Enhanced Residual Connections Method for Low Resolution Images in Rice Plant Disease Classification

K. Sathya, M. Rajalakshmi

Enhanced Residual Connections Method for Low Resolution Images in Rice Plant Disease Classification

Sathya K[1][0000-0002-6537-1772] and Rajalakshmi M[2][0000-0002-9229-5971]

[1, 2] Coimbatore Institute of Technology, Coimbatore, 641014, India
sathya.k@cit.edu.in

Abstract. Recent advancements in both raw computing powers as well as the capabilities of cameras in recent times, it has now become possible to capture images of very high quality. This improvement however does come at the cost of the overall space required to store such high-quality images. One possible solution to this problem would be the storage of the images in low resolution and then upsampling the images to obtain the original resolutions. Despite advances in computer vision, deep learning models for super resolution have been introduced to tackle this issue and thus provide promising improved performance results. This paper will explore a novel self-supervised deep learning architecture entitled Enhanced Residual Connections for Image super resolution (ERCSR) that is capable of upsampling extremely low-quality images to their higher quality. Experiments on the different data sets such as DIV2k and rice plant images are made to evaluate this model and experimental results shows that our method outperforms image enhancement. Furthermore, rice plant images are subsequently passed through disease classification layers and achieves desired accuracy for super resolution images.

Keywords: Super Resolution, Residual Learning, Deep Learning, Interpolation, Rice Disease Classification.

1 Introduction

The term super resolution is a general indicative of the processes of converting a low-resolution image (I_{LR}) to a high-resolution (I_{HR}) or super resolution image (I_{SR}) image. This type of upsampling a I_{LR} image has a few use cases such as the need for conserving storage, surveillance where the images captured are usually very low in quality or even in the medical field where capturing high quality scans may be impossible. The term super resolution is also an umbrella term that encompasses both single image upscaling and video feed upscaling. While the fundamental ideas between the two kinds of upscaling remains the same, the main difference between them is the fact that to upscale a video feed at a reasonable frame rate, the overall architecture must perform much faster than what is needed from a single image.

Image super resolution of single images can be processed quite easily using very basic techniques such as the use of bicubic or bilinear interpolation [1], [2]**.** These interpolation techniques however, guess the correct pixel value in the I_{SR} image by means of relatively simple arithmetic on the local pixel information in the I_{LR} image. While this approach is somewhat feasible when considering a small upscaling factor such as x1.5 or x2, interpolation techniques quickly become unusable when considering upscaling factors of x3 or above. This is due to the fact that the local pixel information in the I_{LR} image will be directly responsible for several pixels in the SR image causing extreme blurring and loss of details.

In more recent times, most of the attempts at image super resolution involve some form of a complex deep learning model that tend to do a better job than simple interpolation for finding out I_{SR} pixel values [3], [4]. The deep learning models can be trained in a supervised manner if one has both the I_{LR} and I_{HR} image dataset. This however is quite difficult as finding I_{LR} images that have consistency in the amount downscaling throughout the entire dataset is rare.

The caveat of having large amounts of data for training a deep learning model is easily solved by using a standard and known downgrade functions on I_{HR} images like the bicubic interpolation. With this method, it is easily possible to obtain large amounts of data that are downgraded in the same method from I_{HR} images which can be found in abundance now. This fashion of pseudo self-supervised training has become the standard across the field with many competitions providing high quality freely to promote research in real world problems [5]. This paper uses Rice Plant Disease Image Dataset (RPDID) to validate the performance of the proposed method and achieved desired accuracy for super resolution images compared to low resolution images. By this exploration, the following key contributions of this paper can be summarized;

1. Remodeling the existing residual block defined in the architecture to make it more suitable for the specific case of super resolution.
2. Implementation of residual scaling to reduce the adverse effects of the residual networks to be able to maximize the architectures size.
3. Implementation of an architecture that is capable of producing super resolution images at various scales.
4. Comparison of the proposed architecture against various different models on various different datasets.

2 Related Works

This section will go over some of the basic terminology used in the formulation of the image super sampling problem as well as go over the existing state of the art architectures in supervised image super sampling.

