality datasets for training neural IQA models, since annotating image quality by human observers is extremely expensive and time-consuming. Furthermore, many of the available datasets are for specific degradation types, e.g., for compression artifacts such as blocking, ringing, etc. Deep learned NR-IQA in the wild is still an active research problem of interest.

To address the small training dataset problem Kang *et al.*, 2014 consider 32×32 patches rather than images, thereby augmenting the number of training samples. Bianco *et al.*, 2018 propose to use a pre-trained network to mitigate the same problem. They fine-tune a pre-trained model for the IQA task on a small scale IQA dataset. The resulting SVR model, called DeepBIQ, maps features of sub-regions of the image to IQA scores and then estimates image quality by average-pooling the scores predicted on multiple sub-regions of the image.

An alternative approach to address lack of large annotated IQA datasets is to pre-train an IQA model using synthetically generated ranked image pairs (Liu *et al.*, 2017). While human annotated IQA ground-truth data is difficult to obtain, it is easy to generate distorted and undistorted image pairs from large unlabelled datasets with known rankings. For example, various levels of blur can be applied on reference images, where we know blurred images are of lower quality. A Siamese Network is trained to rank given image pairs according to image quality using such synthetically generated training sets with ground-truth rankings by forward propagating a batch of images through the network and backpropagating gradients derived from all pairs of images in the batch. The parameters of the resulting RankIQA model are then fine tuned on small image IQA datasets in order to output IQA scores.

Another strategy to tackle deep-learned IQA in the wild has been meta learning. The underlying idea of MetaIQA (Zhua *et al.*, 2020) is to learn the meta-knowledge humans employ when evaluating image quality, which generalizes well to unknown distortions. Specifically, the authors first undertake a number of NR-IQA tasks for different distortions. Then, meta-learning framework is adopted to learn the prior knowledge shared in evaluating images with diversified distortions. Finally, the quality prior model is fine-tuned on a target NR-IQA task to obtain the final IQA model. Experimental results suggest that the meta-model learned from synthetic distortions can be easily generalized to authentic distortions, which is highly desired in real-world applications.

3.2.4 NR-IQA for the SR Task: Perception Index (PI)

Different measures for no-reference perceptual quality evaluation of SR images have been proposed. One way of evaluating perceptual quality is by means of real vs. fake analysis, where human observers evaluate whether a test image is real or the output of an algorithm similar to the idea underlying adversarial training (Goodfellow *et al.*, 2016). Then, perceptual quality can be defined as the probability of success in such discrimination experiments, which is proportional to the distance between the distribution of the test image and that of natural images. Accordingly, the KL distance between the distributions of ground truth

and reconstructed images is used as a proxy for perceptual quality to explain the perception-distortion trade-off (Blau and Michaeli, 2018). However, it is difficult to employ this concept as a per image perceptual IQA measure.

Recently, Ma *et al.*, 2017 adapted opinion-aware NR-IQA methods to the SR task. To this effect, they conducted human subject studies using a large set of SR images and propose an NR-IQA measure learned from visual perceptual scores for SR images. They design three types of low-level statistical features in both spatial and frequency domains to characterize SR artifacts, and learn a two-stage regression model to predict the quality scores of SR images without referring to HR ground-truth images. Experimental results show that the proposed metric is effective and efficient to assess the quality of SR images based on human perception. However, human ratings of image quality can be noisy since quality rating scales can vary from evaluator to evaluator.

The perception index (PI) was proposed as a NR perceptual quality measure in the PIRM Challenge (Blau *et al.*, 2018). PI combines the SR-task specific opinion-aware NR IQA measure of (Ma *et al.*, 2017) and the opinion-unaware NR IQA measure NIQE (Mittal *et al.*, 2013) as

$$PI = \frac{1}{2}\left((10 - Ma) + NIQE\right) \tag{3.5}$$

Note that a lower PI indicates better perceptual quality. Comparison of the correlation between PI scores and human-opinion scores on the top 10 submissions in the PIRM Challenge shows that PI is highly correlated with the ratings of human observers (Blau *et al.*, 2018). This provides empirical evidence that PI can faithfully assess perceptual quality similar to subjective evaluations. PI has also been used in other perceptual SR Challenges including NTIRE 2020 Challenge on Perceptual Extreme Super-Resolution (Zhang *et al.*, 2020a).

Another learned NR-IQA measure used in SR Challenges is the Fréchet Inception Distance (FID) score (Heusel *et al.*, 2017), which measures the similarity between real and fake samples by fitting a multi variate Gaussian (MVG) model to the intermediate representation for the real and fake samples, respectively. In the case of FID, again lower scores indicate a better model.

3.3 Video Quality Measures

Generic video quality assessment (VQA) methods are expected to work across a range of distortion types. They are classified as FR VQA methods, which assume the availability of a pristine reference video, and NR VQA methods, which do not have access to a reference video.

A straightforward way of solving the VQA problem would be to consider the frames of a video as images and apply IQA measures discussed in the previous section to each frame and pool the frame level quality scores. However, this approach completely ignores the temporal aspect of video and any violation of temporal consistency of frames.

Videos are spatio-temporal signals, which carry both spatial and motion information. Motion plays a very important role in human perception of video. Hence, in VQA, we need to consider the temporal consistency of frames in addition to spatial quality of texture in each frame. Temporal inconsistencies result in jitter, which causes low perceptual quality even if the texture in each frame looks quite natural.

Popular traditional FRVQA methods that incorporate both spatial and motion/temporal information based on optical flow include the MOVIE index (Seshadrinathan and Bovik, 2010) and FLOSIM (Manasa and Channappayya, 2016). The MOVIE index quantifies the error in the optical flow of the distorted and reference video over several spatio-temporal frequency bands computed using spatio-temporal Gabor filters. It then pools these errors to form the perceptual quality score. FLOSIM is based on local optical flow statistics, which are shown to be sensitive to distortions in video. The deviation of test video optical flow statistics from those of the pristine video is quantified as the perceptual quality score of the test video.

More recently, VQA measures that involve machine learning or deep learning have become popular. Perhaps one of the most well-known FR VQA measures is Video Multi-method Assessment Fusion or VMAF (Blog, 2016), which has enjoyed popularity in the image and video compression community, since it has been trained specifically for image/video compression applications on video samples with compression artifacts. It has been shown that VMAF provides very accurate scores for image and video compression tasks. In the following,

we discuss some examples of generic VQA measures or those that have been applied to video restoration and SR.

Chu *et al.*, 2020 propose two new FR metrics, tOF and tLP, to measure temporal consistency. tOF measures pixel-wise difference of estimated optical flow, and tLP measures perceptual change in time, as

$$tOF = ||OF(g_{t-1}, g_t) - OF(\hat{x}_{t-1}, \hat{x}_t)||_1 \quad (3.6)$$

$$tLP = ||LP(g_{t-1}, g_t) - LP(\hat{x}_{t-1}, \hat{x}_t)||_1 \quad (3.7)$$

where OF denotes estimated optical flow, LP denotes the LPIPS metric, and g_t and $\hat{x}_t$ are the ground-truth and estimated SR video, respectively. In tLP, the behavior of the reference is also considered, as natural videos exhibit a certain degree of change over time. In conjunction, both pixel-wise differences and perceptual changes are crucial for quantifying realistic temporal coherence.

An example of NR VQA is the Fréchet Video Distance (FVD) measure (Unterthiner *et al.*, 2019). FVD builds on the principles underlying Frechet Inception Distance (FID) (Heusel *et al.*, 2017), which is a NR IQA measure. The authors introduce a feature representation that captures the temporal coherence of the content of a video, in addition to the quality of each frame. Unlike popular FR metrics such as PSNR or the SSIM index, FVD considers a distribution over videos, thereby avoiding the drawbacks of pixel/frame level metrics.

A recent neuroscience study (Henaff *et al.*, 2019) hypothesizes that the brain transforms incoming visual input streams to straighten their temporal trajectories, enabling temporal prediction through linear extrapolation. They present a neural network model of early human visual processing, which can reproduce this perceptual straightening property. They show that perceptual representations of frames extracted from a natural video using their neural model follow a straight temporal trajectory, whereas for unnatural video with artifacts the temporal trajectory is not straight. Kancharla and Channappayya, 2021 presents a video super-resolution method motivated by the perceptual straightening hypothesis of the human visual system, which is discussed in Section 5.4.

NR VQA using natural spatio-temporal scene statistics (Dendi and Channappayya, 2020) propose a video representation that is based on a parameterized statistical model for the spatio-temporal statistics of

mean subtracted and contrast normalized (MSCN) coefficients of natural videos. Specifically, they propose an asymmetric generalized Gaussian distribution (AGGD) to model the statistics of MSCN coefficients of natural videos and their spatio-temporal Gabor bandpass filtered outputs. They then demonstrate that the AGGD model parameters serve as good representative features for distortion discrimination. Based on this observation, they propose a supervised learning approach using support vector regression (SVR) to address the NR VQA problem.

3.4 Quality Measures for Optimization of Image Processing

While the emphasis so far has been on discussing measures for evaluation of image/video processing algorithms, another important question is which measures are desirable optimization loss functions in learned image processing. Clearly, IQA models that can be used as loss functions should be continuous and differentiable and should be of low complexity.

In a recent study, Ding *et al.*, 2021 have performed a large-scale comparison of IQA models in terms of their use as objectives for optimization of image processing algorithms for denoising, deblurring, super-resolution and compression tasks. Their findings indicate that DISTS and LPIPS offer the best performance as loss functions for all tasks except denoising, where optimizing for MS-SSIM yields the best results. However, high computational complexity and lack of interpretability of deep-learned FR measures may hinder their wide-spread use, while ℓ_1 loss and MS-SSIM are still valuable for optimizing image processing systems due to their robustness and simplicity.

3.5 Perception - Distortion Trade-off

Distortion refers to lack of accuracy or fidelity of the SR estimate compared to the ground-truth measured by a FR image/video quality measure. On the other hand, Blau and Michaeli, 2018 defines perceptual quality as the visual quality of the SR estimate, regardless of its fidelity to the ground-truth, i.e., it is the extent to which the estimate looks like a natural image. According to this definition, perceptual quality needs to be measured by a NR image/video quality measure.

Perception-distortion trade-off theory (Blau and Michaeli, 2018) claims that image restoration and SR algorithms cannot be simultaneously very accurate (high fidelity) and produce images that fool observers to believe they are real (high perceptual quality), no matter what distortion measure is used to quantify fidelity. This trade-off implies that optimizing a distortion measure alone cannot lead to estimates that cannot be distinguished from natural looking images.

Specifically, Blau and Michaeli, 2018 study the optimal probability for correctly discriminating the outputs of an image restoration/SR algorithm from real images. They show that as the mean distortion decreases, this probability must increase indicating worse perceptual quality, and that this result holds true for any distortion measure, and is not only a problem associated with the PSNR or SSIM criteria.

Accordingly, we can classify image restoration/SR applications as information-oriented and aesthetics-oriented applications. In applications that require extraction of information from degraded images, reconstruction accuracy is of key importance (e.g. license-plate/handwritten character recognition and medical imaging), where we only care for the accuracy (fidelity) of the information that can be gathered from images, we can optimize a distortion criteria only for the loss function. In others, where perceptual quality may be more important, e.g., computer games, we can go for optimization of a combination of fidelity and perceptual quality criteria as the loss function. Generative-adversarial networks (GANs) provide a principled way to approach the perception-distortion trade-off in image restoration/SR problems. We discuss methods for perceptual optimization of image and video restoration/SR in Sections 4.4 and 5.4, respectively

In conclusion, the optimization (loss function) and evaluation criteria for different image restoration/SR applications should differ. In general, it is a good idea to evaluate image restoration and SR algorithms/results by a pair of FR and NR metrics. Hence, the result can be placed on the perception-distortion plane, for reliable assessment of both distortion and perceptual quality, which makes comparison of algorithms/results more meaningful and reliable.

4

Deep Image Restoration and Super-resolution

This chapter addresses image restoration and SR using deep supervised and unsupervised learning. Supervised training refers to optimizing a single SISR model given a training set consisting of HR and LR image pairs, where LR images are generated from the HR images assuming a given blur kernel. This standard SISR problem setting provides the best results, both in terms of PSNR and perceptual criteria, when the blur kernel is known. However, it does not generalize well to the blind SR problem setting, where the blur kernel is unknown, since image restoration and SR problems are highly sensitive to errors in the blur kernel. In the real-world SR setting, the blur kernel is unknown and there is no ground-truth HR image paired with LR images; hence, supervised training is not applicable. An overview of the learned SISR problem settings covered in this chapter is shown in Table 4.1.

Table 4.1: Different learned SISR problem settings.

Setting	Blur Kernel	Training	Section(s)
Standard SR	Known	Supervised	4.1, 4.2
Blind SR	Unknown	Supervised	4.3
Real-world SR	Unknown	Unsupervised	4.4

We start with a brief history of recent advances in ConvNet architectures for image restoration and SR in Section 4.1. Self-ONNs and visual transformers for image restoration and SR are discussed in Section 4.2 and 4.3, respectively. These models have been optimized to minimize a distortion measure. Perceptual optimization of SISR models is discussed in Section 4.4. The problem of model overfit, i.e., overfitting image prior and/or the blur kernel, in supervised training of SISR models and solutions to alleviate model overfit are introduced in Section 4.5. Section 4.6 reviews unsupervised training strategies in the real-world SR setting. Some solutions to this setting are based on an external training set of unpaired LR and HR images, while some others require no training set but need to be optimized for each image independently.

4.1 A Brief History of ConvNets for Image Restoration/SR

Historically, the first work using Convnets for an image restoration task was in the context of natural image denoising (Jain and Seung, 2008). The network consisted of 4 hidden layers with 24 feature channels in each layer. Each feature map was connected to 8 randomly chosen channels in the previous layer and all convolutions were 5×5. The authors employed a layer-by-layer training procedure based on stochastic gradient descent to learn a nonlinear mapping to predict the restored image.

4.1.1 Early SR Network Architectures

The first end-to-end learned SR model, called SRCNN, was published in ECCV (Dong *et al.*, 2014) and later in a journal (Dong *et al.*, 2016). SRCNN network architecture was inspired by patch-based sparse coding approaches and contained only 3 convolutional layers. The first layer performs 9×9 convolutions for patch-based computation of n_1 feature channels, which emulates projecting input image patches onto an LR dictionary of size n_1. The second layer implements 1×1 convolutions for cross-channel pooling and a nonlinearity to map n_1 feature channels to n_2 channels. 1×1 convolutions were preferred in order not to increase the receptive field and preserve the patch-based nature of the end-to-end mapping from LR to HR space. The last layer performs

5×5 convolutions to reconstruct an HR image. The input RGB LR image was mapped into YCbCr components and only the luminance (Y) component was fed into the SRCNN model after first upscaling it using bicubic interpolation. The network was trained based on MSE loss and the results were evaluated by the PSNR metric. Experimental results show that even this simple learned model was sufficient to outperform all state-of-the-art traditional model-based SR methods at the time.

Another important early work is the very deep SR (VDSR) network (Kim *et al.*, 2016), which proposed residual learning for the first time. Residual learning (different from ResNet) proposes to learn the residual (difference) between the HR image and interpolated LR image, which is easier to learn than directly learning the HR image.

4.1.2 When to Upsample?

An important question in SR networks is when to upsample. There exist multiple options: *Pre-upsampling* refers to upsampling the LR image by a traditional interpolation filter before it is input to the SR network, while *post-upsampling* refers to passing the LR image through the network, whose final layer is a subpixel convolutional layer that generates the output HR image. Alternatively, one can place the upsampling layer anywhere before the final layer or perform *progressive upsampling* by placing multiple upsampling layers with intermediate convolutional layers when the scale factor S can be factored into integer factors.

Early approaches such as SRCNN (Dong *et al.*, 2016) and VDSR (Kim *et al.*, 2016) employed pre-upsampling, which is not the best option because it relies on the result of bicubic upsampling, as well as its high computational complexity due to passing the upsampled image through the network. ESPCN (Shi *et al.*, 2016) was the first post-upsampling approach, which introduced the sub-pixel convolution layer (also called the pixel shuffler layer) instead of using deconvolution layer for upsampling. In ESPCN, the sub-pixel convolution layer is used to reconstruct a $S \cdot H \times S \cdot W \times C$ HR image from a $H \times W \times S^2 \cdot C$ LR tensor, where H, W, C and S denote the height, width, output image channels and scale factor, respectively, by shuffling the pixels of the tensor to convert tensor channels into spatial dimensions of HR image. In modern

deep networks, the pixel shuffler layer is typically used to rearrange a $H \times W \times S^2 \cdot C$ feature tensor into a $S \cdot H \times S \cdot W \times C$ feature tensor, while the output HR image is formed by a separate convolution layer.

