Enhancing The Quality of Degraded Images Using Super-Resolution CNN Algorithm

Darshan H N[1], Harsha V[2], Khoushal Devaraju[3], Nikhil Bharadwaj N[4], Mrs. Vidhya K [5]
[1, 2, 3, 4] Dept of Information Science and Engineering
[5]Asst.Prof, Dept of Information Science and Engineering
[1, 2, 3, 4, 5] East West Institute of Technology, Bengaluru, India

Abstract- *The goal of this paper, is to recuperate a high-resolution image from a low-resolution input. To make this goal possible, we will be deploying the super-resolution convolutional neural network (SRCNN) using Keras. As the title endorses, the SRCNN is a deep convolutional neural network that learns end-to-end mapping of low resolution to high resolution images. As an outcome, we can use it to recover the image quality of low-resolution images. To estimate the performance of this network, we will be using three image quality metrics: peak signal to noise ratio (PSNR), mean squared error (MSE), and the structural similarity (SSIM) index. High resolution images are essential in medical imaging for diagnosis. Numerous applications require zooming of a specific area of interest in the image wherein high-resolution images becomes vital, e.g. Crime investigations, surveillance, forensic, medical imaging and satellite imaging applications. However, high resolution images are not always accessible. By using our proposed model, we can overcome all the problems. The proposed work is later uploaded to the cloud to make it available to the public.*

Keywords- Convolutional Neural Networks (CNN); Super-Resolution; SRCNN;

I. INTRODUCTION

Today, Big Data brings paybacks to many areas of scientific research. Though, processing these huge volumes of data frequently requires extensive computing time and a large storage space. In traditional super resolution methods global feature analysis is well-thought-out to be universal, but this is not applicable to Big Data. Concentrating on useful information can make the gigantic data analysis possible and more effective. Using general and significant information and by considering imperative regions, here we propose a new super resolution approach. The training process is performed on the significant parts of the training data set under the framework of a convolutional neural network, and the reconstruction process considers significant parts separately then, according to each different demanda super resolution image will be obtained.

The proposed concept is easy to understand, but it can be achieved only through the Big Data approach with many alike images on the internet and the effective deep learning algorithm. To obtain a high-resolution reconstructed image it requires lots of testing time in the available technologies but experiments has proven that our new approach can reduce the testing time and obtain a high-quality/high resolution reconstructed image.

The vital role of the proposed work is to generate a higher resolution image from lower resolution images. High resolution image provides a high pixel density and thus more details about the original scene/image. The necessity for high resolution is common in computer vision applications to improve the performance in pattern recognition and analysis of images. High resolution images play a major role in medical imaging for diagnosis. Several applications require zooming of a specific area of interest in the image in which high resolution becomes crucial, e.g. forensic, surveillance and satellite imaging applications. The proposed SRCNN has several interesting properties. First, its structure is consciously designed with easiness in mind, and still provides higher accurateness compared with state-of-the-art example-based methods. Figure 1.1 shows a comparison on an example. Second, our method achieves fast speed for practical on-line usage even on a CPU with reasonable numbers of filters and layers. Compared to other example-based method sour method is quicker, as it is completely feed-forward and there is no need to solve any optimization problem on usage. Third, the experimentation results show that the restoration quality of the network can be further improved when(i) larger and more diverse datasets are available, and/or (ii) a larger and deeper model is used.

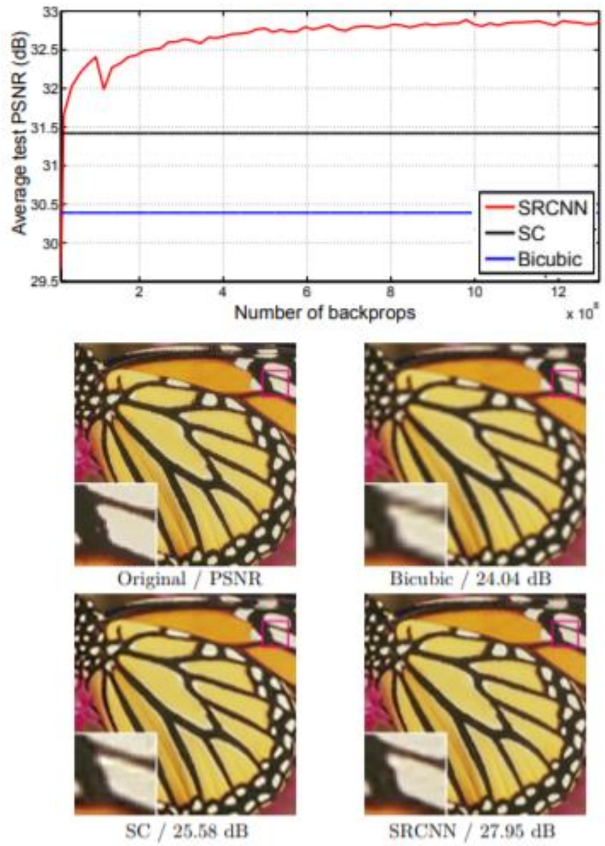

Fig. 1.1: The proposed Super-Resolution Convolutional Neural Network (SRCNN) overcomes the bicubic standards by just a few training iterations, and performs better than the sparse-coding-based method (SC) [9], [10] with moderate training. The performance can be improved with more training iterations. The proposed method provides visually enhanced/reconstructed image.

Furthermore, to achieve improved super resolution performance the proposed network can cope with three channels of color images simultaneously.

