IMAGE SUPER RESOLUTION USING DEEP LEARNING ALGORITHMS

- By Saurabh Singh

Submitted to: Professor Mohammad Saiful Islam

INTRODUCTION

- Image super-resolution (SR), which refers to the process of recovering high resolution (HR) images from low resolution (LR) images, is an important class of image processing techniques in computer vision and image processing
- Image resolution describes the details contained in an image, the higher the resolution, the more image details.
- Traditionally, single image super resolution methods have been performed such as bicubic which was based on interpolation. Other learning methods such as nearest neighbor assumes that their exist transformation between low resolution and high resolution patches.

Low Resolution

High Resolution

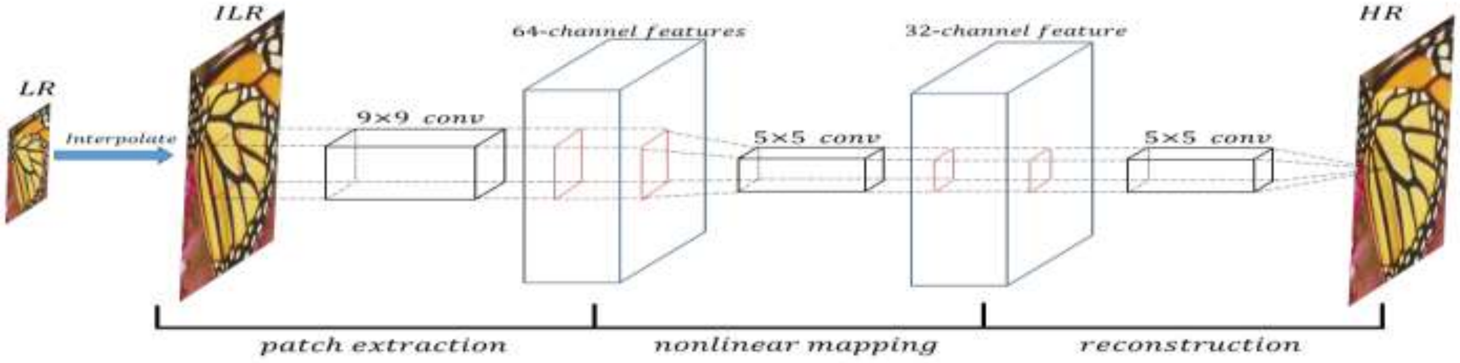
OBJECTIVE

- The project's primary objective is to showcase the image super resolution algorithms on different datasets and implement the different algorithm to upscale and improve the quality of low resolution images
- Another objective is to use adversarial learning to improve the current state-of-the-art in Single image super resolution.
- The project implemented the architecture of two deep learning algorithms namely SRCNN and SRGAN on different dataset to highlight how the different algorithms can be used to enhance the resolution of an image from low resolution to high resolution.
- Our team is excited to showcase the impact of this project and the ways it can be utilized in different applications

MOTIVATION

- The motivation behind working on Image super-resolution project using deep learning is based upon factors such as Enhanced image quality, improved visual experience
- Diverse applications in numerous domains.
- The need of converting low resolution images to high resolution images thus removing the noise, compression and other forms of degradation from the image
- Learning and training related to deep learning algorithms

SRCNN ARCHITECTURE



- Super Resolution using Deep Convolutional Neural Network (SRCNN) is a simple CNN architecture consisting of three layers: one for patch extraction, non-linear mapping, and reconstruction.

- The first layer: The input image with a 9x9 kernel with padding.
- The second layer: expands the three-channel image into 64 feature maps, applies a 5x5 kernel to condense to 32 feature maps.
- The third layer: applies a 5x5 kernel to generate the output image

- Deep learning algorithms such as SRCNN learns end-to-end mapping between the low/high-resolution images. The mapping is represented as a deep convolutional neural network (CNN) that takes the low-resolution image as the input and outputs the high-resolution

BUILDING SRCNN MODEL

- We used the popular dataset DIV2K to train our models. The dataset has been provided in the original published paper. More links in the reference.
- The DIV2K dataset contains 900 high-quality RGB images of different size for training and validation.
- MSE loss function is used to train the network architecture
- The testing LR images were generated by down-sampling HR images with scaling factors (x2) using bicubic interpolation
- Model type was sequential. Model was compiled on local machine, it was slow due to the computer specifications. With total number of parameters of 85,889
- Hyper parameters of SRCNN:

