Reso-Net: Generic Image Resolution Enhancement Using Convolutional Autoencoders

Koustav Dutta[1] and Priya Gupta[2(✉)]

[1] SA&MA: Analytics and Cognitive, Deloitte Touche Tohmatsu India LLP, Bengaluru, India
kousdutta@deloitte.com
[2] Atal Bihari Vajpayee School of Management and Entrepreneurship, Jawaharlal Nehru University, New Delhi, India
priyagupta@mail.jnu.ac.in

Abstract. Images are created in a variety of ways in various industries. These images are tough to work with, and as a result, they can't be used effectively in a variety of fields. In this paper, Image Resolution is improved to carry out the process of generic image enhancement tasks. In this process, the low-resolution image is enhanced so that the high-resolution image is achieved. With the help of Image enhancement, the perception or in other words the process of interpreting information present in images by the human viewers is enhanced and the quality is improved to a large extent. Image resolution augmentation has traditionally been accomplished using a variety of classic image processing approaches. However, these methods are not as robust as they should be in dealing with any form of noise signal associated with the image and unable to handle the problems of Error Control Mechanism, Optimization and some other problems. Therefore, this paper presents a method of image resolution enhancement using Advanced Hybrid Neural Network architecture which brings about significant improvements in the entire process.

Keywords: Autoencoder · Convolutional Neural Network · Image Resolution Enhancement · Hybrid Model · Decoder · Encoder · Up-Sampling · Reconstruction Error

1 Introduction

In this paper, the procedure of image resolution enhancement is carried out to obtain high resolution images from low resolution ones for utilization in various sectors. The term "resolution" has been used to describe a crucial aspect of an image. The images are being processed to improve the resolution. This paper proposes a Self-Supervised Pattern Recognition Algorithm to strengthen the system's ability to handle noisy images without any labelled or supervised learning involvement, hence challenging the results of traditional image processing-based techniques. Thus, Stacked Convolutional-Autoencoder Algorithm is proposed which contributes to the general image resolution improvement of any sort of image including any type of noisy data. (any value of Peak Signal to Noise Ratio) and at any specific circumstances.

S. Nandan Mohanty et al. (Eds.): ICISML 2022, LNICST 470, pp. 298–308, 2023.
https://doi.org/10.1007/978-3-031-35078-8_25

2 Literature Review

For the process of Image Enhancement of Image Resolution, various conventional Image Processing approaches [1] have endured and been utilized for a long time. According to the domain in which they are used, image resolution enhancement techniques can be divided into two categories: 1) Image-Domain; and 2) Transform-Domain. In contrast to transform-domain techniques, which use transformations to accomplish Image Resolution Enhancement, image-domain techniques rely on geometric and statistical data directly extracted from the original image [2]. Nearest neighbour, quadratic, linear, and cubic interpolation functions are among the many traditional Image Resolution Enhancement methods. However, these methods have issues such as edge blurring, ringing around edges, and texture loss [3]. This is since they do not employ any edge-related information in the original image [4]. The decimated discrete wavelet transforms (DWT) has been commonly employed for performing image resolution enhancement [5, 16]. There are a variety of ways that can be used to enhance an image without ruining it. There are two categories of enhancement techniques: 1) Spatial Domain Methods and 2) Frequency Domain Methods.

