# MSCE: An Edge-preserving Robust Loss Function for Improving Super-resolution Algorithms

Ram Krishna Pandey, Nabagata Saha, Samarjit Karmakar, and A G Ramakrishnan

Department of Electrical Engineering, Indian Institute of Science, Bangalore, India ramp@iisc.ac.in, {snabagata, ksamarjit}@student.nitw.ac.in, agr@iisc.ac.in

**Abstract.** With the recent advancement in the deep learning technologies such as CNNs and GANs, there is significant improvement in the quality of the images reconstructed by deep learning based superresolution (SR) techniques. In this work, we propose a robust loss function based on the preservation of edges obtained by the Canny operator. This loss function, when combined with the existing loss function such as mean square error (MSE), gives better SR reconstruction measured in terms of PSNR and SSIM. Our proposed loss function guarantees improved performance on any existing algorithm using MSE loss function, without any increase in the computational complexity during testing. **Keywords:** Loss function, CNN, GAN, Super-resolution, mean square error, mean square Canny error, edge preservation, PSNR, SSIM

# 1 Introduction

Super-resolution is the process of obtaining a high resolution (HR) image from one or more low resolution (LR) images. Classical reconstruction based image super-resolution requires multiple low-resolution images with sub-pixel misalignment at the same scale, whereas single image super-resolution requires a database of LR and HR matched pairs to learn a mapping function between the patch pairs at different scales [1]. Given a low resolution image during testing, this learned function or representation can be used to reconstruct the corresponding HR image.

Since the advent of deep learning technologies in the past decade, superresolution algorithms have shown remarkable improvement in the quality of the reconstructed image. Most of the work reported in the literature have used mean square error (MSE) loss function to minimize the error between the reconstructed model output and the ground truth image [2] [3] [4] [5] [6] [7]. Minimizing this loss function may reduce the high frequency content in the reconstructed image and thus may blur the edges in it. Also, the reconstructed image may not lie precisely in the manifold of the HR image. Researchers have endeavored to find

#### 2 Ram Krishna Pandey et al.

ways to solve this problem to a good extent as can be seen in SRGAN [8], where the authors claim that the reconstructed output lies precisely in the manifold of HR images, even if the reconstructed images have less peak signal to noise ratio (PSNR) and structural similarity (SSIM). Ledig et al. [8] have used a weighted combination of MSE loss, content loss [9] and adversarial loss to reconstruct the HR image. This approach requires a deep architecture, such as the VGG net [10], to obtain the local covariance structure in the image. Most of the image transformation tasks use mean square error as loss function, which provides smooth transformed images.

Our main contributions in this paper are as follows:

- We have performed a large number of experiments to obtain a robust loss function that improves the performance of the existing algorithms that employ MSE loss function.
- While training, we apply Canny edge detector [11] the reconstructed output (in batches) and also separately on the corresponding ground truth image to compute the proposed mean square Canny error (MSCE) and assign weights (convex combination) based on our experiments i.e. the loss function can be given as:  $\mu \times MSE + (1 \mu) \times MSCE$ .
- Our approach guarantees performance improvement in terms of PSNR and SSIM over the existing approaches, if the model is trained on one dataset and tested on different datasets as mentioned in Tables 1 and 2.
- Our model does not incur additional overhead in terms of computation during testing to obtain the performance gain reported in Tables 1 and 2 due to our proposed MSCE loss function.

# 2 Related work

Super-resolution and image denoising can be assumed as image transformation tasks. In super-resolution, a LR image is fed to a transformation network such as a multilayer neural network to generate a HR image. Most of the image processing tasks such as image denoising and super-resolution minimize a per-pixel loss function to obtain reconstruction. In this work, our focus is on improving the quality of existing super-resolution algorithms such as SRCNN [2] and ESPCN [3] that use per-pixel loss function. Recently proposed perceptual loss function has shown significant improvement in the perceptual quality of the images. Simonyan et al [12] use perceptual loss for feature visualization. Gatys et al. [13] and [14] use perceptual loss for texture synthesis and style transfer, respectively. These approaches solve an optimization problem and hence, are slower.

Justin Johnson et al [15] and Pandey et al [16] use the benefits of perpixel as well as perceptual loss functions and propose a computationally efficient, optimization-free approach that provides results for image transformation tasks that are qualitatively similar to those of the above optimization-based approaches. The super-resolution algorithm SRGAN [8] uses a weighted combination of three different loss functions, namely mean square error, perceptual and adversarial loss to obtain a sharper reconstruction. The images reconstructed by these methods perceptually look sharper, even if they have low values of PSNR and SSIM.

