Two Neural Methods for Astronomical Image Restoration

Vinicius S. Monego[1], Ana Paula A. C. Shiguemori[2], Alice J. Kozakevicius[3], Haroldo F. Campos Velho[1]

[1]*Dept. of Something, National Institute for Space Research (INPE)*
Av. dos Astronautas 1758, 12227-010, São José dos Campos, SP, Brazil
vinicius.monego@inpe.br, haroldo.camposvelho@inpe.br
[2]*Federal Institute of Education, Science and Technology of São Paulo (IFSP)*
R. Antônio Fogaça de Almeida, 200, 12322-030, Jacareí, SP, Brazil
anapaula.acs@ifsp.edu.br
[3]*Laboratório de Análise Numérica e Astrofísica, Federal University of Santa Maria (UFSM)*
Av. Roraima 1000, 97105-900, Santa Maria, RS, Brazil
alice.kozakevicius@gmail.com

Abstract. Image analysis is a key issue in astronomy. One critical procedure is the photometric analysis of astronomical images. Therefore, image restoration is a very important operation for the astronomy community, being a permanent topic under investigation. In this paper, two different methodologies based on artificial neural networks are explored. A convolutional neural network and the multiscale neural filter approaches are evaluated on several astronomical images of different objects (stars, constellations, planets) and compared. The structural similarity index measure is the image comparison metric used to determine the best restoration method.

Keywords: Astronomical images, Image restoration, Neural networks

1 Introduction

Digital image has a central role in many areas of science and technology [1]. Medical tomography, remote sensing, surveillance and monitoring, microscopy analysis, robotics, drone autonomous navigation, industrial quality evaluation, meteorology and oceanography are few examples where images are occupying high relevance. One important issue for image processing is image restoration. Astronomy is fully dependent on the best image quality for carrying out photometric analysis. In a previous investigation, we evaluated four different techniques for astronomical image restoration: truncated singular value decomposition, Tikhonov regularization, multi-scale neural network restoration, and wavelet filtering [2]. We did a second investigation, but this time we analyzed different wavelet filters for image restoration [3].

Image restoration is a good example of an inverse problem, where the ill-posedness feature is a characteristic of this class of problem [4]. In the present paper, a deep learning (DL) supervised neural network is employed as a procedure for image restoration. The restoration is also evaluated by comparing the DL result with the neural multi-scaling restoration [5]. The convolutional neural network is the DL under investigation. Neural networks have been shown to be effective for calculating inverse solutions and this is the main motivation for our research on astronomical images [6]. Better results were obtained by using the deep learning neural network.

The paper is organized as follows. Section 2 describes the neural restoration methods. The image restoration techniques are verified using simulated noisy images, by adding a white Gaussian perturbation to the original image. So, the images for testing, and metrics for evaluation are presented in section 3. Section 4 describes the results. The last section addresses the conclusions.

2 Neural networks

Artificial neural networks (ANN) [7] are inspired to emulate the learning process of the human brain. They were initially proposed in 1943 [8], but the interest has been greatly increased after the publication of the back-

propagation algorithm [9]. They are supervised learning models, and thus require a source (i.e. degraded images) and a target (i.e. the corresponding clean image) to learn the process. The simple perceptron neural network has only a single hidden layer. The output is computed by a non-linear activation function with the argument from the weighted combination of the input values plus a bias term:

$$z(w,x) = f\left[\sum_{j=1}^{N_P} w_j x_j + b\right] \tag{1}$$

where $z(w,x)$ is the neuron output, N_P is the number of connections, x is the input vector, w the weight matrix connecting the artificial neurons – the processing units –, b is the bias, and $f[.]$ is the activation function. A common activation function is the Sigmoid one:

$$f[z] = \frac{1}{1+e^{-z}} \ . \tag{2}$$

Considering a reference set: $\{x_n, t_n\}_{n=1}^{N_t}$, with t_n the target values and N_t the number of set entries for the training phase, the neural network weights and biases are adjusted by minimizing a *loss function*. Regression problems will typically use the mean squared error as the *loss function*, as shown below:

$$E(w) = \frac{1}{2}\sum_{n=1}^{N}[z(w,x_n) - t_n]^2 \ . \tag{3}$$

2.1 Convolutional neural networks

While perceptron networks can be used for image restoration, convolutional neural networks (CNN) [10] have been widely used in many applications, especially involving images [11, 12]. The convolutional layer is able to extract attributes from an image through operations of convolution. A convolutional kernel of a pre-defined size slides through the image, learning *feature maps* of the image while performing an operation of convolution of a window with a sliding convolutional kernel.

