Intelligent Super Resolution Image Generation: Survey of Algorithms

Srishti Aryavir*, Prof. M.S. Prasad, Dr. Shivani Verma

Amity Institute of Space Science and Technology,
Amity University, AUUP

Submitted: 15-01-2022 Revised: 23-01-2022 Accepted: 25-01-2022

ABSTRACT—Super Resolution is a process to obtain high resolution images from given low resolution images. For decades, super resolution has been a topic of interest among researchers focusing this problem in two domains, Single Image Super Resolution (SISR) and Multiple Image Super Resolution (MISR). It has found its practical usage in fields like medical image processing, surveillance, and media. A larger number of super resolution algorithms have been developed by classical and deep learning methods. This paper aims to do a study on the existing Single Image Super Resolution algorithms developed on using Neural Networks architectures.

Keywords—Super Resolution, Neural Networks, Machine Learning, CNN, Image processing

I. INTRODUCTION

Super Resolution (SR) is a method to convert a given low resolution (LR) image with coarse details to a high resolution (HR) image with better quality and finer details. This can be done for a single image in Single Super Resolution (SISR) and for multiple images in Multi Image Super Resolution (MISR). SR is an ill posed inverse problem, i.e., instead of a single solution there exists multiple solutions for the same LR image. Other than this complexity of the problem increases with increasing scaling factor. Hence, over the decades, lot of research has been done in this field ranging from use of classical methods like interpolation to deep neural networks to form algorithms to resolve low resolution image to high resolution image. However, deep learning models have outperformed their classical counterparts’ algorithms. Hence this study is largely focussed on the modern Super Resolution image generation based on these deep learning algorithms.

The paper has been organised into five sections. Section II discusses the different SISR approaches in detail. Section III discusses quality metrics for efficiency of discussed algorithms. Section IV discusses the datasets used for training the network, Section V outlines the findings of the study.

II. SINGLE IMAGE SUPER RESOLUTION APPROACHES

The SISR deep learning approaches can be categorized in to Linear, Residual, Recursive, Attention based and Generative network models. Ref fig 1. In this section each network model has been discussed in detail.
1. **Linear Network Model**

Linear Network models are those neural network models that have their structure made up of a single path for signal flow and have no skip connections. They have several convolution layers stacked one on another. They are further differentiated based on the way upsampling operation takes place, that is PreUpsampling and PostUpsampling.

![Classification of Linear Network Models](Image)

**Figure 2:** Classification of Linear Network Models

1.1. **PreUpsampling:**

Models that initially upsample the LR input to match the HR output size before learning the hierarchical feature representations to generate output are termed as Pre Upsampling models. These usually employ bicubic interpolation for upsampling and are computationally expensive models.

1.1.1. **Super Resolution Convolution Neural Network (SRCNN)**

Super Resolution Convolution Neural Network (SRCNN) is a straightforward model that uses 3 convolution layers with each layer being followed by ReLu except the last layer. This model performs three main operations:

1. **Patch extraction and representation:** This operation involves extraction of patches from a low-resolution image \( Y \) by convolving the image by a set of filters. This leads to generation of feature maps for low resolution image which is expressed as:

\[
F_1(Y) = \max(0, W_1 \ast Y + B_1)
\]

where, \( W_1 \) and \( B_1 \) represent filters and biases respectively and ‘\( \ast \)’ denotes convolution operation. \( W_1 \) applies \( n_1 \) convolutions on image, and each convolution has a kernel size \( c \times f_1 \times f_1 \).

2. **Non-linear mapping:** This operation derives a feature map of high-resolution imagery by convolving filters of spatial support of 1x1, 3x3 or 5x5 on patch of the previously obtained feature map of low-resolution image. The operation on the second layer is expressed:

\[
F_2(Y) = \max(0, W_2 \ast F_1(Y) + B_2)
\]

where, \( W_2 \) contains \( n_2 \) filters of size \( n_1 \times f_2 \times f_2 \) and \( B_2 \) is \( n_2 \) dimensional.

3. **Reconstruction:** This operation aggregates the high dimensional patch-wise representations to generate the final high-resolution image that is expected to be similar to the ground truth \( X \) using convolution layer as below:

\[
F(Y) = W_3 \ast F_2(Y) + B_3
\]

where \( W_3 \) corresponds to \( c \) filters of size \( n_2 \times f_3 \times f_3 \) and \( B_3 \) is a \( c \)-dimensional vector.

