Research Article GU J Sci 36(3): 1122-1138 (2023) DOI: 10.35378/gujs.1051655

Gazi University

Journal of Science

http://dergipark.gov.tr/gujs

A Comparative Study on Denoising from Facial Images Using Convolutional Autoencoder

Muazzez Buket DARICI[1, *], Zeki ERDEM[2]

[1]*Kadir Has University, Department of Electrical-Electronics Engineering, Istanbul / Turkey*
[2]*Kadir Has University, Department of Management Information Systems, Istanbul / Turkey*

Highlights

- This paper focuses on several denoising techniques for several noise types on facial images.
- Convolutional Denoising Autoencoder is proposed, and it is compared with traditional methods.
- A highly effective denoising autoencoder was obtained based on noise reduction metrics.

Article Info

Received: 06 Jan 2022
Accepted: 11 Aug 2022

Keywords

Denoising
Autoencoder
Salt&pepper noise
Gaussian noise

Abstract

Denoising is one of the most important preprocesses in image processing. Noises in images can prevent extracting some important information stored in images. Therefore, before some implementations such as image classification, segmentation, etc., image denoising is a necessity to obtain good results. The purpose of this study is to compare the deep learning techniques and traditional techniques on denoising facial images considering two different types of noise (Gaussian and Salt&Pepper). Gaussian, Median, and Mean filters have been specified as traditional methods. For deep learning methods, deep convolutional denoising autoencoders (CDAE) structured on three different optimizers have been proposed. Both accuracy metrics and computational times have been considered to evaluate the denoising performance of proposed autoencoders, and traditional methods. The utilized standard evaluation metrics are the peak signal to noise ratio (PSNR) and structural similarity index measure (SSIM). It has been observed that overall, while the traditional methods gave results in shorter times in terms of computation times, the autoencoders performed better concerning the evaluation metrics. The CDAE based on the Adam optimizer has been shown the best results in terms of PSNR and SSIM metrics on removing both types of noise.

1. INTRODUCTION

In recent years, as deep learning and its applications related to image processing become embedded in our lives in every sense [1], the requirement for a reliable and large amount of image data to obtain accurate results has also increased drastically [2]. However, there are some disadvantages of images such as noise. Images are generally corrupted with noise during acquisition, compression, and transmission [1]. This leads to loss of information in images and can be decreased the performance of the system. Therefore, some preprocessing techniques have been used in such systems before using images in the dataset. The most important preprocessing technique used for images is noise reduction. Image denoising is the fundamental step in image processing and computer vision [2].

In the literature, there exists a lot of noise reduction techniques used for different noise types. Some of the noise reduction techniques are classical denoising methods (Spatial Domain Filtering and Variational Denoising Methods), transform domain filtering methods (Wavelet Transform), CNN-based methods (Autoencoder) [1]. The possible types of noises in images are Gaussian Noise, Salt&Pepper Noise, Poisson Noise, Speckle Noise [3].

*Corresponding author, e-mail: buket.darici@khas.edu.tr

In the last decade, there have been lots of methods designed to remove or reduce noises in image processing. Even traditional noise reduction methods have been used for years, the methods which are based on autoencoder, have attracted much attention in recent 20 years [4].

There are various studies that use autoencoders as a noise reduction method [5-8]. [9] has created a convolutional autoencoder to construct noiseless medical images from noisy medical images. They have combined two different datasets which are mini-MIAS database of mammograms (MMM) and a dental radiography database (DX) to increase the sample size of dataset. They added some different types of noises with different parameters to their images. Images were compared using structural similarity index measure (SSIM) instead of peak signal to noise ratio (PSNR) for its consistency and accuracy. By applying Median filter, which is one of the most popular noise reduction techniques to their images, they made a comparison between Convolutional Autoencoder, and Median Filter based on SSIM values.

In [10], the authors presented automatic unsupervised feature extractors and automatic denoiser, which is convolutional variational autoencoders, to learn and extract good features directly from the raw data. In this paper, they used images from the Plant village dataset for the detection corn diseases or potato diseases. For the adding noise process, they applied Salt&Pepper noise to the input images. They have created some Autoencoders with different numbers of encoding and decoding layers. After each autoencoder, they used Fully Connected Network to make a classification. In the end, they measured the efficiency of the autoencoder as a feature extractor and denoiser. [11] used Autoencoder as a technique for denoising of DNA Microarray images (MAI). The obtaining the spot intensity of MAI is very important to diagnose some diseases, drug discovery, etc. However, obtaining this intensity from MAI could be very challenging because of these images' low contrast and noise. Therefore, in this paper, they proposed autoencoder-based image denoising for enhancing the DNA MAI. They used SIB and Derisi datasets to test their method. Their experimental results show that there is a considerable reduction in noise when compared with other recent related methods such as normal Discrete Wavelet Transform (DWT), Stationary Wavelet Transform (SWT) denoising method, etc. PSNR is the metric used for evaluating performance.

Similarly, [4] used Convolutional Autoencoder Network for facial image denoising. For this purpose, they used the ORL face database. They applied the noise reduction process using autoencoder after adding different noises to their images, such as Gaussian, Salt&Pepper and, Poisson noise. [12] introduced an image denoising technique based on a convolutional denoising autoencoder (CDAE). They trained their proposed CDAE model using 3000 chest radiograms and made comparison using median filter, non-local mean (NLM), and total variation (TV) minimization algorithms. The obtained results show their proposed CDAE model outperformed the state-of-the-art algorithms on 3000 chest radiograms.

In this study, we used convolutional denoising autoencoders (CDAE) based on different optimizers for denoising of facial images. Then, we evaluated the performance of autoencoders by comparing them with traditional denoising methods. The paper is organized as follows; Section 2. describes noise types and autoencoder methods used in this study. Section 3. shows the dataset. Section 4., Section 5., Section 6., and Section 7. include the proposed method, experimental results, discussion and conclusion, respectively. In the last section, references to further studies are given.

2. MATERIAL METHOD

2.1. Noise Types

Denoising is an important preprocessing part of image processing techniques. There may be some different noise types in images. Before processing these images for some purposes, it is necessary to remove noises to obtain better results. In this study, to see the performances of different noise reduction methods for facial images, two different noise types have been added to these images separately. Thus, the noise reduction performance of the proposed methods has been investigated for different noise types.

Gaussian noise is one of the most known noise types. It is a statistical noise whose probability density function (pdf) equals normal distribution. This noise type is presented as the following Equation (1);

$$\text{PDF} = \text{P(x)}=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

where x is the grey level, μ presents the mean of normal distribution, σ denotes the standard deviation. This type of noise has a bell-shaped pdf, which is shown graphically in Figure 1 below.

