Image Restoration and Enhancement using Deep Learning

Vaibhav Singh
Computer Department
Thakur College of Engineering and Technology
Mumbai, India
vaibhav.singh.3001@gmail.com

Vibha Singh
Computer Department
Thakur College of Engineering and Technology
Mumbai, India
svibha081@gmail.com

Virag Dosani
Computer Department
Thakur College of Engineering and Technology
Mumbai, India
viragdosani1699@gmail.com

Dr. Sheetal Rathi
Computer Department
Thakur College of Engineering and Technology
Mumbai, India
sheetal.rathi@thakureducation.or
g

ABSTRACT — Images and videos have always been susceptible to either damage due to human or natural factors such as motion blur, noise, camera misfocus, weather conditions, etc. In this project, our goal is to tackle the problem of restoration and enhancing degraded images. Image restoration is a classic low-level vision problem, aimed at the reconstruction of high-quality images from distorted low-quality inputs. Image enhancement is about improving the perceptual experience of images for human viewers or providing 'better' inputs for other automated image processing tasks. Traditional image restoration and enhancement techniques have been surveyed in order to provide more insight into the nature of the problem. In recent years, there's been an increase in the use of deep learning-based approaches for tackling these problems and are shown to out-perform traditional techniques. In this project, we explore and implement these deep learning-based models for 4 common problems such as degradation in images due to different weather factors, images containing a tear marks or dead pixels, images without any color information i.e. grayscale images and images having small dimensional structure i.e low-resolution images. Using a deep learning-based approach helps us in better formulating the solutions for these tasks.

Keywords — ***Image Colorization, Image Dehazing, Image In-painting, Image Super-resolution, Deep Learning, Convolutional Neural Network, Generative Adversarial Network.***

I. INTRODUCTION

Image restoration is the process that is used to restore old, broken, dead pixels, and physical images into new good quality, repaired images into digital form. It includes processes like image in-painting, image colorization, etc.

Image Enhancement is the process in which remove impurities present in an image as well as improving its quality also. It includes processes like image dehazing, image super-resolution, etc.

Doing this restoration and enhancement requires a lot of human effort and good knowledge of image editing and manipulation along with high power computing resources. Our project is a combination of both Image restoration as well as enhancement using Deep Learning which leads to minimizing human effort and can generate the same or better quality output in less time. The different techniques used are:

- Image Colorization is the process of taking an input grayscale (black and white) image and then producing an output colorized image that represents the semantic colors and tones of the input.
- Image Dehazing includes tasks such as removing weather or natural impurities from the image, focusing on actual real-world things to make the image more attractive.
- Image Inpainting is the process of reconstructing damaged/missing parts of an image.
- Image Super-resolution is a process of enhancing or increasing the resolution of an input image with minimal degradation in quality.

II. LITERATURE REVIEW

A brief review of Image Restoration [1] techniques with different methods and evaluation of those methods is given in it. Different methods and models to remove blur, noise, distortion, etc from images are provided in it.

Image In-Painting is the process of completing or recovering the missing region in the image or removing some objects added to it. Image In-painting is the technique of reconstructing the damaged or missing part of the image. The use of partial convolutions [2] is working well in situations when the degradation present is irregular or free-form fashion.

Image dehazing is a highly desired task in various image processing and computer vision applications that aims to lessen haze effects in a hazy input image.

Image Dehazing is due to weather conditions image quality degrades and to remove this image dehazing with the help of computer vision is described and various evaluation methods are also shown and compared as well.

Future research directions are also mentioned in it. Following the simplified hazy image degradation model, to achieve a reliable haze removal system, two significant estimates are required: transmission map (t) and atmospheric light (AL).

Image colorization is an image processing technique for colorizing grayscale or B/W images and videos. Recently, deep learning techniques progressed notably for image colorization.

Image Colorization is performed by using CNN and useful hints. This process converts black & white or grayscale images into colored ones with proper coloring according to hints and the colorized image is received as an output of this process.

