Exploring methods of image super resolution

Nudrat Nawal Saber
University of Texas at Arlington
nxs3394@mavs.uta.edu

Crupanshu Ashishbhai Udani
University of Texas at Arlington
cau1781@mavs.uta.edu

Abstract

Enhancing image quality is being a popular research topic in the field of computer vision. After the development of deep neural network there was a huge progress in this field. Recent researches focus on using convolutional neural network to enhance the image quality. But to enhance the image quality of an image to a greater scale some researchers are also working with generative adversarial networks. Researches are also being done to improve the performance of the convolutional neural network approach by finding different loss functions, improving the encoders and tuning the optimizers. In this research work we try to understand the underlying process for CNN and GAN based approach and try to find any shortcomings in these approach. We also try to optimize the parameters to achieve improved outputs. We collected data from div2k publicly available dataset as well as some publicly available named as set 5 and set 14 for the test.

1. Introduction

In the world of computer vision researchers always tried to provide ways to enhance the image quality. Zooming into an image and still view clear details was always exciting in sci-fi movies. But is it really possible ? In current days it is possible to some extent. There is an method called Super resolution to zoom and enhance the image. In basic terms, it up-scales and enhances an image using various techniques. For basic up scaling, an image is spread out and the empty pixels filled by replicating from the closet pixels. Nearest neighbor upscaling precisely does this as a result the up-scaled image lacks details and rich features.. Interpolating the image will increase the quality a bit but still it won't be as clear as the original image. From the basic data processing perspective we can say that missing data cannot be recovered with data processing that also says that super resolution is not practically possible. This theory would have stand as it is unless we can manage additional information to fill up the missing gaps. The use of neural network in this field provided additional information from the training set. Meaning larger the data the image should upscale and enhance as envisioned.

There were a number of quality research done in the field of super resolution. But research can be done from the proposed models and improved the outcome. From [1] the authors applied deep convolutional neural network for single image super resolution. It takes low resolution image as inputs and outputs higher resolution image. There are many researches done using CNN for super resolution, many models performs quite better but not all model uses all available features from the input images. Authors in [2] proposed a more complicated model that uses a generative adversarial network (GAN). This method uses a generator to provide high resolution image than uses a discriminator to properly provide upscale images with higher resolution. Enhanced GAN models were also proposed on research [3]. It was widely popularized on gaming community to enhance the vintage games to a higher resolution. These researches opened up a question about which model to use for super resolution on a single image. There are various factors regarding the model selection, memory usage, execution time, performance are some of the examples. In this project we will experiment with both existing super resolution algorithms over single image data. We will see the image enhancement visually as well calculating the loss function and comparing them with the low quality images. We will preprocess an image to its necessity as well as we will downscale some of the samples and determine the outputs of the models with different downscaling factors. We also changed few of the optimizing parameters in the model and tried to enhance the model. In most of the super resolution algorithms using supervised learning the main target is to minimize the loss functions. MSE, PSNR, SSIM are some of the popular ones. In the experiment we also tried to minimized the loss function. In several cases these loss function minimization did not provided with the expected result, we discussed about the matter in the experiment section.

2. Fundamentals

We will discuss about the commonly calculated loss functions in the project. In this section a brief discussion

can be seen over MSE, SSIM and PSNR.

MSE:

MSE stands for mean squared error and it is very common in the field of statistics. It finds the average squared difference between down sampled image and the upscaled image/ It is a loss function that measures the squared error loss. MSE is non-negative and the value closer 0 are considered better. If M and N are the numbers of Rows and Columns in the input images then the MSE equation can be seen as below-

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N}$$

PSNR:

PSNR stands for peak signal to noise ratio. In the case of image it calculates the ratio of signal to noise in decibel between two images. A higher PSNR value indicates that the upscaled image has better quality. We can use the MSE values to calculate the PSNR value. We can see it from the equation where X is the maximum possible pixel value in an image.

$$PSNR = 10log_{10}(\frac{X^2}{MSE})$$

SSIM:

SSIM more elaborately structural similarity index is a important metric determining image quality degradation. It is based on structures visible in the given images. To calculate SSIM we require two similar images but of different quality. In our case we consider the downscaled image as one image and enhanced version as the second image. A practical process for finding SSIM can be understandable from [4]