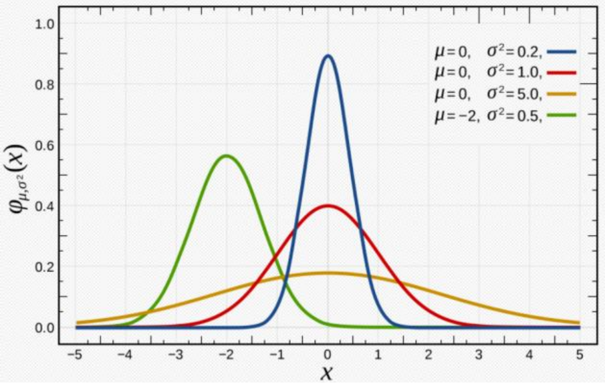


Figure 1. *Gaussian distribution for different* μ *and σ values*

Images with Gaussian noise are obtained by adding the Gaussian noise function to original images.

Noisy image function = Original image function + Gaussian noise function.

Salt&Pepper noise is also known as impulse noise or random noise. In images with impulse noise, each pixel of the image does not contain noise. Just a certain proportion of pixels affected by noise. Some image pixels are replaced by max. (white) or min. (black) pixel values. The remaining unaffected pixels are the original pixel values [13]. The pdf of Salt&Pepper noise can be seen in Figure 2 below.

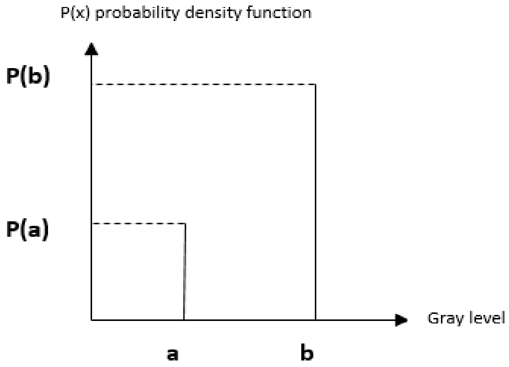


Figure 2. *The pdf of Salt&Pepper noise*

2.2. Traditional Noise Reduction Methods (Filter Types)

The filtering technique is the traditional noise reduction way from image data. In this study, the most popular filters, which are Mean, Median, and Gaussian, have been used to remove different types of noises from facial images.

The mean filter is also called an average filter. The main idea behind the mean filter is to replace each pixel value with the average value of the neighboring pixel by moving the filter box through the image pixel by pixel. This approach makes the images smoother. This filter is one of the linear filters since the basic convolution process is applied to obtain smoother images.

The Median filter is one of the non-linear filtering techniques [3]. Using Median filter, a window is slid along with image in x-axes anda y-axes. Each pixel value in an image is replaced by median value of slid window. Since there is no convolution (matrix multiplication) process, this filter is a non-linear filter.

Similar to the Mean filter, the convolution process is also applied in Gaussian filtering. The difference between these two filters is the kernel. In Gaussian filtering, a kernel that represents the shape of a Gaussian hump is used. That is the reason this filtering technique is called Gaussian.

2.3. Denoising Autoencoder

Autoencoders (AEs) are one of the unsupervised learning algorithms. Autoencoders aim to generate new data using two processes which are encoding and decoding. In the process of encoding, the input is compressed into latent space then reconstruction of input is made by decoder based on information included in the latent variables. Therefore, the encoder-decoder pair can be thought of as a two symmetrical neural network [14]. Autoencoders play an important role in extracting low dimensional features to use in machine learning or deep learning algorithms. Autoencoders can be used for different purposes such as dimension reduction, noise reduction, etc. In this study, the noise reduction feature of the autoencoder will be used. The basic representation of autoencoder as a noise reduction method is shown in Figure 3.

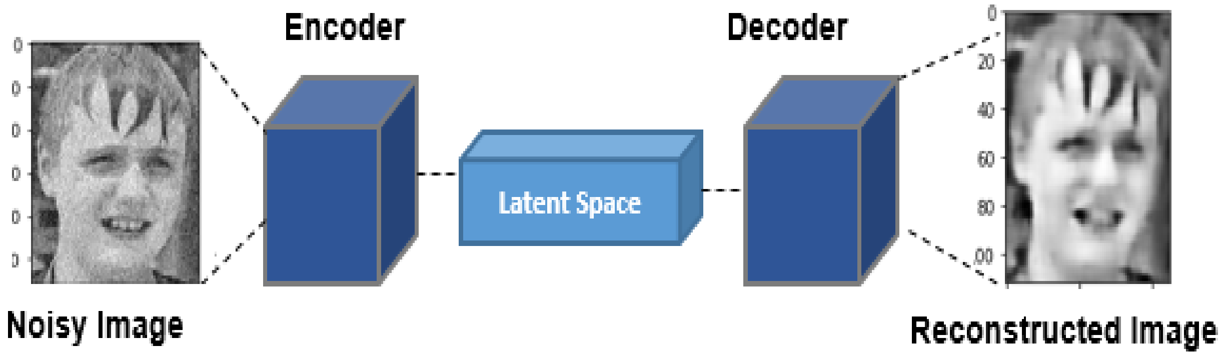


***Figure 3**. The basic representation of an autoencoder*

As the mathematical expression, an autoencoder takes input set X from unlabeled data. For a denoising autoencoder, X consists of noisy images. Encoder function takes the X input into latent space representation, which is Y. In the decoder part, the identity input set was tried to be obtained in original size with minimum error [15]. If the output of the decoder part is represented with Z, a general description of autoencoder can be seen as follows;

X: input (noisy images) (2)

Encoder: $s(WX+b) = Y$ (3)

Decoder: $s(W'Y+b') = Z$ (4)

where W,b, W', b' are parameters used in autoencoder. s is a non-linear function.

3. THE DATASET

To be able to see the effect of CDAE, it has been added some types of noise to the images. In this section, it is shown the dataset with non-noised images, gaussian noised images, and salt&pepper noised images, respectively.

3.1. Original Dataset

The dataset[1] for this study is obtained from Kaggle. This dataset which will be used in this study are facial images, and it consists of two classes which are labeled as autistic and non-autistic. Each class has 1470 RGB images. So, in total, we have 2940 images whose sizes are 224x224x3. Some samples of this dataset can be seen in Figures 4 and 5 below.

