IEC
Institute of Electronics
and Computer

Journal of Artificial Intelligence and Systems, 2024, 6, 1-10
https://iecscience.org/journals/AIS
ISSN Online: 2642-2859

Enhanced Deep Super-Resolution Model Parameter Optimization to Improve the Quality of Geostationary Meteorological Satellite Images

Harimino Andriamalala RAJAONARISOA[1,*], Adolphe Andriamanga RATIARISON[2]

[1]Dynamic Atmosphere, Climate, and Ocean Laboratory, Physics and Applications, Sciences and Technologies, University of Antananarivo, Madagascar
Email: hariminondriamalala@gmail.com
[2]Dynamic Atmosphere, Climate, and Ocean Laboratory, Physics and Applications, Sciences and Technologies, University of Antananarivo, Madagascar
Email: adolphe.ratiarison@univ-antananarivo.mg

How to cite this paper: Harimino Andriamalala RAJAONARISOA, Adolphe Andriamanga RATIARISON (2024). Enhanced Deep Super-Resolution Model Parameter Optimization to Improve the Quality of Geostationary Meteorological Satellite Images. Journal of Artificial Intelligence and Systems, 6, 1–10. https://doi.org/10.33969/AIS.2024060101.

Received: January 17, 2024
Accepted: February 29, 2024
Published: March 4, 2024

Abstract

The aim of this study was to determine the parameters of the EDSR (Enhanced Deep Super-Resolution) model that best improve the quality of images from geostationary meteorological satellites. The datasets used were composed of one hundred pairs of images from geostationary meteorological satellites. Each pair was made up of a low resolution image and a corresponding high resolution image all coming from the same meteorological satellite. The training (respectively the test) of the EDSR model was done with 80% (respectively 20%) of all the datasets. Only the “batch size” and “epoch” parameters of the EDSR model were considered. The other EDSR model parameters were the default parameters used by the EDSR model designer. After training and testing the model using specifically selected datasets, the image generated by the trained EDSR model was compared with the corresponding high-resolution image using the PSNR (Peak Signal-to-Noise Ratio) metric. A high PSNR value indicates a strong resemblance between two images. The “batch size” and “epoch” values corresponding to the best image from the PSNR evolution curve were selected.

Keywords

Artificial intelligence, Batch size, Epoch, EDSR, PSNR, Super-Resolution

1. Introduction

Improving image quality or acquiring a super-resolution began around the 1980s [1] and has had its successes. Its applications then spread to various fields. On the other hand, in recent decades, artificial intelligence has evolved significantly in the field of image processing. Several artificial intelligence algorithms have been developed recently, namely: SRCNN (Super-Resolution Convolutional Neural Network) [2], ESPCN (Efficient Sub-Pixel Convolutional Neural Network) [3], VDSR (Very Deep Super-Resolution Network) [4], SRGAN (Super-Resolution Generative Adversarial Network) [5], EDSR (Enhanced Deep Super-Resolution) [6].

In the field of meteorology, due to the insufficiency of in situ data, we always resort to the exploitation of images from meteorological satellites. In particular, geostationary meteorological satellites offer the opportunity to monitor the same region of the globe. However, the disadvantage of such a satellite lies in the inability to observe small-scale elements because the satellite is placed at an altitude of 36,000km. This raises the question of how to improve the quality of images from geostationary meteorological satellites.

Among the artificial intelligence models popular in recent years, the EDSR model [6] was chosen for this study. The problem was to know which parameters of the EDSR model could best improve the quality of the images from the satellites.

The objective of this study was to optimize the parameters of the EDSR model by training it with specifically selected datasets.

2. Materials and methods

2.1. Datasets

The datasets for training the EDSR model consist of one hundred pairs of image sequences (low and high resolution) from geostationary meteorological satellites downloaded from the EUMETSAT website https://eumetview.eumetsat.int/static-images/MSG/RGB/NATURALCOLOR/FULLRESOLUTION/.

In fact, every hour, a pair of images in two different resolutions (low [7] and high [8] resolution) is generated. The hundred pairs of images corresponded to images taken in the visible channel between April 14, 2023 at 0000UTC and April 17, 2023 at 0400 UTC.