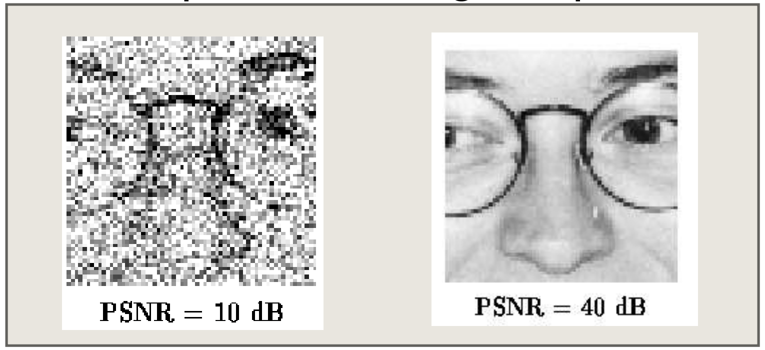
Parameters	SRCNN
Optimizer	ADAM
Learning Rate	10^{-3}
Activation Function	Relu
Loss Function	MSE
Batch Size	16

IMAGE QUALITY ASSESSMENT METRICES

MSE: The difference between the Pixel value of one image and the corresponding Pixel value of the other image. The MSE measures the average of the square of the errors

$$MSE = \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (A(i.j) - B(i.j))^2$$

PEAK SIGNAL TO NOISE RATION (PSNR):PSNR is used to calculate the ratio between the maximum possible signal power and the power of the distorting noise.



Structure Similarity Index Method (SSIM): It measures similarity between two images

SRCNN RESULTS

We tested the model on few of the images and found out that images generated from SRCNN model has improved compared to degraded images.

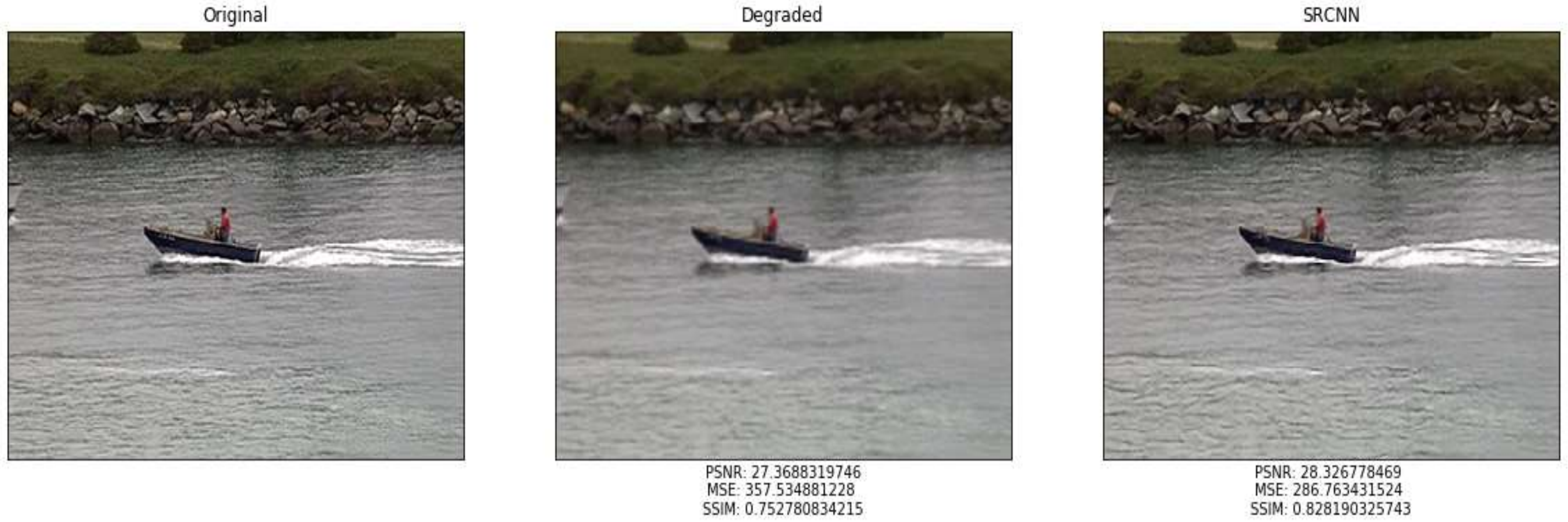
Mean square Error has decreased drastically while the Peak signal to noise ration has improved marginally for SRCNN model compared to degraded image

```
Degraded Image:
PSNR: 27.2486864596
MSE: 367.564000474
SSIM: 0.86906220246

Reconstructed Image:
PSNR: 29.6675381755
MSE: 210.594874985
SSIM: 0.899043290319
```