Image super-resolution is the process of converting low-resolution images to high-resolution images in a realistic way by combining multiple images and producing better HD images as output in less time. There are many methods that work on a single image and generate good results using deep learning methods.

Supervised machine learning approaches, on the other hand, learn mapping functions from low-resolution images to high-resolution images from a large number of examples. Super-resolution models are trained with low-resolution images as input and high-resolution images as targets.

III. OBJECTIVES

The objective of our project is to provide a single application for problems such as low-resolution images, image tearing, black and white images, effects of weather conditions.

The end goals of this project are to provide the following solutions:

A) Image in-painting methods to help with the removal of tears or unwanted objects in an image.

B) Image super-resolution techniques to help with increasing the resolution of low-resolution blurry images.

C) Image dehazing to help in recovering for removing effects of weather conditions such as fog, dust, mist, etc.

D) Image colorization techniques to help in converting grayscale images to color images.

IV. PROPOSED SYSTEM

The proposed system is divided into four parts. Which can be separately used as per requirement or sequentially. Parts are as follows:

A. *IMAGE COLORIZATION*

The task of automatic image colorization is to takean input grayscale (black and white) image and then produce an output colorized image that represents the tones and semantic colors of the input. Since image colorization can be considered as an image-to-image translation problem that works by mapping a high dimensional (grayscale) input to high dimensional (colored) output it brings into consideration that not only we need to be able to reconstruct the original image dimensions but also to provide the color information for each pixel.

In recent years, there have been many research papers using deep learning-based approaches to tackle this problem. In this paper, we are using a deep learning-based approach that utilizes a GAN[3] based architecture as suggested in the paper[7] along with using the L*A*B* color space for the input images as opposed to using RGB images that are susceptible to small perturbations in intensity values.

NETWORK ARCHITECTURE:

The GAN-based architecture is made up of a generator and a discriminator network. For the generator network, we used a convolutional network that is made up of an encoder and a decoder part. Due to an information bottleneck in between, the features from the encoding path are concatenated with the corresponding decoder layer output in the expansive path of the network. This helps in the flow of the low-level information in the network. This also makes the input and output share the locations of prominent edges in grayscale and colored images. These types of architectures are called U-Net[8].

The discriminator network has the same basic architecture as the encoding path of the generator, with each layer consisting of a Convolutional, a Batch Normalization, and a Leaky ReLU layer as the number of channels increases from in powers of 2, from 64 to 512 on each step forward.

Recently many of the prominent pre-trained image classification models are great at determining the shapes and textures in images within their first few convolutional layers of the model. In this work, we use a pre-trained Resnet-34[9] architecture as the backbone of our generator network, this helps the generator network to learn quickly and effectively determine the differences in the structure and semantics of different objects in the image.

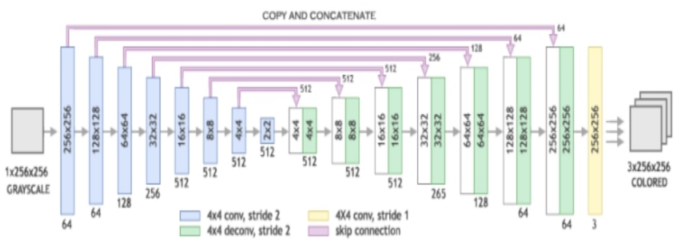

Fig. 1. Baseline generator Architecture

METHODOLOGY:

The dataset used for this project is the training images from the COCO2014[10] dataset. We randomly sampled over 10000 images from the dataset for our training purpose. The input and output image sizes are 256x256. The model has been trained in two stages. In Stage 1, we train only the generator of the model in a supervised and deterministic manner using an *L1* loss for 100 epochs with a learning rate of 0.0002.

This warm-up training helps the generator to avoid starting from scratch and helps in avoiding sudden fluctuations and instability during the early stages of the training. This is because, neither the generator nor discriminator knows anything about the task at the beginning of training.

