IRGUN : Improved Residue based Gradual Up-Scaling Network for Single Image Super Resolution

Manoj Sharma, Rudrabha Mukhopadhyay, Avinash Upadhyay, Sriharsha Koundinya,
Ankit Shukla, Santanu Chaudhury.

CSIR- CEERI, Pilani
Rajasthan-333031, India
{mksnith, rudrabha, avinres, sriharsharaja, anktshkla.1307, schaudhury}@gmail.com

Abstract

Convolutional neural network based architectures have achieved decent perceptual quality super resolution on natural images for small scaling factors (2X and 4X). However, image super-resolution for large magnication factors (8X) is an extremely challenging problem for the computer vision community. In this paper, we propose a novel Improved Residual based Gradual Up-Scaling Network (IRGUN) to improve the quality of the super-resolved image for a large magnification factor. IRGUN has a Gradual Upsampling and Residue-based Enhancment Network (GUREN) which comprises of series of Up-scaling and Enhancement blocks (UEB) connected end-to-end and fine-tuned together to give a gradual magnification and enhancement. Due to the perceptual importance of the luminance in super-resolution, the model is trained on luminance (Y) channel of the YCbCr image. Whereas, the chrominance components (Cb and Cr) channel are up-scaled using bicubic interpolation and combined with super-resolved Y channel of the image, which is then converted to RGB. A cascaded 3D-RED architecture trained on RGB images is utilized to incorporate its inter-channel correlation. In addition to this, the training methodology is also presented in the paper. In the training procedure, the weights of the previous UEB are used in the next immediate UEB for faster and better convergence. Each UEB is trained on its respective scale by taking the output image of the previous UEB as input and corresponding HR image of the same scale as ground truth to the successive UEB. All the UEBs are then connected end-to-end and fine tuned. The IRGUN recovers fine details effectively at large (8X) magnification factors. The efficiency of IRGUN is presented on various benchmark datasets and at different magnification scales.

1. Introduction

Single image super-resolution (SISR) is a technique to construct a high-resolution (HR) image from its corresponding lower resolution version. Since a single low-resolution (LR) image has multiple possible higher resolution images, it is very difficult to reconstruct a high-quality HR image. The problem complicates when the scale of super-resolution increases. In recent years, deep learning based super-resolution architectures have performed reasonably well for lower scaling (2X and 4X) ratios. Most of the deep learning based super-resolution (SR) methods use bicubic interpolation to up-sample LR images as a pre-processing step [4, 5, 10–12, 21, 23, 24, 27, 32, 33, 35]. However, operation on these pre-processed images causes additional computational overhead and adds unwanted artifacts. To address these problems [1, 15, 18, 25, 28, 30] has incorporated convolutional transpose layer which reconstructs HR image in one up-scaling step. This sudden up-scaling to a higher magnification factor causes a lot of information loss, resulting in difficulty in training and the model is not able to reconstruct HR images efficiently. To overcome the issues due to sudden up-scaling, [14] has proposed Laplacian pyramid based super-resolution network(LapSRN) to gradually reconstruct the residuals of HR images at each pyramid level. LapSRN uses convolutional transpose layers for up-scaling between two SR scales. Zhao et.al. [38] has proposed a gradual up-scaling network(GUN) which uses multiple level up-sampling and convolutional layer to learn SR process for large scale. Though LapSRN [14] and GUN [38] were addressing the problem of large-scale SR by learning the process gradually in multiple levels, the results were not satisfactory due to which a need for an improved algorithm for 8X SR arises. In this paper, we propose an Improved Residual based Gradual Up-Scaling Network (IRGUN) to learn the mapping between LR and corresponding HR images. Y channel contains

2160-7516/18 $31.00 © 2018 IEEE
DOI 10.1109/CVPRW.2018.00128

most of the essential information required for SR. Hence, in contrast with earlier deep learning based frameworks [1,4,5,10–12,14,15,18,21,23–25,27,28,30,32,33,35,38], only the luminance part of the image is employed to simplify the learning process and to reduce computational overload. In the proposed framework we are using an upsampling network (U) to upscale the luminance (Y) of the image by a magnification factor of two. This is followed by a residual enhancement network (E), which further enhances the quality of the resultant image. Residue encoder-decoder (RED) based architecture is used in Enhancement Network (E). This up-scaling and enhancement network is treated as a block (UEB) which are connected end-to-end to give the required magnification. The number of UEBs required for a specific scaling can be obtained by the formula $log_2(X)$, where X is the magnification factor required. While repeating the UEBs, the trained weights of the previous UEBs are used for faster and better error convergence. These UEBs are further trained on their respective scales, by taking the resultant images of the previous UEB as input to the successive UEB and corresponding ground truth at the same scale. After reaching the desired scale of SR, these cascaded UEBs are then fine-tuned end-to-end. This network is named as Gradual Upsampling and Residual-based Enhancement Network (GUREN). The output of this network is then further improved by another E block cascaded to GUREN. Cb and Cr channels are simply up-scaled using bi-cubic interpolation. It is then added to the resultant image from the network to get the YCbCr image. This image is then converted to RGB color space. Since the training of the model is done on Y channel and CbCr channels are super-resolved using bicubic interpolation, its conversion to the RGB color space induces some artifacts. Removal of such artifacts and further enhancement of the resultant RGB image is done by training a 3D-RED enhancement network by learning the mapping between Ground Truth in RGB and Resultant RGB image with artifacts. The 3D convolution uses a 3D kernel which convolves in three dimensions, two spatial and one channel (which is taken as depth in this case), hence it preserves the related information between the channels. Instead of simple averaging, a 3D convolution operation adds up the response of all the corresponding pixels, hence it incorporates the co-related information. By using 3D-RED, we are taking advantage of the spatial co-relation in one channel as well as the inter-channel co-relation among R, G and B channels.