The image super sampling problem can be approached in several different ways and in the state-of-the-art models there are several that are vastly different but achieve the same end goal. These different architectures are usually built on top of existing core architecture or training methodologies such as the CNN architecture or the GAN

architecture. In this subsection we shall go over some of the more commonly used architecture types and recent examples in each category, looking at how each of the fundamental architecture are being used to achieve the overall goal.

2.1 SRCNN

SRCNN proposed by Dong, Loy, He, et al. [6] is a fairly straightforward architecture made up of CNN layers. SRCNN is divided into three separate parts: one layer for patch extraction, one for non-linear mapping and one for reconstruction. The SRCNN was made to be an end-to-end system that can easily upscaling the I_{LR} images into their I_{SR} images. And since the most difficult part of super resolution, the upscaling, is already done, the SRCNN architecture is used only to ensure that the I_{LR} upscaled image can be mapped to the I_{SR} image pixels properly. Mean Square Error (MSE) loss function is used for training the SRCNN architecture and Peak Signal-to-Noise Ratio (PSNR) is used as the evaluation metric. The FSRCNN architecture created by Dong, Loy, and Tang [7] works in very similar ways to the SRCNN but there are a few key differences. The main difference between the two architectures is that the FSRCNN architecture follows a post sampling architecture making the architecture to have a learnable deconvolutional upsampling function rather than the bicubic interpolation used in SRCNN. This change allows the FSRCNN to have no pre-processing stage to upsample the LR image before it can be passed through the model.

2.2 VDSR

Kim, Lee, and Lee [8] proposed an upgraded version of the SRCNN, VDSR which had the same overall architecture of SRCNN but was improved in certain areas. The VDSR's main area of improvement of SRCNN was the fact that it was a much deeper network utilizing 3x3 convolutional filters compared to the 1x1 in SRCNN. This methodology of deeper network with larger filters is similar to the VGG architecture [9] and allowed the VDSR to learn the residual between the I_{LR} and I_{SR} images. Gradient clipping is used to train the VDSR model to ensure that learning rates can be higher than if MSE is used.

2.3 Residual Networks

Residual blocks as used in the ResNet architecture, He, Zhang, Ren, et al. [10] have showcased that the overall performance of the model improves when compared to linear convolution networks such as the VGG networks. The same principle can be applied in the case of the super resolution problem, with the skip connections being able to pass on feature information from the I_{LR} image much smoother compared to linear models. The residual learning method was more efficient and further incorporated in several other deep learning based super resolution models [11], [12], [13].

2.4 CARN

The CARN architecture proposed by Ahn, Kang, and Sohn [14] develops on top of the traditional skip connections used in ResNet to incorporate a much smoother means for information travel. The novel" cascading" methodology, allows information gained from LR features on both a local and global level to be fed into multiple layers. The cascading nature of CARN cascading blocks allows each block to receive inputs from all the previous blocks which results in a more effective means of transfer compared to the original ResNet architecture. Furthermore, CARN uses a group-wise convolutions per block which allow the overall size of the model to decrease giving up 14x decrease in the number of computations required. The shared residual blocks in CARN also help in reducing the overall number of parameters used in the architecture.

2.5 Generative Model

Unlike the architecture, have seen thus far, Generative models built on top of the Generative-Adversarial-Network architectures [4] can also be used for LR to SR conversions. The main difference between these approaches is the fact that generative models do not try to minimize the loss between the LR and SR pixels, rather the try to optimize in a way that make the result more realistic.

SRGAN

The SRGAN architecture proposed by Ledig, Theis, Huszar, et al. [15] is a modification of the GAN architecture, that is designed to produce human eye pleasing 8 images from LR samples. The SRGAN architecture uses a novel ResNet based architecture - SRResNet architecture as sort of a large pretrained model to fine tune upon and uses a multi task loss as the loss function. This loss is made up of three components, the adversarial loss that is passed on from the discriminator network, a MSE loss that works similar to non-generative models for capturing pixel similarity and finally a novel perceptual similarity loss that captures high level information. An interesting point about the SRGAN architecture is the fact that, while the model performs worse than other architecture when considering PSNR values, it still outputs a better-quality image to the naked eye (i.e., Mean Opinion Score was higher).

