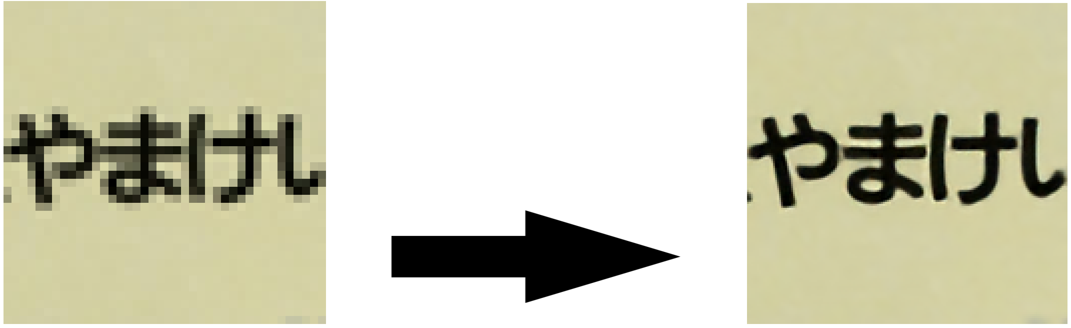
Feedback Network for Image Super-Resolution

[EE645] 3D Computer Vision - Course Project



Viraj Shah 18110188 | **Vrutik Shah** 18110191 | **Yash Kamble** 18110080

TABLE OF CONTENTS

Problem Statement

Super resolution is the task of taking an input of a low resolution (LR) and upscaling it to that of a high resolution. By using the power of deep learning, several methods have been developed in recent years to implement image Super Resolution (SR). However, the feedback mechanism, which commonly exists in the human visual system, has not been fully exploited in existing deep learning based image SR methods.

Our work is based on the paper: *Feedback Network for Image Super-Resolution*

Abstract

This paper proposes an image **Super Resolution Feedback Network (SRFBN)** to implement super resolution. They use hidden states in an RNN with constraints to achieve such feedback manners. A feedback block is designed to handle the feedback connections and to generate powerful high-level representations. The proposed SRFBN comes with a strong early reconstruction ability and can create the final high-resolution image step by step.

Introduction

All of us have seen films and TV series where the CSI team zooms in to an image of a number plate and it improves in quality and the numbers are suddenly visible. This is an application of super resolution. Traditional algorithm based upscaling methods lack fine detail and cannot remove defects and compression artifacts. Deep learning methods can be used to tackle these problems

Previous works:

1. ### Deep learning based image super-resolution

   There have been various deep learning neural networks devised for super-resolution. Researchers have exploited skip connections in their neural networks to obtain improvements in image SR. Due to the limitations caused by small receptive fields, the results obtained lack contextual information. A few examples of the networks would be SRResNet[2] and EDSR[3] which employ residual skip connections and SRDenseNet[4] which employs dense skip connections.

2. Feedback mechanism

   Using feedback mechanisms, the network is able to carry the information of the output to the correct previous states and thereby improving the accuracy of the result.