The SRCNN model minimizes the difference between reconstructed output HR images and ground truth HR images using Mean Squared Error loss function.

Dong et al. in [1] have used a relatively small dataset consisting of 91 images and a large training dataset that consists of 395,909 images from the ILSVRC 2013 ImageNet. In their study, they have highlighted that training time on ImageNet is same as that on 91-image dataset since the number of backpropagations are same and also a higher average PSNR (Peak Signal to Noise ratio) is achieved on ImageNet data than 91-image dataset highlighting that performance enhances on increasing training set.

1.1.2. **Very Dense Super Resolution (VDSR)**

Kim et al. [2] found following limitations SRCNN model:

1. **SRCNN model has only 3 layers leading to a small receptive field of 13x13.**
2. **It uses learning rate of \( 10^{-5} \) which makes the convergence too slow.**
3. **It works for only a single scale.**

Hence, they proposed Very Dense Super Resolution (VDSR) that is a deep CNN architecture that uses the VGG-net (Visual Geometry Group of Oxford University) model to obtain larger context information without pooling to keep pixel wise information.

To avoid slow convergence due to 20 layers and speed up the training process Kim et al. in [2] proposed two effective strategies:
1. Residual image learning: Instead of directly generating a HR image the model learns a residual mapping that generates a difference b/w HR and LR image.

2. Gradient Clipping: This is used to bound the gradient range while training. This handles the vanishing gradient problem generally encountered in back propagation.

   The model trains on a single convolution network to learn and handle multiple scale SR jointly. This joint scale learning helps the model to learn fractional or inter scale factors.

   The model consists of 20 convolution layers with each layer having 64 channels and 3x3 filters. Skip connections are present to learn residual image learning. No pooling is done to avoid dimension reduction. Mean Squared Error is used as the loss function.

   Kim et al. highlight that on running their model on the 91-image dataset and 4 Test datasets (Set5, Set14, B100, Urban100), performance increases as the depth of the network increases and that residual network learns faster and produces better results than non-residual network. They also observed that training multiple scales boosts the performance for larger scales.

1.1.3. Denoising Convolution Neural Network

   By employing residual learning from ResNet model and batch normalisation from Inception-v2 model Denoising Convolution Neural Network (DnCNN) [3], proposed by Zhang et al., implicitly removes the latent clean images in the hidden layers which helps to train a single DnCNN model to be able to handle several tasks such as Gaussian denoising, SISR and JPEG image deblocking.

   The DnCNN architecture comprises of three types of layers: (i) Conv+ReLu for the first layer, 64 filters of size 3x3xC where C is the number of image channels.

   (ii) Conv+BN+Relu for layers 2~ (D-1), 64 filters of size 3x3x64 and batch normalisation being added in between convolution and ReLu.

   (iii) Conv: for the last layer, C filters of size 3x3x64 to reconstruct the output.

   The input of DnCNN model is a noisy observation. Residual learning is hence adopted to train a residual mapping \( R(Y) \) which is formally derived from the average MSE loss function between residual images and estimated ones.

   DnCNN aims to reduce boundary artifacts that are a result of a low-level vision application by zero padding before the convolution to make sure that each filter map of the middle layers has same size as input image.

1.2. Late Upsampling (Post Upsampling)

   Early upsampling is computationally expensive as the network has to deal with larger sized inputs. Late upsampling SR tries to solve this by doing feature extraction in the lower resolution space, then doing upsampling only at the end, therefore significantly reducing computation. Also, instead of using simple bicubic interpolation for upsampling, a learned upsampling in the form of deconvolution/sub-pixel convolution is used, thus making the network trainable end-to-end.

1.2.1. Fast Super Resolution Convolution Neural Network

   Fast Super Resolution Convolution Neural Network (FSRCNN) [4] aims to improve over CNN on speed and accuracy.

   The architecture is quite simple as it consists of 4 convolution layers and 1 deconvolution layer. The 4 convolution layers perform the following tasks respectively:
1. Feature Extraction: This layer is like the feature extraction layer of SRCNN. The only difference is that the bicubic interpolation patch of SRCNN is replaced by the original patch in FSRCNN.

2. Shrinking: This layer reduces the dimension of feature maps by using a filter of size 1x1 from d to s where s << d.

3. Non-Linear Mapping: This layer helps to learn the non-linear functions and hence as a strong role in performance. The size of the filter is set to 3 and the number of channels is kept same as previous layer.