3 Problem Definition

The problem definition of image super resolution built around the ideology of recovering the I_{SR} image from the I_{LR} image. This definition remains true regardless of whether or not the I_{LR} images have been obtained through the means of a known downgrading function or not. Thus, the general modelling of the overall I_{LR} to I_{SR} architecture follows the trend shown in equations below.

where D is the degradation mapping function that maps the low-resolution image, I_{LR}, to the super resolution image, I_{SR}. δ here is the parameters of the degradation function such as the interpolation methods, scaling factors or noise function used.

$$I_{LR=}\mathcal{D}(I_{SR};\delta) \tag{1}$$

$$\mathcal{D}(I_{SR};\delta) = (I_{SR}) \downarrow s \subset \delta \tag{2}$$

where ↓s is the degrading operation. Most of the datasets being used in research as of now follow this form of degradation. The most common form of degradation used is the bicubic interpolation method with anti-aliasing [16] but other forms of degradation are also used [17].

4 Proposed Architecture

In this section, the proposed architecture is discussed in terms of design and implementation. From the previous section, several different methods can be employed to achieve the final result of super resolution. However, while several methods exist, most of the methods focus on the super resolution problem statement in a rigid manner where super resolution to different scales or from differing I_{LR} images will throw off the overall pipeline. These solutions treat the SR problem in a methodology that does not consider using the mutual relationship that exists between different scales of I_{LR} images being converted to different scales of the I_{SR} image. As such, the only possibility for these architectures to work with different scales is to be trained as such on, producing several scale-specific models.

The proposed model will build up on two different models that covered in the literature survey, namely the VDSR and the SRResNet. The VDSR model has been proven that it is capable of handling various scales for super resolution while utilizing a single network. VDSR has also shown that scale-specific networks underperform indicating that while the scales might differ, there exist a lot of information that is common between the different scales. The other network, SRResNet, has several useful architectural designs that can be used as well. Unlike VDSR, SRResNet does not require the input pipeline to begin with a computation and time consuming bicubic interpolated images.

However, the SRResNet presents a different issue as well; while the architecture is good at optimizing both time and computation needed in the preprocessing stage, the SRResNet architecture itself is exactly the same as the original ResNet architecture. The proposed architecture focuses on keeping the scale agnostic methodology of the VDSR while trimming down on the ResNet architecture so it is more suitable for the super resolution task as compared to a more complicated vision problem such as object detection.

4.1 Proposed Residual Block

The use of residual networks in the field of computer vision has drastically gone up in recent years and there are several works that showcase the power of residual connections in for the specific use case of super resolution [7], [8], [13], [19], [20]. The SRResNet architecture by Ledig, Theis, Huszar, et al. [15] implemented a residual block very much similar to the original ResNet architecture, it is possible to further fine-tune the block to improve on the time and computational complexity as shown in Fig. 1.

The standout difference between the three different types of residual block is the removal of the batch normalization layers that are present in the side branch. While the batch normalization layers help immensely when used in a more high-level computer vision task such as object detection, when using it in a task that requires a more fine-grained learning procedure it fails as shown by the work done by Nah, Hyun Kim, and Mu Lee [20] in their image deblurring task. Since the role of the batch normalization layer is to aggregate the information over a set of images, when using it for a task such as image deblurring or super resolution, the aggregation tends to result in over-normalize the features, thereby getting rid of the flexibility of the model.

