




ECASR: Efficient Channel Attention Based Super-Resolution

Sameeran Borah and Nilkanta Sahu (✉) 

IIIT Guwahati, Guwahati, India
nilkanta@iiitg.ac.in

<http://iiitg.ac.in/faculty/nilkanta/>

Abstract. Despite recent advancements in single image super-resolution (SISR) methodologies, reconstruction of photo-realistic high resolution (HR) image from its single low resolution (LR) counterpart remains a challenging task in the fraternity of computer vision. In this work, we approach the problem of SR using a modified GAN with specialized Efficient Channel Attention (ECA) mechanism. CA mechanism prioritizes convolution channels according to their importance. The ECA mechanism, an extension of CA, improves model performance and decreases the complexity of learning. To capture the image texture accurately low-level features are used for reconstruction along with high-level features. A dual discriminator is used with GAN to achieve high perceptual quality. The experimental result shows that the proposed method produces better results for most of the dataset, in terms of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and mean-opinion-score (MOS) over the state-of-the-art methods on benchmark data-sets when trained with same parameters.

1 Introduction

Image Super-Resolution (SR), is the process of reconstructing an HR image from one or more LR image. SR is inherently a challenging ill-posed problem since there exist multiple HR image that corresponds to a single LR image. SR has its application in various image processing and computer vision tasks ranging from surveillance, medical imaging, object detection, satellite imaging to different image restoration and recognition tasks.

The SR algorithms can be broadly classified into two categories, one based on the number of input LR images the other based on the principle used to construct the HR image. Further, based on the principles used in the construction of SR image, the SR algorithms can also be categorized into three categories [24]: interpolation-based, model-based, and deep learning-based algorithms. Interpolation-based algorithms like bi-linear or bi-cubic interpolation use local information in an LR image to compute pixel values in the corresponding SR image, which are characterized by high computational efficiency. Prior knowledge is used in model-based algorithms such as the Maximum a Posteriori (MAP) to constrain the solution space whose performance is improved compared to the interpolation-based approach.

Supervised machine learning approaches learn the mapping function that maps LR images to its corresponding HR images from a large number of examples. The mapping function learned during the training phase is inverse of the down-sample function that is used to transform the HR image to its corresponding LR image, which can be known or unknown. With the help of known downgrade functions like bi-cubic, bi-linear down-sampling, the LR-HR pair can automatically be generated. This allows the creation of large training data-sets from a vast amount of freely available HR images which can be used for self-supervised learning (Figs. 1 and 2).

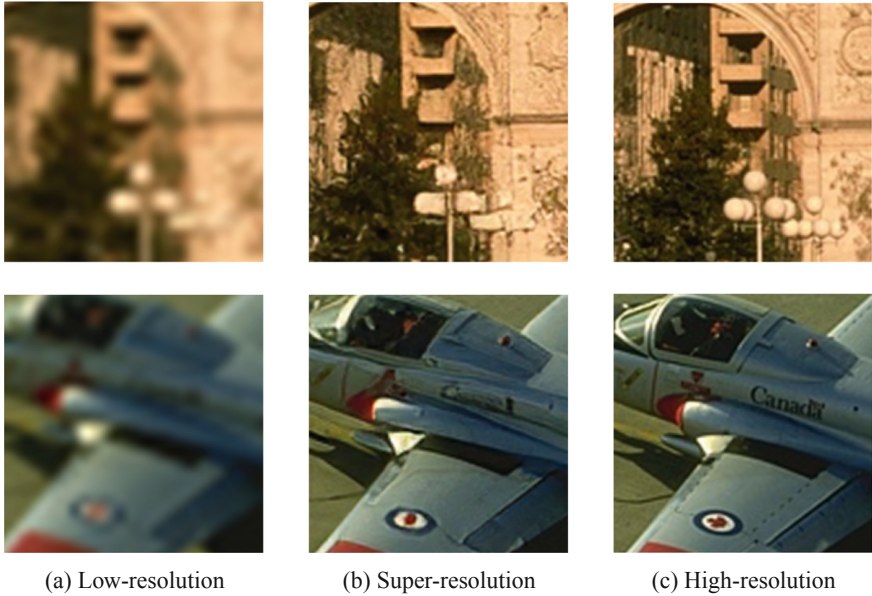


Fig. 1. Reconstructed Super-resolution images comparison with Original High-resolution and Low-resolution images.

