Medical Image Enhancement Based on Convolutional Denoising Autoencoders and GMD Model

Serhii Yaroshchak[1], Mahmoud Smaida[2] and Youness El Barg[3]

[1]*The national university of water and Environmental Engineering, Rivne, Ukraine*
[2]*The national university of water and Environmental Engineering, Rivne, Ukraine*
[3]*Multi-disciplinary Faculty Laraxche, Larache, Morocco*

Abstract

Eye diseases include Glaucoma, Myopia and Diabetic retinopathy are very serious health problems in the life of people. Timely, early diagnosis of these diseases is very important to avoid blindness. There are many methods have been developed for this purpose.

In this paper, we demonstrate the using of Convolutional Denoising Autoencoders (CDAE) to enhance the images we obtained from DCGAN (which we obtained from previous paper). In addition, using GMD model (Glaucoma, Myopia and Diabetic retinopathy) to compare the accuracy of the model using, original data only, original data with synthetic data which we obtained from DCGAN (obtained from pervious paper) and original data with high resolution data which we obtained from (CDAE). Our dataset consists of four types of eye diseases, Glaucoma, Myopia, Diabetic retinopathy and Normal. All these data were collected from Kaggle and proved from ophthalmologist. Firstly, we exploit CDAE to enhance the medical images (make it high-resolution). Then we utilize GMD method for eye diseases classification. finally, training our method using different types of dataset and comparative results in terms of accuracy are presented. The accuracy of the model had improved significantly from 89.84 % in training set and 89.94% in validation set, to 92.82% in training set and 93.32% in validation set based on this work.

Keywords 1

Eye diseases, Deep learning, Denoising Autoencoders, GMD model

1. Introduction

The most common eye diseases in the world are Glaucoma, Myopia, Diabetic retinopathy, all these causes lead to blindness if they are not detected at an early stage. In recent years, the diagnosis of diseases of the human visual system has advanced greatly to technological innovations and developments in the field of artificial intelligence. Ophthalmologist can detect a specific disease after a visual analysis of images. However, in a large number of cases, the diagnosis is not made due to many factors, such as bad experience, fatigue, a variety of shapes, similarities, poor image quality, etc. In these cases, the second opinion is very important and useful, which comes from another expert who uses advanced artificial intelligence and algorithms to accurately analyze the image to diagnose eye diseases. A large number of diagnostic equipment, tools, methods and algorithms have been developed. Recently, Convolutional neural networks (CNNs) are widely uses in medical image classification [1].

When we use supervised learning in CNNs to classify eye diseases, we need to deal with a large number of images to learn our model. One of the main challenges in the field of medical images is how to deal with small data sets, because medical images are not available to all researches as other images.

Although, some of medical datasets are available online, most data sets remain limited in size and only apply to specific medical problems. The collection of medical data is a complex and expensive procedure that requires the cooperation of researchers and clinics.

Many researchers are trying to overcome this challenge by using augmentation diagrams, including making some adjustments to the images of the data set such as rotation, cropping and size. Deep Convolutional Generative Adversarial Network (DCGAN) method have been used to generate synthetic medical images in order to increase the dataset. These data (synthetic medical images) with original dataset were used to classify eye diseases, and it gave better accuracy than using the original data alone. However, we noticed that these images generated using DCGAN were low image quality. Therefore, we suggested that we must work to enhance these images to obtain high image quality in order to improve the accuracy of our model [1], [2].

DCGAN is a deep convolutional neural network and one of the most popular and successful networks for GANs. It consists of a set of convolutional layers without max pooling or fully connected layers. It uses stride and transposed convolution for the downsampling and the upsampling [17].

In this paper, we investigate the problem of low image quality that generated from DCGAN (obtained from previous paper) and we propose Convolutional Denoising Autoencoders (CDAE) to solve this problem, which is able to reconstruct corrupted data. After that we will use our model (GMD model) to compare the accuracy of the model using, original data only, original data with synthetic data which we obtained from DCGAN (obtained from pervious paper) and original data with high resolution data which we obtained from (CDAE).

2. Related work

Gondara, Lovedeep. [3] the authors in this paper show that denoising autoencoders constructed using convolutional neural network can be improve effectively denoising of medical images. To boost the sample size to increase the noise reduction performance, heterogeneous images can be combined. The simplest networks can reconstruct images with so high corruption levels that the noise and signal are indistinguishable from the human eye.

Lee, Donghoon, Sunghoon Choi, and Hee-Joung Kim. [4] In this study, the authors used an image denoising technique based on a convolutional denoising autoencoder (CDAE), they comparing existing image denoising algorithms with traditional algorithms including median filter, total variation (TV) minimization and non-local mean (NLM) algorithms. Chest radiograms were used as a dataset. they trained their model (CDAE) using 3000 chest radiograms. The results demonstrate that the proposed denoising algorithm developed using CDAE achieves a superior noise-reduction effect in chest radiograms compared to the other algorithms mentioned above.