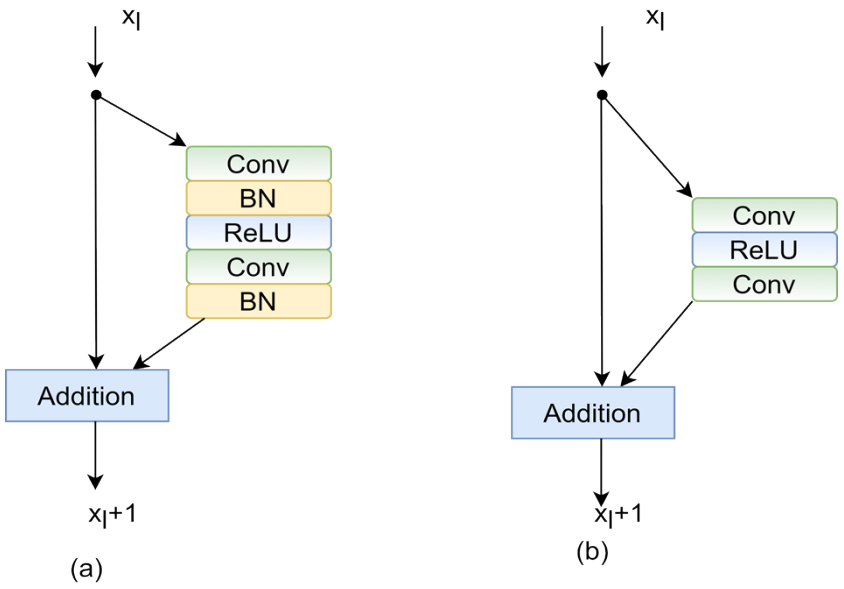

Fig. 1. Comparison between (a) SRResNet (b) Proposed Block

An added advantage to dropping the batch normalization layers is also the fact that these layers are very heavy in terms of the computational requirements as they use the same number of computational resources as the CNN layers. Removal of the batch normalization layers give us a model which when having similar size and depth as the SRResNet architecture requires approximately 35 to 45 percent lesser memory during

the training stage. Due to the reduced compute requirements when using the proposed residual block, it is possible to build up a larger model with more layers which results in better performance compare to SRResNet when trained on the same hardware.

4.2 Loss Function

Since the proposed model is primarily a very simple CNN only network, the loss function chosen for this implementation is the pixel loss methodology. The pixel loss is the simplest loss methodology when compared to loss functions used in other architectures such as the perceptual loss and the adversarial loss which are used in SRGAN. The pixel loss essentially is concerned with minimizing the difference between the actual SR image and the generated SR image. The difference itself can be calculated using any distance metric and minimizing pixel loss generally leads to models which are optimized for generating images with high PSNR values.

The model utilizes L1 loss, Equation 3, over the more common L2 loss, as this use case L2 loss was significantly better at convergence when compared to L1 loss. This choice in L1 loss comes more from a empirical standpoint after several experiments rather than a theoretical one, as generally minimizing L2 loss tends to provide higher PSNR.

$$\mathcal{L}(y - \hat{y}) = \sum_{i=1}^{N} |y - \hat{y}_i| \tag{3}$$

4.3 Proposed model

Just like most computer vision task, the super resolution problem can also take advantage of the fact that increase in the number of parameters in the ERCSR model. There are two primary ways in which a model consisting of CNN layers can increase the number of parameters - increasing the depth of the model (i.e., stacking CNN layers on top of each other) or by increasing the number of filters per CNN layer. In the former method, the relationship between the number of layers and the memory is linear. On the other hand, in the latter method, memory scales in a quadratic from with the increase in number of filters and so the overall model formation considers a balance between the two to achieve the maximum possible parameters given the limited hardware resources.

Fig.2 shows the high-level overview of the proposed model ERCSR with both the residual block structure and the upsampling block. While theoretically, increasing the number of filters used per CNN layer should accordingly scale the performance of the model, the work done by Szegedy, Ioffe, Vanhoucke, et al. [21] showed that there is actually an upper limit. Beyond this limit, any further increase in the number of features makes the training stage highly unstable and the performance of the model quickly decreases.