3. Related works

All Image scaling traditionally was done using various interpolation upsampling methods like nearest-neighbor interpolation, bilinear, and bicubic interpolation, Sinc and Lanczos resampling etc. The nearest-neighbor interpolation is a simple and intuitive algorithm. It selects the value of the nearest pixel for each position to be interpolated regardless of any other pixels. Thus this method is very fast but usually produces blocky results of low quality. The bilinear interpolation [5] first performs linear interpolation on one axis of the image and then performs on the other axis. It shows much better performance than nearest-neighbor interpolation while keeping relatively fast speed. Similarly, the bicubic interpolation (BCI) performs cubic interpolation on each of the two axes. Compared to BLI, the BCI results in smoother results with fewer artifacts but much lower speed.

The interpolation-based upsampling methods improve the image resolution only based on its own image signals, without bringing any more information. Thus they can produce side-effects like noise amplification, blurry results. As a result, researchers moved towards learning based model architecture. Two main types of models architecture widely available for image super-resolution are Convolutional Neural Network(CNN) and Generative adversarial network(GAN). We have looked at some of the interesting works done for CNN. The Super Resolution Convolutional Neural Network[1] is the simplest model and involves bilinear interpolated image as input with few layers of CNN to provide the enhanced image output. While Efficient Sub Pixel Convolutional Neural Network (ESPCNN)[6] takes a bit different approach. The author of the paper proposes a novel CNN architecture where the feature maps are extracted in the LR space. In addition, we introduce an efficient sub-pixel convolution layer which learns an array of upscaling filters to upscale the final LR feature maps into the HR output. By doing so, we effectively replace the handcrafted bilinear or bi-cubic filter in the SR pipeline with more complex upscaling filters specifically trained for each feature map. The author of the paper [7] proposes to leverage denoising auto encoder networks as priors to address image restoration and image SR problems. Which is improved by the authors of this paper[8]. They proposed model containing a chain of convolutional layers and symmetric deconvolutional layers, with Skip connections that are connected symmetrically from convolutional layers to deconvolutional layers.

In recent years, due to the powerful learning ability, the GANs [10] receive more and more attention and are introduced to various vision tasks. To be concrete, the GAN consists of a generator performing generation (e.g., text generation, image transformation), and a discriminator which takes the generated results and instances sampled from the target distribution as input and discriminates whether each input comes from the target distribution. During training, two steps are alternately performed: (a) fix the generator and train the discriminator to better discriminate, (b) fix the discriminator and train the generator to fool the discriminator. Through adequate iterative adversarial training, the resulting generator can produce outputs consistent with the distribution of real data, while the discriminator can't distinguish between the generated data and real data. In terms of super-resolution, it is straightforward to adopt adversarial learning, in which

case we only need to treat the SR model as a generator and define an extra discriminator to judge whether the input image is generated or not.

Based on this concept the first paper that was proposed was SRGAN [2]. SRGAN uses adversarial loss based on cross entropy. While in ProGanSR [11] uses adversarial loss based on least square error. Xu et al. [12] incorporate a multi-class GAN consisting of a generator and multiple class-specific discriminators. And the ESRGAN [13] employs relativistic GAN to predict the probability that real images are relatively more realistic than fake ones, instead of the probability that input images are real or fake, and thus guide recovering more detailed textures.

4. Problem Discussion

We have discussed main theme of the research in the introduction part. Why super resolution is important and what we can achieve with it. The problem was to go deeper in the field of super resolution. Understanding the existing algorithms and trying to optimize it using different parameters and risk functions. Also several loss functions do not provide the perfect result in accordance to the image output. Like higher PSNR does not always provide with higher quality image. So understanding of different loss functions are also important. Super resolution uses a deep neural network or adversarial network in the back most of the time. We tried to understand the internal structure of this network and practically apply the existing algorithms. In the following section we discussed briefly about the methods we used.

5. Proposed Approach

As our main goal was to analyze the two most commonly used super resolution algorithm and trying to optimize the parameters to get better results we approached the model using deep convolutional neural network and generative adversarial training. We will discuss the model, workflow as well as layer structures briefly in this section.

5.1. Super resolution using CNN

For this proposed approach we used a CNN based super resolution model where it uses a deep neural network based algorithm to enhance the image quality. Below we see a flow diagram of how the proposed approach works.