There are typical sizes for convolution kernels of 3×3, 5×5, 7×7. Each convolutional layer is defined by a number of kernels. In the convolutional neural networks, the weights are the kernel values and each neuron is represented by the values of each kernel.

CNNs have been employed succesfully in classification [10], segmentation [13], and restoration [14] problems. Generally, neural networks are independent of the degradation form and may remove any type of noise. The Noise2Noise network [15] network illustrates such property, restoring images of additive, multiplicative, and non-Gaussian noises.

The Noise2Noise network [15] is trained to restore degraded sources using degraded targets. The same network can be trained using clean targets and, in this case, the network is referred to as "Noise2Clean". For Gaussian noise, it is observed that clean targets improve the restoration by a large margin [16]. The implementation of [15] uses a RED30 [17] network architecture trained on 50.000 images from the ImageNet [18] validation set. Training was performed with a Tesla P100 GPU and is mentioned to have taken 12 hours.

In this work, we used the SRResNet [19] architecture provided by [16]. This is a network designed for super-resolution, but was retrained for restoration. The architecture contains 35 convolutional layers with residues [20]. The training set contains only 291 images, also sourced from ImageNet. The source is degraded with variable additive white Gaussian noise of standard deviation between $\sigma = 0$ and $\sigma = 50\%$, sampled randomly from a uniform distribution.

2.2 Multiscale neural filter

Another approach, the Multiscale Neural Filter (MSNF) [5] was used for image restoration. This approach involves three phases: a) image information extraction; b) data clusterization to reduce the amount of data to form the training set; and c) training the network to capture a general inverse reconstruction model. The network is a simple one-layer perceptron, where only one image, which is an 8-bit gray level image of co-centered circles, is applied in the training phase. This image is generated in place and submitted to an artificial degradation process. It is first convolved with a low-pass Gaussian filtering operation and then noise is added to it at 1% rate occurrence. The degraded image data is provided as input to the MSNF and the non-degraded image as the corresponding output in the supervised learning process.

The MSNF works by acquiring local spatial information in a multiscale window. This window is a combination of 3x3, 5x5 and 7x7 windows, as seen in Figure 1 (a). The 5x5 and 7x7 windows are subsampled to form two 3x3 windows. The three 3x3 windows are then brought together to form the input vector for training. The target variable is the central point of the window in the corresponding clean image – see Figure 1 (b). The training image, also shown in Figure 1(a), is a gray level image of co-centered circles degraded with a Gaussian noise and filter, both of 10% standard deviation. Before being introduced to the network, the data has to go through a preprocessing step that is the window linearization, and this process is shown in Figure 1 (b). To reduce the amount of data to feed the network, a data clusterization scheme based on Kohonen neural network was also applied to the dataset.

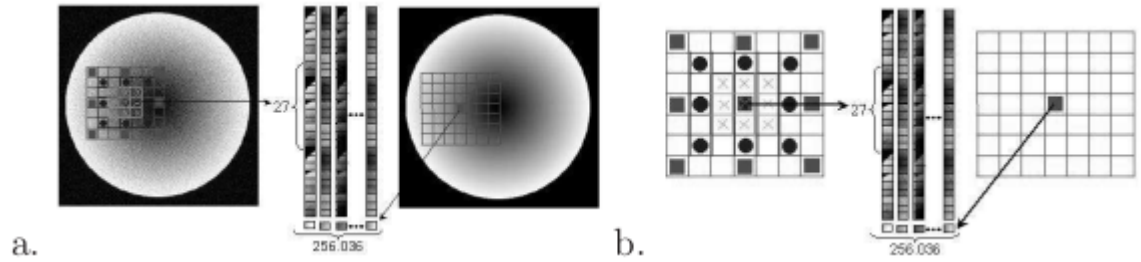

Figure 1. (a) Source image and the multiscale window (left) and the clean image and target pixel (right), (b) Linearization process.