In Stage 2, the generator is then trained with the binary cross-entropy log loss and pixel (*L1*) loss. The weightage for pixel loss is taken as 100.0 which forces the generator to produce images similar to the ground truth. We start with a learning rate of 0.0002 and 0.0004 for the generator and discriminator respectively and decay the learning rates by a factor of 10 whenever the loss function starts to plateau. We trained it for 200 epochs and stopped training when there wasn't any noticeable improvement in the results. We used the Adam[20] optimizer with 0.5 and 0.999 values for beta1 and beta2 respectively for both stages of the training.

EXPERIMENTATION RESULTS:

For performance measurement, we use the mean absolute error (MAE) and accuracy metric. MAE is computed by calculating mean of the absolute difference between the generated and source images on a pixel level for each color channel. Accuracy is the ratio between the number of pixels having the same color information as the ground truth image and the total number of pixels. Any two pixels are considered to have the same color if their underlying color channels lie within some threshold distance epsilon (ε).

TABLE I. EXPERIMENTATION RESULTS OF COLORIZATION

Epsilon (ε)	Mean Absolute Error	Accuracy
2%	11.2	22%
5%	8.6	53%

QUALITATIVE RESULTS:

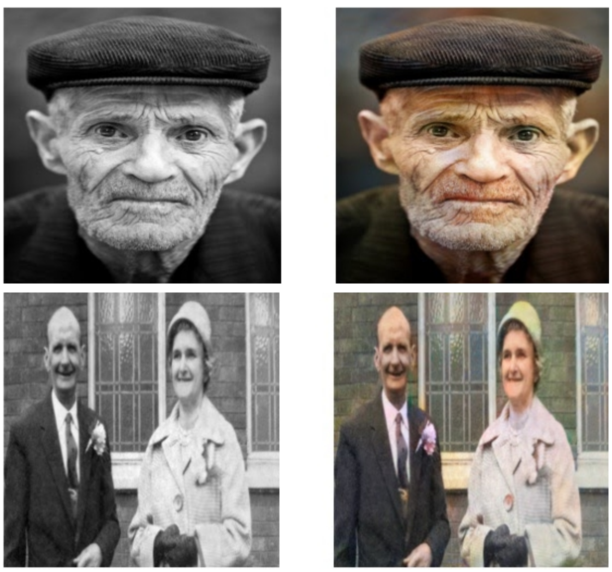

Fig. 2. Image Colorization Results Left-side - Gray Scale Images, Right-side - Colored Images

B. *IMAGE DEHAZING*

The task of single image dehazing is a fundamental low-level vision task. The problem of images obtained under a haze or the existence of smoke or any other weather conditions has serious color attenuation, low contrast and saturation, and poor visual effects. These images are often subject to color distortion, blurring, and other visible quality degradation which affect various systems that rely on optical imaging instruments, such as aerial photography systems, or other visual tasks such as object detection, tracking, etc.

NETWORK ARCHITECTURE:

In this work, we use an end-to-end convolutional neural network as suggested in this paper[21]. The paper utilizes techniques such as skip-connection and the attention mechanism to design a basic block consisting of multiple local residual learning skip connections and feature attention. This helps in improving the model performance and also allows us to train deeper and stable neural networks. The model architecture is made up of an end-to-end feature fusion attention layer that uses the novel feature attention module consisting of a channel and pixel attention mechanism. These are used to help give unequal weights to different image depths and pixel positions of the image. The argument for having an unequal weighting is that the haze unequally affects the image, i.e the degradation at different depths and different positions in the image is not the same. The use of the attention mechanism helps us to find and give more weightage to the portions of an image that have considerably more degradation than the rest of the image. For our use-case, we use a network that consists of 3 group architectures each containing 12 basic blocks with local residual learning. The basic blocks are made up of skip connections and the feature attention module.