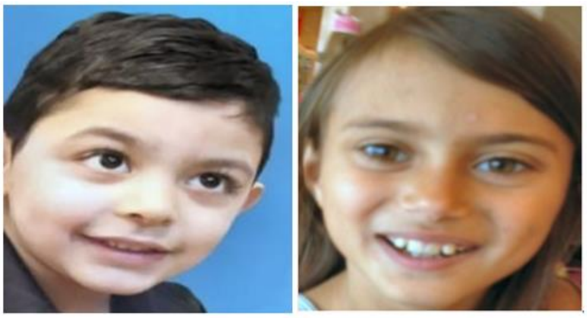


***Figure 4**. Sample images of the dataset which are labeled as autistic.*

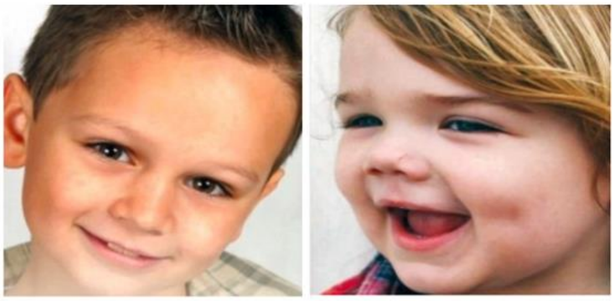


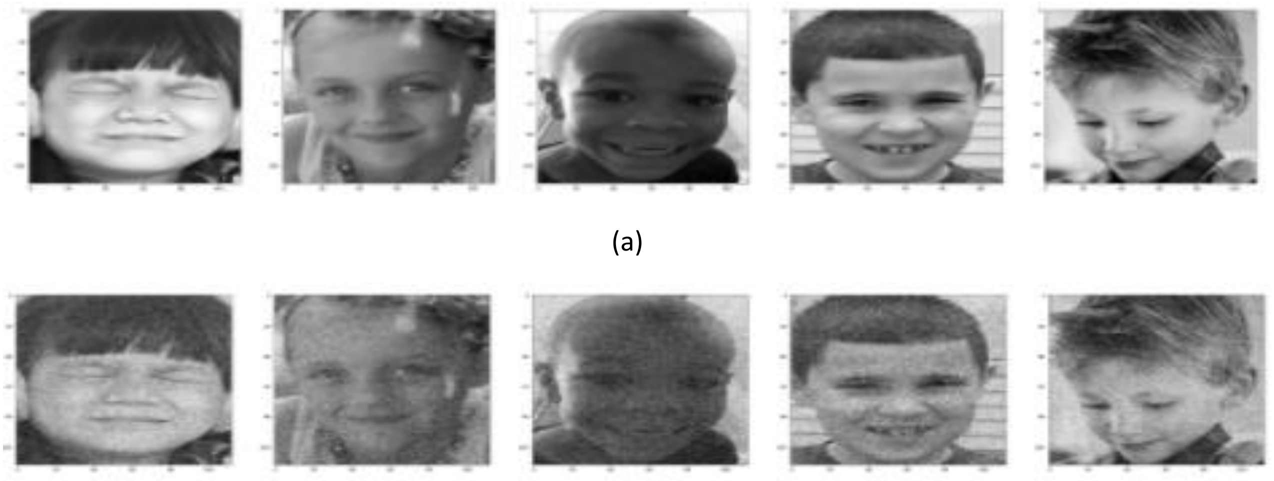
***Figure 5**. Sample images of the dataset which are labeled as non-autistic*

Moreover, all images have been changed to 112X112 shaped grayscale images to make computation easier. Before all processes, the dataset has been split into training and testing samples by the rate of 80% and 20%, respectively.

3.2. The Dataset with Gaussian Noisy Images

Mean μ of normal distribution and standard deviation σ of Gaussian noise are 0 and 0.07 respectively in this study.   Some examples of clean images and their Gaussian noisy states can be seen in Figure 6 below.

[1] https://www.kaggle.com/cihan063/autism-image-data



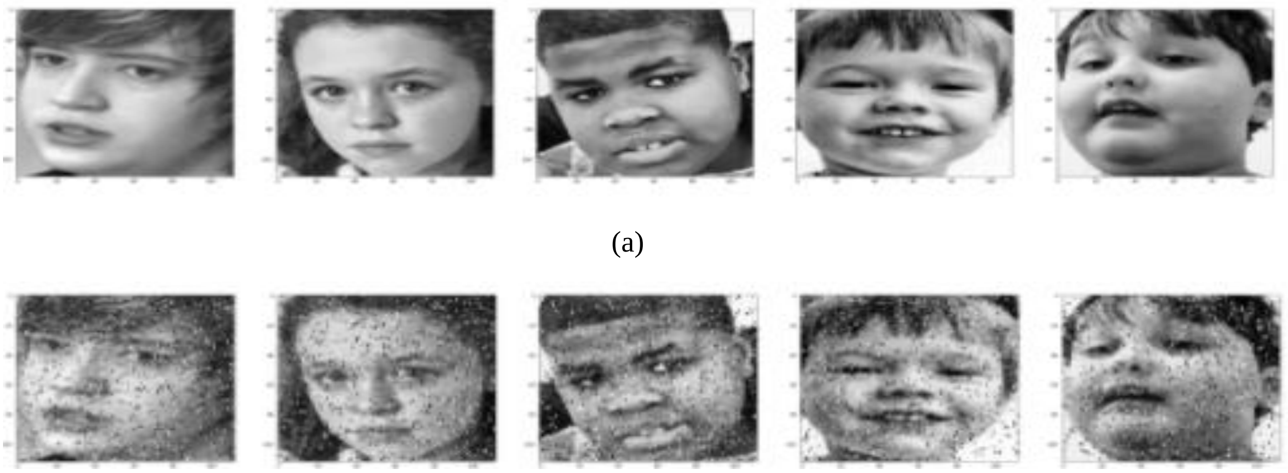
(a)

(b)

***Figure 6.** Some samples of (a) original images, (b) Gaussian noisy images*

3.3. The Dataset with Salt&Pepper Noisy Images

The threshold value was set to 0.05 to add Salt&Pepper noise to images in this study. Some examples of clean images and their Salt&Pepper noisy states can be seen in Figure 7 below.



(a)

(b)

***Figure 7.** Some samples of (a) original images, (b) salt&pepper noisy images*

4. THE PROPOSED METHOD

In this section, the noises in images have been tried to be removed using traditional filtering techniques and the proposed CDAE. All processes related to adding gaussian and salt&pepper noises have been repeated for both autistic and non-autistic image set. The flow chart of the methodology for image denoising can be seen in Figure 8 below.

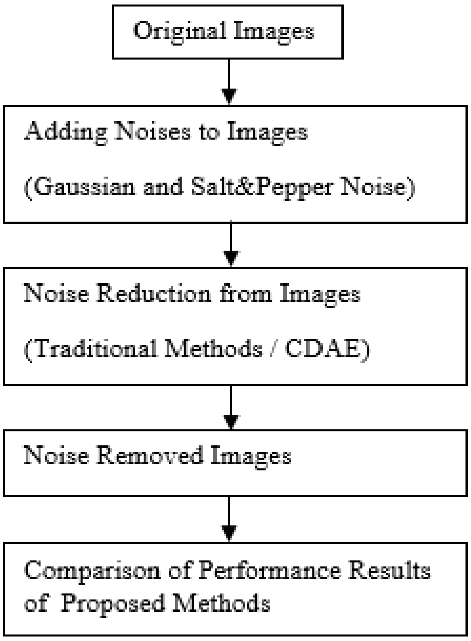

Original Images
Adding Noises to Images
(Gaussian and Salt&Pepper Noise)
Noise Reduction from Images
(Traditional Methods / CDAE)
Noise Removed Images
Comparison of Performance Results of Proposed Methods

Figure 8. *The flow chart of the methodology*

For each part, all images in the dataset were resized to 112X112 grayscale images. Each part of the methods has been described in Section 4.1, 4.2, and 4.3, respectively.

