Article:	**Convolutional Auto Encoder For Image Denoising**
Author(s):	Abdul Ghafar[1], Usman Sattar[2]
Affiliation:	[1]Department of Information Systems, Dr Hassan Murad School of Management, University of Management and Technology, Lahore, Pakistan [2]Department of Management Sciences, Beacon house National University, Lahore, Pakistan
Article QR:	Abdul Ghafar
Citation:	G. Abdul, and S. Usman, "Convolutional auto encoder for image denoising," *UMT Artificial Intelligence Review,* vol. 1, pp. 1–11, 2021. https://doi.org/10.32350/UMT-AIR.0102.01
Copyright Information:	This article is open access and is distributed under the terms of Creative Commons Attribution 4.0 International License

A publication of the
Dr Hassan Murad School of Management,
University of Management and Technology, Lahore, Pakistan

Convolutional Autoencoder for Image Denoising

Abdul Ghafar[1]*, Usman Sattar[2]

[1]Department of Information Systems, Dr Hassan Murad School of Management, University of Management and Technology, Lahore, Pakistan
*Corresponding Author: abdul.ghafar@umt.edu.pk
[2]Department of Management Sciences, Beacon house National University, Lahore, Pakistan

ABSTRACT: Image denoising is a process used to remove noise from the image to create a sharp and clear image. It is mainly used in medical imaging, where due to the malfunctioning of machines or due to the precautions taken to protect patients from radiation, medical imaging machines create a lot of noise in the final image. Several techniques can be used in order to avoid such distortions in the image before their final printing. Autoencoders are the most notable software used to denoise images before their final printing. These software are not intelligent so the resultant image is not of good quality. In this paper, we introduced a modified autoencoder having a deep convolutional neural network. It creates better quality images as compared to traditional autoencoders. After training with a test dataset on the tensor board, the modified autoencoder is tested on a different dataset having various shapes. The results were satisfactory but not desirable due to several reasons. Nevertheless, our proposed system still performed better than traditional autoencoders.

I. INTRODUCTION

Image denoising is a common problem in computer vision, which has been looked at very carefully. This technique employs partial differential equations (PDEs) and other types of changes such as non-local techniques, and a family of models that uses sparse coding techniques and represented in the following expression:

$$z = x + y$$

Where the combination the original image (x), (z) as noise while other factors are (y).

Thanks to recent advancements in deep learning, the results obtained from using these techniques have been effective. Autoencoders are used to clean the images. They are very good at removing noise, as well as not very specific about how noise is made or how the distortion was created. Denoising autoencoders built with convolutional layers can effectively denoise digital images because they can take advantage of strong spatial correlations [1].

Medical imaging, such as X-rays, magnetic resonance imaging (MRI), computed tomography (CT), ultrasound, and other types of imaging, is affected by noise

when they are employed. When the level of radiation goes down, the noise level goes up. Many times, people and machines need to denoise images to be able to properly analyze them [1]. In this study, we attempted to create a simple version of a denoising autoencoder built using a convolutional layer network by using a short dataset.

II. RELATED WORK

Even though block-matching and 3D filtering (BM3D) is well-designed and is thought to be the best way to denoise images, a simple multilayer perceptron (MLP) can do the same work [2]. Denoising autoencoders are a new addition to the literature on image denoising. This design of autoencoders could be used to build deep networks [3] . When you feed the result of one denoising autoencoder to the one below it, you can make a very large network.

This research designed a way to clean up the images using convolutional neural networks. If you use a small number of training images, you can get better results. This is because wavelets and Markov random fields can be used to denoise images. Stacked sparse autoencoders (SSAE) were utilized for picture denoising and inpainting. It accomplished the same thing as K-singular value decomposition (K-SVD). The researchers used stacked sparse autoencoders to build adaptive multi-column deep neural networks for picture denoising. This system was found to be very durable for denoising different types of noise [4].

Segmentation is a critical step in Digital Rock Physics (DRP) as the original images are available in a gray-scale format. Conventional methods often use thresholding to delineate distinct phases and, consequently, watershed algorithm to identify the existing phases [5].

In this paper, autoencoders based deep learning model is proposed for image denoising. The autoencoders learn noise from the training images and then try to eliminate the noise for a novel image. The experimental outcomes prove that this proposed model for PSNR has achieved higher results compared to the conventional models [6].

This research proposed a DCAE to solve the high-resolution SAR image classification. The experimental results demonstrate that the appropriate parameter settings of the network can achieve impressive classification performance[7].

Purpose of this research paper is to develop a novel PSO algorithm (namely, PSOAO) to automatically discover the optimal architecture of the FCAE for image classification problems without manual intervention [8].