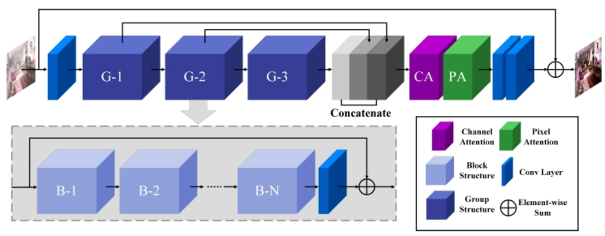

Fig. 3. Feature Fusion Attention network (FFA-Net) Architecture

METHODOLOGY:

The dataset used for this project is the RESIDE[22] outdoor training set which contains 72,135 synthetic hazy images with corresponding ground truth images, Due to model complexity and resource constraints we utilize 8000 images for training. For evaluation purposes, we use the standard SOTS dataset which consists of 500 images. The model was trained on an RGB image and we augment the dataset by randomly applying different 90 degree spaced rotations and occasional horizontal flips. The model takes a 240x240 dimensional hazy image as input. The model is optimized using the Adam optimizer and trained using an initial learning rate set to 0.0001.

We adopt the cosine annealing strategy[23] to adjust the learning rate which adjusts the learning rate as training progresses. The model is trained for a total of 50000 steps. We train the model with an L1 loss which achieves a better performance than L2 loss in terms of PSNR and SSIM metrics.

EXPERIMENT RESULTS:

We evaluated our model on the Synthetic objective testing set (SOTS).

TABLE II. EXPERIMENTATION RESULTS OF DEHAZING

Model	PSNR	SSIM
Ours	28.52	0.9620
Original	33.57	0.9840

QUALITATIVE RESULTS:

Fig. 4. Image Dehazing Results Left-side - Hazy Images, Right-side - Haze Free Images

C. *IMAGE IN-PAINTING*

The process of reconstructing damaged/missing parts of an image is known as Image In-painting. This can be extended to videos easily. The plethora of use cases has been made possible due to image in-painting.

The goal of in-painting is to provide a visual and semantic consistency. We train a neural network to predict missing parts of an image and in-paint those parts with proper color and features.

NETWORK ARCHITECTURE:

The image in-painting model consists of partial convolution at both encoder and decoder layers for better predictions of missing pixels. Due to an information bottleneck in between, the features from the encoding path are concatenated with the corresponding decoder layer output in the expansive path of the network. This allows the flow of low-level information in the network. This also makes the input and output share the locations of prominent edges in masked and in-painted images. This architecture is called U-Net. The size of an input image and mask decreases from 256x256 to 4x4 in the encoding layer while the number of features extracted are increased in powers of 2 and at the decoding path the input is upscaled using transposed convolutions by size of 2 at each step along with concatenation of outputs from the corresponding encoding layer.

When the image and mask size is 4 it easier to fill the missing portion with a better-predicted value and afterward the size of the image is scaled up to 256 back to give the output in the same size of the input. Figure 3 below shows the basic UNet architecture for 512x512 sized image in-painting, but currently, we have created a similar type of model for 256x256 image due to resource constraints.

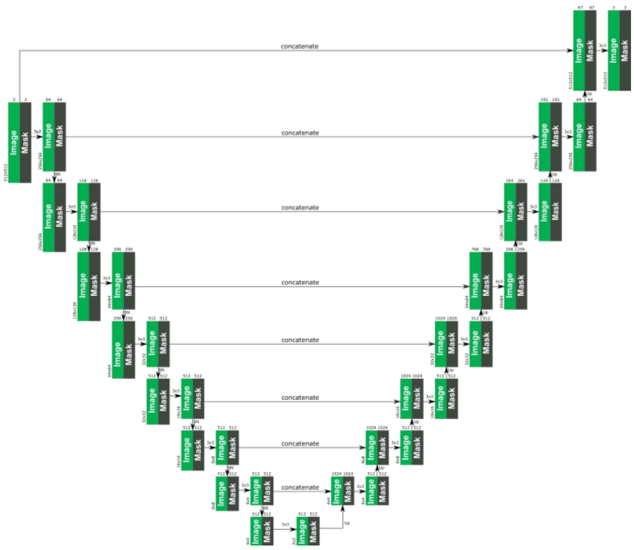

Fig. 5. Image In-painting UNet Architecture

METHODOLOGY:

The dataset used for this project is the training images from the COCO2014 dataset. We randomly sampled over 10000 images from the dataset for our training purpose. The input and output image size is 256x256. We created a random mask on the train and test dataset images of random thickness and at random places inside the image. The model has been trained in two stages. In Stage 1, we train the PConv2D model having batch normalization enabled at both encoder and decoder layers for 200 epochs with learning rate 0.0001 and Adam optimizer. In Stage 2, we disable batch normalization in all encoder layers for 200 epochs with a learning rate of 0.00005.

We start with a learning rate of 0.0001 for stage 1 and 0.00005 for stage 2 as this is best for PConv2D. We trained it for 200 epochs and stopped training when there wasn't any noticeable improvement in the results.

EXPERIMENTATION RESULTS:

For performance measurement, we use the mean absolute error (MAE) and dice coefficient. MAE is computed by taking the mean of the absolute error of the in-painted and actual images on a pixel level. The dice coefficient is measured by the intersection of predicted and ground truth image divided by the sum of the predicted and actual image.

TABLE III. EXPERIMENTATION RESULTS OF IN-PAINTING

Stage	Dice Coefficient	Loss
1	0.6115	0.0350
2	0.6212	0.0211

QUALITATIVE RESULTS

Fig. 6. Image In-painting Results Left-side - Masked Image, Right-side In-painted Image

*D. **IMAGE SUPER RESOLUTION***

Image super-resolution is the process of generating a high-resolution image from a single or multiple low-resolution images without having to incur heavy loss in the translation process. Recently, many deep learning approaches have been proposed aimed at training a model to recover the lost details in new scenes.

NETWORK ARCHITECTURE:

In this, we are using a GAN based model with an objective of increasing a low resolution image to a higher resolution image. The generator network architecture used is the same as the proposed in the SRGAN paper[11]. This architecture focuses the majority of its computation in the lower feature space and applies the up-scaling part later in the network. The basic residual block has been replaced with the novel Residual-in-Residual Dense block (RRDB) architecture as proposed in the ESRGAN paper[12]. The discriminator architecture is almost the same as one mentioned in SRGAN paper, but instead of returning a single value we return output from the convolutional layer as a 16x16 patch size. We use 12 RRDB's for our generator with MRSA[13] initialization for all layers and apply residual scaling to each RRDB's outputs. All convolutional layers of the model are wrapped with spectral normalization[14] which helps us to achieve good results even with a smaller architecture size and helps in training stability. Instead of up-scaling with a simple linear interpolation we are using pixel up-shuffling[15] in the up-sampling layers.

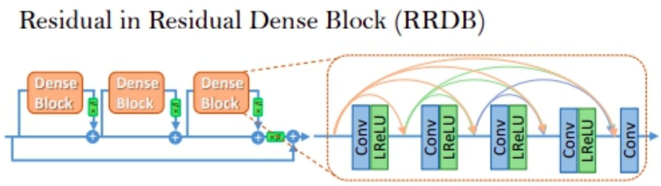

Fig. 7. A Single Residual-in-Residual Dense Block

METHODOLOGY:

For our project, we used the training set of Div2k[16] and a subset of the Flickr2K[17] dataset for training and the validation set of the Div2k for testing purposes. The model has been trained on RGB images and augmentations such as random horizontal flips and 90 degree rotations have been applied to the training set.

The model is being trained on a total of 9000 cropped patches of size 64x64 from 2000 images for a high dimensional output size of 256x256. We train the model in two stages. In Stage 1, we train only the generator with pixel (*L1*) loss function for 100 epochs and then we use this pre-trained generator as an initialization for the generator during the Stage 2 GAN training for about 200 epochs. Pre-training with *L1* loss helps the GAN-based model to obtain more visually pleasing results.

