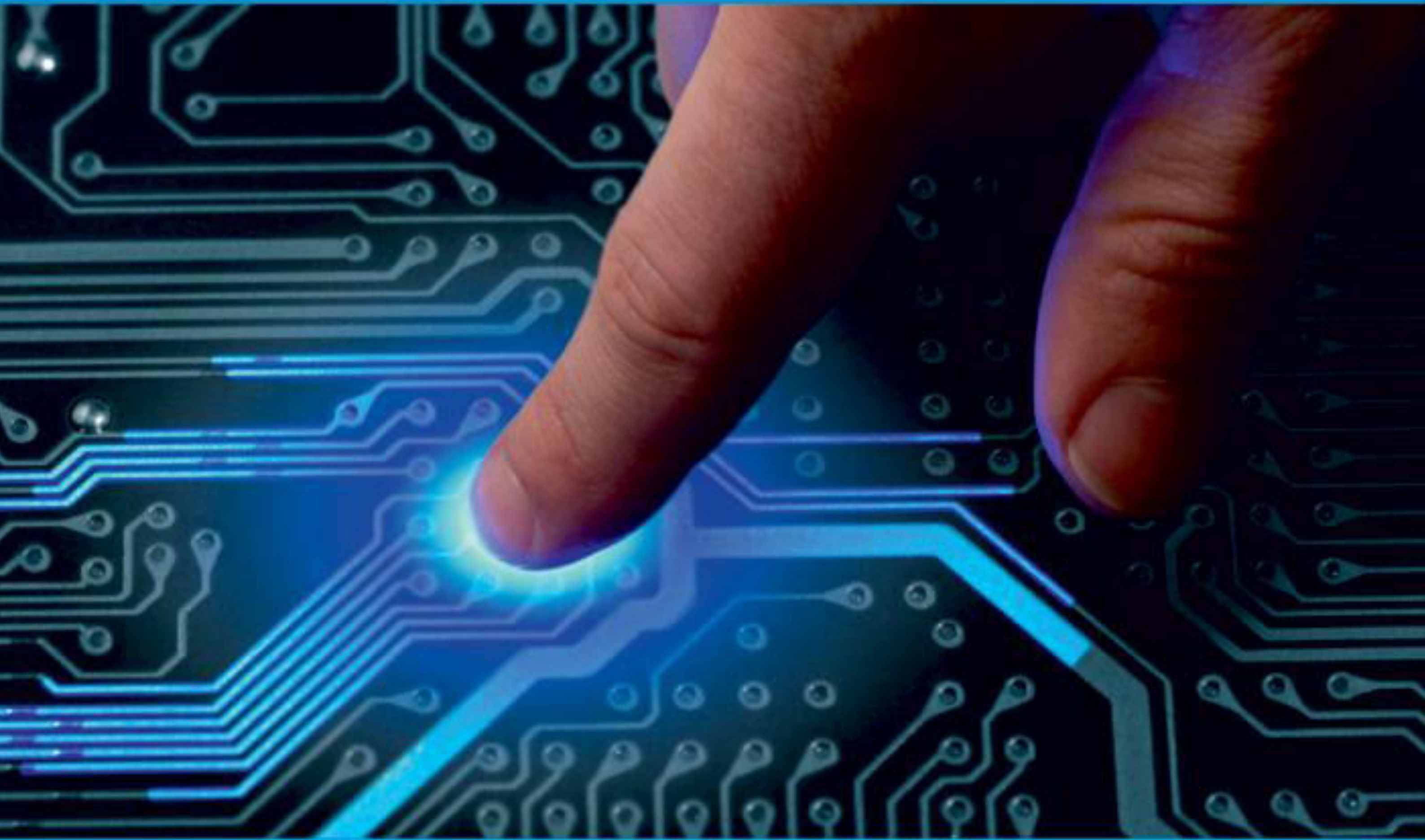
e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 6, June 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

Image Rectification Using Recurrent Convolutional Neural Networks

Anand Ravishankra, Rajan. A, Arushi Makraria, Akshat Agarwal

UG Students, Dept. of ECE, BMS institute of Technology and Management, Karnataka, India

ABSTRACT: Super-resolution encompasses a set of algorithms used to enhance and up-sample the resolution of images in order to increase the information density and the resultant sharpening of the output image. Conventional image rectification algorithms like nearest neighbor interpolation, linear interpolation, bicubic interpolation have been used to enhance images but are susceptible to the drawback of increased pixelation when the spatial dimensions are increased. In this paper we propose a novel algorithm that uses Recurrent Convolutional Neural Networks (AIRCNN) to achieve super resolution with improved PSNR and SSIM levels. Our results show an 11% improvement in both these values, when compared to traditional methods.