Overall, there are three main contributions of this study and they are in three aspects:

1) A convolutional neural network for image super-resolution, where the end-to-end mapping between low and high-resolution images is directly learnt by the network, with slight pre/post processing beyond the optimization.
2) The guidance for the design of the network structure is provided by establishing a relationship between our deep learning-based SR method and the traditional sparse-coding-based SR methods.
3) To achieve good speed and quality in the classical computer vision problem of super resolution, deep learning is very useful.

A preliminary version of this work was presented earlier [12]. The proposed work adds some improvement to the initial version in substantial ways. Firstly, in the non-linear mapping layer we are introducing a larger filter size and explore deeper structures by adding nonlinear mapping layers, by doing so we improve the SRCNN. Secondly, extending the SRCNN color processing channels to three color channels (either in YCbCr, RGB, BGR color space) concurrently. Experimentally, we demonstrate that performance can be upscaled in comparison to the single-channel network. Thirdly, instinctive explanations and significant new analysis are appended to the initial results. We also extend the original experiments from Set5 [11] and Set14 [13] test images to BSD200 [14] (200 test images). Furthermore, using different evaluation metrics we confirm that our model out performs when compared with numerous recently published methods.

II. RELATED WORK

2.1 Image Super-Resolution

A category of state-of-the-art SR approaches [15], [16], [17], [18], [19], [20], [21] learn a mapping between low/high-resolution patches. These studies differ on the most proficient method to get familiar with a compact dictionary or various space to relate low/high-resolution patches, and on how portrayal plans can be led in such spaces. In the work of Freeman et al. [22], the nearest neighbor (NN) of the input patch is found in the low-resolution space, with its corresponding high-resolution patch which is used for image reconstruction and the dictionaries are directly presented as low/high-resolution patch pairs. Chang et al. [23]introduce a manifold embedding technique as an alternative to the NN strategy. In Yang et al.'s work [17],[18] the nearest neighbor (NN) facilitates to a more accurate sparse coding formulation. The sparse coding-based method with its numerous upscaled features [25], [26] are amid the state-of-the-art super resolution (SR) methods these days. In the above methods, the patches are essential for optimization process, the patch extraction and aggregation steps are well-thought-out as pre/post-processing and handled distinctly.

2.2 Convolutional Neural Network.

Convolutional neural networks (CNN) date back decades [28] and due to its success in image classification [27], [29] it has shown huge popularity. They have also been successfully applied to other computer vision fields, such as object detection [30], [31], [32], face recognition [33], and pedestrian detection [34]. There are numerous factors which are of vital importance in this progress: (i) the efficient training implementation on modern powerful GPUs [29], (ii) the proposal of the Rectified Linear Unit (ReLU) [35] which makes convergence much faster while still presents good quality [29], and (iii) the easy access to an abundance of data (like ImageNet [4]) for training larger models. Our method also benefits from these improvements.

2.3 Deep Learning for Image Restoration.

There has been a couple of studies witnessing image reconstruction utilizing deep learning techniques. The multi-layer perceptron (MLP), whose all layers are fully-connected, is applied for natural image denoising [2]and post-deblurring denoising [1]. Which is more closely related to our work, the convolutional neural network is applied for natural image denoising [36] and removing noisy patterns (dirt/rain) [37]. These restoration issues are pretty much denoising-driven. Cui *etal.* [38] On the contrary, the image super-resolution problem has not witnessed the usage of deep learning techniques to the best of our knowledge.

III. PROPOSED SYSTEM

3.1 Architecture

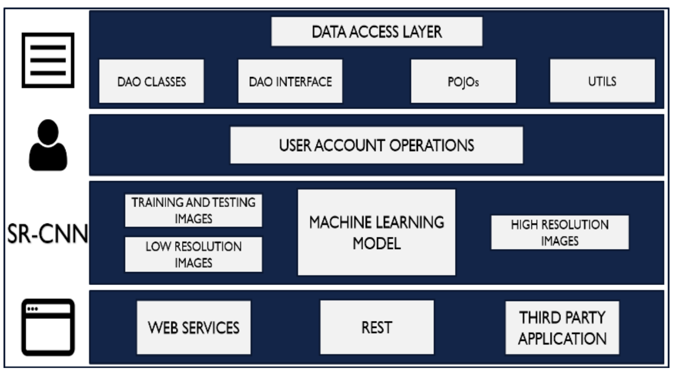

Fig 3.1: The proposed architecture diagram of Super Resolution Convolutional Neural Network with third party applications and web services.

The main goal of the proposed work is to deploy this model onto the cloud to make it available to the public, where it can be used and operated at ease. Data access layer exposes all the possible operations on the database to the outside world. It consists POJOs, Utils, DAO classes, DAO interfaces as internal components. All the remaining modules in the proposed work will be interacting with the DAO layer according to their data access needs. The account operations module delivers the subsequent functionalities to the end users. Register/signup a new account, Login and Logout, Edit the existing Profile, Change Password, Forgot Password and receive the current password over an email, Delete an existing Account. To provide the above functionalities the account operations module will be re-using the DAO layer.

We implement a deep learning method for single image super-resolution (SR). Our method directly learns an end-to-end mapping between the low and high-resolution images.

This mapping is indicated as a deep convolutional neural network (CNN) that takes the low-resolution image as input and yields the high-resolution one. We perform Training and Testing the model for accuracy. Moreover, the model will be trained using numerous datasets and tested for finding the accuracy of the model. Optimization will be done to improve the accuracy if needed. In machine learning, the usual task is to study and construct the algorithms that can learn from and make predictions on data.