4. Expanding: This layer is an inverse of the shrinking step. It uses 1x1 convolution to increase the number of feature maps from s to d.

Final part is upsampling, carried out by deconvolution layer of 9x9 filters and aggregating it to obtain a HR image. Parametric ReLu is used instead of ReLu and loss function is same as that of SRCNN, i.e., MSE.

1.2.2. Efficient Sub Pixel Convolution Neural Network (ESPCN)

ESPCN is a fast SR approach for both images and videos. Proposed by Shi et al. [5], performs feature extraction in LR space. After features are extracted ESPCN uses sub pixel convolution layer at the very end to aggregate LR feature map and simultaneously perform projections to high dimensional space to reconstruct HR image. Sub Pixel convolution converts depth to space. For the architecture this model suppose has H layers for the network then for the first H-1 layers, the input LR images goes through f_h x f_h convolutions and obtain n_{h-1} feature maps. An efficient sub pixel convolution is performed at the last layer to obtain a HR image as output. A 1/2 loss is used to train overall network making ESPCN a competitive SR algorithm that has a high speed of over 26-33 frames per sec, which is useful for time critical missions such as live video recording.

2. Residual Network Models

To avoid vanishing gradient and to make deep networks feasible to design, residual networks unlike linear networks use skip connections. By this approach, the model learns residue, i.e., high frequencies between input images and ground truth. Residual models are divided on the basis of number of stages into: Single Stage Residual Nets and Multi Stage Residual nets.
2.1. Single Stage Residual Nets
These are Residual network models composed of a single network.

2.1.1. SRResNet
SRResNet is a basic model that uses ResNet architecture to perform SR. Only difference in the SRResNet model is the absence of ReLu layer after a residual block.

2.1.2. Enhanced Deep Super Resolution
Enhanced Deep Super Resolution (EDSR) attains SR using modified SRResNet architecture. Key features of EDSR model are the removal of batch normalisation (BN) layer from each residual block. Lim et al. states in [6] that BN normalizes the input, thus limiting the range of the network; removal of BN results in an improvement in accuracy. The BN layers also consume memory, and removing them leads to up to a 40% memory reduction, making the network training more efficient.

EDSR like VDSR model extends its single scale approach to multi scale by proposing Multi scale Deep SR (MDSR). EDSR uses $l_1$ loss function to train the model.

2.1.3. Cascading Residual Network (CARN)
Cascading Residual Network (CARN) model is based on ResNet. The major difference between the two is the presence of local and global cascading blocks instead of residual blocks. The output of intermediary layers is cascaded into higher layers which finally are converged on a single 1x1 convolution layer. These intermediary layers are the global cascading blocks that intrinsically have local cascading connections. Local cascading blocks are almost alike global cascading blocks except that unit blocks are simple residual blocks resulting into effective transformation of information.

Ahn et al. in their study [7] used 64x64 patches from Berkeley Segmented Dataset (BSD) and DIV2k dataset with data augmentation applying $l_1$ loss function to train the model and Adam as the optimizer with learning rate of $10^{-4}$ which is halved after every 4x10^5 steps.

2.2. Multi Stage Residual Nets
Multi stage residual networks are composed of multiple subnets that are generally trained in a successive fashion. The first subnet usually predicts the coarse features while other subnets improve the initial predictions.

2.2.1. FormResNet
Proposed by Jiao et al. [8] FormResNet is a deep CNN to tackle image restoration and SR. The model is composed of two networks, similar to DnCNN, however have a different loss function. The first network is called Formatting layer which aims to reduce corruption on the input image as it makes residual map focus more on image details instead of random distributed noise. This network incorporates Euclidean and Perceptual loss. The second network is DiffResNet which is similar to
DnCNN and learns the structured regions. Input from previous layer is fed in this layer.

2.2.2. Balanced Two-Stage Residual Networks

Balanced Two-Stage Residual Networks (BTSRN) by Fan et al. in [9] proposed a model that mainly contains two stages: low resolution (LR) stage and a high resolution (HR) stage. In these stages, the residual network has 6 and 4 residual blocks respectively. The two stages are connected via upsampling layers. The low-resolution stage has smaller feature maps hence receptive fields are extended to capture enough spatial context and high-level information with stacked convolution layers. In the high-resolution stages, larger feature maps contain larger information and are more correlated to output images. The number of blocks defines the accuracy and computational performance of the BTSRN.