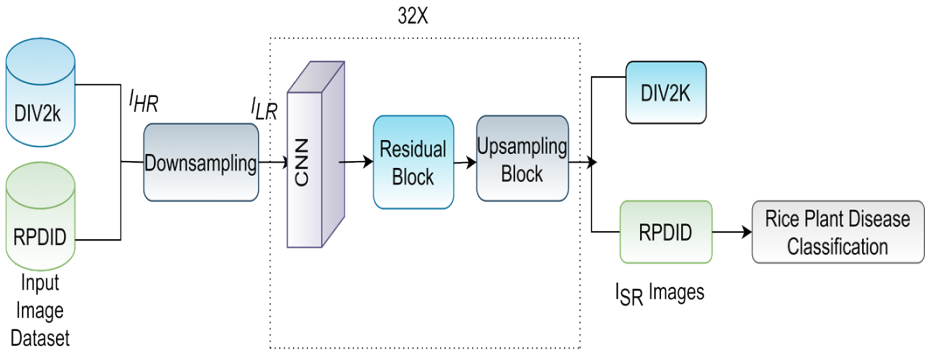

Fig. 2. Architecture of Enhanced Residual Connections for Image super resolution (ERCSR)

This issue occurs mainly due to the fact that once this threshold is crossed, by the time information reaches the average pooling the model primarily produces only zeros after training for a few thousand iterations. This phenomenon is unavoidable even when changing other hyperparameters such as the learning rate. In the proposed model," residual scaling" [21] is implemented in each residual block after the last CNN layer to stabilize the model. Residual Scaling involves scaling down the residuals produced before passing them on to the average pooling layer. Fig. 3 shows the residual block structure broken down into its individual components.

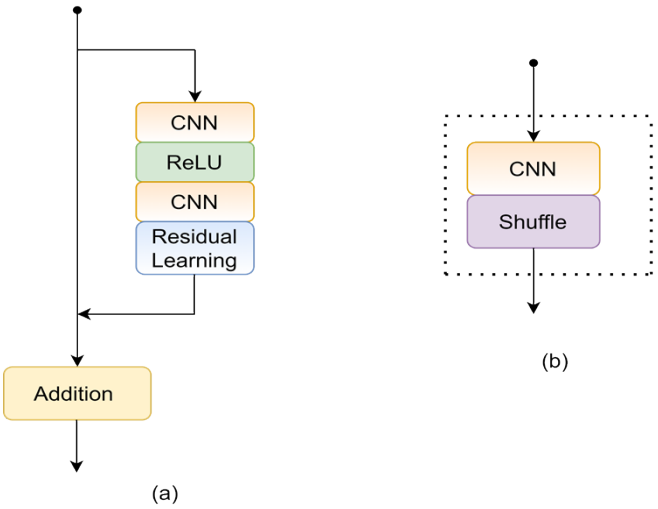

Fig. 3. (a) Scaled Residual Block (b) Upsampling Block

The model itself consists of 32 CNN layers each with a filter size of 256 with a residual scaling factor equal to 0.1. The residual block itself is very similar to the one used in SRResNet but along with the removal of the batch normalization layers, the

final ReLU activation is also removed. The presence of the ReLU activation interacted negatively once the scaling factor was introduced causing instability in the model and the removal of it did not affect the overall performance of the model.

5 Experimental Setup

5.1 Dataset

The DIV2k dataset [22] was used for the training and validation of this model. The dataset consists of 2k resolution images split into 800 training images, 100 validation images and 100 test images. The test images are not released as they are part of the challenge but the test image have a wide range of downgrades applied to them such as extreme blurring apart from the standard bicubic interpolation downgrades. The model is also tested on the Urban100 dataset [23], the B100 dataset [24] and the Set14 dataset [25].

5.2 Training

For training, the model takes in 48x48 RGB patches from the LR image and the corresponding HR image. Since there are only 800 images, a pre-processing stage of image augmentation is also done. In this augmentation stage, the trained data is arbitrarily flipped in both horizontal and vertical directions and this these flipped images have the mean RGB value of the DIV2k dataset subtracted from them. The model is trained with the ADAM optimizer with $\beta 1 = 0.9$, $\beta 2 = 0.998$ and $\epsilon = 10^{-9}$. The x2 scale model is trained first from scratch with the parameters and pre-processing techniques mentioned above. Once the x2 scale model converges, this model is then used as a pretrained model whilst training the x3 and x4 scale models.