2.2. Methods

The established approach was subdivided into four stages:

- Step 1: Data preprocessing;
- Step 2: Training of the EDSR model;
- Step 3: Testing of the EDSR model;
- Step 4: Search for the "epoch" and "batch size" parameters corresponding to the optimal image quality.

a) Step 1: Data preprocessing

Image data preprocessing was the resizing and separation of images used for training and testing the model. Indeed, the hundred pairs of images (low [7] and high [8] resolution) were downloaded from the EUMETSAT website https://eumetview.eumetsat.int/static-images/MSG/RGB/NATURALCOLOR/FULLRESOLUTION/ with a dimension of 3712*3712.

To facilitate data processing, this dimension was reduced to 128*128 for both low and high resolution images.

As part of the experiment, 80% of the images were used to train the EDSR model and the remaining 20% were used to do the test.

b) Step 2: The training of the EDSR Model

The training of the EDSR model is done in two steps with the 80 pairs of images (low and high resolution):

- Set the batch size parameter of the EDSR model to 1 then vary the number of iterations (epoch) from 1 to 35 in steps of 5;
- Fix the epoch parameter of the EDSR model and vary the batch size parameter from 1 to 25.

1) The principle of the EDSR Model

The EDSR model was trained using the low-resolution and high-resolution image pairs. The model learned to generate super-resolved images by adjusting the weights of its layers to minimize a cost function, which is the mean square loss (MSE). Indeed, while training the EDSR model, the cost function measured the difference between the super-resolved images produced by the model and the real high-resolution images. Minimizing the cost function therefore amounted to minimizing this difference.

Figure 1 summarizes the architecture of the EDSR model. Indeed, the EDSR model is based on deep convolutional neural networks. The EDSR model consists of a stack of residual layers, each including convolution, batch normalization, and activation layers (the activation function used is ReLU), with residual connections between them.

➢ Convolution layers

Convolution layers are responsible for detecting local features in the image, such as edges, textures, and patterns. Each convolution layer applies a set of filters (or kernels) to the image by performing convolution operations.
These filters are small matrices that slide over the image to extract information.

➢ Batch normalization

Normalization is a technique to speed up training and improve model convergence. Batch normalization is performed on each activation channel independently. For a given channel, batch normalization works as follows:

- The mean and variance are calculated over the entire batch;
- The channel activations are normalized by subtracting the mean and dividing by the standard deviation. This centers activations at zero and scales them;
- The normalized activations are then multiplied by γ (scale) and β (translation) learned during training to optimize the performance of the model.

➢ Activation layer

The activation layer introduces non-linearity into the model. The activation function used is ReLU which is defined by:

$$\mathrm{ReLU}(x) = \max(0, x) \qquad (1)$$

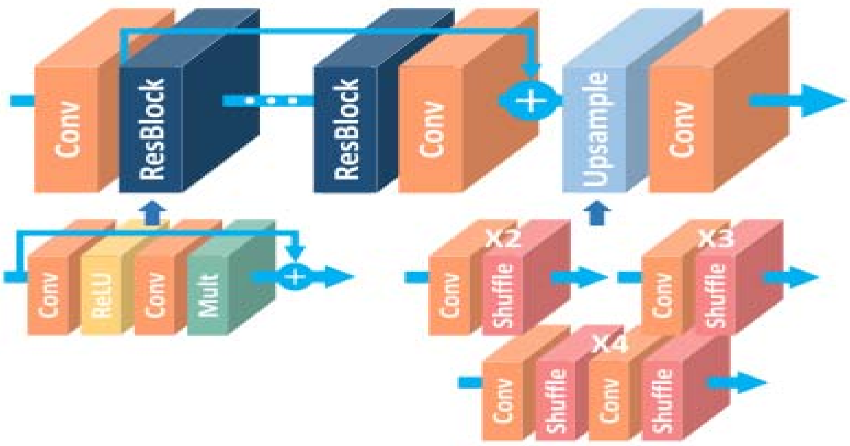

Figure 1. Architecture of the EDSR model [6]

2) EDSR model parameter

The main parameters of the EDSR model are:

- **Number of layers**

  The EDSR model is characterized by a large number of layers, often several hundreds. A high number of layers allows the model to learn complex representations and capture fine patterns [6]. Regarding the modeling in this study, 32 residual layers were used.