In this work, our focus is on improving the perceptual quality, PSNR and SSIM without incurring any additional computational overhead during testing by the addition of a new, robust loss function that aims to preserve the edge information.

## 3 The proposed edge-preserving MSCE loss function

We employ Canny edge detector [11] to detect the edges in the reconstructed and ground truth images. We have chosen this algorithm, since Canny operator provides the most reliable edges amongst all the edge detection algorithms in the literature, and also satisfies all the general edge detection criteria.

Most of the recent papers on image super-resolution and denoising use mean square error as the loss function. This loss function may smooth the edge components in an image. We thought of preserving the edges by defining the loss function as a convex combination of mean square error loss and our edge preserving loss as follows:

Suppose the training set consists of image pairs  $\{L_i, H_i\}$ ; i = 1...N, where N is the total number of training examples. The model  $\Theta$ , parameterized by  $\lambda$ , predicts the output  $O_j$  for a given input  $L_j$ . Let C denote the Canny operator. Let  $C(\Theta_{\lambda}(L_j))$  be the resultant image obtained by applying Canny operator on the predicted output image,  $O_j = \Theta_{\lambda}(L_j)$ . The proposed edge preserving loss function, called the mean square Canny error - (MSCE) is given by :

$$Loss = \underbrace{\mu \times \frac{1}{N} \sum_{j=1}^{N} \| \Theta_{\lambda}(L_j) - H_j \|_F}_{MSE \ Loss \ (l_{mse})} + \underbrace{(1-\mu) \times \frac{1}{N} \sum_{j=1}^{N} \| C(\Theta_{\lambda}(L_j)) - C(H_j) \|_F}_{Edge \ preserving \ loss \ (l_{edge})}$$
(1)

The first term in the equation above is the mean square loss function used to minimize the error between the reconstructed output and the ground truth image. The second term in the loss function is the edge preserving loss function. After a large number of experiments, the weighing factor  $\mu$  has been fixed to lie in the range  $0.8 \le \mu \le 0.99$ . To minimize this loss function, Adam optimizer [17] is used with learning rate (lr) = 0.001,  $\beta_1 = 0.999$  and  $\beta_2 = 0.99$ .

### 3.1 Choosing the value of $\mu$

- Exhaustive Experimentation : We performed experiments varying  $\mu$  in the range  $0.8 \le \mu \le 0.99$  by incrementing its value by 0.01 each time. We found that the models were consistently giving better results for the particular values of  $\mu = 0.84$ , 0.85 and 0.86. For the results reported in the Figures 1, 2, 3 and 4 and Tables 1 and 2, the value of  $\mu$  used is 0.85.

#### 4 Ram Krishna Pandey et al.

- Dynamic choice of  $\mu$ : While performing the experiments, we found that sometimes, values of  $\mu$  (still in the range  $0.8 \leq \mu \leq 0.99$ ) other than the three specific ones mentioned above, gave better results. We made a list of those different values of  $\mu$  and tried each of them parallely in each epoch. For the subsequent epoch, we select the model corresponding to the least value of the loss function. Let  $l_{mse}$  and  $l_{edge}$  denote the mean square error loss and our edge-preserving loss, respectively, as mentioned in equation 1. Let  $\{\hat{\lambda}, \hat{\mu}\}$  be the optimal model parameters and  $\mu$  be the weighing parameter currently chosen during training. In each epoch, we selected the value of  $\mu$ that minimized the loss function in the right hand side of equation 2:

$$\hat{\mu} = \underset{\mu}{\operatorname{argmin}} \{ \mu \times l_{mse} + (1 - \mu) \times l_{edge} \}$$
(2)

We used the earlier approach for calculating the loss in our experiments, results for which have been reported in the Tables. A dynamic choice of the value of  $\mu$  gives similar results in less number of epochs. It can be experimented further to possibly achieve still better results.

## 4 Datasets used for training and testing

The models are trained on DIV2K [18] training dataset with the original architecture (without changing the architectural details of the existing model) proposed in the respective papers. We have performed testing on different datasets such as Set5 [19], Set14 [20], BSD [21] and URBAN100 [22] for the different scale factors of 2, 3, 4 and 8. We have found that there is consistent performance gain over the original models, in terms of PSNR and SSIM, on all the datasets on which our MSCE loss function has been tested so far. These results are seen quantitatively in Tables 1 and 2.

## 5 Experiments and Discussion

We have performed extensive experiments on different super-resolution algorithms proposed recently, by augmenting the original loss function with our proposed mean square Canny error loss function.