The architecture of the MSNF is very simple and it doesn't require external data. That is, the model is able to generate its own training set. The architecture is based on a single layer with 28 neurons and sigmoid activation functions on the hidden and output layer. However, this approach has the consequence of losing three rows or columns in each edge, therefore reducing its dimension by (6,6).

3 Methodology

We used two images from the HubbleSite [21] website, which releases Hubble space telescope images under a license similar to public domain. The figures represent the NGC 3147 galaxy and the Jupiter planet.

The images were resized to a standard size of 512x512 pixels and, if necessary, cropped to a square shape. Then, additive gaussian noise of standard deviation 5%, 15% and 25% was artifically introduced in the images. This noise is represented by eq. (4), where I_{ij} is an image ($i = 512$, $j = 512$), σ is the standard deviation (5%, 15% or 25%) and $\nu_{i,j}$ is a random variable of Gaussian distribution, as seen in eq. (5), of mean $\mu = 0$ and the levels of standard deviation, producing 3 different degraded images for each test image.

$$I_{ij}^{\sigma} = I_{ij}(1 + \sigma\, \nu_{ij}) \tag{4}$$

$$P(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(z-\mu)^2/(2\sigma^2)} \tag{5}$$

All the images have color channels and are represented as RGB. The noise is introduced separately on each channel, and as such the restoration is also executed in the individual channels.

After generating the degraded images, the restoration process is started. For assessing the restoration quality, we chose the Structural Similarity Index Measure (SSIM) [22] metric. The SSIM compares a combination of luminance, contrast and structure between two images. The luminance, contrast and structure are defined in Equations eq. (6), eq. (7) and eq. (8), respectively, where x and y are each image being compared.

$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}, \tag{6}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}, \tag{7}$$

$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma y + c_3}, \tag{8}$$

Then, the similarity is defined on terms of these variables as in eq. (9).

$$SSIM(x, y) = l(x, y) \times c(x, y) \times s(x, y). \tag{9}$$

The result is in the range [0,1] and tells how much one image is similar to the other, with zero being total dissimilarity, and one total similarity. The SSIM is often evaluated as an average over sliding windows in the image, the window size being a parameter. That is, x and y are windows, and the entire image is a special case of the largest possible window.

3.1 Software environment

The convolutional neural network used in the experiments considers the pre-trained weights that can be downloaded from [16]. The restoration is performed with the TensorFlow [23] framework on version 2.2. The code for the multiscale neural filter is developed on MATLAB, but was executed on GNU Octave version 6.2.0 with the Image package on version 2.12.0. The code and network weights are provided by the authors based on the previous contribution at [24]. For the evaluation metric, the structural similarity routine is provided by the *scikit-image* package, version 0.17.2, using the default values. The default sliding window size is 7.

4 Results

Results for the two networks applied to the NGC 3147 and Jupiter images are shown in this Section. The architecture employed for the networks is presented in Table 1. A visual comparison for both is provided to assess restoration quality. The figures only show the case for $\sigma = 15\%$. Furthermore, a table containing the evaluation metrics is also provided – see Table 2.

Table 1. Neural network architectures for astronomical image restoration.

Hyperparameter	MSNF	SRResNet
Input, Hidden, Output neurons	27, 28, 1	3x3, 3x3, 3x3
Activation function	Sigmoid	PReLU
Loss function	MSE	MSE
Learning rate (η)	Adaptive	Adaptive
Momentum (α)	0.7	Adaptive
Max. training epochs	500	60

Figure 2 shows the restoration for the NGC 3147 image. The restoration using the CNN is very good. It does, however, provide a loss of detail on the star speckles around the galaxy disk, as that is very similar to noise. That problem also occurred in the MSNF, which smoothed the image a little.

Figure 3 displays the restoration for the Jupiter image. In this case, it is clear that the restoration of the edges of the planet is a problem for both models. The CNN adds a paint-like effect, while the MSNF makes it grainy. Both methods could restore the plain dark background well, which is a difficult task for other methods [3].

Finally, the evaluation metrics are shown in Table 2. In both images, the CNN could restore with over 90% similarity in the $\sigma = 5\%$ and $\sigma = 15\%$ cases. It does keep a reasonable quality even in the $\sigma = 25\%$ case, where similarity could be improved by 70%. At that level, however, performance started to drop in the Jupiter image. The MSNF maintained a consistent similarity over all three restorations, but performed worse for low noise level.