4.1. The Traditional Filters

In this study, three different most known filtering techniques have been used as a noise reduction method. These filtering techniques are Mean, Median, and Gaussian filters. For all filters, 3x3 kernel size has been used in this study. Each filtering technique has been applied to 112X112 sized grayscale images, which added Gaussian and Salt&Pepper noise separately.

4.2. Convolutional Denoising Autoencoder

As aforementioned, in this study, the autoencoder has been implemented as a noise reduction technique. The reason why this autoencoder is called Convolutional Denoising Autoencoder is that this model consists of convolution layers. The proposed CDAE model has been performed as 7 layers consisting of 3 layers in the encoder part, 3 layers in the decoder part, and an output layer that obtains an image with the original size (Figure 9).

Each of 3 layers in the encoder part consists of a 2D-Convolution and an activation function. Convolution layers have been used as a feature extractor [16], and as an activation function, Leaky-ReLU has been used. The choice of activation function for a model is an important step to good accurate results [17].

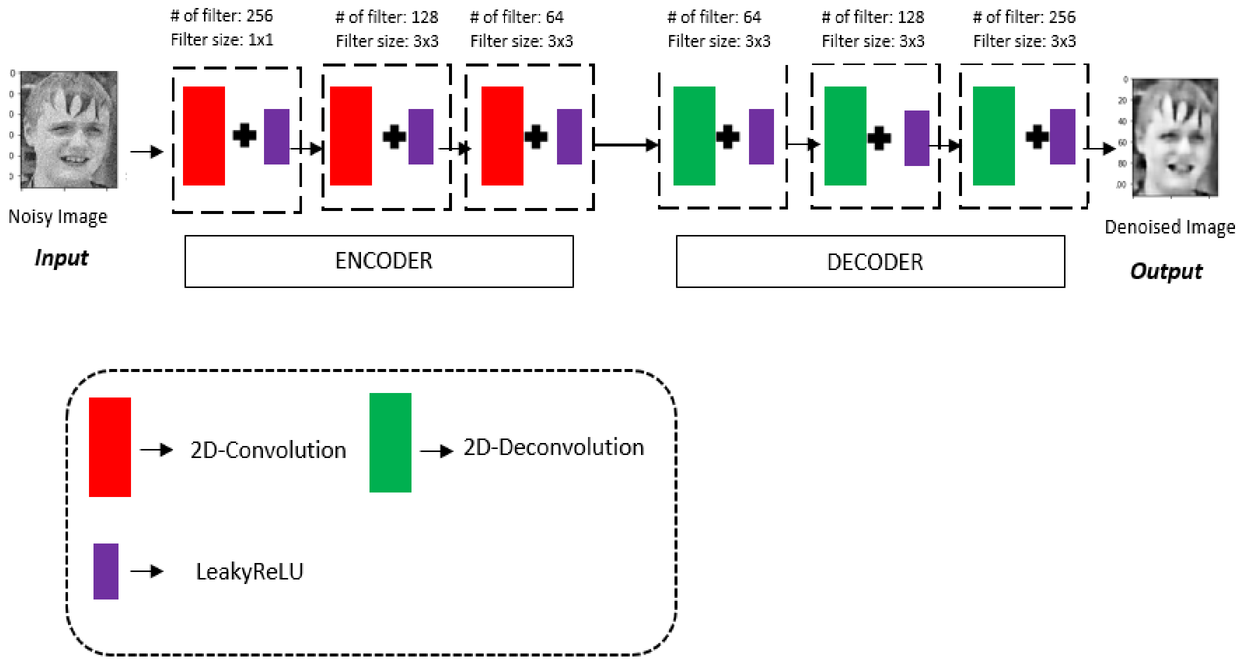


Figure 9. *The proposed CDAE model for image denoising*

There are some different activation functions such as ReLU, sigmoid, softmax, tanh. Leaky-ReLU has been preferred to convert linear inputs to non-linear output in this study because this activation function could have some benefits over ReLU, such as solving the vanishing gradient problem by avoiding the zero gradient's part and speeding up the network's learning and convergence as shown in [18]. For the convolution layers in the encoder part, the number of filters/channels are 256, 128, and 64, respectively. Filter sizes used during convolution processes are 1x1, 3x3, and 3x3, respectively. For the first convolution layer, zero padding has been applied to input, and 1x1 filters are moved one by one, which means stride is 1 during the convolution. For the rest convolution layers, stride value has been set to 2 with zero padding. At the end of the encoder part, we will have 56x56x64 sized output which is used as an input for the decoder part. In the decoder part of the CDAE, we have 3 layers similar to the encoder part. Unlike the encoder, these layers consist of deconvolution with Leaky-ReLU transfer function. The number of filters used for deconvolution processes is 64, 128, and 256, respectively. And the filter sizes are 3x3 for each layer. At the end of the decoder, we should have an output which has the same size with the input of the encoder. While convolution layers are used to reduce the size of the input in the encoder part, deconvolution layers, the reverse of the convolution process, are used to increase the size of input in decoder part. The last layer of the decoder is the output layer which consists of a deconvolution layer with a 3x3 filter and Sigmoid function. The purpose of this layer is to reconstruct the original sized image back. The parameters used during the training process of the autoencoder can be seen in Table 1.

***Table 1.** Hyperparameters*

Hyperparameters
Optimizer = 'RMSprop', 'Adam', 'Adamax', 'Nadam' (lr=0.001)
Loss = binary_crossentropy
Epoch = 50
Batch_size = 8

As seen in Table 1, several experiments were done by changing optimization parameter during training the CDAE. The used optimizers are Root Mean Square Propagation (RMS Prop), Adaptive Momentum (Adam), Nesterov Adaptive Momentum (Nadam), Adaptive Max Pooling (Adamax) [19]. The learning rate (lr) of 0.001 and binary cross entropy to calculate the training loss are used. The number of epoch and batch sizes are set to 50 and 8, respectively for all experiments. These are the well-known parameters in deep learning. In this study, since it is focused on the effect of optimization function on Autoencoder, the other parameters such as learning rate, epoch, batch size were kept constant. The main differences of this study than the similars [4-11] are the used dataset, similarity metrics and parameters. This study differs from the similar studies in the literature by testing well-known traditional and deep learning-based noise removal methods from face images of well-known noises with different metrics.

5. EXPERIMENTAL SETUP AND RESULTS

Peak Signal-to-Noise Ratio (PSNR) and structural index similarity (SSIM) are the most referred metrics to measure the accuracy in respect of removing the noise from images [20, 21]. To be able to make a comparison between traditional noise reduction methods and CDAE, PSNR values, and SSIM indexes have been computed between clean (original) images and noise removed images.

The PSNR value is obtained from Equation (4) [20]:

$$\text{PSNR} = 10log_{10}\left(\frac{(L-1)^2}{MSE}\right) = 20log_{10}\left(\frac{(L-1)^2}{RMSE}\right) \tag{5}$$

where

$$\text{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(O(i,j) - D(i,j))^2 \,. \tag{6}$$

In Equation (5), MSE stands for mean square error. RMSE is the root of MSE. O is the matrix of original images. D is the matrix of images after the denoising process. Obtaining a bigger PSNR value in dB means that the denoising process is more effective.