An example of progressive upsampling approach is the Laplacian pyramid networks (LapSRN) (Lai *et al.*, 2017), which progressively reconstructs sub-band residuals of an HR image, where there is a separate ground truth HR image and a corresponding loss function at each pyramid level. While LapSRN consists of a set of cascaded sub-networks, the network is trained in an end-to-end fashion (i.e., without stage-wise optimization) using a robust Charbonnier loss function.

A somewhat different approach inspired by the traditional iterative back-projection method is to design an end-to-end trainable network architecture with *iterative up- and down-sampling layers*. Deep back projection networks (DBPN) (Haris *et al.*, 2021) propose use of such multiple iterative up- and down-sampling layers with a feedback mechanism that allows the model to have self-correcting property, rather than learning a one-step non-linear mapping of input-to-target space.

4.1.3 SISR Architectures based on ResNet

The SR residual network (SRResNet) and its perceptually optimized version, SR generative adversarial network (SRGAN), are proposed by (Ledig *et al.*, 2017). Unlike the early SR networks, SRResNet employs residual blocks (see Section 2.1.2) that enable training deeper networks by mitigating the vanishing gradients problem. SRResNet consists of 16 residual blocks with 64 feature channels followed by pixelshuffler upsampling layer(s), which are optimized with respect to the MSE loss.

One of the best performing SR networks is the Enhanced Deep Super-Resolution Network (EDSR) (Lim *et al.*, 2017), which improves the performance of SRResNet (Ledig *et al.*, 2017) by removing batch normalization layers. The block diagram of residual blocks used in EDSR is depicted in Figure 4.1. It is experimentally shown that removing batch normalization layers improves SR results and reduces the computational cost of training. The EDSR architecture comes in two flavors: Baseline EDSR consists of 16 residual blocks with 64 feature channels, whereas the EDSR+ contains 32 residual blocks with 256 feature channels.

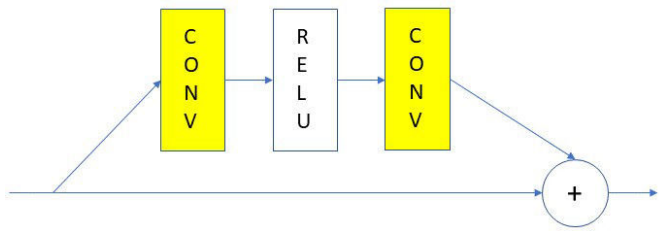


Figure 4.1: Illustration of a residual block in EDSR (Lim *et al.*, 2017).

EDSR treats all feature channels equally; hence, does not have the ability to selectively learn high-frequency details. The deep Residual Channel Attention Networks (RCAN) (Zhang *et al.*, 2018e) propose a residual in residual (RIR) structure with short and long skip connections and a channel attention (CA) mechanism. CA layers adaptively rescale each channel-wise feature to selectively learn high-frequency details while passing low frequency features unmodified. The RIR structure enables training very deep networks (e.g., over 400 layers), where the residual group (RG) serves as the basic module and long skip connection allows residual learning in a coarse level. Each RG module employs several simplified residual blocks with short skip connection. The long and short skip connections as well as the short-cuts in residual blocks ease the flow of information during training.

4.1.4 SISR Architectures based on DenseNet

Different from the skip connections in residual blocks, dense skip connections concatenate input features with the output of the layer rather than summing them. This provides the additional benefit of a multi-resolution feature map as feature channels from different layers have different receptive fields. SRDenseNet (Tong *et al.*, 2017) uses dense blocks (see Section 2.1.3), which contain channel-wise concatenation with short and long connections. Concatenating the outputs of several layers results in accumulation of a large number of feature channels, which significantly increases the model size and memory requirement. Thus, the number of feature channels is typically reduced to 256 or less using a convolution layer with 1×1 kernel, called a bottleneck layer, in order to keep the model compact and improve computational efficiency.

Residual Dense Network (RDN) for image SR (Zhang *et al.*, 2018f) proposes the residual dense block (RDB) with local feature fusion (LFF) as the building module. RDBs combine the benefits of residual blocks and dense blocks (see Section 2.1.3). The output of a RDB has direct connection to each layer of the next RDB, resulting in a contiguous memory mechanism. In addition to LFF within each RDB, the outputs of successive RDBs are also concatenated to conduct global feature fusion to exploit hierarchical global features in a holistic manner.

In order to further improve the performance of image SR networks, the residual in residual dense block (RRDB) (Wang *et al.*, 2018b), where residual learning is employed in different levels, (see Section 2.1.3) has been proposed. A similar network structure that applies multi-level residual learning has also been proposed in (Zhang *et al.*, 2018c). However, RRDB network differs from the one in (Zhang *et al.*, 2018c) as Wang *et al.*, 2018b employ dense skip connections to improve the learning capacity. They also exploit residual scaling, i.e., scaling the residuals down by multiplying them with a constant between 0 and 1 before adding to the main path for more stable training.

Finally, DRCA (Jang and Park, 2019) proposes a different combination of ResNet, DenseNet, and channel attention, where dense skip connections are between residual groups rather than convolution layers.

4.1.5 Comparison of SR Network Architectures

The network architectures reviewed in this Section have different block structures, type of skip connections between convolutional layers and blocks, numbers of layers, and trainable parameters. Each model has its unique basic block structure, e.g., EDSR (Lim *et al.*, 2017) has residual blocks, RCAN (Zhang *et al.*, 2018e) has residual-in-residual blocks within residual groups, and RDN (Zhang *et al.*, 2018f) is composed of residual dense blocks. We provide a comparison network architectures in Table 4.2, where we provide the total number of layers and trainable parameters to compare the computational complexity of different architectures, because counting the total number of blocks would not be fair since residual blocks in EDSR contain 2 convolutional layers, whereas RRDB blocks contain 5 convolutional layers.

Table 4.2: Total number of convolutional layers and trainable parameters

Model	Connection between layers	Connection between blocks	No. of Layers	No. of Parameters
EDSR	residual	residual	36	1.5 M
EDSR+	residual	residual	68	43 M
RDN	dense	residual/dense	150	19 M
RCAN	residual	residual	335	6.6 M
RRDB	dense	residual	351	16 M
DRCA	residual	dense	360	14.2 M

In terms of PSNR performance, RCAN, RRDB, and DRCA provide better results compared to that of EDSR+, although EDSR+ has more parameters. Hence, the performance of image SR networks cannot be predicted by the number of trainable parameters only, and the architecture of the network including the depth and width of the network and the type and number of skip connections matters.

4.1.6 Supervised Training of SR Models

These networks are trained to minimize the average per-pixel distortion between estimated HR images and the corresponding ground truth images using loss functions such as l_2, l_1, Charbonnier loss, or Huber loss given synthetically generated LR and HR training image pairs. The pixel-wise l_2 loss is also known as the mean-squared-error (MSE). The Charbonnier loss given by

$$l_C(x, \hat{x}) = \sqrt{(x - \hat{x})^2 + \epsilon^2} \tag{4.1}$$

where x and $\hat{x}$ are the ground-truth and estimated HR images and ϵ is a positive constant, is an approximation to the l1 loss that is differentiable about 0. Huber loss, given by

$$l_H(x, \hat{x}) = \begin{cases} \frac{1}{2}(x - \hat{x})^2 & \text{if } |x - \hat{x}| \leq \delta \\ \delta|x - \hat{x}| - \frac{1}{2}\delta^2 & \text{if } |x - \hat{x}| > \delta \end{cases} \tag{4.2}$$

is a combination of l_2 and l_1 losses that is also differentiable about 0.

Minimization of l_2 and l_1 loss results in the arithmetic mean-unbiased estimator and median-unbiased estimator, respectively. The l_2 loss has

the disadvantage that it is not robust against outliers when the distribution is heavy tailed. Huber loss combines the sensitivity of the mean-unbiased, minimum-variance arithmetic mean estimator and the robustness of the median-unbiased estimator.

The model accuracy is also affected by the patch size of training samples, mini-batch size, the optimizer, and learning rate schedule.

4.2 Self-Organizing Residual Networks for Image Restoration/SR

The operational neural networks (ONNs) and their "self-organized" variants (Self-ONN) that can approximate any non-linearity via Taylor series have been introduced in Section 2.2 as an alternative to using ConvNets with RELU nonlinearity. We also discussed replacing standard convolutional layers in residual blocks with self-organized layers to form self-organized residual (SOR) blocks in Section 2.2. In this section, we show that using SOR blocks in feedforward network architectures results in better image restoration and SR performance compared to using standard residual blocks in similar architectures with the same number of parameters.

Keleş *et al.*, 2021a explore network architectures with SOR blocks for image restoration and SR and compare their performance with the popular EDSR residual ConvNet architecture. Networks composed of only SOR blocks and a hybrid architecture shown in Figure 4.2 have been proposed. They show that a network composed of only 8 SOR blocks outperform the baseline EDSR network, while a hybrid network with 12 residual and 4 SOR blocks outperform both by an average margin of 0.2 dB. More importantly, they show that networks with SOR blocks avoid some visual artifacts that are seen in the EDSR results.

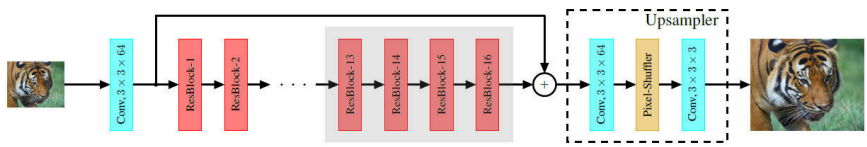


Figure 4.2: A hybrid SR network with residual and SOR blocks (Keleş *et al.*, 2021a). In EDSR, residual blocks in the gray shaded box are replaced by SOR blocks. Regular convolutional layers in the Upsampler are also replaced with self-organizing layers.

4.3 Transformer Networks for Image Restoration and SR

The application of self-attention or vision transformer architectures to image restoration and SR is a very recent research topic. In this section, we review the very few works available in the literature at present.

An approach that is different from the fidelity optimized learned SISR methods discussed so far is the neural texture transfer technique, which aims to integrate similar textures in a different HR reference (Ref) image into a given LR target image. Texture transfer is also different from self-example-based SR approach (Liu *et al.*, 2007) in that it uses an external Ref image rather than self-similarity. It has proven to be promising in recovering HR details when a Ref image with similar content as that of the LR input is available. Yang *et al.*, 2020 propose a new Texture Transformer Network for Image SR (TTSR), in which the LR and Ref images are formulated as queries and keys in a transformer, respectively. TTSR consists of four modules optimized for image generation tasks, including a learnable texture extractor, a relevance embedding module, a hard-attention module for texture transfer, and a soft attention module for texture synthesis. The proposed design encourages joint feature learning across LR and Ref images, in which deep feature correspondences can be discovered by attention, and thus accurate texture features can be transferred. TTSR can be stacked in a cross-scale way to enable texture recovery from different resolution levels, e.g., from 1× to 4× magnification.

Inspired by the success of pre-training transformer-based models on a large text corpus and then fine-tuning them on task-specific smaller datasets in natural language processing (NLP), Chen *et al.*, 2021 propose a pre-trained transformer model for image processing called Image Processing Transformer (IPT). As IPT needs to be compatible with different image processing tasks, such as SR, denoising, and deraining, the network is composed of multiple pairs of head and tail corresponding to different tasks and a single shared body. The overall architecture consists of four components: heads for extracting features from corrupted input images, an encoder-decoder transformer for recovering the missing information in the input data, and tails for mapping the features into restored images. The IPT model with multi-heads and multi-tails is

trained on a large number of synthetically generated corrupted image pairs from ImageNet benchmark. In addition, contrastive learning is introduced for well adapting to different image processing tasks. The pre-trained model can therefore be efficiently employed on desired task after fine tuning. It has been shown that pre-trained IPT can outperform the state-of-the-art methods on various image processing benchmarks.

SwinIR (Liang *et al.*, 2021) combines the benefits of both ConvNet and Swin Transformers. It consists of three modules: shallow feature extraction, deep feature extraction, and image reconstruction. The shallow feature extraction module uses a convolution layer to extract features, which are directly transmitted to the reconstruction module to preserve low-frequency information. The deep feature extraction module is composed of residual Swin Transformer blocks (RSTB), which utilize several Swin Transformer layers for local attention and cross-window interaction. In addition, they add a convolution layer at the end of the block for feature enhancement and use a residual connection to provide a shortcut for feature aggregation. Finally, both shallow and deep features are fused in the reconstruction module. The implementation of reconstruction module uses the sub-pixel convolution layer to upsample the features. Experimental results demonstrate that SwinIR outperforms state-of-the-art methods on different tasks by up to 0.14-0.45dB, while the total number of parameters is reduced by up to 67%.

Uformer (Wang *et al.*, 2022) is an effective and efficient Transformer-based architecture for image restoration, which builds a hierarchical encoder-decoder network using the Transformer block. There are two core designs in Uformer: First, the authors introduce a locally-enhanced window (LeWin) Transformer block, which performs nonoverlapping window-based self-attention instead of global self-attention. This step significantly reduces the computational complexity on high resolution feature maps while capturing local context. Second, they propose a learnable multi-scale restoration modulator in the form of a multi-scale spatial bias to adjust features in multiple layers of the Uformer decoder. The modulator provides superior capability for restoring details for various image restoration tasks while introducing marginal extra parameters and computational cost. Uformer shows high capability for capturing both local and global dependencies for image restoration.

4.4 Perceptual Image Restoration and SR

Early learned SISR methods employed the same l_2 or l_1 loss functions as the classical model-based SR methods for supervised training of ConvNets as discussed in Section 4.1.6. However, it is well-known that minimizing l_2 loss (mean-square error) typically results in unnatural-looking blurry textures since the minimum mean squared error estimator is the conditional mean, and the arithmetic mean of natural images is not necessarily a natural image. Different from classical model-based methods, deep-learning based SR allows for optimization using any differentiable loss, such as perceptual losses.

Perceptual SR methods can be classified as: i) training feedforward models using full-reference perceptually motivated loss functions, discussed in Section 4.4.1, and ii) training generative SR models. Generative models aim to capture the probability distribution of natural images given sample natural images (assuming samples are drawn from the same distribution), and allow drawing new samples from the estimated distribution. Various approaches have been proposed for generative modeling: Deep learning based approaches include generative adversarial networks (GANs) (Goodfellow *et al.*, 2014), variational auto-encoders (VAEs) (Kingma and Welling, 2014), and normalizing flow-based methods (Rippel and Adams, 2013). GANs and VAEs have demonstrated impressive results on learning distribution of natural images. However, neither allows for exact evaluation of the density at a new sample (i.e., the likelihood). Furthermore, training can be challenging due to mode collapse, vanishing gradients and other instabilities (Salimans *et al.*, 2016). Normalizing flow-based methods (Papamakarios *et al.*, 2021; Kobyzev *et al.*, 2021) offer more stable training and allow for both sampling and exact and efficient probability density evaluation.

We review generative adversarial SR models in Section 4.4.2. The feasible solution in image restoration and SR is discussed in Section 4.4.3. Normalizing flow-based SR models that learn the conditional distribution of HR images given LR inputs using the negative log-likelihood loss is discussed in Section 4.4.4. Perceptual SR methods provide sharper details at the expense of a decrease in PSNR as predicted by perception-distortion trade-off theory (see Section 3.5).

4.4.1 Training Regressive SR Models using Perceptual Loss

Images with high perceptual quality can be reconstructed by defining and optimizing full-reference perceptually motivated loss functions based on either hand-crafted low-level image features, such as MS-SSIM (see Section 3.1.3), or features extracted by pre-trained networks, such as LPIPS (see Section 3.1.5). Optimization of SR models with respect to a weighted combination of l_1 loss and MS-SSIM loss has been heavily adopted in practice due to good visual quality of resulting HR images.

Perhaps the earliest work that used a learned feature reconstruction loss for the SR task is (Johnson *et al.*, 2016). Rather than forcing pixels of the reconstructed HR image $\hat{y} = f_W(x)$ to exactly match the pixels of the target image y, they enforce the reconstructed and target images to have similar features as computed by a pre-trained loss network ϕ. If we let $\phi_j(x)$ be the feature tensor with the shape $C_j \times H_j \times Wj$ at the output of the jth layer of the ConvNet $\phi(x)$ processing the input image x, then the feature reconstruction loss is the normalized squared Euclidean distance between features of the reconstructed image $\hat{y}$ and target image y given by

$$\ell^j_\phi(y, \hat{y}) = \frac{1}{C_j \cdot H_j \cdot Wj} ||\phi_j(y) - \phi_j(\hat{y})||^2 \tag{4.3}$$

Finding an image $\hat{y}$ that minimizes the feature reconstruction loss (4.3) for early layers j tends to yield an HR image that is perceptually similar to the target image y, but not necessarily match it pixel-by-pixel. A more recent example of a popular feature loss for perceptual SR is LPIPS (Zhang *et al.*, 2018d).