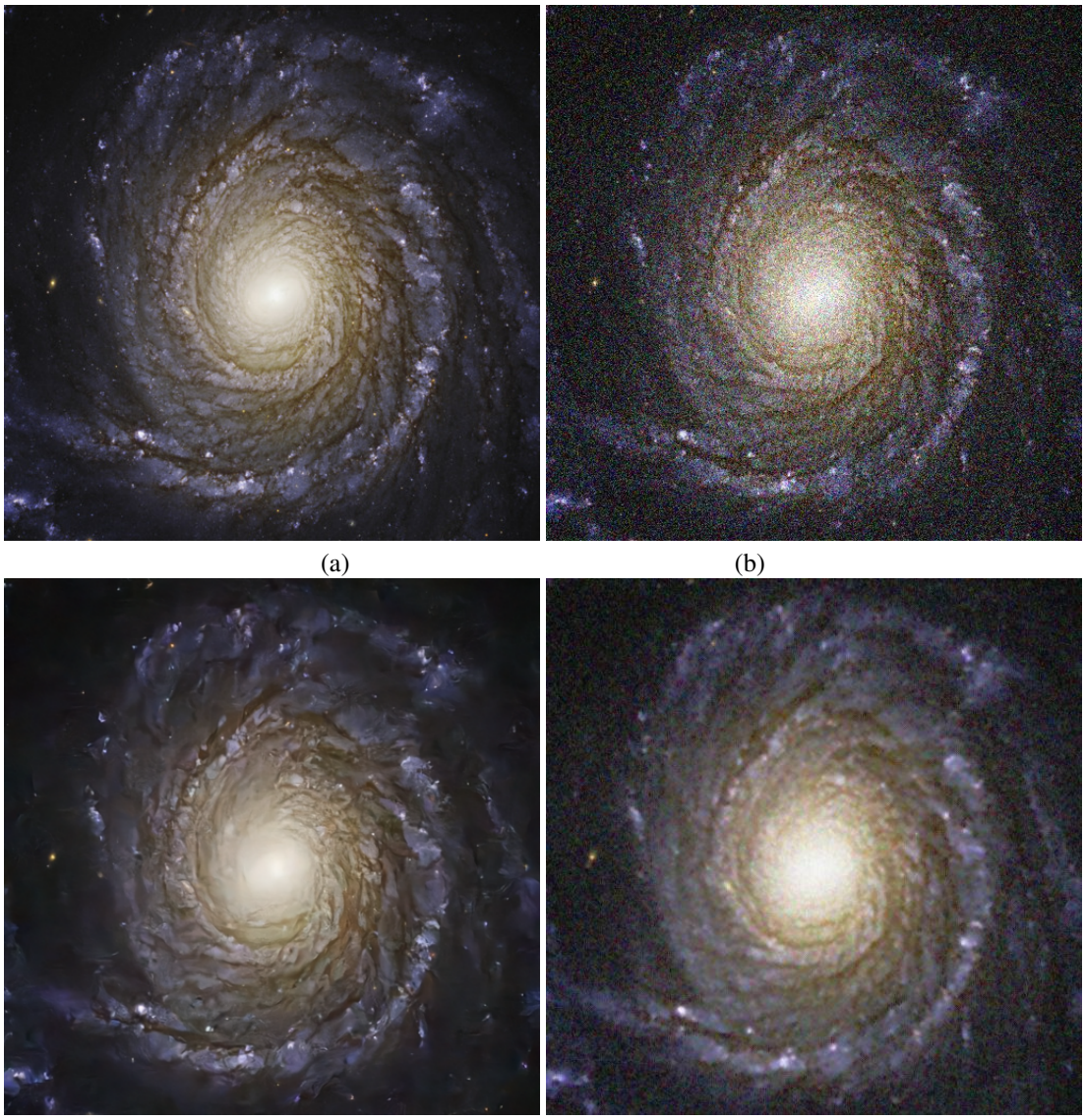

Figure 2. Restoration of the "NGC 3147" image. (a) Original image. (b) Noisy image (15% deviation). (c) Denoised by CNN. (d) Denoised by MSNF.

Table 2. Structural similarity results for each restoration. G = Gain, or how much the similarity improved.