KEYWORDS: Super Resolution, Convolutional Neural Networks, Recurrent Networks, Deep Learning

I. INTRODUCTION

Convolutional neural networks are a class of neural networks that are used to detect and classify objects in an image by performing the convolution operation across the image pixels to obtain increasing complex patterns. These patterns can be utilized to classify the images and map them to some ground truth. One of the main advantages of these networks is the use of local spatial coherence that provides the same weights to some of the edges which minimize the global error.
The architecture of convolutional neural networks is the connectivity pattern present in the neurons of a brain where the network design between nodes looks like the association of the creature's visual cortex. Individual cortical neurons react to improvements just in a limited locale of the visual field known as the responsive field. The open fields of various neurons incompletely cover with the end goal that they cover the whole visual field. Similar to a human brain, the neurons are connected in particular configurations having shared weights which influence the activation of neighboring neurons.
The preprocessing required in Convolutional Neural Networks (CNN) is much lower in comparison to conventional classification algorithms. While in traditional methods like bicubic interpolation the filters are fixed, convolutional layers map input images to output images through a varying set of filters. In this paper, we attempt to achieve superresolution/image enhancement through the use of a recurrent convolutional neural network, which consists of skip connections and convolutional layers.

II. RELATED WORK

In [1] the authors generate realistic textures during single image super-resolution. To enhance the visual quality of the test images, they focused on network architecture, adversarial loss, and perceptual loss and improved on each one of them to derive an enhanced Generative Adversarial Nets (GAN). Their proposed network provides stronger supervision for brightness consistency and texture recovery. This paper provides a classic example of using generative networks instead of adaptive networks to get the required results. However, the adversarial nature of GANs makes them computationally expensive.
In [2] the authors exploit iterative up and down-sampling layers, providing an error feedback mechanism for projection errors at each stage. They constructed mutually connected up and down-sampling stages each of which represents different types of image degradation and high-resolution components.
This resulted in improved super-resolution, yielding superior results and in particular establishing a new state of the art results for large scaling factors. A feed-forward network is used to accumulate the auto-correction features for each of the upsampling stages to create a super resolution image.

The most common techniques for obtaining higher resolutions involve using different Deep Learning architectures, ranging from the early CNN-based method, to recent promising SR approaches using GANs. Generalizing it, we can

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906110 |

say the family of SR algorithms using deep learning algorithms has different features from each other in the mentioned major aspects:

- Types of network architectures as shown by [5], [6] and [7].
- Types of loss functions [8], [9], [10].
- Types of learning principles, and strategies [8], [11], [12].

In this survey, we have given a brief overview of recent advances in image super-resolution with deep learning. Although there are some existing SR surveys in literature, our work differs in that we are focused on combining Convolutional layers along with skip connections to achieve SR, while most of the earlier works aim at surveying traditional SR algorithms. On the other hand, some studies mainly concentrate on providing quantitative evaluations based on full-reference metrics or human visual perception.

III. METHODOLOGY

We began building our model from scratch. We began from just 1 CNN layer with limited dataset and afterwards changed the number of layers, filters and other parameters. At the point when it stopped increasing performance significantly, we attempted to change the model construction and attempted bunches of profound learning methods like mini batch, dropout, standardization, regularizations, analyzers and activators to become familiar with the implications of utilizing each design and methods. At last, we properly picked architecture and hyper boundaries which will best suit our SISR undertaking and assemble our last model.