4.4.2 Generative Adversarial SR Models

The GAN framework was first exploited for the SISR problem in the seminal work SRGAN (Ledig *et al.*, 2017), where the generator G is a SR network that predicts HR images given LR inputs, and the discriminator D estimates the probability that its input is a predicted vs. ground-truth HR image. SRGAN aims to strike a balance between the fidelity and photo-realism of the reconstructed HR images by training the generator G using a loss, which is a weighted sum of a fidelity loss

and an adversarial loss (as a function of probabilities estimated by D). The adversarial loss encourages the predicted HR images to be closer to the manifold of natural images in the solution space. The weighting parameter λ determines the trade-off between the fidelity and photo-realism. The photo-realism provided by hallucinating realistic textures comes at the expense of lower fidelity, i.e., lower PSNR values. The perceptual quality of reconstructed HR images was measured by the mean opinion score (MOS) of human evaluators in the subjective tests. It was observed that the addition of an adversarial loss results in improvements in MOS at the expense of significantly lower PSNR.

Later Blau and Michaeli, 2018 have shown that distortion (fidelity) and perceptual quality are indeed at odds with each other. Specifically, they show that as the mean distortion decreases, the probability of correctly discriminating the outputs of an image reconstruction algorithm from real images must increase, which implies worse perceptual quality. Moreover, this result is not only related to the PSNR or SSIM criteria, but holds true for any distortion (full-reference) measure. Hence, they define a perception-distortion (PD) bound and demonstrate that GANs provide a principled way to approach the PD bound. Following their work, the first Challenge on Perceptual Image Restoration and Manipulation (PIRM) has been announced (Blau *et al.*, 2018). An important problem was to identify a suitable objective measure to evaluate perceptual image quality since conducting subjective tests at a large scale is not practical. The PIRM 2018 SR Challenge employed the perceptual index (PI), a no-reference measure, defined by (3.5), which is shown to be highly correlated with human ratings, to assess perceptual quality.

The winner of the PIRM Challenge was a generative adversarial model, called the Enhanced SRGAN (ESRGAN) (Wang *et al.*, 2018b). The authors observe that details hallucinated by SRGAN are often accompanied with unpleasant artifacts. Hence, they revise three key components of the SRGAN model: i) They improve the architecture of the generator by introducing the Residual-in-Residual Dense Block (RDDB), which has higher learning capacity and is easier to train; ii) They employ the relativistic discriminator originally proposed by (Jolicoeur-Martineau, 2019), which estimates the probability that the given real data is more realistic than fake data, on average, and show that this

helps the generator recover more realistic texture details; iii) They propose an improved perceptual loss, using the VGG features before activation instead of after activation, which provides sharper edges and more visually pleasing results. Benefiting from these improvements, the proposed ESRGAN achieves consistently better visual quality with more realistic and natural textures than SRGAN.

A limitation of adversarial SR models is that they cannot be trained using the evaluation criterion directly as loss, since NIQE and Ma measures that make up the PI measure are not differentiable. To address this problem, Zhang *et al.*, 2019 propose SRGAN with Ranker (RankSRGAN), which first trains a Ranker that learns to rank HR images according to their perceptual scores and then introduces a rank-content loss to optimize the generator for perceptual quality. Experimental results show that RankSRGAN can combine the strengths of different SR models to achieve the state-of-the-art performance in perceptual SR.

4.4.3 The Feasible Solution in Image Restoration/SR

Image restoration and SR are ill-posed inverse problems with potentially infinitely many feasible solutions. The set of feasible solutions is defined as those satisfying all known constraints; namely, all images that are consistent with the given LR image under a known degradation model. At higher upscaling factors, the problem becomes even more difficult, since the set of feasible solutions becomes huge. The problem is further complicated by the fact that there is no single definition of the "best" solution. As discussed in Chapter 3, there are multiple optimization and evaluation criteria some of which are at odds with each other.

Yet, this is not a new observation. The observation that signal restoration problem has multiple feasible solutions and that there is no single definition of the best solution was first made by Trussell and Civanlar, 1984, who proposed the projection onto convex sets (POCS) framework to find a feasible solution. Later, Irani and Peleg, 1991 proposed backprojection method, which is similar to POCS, for image super-resolution from multiple images, and Patti *et al.*, 1997 applied the POCS framework to video SR accounting also for motion blur. Note that none of these classic approaches use fidelity to the original as criterion.

Recently, Bahat and Michaeli, 2020 presented a consistency enforcing module (CEM) that introduced the feasible solution constraint into learned SR frameworks. CEM does not have any learnable parameters and can wrap any SR network to enforce its output matches the LR input when down-sampled. Their reconstruction network adopts an adversarial loss to encourage perceptual plausibility of the output by penalizing deviations from the statistics of natural images, while the CEM enforces consistency constraint. They do not employ a fidelity loss. They also present a GUI to enable users to interactively explore the manifold of feasible SR solutions, by inputting a control signal to the SR network.

4.4.4 Normalizing Flow-Based Generative SR Models

Normalizing flows (NF) provide a general methodology for constructing arbitrary probability distributions over continuous random variables. Let $\mathbf{x}$ be a D-dimensional real vector, and suppose we would like to define a joint distribution over $\mathbf{x}$. The main idea is to express $\mathbf{x}$ as a transformation T of a real vector $\mathbf{z}_0$ sampled from a simple base distribution $p_{Z_0}(\mathbf{z}_0)$, i.e., $\mathbf{x} = T(\mathbf{z}_0)$. Then, the distribution of $\mathbf{x}$ can be expressed by the change of variables formula. Flow methods construct arbitrarily complex densities by composing several simple transformations, i.e., $T = T_K \circ \cdots \circ T_1$ and applying the change of variables formula successively. The defining property of flow-based models is that the transformation T must be invertible and both T and T^{-1} must be differentiable, which is guaranteed if all transformations $T_k,\ k = 1, \ldots, K$ are invertible and differentiable. The path traversed by the random variables $\mathbf{z}_k = T_k(\mathbf{z}_{k-1}),\ k = 1, \ldots, K$, where $\mathbf{x} = \mathbf{z}_K$ is called the flow and the inverse path from $\mathbf{x}$ to $\mathbf{z}_0$ is called a normalizing flow assuming the base distribution $p_{Z_0}(\mathbf{z}_0)$ is a joint Gaussian (normal distribution). The transformation T and the base distribution $p_{Z_0}(\mathbf{z}_0)$ have parameters, denoted by ϕ and ψ, respectively, which induces a family of distributions over $\mathbf{x}$ parameterized by ϕ and ψ. More details can be found in the excellent review articles (Kobyzev *et al.*, 2021; Papamakarios *et al.*, 2021).

Flow-based generative models using deep learning were first described in Rippel and Adams, 2013, then extended in Rezende and

Mohamed, 2015; Dinh *et al.*, 2017, and Kingma and Dhariwal, 2018. Rippel and Adams, 2013 proposed using the change of variables formula and modeling the mapping through an invertible neural network. However, naive application of the change of variable formula produces models which are computationally expensive and poorly conditioned. Rezende and Mohamed, 2015 propose the specification of approximate posterior in variational inference by using normalizing flows and consider normalizing flows for which the Jacobian of the transformation can be computed in linear time. The basic idea in RealNVP (Dinh *et al.*, 2017) is to choose transformations whose Jacobian matrix is triangular by proposing invertible affine coupling layers. Glow (Kingma and Dhariwal, 2018) proposed 1x1 convolutions as a generalization of fixed partition permutation used by (Dinh *et al.*, 2017) and demonstrated the first likelihood-based model in the literature that can efficiently synthesize high-resolution (HR) natural images.

NFs can synthesize HR images efficiently by sampling from a learned distribution, but in the SR task, the learned distribution should be conditioned on a given LR image. Winkler *et al.*, 2019 studied conditional normalizing flows (CNFs), a class of NFs where the mapping from the base random variable $\mathbf{z}_0$ to output HR space $\mathbf{x}$ is conditioned on an LR input $\mathbf{y}$, to model the conditional density $p_{X|Y}(\mathbf{x}|\mathbf{y})$. This is achieved by conditioning the base distribution and transformation parameters on the LR input $\mathbf{y}$ by using conditional affine coupling layers, which concatenates the encoded conditioning variable in the affine coupling layers. Like NFs, CNFs are efficient in sampling and inference, and they are trained with an exact log-likelihood objective.

SRFlow (Lugmayr *et al.*, 2020b) is a CNF based SR method that also extends the Glow architecture (Kingma and Dhariwal, 2018) to learn an exact mapping from HR image manifold to a latent space by modeling the conditional distribution of the HR image given the LR input. In contrast to the conditional affine coupling layers in Winkler *et al.*, 2019, SRFlow proposes an affine injector layer, which directly affects all channels and spatial locations in the activation map by predicting an element-wise scaling and bias using the conditional encoding. A CNF-based SR model allows sampling multiple output images from a learned HR space given an LR image. This way it learns to predict diverse photo-

realistic high-resolution images, directly accounting for the ill-posed nature of the SR problem. The model is trained in a principled manner using a single negative log-likelihood loss. SRFlow outperforms state-of-the-art GAN-based approaches in terms of both PSNR and perceptual quality metrics, while allowing for diversity through the exploration of the space of super-resolved solutions.

SRFlow-DA (Jo *et al.*, 2021) extends SRFlow by stacking more convolutional layers in the affine couplings to enlarge the receptive field and have more expressive power. Compared to SRFlow, SRFlow-DA achieves better or comparable PSNR and LPIPS for $\times 4$ and $\times 8$ SR tasks, while having a reduced number of parameters. As a result, the model can be trained on a GPU with 11GB memory.

4.5 Dealing with Model Overfitting in Supervised Training

The SISR methods discussed so far rely on supervised training of a single SR model for all images in the test set given a training set consisting of HR and LR image pairs. A limitation of supervised training is that the performance of the SR model deteriorates when the characteristics of images in the test set deviate from those in the training set. There are two main sources of variability between the training set and test set: variation of blur kernel and variation of image prior. Methods to deal with overfitting the image prior and the degradation model are discussed in Sections 4.5.1 and 4.5.2, respectively. We note that while both variations result in deterioration of SR performance, using an incorrect blur kernel affects SR performance far more than any choice of an image prior.

4.5.1 Overfitting Image Prior: Multi-Model SR

Kirmemis and Tekalp, 2018 observe that the performance of even the best performing image restoration/SR model, in the ideal case there is no misfit of blur kernel, varies noticeably from image to image over a test set depending on how well the patterns in each test image match those in the training set. They report that the standard deviation of the MSE or PSNR values over a test set is almost equal to their mean, where individual image PSNR values can vary up to 5 dB within a test

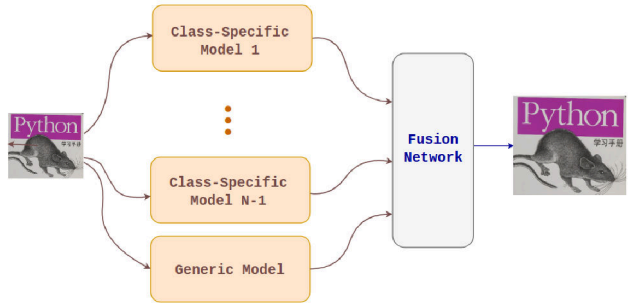


Figure 4.3: The multi-model SR (MMSR) architecture consisting of a number of class-specific SR models and a fusion network (Korkmaz, 2021).

set. This indicates that a single SR model cannot generalize equally well for all images within a test set even when the blur kernel is known. This observation and the success of early class-based image hallucination methods (Baker and Kanade, 2000; Liu *et al.*, 2007) motivate us to explore SR models that exploit class-specific image priors.

Different network architectures can be conceived to benefit from class-specific image priors in image processing tasks. One approach would be to cluster or classify training images into a predetermined number of homogeneous classes, e.g., text, texture, face, etc., and train a different SR network for each class. Then, during inference, a pre-classifier network directs each LR image to the corresponding SR model. However, this approach would run into a problem in the case of non-homogeneous images, i.e., images containing a combination of text, texture, face, etc. In order to handle non-homogeneous images, a multi-model SR (MMSR) architecture consisting of a number of class-specific SR models and a post fusion network depicted in Figure 4.3 was proposed (Korkmaz, 2021).

The MMSR architecture employs a bank of N standard SR models, e.g., EDSR (Lim *et al.*, 2017) or RCAN (Zhang *et al.*, 2018e), as class-specific SR models. The fusion model merges the outputs of these class-specific SR models to benefit from the best aspects of each one of them. Experimental results show that a simple fusion network with three residual blocks is sufficient to obtain very good results. The input to the fusion model is a stack of $3N$ images consisting of R, G and B

outputs of N SR models. The output of the fusion model is a single RGB image. The fusion model is trained by using HR and LR image patches similar to training the class-specific and generic SR models.

Experimental results indicate that the MMSR approach is superior to segmenting the input image into homogeneous regions using an image segmentation network, directing each region to the corresponding SR model, and finally combining the results of different SR models. More interestingly, experimental results also show that even in the case of homogeneous LR images, processing the input image with multiple models each tuned to image priors of a particular class and then fusing the results produced by these multiple models by a post-fusion network outperforms the performance of even the best class-specific model trained for the particular class of images.

4.5.2 Variation of the Blur Kernel: Blind vs. Non-blind SR

The supervised training paradigm given a set of LR-HR image pairs assumes that the LR image formation model (blur kernel) for images in the training set matches that of the LR images in the test set exactly. SR models obtained by supervised training provides the best state-of-the-art performance when the LR image formation model is known and the training set is generated using this known model.

On the other hand, in real SR applications, the LR image formation model is often unknown, and the performance of SR models trained based on an assumed bicubic degradation model is unsatisfactory when the unknown model deviates from the assumed bicubic blur kernel. This is because the blind SR problem, where the blur kernel is unknown, is doubly ill-posed. The standard SR problem with known blur kernel is ill-posed because the solution is not unique, i.e., multiple HR images can be generated from a single LR image. In the blind SR problem, not only multiple HR images can be generated from a single LR image, but also multiple LR images can be generated from the same HR image depending on the choice of the blur kernel. Hence, in blind SR, the true HR image may not even be in the set of all possible solutions that can be generated by an SR algorithm using a blur kernel that deviates from the actual one.

This subsection considers the blind SR setting, where the blur kernel is unknown, but paired training data is still assumed to be available, i.e., we are given LR-HR pairs without knowing the LR image formation model. Possible solution strategies to alleviate overfitting the assumed LR image formation model given paired training data include:
i) Estimate the blur kernel from given HR, LR training image pairs;
ii) Propose SR models that are robust to variations in the blur kernel without explicitly estimating LR image formation model; iii) Perform blind-SR by alternating between kernel estimation and SR image reconstruction; iv) Allow to input an image-specific (pre-estimated) blur kernel along with the LR input image at inference time.
These approaches are discussed in more detail in the following. The real-world SR setting, where the blur kernel is unknown and only unpaired data is available, is considered in Section 4.6.

NTIRE Blind SR Challenges

A lot of progress in this area has been achieved in the NTIRE SR challenges. Hence, we first provide a quick review of solutions proposed in the NTIRE blind/real-world SR challenges.

The first NTIRE Challenge in 2017 (Timofte *et al.*, 2017) had a track called "unknown downscaling," where only synthetically generated LR, HR image pairs were made available, but the blur filter was not disclosed. Since the number of provided image pairs is usually limited, the solution proposed by some teams was to first learn the HR to LR mapping using a simple network consisting of two residual blocks so that they can generate more training pairs consistent with the given ones by applying the learned mapping on extra HR images as a method for data augmentation. Some of these methods also boost performance using "enhanced prediction" or "ensembles" (Timofte *et al.*, 2016), which flips and rotates (in 90° steps) each input LR image to obtain 4 or 8 SR results that are aligned back through the inverse transformation and averaged to get the final result.