Chiang, Hsin-Tien, et al. [5] authors in this study proposed denoising autoencoder (DAE) using the fully convolutional network (FCN) for signal denoising and compressing the size of ECG waveforms. the FCN-based DAE consists of an encoder and a decoder with 13 layers, the purpose is to reduce the size of ECG signals, and the signals will be encoded into low dimensional features. The decoder tries to reconstruct output depending on the low dimensional features. The results showed that FCN performs better compared to noise reduction models based on a deeply connected neural network and convolutional neural network. Additionally, the proposed FCN-based DAE reduces the size of the ECG input signals, with the compressed data being 32 times smaller than the original.

Ali, Irfan, et al. [6] In this paper, the authors use autoencoder network on RGB color scheme dataset, the CIFER-10 have been used as dataset, 60000 color photos utilized in their model. Firstly, they added Gaussian noise on CIFER-10 dataset, then they used 2D convolutional neutral network to encoding. Finally, decode noisy dataset to train their model. After training, the result of the method can learn denoising data and returns effective results.

Roy, Sudipta Singha, et al. [7] the proposed method structures consists of three denoising autoencoder (DAE) and three Convolutional Neural Network (CNN) in different structure layers. where each of the structures is responsible

for dealing with images corrupted by a specific noise level. The CNN in all structures is used for classification purpose. The DAE is trained for regular level noise. Classification with CNN after

restoring images through DAE. Since DAE is trained with regular level noise, different structures with different organizations of this DAE are managed to make the system adaptable.

3. Material and method
3.1. Datasets

Our dataset is divided in two types, the first type is original fundus images which collected in total from Kaggle and iChallenge-GON Comprehension, in high resolution. The second type of dataset is synthetic fundus images generated from Deep Convolutional Generative Adversarial Network (DCGAN) which we obtained from the previous paper. The dataset was analyzed and preprocessed of four different classes, Glaucoma, Myopia, Diabetic retinopathy and Normal images. Figure 1 And figure 2, shows original and synthetic dataset respectively. The total number of fundus images in original dataset are 2050 images, 1742 used as train set and 308 used as validation set. The synthetic fundus images which obtained from DCGAN are 655 synthetic images distributed as in Table.1.
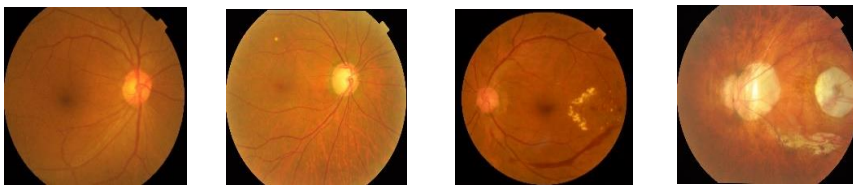

Figure 1: Normal Fundus, Glaucoma, Diabetic retinopathy and Myopia in original dataset

Figure 2: Normal Fundus, Glaucoma, Diabetic retinopathy and Myopia in synthetic dataset

The size of all images should be the same (64x64), colored and png format. In machine learning and deep learning, accuracy of model will be affected if the samples are not equally distributed [8].

Table 1
Eye diseases samples in original dataset and synthetic dataset

Diseases	Original dataset		synthetic data using DCGAN
	Training	Validation	
Glaucoma	411	73	161
Diabetic retinopathy	459	81	133
Myopia	413	73	278
Normal	459	81	83
Total	1742	308	655

The small number of dataset lead to the overfit, due to lack of data, in previous paper we generated medical images using DCGAN. These images used with the original images to help us to increase our dataset in order to improve the accuracy of our model. We need to increase the quality of the generated images to achieve more accuracy. Therefore, in this paper, authors proposed convolutional denoising autoencoders to enhance the resolution of the generated images. In addition, using GMD model to classify eye diseases in order to compare the accuracy of the model using, original data only, original data with synthetic data which we obtained from DCGAN and original data with high resolution data which we obtained from (CDAE).

3.2.   Model building

The main goal in this paper is to enhance the resolution of fundus images which we obtained from previous paper using convolutional denoising autoencoders. In addition, using GMD model to classify eye diseases with, original data only, data generated from DCGAN with original data and high-resolution data with original data.