In Web service implementation we implement the web services to expose the model to the outside world. We implement RESTful APIs for exposing the model to other apps/clients.

A third-party application has been implemented to demonstrate the usage of the web services to the customers. In this application, we implement four steps

User Identity: We collect the user's first name and the last name.

Contact Information: We collect the email ID and mobile number of the client.

Proof: We will send an OTP to customer and ask them to enter it to prove the identity.

Execution: User uploads an input image here and clicking on Run button will invoke the web service implemented.

The downloadable image link will be displayed back to the client once the result is available

We can use the proposed work to enhance the quality of the degraded or low-resolution images. We will be utilizing three image quality metrics to measure the performance of the model and they are: Peak Signal to Noise Ratio (PSNR), Mean Squared Error (MSE) and Structural Similarity Index (SSIM). Additionally, in this work we will be pre and post processing our images by using OpenCV, the open Source Computer Vision Library. It was originally developed by Intel and is utilized for numerous real-time computer vision applications. Here, the images are often converted back and forth between the three-color channels i.e. BGR, RGB and YCrCb color spaces. Since the SRCNN network was trained on the luminance (Y) channel in the YCrCb color space and hence this process becomes essential.

In this work, we will be performing the following tasks:

- Usage of the PSNR, MSE, and SSIM image quality metrics,
- Processing the images using OpenCV,
- Conversion between the RGB, BGR, and YCrCb color spaces,
- Building deep neural networks in Keras,
- Deploying and evaluating the SRCNN network

3.2 Various divisions in the project

- User Profile Operations.
- Implementation of SR-CNN Algorithm for image super resolution.
- Training and Testing the model for accuracy.
- Implementation of RESTful APIs for exposing the model to other apps/clients.
- User Interface design for the model.
- Cloud based deployment process of the model.

IV. CONVOLUTIONAL NEURAL NETWORKS FOR SUPER-RESOLUTION

4.1 Formulation

Considering the initial work [56], we are going to upsize a given low-resolution image or input into desired size through bicubic interpolation and this is the only pre-processing we perform. Now, let us consider the interpolated image as Y and the goal is to recover from Y an image F(Y) that is very similar to the original image but in high-resolution image X.To keep it simple, Y is called low-resolution" image, though it has the same size as X i.e. "high-resolution" image. The mapping F consists of three operations [56]:

1) **Patch extraction and representation:** By performing this operation, the patches (overlapping) are extracted from the image Y i.e. "low-resolution" image. Later, each extracted patch is represented as a high-dimensional vector. These high-dimensional vectors consist of set of feature maps, of which number equals to the dimensionality of the vectors.
2) **Non-linear mapping:** In this operation each high-dimensional vector which is obtained in patch extraction is non-linearly mapped to another high-dimensional vector. Each high-dimensional vector is the representation of a high-resolution patch. These vectors consist another set of feature map.
3) **Reconstruction:** Here, the above high-resolution patch-wise representations are summed up to generate the high-resolution image. The generated high-resolution image is expected that it should be similar to the original image.

A reference diagram is shown in Figure 4.1.

4.1.1 Patch extraction and representation

A widespread approach in image restoration (e.g., [40]) is densely extracting the patches and then representing them by a set of bases such as PCA, DCT, Haar, etc. which are pre-trained. This is similar to convolving the image by a set of filters. With respect to our design, to optimize the network we involve the optimization of these bases. Hence, the first layer of the design is expressed as an operation F1:

$$F1(Y) = \max(0, W1 * Y + B1), \quad (1)$$

where W1 and B1 represent the filters and biases respectively, and'∗' denotes the convolution operation. Here, W1 corresponds to n1 filters of support c×f1 ×f1, where c is the number of channels in the input image, f1 is the spatial size of a filter. Intuitively, W1 applies n1 convolutions on the image, and each convolution hasa kernel size c × f1 × f1.Here the output which we get is composed of n1 feature maps. B1 is an n1-dimensional vector, who's each element is associated with a filter. Furthermore, we are going to apply the Rectified Linear Unit (ReLU, max (0, x)) [35] on the filter responses.

4.1.2 Non-linear mapping

In the first layer [56], we extract an n1-dimensional feature for each patch. Further in the second operation, we are going to map each of these n1-dimensional vectors into an n2-dimensional one. This is similar to applying n2 filters which have a trivial spatial support 1 × 1. This interpretation is only valid for 1×1 filter. However, it is easy to simplify to larger filters like 3 × 3 or 5 × 5.

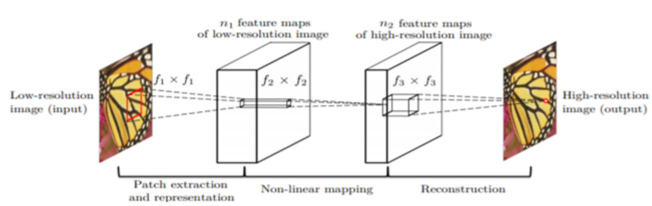

Fig 4.1: Given a low-resolution image Y, the first convolutional layer of the SRCNN extracts a set of feature maps. The second layer maps these feature maps nonlinearly to high-resolution patch representations. The last layer combines the predictions within a spatial neighborhood to produce the final high-resolution image F(Y).

Considering these circumstances, the non-linear mappingis not on a patch of the input map instead, it is on a 3 × 3 or 5 × 5 “patch” of the feature. The operation of the second layer is:

$$F2(Y) = \max(0, W2 * F1(Y) + B2), (2)$$

Here [56] W2 contains n2 filters of size n1 ×f2 ×f2, and B2 is n2-dimensional. Each of the output i.e. n2-dimensional vectors is conceptually a portrayal of a high-resolution patch that will be utilized for reconstruction. It is possible to add more Givenconvolutional layers to increase the non-linearity. But this can increase the complexity of the model (n2 × f2 × f2 × n2 parameters for one layer), and thus demands more training time.