For the NTIRE 2017 challenge, the model was trained on DIV2K dataset. Optimization was done via Adam with initial learning rate of 0.001 which was exponentially decreased after each iteration by a factor of 0.6 and $l_2$ was used as loss function to train the data.

2.2.3. Residual Encoder-Decoder Networks

Very deep Residual Encoder-Decoder Networks (RED-Net) proposed by Mao et al. in [10] is a model mainly consisting of chain of convolution layers and symmetric deconvolution layers. The convolution layers extract features while denoising the corruptions. The subtle details of the image may be lost in the process which are recovered by the deconvolution layer. The skip connections between the convolution feature map to their corresponding deconvolution layer ensure that response from convolution layer to deconvolution layer are propagated directly, both in forward and backward fashion. The passed convolutional feature maps are summed to the deconvolutional feature maps element-wise and passed to the next layer after rectification. A ReLu layer is added after each convolution and deconvolution layer. $l_2$ norm is used as the loss function. Mao et al. used this model in an experiment where two networks re deployed having 20 and 30 layers respectively for image denoising, image super resolution, JPEG deblocking and image inpainting. Network with 30 layers comes out to be best performing architecture using MSE as loss function and on BSD dataset.

3. Recursive Networks

Recursive Networks use either recursively connected convolutional layers or recursively linked units to break down the harder SR problem into a set of easy to solve problems.

![Figure 6: Classification of Recursive Network Models](image)

3.1. Deeply-Recursive Convolutional Network

Deeply-Recursive Convolutional Network (DRCN) proposed by Kim et al. [11] is a model that solves SR using very deep recursive layers. It utilizes a very large context compared to previous SR algorithms with a single recursive layer and aims to improve the network by inclusion of recursive supervision and skip connections. The model consists of three sub-networks:

1. **Embedding net:** This network represents the input image as a set of feature maps. This information is then passed to inference nets.

2. **Inference net:** This is the main component that solves the SR task. A single recursive layer is used to analyse a large image region. Each recursion applies the same convolution followed by a ReLu layer. The receptive field increases with every recursion as the increasing size of convolution filters.

3. **Reconstruction net:** The feature maps from final layer recursive layer are transformed into original image space to obtain a HR image. This is done in reconstruction net. DRCN to resolve vanishing gradient through recursion-supervision and includes skip connections. These skip connections help to save network capacity for storing input signal during recursion and enables the exact copy of input signal to be used during target prediction.

3.2. Deep Recursive Residual Network

Deep Recursive Residual Network (DRRN) proposed by Tai et al. [12] is a deep CNN network upto 52 layers with residual blocks...
outperforming state-of-art approaches like SRCNN, FSRCNN, ESPCN, VDSR, DRCN and RED-Net. DRRN uses recursive learning which involves replication of basic skip connection blocks several times leading to formation of a multi path network block. It is made computationally inexpensive by sharing of parameters among these replicated blocks. The architecture is obtained by stacking multiple recurrent blocks and having SGD optimizer with gradient clipping for learning the parameters. MSE is used as the loss function.

3.3. Super Resolution Feedback Network

Super Resolution Feedback Network (SRFBN) proposed by Li et al [13] is a model that employs feedback mechanism on a recurrent architecture design. It works on the principle that information of a R image with coarse details as ground truth can help to convert a LR image into a better SR image through feedback. The sub network going through each iteration have following parts: LR Feature extraction Block (LRFB), Feedback Block (FB) and Reconstruction Block.

FB contains projection groups sequentially stacked with skip connections. Each projection group can project HR features (deconvolution) to LR ones (convolution). Iteratively, input signals are passed recursively to FB, which learns the residual signal due to existence of a global residual connection.

For cases where there exist multiple types of degradations in the LR images, SRFBN uses curriculum learning approach. In this, HR images with increasing complexity are given to the model as ground truth and is trained with $\ell_1$ loss function and 4 recursive iterations.

4. Attention Based Networks

All the previously discussed models give immense importance to all the spatial features, channels for SR tasks. However, many times it is more helpful to focus on a few features at a given layer. Attention based networks deep learning networks consider only a few features of importance.

4.1. SelNet

Proposed by Choi and Kim[14], SelNet introduces a novel selection unit (SU) that acts as an on-off switching controlling in handling non linearity better than ReLu. SU is a ReLu-1x1Conv-Sigmoid unit called selection module (SM) that controls feature map obtained from previous convolutional layer to be input to the next layer.