SRCNN RESULTS

Mean square Error has decreased drastically while the Peak signal to noise ratio has improved marginally for SRCNN model compared to degraded image



SRCNN RESULTS- CUSTOM DATA

We tested the model on custom dataset, that were generated from computer webcam and below are the results that were achieved from the model.

PSNR: Peak signal to noise ratio improved in SRCNN generated image(43.45) compared to the Degraded image (40.09)

MSME: Mean Square Error decreased drastically from 19.07 for degraded image to SRCNN image (8.80)

SSIM: Structural similarity improves slightly for SRCNN image (0.98) compared to Degraded image

GENERATIVE ADVERSARIAL NETWORK

- GAN is first introduced in 2014 a an especially effective type of generative model and has been subject of intense interest in machine learning community
- A generative adversarial network (GAN) has two parts:
- The generator learns to generate plausible data. The generated instances become negative training examples for the discriminator.
- The discriminator learns to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results.
- SRGAN is a type of GAN

SRGAN ARCHITECTURE

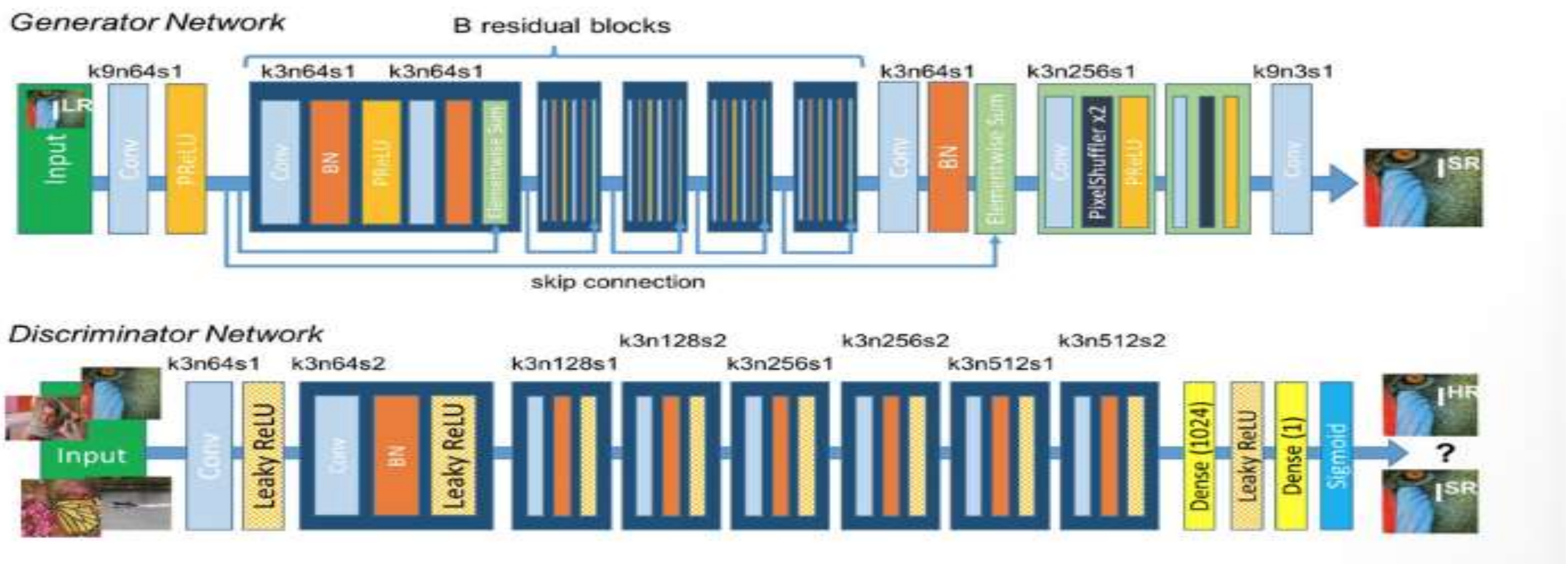


Figure 4: Architecture of Generator and Discriminator Network with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer.