In spatial domain methods, we work directly with an image's pixels. The pixels' values are used to make the necessary enhancements [17]. In frequency domain techniques, the image is initially converted into the frequency domain. The Fourier Transform of an image is computed first in order to do this. The image's Fourier transform is used to execute enhancement operations, and the resultant image is obtained inverse Fourier transform is performed. Many attributes, such as image brightness, contrast, and grey level distribution, can be changed using these enhancement methods. Because of this, the generated image's pixel value is altered by the transformation functions applied to the input values [6, 7]. The evaluation of an image enhancement technique's performance is problematic because subjective judgement is frequently employed in practice. However, there are various approaches for objective evaluation as well. The "Resolution enhancement in MRI" technology improved the image by a factor of three to a factor of seventeen. However, there were issues with this technique's error control system and optimization [8]. Mean and Variance Adjustment was the second approach we discussed. This algorithm produced better results in memory, blur-free images, and terms of speed, but the main issue with this technique was determining the proper exposure time [9]. The Road Image Enhancement Technique was the third technique. This technique improved resolution and removed occlusions, and it was successful in updating a large road image, but it had certain issues, such as significant ego-motion, barriers that did not move, and failure in some circumstances [10]. Unmixing-Based Fusion Approach was the final technique. When compared to other fusion techniques, this procedure produced the best outcomes. It enhanced spatial resolution while causing very little spectral distortion and is simple to apply [11].

3 Proposed Algorithm and Architecture

This research presents an effective technique for image denoising. A robust Self-Supervised Pattern Recognition Algorithm containing of Stacked Convolutional Neural Network Layers along with Autoencoder Networks has been the pillar of this Hybrid

Model. Any kind of noisy image can be effectively handled by the hybrid architecture under any use cases and without any problem faced by the conventional methods. The architecture consists of Convolutional Neural Network (Pooling Layers, Dense Layers, Convolution Layers etc.) and Stacked Autoencoder Network (Decoder & Encoder Networks). The proposed method's overall structure is shown in the block diagram in Fig. 1.

There are three main components to the suggested algorithm such as I. Convolutional Neural Network (CNN) II. Autoencoder III. Hybrid Model (CNN - AE): which combine to create a hybrid model. The following sections of the paper address the model's implementation and architecture development.

- **Convolutional Neural Network** (CNN) is used to design the whole backbone of the system. Convolutional Neural Networks (CNNs) are deep learning techniques for computer vision that can identify and categorize features in images. CNN is composed of several kinds of layers:
- **Fully connected input layer**— A single vector created by "flattening" the outputs of earlier levels can be utilized as an input for the subsequent layer.
- **Convolutional layer**— A feature map is created by applying a filter that scans the entire image, a few pixels at a time, to estimate the class probabilities for each feature.
- **Pooling layer–** By gradually reducing the spatial dimension of the representation, it seeks to minimize the number of parameters and calculations in the network. The pooling layer handles each feature map individually. The most prevalent pooling technique is max pooling. [12]. Two further pooling techniques are minimum pooling and average pooling.
- **Flatten Layer**- Flattening is the process of transforming data into a one-dimensional array for use in the subsequent layer. We can produce a single, lengthy feature vector by flattening the output of the convolutional layers. It's also connected to the final classification model, also known as a fully connected layer.
- **Fully connected layer**— applies weights to the input obtained by the feature analysis to anticipate an accurate label.
- **Fully connected output layer–** produces the final probabilities for determining the image's class [13].

Autoencoders are combined with a Convolutional Neural Network to construct a Hybrid Deep Learning (Self-Supervised) Architecture. An autoencoder [14] is a sort of unsupervised artificial neural network [15] that helps people learn effective data coding. An autoencoder trains the network to ignore signal "noise" and other random elements that may affect the image to discover an encoding (representation) for a set of data, generally for dimensionality reduction. In addition to learning the reduction side, the autoencoder also learns the reconstructing side, which is how it learns to create a representation from the reduced encoding that is as similar to the original input as is practical. The **Encoder** and **Decoder** are the two components of an autoencoder [15], which can be defined as transitions ϕ and ψ, such that:

$$\phi \ : \ \boldsymbol{\chi} \rightarrow \boldsymbol{F}$$

$$\psi \ : \ \boldsymbol{F} \rightarrow \chi$$

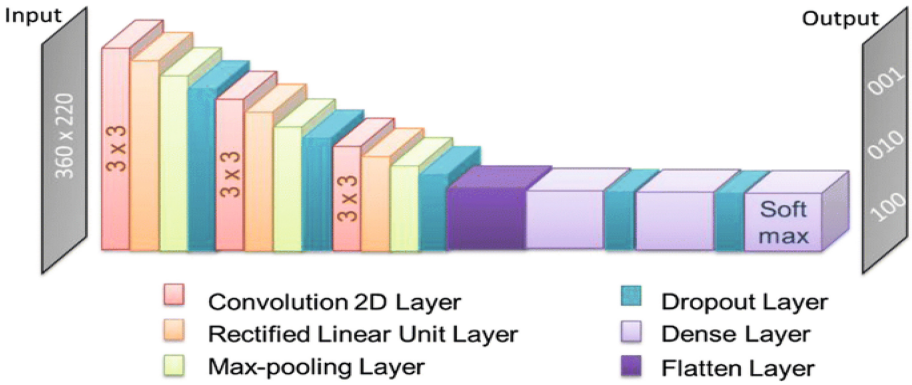


Fig. 1. Convolutional Neural Network Architecture

$$\phi, \psi = arg\ min||X - (\psi\ o\ \phi)X||^2$$

In the simplest scenario, provided one hidden layer, an autoencoder's encoder stage receives input.

$$\boldsymbol{x} \in R^4 = \chi \text{ and maps it to } \boldsymbol{h} \in R^p = F$$

$$\boldsymbol{h} = \sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})$$

This image h is described by the term latent variables, *code*, or *latent representation*. Here, σ is an element-wise activation function such as a rectified linear unit or a sigmoid function. $\boldsymbol{b}$ is a bias vector and $\boldsymbol{W}$ is a weight matrix? Typically, biases and weights are initialized arbitrarily and modified using iterative Backpropagation throughout training. [12]. After that, the Decoder Stage of the autoencoder maps $\boldsymbol{h}$ to the reconstruction $\boldsymbol{x}'$ of the same shape as $\boldsymbol{x}$:

$$\boldsymbol{x}' = \sigma'\left(\boldsymbol{W}'\boldsymbol{h} + \boldsymbol{b}'\right)$$

where $\sigma', \boldsymbol{W}', \textit{ and } \boldsymbol{b}'$ for the Decoder might not be connected to the corresponding $\sigma, \boldsymbol{W}, \textit{ and } \boldsymbol{b}$ for the Encoder.

Reconstruction errors, often known as the "loss," are minimized by autoencoders through training.

$$\boldsymbol{L}(\boldsymbol{x}, \boldsymbol{x}') = \left\|\boldsymbol{x} - \boldsymbol{x}'\right\|^2 = \left\|\boldsymbol{x} - \sigma'(\boldsymbol{W}'(\sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})) + \boldsymbol{b})\right\|^2,$$

where $\boldsymbol{x}$ is generally averaged over some input training set.

Through the use of backpropagation of the error, an autoencoder is trained, just like a regular feedforward neural network (Fig. 2).