SelNet consists of 22 convolutional layers and SU after each layer. Residual learning and gradient clipping are used in SelNet to learn faster. $\ell_2$ loss is used as loss function.

4.2. Deep Residual Channel Attention Network (RCAN) proposed by Zhang et al. [15] is a deep CNN architecture with two novel ideas. First, a
residual-in-residual (RIR) structure that consists of global residual group having long skip connections and inbuilt residual blocks with short skip connections in these groups. Second, channel attention mechanism that uses filter that collapses hwx×cvector to 1×1×c dimensions that allow network to focus on selective feature maps that have greater importance and can model the relationships between feature maps efficiently. The network architecture consists of 4 parts: shallow feature extraction (1 Conv layer), RIR deep feature extraction, upscale module, and reconstruction part. RCAN uses $l_1$ loss function to train the network.

4.3. Densely Residual Laplacian Super Resolution

Proposed by Anwar and Barnes, Densely Residual Laplacian Super Resolution (DRLN) [16] has a hierarchical structure consisting of densely connected residual units. The authors propose the novel design of cascading over residual on the residual (CRIR) that consists of DenseResidual Laplacian module (DRLM) and diverse connection types to learn more accurate representations. Laplacian attention is also introduced to learn the inter and intra-level dependencies between the feature maps.

CRIR structure is a set of cascading blocks that has Medium Skip Connections (MSC), cascading feature concatenation and DRLM modules. Addition of MSC eases information flow across groups of DRLM whereas addition of long skip connections (LSC) helps information flow through cascading blocks. DRLM is made of three components: densely connected residual block units, compression unit and Laplacian pyramid attention. Laplacian attention learns critical features progressively in each DRLM by using global descriptor that captures the statistics of the entire image. To capture channel dependencies from retrieved global descriptor, gating approach employing ReLu and sigmoid function is used. Then these features are passed through Laplacian pyramid to learn critical features at different scales. Similar to RCAN, DRLN uses $l_1$ loss function to train and training parameters are same as RCAN.

5. Generative Adversarial Network (GAN) Models

GAN models work as a two-player game. The famous counterfeiter–police example is often used to describe GAN models where generator is the counterfeiter that makes fake paintings whereas discriminator is the police officer who catches counterfeiter’s fakes. To succeed in the game, the counterfeiter must make a painting real enough to fool the police officer.

The GAN model has a generator and a discriminator trained together. The generator has the task of generating SR images that are fed into discriminator. The discriminator then has to classify whether the input samples are real HR images or artificially super-resolved outputs. The loss function from discriminator is used to increase efficiency of discriminator at classifying the real and fake images and also to update generator to generate images with better perceptual quality to fool the discriminator.

5.1. Super Resolution Generative Adversarial Network

Super Resolution using a Generative Adversarial Network (SRGAN) [17] proposed by Ledig et al. proposes to obtain super-resolved outputs via adversarial objective functions.

The architecture is similar to GAN architecture consisting of a generator of B residual blocks having 64 filters of size 3x3, followed by batch normalisation and ParametricRelu as activation function and a discriminator with architecture similar to DC-GAN having 8 convolution layers with filters increasing 64 to 512 by a factor of 2 and size of 3x3 with LeakyRelu as activation function.

$I^{LR}$ is the low-resolution image of its high-resolution counterpart $I^{HR}$. In training, $I^{LR}$ is obtained by applying a Gaussian filter to $I^{HR}$ and down-sampling it by a factor of r. Hence, $I^{LR}$ is represented in size as $W \times H \times C$, C being color.

Figure 9: Classification of GAN Models
channels and \( I^{HR} \) by \( W \times H \times C \). The goal is to train a generating function \( G \) that estimates a HR image for its corresponding LR input image. To achieve this, a generator network is trained as a feed-forward CNN \( G_\theta \) parametrized by \( \theta = \{ W_{1x}, B_{1x} \} \) denoting weights and biases for L-layer deep network.