3.2.1. Convolutional Denoising Autoencoders (CDAE)

It is appropriate to start explaining the Autoencoder technology separately before starting to use this network in the process of image denoising. Autoencoder is an algorithm that belongs to unsupervised learning, this algorithm allows us to obtain efficient encoding. This algorithm consists of three sections:
- The coding section: which starts from the input layer and ends at the middle layer of the network. Layers in this section can be dense layer or convolutional layer.
- The middle-hidden layer: which is called the bottleneck hidden layer, meaning the neck layer. The layer in this section is flatten layer.
- The decoding section: which starts after the middle-hidden layer and ends at the end of the network. The number of layers in this section must be exactly equal to the number of layers in the coding section. The arrangement in terms of the type of layers and the number of units as well as the number of filters in each layer is opposite to what is in the coding layer. The following figure shows the architecture of this algorithm in its simplest types [9], [10].
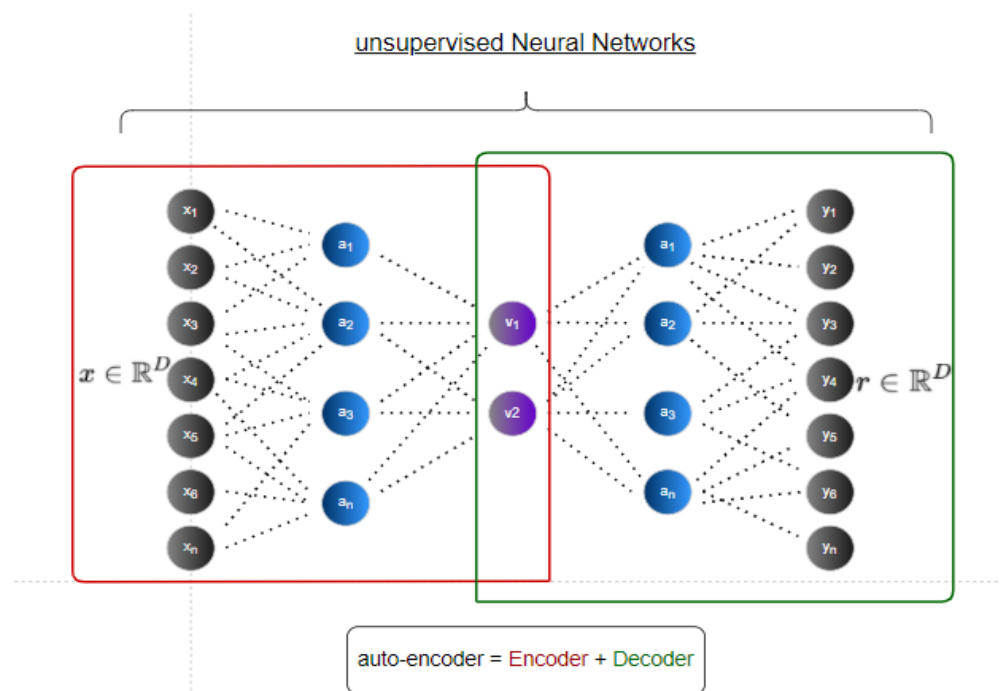

Figure 3: Denoising Autoencoder (DAE) architecture

The process of combining the encoder with the decoding is called an autoencoder. The network is trained on a specific goal of building the original image, the process of updating the weights is rely on the loss function, which is in our case is reconstruction error. The data in the middle layer (bottleneck layer) has few dimensions compared to the original image. This data is considered compressed data representing the original image, this original image can be recovered using a decoder. In briefly, Autoencoder used to reduce the dimension of the images.

The autoencoder algorithm can be used in the process of removing noise that may occur to some images due to certain conditions. The noise that occurs to the images can be in the following forms:

- Noise caused by errors or poor quality of image sensors such as cameras.
- Random differences in brightness or color between some pictures and the other images.
- Noise generated by the quantum noise process, such as the process of generating a small sized data set from a large data set.
- Noise occurs when using the JPEG compression algorithm.
- Noise generated when using the scanner.

Denoising using autoencoder is one of the techniques used in the field of image processing, it also called noise reduction. This technique, can be used in the training of an autoencoder-based model, where the training and learning process here is classified as unsupervised learning.

The input data in the training process are the images plus noise. The process of updating the weights in the training process depends on the loss function, which is the error value between the denoising and the original image.

After training the model, it can be used to remove noise from images and obtain images without noise. Denoising process can be illustrated using the Autoencoder algorithm, the following illustration shows the convolutional denoising autoencoder used in this paper:
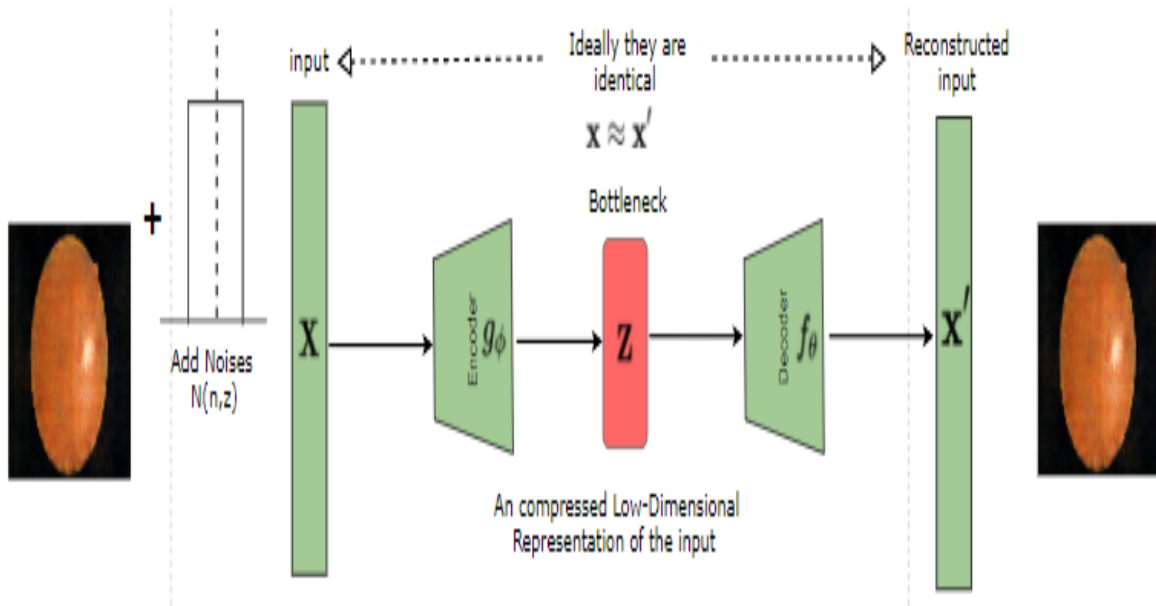