SSIM is a newer measurement metric than PSNR that is based on luminance, contrast, and structure. The SSIM gives the similarity index between two images whose value is between −1 and +1. 1 means that the two images are exactly same. In this study, Equation (6) [20] is used to obtain SSIM value of images:

$$\text{SSIM}\,(I, I_n) = \frac{(2\mu_I\mu_{I_n+c_1})(\,2cov+\,c_2)}{(\mu_I{}^2+\mu_{I_n}{}^2+c_1)(\sigma_I{}^2+\sigma_{I_n}{}^2+c_2)} \tag{7}$$

$$\text{where}\begin{cases} c_1 = (k_1L)^2;\; k_1 = 0.01 \\ c_2 = \;(k_2L)^2;\; k_2 = 0.03 \end{cases}$$

where μ_I is the mean of the original image and $\mu_{I_n}$is mean of the noisy image; ${\sigma_I}^2$ is the variances of the original images and ${\sigma_{I_n}}^2$ is the the variances of noisy images; *cov* is the value of noisy image covariance and L = 2nbp-1, where nbp is the number of bits per pixel. All PSNR values were obtained in dB.

All the source codes are run on Kaggle, provides free access to NVIDIA TESLA P100 GPUs, using OpenCV and Tensorflow that are the libraries of Python programming language.

5.1. Removing of Gaussian Noise and Salt&Pepper Noise

In this part, the results of traditional methods and the proposed CDAE models, which remove the noises in facial images containing Gaussian noise, and Salt&Pepper noise, have been presented and discussed.

In Tables 2 and 3, PSNR values, SSIM indexes, and computational times for both traditional methods and CDAE models trained with different optimizers can be seen for Gaussian noise and Salt&Pepper noise, respectively.

***Table 2**. PSNR and SSIM values between the original and denoised images and computational times of applied traditional methods and the proposed CDAE trained with different optimizers to remove Gaussian noise*

		After Denoising						
		Traditional Filtering Techniques			Proposed CDAE with different optimizers			
	Before Denoising	Median	Mean	Gaussian	RMSprop	Adam	Nadam	Adamax
PSNR (dB)	**23.09**	26.48	26.55	27.44	29.18	**30.2**	29.66	29.43
SSIM	**0.62**	0.77	0.80	0.82	0.89	**0.9**	0.89	0.88
Computational Time/ image (ms)	-	0.06	0.024	0.04	1.67	1.54	1.57	1.54

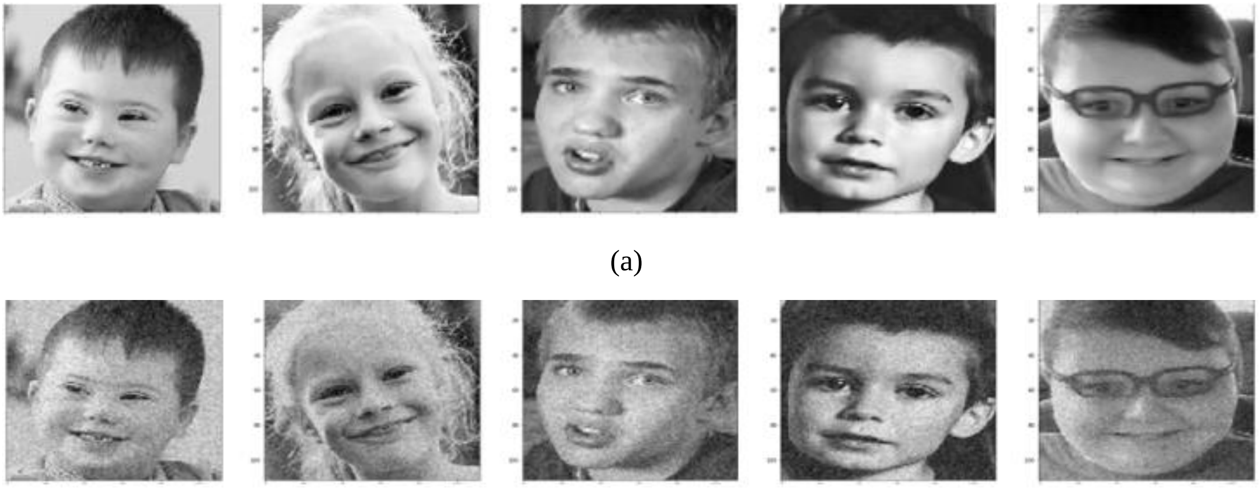
***Table 3.** PSNR and SSIM values between the original and denoised images and computational times of applied traditional methods and the proposed CDAE trained with different optimizers to remove Salt&Pepper noise*

		After Denoising						
		Traditional Filtering Techniques			Proposed CDAE with different optimizers			
	Before Denoising	Median	Mean	Gaussian	RMSprop	Adam	Nadam	Adamax
PSNR (dB)	**15.10**	28.38	22.19	22.0	30.70	**34.1**	31.83	30.2
SSIM	**0.30**	0.90	0.57	0.56	0.95	**0.97**	0.96	0.93
Computational Time/ image (ms)	-	0.034	0.05	0.05	1.55	1.55	1.54	1.54

While the mean values of PSNR and SSIM of images added Gaussian noise is 23.09 dB, 0.62, respectively, these values are increased to 30.02 dB and 0.9 after applying the CDAE trained with Adam optimizer. Similarly, while the mean values of PSNR and SSIM of images added the Salt&Pepper noise result in 15.10 dB, 0.30, respectively, these values are increased to 34.1 dB and 0.97 after applying the CDAE trained with Adam optimizer. The results show that regarding the Gaussian noise, the Gaussian filtering technique showed the highest denoising performance in terms of the value of SSIM with a 20% improvement rate and an increase of 4.35 dB among the traditional methods, and the CDAE trained with Adam optimizer provided a 28% improvement rate and an increase of 7.11 dB than all traditional methods. In case of the Salt&Pepper noise, the Median filtering technique revealed the best noise removal performance in terms of the value of SSIM with a 60% improvement rate and an increase of 13.28 dB among the other traditional methods. However, the CDAE trained with Adam optimizer provided a 67% improvement rate and an increase of 19 dB than all traditional methods for the same noise type.