In this paper, the researcher proposed an energy compaction-based image compression architecture using a CAE. Firstly, we presented the CAE architecture and discussed the impact of different network struction on the coding performance [9].

This research paper focused on developing a semiadversarial network for imparting soft-biometric privacy to face images [10].

III. AUTOCODER

Modern autoencoders are networks that attempt to encode and then decode a piece of input into a low-dimensional latent space. To make the input more realistic, they employ concepts from deep learning and Bayesian inference.

As demonstrated in the diagram below, a traditional image autoencoder takes an image, encodes it into a latent vector space, and then decodes it into an output with the same dimensions as the original picture. This is how a traditional image autoencoder functions. The autoencoder is then trained using the same pictures as the input images in the following stage. In this way, the autoencoder learns to replicate the original inputs. It's possible to make the code (the encoder's output) interesting, causing the autoencoder to learn more interesting hidden data representations. Generally, the code needs to be largely zeros if you want it to be low-dimensional and sparse. In this scenario, the encoder reduces the size of the input data by compressing it into a lower number of bits.
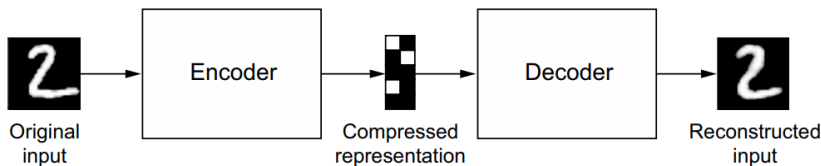

Fig.1. Autoencoder Diagram [11]

Classical autoencoders do not produce usable or well-organized latent spaces in real life. They aren't particularly good when it

comes to compression. For this reason, a large number of individuals dislike them. Conversely, VAEs provide autoencoder statistical wizardry, allowing them to learn a continuous, well-structured latent space. They're proven to be effective when it comes to creating photographs[11].
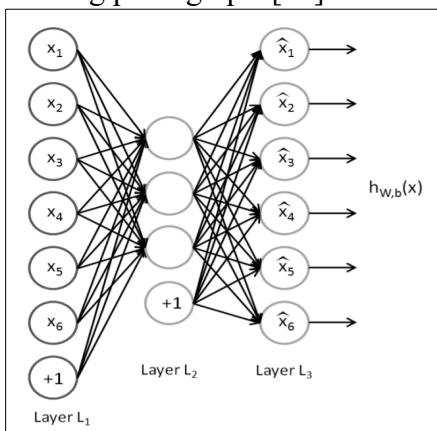

Fig. 2. Basic architecture of an Autoencoder [1]

The process of autoencoding begins with layer L1, which is taken as an input layer. It is encoded with a latent representation in layer L2. The input is then recreated at L3 level. The autoencoder is forced to choose a compact approximation because there are fewer hidden units than inputs. Typically, an autoencoder detects low-dimensional representations, such as Principal Component Analysis (PCA). However, if certain sparsity constraints are followed, hidden units that are bigger than the number of input variables can provide meaningful insights.

A. Denoising Autoencoders

When we apply a denoising autoencoder, we train the model to learn how to rebuild the input from a noisy version. Some of the inputs are randomly made zero using a process known as stochastic corruption. This causes the denoising autoencoder to anticipate missing (corrupted) values for subsets of missing patterns picked at random. The diagram below explains how to construct a denoising autoencoder.
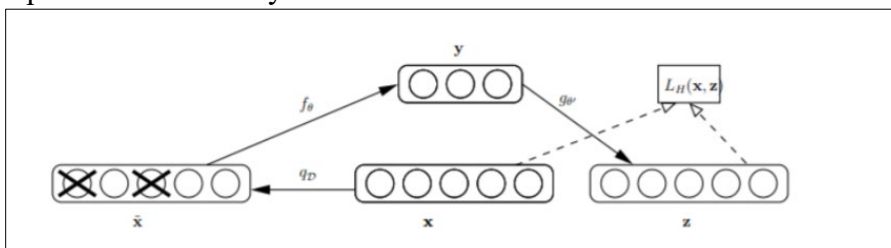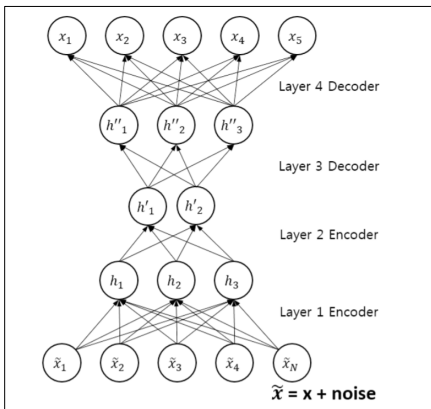

Fig.3.Basic Architecture of a Denoising Autoencoder [1]

It's possible to stack denoising autoencoders to make a very large network using a stacked denoising autoencoder method (SDAE).