This work proposes a GAN based SISR model that produces HR image with realistic texture details. Our contributions can be summarized as follows.

1. The generator architecture consists of multiple ECA blocks which emphasize on certain channels, along with it a 3-layer CNN network is added to the generator that extracts sufficient low-level features. The ECA block avoids dimensionality reduction step which destroys the direct relation with a channel and its weight, instead, we use 1D convolution to determine the cross channel interaction. The generator achieves state-of-the-art PSNRs when it is trained alone without discriminators.
2. The proposed SISR framework utilizes a dual discriminator network inspired by SRFeat [12] architecture, one that works on image domain that uses Mean Square Error (MSE) loss, the other that works on feature domain that uses perceptual feature loss.



Fig. 2. Example of super-resolution images from different models.

3. During the GAN training phase, the generator and the feature discriminator are trained on perceptual loss [12], which utilizes a pre-trained VGG19 network to calculate the difference of feature map extracts between the original HR image and the generated HR image.
4. The performance of the system is measured based on objective evaluation indicators such as PSNR/SSIM on several public benchmarks data-sets Set5, Set14, and BSD100 [16]. To find the perceptually better image, Mean Opinion Score (MOS) is calculated with the help of 10 people.

Comparative study shows that the proposed architecture with low model complexity significantly improve the performance of the model in terms of PSNR/SSIM and MOS score over the state-of-the-art methods [12, 13, 17, 22, 25] on benchmark data-sets when trained with same parameters.

2 Related Work

Before the application of deep learning in computer vision became popular in 2012 on-wards, the problem of super-resolution was approached using traditional computer vision techniques. Chang et al. in their work [3], used Bi-cubic interpolation for super-resolution. Bi-cubic interpolation and Lanczos re-sampling [15] are very computationally effective techniques, but they suffer from the drawback that they can't produce an accurate super-resolution image. The Markov Random Field (MRF) [23] approach was first embraced by Freeman et al. to investigate the accessible real images. Similarly, scientists applied sparse coding techniques to SISR [8] problems. Random forest [14] additionally accomplished a lot of progress in the reconstruction of the SISR. Many use a combination of reconstruction based as well as a learning-based approach to decrease the artifacts produced by external training examples.

In recent years, Deep Learning based Image Super-resolution models has demonstrated noteworthy improvement to reconstruction based and other learning-based methods. Dong et al. [5] first presented CNN for SISR reason, from that point forward there have been different enhancements to SISR methods utilizing Deep learning-based methodology [6, 10, 12, 13, 17, 21, 22, 25]. SRCNN had its downside since it utilized a shallow three-layer architecture, thus high-light features couldn't be caught. Kim et al. proposed VDSR [10], which was an

improvement to SRCNN by increasing the number of convolution layers to 20. Influenced by ResNet engineering Lim et al. [13] presented EDSR model which had long and short skip associations that helped to prepare profound SISR systems. Also removing Batch Normalization (BN) layers in the lingering residual network, they improved computational advantages. Even though the models so far created a high Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) score, they couldn't produce SR image with realistic feature details.

Ledig et al. [12] were probably the first to use GAN for the purpose of SR. A GAN [7] generally consists of two networks, one generates a fake/new image where the other one tries to find whether it's fake or not. Ledig et al. [12] used SRResNet as generator of GAN. SFTGAN [21] demonstrated that it is conceivable to recoup sensible surfaces by tweaking highlights of a couple halfway layers in a single system adapted on semantic division likelihood maps. SRFeat [17] is another GAN-based SISR strategy where the creators previously proposed the double discriminator, one chip away at picture space and other on highlight areas to deliver perceptually practical HR picture. Xintao et al. introduced ESRGAN [22], which is an improved version of SRGAN, where they use Residual in Residual (RIR) generator architecture. Also, they added realistic discriminator that estimates the probability that the given real data is more practical than fake data. They enhanced the SRGAN model by using features before activation to calculate the perceptual loss of the generator during the adversarial learning phase.

