Image Super-Resolution Using CNN Optimized by Enhancement Parameters

Shveta Singh, M.Tech Scholar, Department of Computer Science & Engineering, Kanpur Institute of Technology, Kanpur, India.

Rahul Singh, Assistant Professor, Department of Computer Science & Engineering, Kanpur Institute of Technology, Kanpur, India

Abstract—**Considering the popularity of state-of-the-art single image superresolution methods based on deep convolutional neural networks in respect of redevelopment accuracy and computational efficiency, the majority of suggested models focus on minimising the mean square error function. Mean Square Error (MSE)-based information loss estimation has lately been substituted by loss computed on feature maps of pre-trained networks, such as the VGG-net used mostly for ImageNet classification, as a result of transfer learning. We show that this alternative method is inefficient, resulting in false colour and mosaicking artefacts in the rebuilt images. The first Convolutional Neural Network (CNN) capable of adjusting its parameters by minimising the loss of in-network, self-features is presented in this research. To accomplish this, we propose a new loss function for a light CNN architecture that includes a residual block mapping between low and high resolution pictures. By effectively suppressing misleading color-effects, our suggested method outperforms previous methods that use perceptual loss. We demonstrate that the in-network features used to determine the loss function will provide fresh insights for future research and applications when developing deep learning networks for additional computer vision tasks like demosaicing and denoising.**

Index Terms— CNN, Deep Learning, PSNR, SSIM, LR image, HR image

I. INTRODUCTION

Digital imaging has been a boon and the source of technical advancements in many areas. The developments in the image capturing techniques make digital image processing suitable for many applications in various fields of science and technology. The major application areas of digital imaging include video surveillance (target recognition, license plate recognition), satellite imaging, medical image analysis and computer vision. The images used in these areas, either captured by the camera or generated by image processing techniques, play a vital role. For instance, in medical applications, digital images are very useful for the physicians to make accurate diagnosis. Similarly, in the case of satellite imaging applications, it makes the detection of objects easier from the background as well. The computer vision applications also benefit from the applicability of digital images. Over the years, the image capturing techniques have seen drastic developments. However, certain areas of application still require some techniques to improve the quality of the images. This research work is focused on the development of approaches to improve the image quality.

The problem of generating a high-resolution (HR) image given a low-resolution (LR) image, commonly referred to as Single Image Super-Resolution (SISR) creation. In general, it is assumed that the degradation that happens during image acquisition such as warping, down-sampling, blurring and the presence of noise are results of degrading factors, such as optical diffraction, under-sampling, relative motion and system noise, respectively. Mathematically this turns out to be a highly ill-posed inverse problem, due to the loss of the information. The SR operation is effectively a one-to-many mapping from LR to HR space, which can have multiple solutions, of which determining the correct solution is non-trivial.

Currently, deep learning based approaches have delivered ground breaking performance improvements in many areas of computer vision. Meanwhile the state of the art for many computer vision problems is set by specially designed Convolutional Neural Networks (CNN) architectures following the success of the work by Krizhevsky et al. [1]. Several SISR models [2] based on CNNs have been proposed and have attained superior performance that overshadows all previous handcrafted methods. Dong et al. [3] first demonstrated that SRCNN can be used to learn a mapping from LR to HR in an end-to-end manner. ESPCN [4] performs the feature extraction stages in the LR space instead of upsampled HR space. VDSR [5] shows that applying residual learning is capable of optimizing a very deep network fast, as well as to recover high-frequency details. SRGAN [6] is a very deep ResNet[7] architecture using the concept of GANs [8] to form a perceptual loss function for photo-realistic SISR.

The objective function of these supervised methods is commonly the minimization of the pixel-wise mean square error between the recovered HR image and the ground truth. There is an inverse relationship between MSE and the peak signal-to-noise ratio (PSNR) which is a common objective measure used to evaluate and compare SR algorithms. However, this indicator does not necessarily reflect human visual response to image quality. While SRGAN [6] and Johnson et al. [9] successfully introduced perceptual loss into deep learning-based SR framework, we find three limitations of these approaches: firstly, they rely on pre-trained networks; secondly, they produce disturbing false colourartefacts; thirdly, they apply very deep network structures and hence training converges slowly. Considering the above observations we summarise below the contributions of the proposed novel approach:

- We define a novel loss function using in-network, self-features.
- The parameter consistency of features and network will decrease the false colour effects generated when using pre-trained networks.
- We demonstrate that a lightweight CNN is still capable of achieving a level of quality similar to that achievable by deeper CNN's used in state-of-the-art approaches.