5.3 Evaluation Metrics

The term PSNR is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation. Because many signals have a very wide dynamic range, (ratio between the largest and smallest possible values of a changeable quantity) the PSNR is usually expressed in terms of the logarithmic decibel scale. Using the same set of tests images, different image enhancement algorithms can be compared systematically to identify whether a particular algorithm produces better results. The metric under investigation is the peak-signal-to-noise ratio, show that an algorithm or set of algorithms can enhance a degraded known image to more closely resemble the original, then more accurately conclude that it is a better algorithm.

$$MSE = \frac{1}{N^2}\sum_{w=1}^{N}\sum_{h=1}^{N}(I_{SR}(w,h) - I_{\mathrm{HR}}(w,h))^2 \tag{4}$$

$$PSNR = 10 \log_{10} \frac{MAX^2}{MSE} \tag{5}$$

Where I_{SR}- super resolution image, I_{HR}- high resolution image and N image size respectively. Here the PSNR is measured in terms of Decibel (dB).

6 Results and Discussion

The proposed model is tested against SRResNet which has been modified to showcase various different case scenarios. The SRResNet scores that are given in this section are from a individually retrained model that follows the exact architecture made by Ledig, Theis, Huszar, et al. [15]. One of the main reasons why SRResNet was retrained was the fact that that original model was only trained using the L2 loss, and to establish a fair comparison with the proposed model SRResNet was retrained using L1 loss as well.

All the models (i.e., both variations of SRResNet and the proposed model) were trained solely on the DIV2k dataset and 300,000 steps in the train loop. Evaluation of each model is done at each step by using 10 randomly selected images from the evaluation dataset, the evaluation metrics was mainly the PSNR value. The data passed to all the models followed the same pre-processing technique. When training the proposed model for upsampling factor ×3 and ×4, initialize the model parameters with pre-trained ×2 network. This pre-training strategy accelerates the training and improves the final performance as clearly demonstrated in Fig. 4

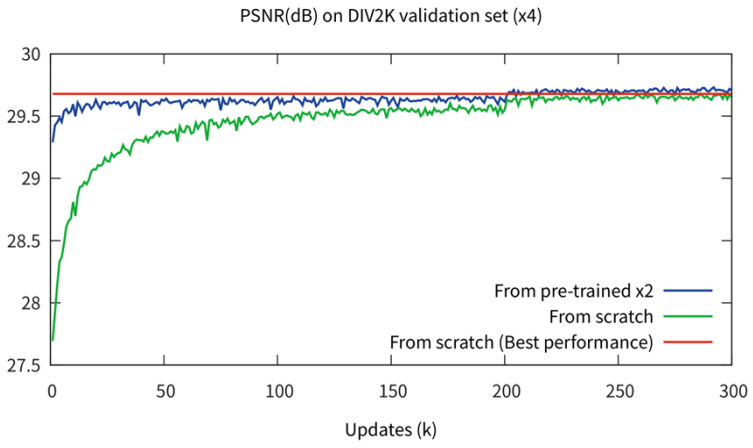

Fig. 4. Sample PSNR on DIV2K

For upscaling ×4, use a pre-trained scale ×2 model (blue line), the training converges much faster than the one started from random initialization (green line).

Fig. 5. Sample DIV2k dataset Qualitative result

Table 1 presents the quantitative results obtained from the DIV2k dataset and Fig. 5 showcases the qualitative result. SRResNet trained with L1 gives slightly better results than the original one trained with L2 for all scale factors. The image augmentation methodology is in particular much more applicable for usage in the proposed model compared to SRResNet since the architecture is much lighter. The lack of batch normalization layers allows the proposed architecture to be further extended to produce better results than shown in Table 1 but the trade off in overall training time was not justifiable.

Table 1. Quantitative results obtained from the DIV2k dataset

Scale	SRResNet (L1 Loss)	SRResNet (L2 Loss)	Proposed Model
x2	34.30/0.9662	34.44/0.9665	35.12/0.9686
x3	30.82/0.9288	30.85/0.9292	31.05/0.9349
x4	28.92/0.8960	28.92/0.8961	29.17/0.9024

Beyond the DIV2k dataset, Table 2 shows the comparison of the proposed model to SRResNet on various another dataset.