NTIRE 2018 (Timofte *et al.*, 2018) featured three synthetically generated paired HR, LR datasets downscaled by a factor of 4 from the DIV2K dataset (Agustsson and Timofte, 2017) with unknown

blur kernels in realistic mild, difficult and wild adverse conditions. In the realistic mild and difficult conditions, LR images are generated by emulating a camera acquisition pipeline, where the degradation was stronger in the latter. In the wild condition, the hidden blur kernel varied from image to image. Team UIUC-IFP obtained the best PSNR and SSIM for all 3 cases using WDSR (Yu *et al.*, 2019). WDSR is an extension of EDSR (Lim *et al.*, 2017), where the number of feature channels are increased before the non-linear activation in each residual block and then reduced again for summation with the shortcut branch of the identity mapping.

NTIRE 2019 Challenge on Real Image Super-Resolution (Cai *et al.*, 2019a) introduced the first paired HR, LR image dataset captured by real camera, called RealSR (Cai *et al.*, 2019b). Participants were asked to map LR images captured by a DSLR camera with a shorter focal length to corresponding HR images captured at a longer focal length. The best results were achieved by the UDSR, which is a U-Net (Ronneberger *et al.*, 2015) based architecture. They employ a three-stage cascaded training framework such that in each stage, they use the output of the previous stage as input and each stage has different ground-truths from coarse to fine. They utilized an adaptive multi-model ensemble method to improve the results. The second best result was obtained by Feng *et al.*, 2019. They apply the MixUp principle (Zhang *et al.*, 2018a) to train networks on interpolations of sample pairs, which encourages the model to support linear behavior in-between training samples. They also propose a data synthesis method with learned degradation to avoid model overfitting under very limited training samples and achieve satisfactory generalization performance. The proposed approach is independent of network architecture and task; hence, can be applied to other image restoration tasks. The results of this challenge reveal that SISR models trained on simulated data (i.e., bicubic downsampling) are hard to generalize to practical applications since the authentic degradations in real-world LR images are more complex.

Because the AIM 2019 and NTIRE 2020 Challenges on Real-World Image SR provided unpaired training data, they will be discussed in Section 4.6.

Single Model that Learns Multiple Degradations

Another approach that competed in the NTIRE 2019 Real SR Challenge is the SRMD network (Zhang *et al.*, 2018b), which is a single model that is trained to handle multiple known degradations. The SISR networks discussed so far compute an estimate the HR image $\hat{\mathbf{x}}$ by a nonlinear mapping $\mathcal{F}$ with parameters θ and input LR image $\mathbf{y}$, given by

$$\hat{\mathbf{x}} = \mathcal{F}_\theta(\mathbf{y}) \tag{4.4}$$

SRMD takes the known or estimated blur kernel $\mathbf{k}$ and noise level σ as input in addition to the LR image $\mathbf{y}$. Hence, it can be expressed by

$$\hat{\mathbf{x}} = \mathcal{F}_\theta(\mathbf{y}, \mathbf{k}, \sigma) \tag{4.5}$$

The LR image, blur kernel, and noise level are formatted into an input tensor of size $H \times W \times (C + b + 1)$, where H, W, and C are the height, width, and number of channels of the LR image, respectively, and b is the number of elements in the blur kernel $\mathbf{k}$. All elements of each channel representing the blur kernel are equal to one element of $\mathbf{k}$. The noise level is also represented by a channel with all elements equal to σ. In the supervised training phase, at each epoch, they randomly sample a pre-determined degradation space to select a blur kernel and a noise level in order to synthesize LR images from HR images. During testing, they either assume the blur kernel and noise level are known or estimate them for each test image to form the input tensor.

Unified dynamic convolutional network for variational degradations (UDVD) (Xu *et al.*, 2020) extends SRMD by introducing dynamic convolution kernels to propose a non-blind SISR network to accommodate two types of blur kernel variations, inter-image variations and spatial (intra-image) variations. While the standard convolution layer learns a kernel that minimize the error across all pixels, dynamic convolution generates per pixel (spatially-varying) kernels by a parameter-generating network (Brabandere *et al.*, 2016). Moreover, the standard convolution kernel is input-agnostic, i.e., fixed after training. In contrast, the proposed dynamic convolution adapts to different input even after training.

Deep SR with Known Blur Kernel via MAP Estimation Framework

A different approach to handle image-specific known blur kernels with a single network is based on integration of model-based methods and deep learning under a unified maximum a posteriori (MAP) estimation framework. The MAP inference problem can be formulated as

$$\hat{\mathbf{x}}(\lambda) = arg_{\mathbf{x}}\ min\ \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 + \lambda \mathbf{R}(\mathbf{x}) \tag{4.6}$$

where $\mathbf{H}$ and $\mathbf{R}$ are blur and regularization (image prior) operators. Half-quadratic splitting formulation introduces an auxilary variable $\mathbf{z}$

$$\hat{\mathbf{x}}(\lambda) = arg_{\mathbf{x}}\ min\ \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 + \lambda \mathbf{R}(\mathbf{z}) \tag{4.7}$$

such that $\mathbf{z} = \mathbf{x}$, to allow decoupling MAP inference into separate data fidelity and image prior subproblems, which can be solved by minimizing the cost function

$$\mathbf{L}_{\mu}(\mathbf{x}, \mathbf{z}) = \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 + \lambda \mathbf{R}(\mathbf{z}) + \frac{\mu}{2}||\mathbf{z} - \mathbf{x}||^2 \tag{4.8}$$

The solution can be obtained by the following two-step iterations:

$$\hat{\mathbf{x}}_k = arg_{\mathbf{x}}\ min\ \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 + \mu||\mathbf{x} - \mathbf{z}_{k-1}||^2 \tag{4.9}$$

$$\hat{\mathbf{z}}_k = arg_{\mathbf{z}}\ min\ \frac{\mu}{2}||\mathbf{z} - \mathbf{x}_k||^2 + \lambda \mathbf{R}(\mathbf{z}) \tag{4.10}$$

These two subproblems can be solved by neural modules, resulting in an end-to-end trainable, iterative network unfolding the MAP inference.

At a high-level, algorithm unfolding or unrolling maps an iterative algorithm to a corresponding deep network by representing each iteration of the algorithm with a subnetwork. these subnetworks are cascaded as many times as the number of iterations of the algorithm needs to be executed (Monga *et al.*, 2021). The parameter vector θ that is updated at each iteration of the algorithm is transferred into network parameters $\theta_{\mathbf{1}}$, $\theta_{\mathbf{2}}$, ... We learn the network parameters $\theta_{\mathbf{1}}$, $\theta_{\mathbf{2}}$, ... from training data sets through end-to-end supervised training. The resulting network could achieve better performance than the original iterative algorithm. In addition, the network naturally inherits interpretability from the iteration procedure.

Unfolding SR network (USRNet) (Zhang *et al.*, 2020b) alternates between a data consistency subnetwork and image prior subnetwork with K = 8 iterations. As the solutions of the subproblems require hyper-parameters α_k and β_k as input, respectively, a hyper-parameter module $\mathcal{H}$ is further introduced into USRNet. The USRNet inherits the flexibility of model-based methods to super-resolve blurry, noisy images for different scale factors via a single model, while maintaining the advantages of learning-based methods.

Blind SR with Image-Specific Kernel Adaptation

We discussed two different approaches to handle known blur kernels, namely, concatenating the blur kernel to the input LR image, and algorithm unfolding. When used in the context of blind SR, these methods employ a two-step procedure: first, they estimate the unknown blur kernel and noise level by some other approach; then, they use the estimated blur kernel as input to the non-blind SR algorithm. However, blur kernel estimation is a difficult ill-posed problem in itself, i.e., it is sensitive to observation noise and does not have a unique solution. The two-step solution involves two independently trained models, which do not cooperate with each other. As a result, an inaccurate blur kernel estimate from the first step directly affects the quality of the SR estimate in the second step.

Gu *et al.*, 2019 observed that artifacts caused by blur kernel mismatch are asymmetric, i.e., if the estimated kernel is smoother than the real one, the SR image is oversmoothed/blurry; on the other hand, if the estimated kernel is sharper than the real one, the SR image is over-sharpened and has ringing artifacts. They propose an Iterative Kernel Correction (IKC) (Gu *et al.*, 2019) method for blind SR to gradually correct the estimated blur kernel during inference. The proposed iterative method consists of a prediction step, which provides the initial blur kernel estimate, an SR model, which takes the blur kernel as input, and a correction step, which estimates the kernel error using intermediate SR results. They also proposed an improved SR model for multiple blur kernels, called SFTMD, by using spatial feature transform (SFT) layers and an advanced ConvNet structure to outperform SRMD.

Deep alternating network (DAN) (Luo *et al.*, 2020), further improves this framework. Specifically, they design two ConvNets, called Restorer and Estimator. The Restorer restores SR image based on the predicted kernel, and the Estimator estimates the blur kernel given the restored SR image. The method alternates between these two modules and unfolds this process to form an end-to-end trainable network. DAN provides superior results compared to IKC because the Estimator uses information from both LR and SR images, which makes the estimation of blur kernel easier; and the Restorer is trained with the kernel estimated by the Estimator, instead of the ground-truth kernel, which makes the Restorer more tolerant to kernel estimation errors.

4.6 Real-World SR by Deep Unsupervised Learning

SR models learned from synthetically generated paired HR, LR image datasets have significantly outperformed conventional methods when trained and tested with the same known degradation model. However, when it comes to real-world problems, they have very limited use because real LR images are degraded by blur and noise, which are unknown in the practical setting. Furthermore, in real-world SR, there is no ground-truth; hence there is no paired data available for training.

As a result, methods that can be trained by unpaired data sets, or without an external training set, or require no training are of interest. We start by discussing three non-blind SR methods: deep image prior that requires no training in Section 4.6.1, deep plug-and-play that relies on a pre-trained deep denoiser in Section 4.6.2, and deep internal learning that does not require an external training set in Section 4.6.3. We then discuss blind SR approaches that can be trained by unpaired external datasets in Section 4.6.4 and Section 4.6.5.

4.6.1 Deep Image Prior

Deep image prior (Ulyanov *et al.*, 2020) uses the inductive bias of an untrained neural network as an image prior and completely removes the regularization term in (4.6). Hence, the problem formulation becomes

$$\hat{\mathbf{x}}(\lambda) = arg_{\mathbf{x}}\ min\ \frac{1}{2}||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 \tag{4.11}$$

such that $\mathbf{x} = G(\mathbf{z}, \theta)$, where $G(\mathbf{z}, \theta)$ is a generative neural network with latent variables $\mathbf{z}$ and trainable parameters θ.

This problem can be solved by optimizing over the trainable parameters θ, while the latent variable $\mathbf{z}$ is initialized randomly and kept fixed. Although, we can also optimize over the latent variable, it has been reported that this does not help improve the performance significantly.

This approach does not use any training or pre-trained model. Only the observed corrupted image is used in the process of optimization. However, it does require solving a new optimization problem for each image for inference. We note that this is a non-blind method; i.e., it requires the blur kernel to be known or estimated by other means.

4.6.2 Plug-and-Play Image Restoration and SR

Deep plug-and-play method is an alternative non-blind solution to the MAP estimate we discussed in Section 4.5. More specifically, we showed that half quadratic splitting of the MAP solution yields the two-step iteration given by Equations (4.9) and (4.10).

Instead of unfolding the iterations for an end-to-end optimized solution, we make the observation that Equation (4.10) defines a denoising problem and replace that step with a denoiser ConvNet, which acts as an image prior for the image restoration/SR problem (Zhang *et al.*, 2017). Hence, the main idea of deep plug-and-play methods is that a pre-trained denoiser ConvNet can implicitly serve as the image prior for model-based methods to solve inverse problems.

This approach is called plug-and-play because we simply plug the pre-trained ConvNet into the iterations as the solution for Equation (4.10), while we typically provide a closed-form solution for Equation (4.9) given the blur kernel. Hence, deep plug-and-play methods are essentially iterative model-based methods, which suffer from a high computational load at inference and they involve manually selected hyper-parameters. A thorough analysis of this approach including parameter setting, intermediate results, and empirical convergence to better understand the working mechanism, is provided in (Zhang *et al.*, 2021).

4.6.3 Deep Internal Learning

Zero-shot SR (ZSSR) (Shocher *et al.*, 2018) is a non-blind unsupervised deep learning method, which takes advantage of self-similarity and internal statistics of images and do not rely on an external training set, given the blur kernel. Similar to deep image prior and plug-and-play methods, we need to train a different inference model for each test image. However, unlike those two methods, the model learns an image prior for each test image.

ZSSR generates an image-specific SR model, where a small ConvNet model, with 8 convolutional layers and 64 output channels, is trained on image patches obtained from the test image itself. The main idea of ZSSR is to exploit internal self-similarity of images, i.e., repetition of patches within an image. It is stated that no matter how large a dataset is used for training the internal and unique characteristics of an image can be only found in the image itself. ZSSR is a non-blind SISR method, which requires the blur kernel as input. The blur kernel is estimated by another method, such as KernelGAN (Bell-Kligler *et al.*, 2019). The main disadvantage of, zero-shot models is that they need to be trained for each test image individually, which is time consuming.

4.6.4 Blind Real-World SR

A promising solution to the real-world SR problem is a two-stage method, where in the first stage real-world LR images with unknown degradation are mapped to bicubic downsampled "look-alike" images, and in the second stage these intermediate domain images are super-resolved by a second network trained on synthetically generated bicubic sampled paired data. In this strategy, the performance of the first stage is crucial, where there are two possible approaches: i) in an actual real-world setting, there are no paired dataset; hence, the training of the first stage network is based on unpaired dataset, ii) there is some external paired real-world dataset available; hence, training of the first-stage network can be supervised.

Yuan *et al.*, 2018 propose a Cycle-in-Cycle GAN (CINCGAN) network, with GAN as the basic building block, using unpaired data for training. The CINCGAN architecture consists of two CycleGANs:

The first CycleGAN maps the LR image to the clean and bicubic-downsampled LR space. For training, LR images are generated synthetically from HR images. Then, the first network learns to map the distribution real LR images to the distribution of synthetically generated LR images from unpaired data. This module ensures that each LR input image is properly denoised/deblurred. Then, the intermediate bi-cubicly downsampled look-alike image is up-sampled to the desired size by a pre-trained deep model based on bicubic downsampling model. Finally, the parameters of the whole network are fine-tuned using adversarial learning in an end-to-end manner. Experiments on NTIRE2018 datasets demonstrate that the proposed unsupervised method achieves comparable results as the state-of-the-art supervised models.

Rad *et al.*, 2021 also proposed a similar two-stage process to handle the real-world SR problem in two steps. However, they assumed availability of paired real datasets captured by real cameras, such as the RealSR dataset. This way they trained the first stage network, converting real-world LR images to bicubic look-alike ones using paired real LR and bicubic LR pairs, where bicubic LR images are synthetically generated from ground-truth HR images. The second stage network, super-resolving bicubic look-alike images is trained on bicubic LR and ground-truth HR pairs. This approach yields excellent results provided that the real LR images have blur kernel that is the same as in the available real paired training data.

4.6.5 Real-World SR Challenges

AIM 2019 (Lugmayr *et al.*, 2019) and NTIRE 2020 (Lugmayr *et al.*, 2020a) real-world SR challenges were designed according to the real-world image acquisition setting, i.e., the blur kernel was unknown and no paired LR-HR images were provided.

In AIM 2019, there were two tracks. In Track 1, only one set of source (LR) input images is provided for training, where the goal is to super-resolve images while preserving the characteristics of the source input domain. In Track 2, a set of high-quality images is also provided for training, which defines the desired quality of target (HR) domain images. To allow for quantitative evaluation, the source LR images in both tracks

are generated by applying synthetic, but realistic degradations to a combination of DIV2K (Agustsson and Timofte, 2017) and Flickr2K (Wang *et al.*, 2018b) datasets. The quality of SR images was evaluated by a human study in terms of Mean Opinion Score (MOS). The winner team, MadDemon, first trained a network, called DSGAN, that can simulate the natural image characteristics (i.e. degradations). The team then used the generated data to train an SR model based on ESRGAN (Wang *et al.*, 2018b), which improves its performance on real-world data. Furthermore, they propose to separate the low and high frequency images and treat them differently during training. Since low frequencies are preserved by the downsampling operation, its corresponding upsampling operation can be trained using a simple pixel-wise loss. The teams Nam and CVML employ the inverse strategy, i.e. they first learn a network that cleans the image before super-resolution.

In NTIRE 2020 (Lugmayr *et al.*, 2020a), a set of images from the LR source domain and a set of unpaired HR images from the target domain were provided. In Track 1, the aim is to super-resolve images with synthetically generated image processing artifacts, which allows for quantitative benchmarking of the approaches compared to a ground-truth image. In Track 2, real low-quality smart phone (iPhone3) images of the DPED dataset (Ignatov *et al.*, 2017) have to be super-resolved. The results of both tracks are evaluated for perceptual quality based on a human study. Team Impressionism (Ji *et al.*, 2020), which focuses on estimating blur kernels and real noise distributions at the same time, obtained the best results for the distortion and perception metrics in both tracks. They employed KernelGAN (Bell-Kligler *et al.*, 2019) method for explicit estimation of blur kernel for Track 2.