II. LITERATURE REVIEW

Image super-resolution is very effective in enhancing the quality of digital images for analysis. It is an efficient way to improve the quality of the captured images. Better image quality improves the accuracy of the image analysis process. The increase in the quality of input images directly impacts the results.Over the years several attempts have been made to devise methodologies to improve the resolution of digital images. The researchers have attempted various approaches to generate high resolution images from low resolution input images. An elaborated study on the various image super-resolution approaches available in literature has been made.

- **Super-resolution optical flow (Baker and Kanade, 1999)**

The first implementation of optical flow for super-resolution was explained by Baker and Kanade in the year 1999. The detailed experimental results were presented, and the relationship between optical flow for superresolution and pyramid-based image representations such as the Laplacian pyramid was described. This super-resolution optical flow took a conventional video stream as the input, and computed both the optical flow and an enhanced version of the entire video using super-resolution simultaneously. The four major steps in super-resolution were registration, warping, fusion and deblurring. Warping is the process of constructing the high resolution image. Once the mapping from pixels in the HR image to points in the LR images was done, warping was applied. Warp uses the interpolation of lower resolution images. One of the well-known interpolation algorithms, such as nearestneighbor, bilinear, or cubic B-spline was used for interpolation.

- **Super-resolution: reconstruction or recognition? (Baker and Kanade, 2001)**

The study by Baker and Kanade resulted in the key finding that there was more to super-resolution than just image reconstruction. The constraints for reconstruction were analyzed as the first step and found to be providing less and less useful information with the increase in magnification factor. Afterwards, a hallucination algorithm was used in order to incorporate the recognition of local features in the low resolution images.

- **Knowing and breaking the limits on super-resolution (Baker and Kanade, 2000) (Baker and Kanade, 2002)**

The sequence of analytical results derived by Baker and Kanade showed that the useful information provided by the reconstruction constraints decreases with the increase in scaling factors. The empirical validation of the results showed that for large enough scaling factors any smoothness prior lead to overly smooth results with very little useful information. The algorithm attempted to recognize the local features in the low resolution images for enhancing the resolution. This kind of image super-resolution algorithm was termed as a hallucination or reconstruction algorithm.

To generate the multi-scale derivative features, the following series of operations were carried out. The Gaussian pyramid was formed as the first step. The Laplacian pyramid was formed by finding the horizontal and vertical first derivatives of the Gaussian pyramid as well as the horizontal and vertical second derivatives of the Gaussian pyramid. Baker and Kanade carried out experiments with various types of environments such as human faces, robustness to additive intensity noise, variation in performances with the change in input image size, results on non-FERET (Facial Evaluation Recognition Technology) test images, results on images not containing faces as well as the experimental results on text data.

- **Learning-based super-resolution of 3D face model (Peng et al. 2005**)

Peng et al. focused on the generation of a 3D face model with higher resolution from a single 3D face model. Initially, the 3D face models were regularized by resampling them in cylindrical representation. Then the superresolution was performed in the regular domain of cylindrical coordinate. The regularized training samples, in cylindrical coordinate, were used in building a special feature structure called the parent structure. The parent structure was built for each sample. If a low resolution 3D face model was given as the test image, the algorithm regularized it and calculated its parent structure. Based on the parent structures, the prediction algorithm produced the regularized image with higher resolution, which helped in recovering the high resolution of 3D face model.

- **Example-based super-resolution of a single document using a global MAP approach (Datsenko and Elad, 2007)**

Datsenko and Elad developed a scheme using image examples as driving a powerful regularization and applied for the super-resolution (image scale-up) problem. They focused on the super-resolution of a specific target, the scanned documents having written text, graphics, and equations. The algorithm assigned several candidate high-quality patches per each location in the downscaled or down sampled image as the starting step. The candidates which were the nearest-neighbors in the learning dictionary were used for the definition of an image prior expression. These candidates were then merged into a global MAP penalty function. The penalty function was used for the rejection of some of the irrelevant outlier examples, and then for the reconstruction of the expected image.

- **Application of Eigentransformation and error regression model for hallucinating color face images (Boonim and Sanguansat, 2010)**

Boonim and Sanguansat exploited the benefits of an error regression model in order to improve the efficiency of the reconstruction of facial images. The inclusion of error information helped the framework to correct the resultant outcome. Similarly, the application of regression analysis was adopted in order to find the error estimation, which could be obtained from the existing LR in eigen space for each color channel. By processing each color channels in the RGB model separately, this framework was made to work for color images.Initially, the error of face image reconstruction was learned from training dataset by Eigen transformation. Then regression analysis was performed to find relationship between input and error (Zhuanget al. 2007). Second phase hallucinated the image using normal method by Eigentransformation, then the final result was corrected with the help of error estimation.