3. Curriculum learning

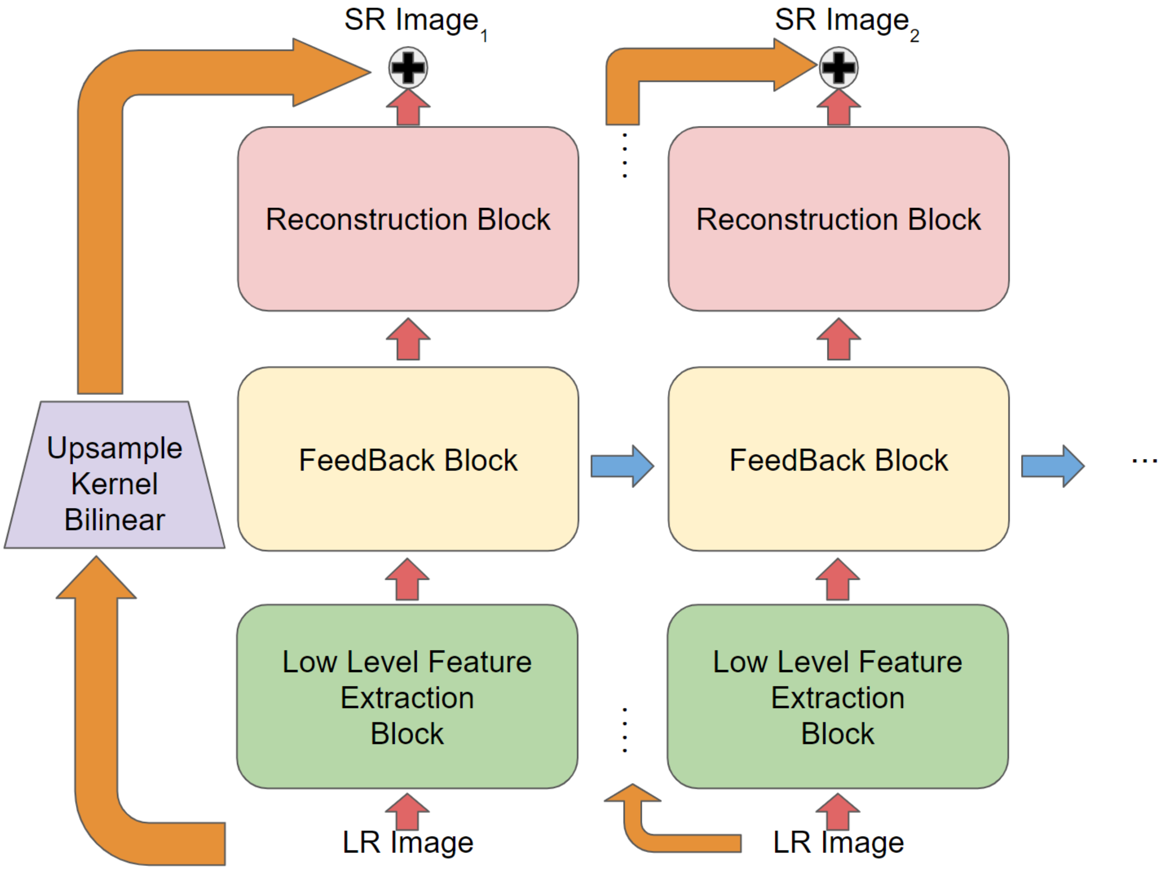
   A curriculum is an efficient tool for humans to progressively learn from simple concepts to hard problems. Similarly, Curriculum learning employs the idea of increasing the difficulty of the learned target, so that the efficiency of the model increases. In the context of imageSR, *Wang et al*. [5] designed a curriculum for the pyramid structure. This blends with the previously trained networks to upscale the input images.

Approach

The network proposed in the paper has 3 parts to enforce the feedback scheme:

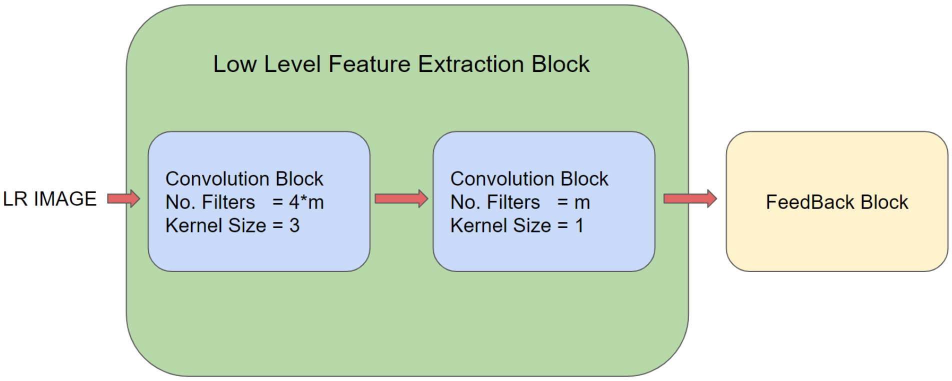
1. Passing high level features by tying loss at each iteration, that can improve the quality of SR image
2. Passing low level features to the same feedback block, so as to refine the required low level features
3. Using Recurrent Network to improve both of the outputs

So, the final network architecture has a recurrent network folded into T iterations, each of which except the first takes the previous LR image and previous output of the feedback block to produce the SR image. The complete network can be broken down into 3 main sections:



1. Low level feature extraction block

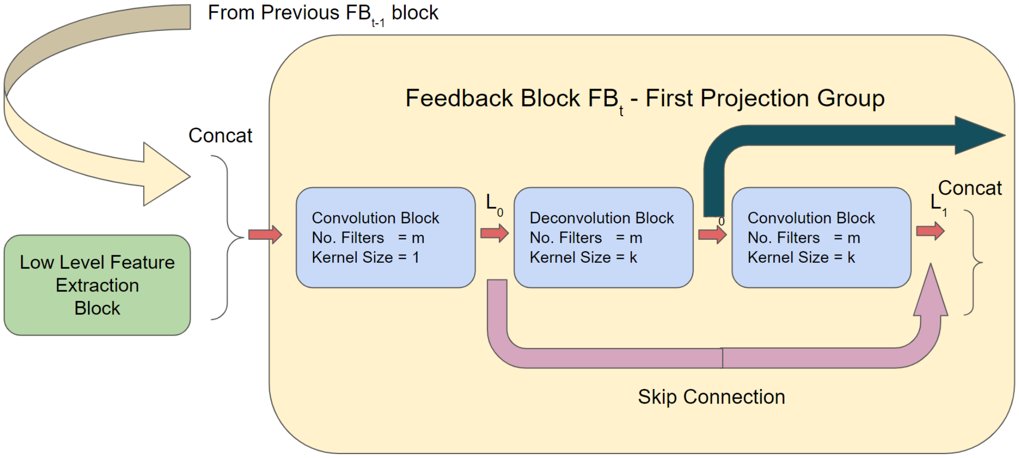
This subpart of the network is used to capture low level features of the image using 2D Convolution. The actual schematic for this block is shown below



2. Feedback Block

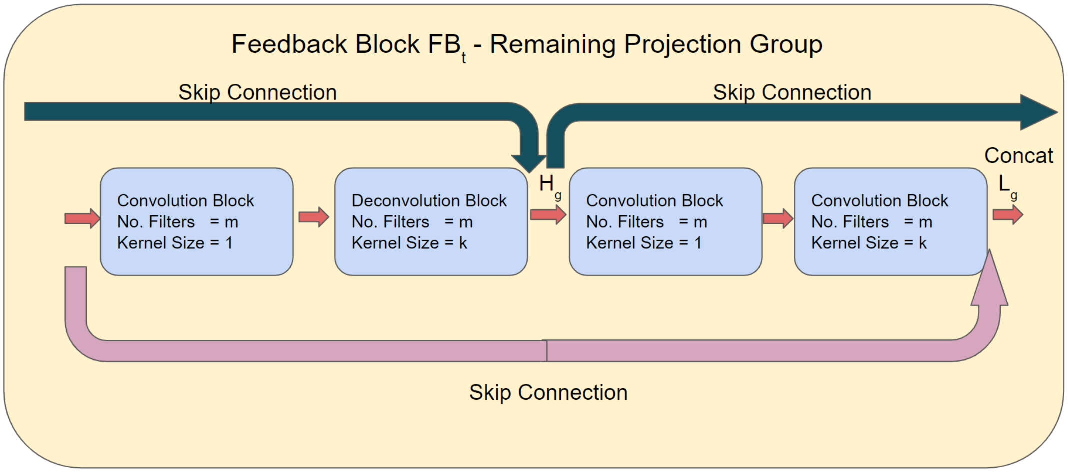
Feedback block comprises several groups of projection blocks with skip connections attached between each group. Input to the next projection block is concatenation of the inputs of all the previous projection blocks.

There is a slight difference between the first and the remaining blocks as shown below.


From Previous FB_{t-1} block
Concat
Feedback Block FB_t - First Projection Group
Low Level Feature Extraction Block
Convolution Block No. Filters = m Kernel Size = 1
L_0
Deconvolution Block No. Filters = m Kernel Size = k
Convolution Block No. Filters = m Kernel Size = k
Concat
L_1
Skip Connection

First Projection block

For remaining projection groups, an additional convolution of kernel size 1x1 and ‘m’ number of filters has been added to reduce the computational resources as the depth level would go on increasing since all the previous inputs are concatenated along the depth axis.