Figure 4: Convolutional denoising autoencoder (CDAE) that we used for denoising the eye diseases

Figure 4 shows, encoder which consists of two convolutional layers denoted by g_φ, bottleneck denoted by z, decoder also consists of two convolutional layers denoted by f_θ, and noises represented by N(n,z). The encoder feeds by an input vector of images with noises, and passes to the bottleneck as a compressed vector 'z', the decoder tries to reconstruct that input image from the compressed representation.

In our proposed CDAE, input image of size (64x 64) has to be flattened before feeding into a neural network. A flattened representation of this image would be (4096). This is passed into the encoder. The output of the encoder is then fed to the bottleneck or latent space which should be a reduced version, it simply means we have succeeded in compressing an image size. The decoder network tries to recreate the original input image (64 x 64) from the compressed state in the bottleneck.

As soon as the image is reconstructed, we compare the reconstructed image with the original image, and calculate the loss which can then be minimized.

Loss Function MSE [12], [13], [14]:

$$L(\theta, \Phi) = \frac{1}{n}\sum_{i=1}^{n}(x^i - f\theta(g\varphi(x^i)))^2 \qquad (1)$$

here we'll use sparsity DCD-AutoEncoder to make the activation function non-linearity

$$L_{sparse} = L_{vanilla} + \Omega(h) \qquad\qquad\qquad (2)$$

Where, $\Omega(h) = \sum_{j=1}^{n_h} KL(pj\backslash\backslash qj)_{Ber}$ $\qquad\qquad$ (3)

3.2.2. GMD model

Figure 5 shows, the input image is set to 64×64 pixels with 3 RGB channels. The features have been extracted from the images used, two convolution layers were used: the first is 256 filters of size 3×3 pixels and the second is 128 filters of size 3×3 pixels. For the pooling layer, a window of size 2×2 pixels were used, which compresses the original image size for further processing. After that, another three convolution layers were used of 128, 64, and 32 filters with size 3×3 pixels with a maximum pooling size 2×2 pixels. Then, Dense 100 units and 60 units were used as a fully connection, and output layer (4 units) to predict the eye cases. CNNs adjust their filter weights through backpropagation, which means that after the forward pass, the network is able to look at the loss function and make a backward pass to update the weights [16].

The GMD model performance was evaluated against training and validation datasets using the accuracy measure, which reflects the ability of the model to classify the whole database into four categories with almost no error. The following represents the accuracy measure equation [11]:

$$Accuracy = \quad TP / TP + FP + FN + TN \qquad\qquad (4)$$

Where, TP is True Positive: Your predicted positive and it is true, TN is True Negative: Your predicted negative and it is true, FP is False Positive: your predicted positive but it is false and False Negative FN: Your predicted negative but it is false.
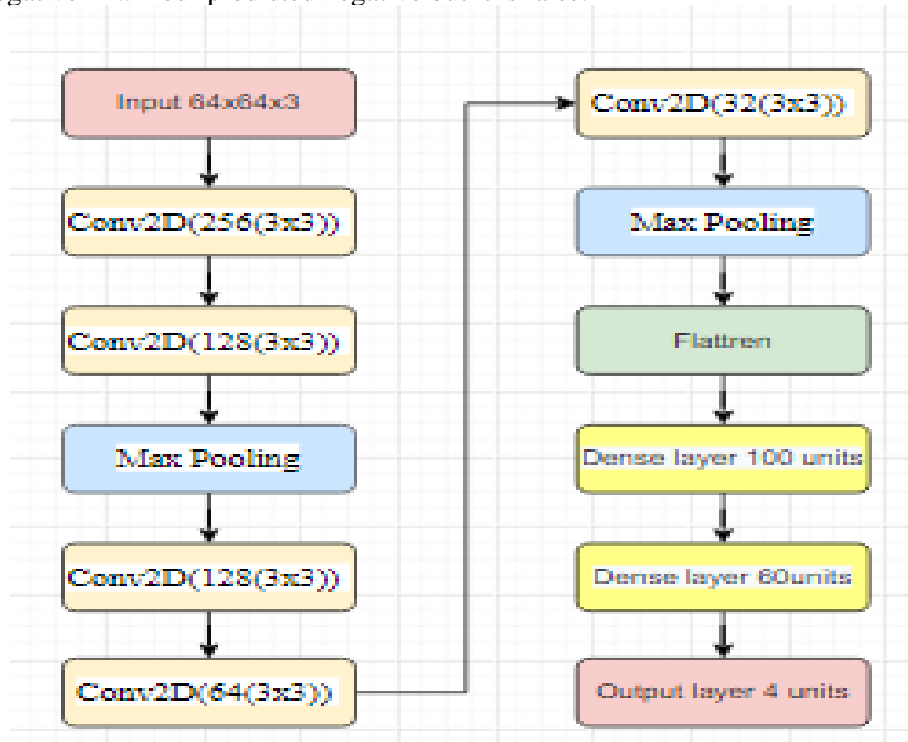