- **Residual Learning for SR**

Recent research on SR has progressed with the development of deep convolutional neural networks. In particular, residual learning techniques exhibit improved performance. To efficiently train deep network architectures, He et al. [7] introduced the concept of residual blocks for image recognition, which is proposed to solve higher-level computer vision problems, such as image classification and detection. Since the capability of easing the training procedure, deep residual networks have been shown to increase performance for SISR. e.g. Legit et al. [6] formulated a SR ResNet by simply employing the ResNet architecture without much modification to recover photo-realistic images and present state-of-the-art results. VDSR [5] learns residuals only for training much deeper network architectures, which largely increases the convergence speed during training. While achieving superior performance, VDSR can handle SR of several scales jointly in the single network. Lim et al. [10] proposed an enhanced SR algorithm by removing unnecessary modules from conventional ResNet architecture and employing residual scaling techniques to train large models in a stable manner.

- **Feature Extraction**

A deep CNN structure can compute a feature hierarchy, layer-by-layer. With subsampling layers, the feature hierarchy has an inherent multi-scale, pyramidal shape. This kind of an in-network feature hierarchy produces feature maps of different spatial resolutions, but introduces large semantic gaps caused by different depths. Carciaet al. [11] discussed the behaviour of CNN for feature extraction: Deep neural networks are representation learning techniques. During training, a deep net is capable of generating a descriptive language of unprecedented size and detail inmachine learning. Extracting the descriptive language codedwithin a trained CNN model, and reusing it for other purposesis a field of interest, as it provides access to the visual descriptors previously learnt by the CNN after processing millions of images, without requiring an expensive training phase. Lin et al. [12] leveraged the CNN architecture as a generic feature extractor for object detection. The pixel MSE loss is the most common optimization target for image SR on which deep learning-based approaches rely. Nevertheless, while achieving high PSNR, solutions of MSE optimization problems often result in perceptually unsatisfying solutions. In order to make the reconstructed image more visually pleasing, perceptual loss was proposed in recent work and was widely used since. Johnson et al. [9] proposed the use of perceptual loss functions for training feed-forward networks for image style transfer and super-resolution image creation. Ledig et al. [6] defined the VGG loss based on ReLU activation layers of the pre-trained VGG network described in Simonyan and Zisser-man [13].

III. CNN

A CNN is a special case of the neural network described above. A CNN consists of one or more convolution allayers, often with a sub sampling layer, which are followed by one or more fully connected layers as in a standard neural network. The design of a CNN is motivated by the discovery of a visual mechanism, the visual cortex, in the brain. The visual cortex contains a lot of cells that are responsible for detecting light in small, overlapping sub-regions of the visual field, which are called receptive fields. These cells act as local filters over the input space, and the more complex cells have larger receptive fields. The convolution layer in a CNN performs the function that is performed by the cells in the visual cortex [9].A typical CNN for recognizing traffic signs is shown in Figure 1.

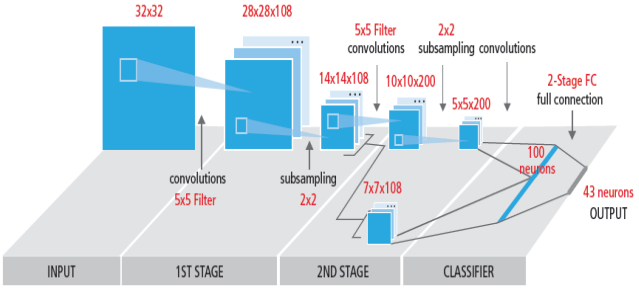

Figure 1: Typical block diagram of a CNN [12]

Each feature of a layer receives inputs from a set of features located in a small neighborhood in the previous layer called a local receptive field. With local receptive fields, features can extract elementary visual features, such as oriented edges, end-points, corners, etc., which are then combined by the higher layers. In the traditional model of pattern/image recognition, a hand-designed feature extractor gathers relevant information from the input and eliminates irrelevant variability. The extractor is followed by a trainable classifier, a standard neural network that classifies feature vectors into classes.

In a CNN, convolution layers play the role of feature extractor. But they are not hand designed. Convolution filter kernel weights are decided on as part of the training process. Convolutional layers are able to extract the local features because they restrict the receptive fields of the hidden layers to be local.

CNNs are used in variety of areas, including image and pattern recognition, speech recognition, natural language processing, and video analysis. There are a number of reasons that convolutional neural networks are becoming important. In traditional models for pattern recognition, feature extractors are hand designed. In CNNs, the weights of the convolutional layer being used for feature extraction as well as the fully connected layer being used for classification are determined during the training process. The improved network structures of CNNs lead to savings in memory requirements and computation complexity requirements and, at the same time, give better performance for applications where the input has local correlation (e.g., image and speech).