Method	Image	$\sigma = 5\%$	$\sigma = 15\%$	$\sigma = 25\%$	G ($\sigma = 5\%$)	G ($\sigma = 15\%$)	G ($\sigma = 25\%$)
Degraded	NGC 3147	0.7	0.26	0.13	0.0	0.0	0.0
Degraded	Jupiter	0.61	0.15	0.06	0.0	0.0	0.0
SRResNet	NGC 3147	0.96	0.9	0.85	+0.26	+0.64	+0.72
SRResNet	Jupiter	0.97	0.91	0.76	+0.26	+0.74	+0.7
MSNF	NGC 3147	0.55	0.55	0.55	-0.15	+0.29	+0.42
MSNF	Jupiter	0.44	0.41	0.36	-0.17	+0.26	+0.3

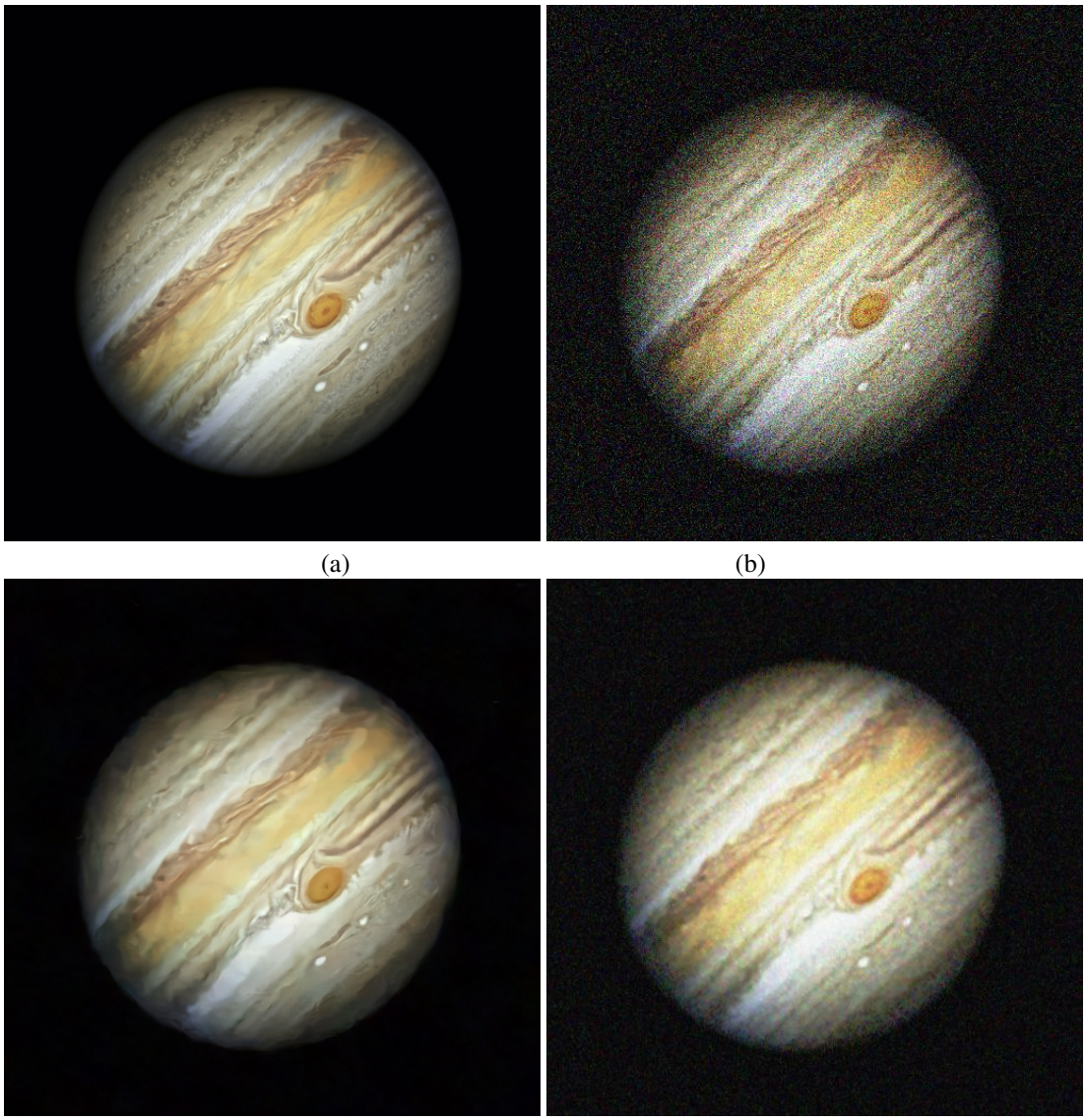

Figure 3. Restoration of the "Jupiter" image. (a) Original image. (b) Noisy image (15% deviation). (c) Denoised by CNN. (d) Denoised by MSNF.

