Systems & Biomedical Engineering
Faculty of engineering
Cairo University

SBME Second Year-MTH2245

Eye Diseases Classification

Dr. Samah El-Tantawy

Website

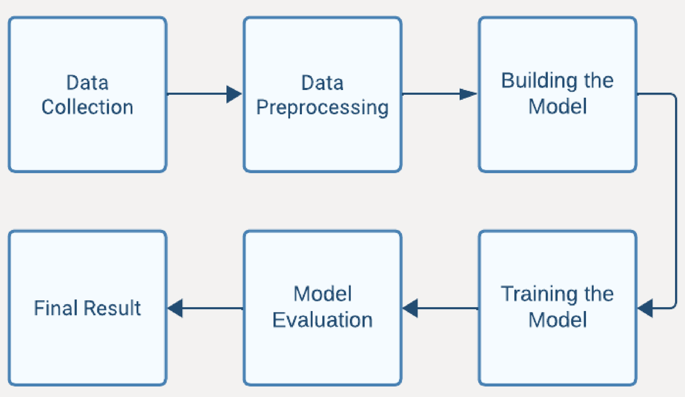
Abstract

- At least 2.2 billion people are suffering from near or distant vision impairment according to WHO. 1 billion of those cases required early intervention to hinder vision loss. The leading causes are cataracts (94 million), glaucoma (7.7 million), and diabetic retinopathy (3.9 million) which will be the focus of the study.
- This research centers on classifying the mentioned eye diseases through deep learning, employing convolutional neural networks (CNNs), and integrating partial differential equations (PDEs) into their development.
- Training and testing were done on a diverse, large-scale dataset of 4217 images, distributed among the 3 eye conditions plus the normal one, which improved the precision of detection reaching accuracy of 91.46% and loss of 0.3127

Problem Definition

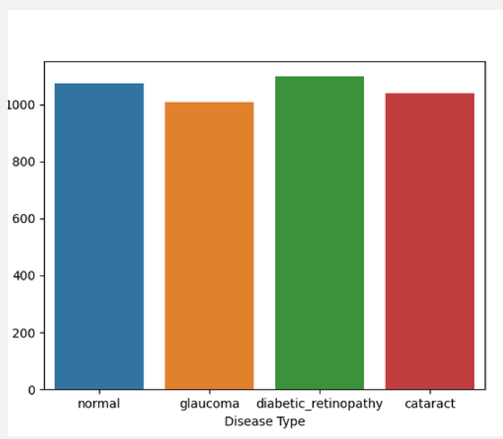
- The primary objective of this project is to leverage the power of deep learning and computer vision to build a system that enable early detection of eye diseases like glaucoma, cataract and diabetic retinopathy
- Classifying these diseases accurately using deep learning poses challenges due to variations in disease patterns and diverse eye images. The research aims to overcome these challenges by leveraging deep learning techniques to improve disease classification.

Data & Methods



Data Collection:
"Eye Disease Classification" dataset obtained from Kaggle

Dataset split into training (3374 images) and validation sets (843 images).



Data Preprocessing:
- Employed random transformations like rotation and zooming for data augmentation to mitigate overfitting.
- Preprocessing normalized pixel values to meet the model's architectural requirements.
- Image data generator seamlessly combines data augmentation and preprocessing during model training.

Building The Model:
we opted to use the pre-trained **VGG19** model due to its ability to understand intricate image features.

The architectural details of VGG19 are as follows:
- Input Size: (224 * 224) RGB images.
- Convolutional Layers: (3 * 3) kernels, stride of 1 pixel.
- Spatial Padding: Maintains image spatial resolution.
- Pooling: Max pooling (2*2) with a stride of 2 for downsampling.
- Activation Function: ReLU for non-linearity.
- Fully Connected Layers: Two layers of size 4096, followed by a 1000-channel layer for ILSVRC classification, and a softmax layer for final predictions.