Attention Mechanism. Recent studies show that the uses of attention mechanisms enhance the performance of CNN networks for various tasks. Attention mechanism was first introduced in CNN for image classification task [9, 19, 20]. Hu et al. [9] utilized channel-wise inter-dependencies among various feature channels. [4, 25] used channel attention mechanism for SISR purpose. The dimensionality reduction step used in channel attention mechanism makes correspondence between the channel and its weight indirect. This limitation was surmounted by [18] with the attention mechanism that introduced a local cross channel interaction method using 1D convolution.

3 Network Architecture

The network architecture as shown in Fig. 3 utilizes a GAN based approach with a generator that produces super-resolution images along with a dual discriminator architecture that classifies the produced SR images into real and fake classes which helps the generator to produce realistic-looking images. The efficient channel attention mechanism used by the generator allows the network to focus on certain channels by modeling their inter-dependencies. The feature maps of the CNN network are then passed through two sub-pixel convolution layers where HR images are generated by upsampling.

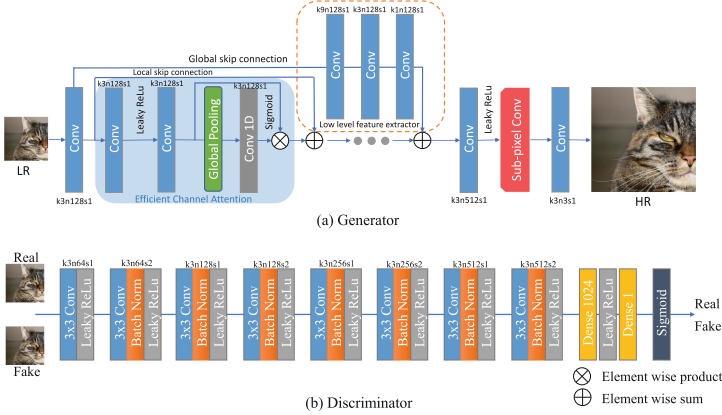


Fig. 3. ECASR architecture.

3.1 Generator

The LR image after passing through the first convolutional layer of the generator enters the high-level and low-level feature extractors

$$F_0 = H_{SF}(I_{LR}), \tag{1}$$

where F_0 , H_{SF} and I_{LR} denotes initial feature map, first convolutional layer and input low-resolution image respectively.

$$F_{HF} = H_{ECA}(F_0) \tag{2}$$

$$F_{LF} = H_{3CN}(F_0) \tag{3}$$

H_{ECA} denotes a deep feature extractor consisting of multiple ECA blocks which extract high-level features (F_{HF}) and H_{3CN} denotes the 3-layer shallow convolutional feature extractor that controls the flow of low-level features (F_{LF}) to the final output.

$$F_{Total} = F_{HF} + F_{LF} \tag{4}$$

F_{Total} is the total feature after adding both the features maps element-wise, the total feature map is then passed through the upsampling layer.

$$I_{SR} = H_{REC}(H_{UP}(F_{TF})) \tag{5}$$

H_{UP} and H_{REC} denotes the upsampling and reconstruction layer respectively that generates the super-resolution image.

Channel Attention. The resultant weight of the channel attention mechanism introduced in SENet [9] also utilized by [4, 25] can be expressed as

$$\omega = \sigma(\mathcal{W}_2 \delta(\mathcal{W}_1 \mathcal{G}(\chi))), \tag{6}$$

$\mathcal{G}(\chi)$ is the channel-wise global average pooling (GAP) of the output of a convolution block, $\chi \in \mathbb{R}^{W \times H \times C}$, where W , H and C are width, height and channel dimension. GAP can be expressed as

$$\mathcal{G}(\chi) = \frac{1}{WH} \sum_{i=1, j=1}^{W, H} \chi_{ij} \tag{7}$$

where $\delta(\cdot)$ and $\sigma(\cdot)$ indicates the Rectified Linear Unit [1] and Sigmoid activation function respectively. In order to avoid high computational complexities dimension of the channels are first reduced into $(\frac{C}{r})$ and then transformed back into (C) . This step of first reducing the dimension of channels into a low dimension space and then mapping it back to the original dimension makes indirect correspondence between the channel and its weight which degrades performance by losing valuable information. In this paper we use Leaky ReLU over ReLU activation which fixes the ‘‘dying ReLU’’ problem, as it doesn’t have zero-slope parts, also it provides additional benefit in training speed.