2. Related Work

2.1. Traditional Single Image Super Resolution

Reconstruction of HR images from LR images is always a challenging task. Traditional methods involving interpolation based algorithms such as linear, bicubic, bilinear, nearest-neighbour [9,16] causes aliasing, blurring and ringing effect. To address these problems, solutions like altering interpolated grid [3], edge prior knowledge [17], and edge sharpening [7,31], etc has been used. These methods efficiently removes the unnatural artifacts however, they are still not capable to recover finer details. Another approach to SISR is reconstruction-based algorithm such as gradient-based constraints [26,40], total variation regularizer [19,20], local texture constraint [37,39], deblurring-based models [6,8] etc. These methods are good for a small magnification ratio, however when we approach for a larger scale magnification its performance declines as the basic similarity constraint is implied on LR space. Reconstruction of missing information with known LR/HR example pair uses learning based methods such as neighbor embedding based methods [2], local self-exemplar methods and sparse representation based methods [34,36]. Sharp edges can be recovered by using these methods but it employs patch-wise optimization of weights which makes it computationally complex .

2.2. Using Convolutional Neural Networks (CNN)

Several CNN based SISR methods have been proposed in the recent years. Dong et al. [4] proposed a simple CNN based SISR method, which is further improved by Kim et al. [10,12] by using very deep CNNs. Methods such as sparse convolutional network [33], recursive convolutional network [12], combined deep and shallow CNN [32], deep residual network [5,21], bi-directional recurrent convolutional network [35], have also been proposed. In most of these methods, SISR is regarded as an image reconstruction problem. Till date CNN based SR methods have produced best state-of-the-art results. Two types of upsampling methods are used in these models. Some of the models, increases the scale of the input image using conventional methods such as interpolation etc [4,5,10–12,21,23,24,27,32,33,35], and then reconstruct the HR image from this input. However, in some of the models the increment in scale is also done using convolutional layers [1,15,18,25,28,30]. In GUN [38] and LapSRN [14] they have used gradual up-scaling to gradually reconstructs the HR image from the LR image.

2.3. Our Contributions

The main contributions of this paper are: 1. A novel gradual upsampling and residual based enhancement network (GUREN) which gradually up-scales the Y channel of a LR image in multiple steps with a small magnification ratio by an up-scaling network (U) followed by an Enhancement network (E) in one step. 2. Using a 3D-RED to improve the quality of the resultant RGB images by preserving the inter-channel co-relation among R,G and B channels. 3. Using trained weights from the previous UEB of GUREN

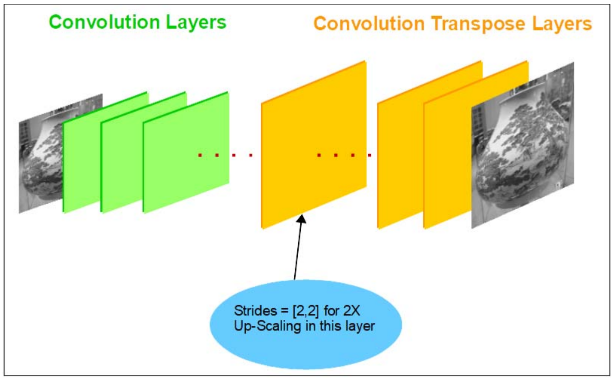

Figure 1. Block Diagram for Up-Scaling Network

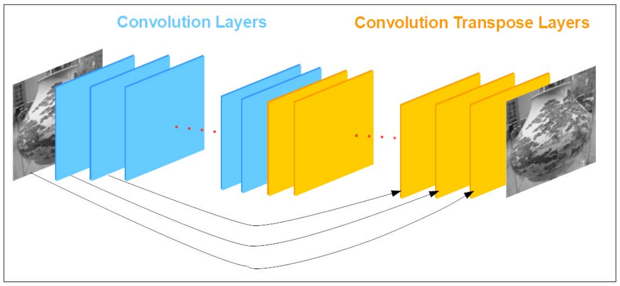

Figure 2. Block Diagram for Enhancement Network

in successive UEBs for better initialization of weights in the end-to-end network for large magnification ratio. 4. We show the effectiveness of the proposed architecture via experiments on different benchmark data-sets. The rest of the paper is organized as follows. In section III we explain the methodology, section IV covers the experiments and results analysis and in the last section we conclude our results.