Table 2. Comparison of quantitative results (PSNR and SSIM) for DIV2k dataset

Dataset	Scale	SRCNN	VDSR	SRResNet (L2 Loss)	Proposed Model
Set14	x2	32.42/0.9063	33.03 / 0.9124	-	34.02/ 0.9204
	x3	29.28/0.8209	29.77 / 0.8314	-	30.66/ 0.8481
	x4	27.49/0.7503	28.01 / 0.7674	28.53/0.7804	28.94/ 0.7901
Urban100	x2	29.50/0.8946	30.76 / 0.9140	-	33.10/ 0.9363
	x3	26.24/0.7989	27.14 / 0.8279	-	29.02/ 0.8685
	x4	24.52/0.7221	25.18 / 0.7524	27.57/0.7354	27.79/ 0.7437
B100	x2	31.36/0.8879	31.90 / 0.8960	-	32.37/ 0.9018
	x3	28.41/0.7863	28.82 / 0.7976	-	29.32/ 0.8104
	x4	26.90/0.7101	27.29 / 0.7251	26.07/0.7839	27.79/ 0.7437

6.1 Real World Performance

While the main goal of this work is simply to create an architecture that is capable of creating a SR image from LR images, it is important to test the generated images in a real-world setting. To this end we use the RPDID dataset to evaluate the result of upsampling in terms of the classification accuracy. The RPDID dataset is a quite varied dataset, and to accurately test the model, the downsampling methodology is the same as the one use before in this work. These labelled images were then used as the testing data on various standard large pretrained image models.

Table 3. Rice Plant Disease Image Dataset

Rice diseases	**Total no of image samples**
Leaf Blast	2219
Brown Spot	2163
Tungro	1308
Bacterial leaf blight	1624
Hispa	1488
Healthy	490
Leaf smut	565
Total	**9857**

The RPDID dataset is a quite varied dataset [27], [28], Table 3, and to accurately test the model, the downsampling methodology is the same as the one use before in this work. These labelled images were then used as the testing data on various standard large pretrained image models. One thing in common with all these models, however, was that for testing only the models trained on the ImageNet dataset were used to maintain consistency with labelling of the generated SR images. This setup allowed truly unbiased testing of the images generated by the architecture showcased in this work. As seen from Table 4 shows consistently got results close to the top accuracies reported by these individual models.

Table 4. Comparison of average accuracy of RPDID classification

Model	**Accuracy on LR Images**	**Accuracy on Generated SR Images**
VGG-16	54.4%	72.9%
Inception V3	58.8%	77.3%
EfficientNet B0	63.8%	74.2%
ResNet-50	56.76%	81.6%

Fig. 5, also showcases both the accuracy and the loss for training and testing where we got the best results. From this, easily conclude the quality of the images generated by the architecture in this work can also extend the use case of such a model into a more real world setting without having to redesign the whole pipeline when it comes to image classification.