4.1.3 Reconstruction

In the traditional methods [56], the predicted overlapping high-resolution patches are often averaged to produce the final full image. The averaging can be considered as a pre-defined filter on a set of feature maps (in which each position is the “flattened” vector form of a high-resolution patch). And by this, we will define a convolutional layer to produce a high-resolution image:

$$F(Y) = W3 * F2(Y) + B3. \qquad (3)$$

W3 corresponds to the c filters of size n2 × f3 × f3, and B3 is a c-dimensional vector. If the high-resolution representational patches are in the image domain (i.e., we can simply reshape each representation to form the patch), we expect it to act like an averaging filter, if the representation patches are in some other domains (e.g., coefficients in terms of some bases), we expect that W3 behaves like first projecting the coefficients onto the image domain and then averaging. Either way, the W3 is a set of linear filters. Even though the above three operations are motivated by different intentions, they all lead to the same form as a convolutional layer. All three operations will be put together to form a convolutional neural network. In the proposed model, all of the weights and biases representing filters are to be optimized. Though the overall structure is very compact, the proposed SRCNN model is carefully developed with widespread experience which in turn resulted from significant progresses in super-resolution [41], [42]. We have detailed the relationship in the next section.

4.1.4 Loss Function

Here, the loss functions are used to measure the difference between the High-Resolution image generated and the ground truth High Resolution image. This difference or the error will be then used to optimize the supervised learning model. A several classes of loss functions will exist, where each of them will penalize a different aspect of the generated image. Often, more than one loss function will be used for weighting and summing up the errors obtained from the loss function individually. This will enable the proposed model to focus on aspects contributed by multiple loss functions simultaneously.

4.2 Relationship with Sparse Coding

We show that the sparse-coding based SR methods [41], [42] can

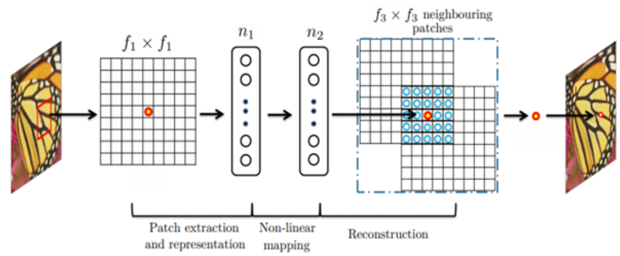

Fig 4.2: An illustration of sparse-coding-based methods in the view of a convolutional neural network.

be viewed as a convolutional neural network. Figure 4.2 shows can an illustration. In the sparse-coding-based methods, let us consider that an f1 × f1 low-resolution patch is extracted from Sign [43], will first project the patch onto a (low-resolution) dictionary. We can apply n1 linear filters (f1 × f1) on the input image, only if the dictionary size is n1. It is shown as the left part of Figure 4.2. The n1 coefficients will be processed by the sparse coding solver iteratively. n2 coefficients will be the output of this solver, and here in the case of spare coding usually n2 = n1. The output of the n1 coefficients i.e. n2 coefficients, will be the representation of the high-resolution patch. Therefore, the sparse coding solver behaves as a special case of a non-linear mapping operator, whose spatial support is 1 × 1. Have a look at the middle part of Figure 4.2. The sparse coding solver is an iterative algorithm since it is not a feed-forward. But the non-linear operator is fully feed-forward and can be computed efficiently. The non-linear operator can be considered as a pixel-wise fully-connected layer, if we set f2 = 1. To note this, “the sparse coding solver" in SRCNN refers to the first two layers, but not only the second layer or the activation function (ReLU). Hence, the nonlinear operation in SRCNN is also well optimized through the learning process.

The n2 coefficients after sparse coding are projected to another dictionary (high-resolution) to produce a high-resolution patch. Then the high-resolution patches which are overlapping are averaged. As we have mentioned, this is equivalent to linear convolutions on the n2 feature maps. The sparse-coding based SR method can be viewed as a kind of convolutional neural network (but with a different non-linear mapping through the discussions we have made. But in the sparse-coding-based SR methods, not all operations have been considered in the optimization. In our convolutional neural network, the low-resolution dictionary, high-resolution dictionary, non-linear mapping, also with mean subtraction and averaging, are all involved in the filters to be optimized. Hence, we can tell that our method optimizes an end-to-end mapping that consists of all operations.

The above relation can also help us to design hyper parameters. For example, we can set the filter size of the last layer to be smaller than that of the first layer, and hence we depend more on the central part of the high-resolution patch. We can also set n2 < n1 because it is expected to be sparser. A typical and basic setting is f1 = 9, f2 = 1, f3 = 5, n1 = 64, and n2 = 32 (we evaluate additional settings in the experiment section). Through the estimation we got to know that high-resolution pixel utilizes the information of (9 5 − 1)2 = 169 pixels. Clearly, the information exploited is comparatively larger than that used in existing approaches for reconstruction, e.g., using (5 5−1)2 = 81 pixels5 [44], [42]. And hence we can say that, this is one of the reasons why the SRCNN gives predominant performance.