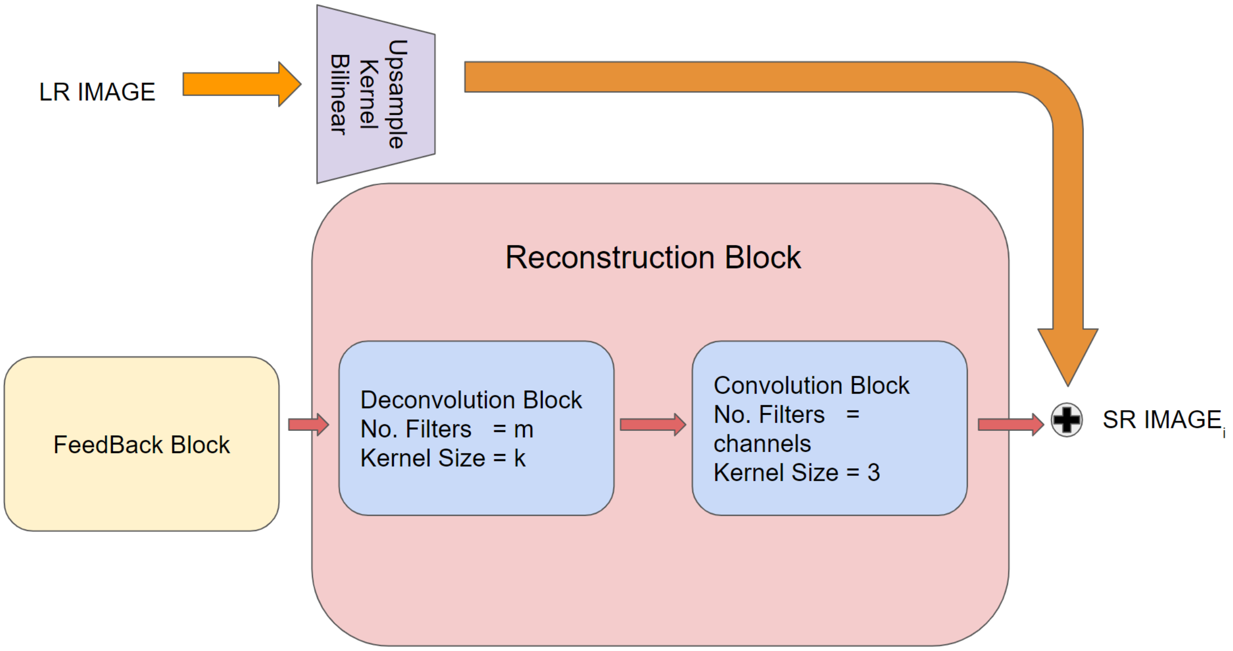

Feedback Block FB_t - Remaining Projection Group
Skip Connection
Skip Connection
Convolution Block No. Filters = m Kernel Size = 1
Deconvolution Block No. Filters = m Kernel Size = k
H_g
Convolution Block No. Filters = m Kernel Size = 1
Convolution Block No. Filters = m Kernel Size = k
Concat
L_g
Skip Connection

Remaining Projection blocks

Skip connections to reduce the overfitting and help in passing both low and high level features. Also, a last convolution block with ‘m’ filters and kernel size of 1 before passing the data to the reconstruction block.

The authors have also experimented with the *number of projection groups*(G) that network should have and found best results for G = 6.

3. Reconstruction Block



4. Up Samping Kernel

The Bilinear Kernel is used as the upsampling kernel. This is added to the reconstruction block to get the SR image. The choice of this kernel is random.

Apart from the network, the authors have also tried to implement curriculum learning. Curriculum learning can be explained in easy words as passing easier samples early on and then passing difficult samples. In this paper, authors have added curriculum learning to their model by introducing 2 different modes: “BD” and “DN” mode. In “BD” mode, Gaussian Blur is added to the HR image (ground truth image) for first half outputs of the network (SR_1, SR_2 … $SR_{T/2}$) and actual HR image as ground truth image for the remaining half ($SR_{T/2+1}$, … SR_T).

We take the SR image SR_T at the last iteration as our final SR result and will be used for calculating all the metrics.