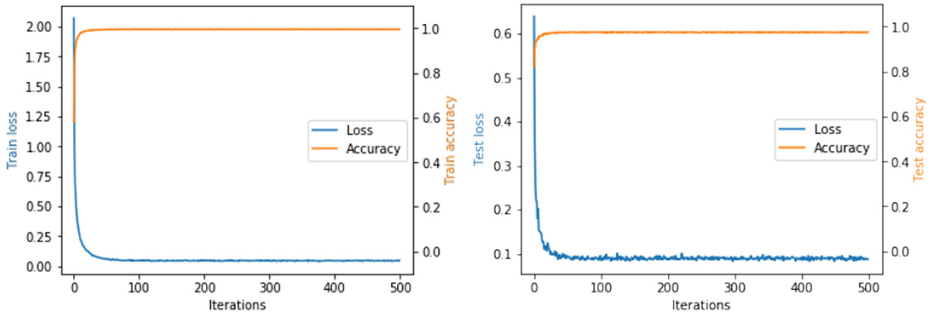

Fig. 2. Comparison of accuracy and loss over training and testing

7 Conclusion

This paper highlighted a deep learning model for the use case of super resolution problem that work on top of existing architectures such as the SRResNet. The proposed model ERCSR improves upon the original architectural designs by remaking the core component of the SRResNet - the residual block. In addition to the newly improved residual block, the proposed model also employs residual scaling to combat against the issues that all residual networks have. This residual scaling allows the maximum possible expansion of the network without causing the model itself to get unstable to the point of unviability as well as allow the same model to be used for SR at different scales. The two factors - the improved residual block and the residual scaling - allows the proposed model to achieve state of the art results while utilizing much lesser computation time and resources to other models. In future, in this way super resolution algorithm can be applied for object detection, segmentation of blobs etc. for more real-world problems.

References

1. Freeman, W. T., Jones, T. R., & Pasztor, E. C., Example-based super-resolution. IEEE Computer graphics and Applications, 22(2), 56–65, (2002).
2. Freedman, G., & Fattal, R., Image and video upscaling from local self-examples. ACM Transactions on Graphics (TOG), 30(2), 1–11, 2011.
3. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., & Change Loy, C.: Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European conference on computer vision (ECCV) workshops, (2018).
4. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems, 27, (2014).
5. Lugmayr, A., Danelljan, M., & Timofte, R. Unsupervised learning for real-world super-resolution. In 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3408-3416, IEEE, (2019).

6. Dong, C., Loy, C. C., He, K., & Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence, 38(2), 295-307, (2015).
7. Dong, C., Loy, C. C., & Tang, X.: Accelerating the super-resolution convolutional neural network. In: European conference on computer vision (pp. 391-407). Springer, Cham. (2016).
8. Kim, J., Lee, J. K., & Lee, K. M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1646-1654, (2016).
9. Simonyan, K., & Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, (2014).
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition pp. 770-778, (2016).
11. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., & Fu, Y.: Residual dense network for image super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2472-2481, (2018).
12. Lim, B., Son, S., Kim, H., Nah, S., & Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 136-144, 2017.
13. Tai, Y., Yang, J., & Liu, X.: Image super-resolution via deep recursive residual network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3147-3155, (2017).
14. Ahn, N., Kang, B., Sohn, K. A.: Fast, accurate, and lightweight super-resolution with cascading residual network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 252-268, (2018).
15. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition pp. 4681-4690, (2017).
16. Keys, R.: Cubic convolution interpolation for digital image processing. IEEE transactions on acoustics, speech, and signal processing, 29(6), 1153-1160, (1981).
17. Zhang, K., Zuo, W., Zhang, L.: Learning a single convolutional super-resolution network for multiple degradations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3262-3271, (2018).
18. 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, Springer, Cham, (2014).
19. Kim, J., Lee, J. K., Lee, K. M.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1637-1645, (2016).
20. Nah, S., Hyun Kim, T., & Mu Lee, K.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: Proceedings of the IEEE conference on computer vision and pattern recognition pp. 3883-3891, (2017).
21. Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence, (2017).

22. Timofte, R., Gu, S., Wu, J., & Van Gool, L.; Ntire 2018 challenge on single image super-resolution: Methods and results. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops (pp. 852-863, (2018).
23. Huang, J. B., Singh, A., & Ahuja, N.: Single image super-resolution from transformed self-exemplars. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5197-5206, (2015).
24. 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: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vol. 2, pp. 416-423, IEEE (2001).
25. 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.
26. Wang, Z., Chen, J., & Hoi, S. C.: Deep learning for image super-resolution: A survey. IEEE transactions on pattern analysis and machine intelligence, 43(10), 3365-3387, (2020).
27. Sethy, P. K., Barpanda, N. K., Rath, A. K., & Behera, S. K.: Deep feature-based rice leaf disease identification using support vector machine. Computers and Electronics in Agriculture, 175, 105527, 2020.
28. Prajapati, H. B., Shah, J. P., & Dabhi, V. K.: Detection and classification of rice plant diseases. Intelligent Decision Technologies, 11(3), 357-373, 2017.