3. Methodology

The proposed framework contains three sections: Gradual Up-scaling and Residual-based Enhancement Network (GUREN), Enhancement Network (EN) and a 3D CNN based Residual network (3D-RED). The final framework is given in Figure 4. GUREN consists of three Upsampling and Enhancement Blocks (UEB) each contributing 2X upsampling. An UEB consists of an U and an E block. U block consists of a three convolution layers and three convolution transpose layers connected to each other as shown in Figure 1. Every layer is followed by a Rectified Linear Unit (ReLU) layer. The first convolutional transpose layer of the U block have stride [2,2] to upsample the image by a scale factor of two. An E block is then cascaded with the U block. The E block consists of five convolution and five convolution transpose layers with skip connections as shown in Figure 2. This network is used to learn the residue between an up-scaled image and its corresponding higher resolution ground truth. The cascaded U and E blocks are then end-to-end fine-tuned i.e. the loss is back-propagated from the last layer of E block to the first layer of U block. Afterwards, weights of this UEB block which consists of weights of U block and E block are loaded into the consecutive UEB blocks which are again end-to-end fine-tuned for their respective scales. After individual training of these UEBs, the cascaded UEB network i.e. GUREN is end-to-end fine-tuned such that the loss from the output of last UEB network is used to optimize all the layers present in the GUREN network. GUREN is used to obtain 8X SR with 2X SR at every $(log_2 S)$ UEB where S is the scaling factor. The output from the GUREN is then fed to another EN network which is same as E. The GUREN and EN network is trained on the Y channel hence, the image is converted into YCbCr color space from RGB. The Cb and Cr channel of the image is up-scaled to 8X by using bicubic interpolation. Then they are concatenated with the super resolved Y channel output from the EN network. The YCbCr image is then converted back to RGB Color space. A 3D-RED network as shown in the Figure 3 is used to learn the inter-channel co-relation among R, G and B channels of the image. This network consists of five 3D convolution and five 3D convolution transpose layers with symmetric skip connections. A 3D kernel convolves in three dimensions, two spatial and one channel, thus incorporates the related information between the channels. Instead of simple averaging, a 3D convolution operation adds up the response of all the corresponding pixels, thereby preserving the co-related information.

3.1. Up-scaling Network

For training the up-scaling network, we have generated a dataset, where ground truth HR images at particular resolution scale are given as target and its corresponding LR images are given as inputs. The block diagram of the network is given in Figure 1. The mathematical representation of the network is explained as follows. Let Y_0 be the luminance part of LR image, $cbcr$ be the chrominance part of LR image and HR_{2X} be the corresponding ground-truth. With these as inputs, the convolutional feature extraction (k^{th} feature map) is given by,

$$f_1^k = U_1(Y_0) = max(0, Y_0 * W_1^k + B_1^k). \qquad (1)$$

Here W_1^k and B_1^k are the convolutional kernel and the bias of the k^{th} feature map respectively, here, $k = 1, 2, ..., K$, where K is the total number of feature maps and $'*'$ is the convolutional operator. subsequent convolutional layer features can be expressed similarly,

$$f_N^k = U_N(f_{N-1}^k) = max(0, f_{N-1}^k * W_N^k + B_N^k). \qquad (2)$$

where, N is the total number of convolutional layers. Further, we try to reconstruct HR image features from the LR

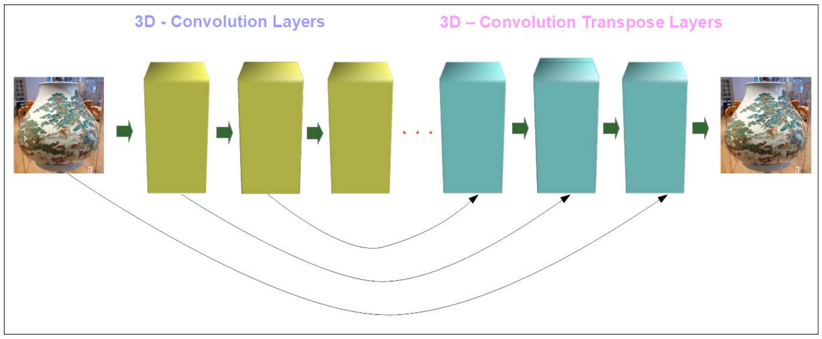

Figure 3. Block Diagram for 3D-RED network

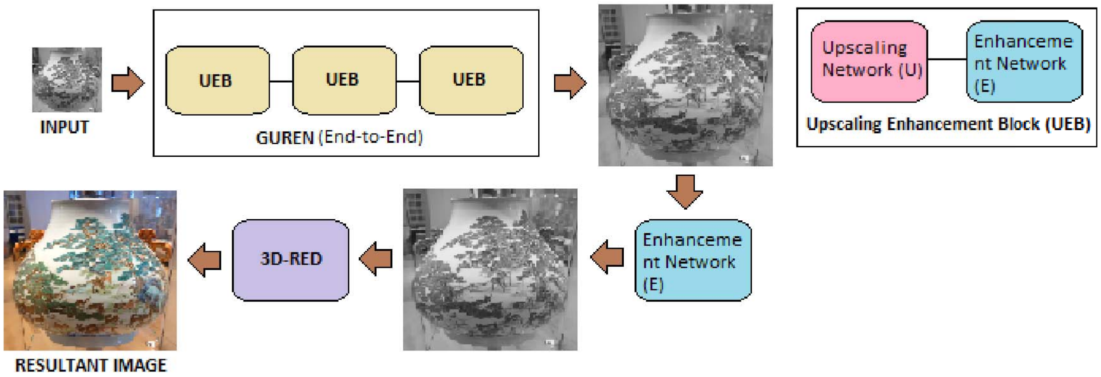

Figure 4. Block Diagram for IRGUN

image feature maps by using convolutional transpose layers.

$$f^k_{N+1} = U_{N+1}(f^k_N) = max(0, f^k_N * W'^k_{N+1} + B^k_{N+1}). \quad (3)$$

where, $'*'$ is the convolution transpose operator. In the first convolution transpose layer we take strides=2 and for the rest of the convolution transpose layers we take strides=1. Similarly other features are calculated by,

$$f^k_{2N-1} = U_{2N-1}(f^k_{2N-2}) = max(0, f^k_{2N-2} * W'^k_{2N-1} + B^k_{2N-1}). \quad (4)$$

The reconstruction of gradual up-scaled image (2X) is given by,

$$\begin{aligned} Y_0^{2X} &= f^k_{2N} = U_{2N}(f^k_{2N-1}) \\ &= max(0, \sum_k f^k_{2N-1} * W'^k_{2N} + B^k_{2N}). \end{aligned} \quad (5)$$

here, convolution transpose layer tries to reconstruct the HR image by minimizing the loss function

$$Loss = \frac{1}{P} \sum_{i=1}^{P} \parallel HR^i_{2x} - \mathcal{F}({Y_0}^i, \theta) \parallel^2_F \quad (6)$$

where, $i = 1, ..., P$ and P is the total number of images in database. After minimizing the loss by Adamboost optimizer, we obtain the trained weights.