A. Model Overview

Our model (AIRCNN) is a convolutional neural network combined with skip connections. As displayed in Figure 1, AIRCNN comprises a component extraction network and a recreation network. We combine groups of CNN weights, biases and non-linear layers to the input image. Then, at that point, to extricate both the local and the global features, all outputs of the hidden layers are associated with the reconstruction network as skip connections. At the other end of the skip connections, parallelized CNNs (Network in Network [2]) are utilized to reproduce the picture subtleties. The last CNN layer yields the output (or the channels of the square of scale factor) picture. Finally, the super resolute unique picture is assessed by adding these outputs to the up-sampled picture built by bicubic interpolation. Along these lines, the proposed CNN model focuses on learning the residuals between the bicubic interpolation of the low resolution (LR) picture and the High Resolution (HR) original picture.

In traditional SR techniques, up-sampled pictures were frequently utilized as the input for the DL-based models. In these models, the SISR organizations will be pixel-wise, leading to an increase in computational power. Moreover, only 20-30 CNN layers are essential for each up-sampled pixel and substantial calculation (up to 4x, 9x and 16x) is needed, as displayed in Figure 1. Also, we can easily understand that extracting features from original low-resolution images will be more helpful than artificially upsampled images.

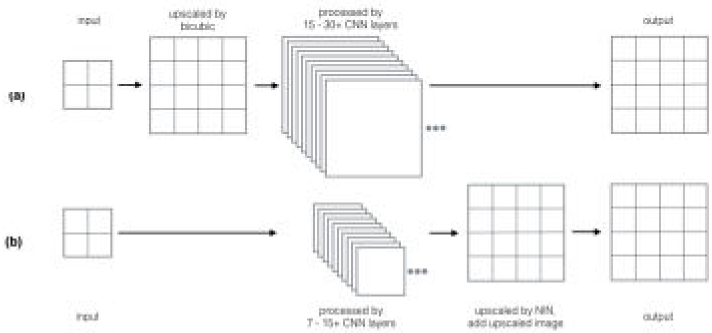


Fig 1. Simplified process structures of (a) other models and (b) our model (AIRCNN)

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906110 |

B. Feature Extraction Network

In the primary feature extraction network, we cascade 7 arrangements of 3x3 CNN, bias and Parametric ReLU units. Each output of the units is passed to the following unit and all the while jumped to the reconstruction network. As opposed to other significant deep learning-based super-resolution models, the quantity of units of CNN layers is reduced from 96 to 32, as displayed in Table 1. As talked about in Yang et al. [10], for model pruning, utilize a proper number of preparing boundaries to upgrade the network. Since the local features have a higher priority than the global component in SISR issues, we decrease the highlights by the accompanying layer and it results in better execution with quicker calculation. We additionally utilize the Parametric ReLU units as initiation units to deal with the "Dying ReLU" issue [11]. This keeps loads from learning a huge negative bias term and can prompt a marginally better execution.

C. Image Reconstruction Network

As expressed in the Model Overview, AIRCNN straightforwardly measures original pictures with the goal that it can extricate features proficiently. The last HR picture is remade in the last part of the model and the organization structure resembles the Network in Network [2]. Since we are concatenating all the input layer features to the reconstruction network, the dimensions of data is somewhat enormous. So we utilize 1x1 CNNs to decrease the input dimensions prior to producing the HR pixels.

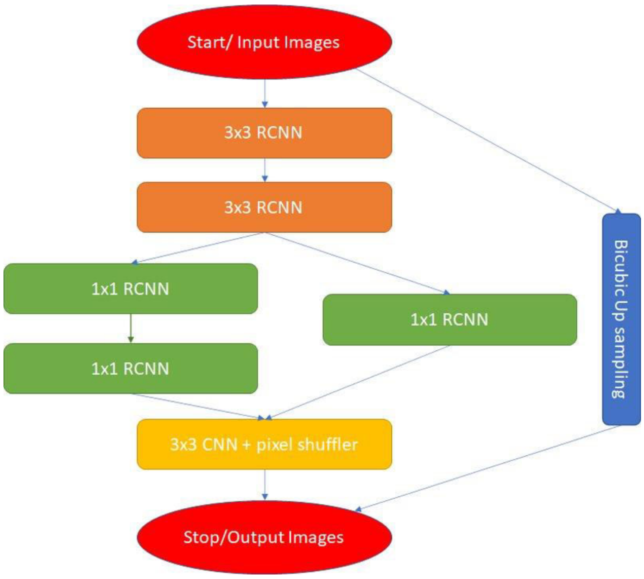


Fig 2.Architecture of AIRCNN

The last CNN, addressed by the yellow tone in Figure 2, yields 4 channels (when the scale factor is set to 2) and each channel addresses each corner-pixel of the up-tested pixel. AIRCNN reshapes the LR picture to a HR(4x) picture and afterward at last it is added to the bi-cubic upsampled original given picture. Typically with residual learning networks, the model is made to learn on residual output and this enormously helps learning rates, even in instances of shallow (under 7 layers) models.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906110 |