Note: The above mentioned algorithm is implemented on a patch size of 40 x 40 of the LR image as mentioned in the paper for upscale factor of 4

Experimental Settings

The value of T (number of time iterations) has been set to 4, Number of Projection Group (G) has been set to 6, the number of base filters (m) has been set as 32 and the value of k is kept variable depending upon the upscale factor. Following tables list the parameter for all convolutions with kernel size ‘k’.

Upscale Factor	Kernel Size (k)	Stride	Padding
x2	6	2	2
x3	7	3	2
x4	8	4	2

Activation function, PReLU or parametric ReLu, is added after all the convolution and deconvolution blocks except the last one. Before every activation function, batch normalization is done to speed up the training process and provide a regularization effect to the output of the previous block.

Also the model weights for convolution blocks are initialized using kaiming initialization, and BatchNorm layers are initialized with constant weight and no bias. This is to avoid exploding gradients.

The metrics used for evaluation are peak signal noise ratio (PSNR) and structural similarity index (SSIM) only on the Y-channel (Luminous channel) of the image.

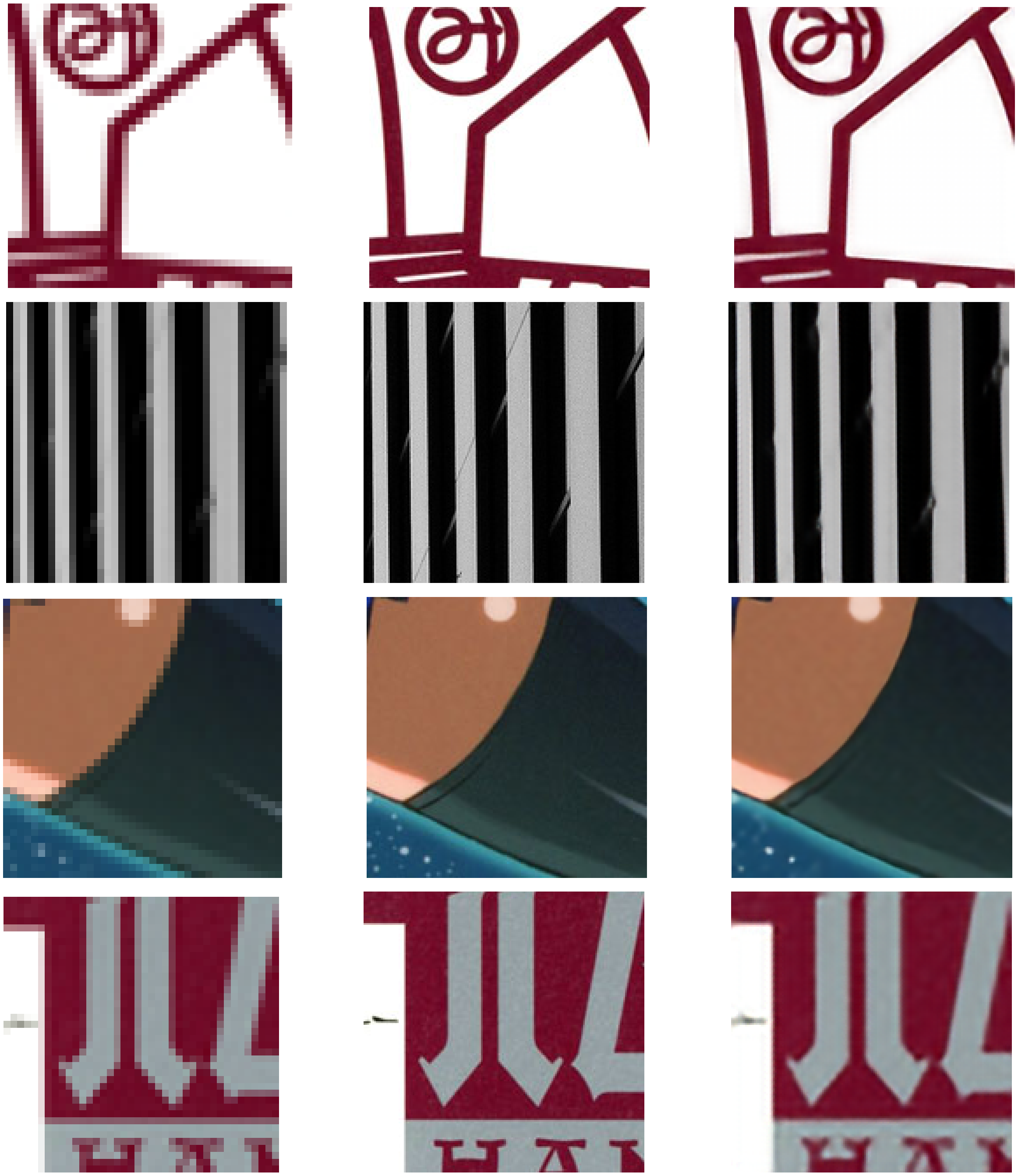
We trained all the models for 500 epochs compared to 1000 epochs as there were computation limitations. The paper was trained on the Div2K+Flickr2K dataset, however the Flickr2K dataset was over 20GB in size and thus we have trained only on Div2K [6] dataset. The data has been augmented by reducing the sizes of the