3.2. Analysis of up-scaling network

Assuming that we have 2N layers. The following equations show the relation between input to the target output. $Y_0^{2X} = U_{2N}(f_{2N-1}) = U_{2N}(U_{2N-1}(f_{2N-2})) = U_{2N}.U_{2N-1}........U_1(Y_0)$

$$Y_0^{2X} = U(Y_0) \quad (7)$$

where, $U = U_{2N}.U_{2N-1}........U_1$

3.3. Enhancement network

The mathematical representation of the network is explained as follows. Here, the resultant images of upsampling network (U) Y_0^{2X} be the input image and the HR_{2X} be the corresponding ground-truth image. With these inputs the convolutional layers can be expressed as,

$$E_n(Y^{2X}_{n-1}) = f^k_n = max(0, Y^{2X}_{n-1} * W^k_n + B^k_n). \quad (8)$$

where, $n = 1, 2, ..., N$ Thereafter, we are try reconstruct the HR clean version by convolutional transpose layer.

$$f_m = f_m + f_{m-r} \quad (9)$$

here, $m = N + 1, N + 2, ...$ and $r = 2, 3, 4...$, respectively. now the reconstruction of gradual up-scaled image (2X) is

Table 1. IRGUN model specifications

Specification	IRGUN
Up-scaling blocks layers	6
Up-scaling blocks feature-map	32
Enhancement blocks layers	10
Enhancement blocks feature-map	32
3D-Enhancement blocks layers	10
3D-Enhancement blocks feature-map	16

given by,

$$Y_E^{2X} = E_{2N}(f_{2N-1}^k) = max(0, \sum_k f_{2N-1}^k * W_{2N}^{'k} + B_{2N}^k). \tag{10}$$

Here, convolution transpose layer tries to reconstruct HR images by minimizing the loss function given by,

$$Loss = \frac{1}{P}\sum_{i=1}^{P} \parallel HR_{2X}^i - \mathcal{F}({Y_0}^{2X,i}, \theta) \parallel_F^2 \tag{11}$$

where, P is the total number of images in database. After minimizing loss by Adamboost optimizer, we obtain the trained weights.

3.4. Analysis of enhancement network

Assuming that, we have 2N layers. The following equations show the relation between the input to the target output.

$$Y_E^{2X} = f_{2N} = E_{2N}(f_{2N-1}) \tag{12}$$

$$Y_E^{2X} = E_{2N}(f_1 + E_{2N-1}(f_{2N-2})) \tag{13}$$

$$Y_E^{2X} = E_{2N}(f_1 + E_{2N-1}(f_2 + E_{2N-2}(f_{2N-3}))) \tag{14}$$

and finally we can write,
$Y_E^{2X} = E_{2N}(f_1) + E_{2N}.E_{2N-1}(f_2) + E_{2N}.E_{2N-1}.E_{2N-2}(f_3 + ... + E_{2N}.E_{2N-1}.E_{2N-2}.....E_1(Y_0^{2X}))$ as, $f_1, f_2, f_3...$ are also a function of Y_0^{2X}

$$Y_E^{2X} = E(Y_0^{2X}) \tag{15}$$

here, $E = E_{2N}.E_1 + E_{2N}.E_{2N-1}.E_2.E_1 + ... + E_{2N}........E_1$ The equation above clearly explain the relation of Y_E^{2X} with input Y_0^{2X}.

3.5. 3D Enhancement network

Enhancement 3D network E_{3D} is an extension of enhancement $2D$ network to enhance color images. Here, we consider color channels as depth and use $3D$ convolutional kernel instead of a $2D$ convolutional kernel.

3.6. Training

The Y components of all the images were extracted and used subsequently for the proposed framework. The training was carried out on patches. Initially, we trained a single up-scaling residual block on the lowest scale i.e. on one eighth scale to one fourth scale. For this purpose, the ground truth is created by down-scaling an image to its quarter size and then patches of size 128 X 128 pixels were extracted. The input is created by down-scaling the image to its 1/8th size and then creating patches of 64 X 64 pixels. We trained this block by minimizing Mean square error (MSE). Hence, 2X super-resolution from 1/8th scale to 1/4th scale is trained on the first UEB. We trained our model with Adam optimizer [13]. Consecutive UEBs are initialized with trained weights from the first UEB. The input to the GUREN is created by down-scaling the image to its 1/8th size and the original image is considered as the ground truth for this network. All the down-scalings are done using bi-cubic interpolation. Patches of 16X16 pixels and 128X128 pixels are extracted from the images of the input set and the ground truth set respectively. This network is optimized by minimizing the mean square error between ground-truth and the resultant image from the model. The resultant images from GUREN are then passed through an EN network. For this purpose the resultant images and corresponding ground truth images are divided into patches of 128 X 128 dimensions. Weights of this EN network are initialized randomly. This cascaded EN network is then trained. The super-resolved images obtained from the EN network are concatenated with the Cb Cr channel of the respective image which is up-scaled using bicubic interpolation. The YCbCr image is then converted back to RGB. These RGB images are divided into patches of dimensions 128 X 128 X 3 which are then used as an input to 3D-RED. We consider the corresponding patches of the same dimensions from the high resolution RGB images as ground truth. For training on this network, the three-dimensional array of the image is converted to four dimensions by splitting every channel and then concatenating them on the fourth axis such that the RGB image acts like a monochrome video with three frames.