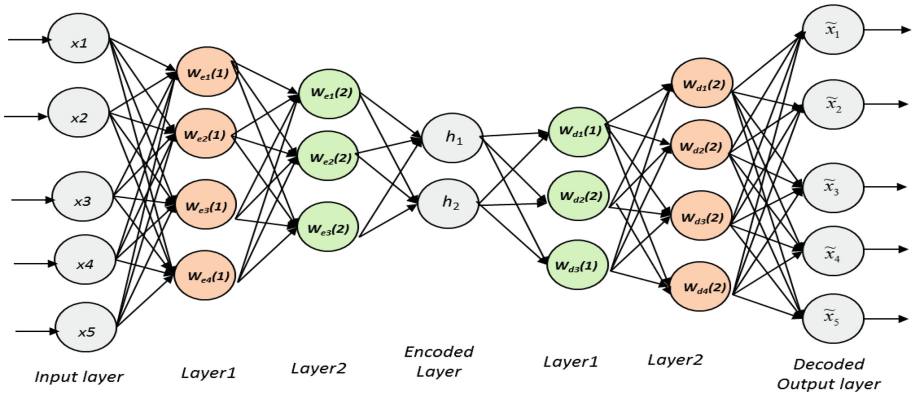


Fig. 2. Autoencoder Architecture

4 Proposed Deep Learning Hybrid Model

The Convolutional Neural Network and the Autoencoder Architecture of the Hybrid Model (Fig. 3) combine to provide the required output [15]. Since Convolutional Neural Networks are effective at handling images and extracting features and fine details from images, CNNs are used to obtain low resolution image data as input layer data and to efficiently extract features from the image data using Pooling Layers, Convolution Layers (present in the hidden layers), and other layers. Additionally, Autoencoders are utilized in combination with CNN Architecture to encode the features in a latent space vector using the encoding component, and then to reconstruct the image using the decoding part to produce an improved high-resolution image.

5 Detailed Architecture Design

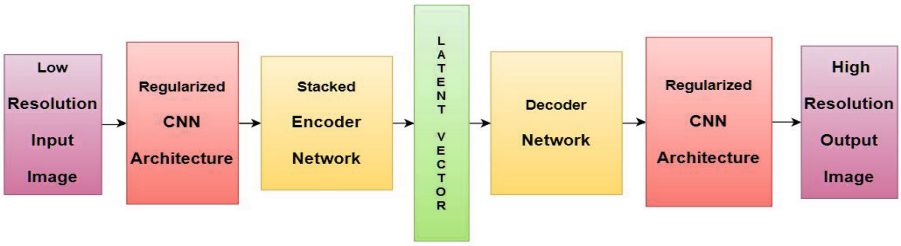


Fig. 3. Convolutional-Autoencoder Architecture

The following is the fundamental concept behind utilizing autoencoders for image denoising:

Encoder The autoencoder will figure out how original photos are augmented with noise as part of its learning process. We now have a function F(X) = Y, where Y is the noisy image and X is the original, clear image, describing how noise is formed.

Decoder A portion of the autoencoder will attempt to reverse the image noise. Now that Y is present in the equation F(X) = Y, we are aiming to create the input X from which we will obtain the output.

Depending on the different approaches, we might obtain the same low-resolution image from many input images. This results in some loss in the process, which we want to reduce and provide the ideal input image. These are the several architectural layers:

1. Low Resolution Images of dimension: (80 × 80 × 3) containing various types of noise (In this instance, Salt & Pepper Noise is used as an example) are fed into the Neural Network (Fig. 4).
2. Convolution Layers with 64 Hidden Neurons make up the first and second layers of the encoder part. In order to extract the small details and characteristics from the input image, a (3 × 3) filter is applied to both layers in this instance. To protect the data in the following layers, zero padding is likewise employed in these two layers. Additionally, the L1 Regularization approach, commonly known as Lasso Regularization, is incorporated into each of the convolution layers to help the model learn all of the hyperparameters in a precise and well-defined manner, help it avoid overfitting, and provide Low Variance and Low Bias. [18] After each of the two layers, ReLU (Rectified Linear Unit) is utilized as the activation function[$ReLU : max(0, x)$], it aids in removing the image's non-linear features.
3. The third layer of the structure is a pooling layer. In this case, the Max Pooling layer is applied to perform spatial down-sampling. The process of creating feature maps, which in turn create the network's encoder, involves extracting the high-intensity fine details. Zero Padding is once more utilized in this scenario to maintain the image's details.
4. An additional Convolution Layer with 128 Hidden Neurons is present in the fourth and fifth layers as well. In this instance, additional features from the input image, such as numerous contours, edges, texture, corners, forms, etc., are extracted using a (3 × 3) filter. To maintain the spatial information in the following layers, Zero Padding is once more applied in this layer. Additionally, the L1 Regularization Technique is incorporated into the two layers to further reduce overfitting issues. After this layer, the ReLU (Rectified Linear Unit) Activation function is once more employed to extract the non-linear characteristics from the image [19].
5. Next, Max Pooling layer is used in the sixth layer to perform further spatial down-sampling and zero padding is again used along with this layer. And thus, in the final layer of the encoder part, the Convolution Layer, each consisting of 256 Hidden Neurons, is present. In this instance, more detailed final fine-tuned features are extracted from the input of the preceding layer using a (3 x 3) filter. This layer also makes use of zero padding. This layer also contains an incorporated L1 regularization algorithm. Finally, the crucial and higher-level feature characteristics and representations of an image are encoded in a Latent Vector by using the ReLU (Rectified Linear Unit) Activation Function, which aids in extracting the non-linear features from the image [20].