The input images acquired via still or video cameras might suffer from noise, bad illumination or unrealistic color. Therefore, noise removal might be a necessary block in some cases. Histogram equalization is the most common method used for image enhancement when images have illumination variations [25]. Even for images under controlled illumination, histogram equalization improves the recognition results by flattening the histogram of pixel intensities of the images.

- **CNN Architectures**

The architecture depends on the size of training data. Less data should drive a smaller network (fewer layers and filters) to avoid over fitting. In this study, the size of our training data is much smaller than those used by [29]; therefore, smaller architectures are designed. We propose three CNN architectures suitable for LFW. These architectures are of three different sizes: small (CNN-S), medium (CNN-M), and large (CNN-L). CNN-S and CNN-M have 3 convolutional layers and two fully connected layers, while CNN-M has more filters than CNN-S. Compared with CNN-S and CNN-M, CNN-L has 4 convolutional layers. The activation function we used is Rectification Linear Unit (RELU) [24].
In our experiments, dropout [23] did not improve the performance of our CNNs, therefore, it is not applied to our networks. Following [24, 31], softmax is used in the last layer for predicting one of K (the number of subjects in the context of face recognition) mutually exclusive classes. During training, the learning rate is set to 0.001 for three networks, and the batch size is fixed to 100.

- **Bilinear CNNs**

This method proposes a novel approach for recognizing the human faces. The recognition is done by comparing the characteristics of the new face to that of known faces. It has Face localization part, where mouth end point and eyeballs will be obtained. In feature Extraction, Distance between eyeballs and mouth end point will be calculated. The recognition is performed by Neural Network (NN) using Back Propagation Networks (BPN) and Radial Basis Function (RBF) networks. Back propagation can train multilayer feed-forward networks with differentiable transfer functions to perform function approximation, pattern association, and pattern classification.
The BPN is designed with one input layer, one hidden layer and one output layer. The input layer consists of six neurons the inputs to this network are feature vectors derived from the feature extraction method in the previous section. The network is trained using the right mouth end point samples. The Back propagation training takes place in three stages: Feed forward of input training pattern, back propagation of the associated error and Weight adjustment. During feed forward, each input neuron receives an input value and broadcasts it to each hidden neuron, which in turn computes the activation and passes it on to each output unit, which again computes the activation to obtain the net output. During training, the net output is compared with the target value and the appropriate error is calculated. From this, the error factor has been calculated which is used to distribute the error back to the hidden layer.
The weights are updated accordingly. In a similar manner, the error factor is calculated for a single unit. After the error factors are obtained, the weights are updated simultaneously. The output layer contains one neuron. The result obtained from the output layer is given as the input to the RBF. RBF uses the gaussian function for approximation. For approximating the output of BPN, it is connected with RBF. The Radial Basis Function neural network is found to be very attractive for the engineering problems. They have a very compact topology, universal approximations; their learning speed is very fast because of their locally tuned neurons. The RBF neural network has a feed forward architecture with an input layer, a hidden layer and an output layer. A RBF neural network is used as recognizer in face recognition system and the inputs to this network are the results obtained from the BPN. This neural network model combined with BPN and RBF networks is developed and the network is trained and tested.
A key advantage is that the bilinear CNN model can be trained using only image labels without requiring ground-truth part-annotations. Since the resulting architecture is a directed acyclic graph (DAG), both the networks can be trained simultaneously by back-propagating the gradients of a task-specific loss function. This allows us to initialize generic networks on ImageNet and then fine-tune them on face images. Instead of having to train a CNN for face recognition from scratch, which would require both a search for an optimal architecture and a massive annotated database, we can use pre-trained networks and adapt them to the task of face recognition.
When using the symmetric B-CNN (both the networks are identical), we can think of the bilinear layer being similar to the quadratic polynomial kernel often used with Support Vector Machines (SVMs). However, unlike polynomial-kernel SVM, this bilinear feature is pooled overall locations in the image and can be trained end-to-end.

- **Training-CNN**

The training of CNNs is very similar to the training of other type NNs, such as ordinaries MLPs. A set of training example is required, and it is preferable to have separate validation set in order to perform cross-validation and “early stopping” and to avoid over training. To improve generalization, small transformation, such as shift distortion, can be manually applied to the training set. Consequently the set is augmented by example that is artificial but still form valid representation of the respective object to recognize. In this way, the CNN learns to be invariant to these types of transformation.In terms of training algorithm, in general on line Error Back propagation leads to the best performance of the resulting CNN. Therefore, this algorithm has been applied for all experimental throughout this work.