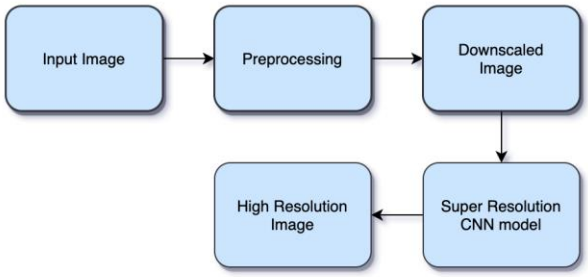

Figure 1: Flow diagram of model using DCNN

The model Learns end-to-end mapping converting the low resolution images to high resolution images. Thus we used it for improving image quality. To calculate the total loss from the actual image and to evaluate the performance of our network. We calculated the PSNR(peak signal to noise ratio),MSE(mean squared error),and SSIM(the structural similarity) between the original images and the low resolution images. While up scaling the image to get high resolution the target was to minimize the loss functions. We can also see a architecture of the proposed approach in the following figure

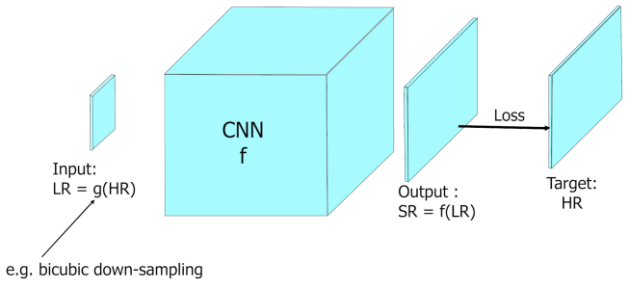

Figure 2: Architecture of the proposed approach

At first we took the images from our dataset. As we had those images from the dataset, we produced the low resolution images from the same images. We found those low resolution versions by resizing the images, both upward and downward. There are different interpolation methods for resizing the images and for our model ,we used bilinear interpolation. To make sure our image quality metrics were calculated without any faults and degraded effectively, we calculated PSNR,MSE,SSIM between our referenced image and degraded image. Then after having our low resolution images and quality metrics being measured, we started building our proposed Convolutional Deep Neural network. Before passing them through the network ,we defined some preprocessing functions. We converted our images to RGB,BGR and YCrCb color spaces. It is necessary because the network was trained on the luminance (Y) channel in the YCrCb color spaces. Thus once we tested our network, we achieved single image

high resolution on all of our input images. Then again we calculated PSNR,MSE and SSIM on the produced images . Now we are considering the inner structure of our neural network. Selecting appropriate filters , activation layers are important for the model execution. We also tried different optimizers and chosen the one that provides best output. We can see the architecture of the CNN in figure 3.-

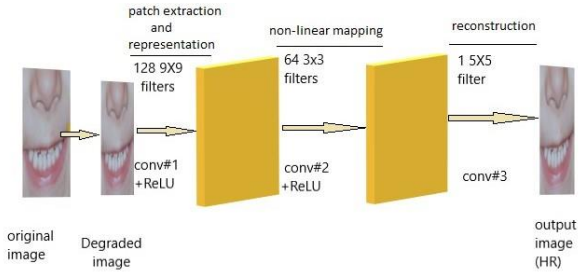

Figure 3: Architecture of the proposed DCNN model

In this cnn, our network is not deep. There are three parts only, that is patch extraction and representation ,non-linear mapping, and reconstruction. The patch extraction is an operation that extracts the overlapping patches from the degraded images let's assume X and illustrates each patch as a vector that has higher dimension. Those vectors contains a set of feature maps, the dimensionality of the vectors are equals to the numbers. In nonlinear mapping the procedure maps the high dimensional vector into another high dimensional vector .These vector also contains another set of feature maps. And the reconstruction function combines the whole high resolution patch wise portrayal to generate the final super resolutional image. It is supposed to be similar to the ground truth. The original picture is degraded to low resolution picture. The low resolution input is first upscale to the desired size using the bilinear interpolation before sending to the network. The first layer of the CNN perform a standard convolution with Relu with 128 filters. After that, a non-linear mapping is performed. After mapping, we needed to reconstruct the image. Thus, we did convolution again. For optimizers we considered SGD, Adam and Adagrad where, Adam optimizer performed better than other available optimizers.