- **Filter width**

  The width of the filters determines the size of the features extracted at each layer of the network [6]. In this study, the filter width used was 256, which means that each convolutional layer had 256 output channels.

- **Convolution kernel size**

  Convolution kernels are the filters applied to the inputs of each layer [6]. The kernel size used in this study was 3x3 in all layers of the network.

- **Scale factor**

  The scale factor defines the desired degree of super-resolution [6]. For example, a scale factor of 2 means that the output should have twice the resolution of the input. For this study, the same scale factor as the designer of the EDSR [6] model was used. That is to say the model was evaluated by the scale factor x2 then x3.

- **Activation function**

  An activation function is applied after each convolution layer to introduce non-linearity into the model [6]. ReLU (Rectified Linear Unit) was the function used in this study.

- **Normalization**

  Normalization techniques, such as Batch Normalization, can be used to stabilize and accelerate network training [6]. No batch normalization was used in the modeling in this study.

- **Loss function**

  The loss function measures the deviation between the model's predictions and the true values [6]. For the super-resolution approach in this work, loss functions such as Mean Squared Error (MSE) which measures the root mean square difference between the pixels of the predicted image and the actual image were used.

- **Optimizer**

  The optimizer is responsible for updating the network weights during training [6]. Optimizers such as Adam or SGD are commonly used. For the modeling in this study, the Adam optimizer was used with an initial learning rate of 10^{-4}.

- **Learning rate**

  The learning rate controls the size of the steps taken when updating the network weights [6]. If it is too high, a learning rate can result in rapid convergence, but it can also cause oscillations or jumps around the global minimum.

- **Batch size**

  The batch size is the number of image samples used in a single training iteration of the model. The choice of the batch size has implications on model convergence, memory usage and training time [6]. In this study, the batch size parameter was considered variable.

- **Epoch**

  An epoch corresponds to a complete pass through the entire training dataset. Typically, a model is trained over multiple epochs to ensure that it has sufficiently learned the patterns present in the data [6]. The number of epochs was the second variable parameter in the context of this study.

c) Step 3: Testing of the EDSR Model

As part of this research, the EDSR model that was trained with selected datasets was tested with 20 pairs (low and high resolution) of image sequences not used during training.

To be more objective in testing the model, the resemblance between the high-resolution image (expected output) and the image generated by the model (from a low-resolution image) was evaluated with the PSNR (Peak Signal -to-Noise Ratio) metric. Note that high PSNR values correspond to better simulation.

The following formula allowed to calculate this metric:

$$PSNR = 10\ log_{10}\left(\frac{MAX^2}{MSE}\right) \qquad (2)$$

Where MAX is the probable intensity of the image

And $MSE = \frac{\sum_{i=1}^{n}(y_i - x_i)^2}{n}$ (3)

x_i and y_i denote the intensity values of the high-resolution image and the image predicted by the model, respectively.

d) Step 4: Selection of batch size and epoch parameters of the EDSR model

The selection of the batch size (batch size) and number of iteration (epoch) parameters is based on the evolution curve of the PSNR metric as a function of these parameters taken individually.

The selection was made in two stages:

- Set the batch size value to 1 and vary the epoch from 1 to 35 in steps of 5. Then we select the epoch value which maximizes the PSNR value.
- Fix the epoch value with the PSNR value found previously and vary the batch size parameter from 1 to 20 in steps of 5. The batch size value corresponding to the maximum PSNR was then retained.

Thus, after these two steps, the desired values of the EDSR model parameters which are batch size and epoch were obtained.

3. Results

3.1. Result of the evolution of PSNR according to the epoch for batch size 1

The Figure 2 shows the results of the evolution of the PSNR metric as a function of the epoch which varied from 1 to 35, with the batch size value set to 1. In blue (respectively in orange) we have the curve corresponding to the average PSNR of the images used during training (respectively during the test).

After reading the curve, it was found that the value of the Epoch 35 was selected.