Efficient Channel Attention (ECA) [18]. Instead of using dimensionality reduction to reduce model complexity which destroys the direct correspondence between the channel and its weight, this work uses channel attention mechanism by efficiently using 1D convolution of kernel size k which. Given an output of a convolution block, χ , the resultant weight of efficient channel attention can be expressed as

$$\omega = \sigma(\mathcal{W}\mathcal{G}(\chi)), \tag{8}$$

where \mathcal{W} is a $C \times C$ parameter matrix. Let $y = \mathcal{G}(\chi)$ and $y \in \mathbb{R}^C$ where the weight of i^{th} channel (y_i) can be calculated by considering the interaction among y_i and its k neighbors that can be given as,

$$\omega_i = \sigma\left(\sum_{j=1}^k \omega^j y_i^j\right), y_i^j \in \Omega_i^k \tag{9}$$

where Ω_i^k indicates the set of k adjacent channels of y_i , ω_i is weight of y_i and ω^j is convolutional kernel. Equation (9) can be efficiently achieved by using 1D convolution as shown in Eq. (10)

$$\omega = \sigma(Conv1D_k(y)), \tag{10}$$

The size of the kernel can be determined with the Eq. (11) as mentioned in [18].

$$k = \left\lfloor \frac{\log_2(C)}{\gamma} + \frac{b}{\gamma} \right\rfloor_{odd}, \tag{11}$$

here γ and b are constants, we use 2 and 1 respectively in our training process. $|x|_{odd}$ denotes the nearest odd integer to x . Let F_{ECA-1} be the input to our ECA block, the weighted output based on channel attention mechanism can be finally expressed as

$$F_{ECA} = \omega \otimes (F_{ECA-1}) \tag{12}$$

3.2 Discriminator

The generator is coupled with a dual discriminator network which is similar to the architecture proposed by SRFeat [17]. One discriminator works on the image domain and the other works on the feature domain. The image discriminator is similar to the discriminator network used by SRGAN [12] while the feature discriminator has the same architecture as the image discriminator but the only difference is that the inputs to the feature discriminators are feature map extracts