4.3 Model Training

Here, the model will be trained using the datasets and tested for finding the accuracy of the model. Optimization will be done to improve the accuracy if needed. In machine learning, a common task is the study and construction of algorithms that can learn from and make predictions on data. Such algorithms work by making data-driven predictions or decisions, through building a mathematical model from input data. The data used to build the final model usually comes from multiple data sets. In particular, three datasets are commonly used in different stages of the creation of the model. The model is initially fit on a training dataset, that is a set of examples used to fit the parameters (e.g. weights of connections between neurons in artificial neural networks) of the model.

V. EXPERIMENTS

First, we study the impact on the performance of the model by using different datasets. Next, we examine the filters that are learnt by our approach. We then inspect different architecture designs of the network, and also studying the relationship between super-resolution performance and factors like depth, number of filters, and filter sizes. Eventually, we compare our proposed model with the recent state-of-the-art approaches both quantitively and qualitatively. Following [45], super-resolution is only applied on the luminance channel (Y channel in YCbCr color space), so c = 1 in the first/last layer, and performance (e.g., PSNR and SSIM) is evaluated on the Y channel. Lastly, we extend the network to cope with color images and evaluate the performance on different channels.

5.1 Training Data

As we have told in the introduction, deep learning is benefitted from big data training. To compare, we use a relatively small training set [46], [42] that consists of 91 images, and considerably a large training dataset that consists of 395, 909 images from the ILSVRC 2013 ImageNet detection training partition. The size of the training sub-images is i.e. f_{sub} = 33. Therefore, the 91-image dataset can be disintegrated into 24,800 sub-images, which are taken out from original images with a stride of 14. Even using a stride of 33 the ImageNet provides over 5 million sub-images. Here, we use only the basic network setting, i.e. f1 = 9, f2 = 1, f3 = 5, n1 = 64, and n2 = 32. We use the Set5 [47] as the validation set. Even if we use the larger Set14 set [48] We observe a similar trend. The upscaling factor is 3. We use the sparse-coding-based method [42] as our baseline, which achieves an average PSNR value of 31.42 dB.

Since the number of backpropagations is the same, the training time on ImageNet is same as the 91-image dataset. The SRCNN + ImageNet achieves 32.52 dB, higher than 32.39 dB yielded by that trained on 91 images with the same number of backpropagations (i.e.,8 × 108). If we observe, the results show that the performance of the SRCNN can be further boosted using a larger training dataset. Nevertheless, we adopt big data approach, which contains more diverse data.

5.2 Comparison with State-of-the Art Approaches

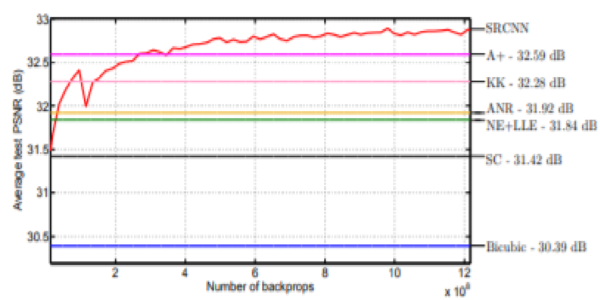

Fig. 5.1: The test convergence curve of SRCNN and results of other methods on the Set5 dataset.

Here, we portray the quantitative and qualitative analysis as well as results of our proposed method by comparing it with state-of-the-art approaches in all the experiments. Note that the results of our SRCNN are based on the checkpoint of 8 × 108 backpropagations. Specifically, the average gains [56] on PSNR achieved by SRCNN are 0.15 dB, 0.17 dB, and 0.13 dB, higher than the next best approach for the upscaling factor 3. A+ [50], on the three datasets. When we take a look at other evaluation metrics, we get to know that bicubic interpolation on IFC and NQM stands higher compared to SC. Obviously the aftereffects of SC are more outwardly satisfying than that of bicubic introduction. This signifies that these metric does not tell the reality of te image quality. But the SRCNN achieves the best performance among all methods and scaling factors regardless of these two metrics.

5.3 Running time

Fig. 5.2 shows the comparison between running time and several state-of-the-art methods, also with their restoration performance on Set14from the corresponding authors MATLAB+MEX implementation, whereas ours are in pure C++.

Since all images go through the same number of convolutions, the processing time of our proposed method is highly linear to the test image resolution. Our strategy is constantly an exchange off among performance and speed. On the vison of portraying it, we train three networks for comparison, which are 9-1-5, 9-3-5, and 9-5-5.By doing this

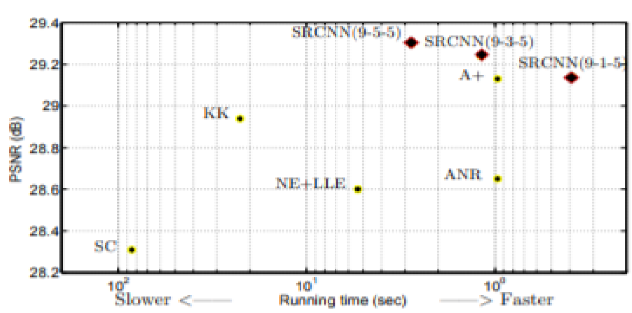

Fig. 5.2: The proposed SRCNN achieves the state of-the-art super-resolution quality, whilst maintains high and competitive speed in comparison to existing external example-based methods. The chart is based on Set14 results summarized in Table 2.

By doing this we got to know that the fastest network is 9-1-5, while it still achieves better performance than the next state-of-the-art A+. In comparison with 9-1-5 network Other methods are even slower. Our method is completely feed-forward but other methods need to solve complex optimization problems on usage (e.g., sparse coding or embedding). The 9-5-5 network achieves the best performance but it requires very long running time. The test-time speed of our CNN can be additionally quickened from numerous ways, e.g., approximating or simplifying the trained networks [51], [52], [53] with possible slight degradation in performance.