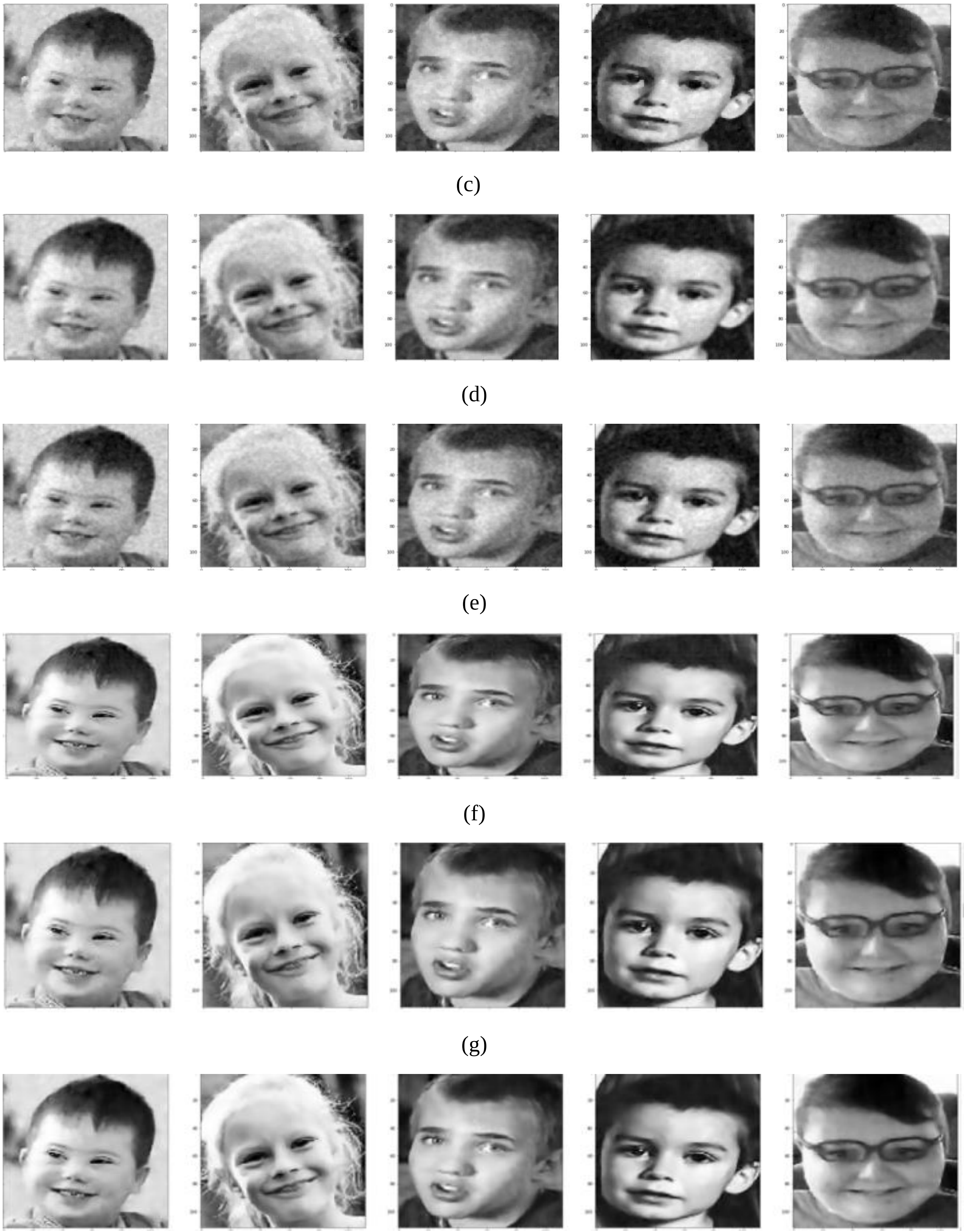
Concerning the PSNR and SSIM, when we compare traditional noise removing techniques with the proposed CDAE in order to remove the two types of noise from facial images, we can quantitatively observe that the CDAE models based on all optimizers outperform the traditional filters. Furthermore, the Adam-optimized CDAE model has given the best results also compared to the other optimizers for PSNR and SSIM values.

Additionally, some examples of visual outputs have been given for all methods that remove the two types of noise. The differences between the results of the denoising methods can be visually observed in Figures 10 and 11, respectively. So, the visual improvements in denoised images also support the numerical PSNR and SSIM results qualitatively. We also measured the computational times for all noise removing methods. The CDAE models, which have higher computational costs [7 ,21] have been taken a little more time to remove noise than conventional methods. However, it is considered that these times are within acceptable values. It should be more focused on optimizing the CDAE model architecture and parameters to reduce the denoising time further.



(a)

(b)



(c)

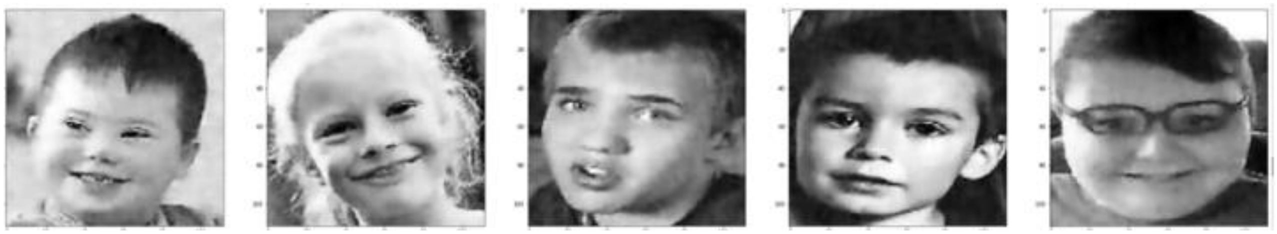
(d)

(e)

(f)