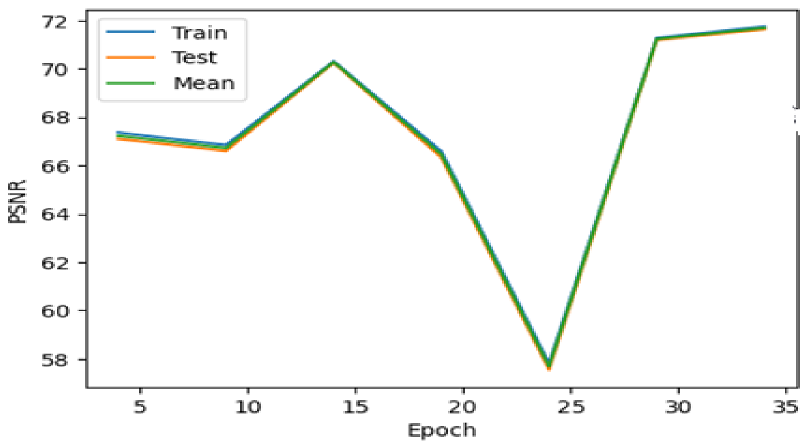

Figure 2. PSNR evolution curve as a function of Epoch to assess the EDSR model

3.2. Result of the evolution of the PSNR as a function of the batch size parameter

The Figure 3 shows the result of the evolution of the PSNR as a function of the batch size parameter with the epoch value fixed at 35. Like in Figure 2, there are three curves corresponding to learning, model testing and the average between the two.

It was observed that the curve decreased and the value of the batch size 1 corresponded to the maximum of PSNR. This value was therefore retained. Thus, in this study, the best modeling with the EDSR model was obtained with the batch Size 1 and epoch 35 parameters.

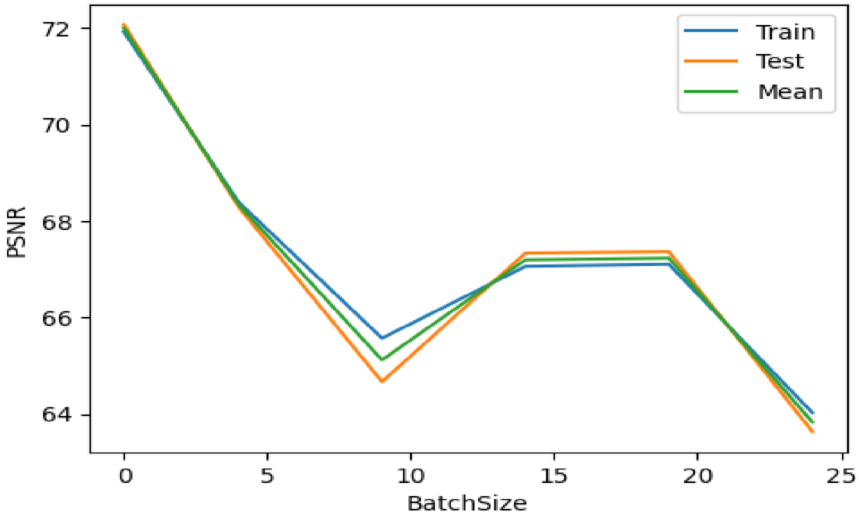

Figure 3. PSNR evolution curve as a function of the batch size parameter for epoch set to 35 to evaluate the EDSR model

3.3. Super-resolution result with EDSR model with the obtained batch size and epoch parameters

The Figure 4 shows an example of an image triplet: corresponding low and high resolution images and the image predicted by the EDSR model after training and testing. The parameters of the EDSR model are 1 for batch size and 35 for epoch. It can be seen in these figures that the EDSR model that was trained was indeed capable of predicting a super-resolved image. This was the best image predicted by the model.