5.4 Experiments on Color Channels

In the previous methods, to resolve color images we follow the conventional approach i.e. we first convert the color images into YCbCr space. Super Resolution algorithms are applied only on the Y channel, but the Cb, Cr channels are upsized through bicubic interpolation. If we jointly consider all the three channels in the process, we can improve the performance of the super-resolution.

Without making any changes to the learning mechanism and network design, our method can accept more channels. To be precise, by setting the input channels to c = 3 it can simultaneously deal with three channels. In the following experiments, we tryout numerous training strategies for color image super-resolution also evaluate their performance on different channels.

5.5Implementation details.

The training is performed on the 91-image dataset, and testing is conducted on the Set5 [47]. The network settings are: $c = 3$, $f1 = 9$, $f2 = 1$, $f3 = 5$, $n1 = 64$, and $n2 = 32$. Here we only evaluate the performance of upscaling factor 3, as we have proved the effectiveness of SRCNN on different scales.

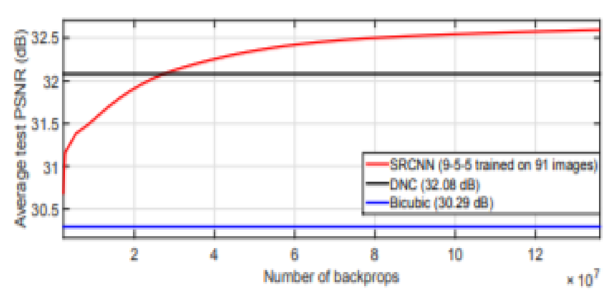

Fig. 5.3: The test convergence curve of SRCNN and the result of DNC on the Set5 dataset.

We compare our method with the state of-art color SR method – KK [54], [56].

- **Y only:** This is basically a base method where c=1 i.e. it's a single-channel network trained only on luminance channel. And using bicubic interpolation, the other channels i.e. Cb, Cr are upsized.

- **YCbCr:** The training is performed by considering the three channels of the YCbCr space.
- **Y pre-train:** First, as the loss to pre-train the network we only use the MSE of the Y channel only to guarantee the performance on the Y channel. Then to fine-tune the parameters, we employ the MSE of all channels.
- **CbCr pre-train:** As the loss to pre-train the network, we use the MSE of the Cb, Cr channels, then fine-tune the parameters on all channels.
- **RGB:** The training is performed on the three channels of the RGB space.

Table 1: The average results of PSNR (dB), SSIM, IFC, NQM, WPSNR (dB) and MSSIM on the Set5 dataset.

Eval. Mat	Scale	Bicubic	SC [42]	NE+LLE [55]	KK [54]	ANR [46]	A+ [46]	SRCNN
PSNR	2	33.66	-	35.77	36.20	35.83	36.54	36.65
	3	30.39	31.42	31.84	32.28	31.92	32.59	32.75
	4	28.42	-	29.61	30.03	29.69	30.28	30.47
SSIM	2	0.9299	-	0.9490	0.9511	0.9499	0.9544	0.9542
	3	0.8682	0.8821	0.8956	0.9033	0.8698	0.9088	0.9090
	4	0.8104	-	0.8406	0.8541	0.8419	0.8603	0.8628
IFC	2	6.10	-	7.34	6.87	8.09	8.48	8.05
	3	3.52	3.16	4.0	4.14	4.52	4.84	4.58
	4	2.35	-	2.94	2.81	3.02	3.26	3.01
NQM	2	36.73	-	42.90	39.49	43.28	44.58	41.14
	3	27.54	27.29	32.77	32.10	33.10	34.48	33.21
	4	21.42	-	25.56	24.99	25.72	26.97	25.96
WPSNR	2	50.06	-	58.45	57.15	58.61	60.06	59.49
	3	41.65	43.64	45.81	46.22	46.02	47.17	47.11
	4	37.21	-	39.85	40.40	40.01	41.03	41.13
MSSSIM	2	0.9915	-	0.9953	0.9953	0.9954	0.9960	0.9958
	3	0.9754	0.9797	0.9841	0.9853	0.9844	0.9867	0.9867
	4	0.9516	-	0.9666	0.9695	0.9672	0.9720	0.9725

Table 2: The average results of PSNR (dB), SSIM, IFC, NQM, WPSNR (dB) and MSSIM on the Set14 dataset.

Eval. Mat	Scale	Bicubic	SC [42]	NE+LLE [55]	KK [54]	ANR [46]	A+ [46]	SRCNN
PSNR	2	30.23	-	31.76	32.11	31.80	32.28	32.45
	3	27.54	28.31	28.60	28.94	28.65	29.13	29.30
	4	26.00	-	26.81	27.14	26.85	27.32	27.50
SSIM	2	0.8687	-	0.8993	0.9026	0.9004	0.9056	0.9067
	3	0.7736	0.7954	0.8076	0.8132	0.8093	0.8188	0.8215
	4	0.7019	-	0.7331	0.7419	0.7352	0.7491	0.7513
IFC	2	6.09	-	7.59	6.83	7.81	8.11	7.76
	3	3.14	2.98	4.14	3.83	4.32	4.45	4.26
	4	2.23	-	2.71	2.57	2.78	2.94	2.74
NQM	2	40.98	-	41.34	38.86	41.79	42.61	38.95
	3	33.15	29.06	37.12	35.23	37.22	38.24	35.25
	4	26.15	-	31.17	29.18	31.27	32.31	30.46
WPSNR	2	47.64	-	54.47	53.85	54.57	55.62	55.39
	3	39.72	41.66	43.22	43.59	43.36	44.25	44.32
	4	35.71	-	37.75	38.26	37.85	38.72	38.87
MSSSIM	2	0.9813	-	0.9886	0.9890	0.9888	0.9896	0.9897
	3	0.9512	0.9595	0.9643	0.9653	0.9647	0.9669	0.9675
	4	0.9134	-	0.9317	0.9338	0.9326	0.9371	0.9376