Figure 5: Block diagram of GMD model

4. Experiments and Results

CDAE and GMD models have been applied by python language using dual Graphics Processing Unit (GPU) in google colab. The main objective of this work is to increase the resolution of the medical images using DCAE. These images we have obtained from the previous paper in order to enhance the

accuracy of diagnostic of eye diseases. Our dataset which we obtained from the previous paper (DCGAN) has been explained in section 3.1. These data will be feeds to GMD model to classify eye diseases. Accuracy will be measured in three stages with, original dataset, original dataset with synthetic dataset which we obtained from previous paper and original dataset with high resolution images which we obtained from DCAE.

4.1.   DCAE results

The purpose of using CDAE is to increase the resolution of the medical images by denoising. The schema of our experiment as shown in Figure 4, has been applied. We utilized our fake (synthetic) dataset which we obtained from DCGAN (obtained from previous paper). This dataset consists of 161 images of Glaucoma, 133 images of Diabetic retinopathy, 278 images of Myopia and 83 images of Normal eye. All images were processed prior to modelling. all images were 64 x 64 and noise factor value 0.02. Relatively simple architecture was used for convolutional denoising autoencoder (CDAE), shown in Figure 6.
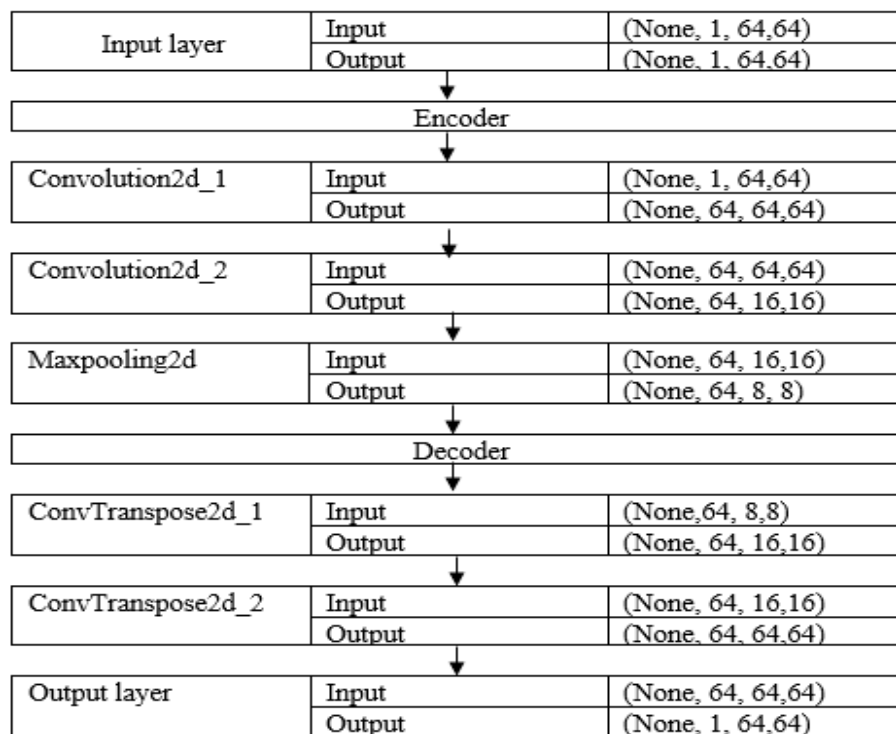
| Input layer | Input | (None, 1, 64,64) |
| | Output | (None, 1, 64,64) |

| Encoder | | |

| Convolution2d_1 | Input | (None, 1, 64,64) |
| | Output | (None, 64, 64,64) |

| Convolution2d_2 | Input | (None, 64, 64,64) |
| | Output | (None, 64, 16,16) |

| Maxpooling2d | Input | (None, 64, 16,16) |
| | Output | (None, 64, 8, 8) |

| Decoder | | |

| ConvTranspose2d_1 | Input | (None,64, 8,8) |
| | Output | (None, 64, 16,16) |

| ConvTranspose2d_2 | Input | (None, 64, 16,16) |
| | Output | (None, 64, 64,64) |