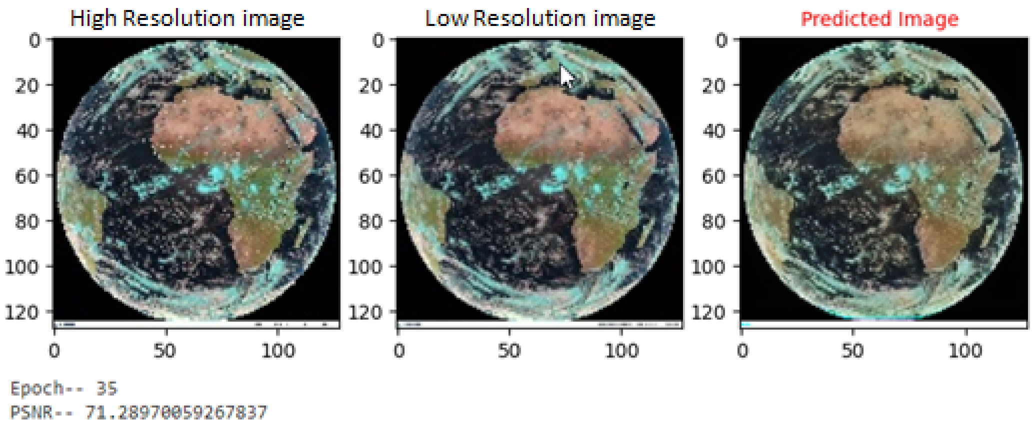

Figure 4. Image triplet: high resolution, low resolution and predicted image using the EDSR model.

4. Discussion

The EDSR [6] algorithm was invented in 2017 by Bee Lim et al. The original EDSR model article does not provide details on the impact of varying EDSR model parameters on image quality. As part of this work, we kept the same parameters of the EDSR model of Bee Lim et al. were kept, except for the batch size and epoch parameters. The work of Bee Lim et al was continued within the framework of this study by evaluating the impact of varying these parameters on the quality of the images produced by the model.

Note that the approach used in the present work to determine the optimal batch size and epoch parameters was similar to our previous article when we experimented with the SRGAN model [9].

The PSNR metric was used to compare the image output from the model and the expected high-resolution image. This was the same metric used in our previous article [9]. Note that other metrics such as MSE [10], SSIM [11] are also common.

5. Conclusion

In this work, the EDSR model was trained with selected datasets consisting of 200 pairs of images from geostationary meteorological satellite to improve the resolution of the images. The work consisted of determining the batch size and epoch parameters of the EDSR model to obtain the best modeling. The experiment showed that the image quality predicted by the trained EDSR model varied depending on the batch size and the number of iteration (epoch) parameters. The established approach made it possible to determine the optimal values of these parameters (1 for batch size and 35 for epoch). With these parameters, the trained model was able to generate a super-resolved image.

References

[1] M. S. Nadler and J. M. Mendel. (1981). *Image sequence enhancement by digital signal processing, Proceedings of the IEEE.*
[2] Dong, C., Loy, C. C., He, K., & Tang, X. (2015). Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, *38*(2), 295-307.
[3] Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., & 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).
[4] Kim, J., Lee, J. K., & Lee, K. M. (2016). Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1646-1654).
[5] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., & Shi, W. (2017). 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).
[6] Lim, B., Son, S., Kim, H., Nah, S., & Mu Lee, K. (2017). 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).
[7] EUMETSAT, Meteosat 0 degree Natural Colour Full disc low resolution. https://eumetview.eumetsat.int/static-images/MSG/RGB/NATURALCOLOR/FULLDISC/
[8] EUMETSAT, Meteosat 0 degree Natural Colour Full disc high resolution https://eumetview.eumetsat.int/static-images/MSG/RGB/NATURALCOLOR/FULLRESOLUTION/
[9]Harimino Andriamalala RAJAONARISOA, S.RAKOTONASY, F. RANDRIAMAHALEO, A. A.RATIARISON (2023). Super Resolution Generative Adversarial Network Model parameter optimization to improve the quality of geostationary meteorological satellite images. *Journal of Artificial Intelligence and Systems, 5*, 115–124. https://doi.org/10.33969/AIS.2023050108.
[10] Alubinskas, G. (2017). Image similarity/distance measures: what is really behind MSE and SSIM?. *International Journal of Image and Data Fusion, 8*(1), 32-53.
[11] Hore, A., & Ziou, D. (2010, August). Image quality metrics: PSNR vs. SSIM. In *2010 20th international conference on pattern recognition* (pp. 2366-2369). *IEEE.*