Qualitative Results

The paper was tested on 3 different datasets (B100, Manga109 and Urban100) and the quantitative results achieved are as follows:

A point to note that since we are taking random 40x40 patches, the results could vary, so we have fixed seed to 42.

Some of the sample outputs using the our trained model are:



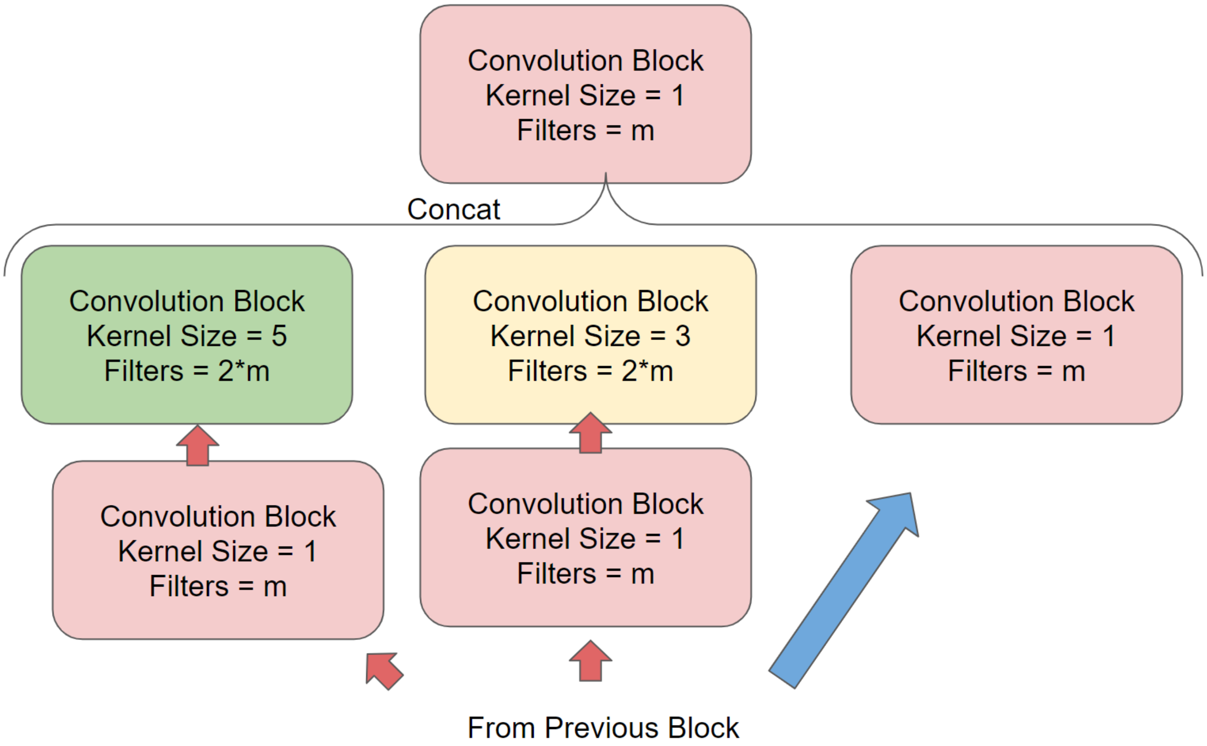
LR (enlarged to same size) HR Our Output

Quantitative Results and our Contribution

The authors of the paper have experimented a lot with the feedback block of the network in the supplementary material for the paper. However, they have not experimented with the other 2 blocks - low level feature extraction blocks and the reconstruction block. Thus we decided to experiment with the feature extraction block.