IV. PROPOSED ALGORITHM

The input to our network is the bi-cubic interpolated LR image with RGB channels, which is the same size as the original HR (ground truth) image. Proposed Algorithm flow chart diagram in figure 2

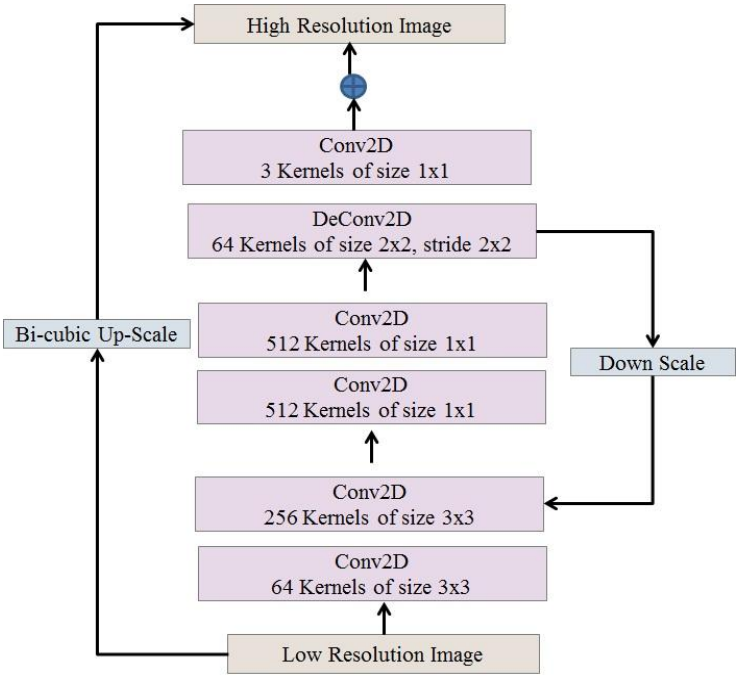

Figure 2: Flow Chart of Proposed Algorithm

Figure 1 show as flow chart of proposed algorithm. It is change low resolution image to high resolution image with different internal steps.

- **conv2D Kernel**

A filter or a kernel in a conv2D layer has a height and a width. They are generally smaller than the input image and so we move them across the whole image. The area where the filter is on the image is called the receptive field.

The Kernels deep learning Conv2D parameter, filter size, determines the dimensions of the kernel. Common dimensions include 1×1, 3×3, 5×5, and 7×7 which can be passed as (1, 1), (3, 3) , (5, 5) , or (7, 7) tuples.

- *Kernel size*

The kernel size here refers to the widthxheight of the filter mask. The max pooling layer, for example, returns the pixel with maximum value from a set of pixels within a mask (kernel). That kernel is swept across the input, subsampling it.

A fully connected layer connects every input with every output in his kernel term. For this reason kernel size = n_inputs * n_outputs. It also adds a bias term to every output bias size = n_outputs. Usually, the bias term is a lot smaller than the kernel size so we will ignore it.

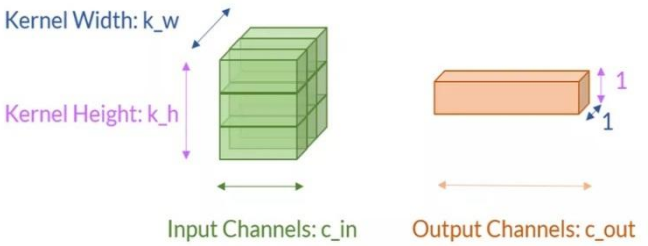

Figure 3: Kernel Size

If you consider a 3D input, then the input size will be the product the width bu the height and the depth.A convolutional layer acts as a fully connected layer between a 3D input and output. The input is the "window" of pixels with the channels as depth. This is the same with the output considered as a 1 by 1 pixel "window".

Convolution is basically a dot product of kernel (or filter) and patch of an image (local receptive field) of the same size. Convolution is quite similar to correlation and exhibits a property of translation equivariant that means if we move or translate the input and apply the convolution to it, it will act in the same manner as we first apply convolution and then translated an image.

During this learning process of CNN, you find different kernel sizes at different places in the code, then this question arises in one's mind that *whether there is a specific way to choose such dimensions or sizes*. So, the answer is no. In the current Deep Learning world, we are using the most popular choice that is used by every Deep Learning practitioner out there, and that is 3x3 kernel size. Now, another question strikes your mind, why only 3x3, and not 1x1, 2x2, 4x4, etc. Just keep reading and you will getthe most crisp reason behind this in next few minutes!!

Basically, itdivide kernel sizes into smaller and larger ones. Smaller kernel sizes consist of 1x1, 2x2, 3x3 and 4x4, whereas larger one consists of 5x5 and so on, but we use till 5x5 for 2D Convolution. In 2012, when **AlexNet** CNN architecture was introduced, it used 11x11, 5x5 like larger kernel sizes that consumed two to three weeks in training. So because of extremely longer training time consumed and expensiveness, we no longer use such large kernel sizes.