3.7. Testing

Although the network is trained on patches, it can perform 8X super resolution on images of any arbitrary dimension. For obtaining results, one needs to take a forward path inference on any lower resolution image. A lower resolution image in RGB color space should be initially converted into YCbCr. A copy of the image in RGB color space is also bi-cubically interpolated to its 8X resolution and then converted to YCbCr color space. The CbCr channels are extracted from this image and are stored separately. The Y channel information from the lower resolution image is fed

Table 2. Average PSNR and SSIM comparison of different Image SR algorithms on 8X scale for different datasets. Red shows highest while blue shows second highest

Dataset	**Scale**	**Bicubic**	-	**VDSR** [11]	-	**GUN** [38]	-	**RDN** [27]	-	**LapSRN** [14]	-	**IRGUN**	-
-	-	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Set5	8	24.39	0.657	25.72	0.711	25.99	0.713	26.10	0.730	26.14	0.738	**26.28**	**0.740**
Set14	8	23.19	0.568	24.21	0.609	24.23	0.610	24.39	0.614	24.44	0.623	**24.53**	**0.641**
BSD100	8	23.67	0.547	24.37	0.576	24.42	0.579	24.53	0.584	24.54	0.586	**24.61**	**0.591**
URBAN100	8	20.74	0.515	21.54	0.560	21.66	0.565	21.70	0.577	21.81	0.581	**21.89**	**0.594**
MANGA109	8	21.47	0.649	22.83	0.707	23.00	0.717	23.28	0.728	23.39	0.735	**23.43**	**0.748**
DIV2K Validation Dataset	8	23.82	0.549	23.94	0.607	24.36	0.609	24.47	0.628	24.63	0.645	**24.99**	**0.652**

Table 3. Average PSNR comparison of different Image SR algorithms on 4X scale for different datasets. Red shows highest while blue shows second highest

Dataset	**Scale**	**Bicubic**	**SRGAN** [15]	**VDSR** [11]	**GUN** [38]	**RDN** [27]	**EDSR** [28]	**LapSRN** [14]	**IRGUN**
Set5	4	28.42	29.28	31.35	31.50	31.58	32.62	31.54	**32.65**
Set14	4	26.10	26.02	28.03	28.04	28.29	28.94	28.19	**28.98**
BSD100	4	25.96	25.16	27.29	27.44	27.55	27.79	27.32	**28.01**
URBAN100	4	23.15	22.75	25.18	25.24	25.44	26.86	25.21	**25.48**
MANGA109	4	24.92	23.78	28.82	28.97	29.09	29.12	29.06	**29.22**
DIV2K Validation Dataset	4	27.32	25.81	28.22	28.67	28.92	28.99	28.98	**29.1**

to the GUREN. The output images of GUREN is then fed to the cascaded EN network to obtain the SR Y images. The CbCr channel which was extracted earlier is later concatenated with this image to form YCbCr image. The resultant YCbCr image is converted back to RGB. Furthermore this RGB image is fed to the 3D-RED to obtain the final results.

4. Experiments

In this section we discuss about the experiments that were performed using our framework. We evaluated the performance of our framework on various well known datasets. We have also compared results achieved by our framework with other state-of-the art super resolution techniques present in the current literature. While comparing we have used metrics like Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). We have performed all the experiments on a Intel Core i7 CPU at 3.6 GHz with 64 GB RAM and Nvidia 8 GB GTX 1080 GPU.

4.1. Datasets

We carried out all experiments on popular datasets. For training, we used 50,000 random images from the ImageNet dataset. For comparing results with other state-of-the art SR techniques we use the following datasets, **Set5** [4], **Set14** [4], **BSD100** [22] , **URBAN100** [14] and **Manga109** [14].

4.2. Comparisons with other State-Of-The-Art methods

We compare our proposed framework's visual performance with 3 other state-of-the-art algorithms for higher scale super resolution, i.e. scale factor = 8. They are GUN [38], RDN [27] and LapSRN [14]. GUN and RDN were originally developed for 4X super resolution while LapSRN had shown results for 8X super resolution. We use publicly available codes for implementing GUN and RDN. We train 8X SR on those frameworks. We use a publicly available trained model for LapSRN for comparing results with our framework for 8X SR. For 4X SR, we compare results obtained from our framework with six other state-of-the-art algorithms. They are, SRGAN [15], VDSR [11], GUN [36], RDN [40], EDSR [26] and LapSRN [1]. All of these algorithms have proved to work well for 4X SR. Among the above mentioned frameworks, SRGAN gives good perceptual quality while its PSNR and SSIM metrics are poor in comparison to other methods. RDN and EDSR are considered to be the best frameworks for 4X SR. We use publicly available codes to train all of these frameworks along with IRGUN for 4X SR. For training IRGUN we use two UEBs cascaded to an E network which is then cascaded to a 3D RED network. We do not compare the performance of our framework for any other scale factor other than 4 and 8.