Inception Network

We tried replacing the feature extraction block with a **subset of Inception network architecture** stacking 3 different kernel convolution blocks along the depth channel. Hereafter, we have referred to this version of our network as “inception network”. Following is the sub-block schematic for the same.



We have added 2 of the above sub-blocks sandwiched with convolution blocks on either side using. The last convolution is of kernel size 1, to make it less expensive computationally.
The metric that we have used for evaluating the performance of the networks (ours and the one in the research paper) are PSNR and SSIM.

PSNR: The term peak signal-to-noise ratio (PSNR) is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation. The signal in our case is the HR image. The distortion in our case results from the reconstruction procedure and hence PSNR is an effective quality measure for SR.

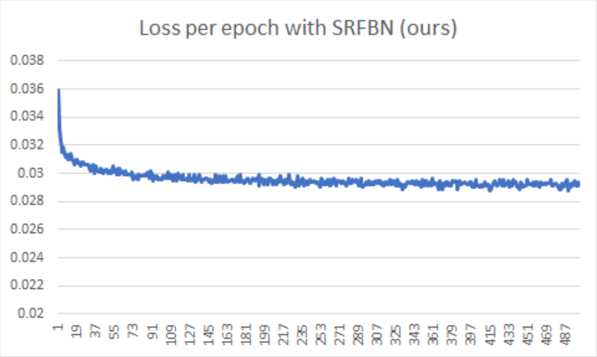
SSIM: The structural similarity index measure (SSIM) is a method for predicting the perceived quality of the image. SSIM is used for measuring the similarity between two images. In our case, the two images are the HR image and the SR image. The SSIM metric takes into account luminance, contrast and structure and hence this metric is better from a human eye perspective.

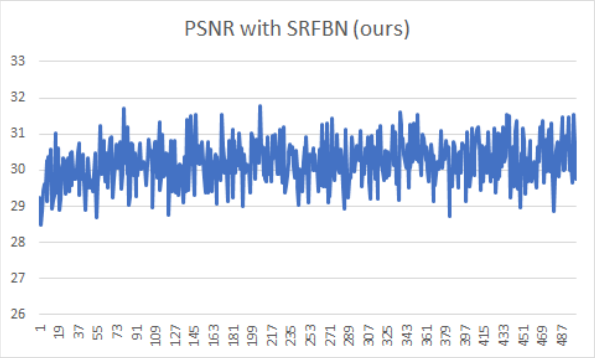
Adding such a network for low level feature extraction, we were able to get the following results (PSNR is in dB):

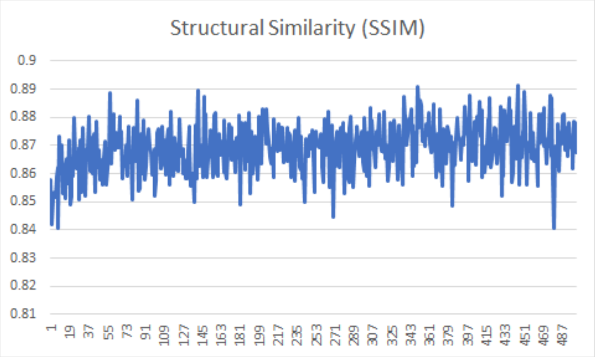
x4	**Inception Network** **(SRFBN + Subset)**	**SRFBN** **(our implementation)**	**SRFBN** **(reported in paper)**
	PSNR/SSIM	**PSNR/SSIM**	**PSNR/SSIM**
B100	**27.47 / 0.840**	27.435 / 0.8397	27.72/0.7409
Manga109	**30.39 / 0.938**	30.197 / 0.937	31.15/0.9160
Urban100	**26.419 / 0.833**	26.41 / 0.833	26.60/0.8015

There is an improvement in the structural similarity when using the inception network over the normal SRFBN that they had used. However, the resulting PSNR was not more than their reported values. We attribute this to the limited training data we had used (due to GPU restrictions)