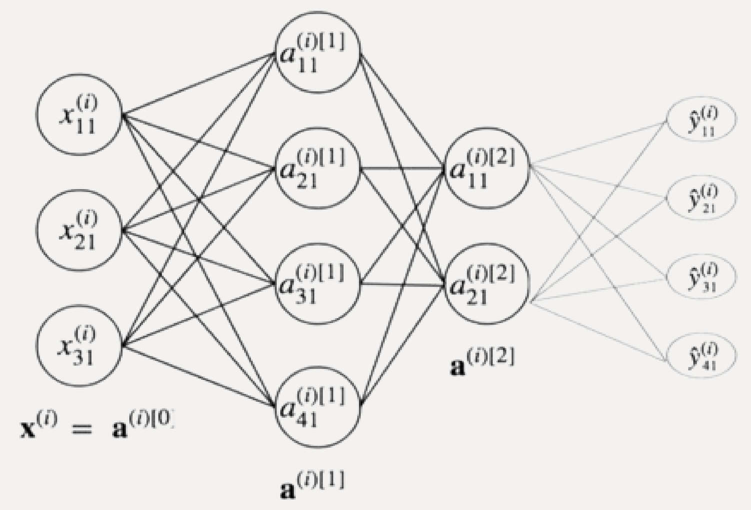
Training The Model:

Compile Parameters
- Utilizes Keras function for model configuration.
- Key arguments include optimizer, loss function, and metrics.
- 'Adam' optimizer chosen for adaptive learning rates.
- Categorical_crossentropy used for multi-class classification.

Fitting Parameters
- Employs the **fit()** function.
- Components include training data, batch size, epochs, validation data, verbosity, and callbacks.
- Early Stopping halts training if no improvement in validation accuracy after a certain number of epochs.

Analysis

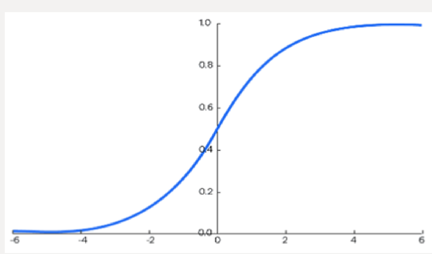
Delving into the realm of deep learning, our neural network operates through mathematical operations akin to partial differential equations (PDEs).



Forward Propagation:
The forward propagation process involves passing input data through the neural network to obtain predictions. For each layer $a^{(i)}$, the output $\hat{y}^{(i)}$ is computed as:

$$z^{(i)} = x^{(i)}w + b$$
$$a^{(i)[j]} = g(z^{(i)[j]})$$
$$\hat{y}^{(i)} = soft(z^{(i)[j]})$$

Softmax activation function



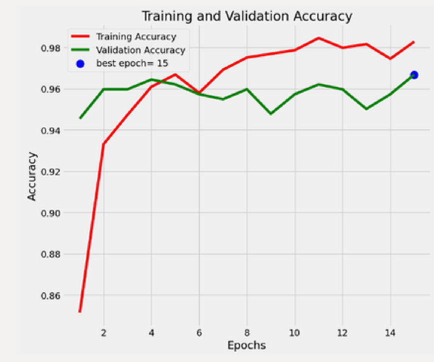
$$\frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}}$$

Backward Propagation:
Backward propagation involves computing the gradients of the loss concerning the parameters, enabling parameter updates during training. The key equations for each layer $a^{(i)}$ are:

- $$\frac{\partial E(X,\theta)}{\partial w_{ij}^k} = \frac{1}{N}\sum_{d=1}^{N}\frac{\partial\left(\frac{1}{2}(\hat{y}_d - y_d)^2\right)}{\partial w_{ij}^k} = \frac{1}{N}\sum_{d=1}^{N}\frac{\partial E_d}{\partial w_{ij}^k}$$ **Overall Gradient Calculation**
- $$\frac{\partial E_d}{\partial w_{ij}^k} = \delta_j^k o_i^{k-1}$$ **Error Gradient concerning Weight**
- $$\delta_j^m = (\hat{y} - y)\cdot g_o'(a_j^m)$$ **Output Layer Error Term**
- $$\delta_j^k = g'(a_j^k)\sum_{l=1}^{r^{k+1}} \delta_l^{k+1} w_{jl}^{k+1}$$ **Hidden Layer Error Term**
- $$\Delta w_{ij}^k = -\alpha\frac{\partial E(X,\theta)}{\partial w_{ij}^k}$$ **Weight Update Calculation**