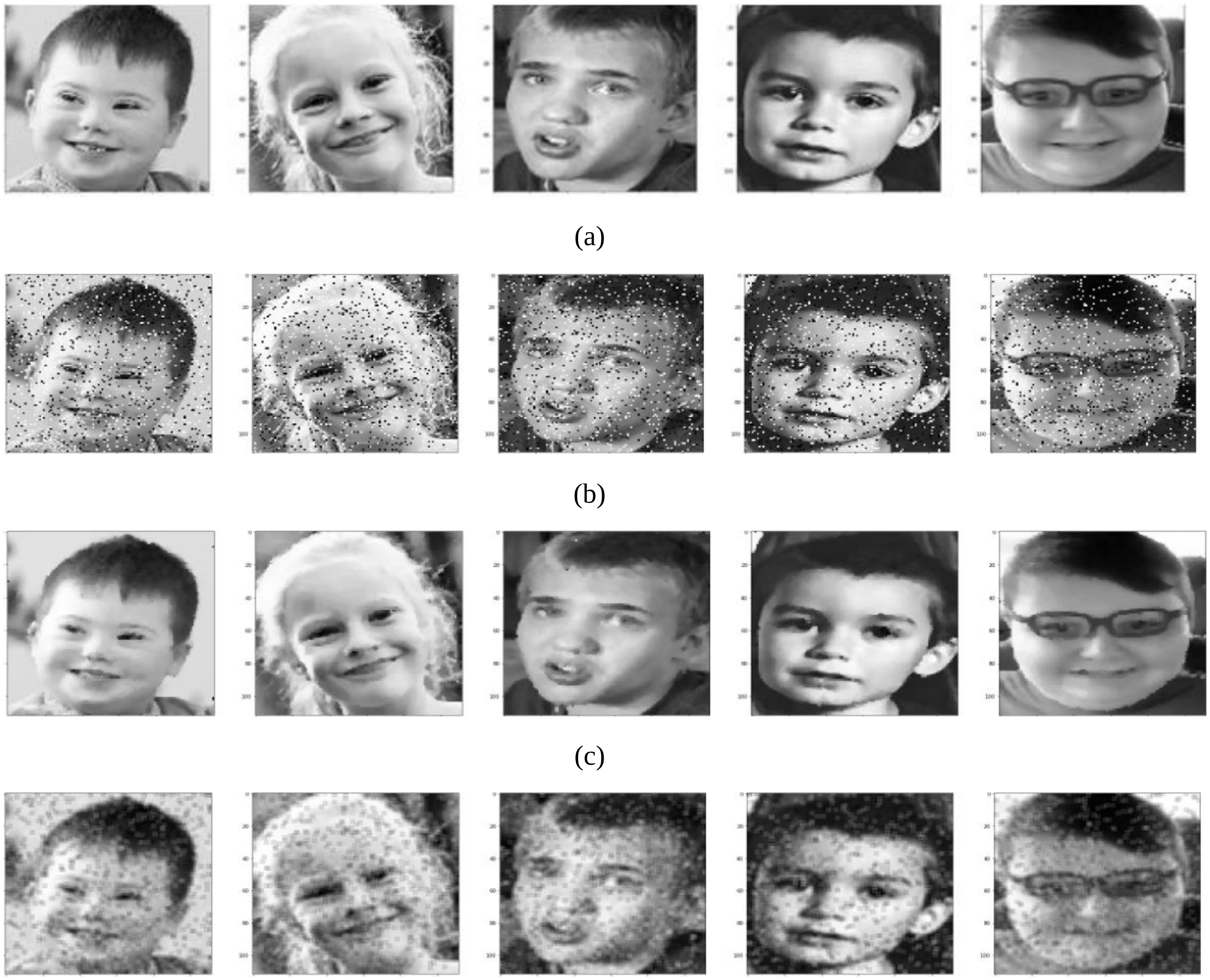
(g)

(h)



(i)

Figure 10. *The results of all denoising methods on facial images: (a) Original, (b) with Gaussian Noise, Traditional Noise Removers: (c) Median filter, (d) Mean Filter, (e) Gaussian Filter, The Proposed CDAE trained with optimizer of: (f)RMSprop, (g) Adam, (h) Nadam, (i) Adamax*

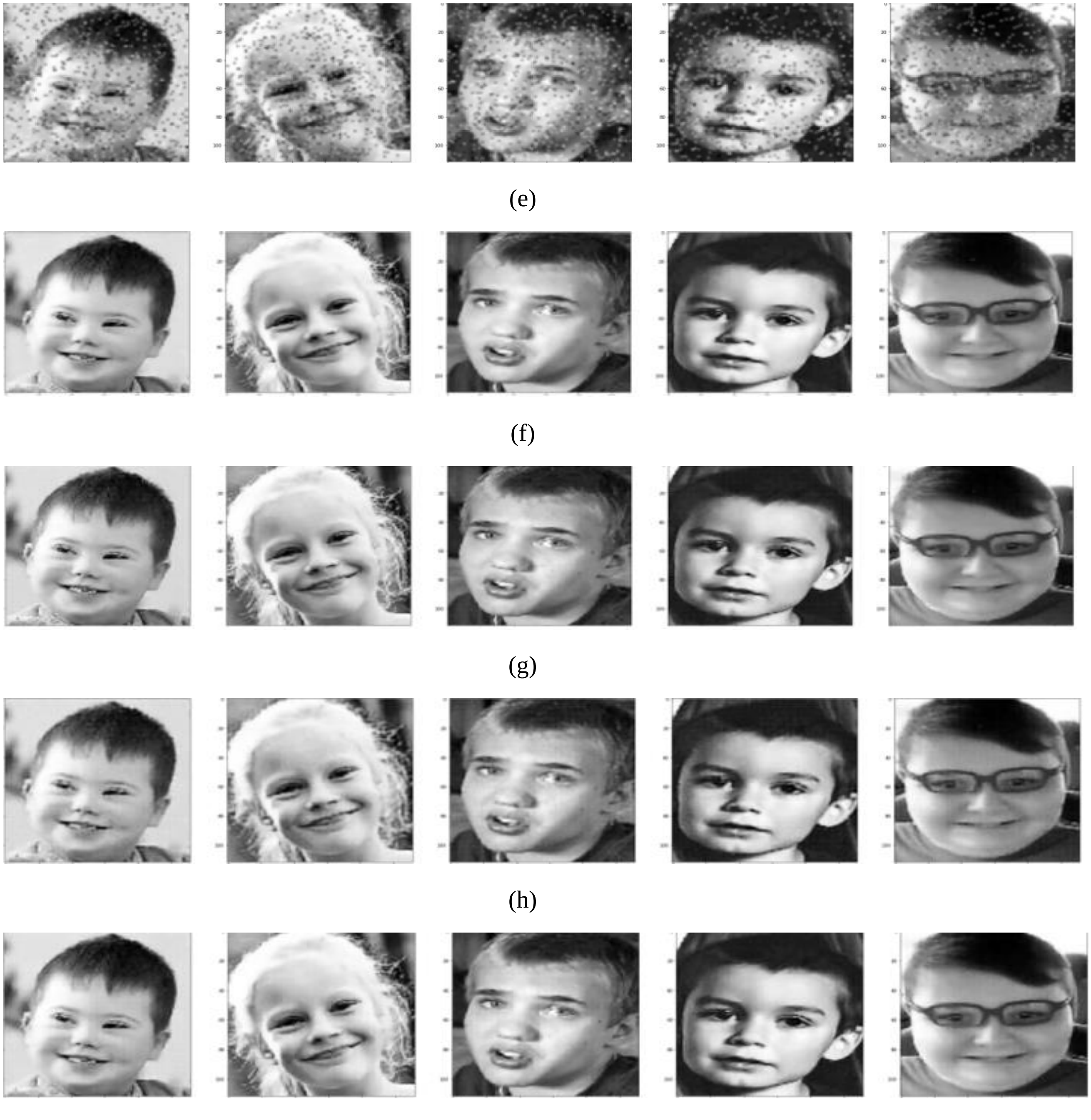


(a)

(b)

(c)

(d)



(e)

(f)

(g)

(h)

(i)

Figure 11. *The results of all denoiser methods on facial images: (a) Original, (b) with Salt&Pepper Noise, Traditional Noise Removers: (c) Median filter, (d) Mean filter, (e) Gaussian Filter, The Proposed CDAE trained with optimizer of: (f)RMSprop, (g) Adam, (h) Nadam, (i) Adamax*

6. DISCUSSION

This study shows that the best performance of the noise reduction from facial images was 30.2 dB PSNR and 0.9 SSIM for Gaussian noised images and 34.1 dB PSNR and 0.97 SSIM for Salt&Pepper noised images. These results were obtained by CNN-based autoencoder and Adam optimization function for both noise type. Since this study researched the effectiveness of optimization function on noise reduction Autoencoder, the other parameters used in Autoencoder were kept constant. In the future, the effect of other parameters on the model performance should be investigated.