5.2. Super resolution using GAN

For the purpose of this project we are using model based on SRGAN which would upscale image by 4 times. As described in section 3 architecture of any GAN consist of two main parts/models Generator and Discriminator Network. We can see the general architectures below-

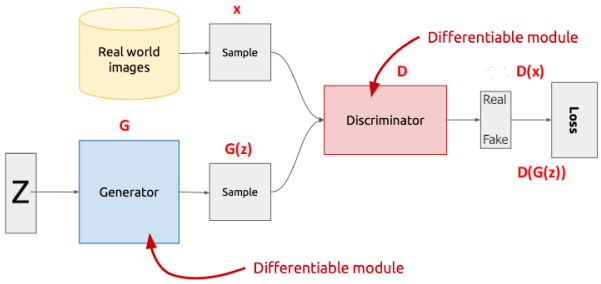

Figure 4: Architecture of the GAN based proposed approach

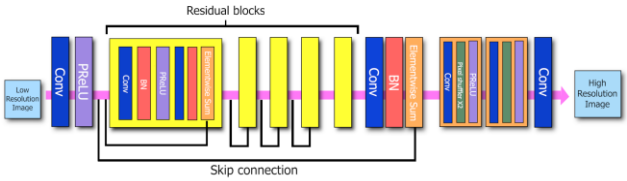

Figure 5: Architecture of the Generator Network (kernel size (k), number of feature maps (n) and stride (s))

For Generator network, LR image of size 64*64 is used as Input. Which is passed to two convolution layers with small 3x3 kernel and 64 feature maps followed by batch normalization and parametric leaky relu as activation function. We then have used 16 residual network blocks as per specification in paper and shown in figure 5 The generator network uses cross entropy loss function. This way the resolution of the input image is increased by 4 times with two trained sub-pixel convolution layers.

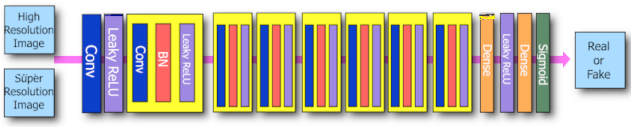

Figure 6: Architecture of the Discriminator Network (kernel size (k), number of feature maps (n) and stride (s))

To discriminate real HR images from generated SR samples we train a discriminator network. It contains eight convolutional layers with an increasing number of 3 x 3 filter kernels, increasing by a factor of 2 from 64 to 512 kernels. Stridded convolutions are also used to reduce the image resolution each time the number of features is doubled. The resulting 512 feature maps are followed by two dense layers and a final sigmoid activation function to obtain a probability for sample classification. Loss function for Discriminator used is MSE (Mean squared error). The entire discriminator architecture is shown in fig (above). For each iteration first the discriminator is trained and then based on the losses of the 2 parts the both the generator and discriminator, the generator model is trained.

6. Experiment

We collected dataset from publicly available resources. We experimented with two types of algorithms for super resolution, CNN based and GAN based. In this section we will discuss about the experimental environment, datasets and outputs briefly.

6.1. Experiment Environment

For the experiment we used a computer with Intel Core-i7 processor, 16gb RAM and 4gb gtx-1060 graphics card. The setup was on a windows 10 environment and executed as an .ipynb notebook.

6.2. Dataset

For The dataset is known as div2k This dataset is publicly available containing 1000 3k resolution [9]. testing we also use benchmark datasets publicly available named as set 5 and set 14. For training we used pre-trained model 3051crop_weight_200.h5 for CNN architecture. With this proposed model we can also take any other RGB or gray scale bmp files to enhance the image quality.

6.3. Results for CNN based approach

In this section we will see some of the degraded images and enhanced version of those images. We applied keras optimizers like SGD, Adamax, Adam optimizers considering all of the loss functions. Adam optimizer provided the best result. We also downscaled the image 2x and 4x times and tried to enhance those degraded image. In some cases we can see that the image quality doesn't provide greater quality although giving higher PSNR values. But considering the SSIM values we can understand the differences. Below some sample images can be seen with the comparison between degraded image and enhanced image. First 2 images are the samples that were downgraded 2 time, and the following 2 images are the samples with 4x downscaling.