	Feature extraction network							Reconstruction network			
	1	2	3	4	5	6	7	A1	B1	B2	L
AIRCNN	96	76	65	55	47	39	32	65	32	32	4
c-AIRCNN	32	26	22	18	14	11	8	24	8	8	4

Table 1. The numbers of filters of each CNN layer of our proposed model

IV. CODE SNIPPET

The code snippet provided in this section demonstrates the core functioning of our Algorithm. It takes in an input image and performs the necessary transformations resulting in a higher resolution image.

```
DEF DO(SELF, IP_IM, BICUBIC_IP_IM=NONE):
        SSVAL=SELF.SCALE
        H, W = IP_IM.SHAPE[:2]
        CH = IP_IM.SHAPE[2] IF LEN(IP_IM.SHAPE) > 2 ELSE 1

        IF BICUBIC_IP_IM IS NONE:
        BICUBIC_IP_IM = UTIL.RESIZE_IMAGE_BY_PIL(IP_IM, SSVAL,RESAMPLING_METHOD=SELF.RESAMPLING_METHOD)
        IF SELF.MAX_VALUE != 255.0:
        IP_IM = NP.MULTIPLY(IP_IM, SELF.MAX_VALUE / 255.0)  # TYPE: NP.NDARRAY
        BICUBIC_IP_IM = NP.MULTIPLY(BICUBIC_IP_IM, SELF.MAX_VALUE / 255.0)  # TYPE: NP.NDARRAY

        IF SELF.SELF_ENSEMBLE > 1:
        OUTPUT = NP.ZEROS([SSVAL * H, SSVAL * W, 1])

        FOR I IN RANGE(SELF.SELF_ENSEMBLE):
                IMAGE = UTIL.FLIP(IP_IM, I)
                BICUBIC_IMAGE = UTIL.FLIP(BICUBIC_IP_IM, I)
                Y = SELF.SESS.RUN(SELF.Y_, FEED_DICT={SELF.X: IMAGE.RESHAPE(1, IMAGE.SHAPE[0], IMAGE.SHAPE[1],
CH),SELF.X2: BICUBIC_IMAGE.RESHAPE(1, SSVAL * IMAGE.SHAPE[0],SSVAL * IMAGE.SHAPE[1], CH), SELF.DROPOUT: 1.0,
SELF.IS_TRAINING: 0})
                RESTORED = UTIL.FLIP(Y[0], I, INVERT=TRUE)
                OUTPUT += RESTORED

        OUTPUT /= SELF.SELF_ENSEMBLE
        ELSE:
        Y = SELF.SESS.RUN(SELF.Y_, FEED_DICT={SELF.X: IP_IM.RESHAPE(1, H, W, CH),SELF.X2: BICUBIC_IP_IM.RESHAPE(1, SSVAL *
H,
       SSVAL * W, CH),SELF.DROPOUT: 1.0, SELF.IS_TRAINING: 0})
        OUTPUT = Y[0]

        IF SELF.MAX_VALUE != 255.0:
        HR_IMAGE = NP.MULTIPLY(OUTPUT, 255.0 / SELF.MAX_VALUE)
        ELSE:
        HR_IMAGE = OUTPUT

        RETURN HR_IMAGE
```