5 Conclusions

Convolutional neural networks are widely used as image restoration technique. Indeed, they have a very high restoration quality. The restoration is extremely amazing for small (5%) noise levels – see Table 2. However, it comes at the higher cost of training time and huge amount of data availability.

In contrast, the MSNF is able to generate its own training dataset and has a much lower training time, at the cost of a lower quality restoration. It is also more resistant to noise variation, and is able to maintain a consistent quality for different noise levels.

Therefore, depending on the degree of accuracy for the image restoration process and/or the amount of data for the training phase, this paper describes two neural approaches for the astronomical image restoration techniques as good options for the astronomy community.

Acknowledgements. The authors would like to thank the Brazilian agencies for research support. Author HFCV thanks to the National Council for Scientific and Technological Development (CNPq, Portuguese) for the research grant (CNPq: 312924/2017-8).

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] R. C. Gonzalez and R. E. Woods. *Processamento de imagens digitais*. Blucher, 2010.
[2] A. P. Shiguemori, M. S. Dantas, E. H. Shiguemori, A. J. Kozakevicius, V. S. Monego, R. S. R. Ruiz, C. Strieder, and H. F. Campos Velho. Methods for astronominal image restoration. In *Proceedings...* Iberian Latin-American Congress on Computational Methods in Engineering, 2017.
[3] V. S. Monego, A. J. Kozakevicius, and H. F. Velho. Campos˙ Restoration of astronomical images by wavelet techniques. In *Proceedings...* Iberian Latin-American Congress on Computational Methods in Engineering, 2020.
[4] M. Bertero and P. Boccacci. *Introduction to inverse problems in imaging*. CRC Press, 1998.
[5] A. P. A. Castro, I. N. Drummond, and J. D. S. Silva. A multiscale neural network for image restoration. *TEMA*, vol. 9, n. 1, pp. 41–50, 2008.
[6] E. H. Shiguemori, H. F. Campos Velho, J. D. Silva, and J. C. Carvalho. Neural network based models in the inversion of temperature vertical profiles from radiation data. *Inverse Problems in Science and Engineering*, vol. 14, n. 5, pp. 543–556, 2006.
[7] S. Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1999.
[8] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, vol. 5, n. 4, pp. 115–133, 1943.
[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, vol. 323, n. 6088, pp. 533–536, 1986.
[10] Y. LECUN, L. BOTTOU, Y. BENGIO, and P. HAFFNER. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, vol. 86, n. 11, pp. 2278–2324, 1998.
[11] B. Sun, T. Zhang, J. Su, and H. Sha. Gnetdet: object detection optimized on a 224mw cnn accelerator chip at the speed of 106fps, 2021.
[12] E. Poder. Cnn-based search model underestimates attention guidance by simple visual features, 2021.
[13] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, eds, *Proceedings...*, pp. 234–241, Cham. Springer, 2015.
[14] S. Ye, Y. Long, and I. Y. Chun. Momentum-net for low-dose ct image reconstruction, 2020.
[15] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila. Noise2noise: learning image restoration without clean data, 2018.
[16] Y. Uchida. An unofficial and partial keras implementation of 'noise2noise: learning image restoration without clean data', 2018.
[17] X.-J. Mao, C. Shen, and Y.-B. Yang. Image restoration using convolutional auto-encoders with symmetric skip connections, 2016.
[18] IMAGENET. Image database organized according to the wordnet hierarchy, 2020.
[19] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network, 2017.
[20] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.
[21] Hubblesite, 2020.
[22] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, vol. 13, n. 4, pp. 600–612, 2004.
[23] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org, 2015.
[24] A. P. A. Castro. *Restauração de imagens por operadores modelados por redes neurais artificiais*. Doutorado em computação aplicada, Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2009.