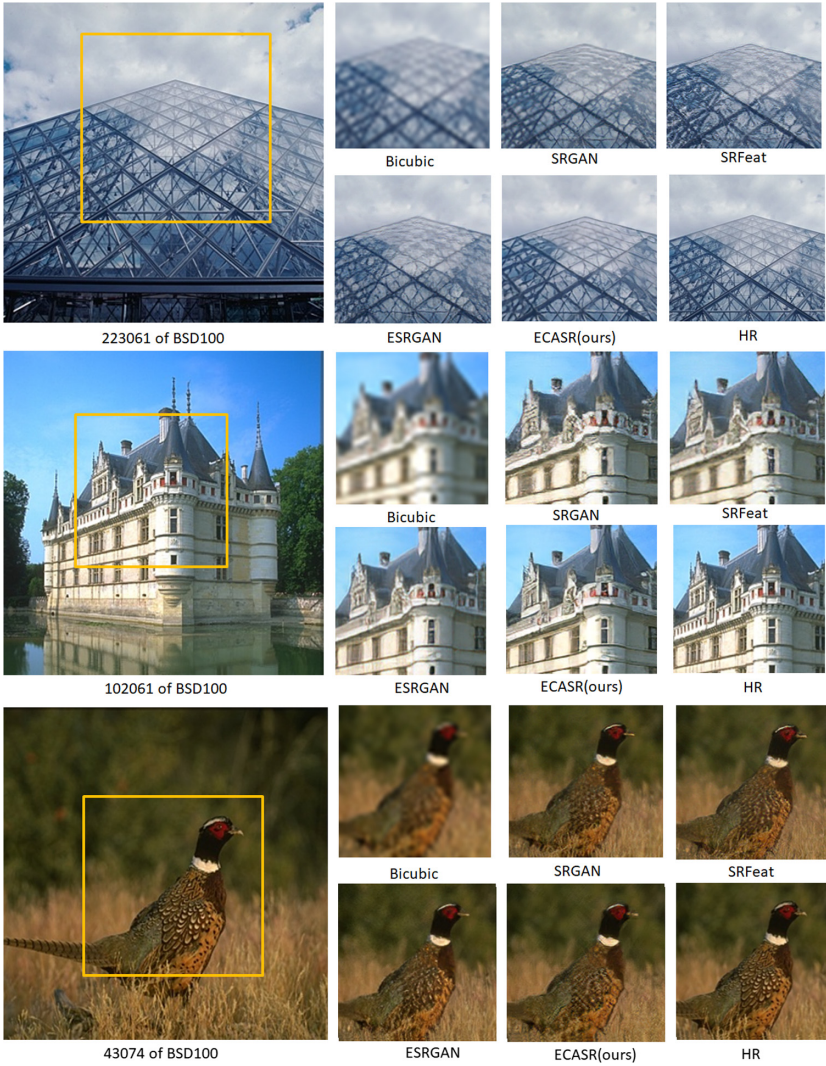


Fig. 4. Visual comparison of 4x SR on BSD100.

of the HR image and the super-resolved image from Conv5 of VGG-19 network. Both of them try to classify an image into a real and a fake class.

3.3 Loss Functions

A network that minimizes MSE loss tends to produce excessively smooth images. The generator network thus is pre-trained on Mean Absolute Error (MAE) loss that produces perceptually better-looking images to human eyes.

$$L_{MAE} = \frac{1}{WH} \sum_i^W \sum_j^H |I_{i,j}^h - I_{i,j}^g| \tag{13}$$

Here W, H indicates the dimension of the image. The objective of utilizing GAN system here is to improve the perceptual quality. GAN structure can be explained as a minmax game where the generator tries to minimize the discriminator’s gain and the discriminator tries to minimize the generator’s gain which can be characterized as below:

$$\min_g \max_d (E_{y \sim P_{data}(y)} [\log(d(y))] + E_{x \sim P_x(x)} [1 - \log(d(g(x)))] \tag{14}$$

Here $g(x)$ is the output of a generator network g for x where x is irregular noise. On the other hand $d(y)$ is the output of discriminator for y , where y is an example of genuine information distribution. $P_{data}(y)$ refers to distribution of real data and $P_x(x)$ is the distribution of generator output.

The discriminators, d^i and d^f denotes a pair of dual discriminators taking a shot at picture and feature area, separately. The generative adversarial procedure with a pre-trained generator and discriminators follows the loss-function characterized as:

$$L_g = L_p + \lambda(L_a^i + L_a^f) \tag{15}$$

where L_p is a perceptual similarity loss, L_a^i is a image GAN loss for the generator L_a^f is a feature GAN loss for the generator and λ is the weight for the GAN loss terms. To prepare discriminators d_i and d_f , we minimize the loss of L_d^i and, L_d^f which corresponds to L_a^i and L_a^f . The generator and discriminators are trained by thus limiting L_g, L_d^i and L_d^f . For adversarial learning, the discriminator uses loss functions introduced in [21]. There are by and large three-loss terms adding to the all-out loss, to be specific, perceptual similarity loss, image GAN loss, and GAN loss calculated based on features obtained by passing the super-resolution and high-resolution images through a VGG19 network. The description of each loss will be expressed as follows.

Perceptual Similarity Loss L_p . This loss processes the difference between two pictures in the feature area, rather than the pixel space, prompting all the more perceptually fulfilling outcomes. The perceptual loss is characterized as:

$$\frac{1}{W_m H_m C_m} \sum_i^{W_m} \sum_i^{H_m} \sum_i^{C_m} (\phi_{i,j,k}^m(I^h) - \phi_{i,j,k}^m(I^g))^2 \quad (16)$$

where W_m, H_m, C_m describes the dimensions of the m -th feature map extract from a pre-trained VGG network with $\phi_{i,j,k}^m$ indicating the feature map obtained by the j -th convolution (after activation) before the i -th maxpooling layer within the VGG network.

Image GAN Losses L_a^i and L_d^i . The image GAN loss term L_a^i for the generator and the loss function L_d^i for the image discriminator are characterized as:

$$L_a^i = -\log(d^i(I^g)), \quad (17)$$

$$L_d^i = -\log(d^i(I^h)) - \log(1 - (d^i(I^g))), \quad (18)$$

where $d^i(I)$ is the yield of the image discriminator d^i .