5

Deep Video Restoration and Super-resolution

Video SR (VSR) can be posed as a sequence of single-image SR (SISR) problems or as a multi-frame SR (MFSR) problem. In contrast to SISR, which relies only on natural image priors and/or self-similarity within images to recover missing high-frequency details, MFSR additionally exploits temporal correlations between frames for improved performance. Basic VSR architectures consist of temporal propagation, frame/feature alignment, feature aggregation, and upsampling blocks and design choices for propagation and alignment can result in significant performance differences (Chan *et al.*, 2021a). Temporal information can be propagated locally within sliding temporal windows or over longer durations by recurrent architectures. Perhaps the most simplistic model of a video is to represent all frames of a scene by a single image and a set of motion trajectories (Tekalp, 2015), which is the basic assumption behind motion compensated frame/feature alignment. We review the state of the art sliding temporal window VSR architectures and recurrent VSR architectures in Section 5.1 and Section 5.2, respectively. Blind VSR architectures are introduced in Section 5.3. Perceptual video restoration and SR is discussed in Section 5.4. Section 5.5 provides a short introduction to commonly used VSR datasets.

5.1 Video SR based on Sliding Temporal Window

Sliding temporal window approaches leverage temporal information within $2N+1$ frames centered about the current frame t to be processed, i.e., propagation of temporal information is local inside the temporal window $t-N, \cdots, t+N$. Caballero *et al.*, 2017 classify sliding window temporal modelling approaches as: i) feature fusion using 2D ConvNets, and ii) spatio-temporal feature extraction using 3-D ConvNets. We discuss these two approaches in Section 5.1.1 and 5.1.2, respectively.

5.1.1 ConvNet Architectures for Feature Fusion

There exist a variety of ConvNet architectures for sliding temporal window feature fusion, which differ in their alignment and aggregation strategies. The frames can be aligned by either explicitly estimating optical flow in the pixel domain using a separate network or implicitly in the feature space by using deformable convolutions (Dai *et al.*, 2017), (Tian *et al.*, 2020) in order to fully exploit the temporal information. The frame/feature aggregation strategies can be classified as early fusion and slow fusion architectures (Caballero *et al.*, 2017).

In the early fusion architecture, the depth of the input layer of the network is set equal to the number of input frames $2N+1$ within the sliding window. The frames are concatenated after proper alignment and fed to the network as a stack. This will collapse all temporal information into a fused feature tensor in the first layer of the network (hence, the terminology early fusion) and the remaining layers are similar to those in a single image SR network. Alternatively, each frame can be input to a separate convolution layer (Kappeler *et al.*, 2016) or pairs of frames (with proper alignment) can be input to convolution layers (Caballero *et al.*, 2017) in parallel resulting in multiple feature tensors that can be concatenated in subsequent layers of the network. This process is generally referred to as slow fusion.

One of the earliest works that used neural networks for VSR proposed a two-layer network with fully connected layers (Cheng *et al.*, 2012). They take a sliding temporal window with 5 consecutive LR frames as input to reconstruct the center (current) frame. The video is processed

patch by patch, where a 3×3 HR patch from the current frame is reconstructed based on a $5 \times 5 \times 5$ LR volume. In order to generate the input LR volume, 5×5 patches from neighboring frames are aligned with the reference patch from the current frame by block matching.

Liao *et al.*, 2015 propose a two-stage method, where they generate high-resolution SR-drafts under different flow models. In the first stage, two motion compensation algorithms with 9 parameter settings are used to generate SR drafts in order to mitigate motion compensation errors. In the second stage, all drafts are combined using a ConvNet. Later, Kappeler *et al.*, 2016 proposed VSRNet, which estimates optical flow field to align corresponding patches across multiple frames by backward warping. In both methods, motion estimation step is separated from training the reconstruction ConvNet. Furthermore, Kappeler *et al.*, 2016 first upsamples and then warps frames, where both operations involve an interpolation, which causes loss of high-frequency details.

Caballero *et al.*, 2017 and Makansi *et al.*, 2017 independently proposed end-to-end video SR frameworks, which incorporate motion compensation as a submodule in the VSR network architecture. In these methods, flow estimation and HR reconstruction modules are trained simultaneosly using a single objective function. The integration requires motion compensation to be performed by a differentiable layer. To this effect, Caballero *et al.*, 2017 used an efficient spatial transformer network for motion compensation, while Tao *et al.*, 2017 proposed a sub-pixel motion compensation (SPMC) layer. Makansi *et al.*, 2017 proposed a joint upsampling and backward warping (JUBW) layer to perform upsampling and warping in a single step and showed this provides superior results. In addition, they showed image-based training provides results that are superior to patch-based training in Kappeler *et al.*, 2016.

Accuracy of estimated optical flow is crucial for temporal modeling and erroneous motion compensation can undermine results of video SR. To this effect, (Liu *et al.*, 2018) propose temporal adaptive neural network in order to robustly handle various types of motion and adaptively select the optimal range of temporal dependency to extract useful information among consecutive frames and alleviate the detrimental effect of erroneous motion estimation. The deep dual attention network (Li *et al.*, 2020) proposes using pyramid representation of motion. It con-

sists of two subnets: i) motion compensation net (MCNet) employs a pyramid representation of the reference and supporting frames to learn the optical flow and leverages the detail components of LR frames as complementary information to decrease mis-registration errors. ii) SR reconstruction net (ReconNet) implements dual attention mechanism to focus on informative features to recover high-frequency details.

A more effective approach to deal with the issues related to accuracy of optical flow estimation is to perform frame alignment in the feature space as opposed to pixel-domain alignment. The temporal deformable alignment network (TDAN) (Tian *et al.*, 2020) has been proposed to adaptively align the reference frame and each supporting frame using deformable convolutions in the feature space without computing optical flow. In the deformable convolution (Dai *et al.*, 2017), pixels under the kernel are displaced according to learned offsets $\Delta p_{(i,j)}$. Hence, deformable convolution can be expressed as:

$$y(p_{(m,n)}) = \sum_{(i,j)} w(i,j) \cdot x(p_{(m-i,n-j)} + \Delta p_{(i,j)}), \tag{5.1}$$

where $p_{(m,n)}$ denotes the current pixel, $w(i,j)$ are kernel weights, and $i,j \in (-1,0,1)$ for a 3×3 deformable convolution. TDAN uses features from both the reference frame and each supporting frame to dynamically predict sampling offsets of deformable convolution kernels. The HR video frame is predicted by aggregating aligned feature maps using a reconstruction network similar to other VSR approaches.

A highly successful implementation of the early fusion ConvNet architecture is the EDVR (Wang *et al.*, 2019). EDVR extends TDAN with two improvements: i) a Pyramid, Cascading and Deformable (PCD) alignment module, in which frame alignment is done at the feature level using deformable convolutions in a coarse-to-fine manner to handle large motions is implemented; ii) a Temporal and Spatial Attention (TSA) fusion module is proposed. Temporal attention weighs each neighboring feature at each location by the element-wise correlation between features of the reference frame and each neighboring frame. Spatial attention assigns weights to each location in each feature channel to exploit cross-channel more effectively. EDVR won all four tracks in the NTIRE 2019 video restoration and enhancement challenge (Nah *et al.*, 2019b).

Video Enhancement and SR Net (VESR-Net) (Chen *et al.*, 2020) improves upon the performance of EDVR by means of two innovations: i) Separate Non-Local (SNL) architecture for fusion of aligned features, and ii) Channel-Attention Residual Block (CARB) for reconstruction network. VESR-Net employs PCD alignment as in EDVR, but uses SNL architecture to aggregate information across aligned frames. For reconstruction, VESR-Net utilizes stacked CARB as in RCAN (Zhang *et al.*, 2018e) followed by a feature decoder. They employ L1 loss on the central target frame for training. Efficient Video Enhancement and Super-Resolution Net (EVESRNet) (Fuoli *et al.*, 2020) extends VESRNet by replacing the SNL module with a more efficient Efficient Point-Wise Temporal Attention Block (EPAB). This block aggregates the spatio-temporal information with less operations and memory consumption, while still providing the same high performance. EVESR-Net has won AIM 2020 (Fuoli *et al.*, 2020) Extreme Video SR Challenge.

Local-Global Fusion Network (LGFN) for Video SR (Su *et al.*, 2020) propose deformable convolutions (DCs) with decreased multi-dilation convolution units (DMDCUs) for efficient implicit alignment. Moreover, a structure with two branches, consisting of a Local Fusion Module (LFM) and a Global Fusion Module (GFM), is proposed to combine information from two different aspects. Specifically, LFM focuses on the relationship between adjacent frames and maintains the temporal consistency while GFM attempts to take advantage of all related features globally with a video shuffle strategy.

5.1.2 3-D Convolutional Networks

Perhaps the most straightforward extension of SISR ConvNet architectures to VSR is to replace 2-D filter kernels with 3-D kernels, resulting in 3-D ConvNets. Unlike 2-D filters with size $height \times width$ that slid horizontally and vertically and are applied on the full channel depth, 3-D filters have a third size parameter, $temporaldepth$, so that they are swept horizontally, vertically, and temporally (Tran *et al.*, 2015). 3-D ConvNets are effective because they extract spatio-temporal features directly from raw video. A 3-D convolution layer can be seen as a special case of slow fusion (Caballero *et al.*, 2017). In slow fusion, each

layer merges features from d frames, where d is smaller than the number of frames $2N+1$ in the sliding temporal window. If successive 2-D layers share weights, then slow fusion is equivalent to a single layer of 3-D convolution to extract spatio-temporal features.

For a given number of layers, 3-D ConvNets have larger number of learnable parameters compared to 2-D ConvNets; hence, require very large video datasets and more computational resources for effective training (Hara *et al.*, 2018). Resource efficient 3-D ConvNet architectures that are extensions of well-known efficient 2-D ConvNets (e.g., SquezeNet, MobileNet, etc.) with group convolutions and depth-wise separable convolutions as main building blocks have been introduced to alleviate this problem and compared in terms of number of layers, nonlinearities, and skip connections in (Köpüklü *et al.*, 2019).

VSR architectures that employ 3-D convolution layers include Jo *et al.*, 2018, which is an end-to-end deep neural network that generates dynamic upsampling filters (DUF) and a residual image that are computed depending on the local spatio-temporal neighborhood of each pixel to avoid explicit motion compensation. Kim *et al.*, 2019 stacked multiple 3-D convolutional layers to extract both spatial and temporal features within a temporal sliding window over an entire video. Deformable 3-D Net (Ying *et al.*, 2020) integrates 3-D convolution (Tran *et al.*, 2015) and deformable convolution (Dai *et al.*, 2017) to propose deformable 3-D convolution (D3D), which can achieve efficient spatio-temporal information exploitation and adaptive motion compensation.

An empirical investigation of efficient spatio-temporal modeling (Fan *et al.*, 2019) has shown that early fusion models can achieve comparable results to 3-D ConvNet models with less computational complexity.

5.2 Video SR based on Recurrent Architectures

VSR methods based on a sliding temporal window treat reconstruction of each HR frame as a separate multi-frame SR task, which has weaknesses, including: i) reconstructing each HR frame, conditioned on the input frames, independently limits the ability of the system to produce temporally consistent HR frames, and ii) each input LR frame is warped multiple times, increasing the computational cost.

Recurrent neural network (RNN) architectures have been shown to be highly effective in sequential processing of time series data. The recurrent model captures the temporal dependencies in a hidden state that is passed to successive time steps. RNNs are trained using backpropagation-through-time (Werbos, 1990) in a stateless or stateful manner. When unfolded into time steps, RNNs are analogous to deep ConvNets with shared parameters. Hence, training vanilla RNNs suffers from vanishing/exploding gradients problem after a number of time steps (layers) similar to training vanilla deep ConvNets. The vanilla RNN also has limited capability to remember long term dependencies.

Long Short Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997) have been proposed to address both vanishing/exploding gradients problem and modeling long-term dependencies. LSTM introduces input, output, and forget gates to control the flow of information. These gates learn the relevant information to keep in the cell state, i.e., the long-term dependencies, during training using sigmoid activations. The gates also enable flow of information to avoid vanishing/exploding gradients. Gated recurrent unit (GRU) (Cho *et al.*, 2014) is a simplified form of LSTM with only two gates: update gate and reset gate.

ConvNets are ideally suited to process images, while LSTMs are effective for processing time series data. Since video can be considered as a time sequence of images, a natural choice would be to combine the benefits of both architectures. To this effect, ConvLSTM layer (Shi *et al.*, 2015) has been proposed as an effective tool for video processing. The hidden states in ConvLSTM are 2-D feature maps rather than 1-D vectors as in RNN/LSTM. ConvLSTM is a recurrent layer, just like LSTM, but internal matrix multiplications representing fully-connected neurons are replaced by convolution operations; hence, the number of parameters are reduced from millions to a few tens of thousands. As a result, ConvLSTM layers are highly efficient to compute and to train.

A simple approach for VSR without motion estimation and compensation would be to consider LR frames as input to a ConvLSTM layer; then, the predicted HR frames can be reconstructed by upsampling the outputs at different time steps. Deep recurrent Resnet (DRRNet) for video SR (Lim and Lee, 2017) consists of 10 residual blocks, where the first residual block has two ConvLSTM layers followed by 9 regular

residual blocks each formed by Convolution-RELU-Convolution layers. There is also a global skip connection to implement residual learning. All convolutions are 3×3, and the number of channels is 64. The DRRNet upsamples LR feature maps at the last stage using a combination of convolution and pixel shuffle layers before forming HR frames.

Huang *et al.*, 2015; Huang *et al.*, 2018 proposed a bidirectional recurrent convolutional network (BRCN), which is a ConvLSTM network with 3-D convolutions for inputs rather than the conventional 2-D convolutions. The 3D convolutions extract features from not only the input at the current time step but also from multiple adjacent layers at previous time steps to transfer to the current hidden state. Hence, the hidden state is able to capture informative patterns along both spatial and temporal dimensions to more effectively describe slow and fast varying motions across a series of frames. Different from recurrent convolutions connecting hidden states that mainly deal with long-term slow-varying motions, 3D convolutions connecting inputs focus on features of fast varying motions. They directly operate on original frames that can provide more visual details than abstracted hidden layers. The classic LSTM runs in the forward direction along the timeline, which models the dependency between the current frame and its previous frames. To additionally consider the dependency between the current frame and future frames, BRCN combines a forward and a backward sub-networks to jointly make the final prediction.

Spatial-temporal recurrent residual network (STR-ResNet) (Yang *et al.*, 2018) takes not only the LR frames but also the differences of adjacent LR frames as input. When the recurrent units are unrolled, STR-ResNet connects SISR networks for each frame to embed the temporal correlation. It reconstructs an HR frame through fusing its corresponding LR frame and the predicted spatial residue, under the guidance of the predicted temporal residues among adjacent frames.

While Lim and Lee, 2017 and Huang *et al.*, 2018 pass the hidden state that represents long-term features to the next step, the frame-recurrent video SR (FRVSR) framework (Sajjadi *et al.*, 2018) passes the previous output frame to the next step to produce temporally consistent HR videos. However, FRVSR requires explicit motion estimation and warping operations to exploit temporal information.

Efficient video super-resolution through recurrent latent space propagation (RLSP) (Fuoli *et al.*, 2019) also passes the previous output frame to the next step similar to (Sajjadi *et al.*, 2018) but does not employ on an explicit motion compensation module, instead relies on a recurrent hidden state, and takes multiple LR frames within a sliding temporal window as input similar to (Huang *et al.*, 2018) to efficiently leverage temporal information implicitly. The multiple input LR frames are concatenated together with the recurrent LR state tensor h_{t-1} and subsampled previous output frame. The combined tensor is then fed into a convolution RLSP cell with $n = 7$ layers and ReLU activation. The outputs of the RLSP cell at each time step t are the hidden state h_t for the next time step and the output HR frame.

Recurrent Back-Projection Network (Haris *et al.*, 2019) extends Deep Back-Projection Networks (DBPN) (Haris *et al.*, 2021) developed for the SISR task to the multiple-image SR (MISR) or VSR task. DBPN produces a HR feature map, which is iteratively refined through multiple up- and down-sampling layers. RBPN relates multiple input video frames as LR images to compute residuals for the target HR frame, where HR feature maps representing missing details are iteratively refined by up- and down-sampling processes to improve the quality of SR.