6. The process of image reconstruction now requires the decoding procedure to be completed. The Decoding part thus helps in the process of Image Resolution Enhancement. Up sampling of the encoded latent vector is done at first by an order of 2 before passing into the further layers of the decoder network.
7. A Convolution Layer [8] with 128 neurons makes up the first and second layers of the Decoder Part. The Encoded Image feature map is accepted as input into the first layer of the Decoder section. To glean the fine details and features from the encoded feature image feature map, a (3×3) filter is applied to both layers. To protect the data in the following layers, zero padding is also employed in this layer. Lasso Regularization is also used and embedded with this layer. The Activation Function used after this layer is ReLU (Rectified Linear Unit) it aids in removing the image's non-linearly encoded units.
8. Next, the first two layers of the decoder network are added to extract finer details from the previous layer's output. The output image obtained from the first layer of the Decoder portion is further processed through up-sampling. Up-Sampling is essentially a straightforward scaling up of the image using nearest neighbor or bilinear up sampling; in this network, it is done by an order of 2. This procedure smoothens the image.
9. Convolution layers are once more added to the 64-neuron network in the next two layers. The convolution operation aids in extracting the image's fine details. A (3×3) kernel is utilized in the procedure as a filter in both the layers along with the Zero Padding layer. Each of the convolution layers has the L1 regularization approach, also known as Lasso Regularization, which enables the model to learn all of the hyperparameters in a precise and well-defined manner, solve the issue of Overfitting, and provide Low Variance and Low Bias to the model. The ReLU Activation Function is used to extract complicated and fine-grained information from the image. ReLU is employed frequently in the procedure since it doesn't experience the Vanishing Gradient Problem and because the neurons are selectively activated and deactivated to obtain comprehensive information from the picture feature map from the preceding layers sequentially (Fig. 5).
10. Next, addition of the last two convolution layers is done and further in the process, on the output image from the previous layer, up-sampling is used.
11. Finally, a Convolution Layer using a ReLU (Rectified Linear Unit) is added as the final layer of this Convolutional-Autoencoder Network [$ReLU : max(0, x)$] To keep the features intact, the method also uses a Zero-Padding function and an activation function. Finally, the Enhanced High-Resolution Image is obtained as output in this final layer.
12. To determine the Reconstruction Error or Loss (Mean Squared Error Loss), the High-Resolution image is obtained as the final layer's output and the input image is contrasted with this output [15]. Consequently, Backpropagation is carried out with the aid of the Adaptive Delta Optimizer in order to lessen the loss. [21] (to reach the Global Minima) in order to update each and every component of the network's filters. Thus, as the process continues, the loss gets smaller until the Enhanced

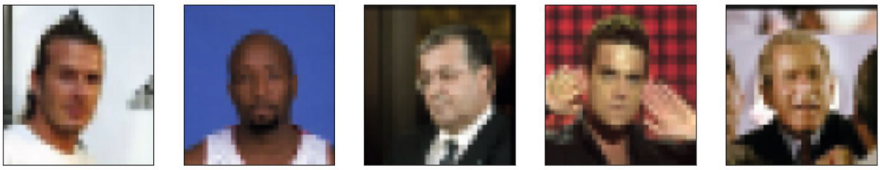
High-Resolution Image is obtained (Fig. 6). The Reconstruction Error (Loss) is given by:

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i.log(p(y_i)) + (1 - y_i).log(1 - p(y_i))$$

6 Results

1. Input Low Resolution Image:

a.



b.