We have validated the effectiveness of our proposed MSCE loss function on the recent techniques of SRCNN [2] and ESPCN [3]. We found that the addition of our MSCE loss leads to better results and the improvement is consistent on both methods across different upscaling factors of 2, 3, 4 and 8.

## 6 Results

Figures 1, 2, 3 and 4 show both the results qualitatively: one obtained by passing the input image directly to the original models SRCNN [2] and ESPCN [3] with the loss functions used in the original papers, and the other obtained by augmenting the loss function with our MSCE loss function.

Tables 1 and 2 list the quantitative results obtained by the two superresolution methods on the datasets Set5, Set14, URBAN and BSD for different upscaling factors and the corresponding values obtained after they are modified by our MSCE loss function.

Note 1: Table 1 lists the results obtained from the LR images created by downsampling using normal bicubic interpolation. Whereas, the results reported in Table 2 are obtained by blurring the image by a Gaussian filter with radius 2 and then downsampling by bicubic interpolation to obtain the LR images at different scales.



(a) SRCNN original (b) SRCNN MSCE (c) ESPCN original (d) ESPCN MSCE

**Fig. 1.** Qualitative comparison of the results for an upscale factor of 2, when the ppt image from Set14 is directly fed to the original model and the model modified with MSCE loss trained by us. (a) The output image reconstructed by the original SRCNN model. (b) The output image reconstructed by SRCNN model modified with MSCE loss function. (c) Image reconstructed by the original ESPCN model. (d) Output image reconstructed by ESPCN model modified with MSCE loss function.

# 7 Conclusion

A large number of research papers have been published in the recent past by designing different models or algorithms that work reasonably well. The unique contribution of our work is that it improves the performance of any existing method, rather than proposing another technique. In this paper, we have proposed a robust edge-preserving loss function that adds performance gain in terms of PSNR and SSIM to any existing model, without increasing the computational cost involved in testing. We train the existing model by adding weighted Canny edge based loss. Minimizing this loss function helps to preserve the edges by giving more weightage to the edges. As shown by the Tables of results, the PSNR and SSIM values obtained after including our MSCE loss function are consistently better.



(a) SRCNN original (b) SRCNN MSCE (c) ESPCN original (d) ESPCN MSCE

**Fig. 2.** Comparison of the results for an upscale factor of 3, when the comic image from Set14 is directly fed to the original model and the model modified with MSCE loss trained by us. (a) Output image reconstructed by the original SRCNN model. (b) Output image reconstructed by SRCNN model modified by MSCE loss function. (c) Output image reconstructed by the original ESPCN model. (d) Output image reconstructed by MSCE loss function.



(a) SRCNN original (b) SRCNN MSCE (c) ESPCN original (d) ESPCN MSCE

**Fig. 3.** Comparison of the results for an upscale factor of 4, when the baby input image from Set5 is directly fed to the original model trained by us and the model modified with MSCE loss trained by us. (a) Output image reconstructed by the original SRCNN model. (b) Output image reconstructed by SRCNN model modified by MSCE loss function. (c) Output image reconstructed by the original ESPCN model. (d) Output image reconstructed by ESPCN model modified by MSCE loss function.



(a) SRCNN original (b) SRCNN MSCE (c) ESPCN original (d) ESPCN MSCE

Fig. 4. Comparison of the results for an upscale factor of 8, when the baboon input image from Set14 is directly fed to the original model trained by us and the model modified with MSCE loss trained by us. (a) Output image reconstructed by the original SRCNN model. (b) Output image reconstructed by SRCNN model modified by MSCE loss function. (c) Output image reconstructed by the original ESPCN model. (d) Output image reconstructed by ESPCN model modified by MSCE loss function.

**Table 1.**  $P_*$  and  $S_*$  are the PSNR and SSIM values obtained by the method \* for the upscaling factors of 2, 3, 4 and 8, whereas  $P_*^c$  and  $S_*^c$  are the corresponding PSNR and SSIM values obtained after augmenting the loss function by the MSCE loss function designed by us. All the models other than bicubic (non-learnable) have been trained on DIV2K training dataset. For testing, we have used 4 datasets, namely Set5, Set14, Urban and BSD.