Isobe *et al.*, 2020 revisits the question of what is the best temporal modeling strategy, and makes the observation that it is hard to compare the effectiveness of various approaches directly from published results because different methods adopt different network sizes, loss functions, and training sets to train their models. The experimental results in Isobe *et al.*, 2020, which were obtained with comparable training strategies and data, show that recurrent models are highly efficient and effective for the VSR task. They propose incorporating residual connection into the hidden state of the recurrent network and call the resulting model as Recurrent Residual Network (RRN) for VSR and show that RRN achieves the state-of-the-art performance on three benchmarks.

Chan *et al.*, 2021a observe that most VSR methods consist of four inter-related components, namely, propagation, alignment, aggregation, and upsampling. They find that different propagation and alignment strategies significantly affect the results and that bidirectional propagation coupled with a simple optical flow-based feature alignment suffice to

outperform many state-of-the-art methods. Many recurrent methods do not perform alignment of features/images during propagation. Without proper alignment, local convolution operations, which have relatively small receptive fields, are inefficient in aggregating the information across multiple frames. It is also observed that image-based alignment is inferior to feature alignment. BasicVSR (Chan *et al.*, 2021a) adopts bi-directional recursion for temporal propagation and optical flow for spatial alignment, but instead of warping images as in previous works, feature warping is performed for better performance. They show there is a drop of 1.19 dB in PSNR if alignment is skipped. The aggregation and upsampling steps of BasicVSR are similar to other VSR works.

BasicVSR++ (Chan *et al.*, 2021b) improves the performance of BasicVSR further by introducing second-order grid propagation and flow-guided deformable alignment. Benefiting from these new components, BasicVSR++ surpasses BasicVSR by 0.82 dB in PSNR with similar number of parameters. In addition, BasicVSR++ generalizes to other video restoration tasks, such as compressed video enhancement, well. In the NTIRE 2021 Video Super-Resolution and Compressed Video Enhancement Challenges, BasicVSR++ was ranked first-place three times and runner-up once.

5.3 Blind Video Restoration and Super-resolution

All learned VSR methods discussed so far assume a very simple LR frame formation model with a fixed down-sampling filter, e.g., a bi-cubic or Gaussian kernel. In Section 4.5.2, we stated that real-world learned SISR is complicated because real-world LR images are formed by various types of down-sampling filters and factors, and learned SISR methods have a tendency to overfit the down-sampling filter used in the synthetic training set. Real-world LR video frame formation is further complicated by motion-induced spatial blurring in addition to the down-sampling anti-alias filter. Since motion blur is video-dependent, the corresponding kernel should be estimated for each test video individually. Hence, blur kernel estimation should be part of the VSR network architecture.

Methods that include a blur kernel estimation module are called blind SISR/VSR architectures. Blind VSR methods first estimate the un-

known blur kernel, and then employ the predicted kernel in the SR model. Existing kernel estimation approaches either exploit self-similarity with the hypothesis that similar patterns and structures across different scales appear in natural images or design an iterative algorithm. In the blind VSR setting, each LR video may have resulted from a different down-sampling process and may suffer from motion blur; therefore, test-time kernel estimation/adaptation is crucial.

Deep blind video SR (Pan *et al.*, 2021) propose a deep ConvNet that consists of motion blur kernel estimation, motion estimation, and latent image restoration modules. The motion blur estimation module has two fully connected layers, where the first layer is followed by a ReLU activation and the second one by a softmax function to ensure that elements of the blur kernel are non-negative and their sum is 1. Following the plug-and-play SISR with deep denoiser prior (Zhang *et al.*, 2021), they first estimate an intermediate latent HR image using a closed-form deconvolution model with the estimated blur kernel and then explore the information in adjacent frames using a deep ConvNet prior to restore the final high-quality HR image.

DynaVSR (Lee *et al.*, 2021) introduces an efficient Multi-Frame Downscaling Network (MFDN), and combines it with the VSR network to adapt to each dynamically-varying input video. The training process of DynaVSR consists of three stages: 1) estimation of the unknown down-sampling process with the MFDN, 2) joint adaptation of MFDN and VSR network parameters with respect to each input video, and 3) meta-updating the base parameters for MFDN and VSR network. At test time, only steps 1) and 2) are processed, and updated parameters of the VSR network are used to generate the final HR frames.

5.4 Perceptual Video Restoration and Super-resolution

Learned VSR methods in Section 5.1-Section 5.3 have been trained to minimize the average per-pixel distortion, such as l_2, l_1, Charbonnier, or Huber loss, between estimated HR frames and the corresponding ground truth frames given synthetically generated LR and HR training video pairs. However, it is well-known that optimization with respect to a distortion loss typically results in blurry unnatural looking textures.

Different from the classical model-based methods, learned SR allows for optimization with respect to perceptual losses. This was first exploited in the SISR problem, where it was shown that a trade-off exists between fidelity and perceptual image quality (see Section 4.4). The first Challenge on Perceptual Image SR (Blau *et al.*, 2018) was motivated by this trade-off, and the winner was a generative model, called ESRGAN (Wang *et al.*, 2018b). Recently researchers started to extend perceptual SISR methods to perceptual VSR. Perceptual SR methods provide sharper texture in each frame of video at the expense of a decrease in PSNR as predicted by perception-distortion trade-off (see Section 3.5). Perceptual SR methods can be classified as: i) methods that employ full reference perceptually motivated loss functions, and ii) generative methods that employ a no-reference perceptual loss or adversarial loss in addition to l2/l1 loss. We focus on the latter here.

A Generative Adversarial Network (GAN) formulation for VSR with an adversarial texture loss was proposed in (Lucas *et al.*, 2019). They introduce VSRResNet as the Generator along with a discriminator network. However, they address temporal consistency of SR frames neither in the training process nor in evaluation. IseeBetter (Chadha *et al.*, 2020) is another GAN-based spatio-temporal approach to VSR that aims to render temporally consistent SR videos. ISeeBetter extracts spatial and temporal information from the current and neighboring frames using the recurrent back-projection networks (Haris *et al.*, 2021) as its generator. The authors use the discriminator from SRGAN. They employ a a combination of four loss functions, namely, MSE, perceptual, adversarial, and total-variation (TV) loss, for training.

It is important to note that typical VSR methods, whether based on fidelity alone or perceptual optimization, calculate losses per frame, and therefore, do not take temporal consistency into account explicitly. Temporal inconsistencies result in a low perceptual quality SR video even if the texture in each frame looks natural, as humans can detect motion jitter easily. There exist few works that model or enforce temporal consistency of frames or naturalness of motion in VSR explicitly.

Pérez-Pellitero *et al.*, 2018 and Pérez-Pellitero *et al.*, 2019 treat temporal consistency explicitly using a recurrent network with adversarial training. The proposed recurrent architecture leverages information

from previous frames, i.e., the input to the generator is composed of the LR image and the warped output of the network at the previous step. Together with a video discriminator, they propose static temporal loss and temporal statistics loss to further reinforce temporal consistency in the generated sequences.

TempoGAN (Xie *et al.*, 2018) is a temporally coherent generative model addressing the SR problem for fluid flows. It employs a spatial discriminator and a temporal discriminator, which take a triple set of aligned ground-truth and super-resolved frames as inputs. While other works make use of manually selected layers of pre-trained networks, such as the VGG net, as feature loss, TempoGAN uses features of the discriminator as constraints instead.

TecoGAN (Chu *et al.*, 2020) propose a temporally self-supervised algorithm and show that temporal adversarial learning is key to achieving temporally coherent solutions without sacrificing spatial detail. They introduce a spatio-temporal discriminator structure together with a set of training objectives for a realistic and coherent VSR task. They also propose a Ping-Pong loss to improve the long-term temporal consistency.

Motivated by the perceptual straightening hypothesis (Henaff *et al.*, 2019) of the human visual system, (Kancharla and Channappayya, 2021) proposed a quality-aware discriminator model to enforce the straightness on the trajectory of the perceptual representations of video frames. To extract the perceptual representation, lateral geniculate nucleus (LGN) is implemented using a two-stage model, where the first stage is composed of bandpass filters and the second stage is a non-linear block that performs luminance and contrast gain control.

The VSR models are typically evaluated by the classic distortion measures, such as PSNR and SSIM, as well full-reference perceptually-motivated measures, such as LPIPS (Zhang *et al.*, 2018d), and no-reference perceptual measures, such as Natural Image Quality Evaluator (NIQE) (Mittal *et al.*, 2013). However, all these metrics are originally designed for single images. When used for video, different poolings over frames are reported. Consequently, these measures cannot reflect the effect of temporal consistency and motion artifacts. An early measure that was developed for video is MOtion-based Video Integrity Evaluation (MOVIE) (Seshadrinathan and Bovik, 2010), which utilizes a spatio-

spectrally localized multiscale framework for evaluating dynamic video fidelity that integrates both spatial and temporal (and spatio-temporal) aspects of distortion assessment. More recently, TecoGAN introduced tOF and tLP as temporal coherency metrics (See Section 3.3). tOF measures the pixel-wise difference of motions estimated from sequences and tLP is based on the error between LPIPS of sequential frames for predicted and pristine videos, justifying that natural videos exhibit a certain amount of perceptual changes over time. STraightness Evaluation Metric (STEM) (Kancharla and Channappayya, 2022) combines the NIQE metric with a blind temporal NR-VQA algorithm, which is based on the straightness of perceptual trajectory of natural video frames.

5.5 Video SR Datasets

Popular datasets for VSR research include Vimeo-90K (Xue *et al.*, 2019), REDS (Nah *et al.*, 2019a), and RealVSR (Yang *et al.*, 2021).

Vimeo-90K consists of more than 90,000 septuplets collected over the Internet. Each septuplet contains 7 frames with spatial resolution 256×448. REDS dataset consists of 300 sequences captured by the GOPRO sport camera. Each sequence contains 100 frames with spatial resolution 720×1280. When using these datasets, LR sequences are synthesized from HR sequences with simple degradation models, such as bicubic downsampling or direct downsampling after Gaussian smoothing.

When applying the VSR models trained on these datasets to real-world LR videos, the super-resolved videos are often over-smooth and prone to visual artifacts because the degradation process of real-world videos is more complex. This motivates the need for a new dataset for real-world VSR research. The RealVSR dataset (Yang *et al.*, 2021) consists of paired LR-HR video sequences recorded by the multi-camera system of iPhone 11 Pro Max to capture the same dynamic scene simultaneously. Since the LR-HR video pairs are captured by two separate cameras, the users should be aware that there may be misalignment and luminance/color differences between them.

6

Conclusions

Learned image/video restoration and SR rest on three pillars, the architecture, the loss function, and training data and methodology, which affect the quality of results. The quality of the results need to be discussed with respect to the appropriate evaluation/assessment criteria, which could be fidelity or perceptual criteria. The desired trade-off between them depends on the application, where for information-centric applications fidelity is more important than perceptual quality, while it is vice versa for aesthetics-centric applications. Sections 6.1 and 6.2 summarize the current state of the art, open problems, and future directions in each of the three pillars for SISR and VSR tasks, respectively.

6.1 State-of-the-art and Future Directions in Learned SISR

Standard SISR problem: We can consider the problem of model training with l_1 or l_2 loss using paired LR-HR data, where LR images are synthetically generated with a known blur kernel as solved. The only factor that would affect the results is the network architecture, and network architectures for SISR have saturated, i.e., there is a small difference on the quality of the results measured by PSNR or MS-SSIM between the recent competing architectures, perhaps less than 0.5 dB.

Open problems for further research include:

Perceptual SISR problem: Conditional generative models hallucinate textures learned from distribution of HR images, which result in sharper SR images that are more appealing to human viewers. However, this comes at the expense of reduced fidelity compared to the ground-truth since some of the hallucinated texture may not be real or may not be well registered with the ground-truth. This brings the issue of importance of fidelity for a given application and what is the best distortion-perception tradeoff. Furthermore, there are several approaches to generative modeling including GAN vs. flow-based models vs. diffusion models. Which generative modeling approach is best suited to perceptual SR is an open research problem.

Real-world SISR problem: This is perhaps the most interesting open research problem, since both the standard SR and perceptual SR problem formulations assume a known blur kernel to generate a synthetic paired training set and learned SR models do not generalize well to blur kernels other than the one used in the training set. In the real-world setting, the blur kernel is unknown and there are no paired data to train the model, which makes the problem challenging. The main solutions proposed in the literature are: i) two-step process, where the unknown blur kernel is first identified and then a non-blind restoration/SR method is employed, or ii) unsupervised training of a blind SR model using an unpaired dataset. Since the former approach is prone to blur kernel estimation errors, the latter seems more promising.

6.2 State-of-the-art and Future Directions in Learned VSR

Multi-frame image SR vs. VSR: The traditional sliding temporal window problem formulation is to super-resolve each video frame independently benefiting from temporal correlations among adjacent frames. However, this approach neglects the temporal consistency between the SR video frames and may result in flickering artifacts. Proposed solutions to address this problem include employing recurrent networks and/or explicit temporal consistency constraints.

VSR architectures: While the SISR architectures have saturated, the VSR architecture is still an open problem. Recurrent network

architectures appear to be more efficient and effective compared to fixed-length sliding temporal window approach for propagation of temporal information. Furthermore, both pixel-domain optical flow-based and feature-space deformable convolutions have their own merits for frame alignment. Flow-guided recurrent architectures, which utilized a combination of these methods appear to be the most promising approach to exploit both short-term and long-term temporal correlations in a video. This is still an open research problem.

Perceptual VSR: While generative models can render sharper textures in each frame, they exacerbate the temporal consistency problem in video SR. Hence, a current research problem of interest is to use a combination of adversarial losses (for better spatial texture) and temporal consistency losses to obtain SR videos with sharper texture and temporal consistency.

Real-world VSR: Blind restoration/super-resolution of real videos with unpaired training data is a relatively unexplored research problem.

Acknowledgements

The author acknowledges support from TUBITAK 2247-A grant number 120C156. The author also thanks his graduate students Onur Keleş, Ogün Kırmemiş, Nasrin Rahimi, Cansu Korkmaz, M. Akın Yılmaz, and Ronglei Ji for their research contributions.

References

Agustsson, E. and R. Timofte. (2017). "NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshops*.

Anwar, S., S. Khan, and N. Barnes. (2021). "A deep journey into super-resolution: A survey". *ACM Computing Surveys (CSUR)*. 53(3): 1–34.

Bahat, Y. and T. Michaeli. (2020). "Explorable Super Resolution". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*. 2716–2725.

Baker, S. and T. Kanade. (2000). "Hallucinating faces". In: *Proc. of the IEEE Int. Conf. on Automatic Face and Gesture Recognition*.

Bell-Kligler, S., A. Shocher, and M. Irani. (2019). "Blind super-resolution kernel estimation using an internal-GAN". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 284–293.

Bello, I., B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le. (2019). "Attention augmented convolutional networks". In: *Int. Conf. on Computer Vision (ICCV)*.

Bianco, S., L. Celona, P. Napoletano, and R. Schettini. (2018). "On the use of deep learning for blind image quality assessment". *Signal, Image, and Video Processing (SIViP)*. 12: 355–362.

Blau, Y., R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor. (2018). "The 2018 PIRM Challenge on Perceptual Image Super-resolution". In: *Euro. Conf. Comp. Vision (ECCV) Workshop.*

Blau, Y. and T. Michaeli. (2018). "The perception-distortion tradeoff". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR).*

Blog. (2016). "Toward A Practical Perceptual Video Quality Metric". *Netflix Technology (2016-06-06), Last Accessed: 2021-07-01.*

Brabandere, B. D., X. Jia, T. Tuytelaars, and L. V. Gool. (2016). "Dynamic filter networks". In: *Advances in Neural Information Processing Systems (NeurIPS).*

Buades, A., B. Coll, and J.-M. Morel. (2005). "A non-local algorithm for image denoising". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR).*

Caballero, J., C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi. (2017). "Realtime video super-resolution with spatio-temporal networks and motion compensation". In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR).*

Cai, J., S. Gu, and R. o. Timofte. (2019a). "NTIRE 2019 Challenge on Real Image Super-Resolution: Methods and Results". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshop.*

Cai, J., H. Zeng, H. Yong, Z. Cao, and L. Zhang. (2019b). "Toward real-world single image super-resolution: A new benchmark and a new model". In: *IEEE Int. Conf. on Computer Vision.*

Chadha, A., J. Britto, and M. M. Roja. (2020). "iSeeBetter: Spatio-temporal video super-resolution using recurrent generative back-projection networks". *Springer Jour. of Computational Visual Media, Tsinghua University Press.* 6(3): 307–317.