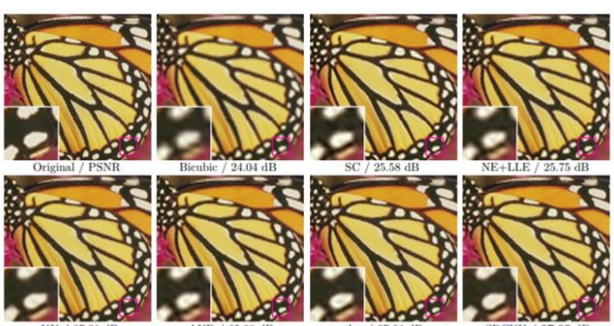

Fig.5.4: The “butterfly” image from Set5 with an upscaling factor 3.

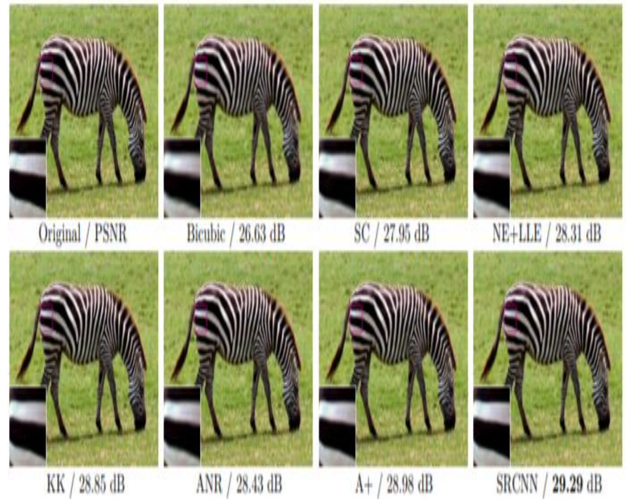

Fig.5.5: The “zebra” image from Set14 with an upscaling factor 3.

VI. CONCLUSIONS

We have presented a novel deep learning approach for Super-Resolution Convolutional Neural Network (SRCNN) which is deployed over the web to make it available to the public and also this is advantageous over other state-of-the-art approaches. We have shown that sparse-coding-based SR methods can also be redeveloped into a deep convolutional neural network. The proposed approach i.e. SRCNN, learns an end-to-end mapping between low and high-resolution images, with little extra pre/post-processing beyond the optimization. Since it has a lightweight structure, the SRCNN has shown a remarkable performance than the state-of-the-art approaches. We estimate that by exploring more filters and different training strategies additional performance can be further gained. With its advantages of lightweight design, robustness and easy availability over cloud makes this work portable and east to use. This work is very useful in upscaling the degraded images taken by the satellite and can play a vital role is solving some of the crime investigations etc. this work can still be trained and extended to medical imaging. The experiment results show that the super resolution image reconstructed with proposed method is superior to the other methods considered from both subjective and objective perspectives.

VII. ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped me in carrying out this project.

We would like to take this opportunity to thank them all. First and foremost we would like to thank Mrs.Vidhya K, Asst. Professor, EWIT, for her moral support and valuable suggestions and expert advice, towards completing our project work. and constantly guiding us to organize the report in a systematic manner.

We thank our Parents, and all the Faculty members of Department of Information science & Engineering for their constant support and encouragement. Last, but not the least, we would like to thank our peers and friends who provided us with valuable suggestions to improve our project.

REFERENCES

[1] Schuler, C.J., Burger, H.C., Harmeling, S., Scholkopf, B.: A machine learning approach for non-blind image deconvolution. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1067–1074 (2013)
[2] Burger, H.C., Schuler, C.J., Harmeling, S.: Image denoising: Can plain neural networks compete with BM3D? In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2392–2399 (2012)
[3] Dong C, Chen C L, He K, et al. Image SuperResolution Using Deep Convolutional Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2016, 38(2), pp295-307.
[4] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009)
[5] Tai Y, Yang J, Liu X. Image Super-Resolution via Deep Recursive Residual Network[C], IEEE Computer Vision and Pattern Recognition. IEEE, 2017.
[6] Denton E, Chintala S, Szlam A, et al. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks[J]. 2015:1486-1494.
[7] Ledig C, Theis L, Huszar F, et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network[J]. 2016.
[8] Bee Lim, Sanghyun Son, Heewon Kim, et al. Enhanced Deep Residual Networks for Single Image Super-Resolution[C], Computer Vision and Pattern Recognition Workshops. IEEE, 2017:1132-1140.
[9] Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE Transactions on Image Processing 19(11), 2861–2873 (2010)
[10] Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing 54(11), 4311–4322 (2006)
[11] Bevilacqua, M., Roumy, A., Guillemot, C., Morel, M.L.A.: Low complexity single-image super-resolution based on nonnegative neighbor embedding. In: British Machine Vision Conference (2012)
[12] Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: European Conference on Computer Vision, pp. 184–199 (2014)
[13] Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Curves and Surfaces, pp. 711–730 (2012)
[14] Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: IEEE International Conference on Computer Vision. vol. 2, pp. 416–423 (2001)
[15] Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision. IJCV 40(1), 25–47 (2000)
[16] Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: CVPR (2004)
[17] Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: CVPR, pp. 1–8 (2008)
[18] Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. TIP 19(11), 2861–2873 (2010)
[19] Bevilacqua, M., Roumy, A., Guillemot, C., Morel, M.L.A.: Low-complexity singleimage super-resolution based on nonnegative neighbor embedding. In: BMVC (2012)
[20] Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: ICCV, pp. 1920–1927 (2013)
[21] Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse representations. In: Boissonnat, J.-D., Chenin, P., Cohen, A., Gout, C., Lyche, T., Mazure, M.-L., Schumaker, L. (eds.) Curves and Surfaces 2011. LNCS, vol. 6920, pp. 711–730. Springer, Heidelberg (2012)
[22] Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. Computer Graphics and Applications 22(2), 56–65 (2002)
[23] Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: CVPR (2004)
[24] Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: ICCV, pp. 1920–1927 (2013)
[25] Yang, J., Wang, Z., Lin, Z., Cohen, S., Huang, T.: Coupled dictionary training for image super-resolution. TIP 21(8), 3467–3478 (2012)
[26] He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual

recognition. In: European Conference on Computer Vision, pp. 346–361 (2014)

[27] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation pp. 541–551 (1989)

[28] Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 1097–1105 (2012)

[29] Ouyang, W., Luo, P., Zeng, X., Qiu, S., Tian, Y., Li, H., Yang, S., Wang, Z., Xiong, Y., Qian, C., et al.: Deepid-net: multi-stage and deformable deep convolutional neural networks for object detection. arXiv preprint arXiv:1409.3505 (2014)

[30] Szegedy, C., Reed, S., Erhan, D., Anguelov, D.: Scalable, highquality object detection. arXiv preprint arXiv:1412.1441 (2014)

[31] Zhang, N., Donahue, J., Girshick, R., Darrell, T.: Part-based RCNNs for fine-grained category detection. In: European Conference on Computer Vision. pp. 834–849 (2014)

[32] Sun, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification. In: Advances in Neural Information Processing Systems. pp. 1988–1996 (2014)

[33] Ouyang, W., Wang, X.: Joint deep learning for pedestrian detection. In: IEEE International Conference on Computer Vision. pp. 2056–2063 (2013)

[34] Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: International Conference on Machine Learning. pp. 807–814 (2010)

[35] Jain, V., Seung, S.: Natural image denoising with convolutional networks. In: Advances in Neural Information Processing Systems. pp. 769–776 (2008)

[36] Eigen, D., Krishnan, D., Fergus, R.: Restoring an image taken through a window covered with dirt or rain. In: IEEE International Conference on Computer Vision. pp. 633–640 (2013)

[37] Cui, Z., Chang, H., Shan, S., Zhong, B., Chen, X.: Deep network cascade for image super-resolution. In: European Conference on Computer Vision, pp. 49–64 (2014)

[38] Image Super-Resolution Using Deep Convolutional Networks. Chao Dong, Chen Change Loy, Member, IEEE, Kaiming He, Member, IEEE, and Xiaoou Tang, Fellow, IEEE

[39] Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing 54(11), 4311–4322 (2006)

[40] Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8 (2008)

[41] Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE Transactions on Image Processing 19(11), 2861–2873 (2010)

[42] Lee, H., Battle, A., Raina, R., Ng, A.Y.: Efficient sparse coding algorithms. In: Advances in Neural Information Processing Systems. pp. 801–808 (2006)

[43] Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning lowlevel vision. International Journal of Computer Vision 40(1), 25–47 (2000)

[44] Timofte, R., De Smet, V., Van Gool, L.: A+: Adjusted anchored neighborhood regression for fast super-resolution. In: IEEE Asian Conference on Computer Vision (2014)

[45] Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: IEEE International Conference on Computer Vision. pp. 1920–1927 (2013)

[46] Bevilacqua, M., Roumy, A., Guillemot, C., Morel, M.L.A.: Lowcomplexity single-image super-resolution based on nonnegative neighbor embedding. In: British Machine Vision Conference (2012)

[47] Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Curves and Surfaces, pp. 711–730 (2012)

[48] Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 1097–1105 (2012)

[49] Timofte, R., De Smet, V., Van Gool, L.: A+: Adjusted anchored neighborhood regression for fast super-resolution. In: IEEE Asian Conference on Computer Vision (2014)

[50] Denton, E., Zaremba, W., Bruna, J., LeCun, Y., Fergus, R.: Exploiting linear structure within convolutional networks for efficient evaluation. In: Advances in Neural Information Processing Systems (2014)

[51] Jaderberg, M., Vedaldi, A., Zisserman, A.: Speeding up convolutional neural networks with low rank expansions. In: British Machine Vision Conference (2014)

[52] Mamalet, F., Garcia, C.: Simplifying convnets for fast learning. In: International Conference on Artificial Neural Networks, pp. 58–65. Springer (2012)

[53] Kim, K.I., Kwon, Y.: Single-image super-resolution using sparse regression and natural image prior. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(6), 1127–1133 (2010)

[54] Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: IEEE Conference on Computer Vision and Pattern Recognition (2004)

[55] Image Super-Resolution Using Deep Convolutional Networks. Chao Dong, Chen Change Loy, Member, IEEE, Kaiming He, Member, IEEE,and Xiaoou Tang, Fellow, IEEE, (2015).