4.3. Result Analysis

The comparison of proposed IRGUN with other state-of-the-art works in 8X SR is shown in Table 2. This table shows that the results of our proposed framework outperforms other state-of-the-art frameworks in terms of PSNR and SSIM. Typically, various state-of-the-art techniques like SRCNN [4], RED , VDSR , SRresnet , EDSR performs exceptionally well for small magnification ratio but for large magnification ratio (8X), these algorithms fail to recover proper local texture/information. In table. 2, we compare results from our framework with the results from other state-of-the-art algorithms which are meant for higher scale super resolution. These algorithms have already been shown to work excellently for 4X SR while they work really well for 8X SR too. They are GUN [36], RDN [40] and LapSRN [1]. Of these, GUN and RDN were developed to handle 4X SR but can also performs well for 8X SR. LapSRN can handle 8X SR effectively. We get an av-

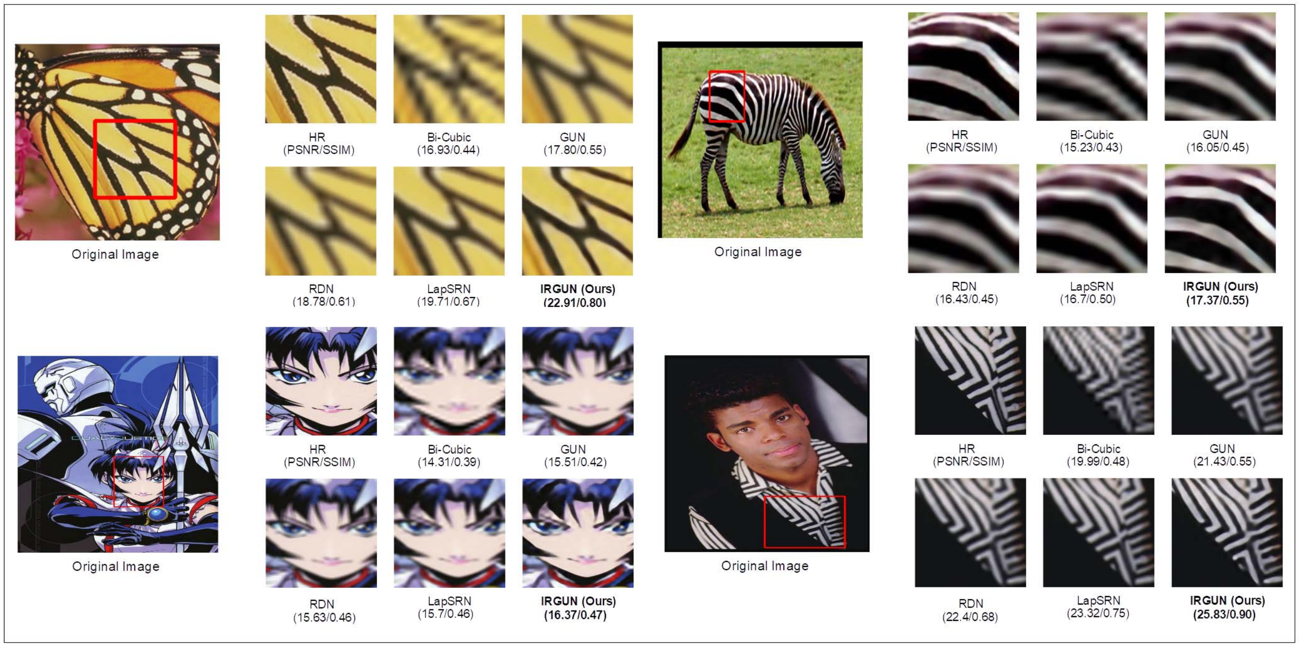

Figure 5. Visual Comparison for 8X SR on Set 5 (top-left), Set 14 (top-right), Manga109 (bottom-left) and BSD100 (bottom-right) dataset

erage improvement of **0.42, 0.72, 1.42, 1.98 dB** from LapSRN [1], RDN [40], GUN [36] and bicubic interpolation respectively for 8X SR on all datasets combined. For all the datasets, IRGUN outperforms other algorithms by a healthy margin which proves the effectiveness of our framework in handling 8X SR and producing state-of-the-art results. We also compare the performance of our framework with other state-of-the-art algorithms for super resolution with magnification factor 4. For 4X SR, we use two UEB's instead of three. We connect the cascaded E network as well as the cascaded 3D residual network in the same way as we do for 8X SR. We compare results with SRGAN [15], VDSR [11], GUN [36], RDN [40], EDSR [26] and LapSRN [1]. As we conclude from Table 3, our framework comfortably outperforms all other frameworks barring EDSR. Our framework performs similar to EDSR on Set 5, Set 14 while gives slightly better result for BSD100 and Manga109 dataset. However, on URBAN100 dataset its performance is inferior to that of EDSR. Overall, our framework produces state-of-the-art results for 4X SR as well. In Figure 5, we have shown the visual comparisons of different algorithms for 8X SR. Our framework restores finer information and local details compared to other methods as evident from Figure 5. We have compared the run-time performance of our framework in Figure 7. Our framework performs better than VDSR but it fails to achieve the speed of GUN, RDN or LapSRN for 8X SR. The reason behind this inferior performance in terms of run time, is the presence of a 3D-RED network in our framework. 3D convolution is computationally complex and thus reduces the run time performance of our framework. In Table 1, we show different model specifications of our method i.e. IRGUN. As we increase the depth of the network and size of the feature map further, performance improves slightly but there is a sharp increase in computational complexity as well. Hence there is a tradeoff between the quality and complexity. Thus, we do not find it feasible to extend IRGUN further in terms of number of layers , the size of the feature map or both to achieve slight better results.