Chan, K., X. Wang, K. Yu, C. Dong, and C. Loy. (2021a). "BasicVSR: The search for essential components in video super-resolution and beyond". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recognition (CVPR).*

Chan, K. C., S. Zhou, X. Xu, and C. C. Loy. (2021b). "BasicVSR++: Improving Video Super-Resolution with Enhanced Propagation and Alignment".

Chandler, D. M. and S. Hemami. (2007). "VSNR: A wavelet-based visual signal-to-noise ratio for natural images". *IEEE Trans. on Image Processing*. 16(9): 2284–2298.

Chellapilla, K., S. Puri, and P. Simard. (2006). "High performance convolutional neural networks for document processing". In: *Int. Workshop on Frontiers in Handwriting Recognition*.

Chen, H. *et al.* (2021). "Pre-trained Image Processing Transformer". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*.

Chen, J., X. Tan, C. Shan, S. Liu, and Z. Chen. (2020). "VESR-Net: The Winning Solution to Youku Video Enhancement and Super-Resolution Challenge". *arXiv preprint arXiv:2003.02115*.

Cheng, M., N. Lin, K. Hwang, and J. Jeng. (2012). "Fast video super-resolution using artificial neural networks". In: *Int. Symp. Commun. Syst. Netw. Digital Signal Proc. (CSNDSP)*. 1–4.

Cheon, M., S.-J. Yoon, B. Kang, and J. Lee. (2021). "Perceptual image quality assessment with transformers". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshops*.

Cho, K., B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. (2014). "Learning phrase representations using RNN encoder–decoder for statistical machine translation". In: *Empirical Methods in Natural Language Processing (EMNLP)*. 1724–1734.

Chu, M., Y. Xie, J. Mayer, L. Leal-Taixe, and N. Thuerey. (2020). "Learning temporal coherence via self supervision for GAN-based video generation". *ACM Trans. on Graphics (TOG)*. 39(4): 75–1.

Ciresan, D. C., U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber. (2011). "Flexible, High Performance Convolutional Neural Networks for Image Classification". In: *Int. Joint Conf. on Artificial Intelligence (IJCAI), Barcelona, Spain*.

Dai, J., H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. (2017). "Deformable Convolutional Networks". In: *IEEE Int. Conf. on Computer Vision (ICCV)*. 764–773. DOI: 10.1109/ICCV.2017.89.

Daly, S. (1993). *The visible difference predictor: An algorithm for the assessment of image fidelity*. Ed. by E. A.B. Watson. *Digital Images and Human Vision*. Cambridge, MA: MIT Press. 179–206.

Dendi, S. V. R. and S. S. Channappayya. (2020). “No-reference video quality assessment using natural spatiotemporal scene statistics”. *IEEE Trans. Image Proc.* 29: 5612–5624.

Ding, K., K. Ma, S. Wang, and E. P. Simoncelli. (2020). “Image quality assessment: Unifying structure and texture similarity”. *IEEE Trans. on Patt. Anal. Mach. Intel. (PAMI).*

Ding, K., K. Ma, S. Wang, and E. P. Simoncelli. (2021). “Comparison of Image Quality Models for Optimization of Image Processing Systems”. *Int. J. Comput. Vis. (IJCV).* 129(4): 1258–1281.

Dinh, L., J. Sohl-Dickstein, and S. Bengio. (2017). “Density estimation using real NVP”. In: *Int. Conf. on Learning Representations (ICLR).*

Dong, C., C. C. Loy, K. He, and X. Tang. (2014). “Learning a Deep Convolutional Network for Image Super-Resolution”. In: *European Conf. on Computer Vision (ECCV).*

Dong, C., C. C. Loy, K. He, and X. Tang. (2016). “Image Super-Resolution Using Deep Convolutional Networks”. *IEEE Trans. Pattern Anal. Mach. Intell.* 38(2): 295–307.

Dosovitskiy, A., L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. (2021). “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *Int. Conf. Learning Representations (ICLR).*

Erdogmus, D. and J. C. Principe. (2006). “From linear adaptive filtering to nonlinear information processing”. *IEEE Signal Processing Magazine*: 14–33.

Fan, Y., J. Yu, D. Liu, and T. S. Huang. (2019). “An empirical investigation of efficient spatio-temporal modeling in video restoration”. In: *IEEE/CVF Int. Conf. on Comp. Vision Workshop (ICCVW).*

Feng, R., J. Gu, Y. Qiao, and C. Dong. (2019). “Suppressing Model Overfitting for Image Super-Resolution Networks”. In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR) Workshops.*

Ferzli, R. and L. J. Karam. (2009). “A no-reference objective image sharpness metric based on the notion of just noticeable blur (JNB)”. *IEEE Trans. on Image Processing.* 18(4): 717–728.

Freeman, W., T. Jones, and E. Pasztor. (2002). "Example-based super-resolution". *IEEE Computer Graphics and Applications*. 22(2): 56–65.

Fukushima, K. (1980). "Neocognitron : A self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". *Biological Cybernetics*. 36(4): 193–202.

Fuoli, D. *et al.* (2020). "AIM 2020 Challenge on Video Extreme Super-Resolution: Methods and Results". In: *IEEE/CVF Int. Conf. on Comp. Vision Workshop (ICCVW)*.

Fuoli, D., S. Gu, and R. Timofte. (2019). "Efficient video super-resolution through recurrent latent space propagation". In: *Int. Conf. Comp. Vision (ICCV) Workshops*.

Gao, F., Y. Wang, P. Li, M. Tan, J. Yu, and Y. Zhu. (2017). "Deepsim: Deep similarity for image quality assessment". *Neurocomputing*. 257: 104–114.

Goodfellow, I., Y. Bengio, and A. Courville. (2016). *Deep Learning*. MIT Press, Cambridge.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. (2014). "Generative adversarial nets". In: *Advances in Neural Information Processing Systems (NIPS)*. Vol. 2. 2672–2680.

Gu, J., H. Cai, C. Dong, *et al.* (2021). "NTIRE 2021 Challenge on perceptual image quality assessment". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshops*.

Gu, J., H. Lu, W. Zuo, and C. Dong. (2019). "Blind super-resolution with iterative kernel correction". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*.

Hara, K., H. Kataoka, and Y. Satoh. (2018). "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet?" In: *IEEE/CVF Conf. on Comp. Vis. Patt. Recog. (CVPR)*.

Haris, M., G. Shakhnarovich, and N. Ukita. (2019). "Recurrent back-projection network for video super-resolution". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR)*.

Haris, M., G. Shakhnarovich, and N. Ukita. (2021). "Deep back-projection networks for single image super-resolution". *IEEE Trans. on Patt. Anal. and Mach. Intel. (PAMI)*.

He, K., X. Zhang, S. Ren, and J. Sun. (2016a). "Deep residual learning for image recognition". In: *IEEE/CVF Computer Vision and Patt. Recog. (CVPR)*.

He, K., X. Zhang, S. Ren, and J. Sun. (2016b). "Identity mappings in deep residual networks". In: *European Conf. on Comp. Vision (ECCV)*.

Helmrich, C. R., M. Siekmann, S. Becker, S. Bosse, D. Marpe, and T. Wiegand. (2020). "XPSNR: A low-complexity extension of the perceptually weighted peak signal-to-noise-ratio for high-resolution video quality assessment". In: *ICASSP*.

Henaff, O. J., R. L. Goris, and E. P. Simoncelli. (2019). "Perceptual straightening of natural videos". *Nature Neuroscience*. 22(6): 984–991.

Heusel, M., H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. (2017). "GANs trained by a two time-scale update rule converge to a local Nash equilibrium". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 6626–6637.

Hochreiter, S. and J. Schmidhuber. (1997). "Long short-term memory". *Neural Computation*. 9(8): 1735–1780.

Hornik, K., M. Tinchcombe, and H. White. (1989). "Multilayer feed-forward networks are universal approximators". *Neural Networks, Pergamon Press*. 2: 359–366.

Hu, H., Z. Zhang, Z. Xie, and S. Lin. (2019). "Local Relation Networks for Image Recognition". In: *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*. 3464–3473.

Huang, G., Z. Liu, L. van der Maaten, and K. Q. Weinberger. (2017). "Densely connected convolutional networks". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*. Honolulu, HI.

Huang, Y., W. Wang, and L. Wang. (2015). "Bidirectional recurrent convolutional networks for multi-Frame super-resolution". In: *Neural Information Processing Systems (NIPS)*. Vol. 28.

Huang, Y., W. Wang, and L. Wang. (2018). "Video super-resolution via bidirectional recurrent convolutional networks". *IEEE Trans. on Patt. Anal. and Mach. Intel.* 40(4): 1015–1028.

Ignatov, A., N. Kobyshev, R. Timofte, K. Vanhoey, and L. V. Gool. (2017). "DSLR-Quality Photos on Mobile Devices with Deep Convolutional Networks". arXiv: 1704.02470 [cs.CV].

Irani, M. and S. Peleg. (1991). "Improving Resolution by Image Registration". *CVGIP: Graphical Models and Image Processing*. 53(3): 231–239.

Isobe, T., F. Zhu, X. Jia, and S. Wang. (2020). "Revisiting temporal modeling for video super-resolution". In: *British Mach. Vision Conf. (BMVC)*.

Jain, V. and S. Seung. (2008). "Natural image denoising with convolutional networks". In: *Advances in Neural Information Processing Systems (NIPS)*. 769–776.

Jang, D.-W. and R.-H. Park. (2019). "DenseNet with deep residual channel-attention blocks for single image super resolution". In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR) Workshops*.

Ji, X., Y. Cao, Y. Tai, C. Wang, J. Li, and F. Huang. (2020). "Real-World Super-Resolution via Kernel Estimation and Noise Injection". In: *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Jo, Y., S. Oh, J. Kang, and S. J. Kim. (2018). "Deep video superresolution network using dynamic upsampling filters without explicit motion compensation". In: *IEEE/CVF Conf. on Comp. Vis. Patt. Recog. (CVPR)*.

Jo, Y., S. Yang, and S. J. Kim. (2021). "SRFlow-DA: Super-resolution using normalizing flow with deep convolutional block". In: *IEEE/CVF Conf. Comp. Vision and Patt. Recog. (CVPR) Workshop*.

Johnson, J., A. Alahi, and L. Fei-Fei. (2016). "Perceptual losses for real-time style transfer and super-resolution". In: *European Conf. on Comp. Vision (ECCV)*.

Jolicoeur-Martineau, A. (2019). "The relativistic discriminator: A key element missing from standard GAN". In: *Int. Conf. on Learning Representations (ICLR)*.

Kancharla, P. and S. S. Channappayya. (2021). "Improving the visual quality of video frame prediction models using the perceptual straightening hypothesis". *IEEE Signal Processing Letters*. 28: 2167–2171.

Kancharla, P. and S. S. Channappayya. (2022). "Completely blind quality assessment of user generated video content". *IEEE Trans. on Image Processing*. 31: 263–274.

Kang, L., P. Ye, Y. Li, and D. Doermann. (2014). "Convolutional neural networks for no-reference image quality assessment". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*. 1733–1740.

Kappeler, A., S. Yoo, Q. Dai, and A. K. Katsaggelos. (2016). "Video super resolution with convolutional neural networks". *IEEE Trans. on Computational Imaging*. 2(2): 109–122.

Keleş, O., A. M. Tekalp, J. Malik, and S. Kıranyaz. (2021a). "Self-organized residual blocks for image super-resolution". In: *IEEE Int. Conf. on Image Processing*. Anchorage, Alaska, USA.

Keleş, O., M. A. Yılmaz, A. M. Tekalp, C. Korkmaz, and Z. Dogan. (2021b). "On the computation of PSNR for a set of images and video". In: *Picture Coding Symp. (PCS)*.

Kim, J., J. K. Lee, and K. M. Lee. (2016). "Accurate image superresolution using very deep convolutional networks". In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR)*.

Kim, S. Y., J. Lim, T. Na, and M. Kim. (2019). "3DSRNet: Video superresolution using 3D convolutional neural networks". In: *IEEE Int. Conf. on Image Proc. (ICIP)*.

Kingma, D. P. and P. Dhariwal. (2018). "Glow: Generative flow with invertible 1x1 convolutions". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 10236–10245.

Kingma, D. P. and M. Welling. (2014). "Auto-encoding variational Bayes". In: *Int. Conf. Learning Representations (ICLR)*.

Kiranyaz, S., J. Malik, H. B. Abdallah, T. Ince, A. Iosifidis, and M. Gabbouj. (2021). "Self-organized Operational Neural Networks with Generative Neurons". *Neural Networks*. 140: 294–308.

Kirmemis, O. and A. M. Tekalp. (2018). "Effect of training and test datasets on image restoration and super-resolution by deep learning". In: *Proc. of the EUSIPCO*. Rome, Italy. 514–518.

Kobyzev, I., S. Prince, and M. A. Brubaker. (2021). "Normalizing flows: An introduction and review of current methods". *IEEE Trans. on Patt. Anal. and Mach. Intel. (PAMI).* 43(Nov.): 3964–3979.

Köpüklü, O., N. Kose, A. Gunduz, and G. Rigoll. (2019). "Resource efficient 3D convolutional neural networks". In: *IEEE/CVF Int. Conf. on Comp. Vision Workshop (ICCVW).* 1910–1919.

Korkmaz, C. (2021). *Multi-Model and Multi-Stage Learned Image Super-Resolution.* MS Thesis, Koç University, Istanbul, Turkey.

Krizhevsky, A., I. Sutskever, and G. E. Hinton. (2012). "ImageNet classification with deep convolutional neural networks". In: *Advances in Neural Information Processing Systems (NIPS).* 1097–1105.

Lai, W.-S., J.-B. Huang, N. Ahuja, and M.-H. Yang. (2017). "Deep Laplacian pyramid networks for fast and accurate super-resolution". In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR).*

LeCun, Y., B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. D. Jackel. (1989). "Backpropagation applied to handwritten zip code recognition". *Neural computation.* 1(4): 541–551.

Ledig, C., L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. (2017). "Photo-realistic single image super-resolution using a generative adversarial network". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR).*

Lee, S., M. Choi, and K. M. Lee. (2021). "DynaVSR: Dynamic adaptive blind video super-resolution". In: *IEEE/CVF Winter Conf. on Appl. of Comp. Vision.* 2093–2102.

Li, F., H. Bai, and Y. Zhao. (2020). "Learning a Deep Dual Attention Network for Video Super-Resolution". *IEEE Trans. on Image Proc.* 29: 4474–4488.

Liang, J., J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte. (2021). "SwinIR: Image Restoration Using Swin Transformer". In: *IEEE Int. Conf. on Computer Vision (ICCV) Workshops.*

Liao, R., X. Tao, R. Li, Z. Ma, and J. Jia. (2015). "Video superresolution via deep draft-ensemble learning". In: *IEEE/CVF Int. Conf. on Comp. Vision (ICCV).* 531–539.

Lim, B., S. Son, H. Kim, S. Nah, and K. M. Lee. (2017). "Enhanced deep residual networks for single image super-resolution". In: *IEEE/CVF Conf. on Comp. Vis. Patt. Recog. (CVPR) Workshop.*

Lim, B. and K. M. Lee. (2017). "Deep recurrent Resnet for video super-resolution". In: *Proc. of APSIPA Annual Summit and Conf.*

Liu, A. *et al.* (2021a). "Blind Image Super-Resolution: A Survey and Beyond". arXiv: 2107.03055 [cs.CV].

Liu, C., H.-Y. Shum, and W. Freeman. (2007). "Face hallucination: Theory and practice". *Int. J. Comput. Vision.* 75(1): 115–134.

Liu, D., Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, X. Wang, and T. Huang. (2018). "Learning temporal dynamics for video super-resolution: A deep learning approach". *IEEE Trans. on Image Processing.* 27(7): 3432–3445.

Liu, H. *et al.* (2020). "Video super resolution based on deep learning: A comprehensive survey". arXiv: 2007.12928 [cs.CV].

Liu, X., J. van de Weijer, and A. D. Bagdanov. (2017). "RankIQA: Learning from Rankings for No-reference Image Quality Assessment". In: *Int. Conf. on Comp. Vision (ICCV).*

Liu, Z., H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong, F. Wei, and B. Guo. (2022). "Swin Transformer V2: Scaling Up Capacity and Resolution". In: *IEEE/CVF Int. Conf. on Computer Vision and Pattern Recognition (CVPR).*

Liu, Z., Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. (2021b). "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows". In: *IEEE/CVF Int. Conf. on Computer Vision (ICCV).*

Lucas, A., S. L. Tapia, R. Molina, and A. K. Katsaggelos. (2019). "Generative Adversarial Networks and Perceptual Losses for Video Super-Resolution". *IEEE Trans. on Image Proc.* 28(7): 3312–3327.