One of the reason to prefer small kernel sizes over fully connected network is that it reduces computational costs and weight sharing that ultimately leads to lesser weights for back-propagation. So then came **VGG** convolution neural networks in 2015 which replaced such large convolution layers by **3x3** convolution layers but with a lot of filters. And since then, 3x3 sized kernel has become as a popular choice. But still, ***why not 1x1, 2x2 or 4x4 as smaller sized kernel?***

- 1x1 kernel size is only used for dimensionality reduction that aims to reduce the number of channels. It captures the interaction of input channels in just one pixel of feature map. Therefore, 1x1 was eliminated as the features extracted will be finely grained and local that too with no information from the neighboring pixels.
- 2x2 and 4x4 are generally not preferred because odd-sized filters symmetrically divide the previous layer pixels around the output pixel. And if this symmetry is not present, there will be distortions across the layers which happen when using an even sized kernel, that is, 2x2 and 4x4. So, this is why we don't use 2x2 and 4x4 kernel sizes.

V. DATA SETS

Datasets we have taken in 2 categories as below,

- HR – high resolution
- LR – Low resolution

Under these categories the datasets have been taken.

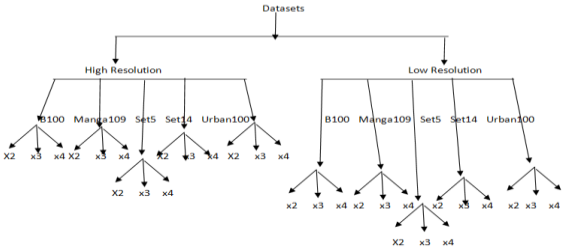

Figure 4: Types of Data Sets

In every category, again we have divided into 5 parts

1. B100
2. Manga109
3. Set5
4. Set14
5. Urban100

Again every datasets divided into 3 parts

- X2
- X3
- X

We have taken about 100 images for x2, x3 and x4 sets.So the images for B100, Manga109, Set5, Set14 and Urban100 are 300 respectively.

The images for High Resolution datasets and Low Resolution datasets are 1500 respectively.Hence total images for High Resolution datasets and Low Resolution datasets are 3000.

- Some of HR images are as follows,

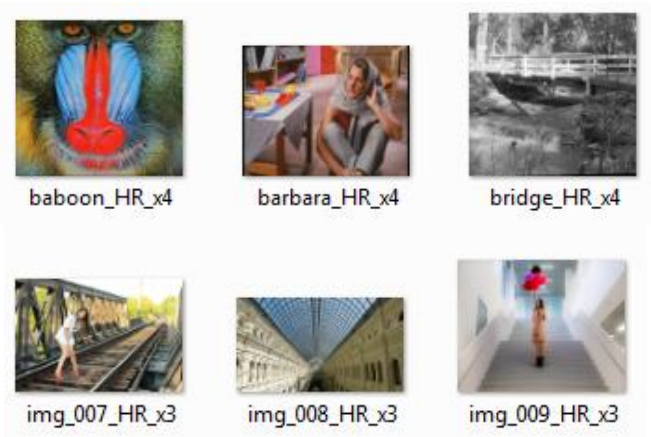

Figure 5:HR Image Data Sets

- Some of LR images are as follows,

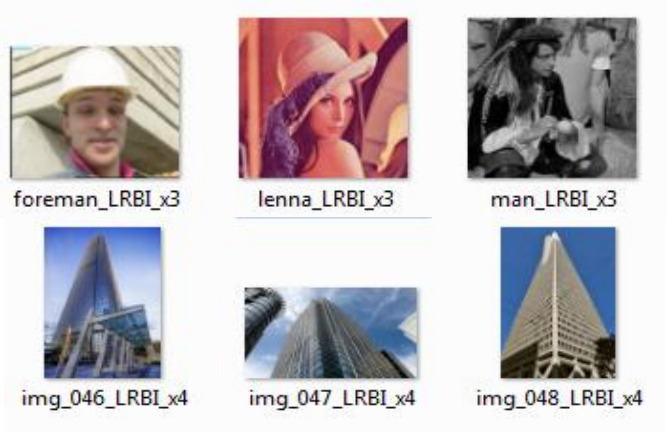

Figure 6:LR Image Data Sets

VI. Result and Discussions

It obtained the LR images as inputs by downsampling the HR images using bilinear CNN kernel.It implemented their approach and trained using the same training dataset as used by us and using the same number of epochs. This work implemented their approach and trained using the same training dataset as used by us and using the same number of epochs. Perceptual loss approach creates noticeable artefacts in its output such as the zippering on the thin edges.