Feature GAN Losses L_a^f and L_d^f . The element GAN loss term L_a^f for the generator and the capacity L_d^f for the element discriminator are characterized as:

$$L_a^f = -\log(d^f(\phi^m(I^g))), \quad (19)$$

$$L_d^f = -\log(d^f(\phi^m(I^h))) - \log(1 - (d^f(\phi^m(I^g)))), \quad (20)$$

where $d^f(\phi^m)$ is the yield of the feature discriminator d^f .

4 Experiments

Experiments are done in two phases, in the first phase generator alone is trained on MAE loss, in the second phase the generator is trained on perceptual loss along with the dual discriminators. The performance of the pre-trained generator is evaluated and compared with other state-of-the-art approaches in terms of PSNR and SSIM score. Finally, the results obtained from the GAN based SISR network are evaluated and compared on PSNR/SSIM and MOS score, which proves the efficiency of our network.

Table 1. Comparison of PSNR/SSIM score with and without Low-level feature extractor after 64000 iterations.

PSNR/SSIM	w/o Low level feature extractor	with Low level feature extractor
Set4	30.06/0.8520	30.12/0.8544
Set14	27.22/0.7485	27.27/0.7501
BSD100	26.74/0.7098	26.76/0.7110

The training data-sets are obtained by bi-cubic down-sampling of the HR images. The cropped HR images are of the size 296×296 , while the LR images are of the size 74×74 , which are then again normalized to $[-1,1]$ intensity. DIV2K [2] data-set is used for training which consists of 800 HR training images and 100 HR validation images. The data-set is augmented into 160000 images by random cropping, rotating (90° , 180° , and 270°) and horizontally flipping. Publicly available data-sets Set5, Set14, BSD100 are used for validation. We train the generator network on Nvidia Titan X GPU for 3.2×10^6 iterations with a batch size of 16 which is optimized by Adam [11] optimizer. The learning rate is initialized 1×10^{-4} with a decay of 0.5 for every 1×10^5 iterations.

Table 2. Comparison with PSNR oriented pre-trained models SRGAN, EDSR, ESRGAN, SRFeat, RCAN and our proposed model ECASR on benchmark data [12, 13, 17, 22, 25]. [4× upscaling]

PSNR/SSIM	Set5	Set14	BSD100
Bicubic	28.42/0.8104	26.00/0.7027	25.96/0.6675
SRCNN	30.48/0.8628	27.50/0.7513	26.90/0.7101
VDSR	31.35/0.8830	28.02/0.7680	27.29/0.7260
SRGAN	31.67/0.8864	28.30/0.7770	27.40/0.7318
EDSR	32.01/0.8917	28.48/ 0.7832	27.53/ 0.7373
ESRGAN	31.83/0.8892	28.44/0.7801	27.46/0.7340
SRFeat	31.45/0.8821	28.14/0.7719	27.26/0.7271
RCAN	32.05/0.8920	28.57/0.7821	27.60 /0.7360
ECASR	32.12/0.8924	28.58 /0.7830	27.54/0.7371

Table 3. Comparison of GAN-trained models, SRGAN [12], SRFeat [17], ESRGAN [22] and the proposed model ECASR on benchmark data-set.

PSNR/SSIM	Set5	Set14	BSD100
SRGAN	29.158/0.8643	26.165/0.7791	25.459/0.5775
ESRGAN	29.752/0.8665	26.323/ 0.7854	25.505 /0.6279
SRFeat	29.420/0.8245	26.100/0.7850	25.417/0.5675
ECASR	29.864/0.8677	26.436 /0.778	25.461/ 0.6454

5 Results

In order to prove the effectiveness of the shallow low-level feature extractor, we do an ablation study. The network is trained and tested with and without the low-level feature extractor. Results are tabulated in Table 1. It shows that the use of the low-level feature extractor helps in achieving a high PSNR/SSIM score.

Table 4. MOS ratings of SRGAN, ESRGAN, SRFeat [12,17,22] and our Proposed Model ECASR on BSD100 [16].