During GAN training we use 3 different loss functions: Perceptual[18] (VGG) loss, RaGan[19] loss and pixel (*L1*) loss. The weightage for the GAN loss and L1 loss are 5e-3 and 1e-2 respectively in the start but are either increased or decreased manually as the training progresses. The model is trained using the Adam optimizer with values of beta1 and beta2 set to 0.9 and 0.999 respectively and a learning rate of 2e-4 for Stage 1 and and decrease every 15k steps by a factor of 10. For Stage 2 we use a learning rate of 0.0001 and 0.0004 for generator and discriminator network, each decayed by a factor of 10 every 30k steps.

EXPERIMENT RESULTS:

We evaluated our trained model on the Set5[24] and Set14[25] bench-marking datasets.

TABLE IV. EXPERIMENTATION RESULTS OF SUPER-RESOLUTION

Metric / Dataset	Set5 (PSNR / SSIM)	Set14 (PSNR / SSIM)
Ours	31.33 / 0.889	27.80 / 0.7806
Original	32.73 / 0.9011	28.99 / 0.7917

QUALITATIVE RESULTS:

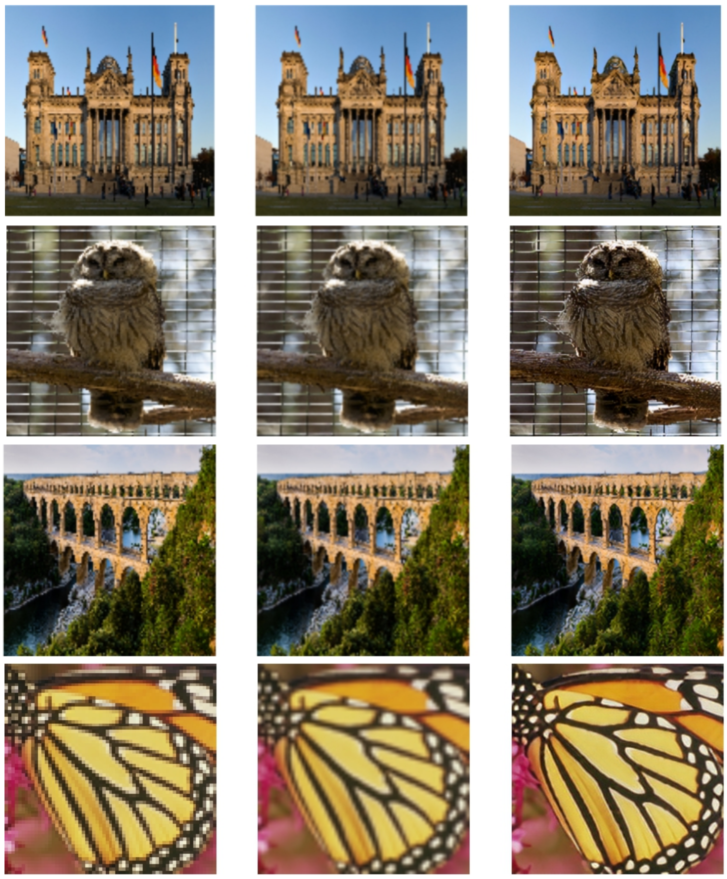

Fig. 8. Image Super-resolution Results Left-side - Low Resolution Images, Center - Bi-cubic Images, Right-side - Model Generated Images

V. SYSTEM ARCHITECTURE

The system is divided into 4 parts named as Image Colorization, Image Dehazing, Image In-painting, Image Super Resolution. All these parts required a specific type of input to generate desired output and one example of each is shown above in the model.

Image Colorization is for converting grayscale images into their colored variation. Image Dehazing is used enhance images affected due to haze or smoke or other weather conditions. Image In-painting will help to remove unwanted parts from the image and reconstruct the dead pixels. Image Super Resolution will help to convert low resolution images into a high resolution image.