Dataset		$P_{bicubic}$	$S_{bicubic}$	$P_{srcnn}$	$S_{srcnn}$	$\mathbf{P_{srcnn}^{c}}$	$\mathbf{S_{srcnn}^{c}}$	$P_{espcn}$	$S_{espcn}$	$\mathbf{P^{c}_{espcn}}$	$\mathbf{S^c_{espcn}}$
Set5	2x	27.02	0.92	28.44	0.93	28.57	0.93	26.48	0.92	26.59	0.92
	3x	25.41	0.89	26.59	0.90	26.75	0.91	25.882	0.91	25.888	0.91
	4x	21.96	0.79	23.22	0.82	23.37	0.83	22.35	0.81	22.49	0.82
	8x	18.10	0.61	18.740	0.63	18.743	0.63	18.33	0.62	18.43	0.62
Set14	2x	24.10	0.86	25.22	0.88	25.32	0.88	23.50	0.87	23.56	0.87
	3x	22.65	0.81	23.62	0.84	23.68	0.84	23.06	0.84	23.06	0.84
	4x	20.01	0.70	20.96	0.73	21.04	0.73	20.12	0.71	20.32	0.72
	8x	17.13	0.53	17.57	0.56	17.58	0.56	17.20	0.56	17.27	0.56
Urban	2x	20.66	0.84	22.26	0.87	22.44	0.87	21.38	0.87	21.42	0.87
	3x	20.22	0.79	21.47	0.83	21.53	0.83	21.18	0.83	21.18	0.83
	4x	16.92	0.65	17.81	0.69	17.84	0.69	17.54	0.70	17.59	0.70
	8x	14.63	0.48	15.04	0.50	15.04	0.50	14.94	0.507	14.99	0.509
BSD	2x	25.88	0.89	25.96	0.90	26.18	0.90	23.36	0.87	23.41	0.88
	3x	21.86	0.77	22.49	0.81	22.54	0.81	22.34	0.81	22.35	0.81
	4x	21.43	0.73	22.08	0.77	22.13	0.77	21.28	0.76	21.41	0.77
	8x	18.43	0.57	18.78	0.59	18.81	0.59	18.47	0.587	18.58	0.589



**Fig. 5.** Comparison of the results obtained on down-sampled (by bicubic interpolation without blurring) images on different upscaling factors. (a), (b), (c) and (d) have been down-sampled by a factor of 2 and reconstructed. (e), (f), (g) and (h) have been down-sampled by a factor of 3 and reconstructed. (i), (j), (k) and (l) have been down-sampled by a factor of 4 and reconstructed. (m), (n), (o) and (p) have been down-sampled by a factor of 8 and reconstructed.

**Table 2.**  $P_*$  and  $S_*$  are the PSNR and SSIM values obtained by the method \* at the different upscaling factors of 2, 3, 4 and 8, whereas  $P_*^c$  and  $S_*^c$  are the corresponding PSNR and SSIM values obtained by augmenting the loss function by the MSCE loss function designed by us. All the models other than bicubic (non-learnable) are trained on DIV2K (blurred by Gaussian blurring, then downsampled by bicubic) training dataset. For testing, we use 4 datasets, namely Set5, Set14, Urban and BSD.

Dataset		$P_{bicubic}$	$S_{bicubic}$	$P_{srcnn}$	$S_{srcnn}$	$\mathbf{P_{srcnn}^{c}}$	$\mathbf{S_{srcnn}^{c}}$	$P_{espcn}$	$S_{espcn}$	$\mathbf{P^{c}_{espcn}}$	$S_{espcn}^{c}$
Set5	2x	21.10	0.77	23.96	0.83	24.06	0.84	21.21	0.75	21.87	0.79
	3x	21.63	0.79	22.38	0.85	24.75	0.86	22.50	0.80	22.88	0.83
	4x	20.12	0.72	21.92	0.78	21.94	0.78	21.53	0.77	21.92	0.78
	8x	17.72	0.59	18.17	0.61	18.34	0.61	18.48	0.613	18.56	0.614
Set14	2x	19.35	0.67	21.75	0.76	21.78	0.76	19.50	0.67	20.08	0.70
	3x	19.84	0.69	20.64	0.77	22.25	0.78	20.63	0.73	20.99	0.74
	4x	18.67	0.62	20.02	0.69	20.07	0.69	19.75	0.68	20.04	0.69
	8x	16.84	0.52	17.16	0.54	17.29	0.54	17.35	0.55	17.43	0.54
Urban	2x	16.57	0.63	18.87	0.74	18.86	0.74	16.93	0.63	17.33	0.66
	3x	17.63	0.66	18.96	0.76	20.03	0.76	18.68	0.71	18.87	0.72
	4x	15.85	0.58	17.05	0.65	17.07	0.65	16.96	0.64	17.05	0.64
	8x	14.42	0.47	14.74	0.49	14.81	0.49	14.96	0.49	14.97	0.48
BSD	2x	21.00	0.71	23.27	0.80	23.28	0.80	20.92	0.71	21.67	0.75
	3x	19.82	0.66	20.27	0.75	21.59	0.75	20.30	0.70	20.60	0.72
	4x	20.13	0.67	21.32	0.73	21.35	0.73	21.06	0.73	21.38	0.73
	8x	18.18	0.56	18.38	0.58	18.51	0.58	18.70	0.58	18.70	0.58