Fig. 4. Low Resolution Input Image

2. Encoder and Decoder Hidden Spatial Vectors:

Fig. 5. Hidden Encoding & Decoding feature vectors

3. Result: Enhanced High-Resolution Image (Reconstructed Output):

a.

b.

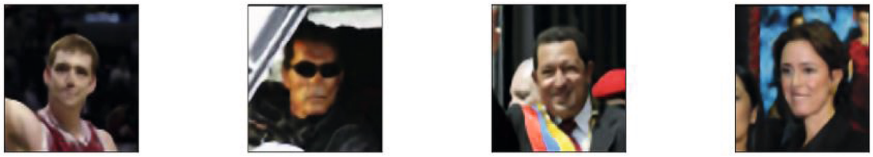


Fig. 6. Improved Resolution Image

7 Conclusion

This study opens up a whole new range of applications for self-supervised learning techniques. Traditional and native image processing methods have long struggled to consistently and effectively handle the process of image enhancement. At the forefront, tasks such as Image Resolution Enhancement. Conventional methods needed a lengthy time to process the images, which resulted in a less-than-ideal output, and they were unable to handle many different types of poor resolution images affected by multiple noise causes.

Because of this, the disciplines of Image Resolution Enhancement have undergone a huge and revolutionary change as a result of the development and implementation of efficient and reliable Hybrid Deep Learning Models like Stacked Convolutional Autoencoders, resulting in results that are quite noticeable from low-resolution images. Furthermore, the model can handle any sort of image noise, and just 0.011 percent reconstruction error (loss) was observed, commonly known as PSNR (Peak Signal to Noise) percent. The PSNR % can be decreased even further by incorporating more layers into the model and adding more non-linearity to the layers by adding more neurons. Furthermore, employing integrated hybrid models of Recurrent Neural Networks and Variational Autoencoders (RNN-VAE), the concept and Hybrid model applications can be expanded and employed for monitoring and detecting of abnormalities in real-time video images or frames, Because the RNN element of the hybrid model could reliably handle sequential data, and Variational Autoencoders could analyze and track anomalies, the hybrid model was a good fit, which work more reliably by reconstructing frames utilizing the parameters and variables of the sequential data's Probability Distribution Function.

References

1. Weeks, A.: Fundamentals of Electronic Image Processing. SPIE PRESS, Bellingham (2006)
2. Chanda, B., Majumder, D.D.: Digital Image Processing and Analysis (2002)