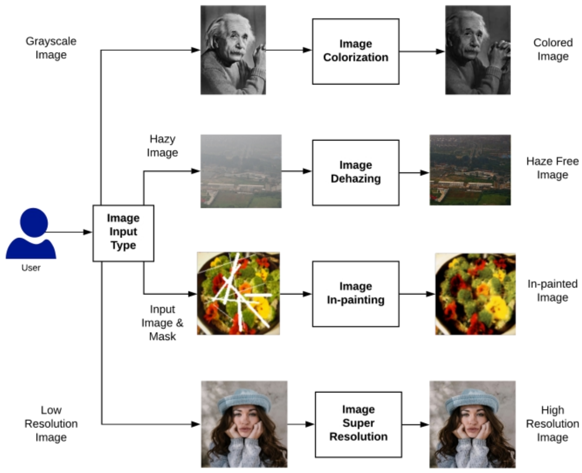

Fig. 9. System Architecture of Image Restoration and Enhancement using Deep Learning

VI. CONCLUSION

Thus we have implemented Image In-Painting, Super-Resolution, Colorization, Dehazing and achieved good results. We will look for more methods which can lead us to best results in less time and more efficiently.

All modules integrated together result in a full fledged image restoration and enhancement system using Deep Learning.

By digitizing old images with the help of image restoration techniques and deep learning is the outcome of our project. Restoring & Enhancing images and making it better is our main objective behind doing this project.

REFERENCES

[1] S. Rani, S. Jindal, B. Kaur, “(PDF) A Brief Review on Image Restoration Techniques.”, IJCA 150(12):30-33, September 2016.

[2] G. Liu, F. A. Reda, K. J. Shih, T. Wang, A. Tao, B. Catanzaro, “Image Inpainting for Irregular Holes Using Partial Convolutions”, ECCV 2018, 20 Apr 2018, arXiv:1804.07723 [cs.CV].

[3] Ian J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, “Generative Adversarial Networks”, 10 Jun 2014, arXiv:1406.2661v1 [stat.ML].

[4] W. Wang, X. Yuan, "(PDF) Recent advances in image dehazing - ResearchGate.", ResearchGate, 23 Dec. 2018.

[5] R. Zhang, P. Isola, A. A. Efros, “Colorful Image Colorization”, ECCV, 2016.

[6] C. Dong, C. C. Loy, K. He, X. Tang, “Image Super-Resolution Using Deep Convolutional Networks”, IEEE, 31 July 2015.

[7] K. Nazeri, E. Ng, M. Ebrahimi, “Image Colorization with Generative Adversarial Networks”, LNCS 10945 (2018), 16 May 2018, arXiv:1803.05400v5 [cs.CV].

[8] O. Ronneberger, P. Fischer, T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”, MICCAI 2015, 18 May 2015, arXiv:1505.04597v1 [cs.CV].

[9] K. He, X. Zhang, S. Ren, J. Sun, “Deep Residual Learning for Image Recognition”, 10 Dec 2015, arXiv:1512.03385v1 [cs.CV].

[10] T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, P. Dollár, “Microsoft COCO: Common Objects in Context”, 21 Feb 2015, arXiv:1405.0312v3 [cs.CV].

[11] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network”, 25 May 2017, arXiv:1609.04802v5 [cs.CV].

[12] X. Wang, Ke Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Yu Qiao, X. Tang, “ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks”, ECCV 2018, 17 Sep 2018, arXiv:1809.00219v2 [cs.CV].

[13] K. He, X. Zhang, S. Ren, J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”, 6 Feb 2015, arXiv:1502.01852v1 .

[14] T. Miyato, T. Kataoka, M. Koyama, Y. Yoshida, “Spectral Normalization for Generative Adversarial Networks”, ICLR 2018, 16 Feb 2018, arXiv:1802.05957v1 [cs.LG].

[15] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, Z. Wang, “Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network”, CVPR 2016, 23 Sep 2016, arXiv:1609.05158v2 [cs.CV].

[16] E. Agustsson, R. Timofte, "NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study," 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 2017, pp. 1122-1131, doi: 10.1109/CVPRW.2017.150.