Models	SRGAN	SRFeat	ESRGAN	ECAS (ours)
MOS	3.72	3.87	4.08	4.12

The performance of the generator network is evaluated and compared to bi-cubic interpolation and other state-of-the-art models which proves the efficiency of our proposed network. Quantitative results are summed in Table 2 that shows our model produces comparable results. EDSR and RCAN have higher values for Set14 and BSD100 which is because of their deeper architecture, if we stack up more ECA blocks in our network it should necessarily produce a higher benchmark score. Though the generator produces images that have high PSNR/SSIM values, they lack in perceptual quality. We then compare the performance of our GAN trained network as shown in Table 3 and Fig. 4 provides visual examples that show considerably better performance. The dip in the PSNR/SSIM score of our GAN trained network is presumably for the competition between the MSE based content loss and adversarial loss. We further obtained MOS ratings from a group of 10 people, who rated the images from 1 to 5 based on the reconstruction quality of the images, higher the better. Table 4 shows our proposed model has better MOS ratings compared to SRGAN, SRFeat, and ESRGAN.

6 Conclusion

In this paper, we have proposed a new GAN based SR scheme. In the proposed scheme, ECA method has been used, probably for the first time to solve SR problem. ECA tends to produce higher PSNR results with a smaller number of parameters. Considering low-level features with high-level features for super-resolution, both the fine texture details as well as overall description of the image are captured accurately. Dual discriminator helps the generator to create photo-realistic SR images. Comparison with various models with the same training data-set and parameters show the superiority of our proposed method. There is always a trade-off between model complexity and quality. Stacking up more ECA layers into our model, expected to result better SR images but it will also increase the training complexity.

References

1. Agarap, A.F.: Deep learning using rectified linear units (relu). arXiv preprint [arXiv:1803.08375](https://arxiv.org/abs/1803.08375) (2018)
2. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: dataset and study. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, July 2017

3. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2004, vol. 1, pp. I-I. IEEE (2004)
4. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 11065–11074 (2019)
5. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(2), 295–307 (2015)
6. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 391–407. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_25
7. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
8. Gu, S., Zuo, W., Xie, Q., Meng, D., Feng, X., Zhang, L.: Convolutional sparse coding for image super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1823–1831 (2015)
9. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
10. Kim, J., Kwon Lee, J., Mu Lee, K.: 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)
11. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
12. Ledig, C., et al.: 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)
13. 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)
14. Liu, Z.S., Siu, W.C., Huang, J.J.: Image super-resolution via weighted random forest. In: 2017 IEEE International Conference on Industrial Technology (ICIT), pp. 1019–1023. IEEE (2017)
15. Madhukar, N.: Lanczos resampling for the digital processing of remotely sensed images. In: Proceedings of International Conference on VLSI, Communication, Advanced Devices, Signals & Systems and Networking (VCASAN-2013), pp. 403–411 (2013)
16. 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 of the 8th International Conference on Computer Vision, vol. 2, pp. 416–423, July 2001
17. Park, S.-J., Son, H., Cho, S., Hong, K.-S., Lee, S.: SRFeat: single image super-resolution with feature discrimination. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11220, pp. 455–471. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01270-0_27
18. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: ECA-net: efficient channel attention for deep convolutional neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

19. Wang, F., et al.: Residual attention network for image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164 (2017)
20. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7794–7803 (2018)
21. Wang, X., Yu, K., Dong, C., Loy, C.C.: Recovering realistic texture in image super-resolution by deep spatial feature transform. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018
22. Wang, X., et al.: ESRGAN: enhanced super-resolution generative adversarial networks. In: Leal-Taixé, L., Roth, S. (eds.) ECCV 2018. LNCS, vol. 11133, pp. 63–79. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-11021-5_5
23. Wu, W., Liu, Z., Gueaieb, W., He, X.: Single-image super-resolution based on Markov random field and contourlet transform. *J. Electron. Imaging* **20**(2), 005–023 (2011)
24. Xu, Y., Yu, L., Xu, H., Zhang, H., Nguyen, T.: Vector sparse representation of color image using quaternion matrix analysis. *IEEE Trans. Image Process.* **24**(4), 1315–1329 (2015)
25. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11211, pp. 294–310. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_18