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906110 |

V. SIMULATION RESULTS

Since every execution happens under distinctive stages and libraries, it's unfair to look at these strategies by test execution time. Here we figure the calculation intricacy of every technique with all things considered. Since deep learning calculation is typically hard to parallelize, a calculation complexity of 1 pixel is utilized as a decent pointer of calculation speed. CNN layers are determined as size square times input channels times yield channels. Bias, ReLU, adding or duplicating layers are determined as the number of filters. When bicubic up-sampling is required, we compute it as 16 duplications and increments. In this manner, the surmised calculation intricacy for every technique is displayed in Table 3. The complexity determined may somewhat vary from genuine complexity. For instance, FSRCNN and RED contain translated CNN furthermore, it needs to be padded with 0 before handling. In any case, those distinctions are a lot more modest than CNN computations and hence can be approximated. We can see our AIRCNN has a cutting edge recreation execution.
As per the problem statement of our project we have successfully demonstrated the rectification of low quality images using a recurrent convolutional neural network. Through this model we have achieved an 11% increase in image quality and PSNR values as shown in Table 2

		PSNR		SSIM	
	Scale	Bicubic	AIRCNN	Bicubic	AIRCNN
set14	x2	30.228917	33.28912	0.86822	0.914005
	x3	28.325812	30.269423	0.853512	0.908652
	x4	25.623412	27.253961	0.823197	0.883519
set5	x2	33.654669	37.832237	0.92944	0.959015
	x3	30.825124	35.973251	0.91642	0.951189
	x4	29.864912	32.436129	0.90842	0.929845
bsd	x2	29.555535	32.036919	0.842629	0.897012
	x3	29.023412	30.32576	0.826849	0.873151
	x4	28.132456	29.963581	0.816812	0.865312

Table 1. PSNR and SSIM values

VI. CONCLUSION AND FUTURE WORK

In this paper, we have presented a novel deep learning approach for single image super-resolution (SR). We utilize a combination of Convolutional layers along with Skip connections and Residual networks. In the feature extraction network of our method, the structure is optimized and both local and global features are sent to the reconstruction network by skip connection. In the reconstruction network, residual networks are used to obtain a better reconstruction performance with less computation. In addition, the model is designed to be capable of processing original size images. The proposed approach, AIRCNN, maps low and high resolution images, with minimalistic processing apart from the network architecture. With a lightweight structure, AIRCNN has performed equally if not better than state-of-the-art methods in terms of PSNR and SSIM values. Additional performance can be further gained by exploring diverse filters and applying multiple training strategies. Besides, the proposed structure, with its advantages of simplicity and robustness, could be applied to other low-level vision problems, such as image de-blurring or simultaneous SR+denoising. One could also investigate a network to cope with different upscaling factors.

REFERENCES

1. Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, Xiaoou Tang, ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks, ECCV 2018 workshop, arXiv:1809.00219
2. Muhammad Haris, Greg Shakhnarovich, Norimichi Ukita, Deep Back-Projection Networks For Super-Resolution, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 1664-1673
3. Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, Kyoung Mu Lee, Enhanced Deep Residual Networks for Single Image Super-Resolution, Proceedings of the IEEE Conference on Computer Vision and Pattern

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906110 |

Recognition (CVPR) Workshops, 2017, pp. 136-144

4. Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, Yun Fu, Residual Dense Network for Image Super-Resolution, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 2472-2481
5. Ying Tai, Jian Yang, Xiaoming Liu, Image Super-Resolution via Deep Recursive Residual Network, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3147-3155
6. Tong Tong, Gen Li, Xiejie Liu, Qinquan Gao, Image Super-Resolution Using Dense Skip Connections, Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 4799-4807
7. Kai Zhang, Wangmeng Zuo, Lei Zhang, Learning a Single Convolutional Super-Resolution Network for Multiple Degradations, Proceedings of IEEE Conference on Computer Visions and Pattern Recognition (CVPR), 2018, pp. 3262-3271
8. M. S. Sajjadi, B. Sch ̈olkopf, and M. Hirsch, "Enhancenet: Singleimage super-resolution through automated texture synthesis," inICCV, 2017.
9. J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," inECCV, 2016.
10. A. Bulat and G. Tzimiropoulos, "Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans," in CVPR, 2018.
11. B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in CVPRW,2017.
12. Y. Wang, F. Perazzi, B. McWilliams, A. Sorkine-Hornung,O. Sorkine-Hornung, and C. Schroers, "A fully progressive ap-proach to single-image super-resolution," in CVPRW, 2018.