3. Simant, D., Li, H.: An adaptive algorithm for image resolution enhancement **2**, 1731–1734 (2000). https://doi.org/10.1109/ACSSC.2000.911284
4. Yinji, P., Il-hong, S., Hyunwook, P.: Image resolution enhancement using inter-subband correlation in wavelet domain. In: International Conference on Image Processing (ICIP 2007), vol. 1, pp. I–445 (2007). https://doi.org/10.1109/ICIP.2007.4378987
5. Chang, S., Cvetkovic, Z., Vetterli, M.: Locally adaptive wavelet-based image interpolation. IEEE Trans. Image Process. Public. IEEE Signal Process. Soc. **15**, 1471–1485 (2006). https://doi.org/10.1109/TIP.2006.871162
6. Rajput, S., Suralkar, S.R.: Comparative study of image enhancement techniques. Int. J. Comput. Sci. Mobile Comput. **2**(1), 11–21 (2013). https://www.ijcsmc.com/docs/papers/january2013/V2I1201303.pdf
7. Maini, R., Aggarwal, H.: A comprehensive review of image enhancement techniques. J. Comput. **2**(3), Issn 2151–9617 (2010). https://arxiv.org/ftp/arxiv/papers/1003/1003.4053.pdf
8. Carmi, E., Liu, S., Alon, N., Fiat, A., Fiat, D.: Resolution enhancement in MRI. Magn. Reson. Imaging **24**(2), 133–154 (2006). https://doi.org/10.1016/j.mri.2005.09.011
9. Babu, K.R., Sunitha, K.V.N.: a new approach to enhance images of mobile phones with inbuilt digital cameras using mean and variance. In: 2010 International Conference on Advances in Computer Engineering, pp. 232–234 (2010). https://doi.org/10.1109/ACE.2010.57
10. Noda, M., et al.: Road image update using in-vehicle camera images and aerial image. In: 2011 IEEE Intelligent Vehicles Symposium (IV), pp. 460–465 (2011). https://doi.org/10.1109/IVS.2011.5940470
11. Bendoumi, M.A., He, M., Mei, S.: Hyperspectral image resolution enhancement using high-resolution multispectral image based on spectral unmixing. IEEE Trans. Geosci. Remote Sens. **52**(10), 6574–6583 (2014). https://doi.org/10.1109/TGRS.2014.2298056
12. Lenka, R., Dutta, K., Khandual, A., Nayak, S.R.: Bio-Medical image processing: medical image analysis for malaria with deep learning. In: Nayak, S.R., Mishra, J. (ed.), Examining Fractal Image Processing and Analysis, pp. 158–169. IGI Global (2020). https://doi.org/10.4018/978-1-7998-0066-8.ch007
13. Lawrence, S., Giles, C.L., Tsoi, A.C., Back, A.D.: Face recognition: a convolutional neural-network approach. IEEE Trans. Neural Networks **8**(1), 98–113 (1997). https://doi.org/10.1109/72.554195
14. Masci, J., Meier, U., Cireşan, D., Schmidhuber, J.: Stacked convolutional auto-encoders for hierarchical feature extraction. In: Honkela, T., Duch, W., Girolami, M., Kaski, S. (eds.) ICANN 2011. LNCS, vol. 6791, pp. 52–59. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21735-7_7
15. Baldi, P., Lu, Z.: Complex-valued autoencoders. Neural Networks Official J. Int. Neural Network Soc. **33**, 136–147 (2012). https://doi.org/10.1016/j.neunet.2012.04.011
16. Celik, T., Tjahjadi, T.: Image resolution enhancement using dual-tree complex wavelet transform. IEEE Geosci. Remote Sens. Lett. **7**(3), 554–557 (2010). https://doi.org/10.1109/LGRS.2010.2041324
17. Lenka, R., Khandual, A., Dutta, K., Nayak, S.R.: image enhancement: application of dehazing and color correction for enhancement of nighttime low illumination image. In: Nayak, S.R., Mishra, J. (eds.) Examining Fractal Image Processing and Analysis:, pp. 211–223. IGI Global (2020). https://doi.org/10.4018/978-1-7998-0066-8.ch011
18. Stephen, R., Tibshirani, R., Friedman, J.: A study of error variance estimation in lasso regreSION. Statistica Sinica **26**(1), 35–67 (2016). www.jstor.org/stable/24721190. Accessed 16 July 2020
19. Hara, K., Saito, D., Shouno, H.: Analysis of function of rectified linear unit used in deep learning. In: 2015 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. Killarney (2015). https://doi.org/10.1109/IJCNN.2015.7280578

20. Zhang, Z., Song, Y., Qi, H.: Age progression/regression by conditional adversarial autoencoder. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4352–4360 (2017). https://doi.org/10.1109/CVPR.2017.463
21. Kingma, D.P., Ba, J.L.: Adam: a method for stochastic optimization. In: International Conference on Learning Representations, pp. 1–13 (2015). https://arxiv.org/pdf/1412.6980.pdf