Lugmayr, A. *et al.* (2019). "AIM 2019 Challenge on real-world image super-resolution: Methods and results". In: *Int. Conf. Comp. Vision (ICCV) Workshops.*

Lugmayr, A. *et al.* (2020a). "NTIRE 2020 Challenge on Real-World Image Super-Resolution: Methods and Results". arXiv: 2005.01996 [eess.IV].

Lugmayr, A., M. Danelljan, L. V. Gool, and R. Timofte. (2020b). "SRFlow: Learning the super-resolution space with onrmalizing flow". In: *Euro. Conf. Comp. Vision (ECCV)*.

Luo, Z., Y. Huang, S. Li, L. Wang, and T. Tan. (2020). "Unfolding the alternating optimization for blind super resolution". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 34.

Ma, C. *et al.* (2017). "Learning a no-reference quality metric for single-image super-resolution". *Comp. Vision and Image Understand (CVIU)*.

Makansi, O., E. Ilg, and T. Brox. (2017). "End-to-end learning of video super-resolution with motion compensation". In: *German Conf. on Pattern Recognition (GCPR)*. Basel, Switzerland.

Manasa, K. and S. S. Channappayya. (2016). "An optical flow-based full reference video quality assessment algorithm". *IEEE Trans. Image Proc.* 25(June): 2480–2492.

Mannos, J. and D. Sakrison. (1974). "The effects of a visual fidelity criterion on the encoding of images". *IEEE Trans. Inform. Theory*. 20(4): 525–536.

McCulloch, W. and W. Pitts. (1943). "A logical calculus of the ideas immanent in nervous activity". *Bulletin of Mathematical Biophysics*. 5: 115–133.

Minsky, M. and S. Papert. (1969). *Perceptrons : An Introduction to Computational Geometry*. Cambridge, Mass., USA: M.I.T. Press.

Mittal, A., A. K. Moorthy, and A. C. Bovik. (2012). "No-Reference Image Quality Assessment in the Spatial Domain". *IEEE Trans. on Image Processing*. 21(12): 4695–4708.

Mittal, A., R. Soundararajan, and A. C. Bovik. (2013). "Making a completely blind image quality analyzer". *IEEE Signal Processing Letters*. 20(3): 209–212.

Monga, V., Y. Li, and Y. C. Eldar. (2021). "Algorithm unrolling". *IEEE Signal Processing Magazine*. Mar.: 18–44.

Nah, S. *et al.* (2019a). "NTIRE 2019 Challenge on Video Deblurring and Super-Resolution: Dataset and Study". In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR) Workshops*.

Nah, S. *et al.* (2019b). "NTIRE 2019 Challenge on Video Super-Resolution: Methods and Results". In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR) Workshops*.

Oh, K.-S. and K. Jung. (2004). "GPU implementation of neural networks". *Pattern Recognition*. 37(6): 1311–1314.

Pan, J., H. Bai, J. Dong, J. Zhang, and J. Tang. (2021). "Deep blind video super-resolution". In: *IEEE/CVF Int. Conf. on Comp. Vision (ICCV)*. 4811–4820.

Papamakarios, G., E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan. (2021). "Normalizing flows for probabilistic modeling and inference". *Jou. of Machine Learning Research*. 22(Nov.): 1–64.

Papyan, V., Y. Romano, J. Sulam, and M. Elad. (2018). "Theoretical foundations of deep learning via sparse representations". *IEEE Signal Processing Magazine*: 72–89.

Patti, A. J., M. I. Sezan, and A. M. Tekalp. (1997). "Superresolution video reconstruction with arbitrary sampling lattices and nonzero aperture time". *IEEE Trans. on Image Processing*. 6(8): 1064–1076.

Pérez-Pellitero, E., M. S. M. Sajjadi, M. Hirsch, and B. Schölkopf. (2019). "Perceptual video super-resolution with enhanced temporal consistency". *arXiv preprint arXiv:1807.07930*.

Pérez-Pellitero, E., M. S. M. Sajjadi, M. Hirsch, and B. Schölkopf. (2018). "Photorealistic video super-resolution". In: *ECCV Workshops (PIRM)*.

Ponomarenko, N., F. Silvestri, K. Egiazarian, M. Carli, J. Astola, and V. Lukin. (2007). "On Between-coefficient Contrast Masking of DCT Basis Functions". In: *Int. Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM), Scottsdale, Arizona*.

Prashnani, E., H. Cai, Y. Mostofi, and P. Sen. (2018). "PieAPP: Perceptual image-error assessment through pairwise preference". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*.

Rad, M. S., T. Yu, C. Musat, H. K. Ekenel, B. Bozorgtabar, and J.-P. Thiran. (2021). "Benefiting from Bicubically Down-Sampled Images for Learning Real-World Image Super-Resolution". In: *IEEE/CVF WACV*.

Ramachandran, P., N. Parmar, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens. (2019). "Stand-Alone Self-Attention in Vision Models". In: *(NeurIPS)*.

Ranftl, R., A. Bochkovskiy, and V. Koltun. (2021). "Vision Transformers for Dense Prediction". In: *IEEE Int. Conf. on Computer Vision (ICCV)*.

Rezende, D. J. and S. Mohamed. (2015). "Variational inference with normalizing flows". In: *Int. Conf. on Machine Learning (ICML)*. Vol. 37. 1530–1538.

Rippel, O. and R. P. Adams. (2013). "High-dimensional probability estimation with deep density models". arXiv: 1302.5125 [cs.CV].

Ronneberger, O., P. Fischer, and T. Brox. (2015). "U-Net: Convolutional networks for biomedical image segmentation". In: *Medical Image Computing and Comp. Assisted Interven. (MICCAI)*. Vol. 9351. 234–241.

Rosenblatt, F. (1958). "The perceptron: A probabilistic model for information storage and organization in the brain". *Psychological Review*. 65(6): 386–388.

Rumelhart, D., G. Hinton, and R. Williams. (1986). "Learning representation by back-propagating errors". *Nature*. 323: 533–536.

Sajjadi, M. S., R. Vemulapalli, and M. Brown. (2018). "Frame-recurrent video super-resolution". In: *IEEE/CVF Conf. on Comp. Vis. Patt. Recog. (CVPR)*.

Salimans, T., I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. (2016). "Improved Techniques for Training GANs". In: *Advances in Neural Information Processing Systems (NeurIPS)*.

Seshadrinathan, K. and A. C. Bovik. (2010). "Motion tuned spatio-temporal quality assessment of natural videos". *IEEE Trans. on Image Processing*. 19(2): 335–350.

Sheikh, H. R. and A. C. Bovik. (2006). "Image information and visual quality". *IEEE Trans. on Image Processing*. 15(2): 430–444.

Sheikh, H. R., A. C. Bovik, and G. de Veciana. (2005). "An information fidelity criterion for image quality assessment using natural scene statistics". *IEEE Trans. on Image Processing*. 14(12): 2117–2128.

Shen, Z., M. Zhang, H. Zhao, S. Yi, and H. Li. (2021). "Efficient attention: Attention with linear complexities". In: *IEEE/CVF Winter Conf. on Applications of Computer Vision (WACV)*. 3531–3539.

Shi, W., J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. (2016). "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network". In: *IEEE/CVF Conf. on Comp. Vis. Patt. Recog. (CVPR)*. 1874–1883.

Shi, X., Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo. (2015). "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting". In: *Int. Conf. on Neural Information Processing Systems*. Vol. 1. *NIPS'15*. Montreal, Canada: MIT Press. 802–810.

Shocher, A., N. Cohen, and M. Irani. (2018). "'Zero-Shot' Super-Resolution using Deep Internal Learning". In: *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Srivastava, R. K., K. Greff, and J. Schmidhuber. (2015). "Training very deep networks". In: *Int. Conf. on Neural Information Processing Systems (NIPS)*. Vol. 2. 2377–2385.

Su, D., H. Wang, L. Jin, X. Sun, and X. Peng. (2020). "Local-global fusion network for video super-resolution". *IEEE Access*. 8(Sept.): 172443–172456.

Tao, X., H. Gao, R. Liao, J. Wang, and J. Jia. (2017). "Detail-revealing deep video superresolution". In: *IEEE/CVF Int. Conf. on Comp. Vision (ICCV)*.

Tekalp, A. M. (2015). *Digital Video Processing*. 2nd. USA: Prentice Hall. ISBN: 0133991008.

Tekalp, A. M. and I. Sezan. (1990). "Quantitative analysis of artifacts in linear space-invariant image restoration". *Multidimensional Systems and Signal Processing*. 1: 143–177.

Tian, Y., Y. Zhang, Y. Fu, and C. Xu. (2020). "TDAN: Temporally deformable alignment network for video super-resolution". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recognition (CVPR)*. 3360–3369.

Tikhonov, A. and V. Arsenin. (1977). *Solution of Ill-posed Problems*. Winston.

Timofte, R. *et al.* (2017). "NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshops*.

Timofte, R. *et al.* (2018). "NTIRE 2018 Challenge on single image superresolution: Methods and results". In: *IEEE/CVF Conf. on Comp. Vision and Pattern Recog. (CVPR) Workshops*.

Timofte, R., R. Rothe, and L. J. V. Gool. (2016). "Seven ways to improve example-based single image super resolution". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR)*.

Tong, T., G. Li, X. Liu, and Q. Gao. (2017). "Image super-resolution using dense skip connections". In: *Int. Conf. on Comp. Vision (ICCV)*.

Tran, D., L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. (2015). "Learning spatiotemporal features with 3D convolutional networks". In: *Int. Conf. Comp. Vision (ICCV)*. 4489–4497.

Trussell, H. J. and R. Civanlar. (1984). "The feasible solution in signal restoration". *IEEE Trans. on Acoustics, Speech, and Signal Processing*. 32(2): 201–212.

Ulyanov, D., A. Vedaldi, and V. Lempitsky. (2020). "Deep image prior". *Int. Jour. of Computer Vision*. 128(Mar.): 1867–1888.

Unterthiner, T., S. van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly. (2019). "FVD : A new metric for video generation". In: *Int. Conf. Learn. Represent. (ICLR) Workshop on Deep Generative Models for Highly Structured Data*.

Wang, X., K. C. Chan, K. Yu, C. Dong, and C. Change Loy. (2019). "EDVR: Video restoration with enhanced deformable convolutional networks". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshops*.

Wang, X., R. Girshick, A. Gupta, and K. He. (2018a). "Non-local neural networks". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*. 7794–7803.

Wang, X., K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy. (2018b). "ESRGAN: Enhanced super-resolution generative adversarial networks". In: *European Conf. Comp. Vision (ECCV) Workshop*.

Wang, Z. and A. C. Bovik. (2009). "Mean squared error: Love it or leave it?" *IEEE Signal Processing Magazine*. 26(1): 98–117.

Wang, Z., A. Bovik, H. Sheikh, and E. Simoncelli. (2004). "Image quality assessment: from error visibility to structural similarity". *IEEE Trans. on Image Processing*. 13(4): 600–612.

Wang, Z., J. Chen, and S. C. H. Hoi. (2021). "Deep learning for image super-resolution: A survey". *IEEE Trans. on Patt. Anal. and Mach. Intel. (PAMI)*. 43(10): 3365–3387.

Wang, Z., E. Simoncelli, and A. Bovik. (2003). "Multiscale Structural Similarity for Image Quality Assessment". In: *Asilomar Conf. on Signals, Systems, and Computers, Pacific Grove, CA, USA*. 1398–1402.

Wang, Z., X. Cun, J. Bao, and J. Liu. (2022). "Uformer: A General U-Shaped Transformer for Image Restoration". In: *IEEE/CVF Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Werbos, P. J. (1990). "Backpropagation through time: What it does and how to do it". *Proceedings of the IEEE*. 78(10): 1550–1560.

Winkler, C., D. Worrall, E. Hoogeboom, and M. Welling. (2019). "Learning likelihoods with conditional normalizing flows".

Xie, Y., E. Franz, M. Chu, and N. Thuerey. (2018). "TempoGAN: A temporally coherent, volumetric GAN for super-resolution fluid flow". *ACM Trans. on Graphics (TOG)*. 37(4): 1–15.

Xu, Y.-S., S.-Y. R. Tseng, Y. Tseng, H.-K. Kuo, and Y.-M. Tsai. (2020). "Unified dynamic convolutional network for super-resolution with variational degradations". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*. 12 496–12 505.

Xue, T., B. Chen, J. Wu, D. Wei, and W. T. Freeman. (2019). "Video Enhancement with Task-Oriented Flow". *Int. Jour. of Computer Vision (IJCV)*. 127(8): 1106–1125.

Yang, F., H. Yang, J. Fu, H. Lu, and B. Guo. (2020). "Learning Texture Transformer Network for Image Super-Resolution". In: *CVPR*.

Yang, W., J. Feng, G. Xie, J. L. Liu, Z. Guo, and S. Yan. (2018). "Video super-resolution based on spatial-temporal recurrent residual networks". *Computer Vision and Image Understanding*. 168: 79–92.

Yang, X., W. Xiang, H. Zeng, and L. Zhang. (2021). "Real-world video super-resolution: A benchmark dataset and a decomposition based learning scheme". In: *IEEE/CVF Int. Conf. on Comp. Vision (ICCV) Workshop*. 4781–4790.

Ye, P. and D. Doermann. (2011). "No-reference image quality assessment using visual codebook". In: *IEEE Int. Conf. Image Process. (ICIP)*.

Ying, X., L. Wang, Y. Wang, W. Sheng, W. An, and Y. Guo. (2020). "Deformable 3D convolution for video super-resolution". *IEEE Signal Processing Letters*. 27: 1500–1504.

Yu, J., Y. Fan, and T. Huang. (2019). "Wide activation for efficient image and video super-resolution". In: *British Mach. Vision Conf., Cardiff Univ.*

Yuan, Y., S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. Lin. (2018). "Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshops*.

Zhang, H., M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. (2018a). "mixup: Beyond Empirical Risk Minimization". In: *Int. Conf. on Learning Representations (ICLR)*.

Zhang, K. *et al.* (2020a). "NTIRE 2020 Challenge on perceptual extreme super-resolution: Methods and results". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR) Workshops*.

Zhang, K., Y. Li, W. Zuo, L. Zhang, L. van Gool, and R. Timofte. (2021). "Plug-and-play image restoration with deep denoiser prior". *IEEE Trans. on Patt. Anal. and Mach. Intel. (PAMI)*. June.

Zhang, K., W. Zuo, and L. Zhang. (2018b). "Learning a single convolutional super-resolution network for multiple degradations". In: *EEE/CVF Conf. CVPR*.

Zhang, K., L. van Gool, and R. Timofte. (2020b). "Deep unfolding network for image super-resolution". In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*.

Zhang, K., W. Zuo, S. Gu, and L. Zhang. (2017). "Learning deep CNN denoiser prior for image restoration". In: *IEEE Conf. on Comp. Vision and Patt. Recog. (CVPR)*. 3929–3938.

Zhang, K., M. Sun, T. X. Han, X. Yuan, L. Guo, and T. Liu. (2018c). "Residual Networks of Residual Networks: Multilevel Residual Networks". *IEEE Trans. Circuits and Systems for Video Tech.* 28(6): 1303–1314.

Zhang, L., L. Zhang, and A. C. Bovik. (2015). “A feature-enriched completely blind image quality evaluator”. *IEEE Trans. on Image Processing*. 24(8): 2579–2591.

Zhang, R., P. Isola, A. A. Efros, E. Shechtman, and O. Wang. (2018d). “The unreasonable effectiveness of deep features as a perceptual metric”. In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog.(CVPR)*. 586–595. DOI: 10.1109/CVPR.2018.00068.

Zhang, W., Y. Liu, C. Dong, and Y. Qiao. (2019). “RankSRGAN: Generative adversarial networks with ranker for image super-resolution”. In: *Int. Conf. on Comp. Vision (ICCV)*.

Zhang, Y., K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu. (2018e). “Image Super-Resolution Using Very Deep Residual Channel Attention Networks”. In: *IEEE/CVF ECCV*.

Zhang, Y., Y. Tian, Y. Kong, B. Zhong, and Y. Fu. (2018f). “Residual dense network for image super-resolution”. In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*. 2472–2481.

Zhua, H., L. Lia, J. Wua, W. Donga, and G. Shia. (2020). “MetaIQA: Deep Meta-learning for No-Reference Image Quality Assessment”. In: *IEEE/CVF Conf. on Comp. Vision and Patt. Recog. (CVPR)*.

View publication stats