4.4. Evaluation on DIV2K Validation Dataset

The framework was initially designed for the purpose of participation in the NTIRE 2018 Challenge on Super resolution [29]. This challenge consisted of four tracks among which one of them dealt with 8X Super Resolution on a classic bicubically interpolated image. The proposed framework was used for participating in this track. For this purpose the framework was trained only on the training image dataset that was supplied by the organizers. The results that were obtained on the supplied validation dataset are compared in Table 2 and 3. We obtain an average improvement of **1.18 dB** over bicubic interpolation for 8X SR and an aver-

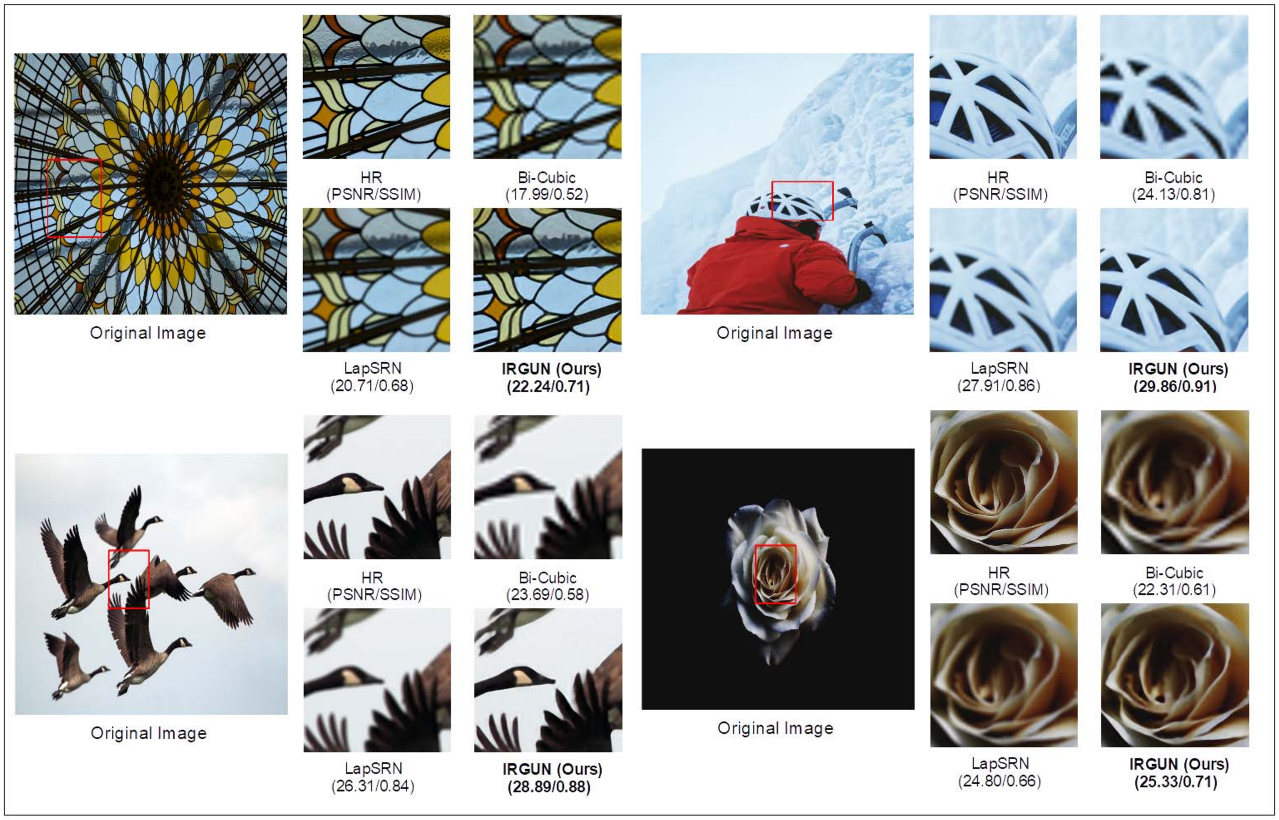

Figure 6. Visual Comparison for 8X SR on DIV2K Validation Dataset (NTIRE SR challenge - 2018)

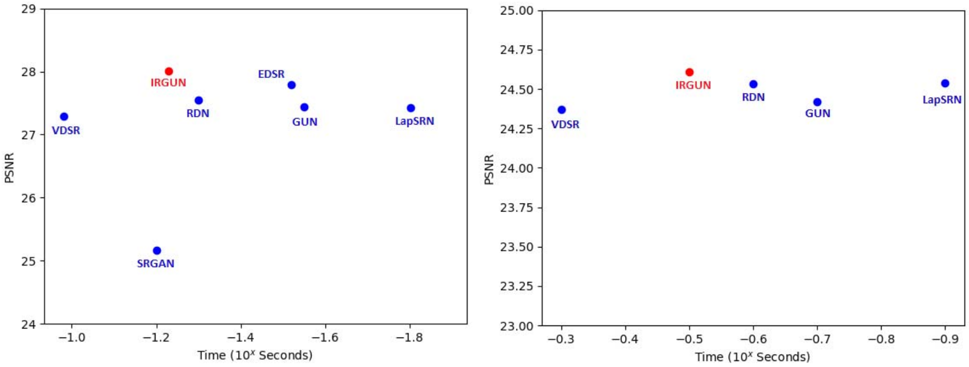

Figure 7. Run-time performances of different frameworks for 4X SR (left) and 8X SR (right) on BSD100 dataset

age **1.65 dB** improvement over bicubic interpolation for 4X SR. We have shown the visual performance of our framework on this dataset in Figure 6.