# References

- 1. Glasner, D., Bagon, S., Irani, M.: Super-resolution From a Single Image. In: 12th IEEE International Conference on Computer Vision (pp. 349-356) (2009).
- 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).
- Shi, W., Caballero, J., Huszr, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, Wang, Z.: Real-time Single Image and Video Super-resolution Using an Efficient Sub-pixel Convolutional Neural Network. In: IEEE Conference on Computer Vision and Pattern Recognition (pp. 1874-1883) (2016).
- Pandey, R. K., Ramakrishnan, A., G.: A Hybrid Approach of Interpolations and CNN to obtain Super-Resolution. arXiv preprint arXiv:1805.09400 (2018).
- Pandey, R. K., Ramakrishnan, A. G.: Language Independent Single Document Image Super-Resolution Using CNN for Improved Recognition. arXiv preprint arXiv:1701.08835. arXiv:1701.08835 (2017).
- Pandey, R. K., Ramakrishnan, A. G.: Efficient Document-Image Super-resolution Using Convolutional Neural Network. Sadhana, 43(15), (2018).
- Pandey, R.K., Maiya, S.R., Ramakrishnan, A.G.: A New Approach for Upscaling Document Images for Improving their Quality. In: 14th International IEEE India Conference INDICON 2017, IIT Roorkee, Dec. 15-17 (2017).
- 8. Ledig, C., Theis, L., Huszr, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A.P., Tejani, A., Totz, J., Wang, Z. and Shi, W.: Photo-realistic single image super-

#### 10 Ram Krishna Pandey et al.

resolution using a generative adversarial network. In: Proc. IEEE Conf. Comput. Vis. and Pattern Recog. (2017).

- Johnson, J., Alahi, A., Fei-Fei, L.:Perceptual Losses for Real-time Style transfer and Super-resolution. In: European Conf. Comput. Vis. (ECCV), pages 694-711. Springer (2016).
- Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-scale Image Recognition. In: Int. Conf. Learning Representations (ICLR) (2015).
- 11. Canny, J.: A computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, (6), 679-698 (1986).
- Simonyan, K., Vedaldi, A. Zisserman, A.: Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv preprint arXiv:1312.6034 (2013).
- Gatys, L.A., Ecker, A.S., Bethge, M.: Texture Synthesis Using Convolutional Neural Networks. In: Advances in Neural Information Processing Systems 28 (2015).
- Gatys, L.A., Ecker, A.S., Bethge, M.: A Neural Algorithm of Artistic Style. arXiv preprint arXiv:1508.06576 (2015).
- Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual Losses for Real-time Style Transfer and Super-resolution. In: European Conference on Computer Vision. Springer, Cham (2016).
- Pandey, R.K., Karmakar, S., Ramakrishnan, A.G.:Computationally Efficient Approaches for Image Style Transfer. arXiv preprint arXiv:1807.05927 (2018).
- Kinga, D., J. Ba.:Adam: A method for Stochastic Optimization. In: Proc. Int. Conf. Learning Representations (ICLR) (2015).
- Agustsson, E., Timofte, R.:NTIRE 2017 Challenge on Single Image Superresolution: Dataset and Study. In: Proc. IEEE Conf. Comput. Vis. and Pattern Recog. (CVPR) Workshops (2017).
- Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.:Low-complexity Single-image Super-resolution Based on Nonnegative Neighbor Embedding. Proc. BMVC (2012).
- 20. Zeyde, R., Elad, M., Protter, M.:On Single Image Scale-up Using Sparserepresentations. In: Proc. Int. Conf. Curves and Surfaces, Avignon-France (2010).
- 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. Proc. 8th Int. Conf. Comput. Vis., vol. 2, 416–423 (2001).
- Huang, J.B., Singh, A., Ahuja, N.:Single Image Super-resolution from Transformed Self-exemplars. In: Proc. IEEE Conf. Comput. Vis. and Pattern Recog., 5197–5206 (2015).