| Output layer | Input | (None, 64, 64,64) |
| | Output | (None, 1, 64,64) |

Figure 6: Architecture of the proposed CDAE

The proposed CDAE has been trained Consecutively on 4 classes provided in the actual dataset, Glaucoma, Myopia, Diabetic retinopathy and Normal images, and the result in test set as it shown in Figure. 7.
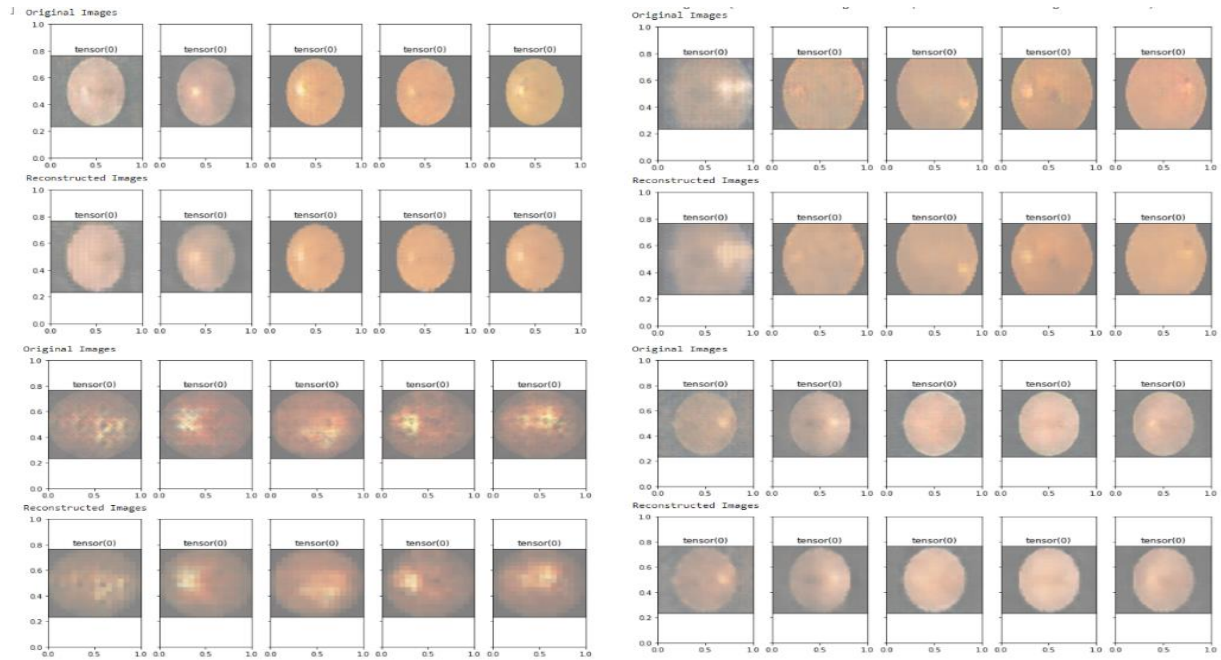
Figure 7: original images and Reconstructed Images on test set Glaucoma, Diabetic retinopathy, Myopia and Normal, Respectively.

The high-resolution images in each class have been saved in order to measure the accuracy of GMD model in three stages, original dataset only, synthetic dataset which we obtained from previous paper with original dataset and high-resolution images which we obtained from DCAE with original dataset.

4.2.   GMD model results using original dataset

GMD model has been applied utilized an adaptive learning rate optimization algorithm Adam, initial learning rate was set to 0.001 which showed a good performance to start. In Adam, the model was defined and the learning rate value was set to initial as follows.

```
# fit CNN model using Adam optimizer
epochs = 200
model = cnn_model()
```

Finally, compiling, fitting and plotting GMD model, and the results obtained as it shown in figure.8. The Accuracy on training set 89.84 % and accuracy on validation set 89.94%.

```
model.compile(loss=keras.losses.sparse_categorical_crossentropy,
optmizer=Adam(lr=0.001, beta_1=0.9, beta_2=0.999, epsilon=1e-
08, decay=0.0), metrics=['accuracy'])
history = model.fit(X_train, y_train,
                    validation_data=(X_val, y_val),
                    epochs=epochs,
                    batch_size=batch_size,
                    verbose=1)
```
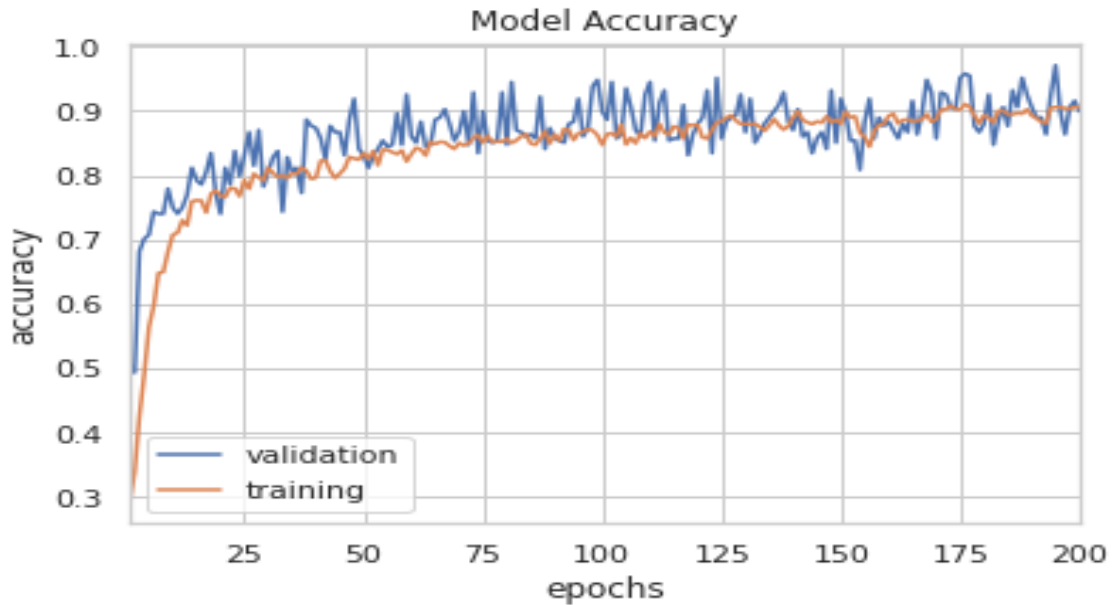
Figure 8: model accuracy with original dataset