The key feature of their study is the multi-task loss function called Perceptual loss function which is the weighted sum of content loss \( (l_{cont}) \) and adversarial loss component \( (l_{gan}) \) as equation (4):

\[
l^{SR}_{MSE} = \frac{1}{x^2y^2} \sum_{x=1}^{W} \sum_{y=1}^{H} (I^{HR}_{x,y} - G_{\theta}(I^{LR}_{x,y}))^2
\]

\[
l^{SR}_{VGG/i-j} = \frac{1}{w_{i}h_{i}} \sum_{x=1}^{W} \sum_{y=1}^{H} (\phi_{ij}(I^{HR})_{x,y} - \phi_{ij}(G_{\theta_c}(I^{LR}))_{x,y})^2
\]

For the min-max game of generator and discriminator, adversarial loss, seen in equation (6) is used for \( n = 1, \ldots , N \) images:

\[
l^{SR}_{G} = \sum_{n=1}^{N} - \log D_{bd}(G_{\theta_c}(I^{LR}))
\]

where \( D_{bd}(G_{\theta_c}(I^{LR})) \) is the probability that reconstructed image \( G_{\theta_c}(I^{LR}) \) is a natural HR image.

This model gives a low PSNR value yet the images appear to be perceptually better hence to quantify the capability of their model, the authors introduced a new Mean Opinion Score (MOS) which indicates the bad or excellent quality of each super-resolved image attested by human raters.

5.2. EnhanceNet

EnhanceNet[8] by Sajjadi et al. proposed a model that focusses on texture instead of PSNR as image quality metric. Hence, the model takes a residual CNN with following loss functions besides the regular pixel level MSE:

(i) Perceptual loss: Instead of calculating MSE in spatial domain, the estimated image and ground truth are fed into VGG feature space and its MSE in the feature space is calculated.

2. Texture loss: In this loss function the correlation of the VGG feature space is calculated and the MSE over that is taken.

Since the model is a GAN model, adversarial loss is also calculated to train discriminator and subsequently improve performance of EnhanceNet. The authors show in their result that even though best PSNR is achieved when only pixel level loss is used, the output image is not realistic. Hence these additional losses lead to more realistic and perceptual output images. One limitation of this model is that it creates visual artifacts while super-resolving high texture regions.

5.3. Enhanced Super Resolution Generative Adversarial Network

Enhanced Super Resolution Generative Adversarial Network (ESRGAN) [19] proposed by Wang et al. is a model build upon SRGAN with difference being the absence of batch normalisation and replacement of original basic block with Residual-in-Residual Dense Block (RRDB). Removing batch normalisation increases performance and leads to reduction of computational complexity in PSNR oriented tasks. RRDB combines a multi-level residual net and dense connections making the architecture deeper and boosting the performance. ESRGAN also proposed Relativistic Discriminator, an enhanced discriminator. Instead of estimating the probability that the input image is real as done in standard discriminator, it tries to predict the probability that a real image is relatively more realistic than a fake one. The adversarial loss for generator contains information about both real and fake data. Hence, the generator benefits from gradients from both generated data and real data in adversarial training unlike in SRGAN where only generated data trains the generator.

III. QUALITY METRICS

For the calculating the efficiency of a network, SR algorithms are graded on the basis of quality metrics such as Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM)[27]. PSNR is the ratio between the maximum possible value of a signal and he power of distorting noise that effects its representation and is expressed in equation (8):

\[
\text{PSNR} = 20 \log_{10} \left( \frac{\text{MAX}}{\sqrt{\text{MSE}}} \right)
\]
Unlike PSNR it is based on visible structures in the image. Instead of using traditional error summation methods, the SSIM models image distortion as a combination of three factors that are loss of correlation, luminance distortion, and contrast distortion.

IV. DATASETS

In this section, above discussed algorithms have been tested on publicly available benchmark datasets which include Set5[20], Set14[21], BSD100[22], Urban100[23], DIV2K [24] and Manga109[25] and have been summarized in the table below.
V. CONCLUSION

A diverse set of approaches have been suggested in the paper. It has been noticed that (a.) GAN based approaches deliver visually better results while having low PSNR value, (b.) for higher magnification, the quality metrics is suboptimal, (c.) deeper networks help to achieve better results on the expense of computational complexities and (d.) residual networks have been major contributing factors in increasing the overall performance. These learnings can be kept in mind while designing a novel super resolution algorithm using deep learning networks.

ACKNOWLEDGMENT

I would like to thank Prof. M.S. Prasad, HOI, Amity Institute of Space Science & Technology and Prof. Shivani Verma, Assistant Prof, Amity Institute of Space Science & Technology for guiding me throughout my study.

REFERENCES