- During the training, A high-resolution image (HR) is down sampled to a low-resolution image (LR). The generator architecture than tries to up sample the image from low resolution to super-resolution.
- After then the image is passed into the discriminator, the discriminator and tries to distinguish between a super-resolution and High-Resolution image and generate the adversarial loss which then backpropagated into the generator architecture.

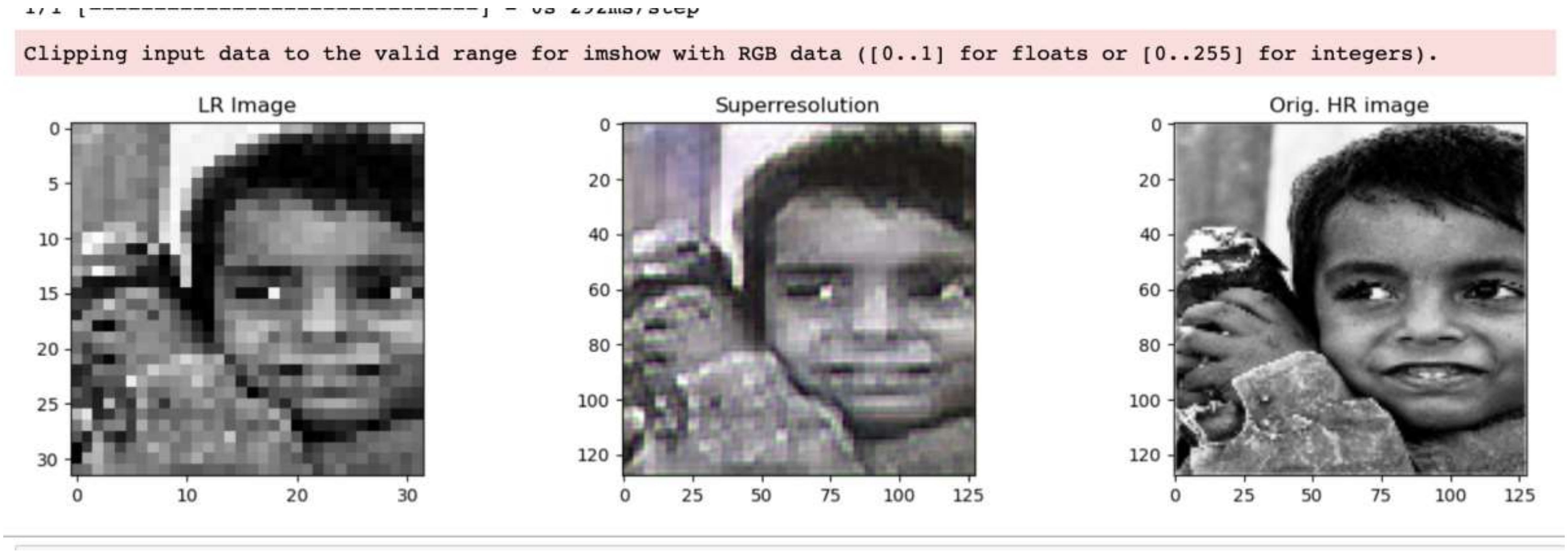
BUILDING SRGAN MODEL

- We used the MIRFLICKR dataset having 25000 images and 3gb in size. Images were in .PNG format.
- The dataset is referred from original published paper, links in the reference.
- All images resized to 128x128 to represent HR and 32x32 to represent LR.
- The generator and discriminator are jointly trained using adversarial loss (binary_crossentropy and content loss (MSE - Mean Squared Error) to achieve photo-realistic single-image super-resolution.
- The adversarial loss is used to train the generator to produce more realistic HR images, while the content loss helps the generator generate HR images that preserve the content features of the input LR images
- Model was trained on training sample of 700 images and made prediction on test sample of 300 images. With total number of parameters for generator is 2044291 and for discriminator is 2325568
- Hyper parameters of SRCNN:

Parameters	SRCNN
Optimizer	ADAM
Learning Rate G	10^{-3}
Learning Rate D	10-3
Activation Function	PRelu
Sample size	1000
Batch Size	1
Epochs	10

SRGAN RESULTS

- Image quality assessment for the SRGAN results would be result on subjective methods based on pair wise similarity judgement.

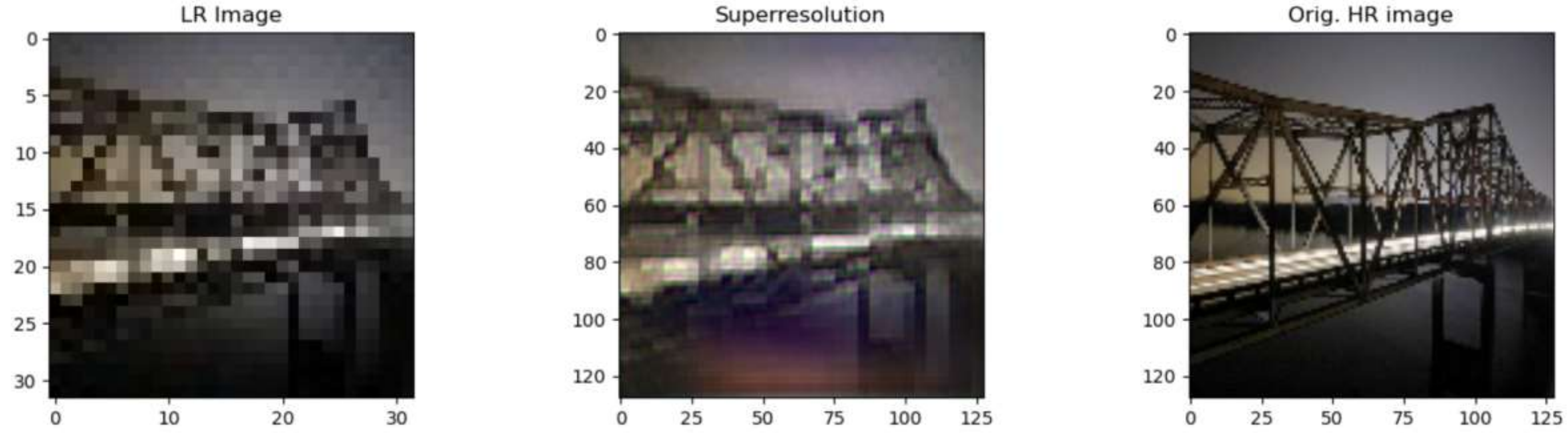


SRGAN RESULTS

- Image quality assessment for the SRGAN results would be result on subjective methods based on pair wise similarity judgement.

```
1/1 [==============================] - 0s 282ms/step
```

```
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
```



CONCLUSION

- In conclusion, our project demonstrated working capability of SRCNN and SRGAN deep learning architecture and was able to successfully create a model.
- The output generated from SRCNN was super resolution image with improved parameters of MSE, PSNR and SSIM
- The output generated from SRGAN is achieved by different combination of batch size, early stopping mechanism and learning rate and epochs. while the model ran for significant duration of time with limited sample size. Model could have achieved the better results if it were to be trained on large sample size with increased no of epochs.
- The project is gradually producing improved resolution of images from Low resolution images with limited train and test dataset and epochs on train data.

FUTURE WORKS

- Improving results of the models by training the models on more time and larger datasets.
- Working on different deep learning algorithms such as FSRCNN, ESPCN, RDN, RDN and ESRGAN according to their architecture.
- Build a Web app with stable server that can handle the models and predict fast super resolution images.
- Build a Mobile app (IOS, and Android) that can predict fast super resolution images.
- Modification of the architecture used in building the model so that we can achieve higher accuracy and less information loss.
- Working on video Super Resolution.