[17] R. Timofte et al., "NTIRE 2017 Challenge on Single Image Super-Resolution: Methods and Results," 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 2017, pp. 1110-1121, doi: 10.1109/CVPRW.2017.149.

[18] J. Johnson, A. Alahi, Li. Fei-Fei, “Perceptual Losses for Real-Time Style Transfer and Super-Resolution”, 27 Mar 2016, arXiv:1603.08155v1 [cs.CV].

[19] A. Jolicoeur-Martineau, “The relativistic discriminator: a key element missing from standard GAN”, 10 Sep 2018, arXiv:1807.00734v3 [cs.LG].

[20] D. P. Kingma, J. Ba, “Adam: A Method for Stochastic Optimization”, Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015, 30 Jan 2017, arXiv:1412.6980v9 [cs.LG].

[21] X. Qin, Z. Wang, Y. Bai, X. Xie, H. Jia, “FFA-Net: Feature Fusion Attention Network for Single Image Dehazing”, AAAI2020, 5 Dec 2019, arXiv:1911.07559v2 [cs.CV].

[22] B. Li, W. Ren, D. Fu, D. Tao, D. Feng, W. Zeng, Z. Wang, “Benchmarking Single Image Dehazing and Beyond”, IEEE TIP 2019, 22 Apr 2019, arXiv:1712.04143v4 [cs.CV].

[23] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, Mu Li, “Bag of Tricks for Image Classification with Convolutional Neural Networks”, 5 Dec 2018, arXiv:1812.01187v2 [cs.CV].

[24] M. Bevilacqua, A. Roumy, C. Guillemot and M. A. Morel, “Low-Complexity Single-Image Super-Resolution based on Nonnegative Neighbor Embedding”, BMVC 2021, BMVA Press,

[25] Z. R., E. M., P. M., “On Single Image Scale-Up Using Sparse-Representations.”, Curves and Surfaces 2010. Lecture Notes in Computer Science, vol 6920. Springer, Berlin, Heidelberg, 2012.

[26] H. Reza, V. Joze, I. Zharkov, K. Powell, C. Ringler, L. Liang, A. Roulston, M. Lutz, V. Pradeep, "Image Pairs: Realistic Super Resolution Dataset via Beam Splitting", Microsoft, 18 April 2020.

[27] R. Zhang, J.Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, A. A. Efros, "Real-Time User-Guided Image Colorization with Learned ....", SIGGRAPH, 8 May. 2017, arXiv:1705.02999v1.

[28] Zhang Q, Lin J “Exemplar-based image inpainting using color distribution analysis.”, J Inf Sci Eng 28(4):641–654, 2012.

[29] Cai N, Su Z, Lin Z, Wang H, Yang Z, Ling BW-K "Blind inpainting using the fully convolutional neural network.", Vis Comput 33(2):249–261, 2017.

[30] Chen Y, Hu H “An improved method for semantic image inpainting with gans: progressive inpainting.”, Neural Process Lett 49(3):1355–1367, 2019.

[31] Jiao L, Wu H, Wang H, Bie R “Multi-scale semantic image inpainting with residual learning and GAN.”, Neurocomputing 331:199–212, 2019.

[32] Zeng, L., Yan, B., Wang, W., “Contrast Enhancement Method Based on Gray and Its Distance Double-Weighting Histogram Equalization for 3D CT Images of PCBs.”, Math. Probl. Eng. 2016, 1529782, 2016.

[33] Chao D., Chen Change L., Kaiming H. and Xiaoou T., “Image Super-Resolution Using Deep Convolutional Networks”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 2, pp. 295-307, 2016.

[34] Jiwon Kim, Jung Kwon Lee and Kyoung Mu Lee, “Accurate Image Super-Resolution Using Very Deep Convolutional Networks”.

[35] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah and Kyoung Mu Lee, “Enhanced Deep Residual Networks for Single Image Super-Resolution”, 2017.