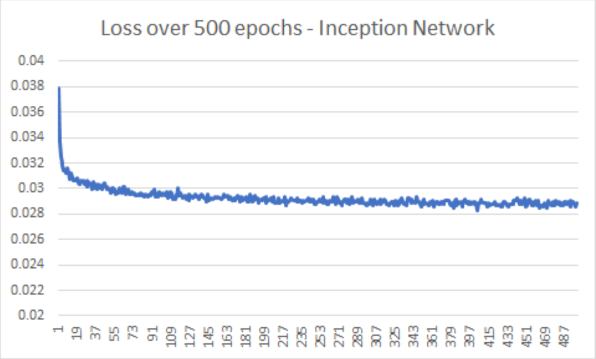
SRFBN (our implementation)

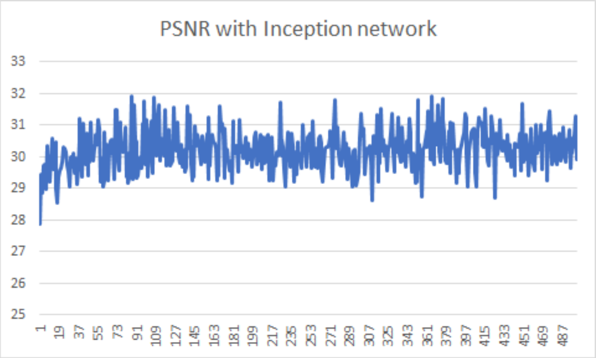


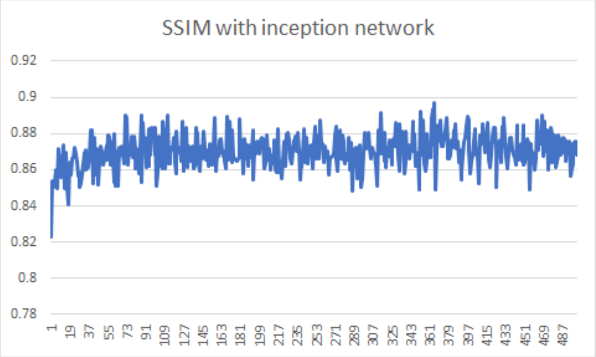




Inception Network







Key observations

Some of the key observations were that the model was able to

- Because of randomly selecting patches, few patches were of the same color completely leading to high value of PSNR and SSIM. Most of the images were complete white/black. We tried to replace those patches whenever we encountered them. This also led to better results as more complex images were available for training.
- The results vary a lot when we remove the seed and test for random value for seed. This was expected as upscaling would have been different for different parts of the images.
- BatchNormalization helps in increases the training speed and led to faster convergence
- Kaiming Initialization helped in improving the test results by a lot, indicating how important initialization is to the network. It also prevented exploding gradients.

References/Citations

1. TowardsDataScience
2. Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero,Andrew Cunningham, Alejandro Acosta, Andrew P. Aitken,Alykhan Tejani, Johannes Totz, Zehan Wang, and WenzheShi. Photo-realistic single image super-resolution using a generative adversarial network. In CVPR, 2017.
3. Qianli Liao and Tomaso Poggio. Bridging the gaps between residual learning, recurrent neural networks and visual cor-tex.arXiv preprint arXiv:1604.03640, 2016.
4. Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, andYun Fu. Residual dense network for image super-resolution.In CVPR, 2018
5. Yifan Wang, Federico Perazzi, Brian Mcwilliams, Alexan-der Sorkinehornung, Olga Sorkinehornung, and Christopher Schroers. A fully progressive approach to single-image super-resolution. In CVPRW, 2018
6. Eirikur Agustsson and Radu Timofte.NTIRE 2017 Challenge on SingleImage Super-Resolution: Dataset and Study.

Individual Contribution

Team Member	Contribution
Viraj Shah	PyTorch Implementation of SRFBN, Curriculum learning implementation, Inception Network addition, Ablation study, Hyperparameter tuning, Report writing.
Vrutik Shah	Curriculum learning implementation , Inception Network addition, ablation study, Hyperparameter tuning, Collection of data (while testing), Report Writing.
Yash Kamble	Testing baseline and the paper’s code, Ablation Study: Experimented with Batch Normalization and compared with previous models, Collection of data (while testing), Report Writing.