4.3.   GMD model results using original dataset with synthetic dataset

By adding original dataset with synthetic dataset, the model has been applied utilized an adaptive learning rate optimization algorithm Adam, initial learning rate was set to 0.001 which showed a good performance to start. Then, compiling, fitting our model and plotting the model accuracy as shown in Figure.9. The result obtained, accuracy on training set 85.94% Accuracy on validation set 89.76 %.
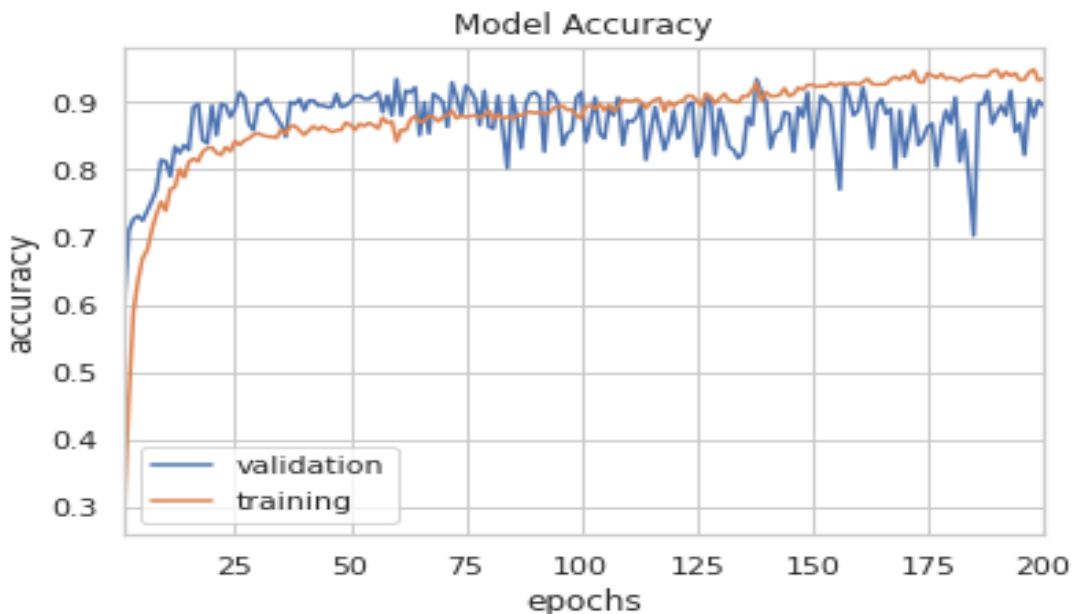

Figure 9: model accuracy using original dataset with synthetic dataset

4.4.   GMD model results using original dataset with high-resolution dataset (CDAE)

Authors used CDAE to enhance the image resolution which obtained from DCGAN (synthetic dataset), we utilized the same number of images in each class as mentioned in section 3.1. the model applied using an adaptive learning rate optimization algorithm Adam, initial learning rate was set to

0.001. Then, compiling, fitting our model and plotting the model accuracy as shown in Figure.10. The result obtained, accuracy on training set 92.82% Accuracy on validation set 93.32%.
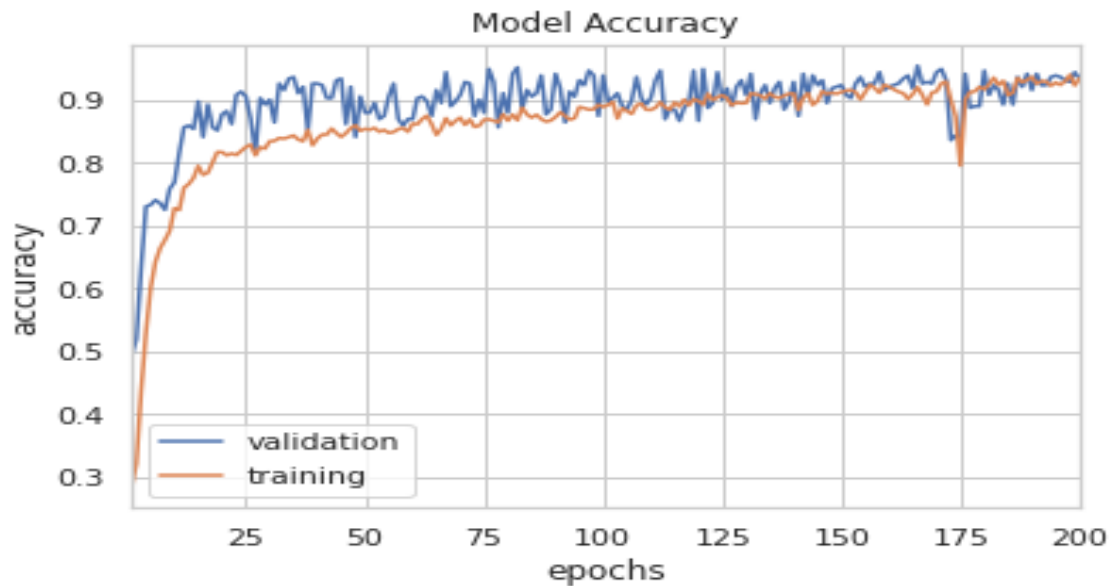