Similar studies to ours in the literature have used different types of images, different noise reduction methods with different similarity measures. Gondara et al. [9] proposed an autoencoder-based noise reduction method on medical images. They used just Median filter as a traditional method and SSIM as a similarity metric to be able make a comparison. Zilvan et al. [10] focused on the noise reduction performance of autoencoder with different number of layers. They used plant images for their disease classification model. In [11], Mohandas et al. used autoencoder as a noise reduction technique on DNA Microarray images. They compared their proposed method with some denoising methods such as Discrete Wavelet Transform (DWT), Stationary Wavelet Transform (SWT) based on just PSNR evaluation metric.

Since the similar studies based on noise reduction of images used different types of images and methods, it is not appropriate make a comparison between theirs and ours. This study differs from the others since it includes comprehensive comparison between different types of noises (Gaussian, Salt&Pepper), different types of noise reduction methods (traditional and deep learning-based) based on different types of evaluation metrics (SSIM, PSNR). Morever, in this study, different from the others, the computational time is also included.

7. CONCLUSION

In this study, the performance of some noise reduction methods has been investigated based on PSNR and SSIM values. To be able to make a comparison between the used methods, two different types of noise, Gaussian noise, and Salt&Pepper noise, have been added to facial images. In conclusion, we observed that the traditional methods used in this study are heavily affected by the type of noise. Even some methods give very accurate results for some noise types; others may give really poor results. The proposed CDAE, on the other hand, removes noise from the image regardless of the type of noise. Since we can’t always know the type of noise in real life, we need systems that remove all kinds of noise from the image well. For this reason, the CDAE, which removes two different noises from the images better than the traditional methods, has been recommended in this study. Thanks to CDAE trained with Adam optimizer for both noise types, the noise has been removed, and the closest images to the original ones have been obtained. It showed the best denoising performance in comparison to the CDAE based on other optimizers and all traditional methods with its acceptable computational time considering the high computational cost. In the future, we plan to investigate optimal model architectures and parameters, which will be able to reduce the denoising time further. Moreover, we will examine the effects of original images and denoised images on the performance of classifiers.

CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

[1] Fan, L., Zhang, F., Fan, H., and Zhang, C., “Brief review of image denoising techniques”, Visual Computing for Industry, Biomedicine and Art, 7, (2019).

[2] Alisha P. B., and G. S. K., “Image Denoising Techniques-An Overview”, 11: 78–84, (2016).

[3] Patidar, P., Gupta, M., Srivastava, S., and Nagawat, A. K., “Image De-noising by Various Filters for Different Noise”, International Journal of Computer Applications, 9: 45–50, (2010).

[4] Tun, N. M., Gavrilov, A. I., and Tun, N. L., “Facial image denoising using convolutional autoencoder network”, Proceedings - 2020 International Conference on Industrial Engineering, Applications and Manufacturing, ICIEAM 2020, 1–5, (2020).

[5] Bajaj, K., Singh, D. K., and Ansari, M. A., “Autoencoders Based Deep Learner for Image Denoising”, Procedia Computer Science, 171: 1535–1541, (2020).

[6] Nishio, M., “Convolutional auto-encoders for image denoising of ultra-low-dose CT”, Heliyon, 3, (2017).

[7] Çetinkaya, E., and Kiraç, M. F., “Image denoising using deep convolutional autoencoder with feature pyramids”, Turkish Journal of Electrical Engineering and Computer Sciences, 28: 2096–2109, (2020).

[8] Kumar, A., and Sodhi, S. S., “Comparative Analysis of Gaussian Filter, Median Filter and Denoise Autoencoder”, in Proceedings of the 29 th International Conference on Machine Learning, 1627–1634, (2012).

[9] Gondara, L., “Medical Image Denoising Using Convolutional Denoising Autoencoders”, IEEE International Conference on Data Mining Workshops, ICDMW, 241–246, (2016).

[10] Zilvan, V., Ramdan, A., Suryawati, E., Kusumo, R. B. S., Krisnandi, D., and Pardede, H. F., “Denoising Convolutional Variational Autoencoders-Based Feature Learning for Automatic Detection of Plant Diseases”, ICICOS 2019 - 3rd International Conference on Informatics and Computational Sciences: Accelerating Informatics and Computational Research for Smarter Society in The Era of Industry 4.0, Proceedings, 0–5, (2019).

[11] Mohandas, A., Joseph, S. M., and Sathidevi, P. S., “An Autoencoder based Technique for DNA Microarray Image Denoising”, Proceedings of the 2020 IEEE International Conference on Communication and Signal Processing, ICCSP 2020, 1366–1371, (2020).

[12] Lee, D., Choi, S., and Kim, H. J., “Performance evaluation of image denoising developed using convolutional denoising autoencoders in chest radiography”, Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 884: 97–104, (2018).

[13] Hambal, A. M., Pei, Z., and Libent Ishabailu, F., “Image Noise Reduction and Filtering Techniques”, International Journal of Science and Research, 6: 2319–7064, (2015).

[14] D’Angelo, G., and Palmieri, F., “Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction”, Journal of Network and Computer Applications, 173, (2021).

[15] Attia, A., and Chaa, M., “Individual Recognition System using Deep network based on Face Regions”, International Journal of Applied Mathematics Electronics and Computers, 6: 27–32, (2018).

[16] Darici, M. B., Dokur, Z., and Olmez, T., “Pneumonia Detection and Classification Using Deep Learning on Chest X-Ray Images”, International Journal of Intelligent Systems and Applications in Engineering, 8: 177–183, (2020).

[17] Adweb, K. M. A., Cavus, N., and Sekeroglu, B., “Cervical Cancer Diagnosis Using Very Deep Networks over Different Activation Functions”, IEEE Access, 9: 46612–46625, (2021).

[18] Xu, B., Wang, N., Chen, T., and Li, M., “Empirical Evaluation of Rectified Activations in Convolutional Network”, [Online]. Available: http://arxiv.org/abs/1505.00853 (2015).

[19] Yaqub, M., Feng, J., Zia, M., Arshid, K., Jia, K., and Rehman, Z., “State-of-the-art CNN optimizer for brain tumor segmentation in magnetic resonance images”, Brain Sciences, 10: 1–19, (2020).

[20] Ebrahimnejad, J., and Naghsh, A., “Removal of High-Density Salt-and-pepper Noise for Robust ROI Detection Used in Watermarking of MRI Images of the Brain”, Computers in Biology and Medicine, 137: 104831, (2021).

[21] Chen, M., Xu, Z., Weinberger, K. Q., and Sha, F., “Marginalized denoising autoencoders for domain adaptation”, Proceedings of the 29th International Conference on Machine Learning, ICML 2012, 1: 767–774, (2012).