The current layer receives the output from the layer below, after which the training is done layer by layer.

Fig.4. A Stacked Denoising Autoencoder [1]

B. Convolutional Autoencoder

Convolutional autoencoders are based on standard autoencoder architecture, except instead of having the usual autoencoder design, it uses convolutional encoding and decoding layers. Furthermore, this type of encoder is better at image processing than traditional autoencoders because they employ deep neural networks at full capacity to examine the structure of the images they're processing. Weighs are shared among all the input places in convolutional autoencoders to stabilize the local spatiality of each input. The representation of feature map is given as

$$h^i = s(x * W^i + b^i).$$

In the equation, the bias is spread across the whole map, for two dimensional convolution (2D), and s stands for activation. A single bias is used for each latent map. Whereas the reconstruction is done through following expressin

$$y = s(\sum_{i \in H} h^i * \tilde{W}^i + c)$$

For each input channel, c is the amount of bias that is applied. H is the group of latent feature maps that are turned over. Backpropagation is used to figure out the gradient of the error function as a function of the parameters[1].

IV. DATA SET

We created our own dataset for experimentation. In order to create a dataset, we placed different objects on a canvas and then recorded a video.
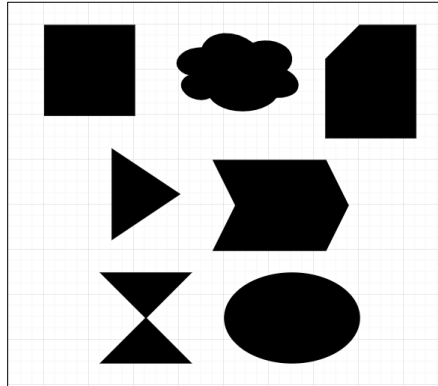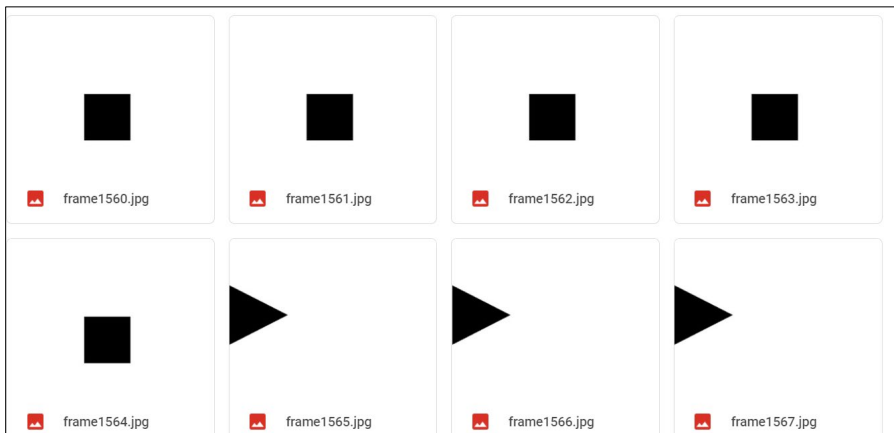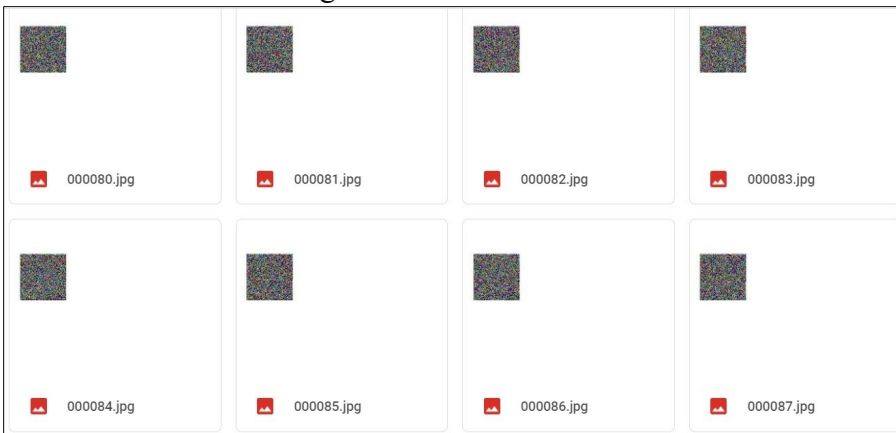
Fig. 5.

Fig. 6. Extracted Dataset

Fig. 7. After Adding Noise to Dataset

Subsequently, we wrote a custom function to add noise to every image. This created a noisy image as shown above.