Figure 7: Sample enhanced output for 2x downscaled image

Figure 8: Sample enhanced output for 4x downscaled image

We can also see the loss function comparisons in few of the test outputs below

	PSNR	MSE	SSIM
boy.bmp 4x	25.08639	604.7307	0.53186
Enhanced ver.	25.15951	594.6345	0.543671
Butterfly.bmp 4x	20.21670	1855.801	0.716331
Enhanced ver.	20.38473	1785.370	0.741822
boy.bmp 2x	27.444030	351.39739	0.66856
Enhanced ver.	28.339342	285.9350	0.720846
Butterfly.bmp 2x	24.752462	653.0634	0.878841
Enhanced ver.	30.382138	178.6437	0.951972

Table 1: Loss function comparison for sample test output

6.4. Results for GAN based approach

For training a HR images is first rescaled to small image and then trained upon to produce the HR image. The model was trained for 10000 epochs with batch size of 20 images. Based on the trained model, we computed the PSNR (Picture Signal to Noise Ratio), MSE (Mean Squared Error) as well SSIM (Structure Similarity Index Measure) and we were able to achieve average scores of 27.6756, 11.9794,

and 0.8704 respectively. Below we can see a sample ouput of an original image and the generated enhanced image.

Figure 9: Original image before passing it through GAN model

Figure 10: Generated enhanced image from the GAN model

7. Conclusion

To conclude we understood from the conducted project about the mechanism and workflow of super resolution. We did it using 2 algorithms. Although this two are the most commonly used algorithms nowadays researchers are testing with more advance algorithm using better encoders using improved resnet and u-net as decoders, We understood that PSNR won't provide the accurate result as it is described mathematically. Therefore researchers are using different loss function to find the most efficient one. like MAE (mean absolute error). We have seen that CNN model is much quicker than a GAN model but, GAN model provides with higher quality enhancements. In some images we have seeing distortion and ghosting rather than enhancement even though the loss functions were all indicating positive output. Due to the short time frame of this project we couldn't fix the problem for few of the images. Acquiring datasets were not hard as there is many active researches where they use small image data for testing purpose. Understanding how to optimize the training data from a pretrained weight data was an important aspect. We also dug deeper in to the use optimizers , activation layers and image kernel. We tried different preprocessing approaches to tune the input image before running it through the model for a better output. We are hoping to use different loss functions as well as new algorithms or enhanced version of the existing ones in the near future.

References

[1] Dong, Chao, Chen Change Loy, Kaiming He, and Xiaoou Tang. "Image super-resolution using deep convolutional networks." IEEE transactions on pattern analysis and machine intelligence 38, no. 2 (2015): 295-307.

[2] Ledig, Christian, Lucas Theis, Ferenc Husz´ar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, An-drew Aitken 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.

[3] Wang, Xintao, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. "Esrgan: Enhanced super-resolution generative adversarial networks." In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, pp. 0-0. 2018.

[4] Wang, Z., Bovik, A.C., Sheikh, H.R. and Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 13(4), pp.600-612.

[5] Keys, R., 1981. Cubic convolution interpolation for digital image processing. IEEE transactions on acoustics, speech, and signal processing, 29(6), pp.1153-1160.

[6] Shi, W., Caballero, J., Husz´ar, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D. and Wang, Z., 2016.Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1874-1883).

[7] Bigdeli, S.A. and Zwicker, M., 2017. Image restoration using autoencoding priors. arXiv preprint arXiv:1703.09964

[8] Mao, X.J., Shen, C. and Yang, Y.B., 2016. Image restoration using convolutional auto-encoders with sym metric skip connections. arXiv preprint arXiv:1606.08921.

[9] https://data.vision.ee.ethz.ch/cvl/DIV2K/

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in NIPS, 2014.

[11] Y. Wang, F. Perazzi, B. McWilliams, A. Sorkine-Hornung, O. Sorkine-Hornung, and C. Schroers, "A fully progressive approach to single-image super-resolution," in CVPRW, 2018.

[12] X. Xu, D. Sun, J. Pan, Y. Zhang, H. Pfister, and M.-H. Yang, "Learning to super-resolve blurry face and text images," in ICCV, 2017.

[13] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang, "Esrgan: Enhanced super-resolution generative adversarial networks," in ECCV Workshop, 2018.

.