The architecture can be designed as a feedforward pass of a CNN to extract features at different scales. However, in the proposed approach, both down-sampling and up-sampling operations are necessary within the feedforward pass. It is noted that any subpixel level interpolation before feeding into the network is costly and adds computational complexity since they do not bring additional information to solve the ill-posed reconstruction problem. Therefore in the proposed approach, since we have to compare the features between both the ground truth image and the reconstructed image, keeping the images as the same size is essential. Pixel wise loss forces weights to learn via back-propagation mechanism. Compared to final stage pixel wise loss, when regularizing the features, the optimization procedure adjusts the parameters quicker for the front layers and hence speeds up the convergence.

In this work compared the PSNR and SSIM parameters of Date set. This comparisons show as figure 7 to 10.

- **Peak signal to noise ratio (PSNR) and Mean Squared Error (MSE)**

The term peak signal-to-noise ratio (PSNR) is an expressed as the ratio of maximum possible value of a signal and the power of distorting noise that affects the quality of its representation. The dimensions of the two images must be the same. Mathematical representation of the PSNR is as follows:

$$PSNR = 20log_2\left(\frac{MAX_f}{\sqrt{MSE}}\right) \qquad (1)$$

Where the MSE (Mean Squared Error) is:

$$MSE = \frac{1}{mn}\sum_{0}^{m-1}\sum_{0}^{n-1}|f(i,j) - g(i,j)|^2 \qquad (2)$$

Where, f is the matrix data of the original image, g is the matrix data of processed image, m and n represents the numbers of rows and the columns of pixels of the images and i and j represents the index of row and the column respectively. MAX_f is the maximum signal in image f. The major shortcoming of PSNR metric is that it relies on numeric comparison and does not actually take into consideration the biological factors of the human vision system such as the structural similarity index (SSIM).

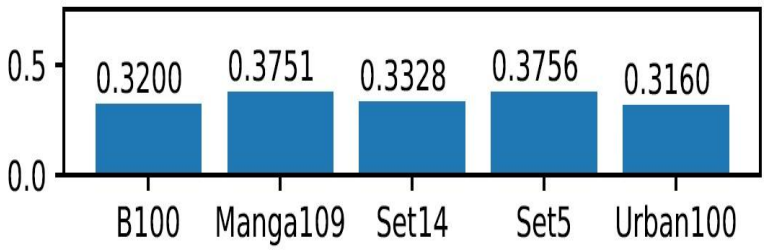

Figure 7: Date sets Compression of PSNR X2

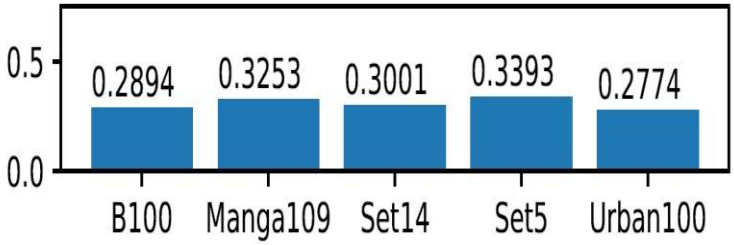

Figure 8: Date sets Compression of PSNR X3

- **Structural similarity index (SSIM)**

Wang et al. proposed SSIM metric for assessing image quality. The structural similarity (SSIM) index computes the similarity index between two images. It is more consistent with human perception as opposed to conventional methods such as mean square error (MSE). As it is correlated to human visual perception, SSIM has become a universal quality metric for image and video applications for quantitative analysis. For input image O and R, let μ_O, σ_O and σ_{OR} denote the mean of O, the variance of O, and the covariance of O and R respectively, SSIM is mathematically given as

$$SSIM = \frac{(2\mu_O\mu_R + C_1)(2\sigma_{OR} + C_2)}{(\mu_O^2 + \mu_R^2 + C_1)(\mu_O^2 + \mu_R^2 + C_2)} \quad (3)$$

Where C_1 and C_2 are constants, this metric has been suggested for the environment for quantitative analysis in Lu et al.