V. EXPERIMENTAL SETUP

All of the images were edited before they were modelled. Pre-processing involved resizing of all images to 200 x 200 so that they take up less space on the computer.

Fig.8. Model Summary

VI. MODEL ARCHITECTURE

We used Keras library to create a Conv autoencoder. First, we define the input layer, then we define 2 conv layers followed by max pooling layer for download sampling. Subsequently, we applied 2 conv2d transpose layers for upsampling.

VII. EXPERIMENT RESULTS AND DISCUSSION

We use the tensor board to monitor the experimental results. It also monitors the training loss and the validation loss. The model did not perform up to mark as shown in figure 9.
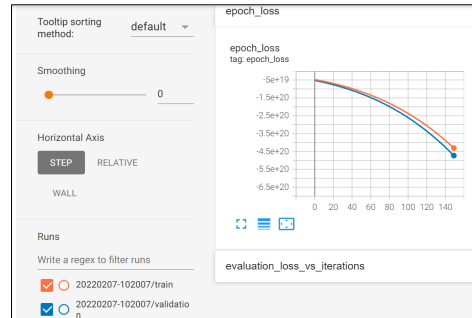

Fig.9. Baseline Model Traning

- We ran our experiment on Colab, which provides limited GPU consumption. For this reason, we need to keep our batch size as small as possible to not cause a memory error.
- In order to train the autoencoder, we cannot use the data loader library for denoising the data at run time. In order to fix this, we first add noise into that data, then load the all dataset into RAM.
- Due to RAM limitations, we cannot load the entire dataset into RAM. For this reason, we need to keep our dataset as small as possible.
- We did keep the autoencoder to up to 2 layers during the

encoder part and 2 layers during the decoder to keep it as simple as possible producequality digital images. Software, such as autoencoders, can be used to remove noise in digital images to obtain their high-quality print . A modified autoencoder, using a deep convolutional neural network, is the most advanced denoising software. It can help produce quality images, especially in the medical field, as well as in high-quality photography. After training with various trained datasets on a tensor board, we tested the convolutional autoencoder with test datasets. Its results were quite satisfactory but not desirable due to the physical limitations of the system, such as limited GPU consumption availability and RAM limitations. Nevertheless, our proposed system performed better than other traditional autoencoders.

•

VI. CONCLUSION

Image denoising is one of the most important techniques used to producequality digital images. Software, such as autoencoders, can be used to remove noise in digital images to obtain their high-quality print . A modified autoencoder, using a deep convolutional neural network, is the most advanced denoising software. It can help produce quality images, especially in the medical field, as well as in high-quality photography. After training with various trained datasets on a tensor board, we tested the convolutional autoencoder with test datasets. Its results were quite satisfactory but not desirable due to the physical limitations of the system, such as limited GPU consumption availability and RAM limitations. Nevertheless, our proposed system performed better than other traditional autoencoders.

REFERENCES

1. L. Gondara, "Medical image denoising using convolutional denoising autoencoders," in *2016 IEEE 16th international conference on data mining workshops (ICDMW)*, 2016, pp. 241-246.
2. H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?," in *2012 IEEE conference on computer vision and pattern recognition*, 2012, pp. 2392-2399.

3. P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 1096-1103.

4. V. Jain and S. Seung, "Natural image denoising with convolutional networks," *Advances in neural information processing systems,* vol. 21, 2008.

5. S. Karimpouli and P. Tahmasebi, "Segmentation of digital rock images using deep convolutional autoencoder networks," *Computers & geosciences,* vol. 126, pp. 142-150, 2019.

6. K. Bajaj, D. K. Singh, and M. A. Ansari, "Autoencoders based deep learner for image denoising," *Procedia Computer Science,* vol. 171, pp. 1535-1541, 2020.

7. J. Geng, J. Fan, H. Wang, X. Ma, B. Li, and F. Chen, "High-resolution SAR image classification via deep convolutional autoencoders," *IEEE Geoscience and Remote Sensing Letters,* vol. 12, pp. 2351-2355, 2015.

8. Y. Sun, B. Xue, M. Zhang, and G. G. Yen, "A particle swarm optimization-based flexible convolutional autoencoder for image classification," *IEEE transactions on neural networks and learning systems,* vol. 30, pp. 2295-2309, 2018.

9. Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Energy compaction-based image compression using convolutional autoencoder," *IEEE Transactions on Multimedia,* vol. 22, pp. 860-873, 2019.

10. V. Mirjalili, S. Raschka, A. Namboodiri, and A. Ross, "Semi-adversarial networks: Convolutional autoencoders for imparting privacy to face images," in *2018 International Conference on Biometrics (ICB)*, 2018, pp. 82-89.

11. F. Chollet, *Deep learning with Python*: Simon and Schuster, 2021.