5. Conclusion

In this work we have shown that gradual up-scaling and further improvement by RED based architecture has improved the performance of super-resolution for large magnification factors. Usage of trained weights to initialize successive UEBs helps in faster and better convergence of the error. We have also experimented and have conclusively proven that the resultant luminance channel obtained from GUREN followed by E network provides better perceptual and objective quality while using optimal computational resources. 3D-CNN enhancement network has further enhanced the quality of SR image by incorporating the inter channel co-related information. Our proposed IRGUN outperforms the current state-of-the-art results for large magnification factors.

References

[1] M. Bosch, C. M. Gifford, and P. A. Rodriguez. Super-resolution for overhead imagery using densenets and adversarial learning. *CoRR*, abs/1711.10312, 2017.

[2] H. Chang, D.-Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, 2004.

[3] S. Dai, M. Han, W. Xu, Y. Wu, and Y. Gong. Soft edge smoothness prior for alpha channel super resolution. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[4] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[5] C. Dong, C. C. Loy, and X. Tang. Accelerating the super-resolution convolutional neural network. *CoRR*, abs/1608.00367, 2016.

[6] N. Efrat, D. Glasner, A. Apartsin, B. Nadler, and A. Levin. Accurate blur models vs. image priors in single image super-resolution. In *2013 IEEE International Conference on Computer Vision*, 2013.

[7] R. Fattal. Image upsampling via imposed edge statistics. In *ACM SIGGRAPH 2007 Papers*, SIGGRAPH '07, 2007.

[8] W. T. Freeman and E. C. Pasztor. Learning low-level vision. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999.

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

[10] J. Kim, J. K. Lee, and K. M. Lee. Accurate image super-resolution using very deep convolutional networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[11] J. Kim, J. K. Lee, and K. M. Lee. Accurate image super-resolution using very deep convolutional networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[12] J. Kim, J. K. Lee, and K. M. Lee. Deeply-recursive convolutional network for image super-resolution. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[14] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[15] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. *CoRR*, abs/1609.04802, 2016.

[16] T. M. Lehmann, C. Gonner, and K. Spitzer. Survey: interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging*, 1999.

[17] X. Li and M. T. Orchard. New edge-directed interpolation. *IEEE Transactions on Image Processing*, 2001.

[18] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee. Enhanced deep residual networks for single image super-resolution. *CoRR*, abs/1707.02921, 2017.

[19] X. Liu, D. Zhao, R. Xiong, S. Ma, W. Gao, and H. Sun. Image interpolation via regularized local linear regression. *IEEE Transactions on Image Processing*, 2011.

[20] X. Liu, D. Zhao, J. Zhou, W. Gao, and H. Sun. Image interpolation via graph-based bayesian label propagation. *IEEE Transactions on Image Processing*, 2014.

[21] X. Mao, C. Shen, and Y. Yang. Image restoration using very deep fully convolutional encoder-decoder networks with symmetric skip connections. *CoRR*, abs/1603.09056, 2016.

[22] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, 2001.

[23] M. S. M. Sajjadi, B. Schölkopf, and M. Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. *CoRR*, abs/1612.07919, 2016.

[24] M. Sharma, S. Chaudhury, and B. Lall. Deep learning based frameworks for image super-resolution and noise-resilient super-resolution. In *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017.

[25] W. Shi, J. Caballero, F. Huszr, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[26] J. Sun, Z. Xu, and H.-Y. Shum. Image super-resolution using gradient profile prior. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[27] Y. Tai, J. Yang, and X. Liu. Image super-resolution via deep recursive residual network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[28] R. Timofte, E. Agustsson, and L. V. Gool. Ntire 2017 challenge on single image super-resolution: Methods and results. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.

[29] R. Timofte, S. Gu, J. Wu, L. Van Gool, L. Zhang, M.-H. Yang, et al. Ntire 2018 challenge on single image super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[30] T. Tong, G. Li, X. Liu, and Q. Gao. Image super-resolution using dense skip connections. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[31] L. Wang, S. Xiang, G. Meng, H. Wu, and C. Pan. Edge-directed single-image super-resolution via adaptive gradient magnitude self-interpolation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2013.

[32] Y. Wang, L. Wang, H. Wang, and P. Li. End-to-end image super-resolution via deep and shallow convolutional networks. *CoRR*, abs/1607.07680, 2016.

[33] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang. Deeply improved sparse coding for image super-resolution. 07 2015.

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

[35] W. Yang, J. Feng, J. Yang, F. Zhao, J. Liu, Z. Guo, and S. Yan. Deep edge guided recurrent residual learning for image super-resolution. *CoRR*, abs/1604.08671, 2016.

[36] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. In *Proceedings of the 7th International Conference on Curves and Surfaces*, 2012.

[37] K. Zhang, X. Gao, D. Tao, and X. Li. Single image super-resolution with non-local means and steering kernel regression. *IEEE Transactions on Image Processing*, 2012.

[38] Y. Zhao, R. Wang, W. Dong, W. Jia, J. Yang, X. Liu, and W. Gao. Gun: Gradual upsampling network for single image super-resolution. *CoRR*, 2017.

[39] Y. Zhao, R. Wang, W. Wang, and W. Gao. High resolution local structure-constrained image upsampling. *IEEE Transactions on Image Processing*, 2015.

[40] C. M. Zwart and D. H. Frakes. Segment adaptive gradient angle interpolation. *IEEE Transactions on Image Processing*, 2013.