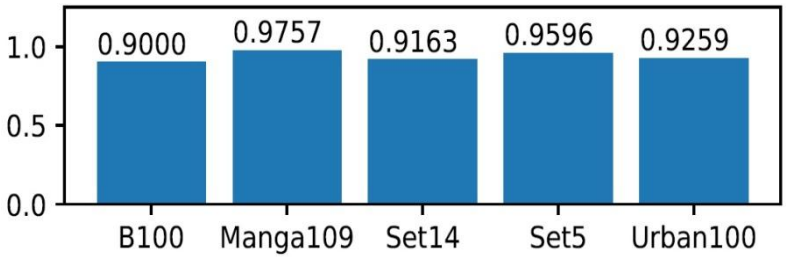

Figure 9: Date sets Compression of SSIM X2

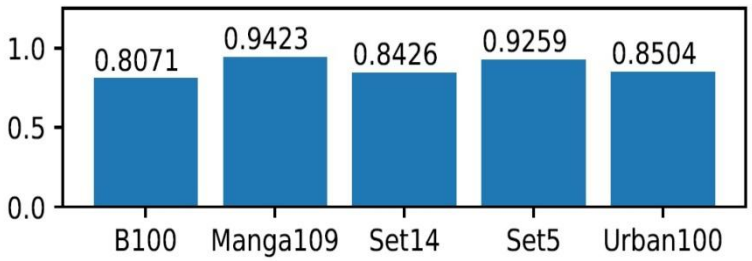

Figure 10: Date sets Compression of SSIM X3

VII. CONCLUSION

Single image super-resolution has been an attractive research area due to its vast area of applications. Over the past couple of decades, several researchers have attempted and proposed many solutions to the image superresolution problem. Generation of high resolution images from low resolution images is used in many image analysis applications for providing them with high resolution images.In this work method presented a simple and effective framework for image super resolution, and proposed a novel CNN with a self-feature based loss network that is capable of reconstructingimages of pleasing visual quality. We speculate that keeping consistency of features and network will remove the artefactscaused by features from pre-trained networks. The proposed use of self-feature loss approach provides new insights into designing CNNs capable of performing optimally in wider application areas of imaging. In this work provide, in particular, a new insight to designing novel loss estimation functions for deep learning architectures.

The results have been evaluated using subjective visual analysis as well as quantitative approaches. To quantitatively evaluate the images, PSNR and structural similarity (SSIM) were used. These metrics quantify signal strength, the amount of feature preservation, and recovery of structural features obtained image.

References

[1] Wenzhe Shi, Jose Caballero, FerencHusz´ar, JohannesTotz, Andrew Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network," Computer Vision and Patter Recognition, pp. 1874–1883, 2016.

[2] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee, "Accurate Image Super-Resolution Using Very Deep Convolutional Networks," .

[3] Christian Ledig, Lucas Theis,FerencHuszar, Jose Caballero, Andrew Aitken, AlykhanTejani, Johannes Totz, Zehan Wang, and Wenzhe Shi, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," arXiv, 2016.

[4] Kaiming He, Xiangyu Zhang, ShaoqingRen, and JianSun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, SherjilOzair, Aaron Courville, and YoshuaBengio, "Generative adversarial nets," in Advances in neural information processing systems, 2014, pp. 2672–2680.
[6] Alex Krizhevsky, IlyaSutskever, and Geoffrey E Hinton,"ImageNet Classification with Deep Convolutional Neural Networks," Advances In Neural Information Processing Systems, pp. 1–9, 2012.
[7] RaduTimofte, EirikurAgustsson, Luc Van Gool, Ming- Hsuan Yang, Lei Zhang, Bee Lim, Sanghyun Son, HeewonKim, Seungjun Nah, Kyoung Mu Lee, et al., "Ntire2017 challenge on single image super-resolution: Methods and results," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on. IEEE, 2017, pp. 1110–1121.
[8] Chao Dong, Chen Change Loy, Kaiming He, and XiaoouTang, "Image Super-Resolution Using Deep Convolutional Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 2, pp. 295–307, 2016.
[9] Justin Johnson, AlexandreAlahi, and Li Fei-Fei, "Perceptual losses for real-time style transfer and superresolution," in European conference on computer vision.Springer, 2016, pp. 694–711.
[10] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee, "Enhanced Deep Residual Networks for Single Image Super-Resolution," 2017.
[11] Dario Garcia-Gasulla,FerranPar´es, Armand Vilalta, Jonatan Moreno, Eduard Ayguad´e, Jes´usLabarta, UlisesCort´es, and ToyotaroSuzumura, "On the behavior of convolutional nets for feature extraction," Journal of Artificial Intelligence Research, vol. 61, pp. 563–592, 2018.
[12] Tsung-Yi Lin, PiotrDoll´ar, Ross Girshick, Kaiming He, BharathHariharan, and Serge Belongie, "Feature pyramid networks for object detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117–2125.
[13] J.-C. Chen, V. M. Patel, and R. Chellappa. Unconstrained face verification using deep CNN features. CoRR, abs/1508.01722, 2015.
[14] L. Kotoulas, I. Andreadis, Real-time computation of Zernike moments, IEEE Transactionson Circuits and Systems for Video Technology, 15 (2005) 801-809.
[15] L. Wiskott, J. Fellous, N. Krüger, C. Malsburg, Face Recognition by Elastic Bunch GraphMatching, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997)775-779.
[16] M. Cimpoi, S. Maji, and A. Vedaldi.Deep filter banks for texture recognition and description.In Proc. CVPR, 2015.