Figure 10: model accuracy using original dataset with high-resolution dataset (CDAE)

5. Conclusion

In this paper, a Convolutional Denoising Autoencoders, and GMD model based on Keras and TensorFlow has been deployed using Python on a dataset of four types of eye diseases: Glaucoma, Myopia, Diabetic retinopathy, and Normal eyes.

The applied CDAE algorithm and GMD model have been trained using Graphics Processing Unit (GPU) in Google Colab, in order to enhance our dataset which, we obtained from previous paper to improve eye diseases classification. In conclusion, CDAE algorithm can be used for enhance the medical images quality by eliminated Gaussian noise from the images. Furthermore, the performance of GMD model using original dataset, synthetic images with original dataset and high-resolution dataset (CDAE) with original data were compared. We confirmed that with the help of CDAE algorithm, GMD model can yield a better accuracy compared to using original dataset alone. Moreover, the accuracy of the model had improved significantly from 89.84 % in training set and 89.94% in validation set, to 92.82% in training set and 93.32% in validation set based on this work.

6. References

[1] Smaida, Mahmoud, and Serhii Yaroshchak. "Bagging of Convolutional Neural Networks for Diagnostic of Eye Diseases." *COLINS*. 2020.

[2] H. Greenspan, B. van Ginneken, and R. M. Summers, "Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique," IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1153–1159, May 2016.

[3] Gondara, Lovedeep. "Medical image denoising using convolutional denoising autoencoders." 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW). IEEE, 2016.

[4] Lee, Donghoon, Sunghoon Choi, and Hee-Joung Kim. "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 (2018): 97-104.

[5] Chiang, Hsin-Tien, et al. "Noise reduction in ECG signals using fully convolutional denoising autoencoders." IEEE Access 7 (2019): 60806-60813.

[6] Ali, Irfan, et al. "Image Denoising with Color Scheme by Using Autoencoders." IJCSNS 18.12 (2018): 158.

[7] Roy, Sudipta Singha, et al. "A robust system for noisy image classification combining denoising autoencoder and convolutional neural network." International Journal of Advanced Computer Science and Applications 9.1 (2018): 224-235.

[8] Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein Generative Adversarial Networks. International Conference on Machine Learning, ICML, 2017.

[9] Hasan Finjan, "Denoising using autoencoders", https://www.alfbaa01.com/2020/09/denoising-using-autoencoders.html, on March 17, 2021.

[10] Chris Nicholson, "Denoising autoencoders", https://wiki.pathmind.com/denoising-autoencoder, on March 17, 2021.

[11] Visa, Sofia, et al. "Confusion Matrix-based Feature Selection." MAICS 710 (2011): 120-127.

[12] Victor E. Irekponor , "Mathematical Prerequisites for Understanding Autoencoders and Variational Autoencoders", https://medium.com/analytics-vidhya/mathematical-prerequisites-for-understanding-autoencoders-and-variational-autoencoders-vaes-8f854025390e, on march 27, 2021.

[13] Jordan, Jeremy. "Introduction to autoencoders." Jeremy Jordan, Mar (2018).

[14] Yongchao Huang, "Kullback-Leibler (KL) Divergence and Jensen-Shannon Divergence", https://yongchaohuang.github.io/2020-07-08-kl-divergence/, on March 25, 2021.

[15] Smaida, Mahmoud, and Yaroshchak Serhii. "Comparative Study of Image Classification Algorithms for Eyes Diseases Diagnostic." International Journal of Innovative Science and Research Technology 4.12 (2019).

[16] Smaida, Mahmoud, Serhii Yaroshchak, and Ahmed Y. Ben Sasi. "Learning Rate Optimization in CNN for Accurate Ophthalmic Classification."

[17] Jonathan Hui, on Jun 18, 2018, "GAN-DCGAN (Deep convolutional generative adversarial networks)", https://jonathan-hui.medium.com/gan-dcgan-deep-convolutional-generative-adversarial-networks-df855c438f.

[18] Smaida, Mahmoud, and Serhii Yaroshchak. "Using Ensemble Learning for Diagnostics of Eye Diseases."

[19] Zhang, Qianjun, and Lei Zhang. "Convolutional adaptive denoising autoencoders for hierarchical feature extraction." Frontiers of Computer Science 12.6 (2018): 1140-1148.