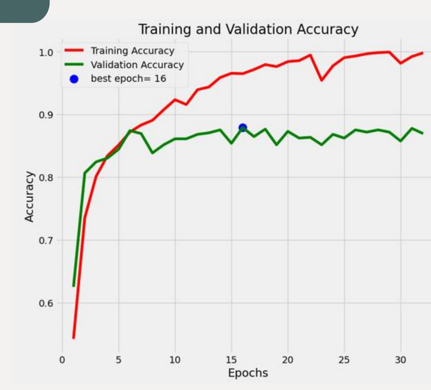
Test & Results

1 Cataract Detection Model



Accuracy: 96.68%
Loss: 0.4733

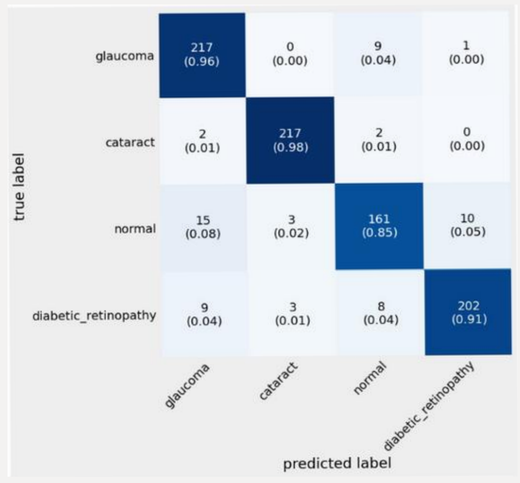
2 CNN Model



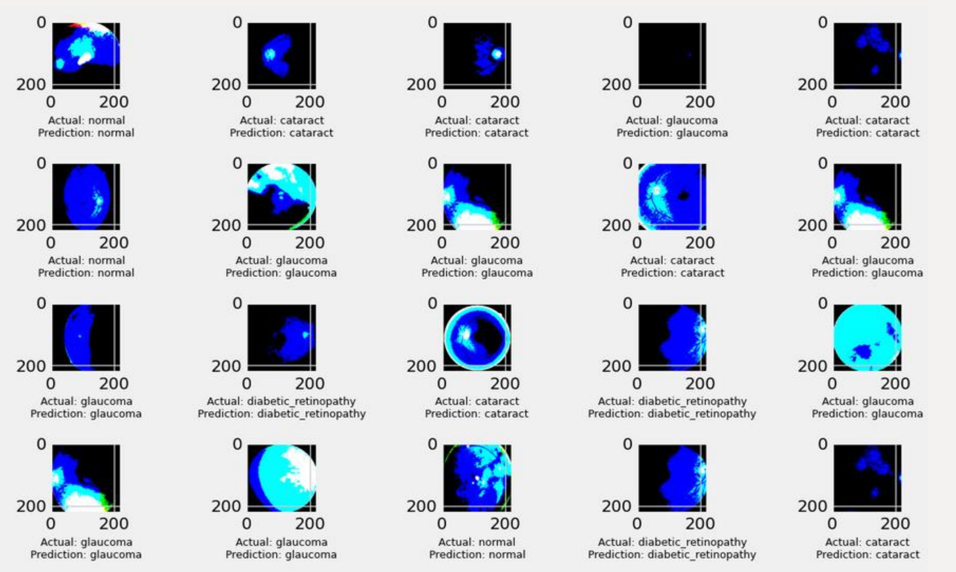
Accuracy: 86.95%
Loss: 0.9638

3 Final Model With VGG 19

Using VGG 19, The model exhibited highly satisfying results, reaching a minimum validation loss of 0.3127 and an accuracy of 91.46% at its best epoch.



Model Evaluation



Conclusion

- In Conclusion, This research tries to offer an instant solution for the categorization of various eye diseases by extracting and analyzing features from retinal fundus images for glaucoma, diabetic retinopathy, cataracts, and normal conditions.
- On implementing a Convolutional Neural Network (CNN) for detection and classification of the previously mentioned four eye conditions first by using CNN, it reached an accuracy of 86.95% and loss of 0.9638 then by using the VGG19 model reaching an accuracy of 91.46% and minimum validation loss of 0.3127 at best epoch.

Future Work

While the current results are promising, there are several avenues for improvement to achieve even better accuracy and validation performance. The following points could be taken into consideration:
- Broadening Dataset Diversity
- Advanced-Data Augmentation Techniques

References

@eng-st.cu.edu.eg

Ahmed Raafat - **Ahmed** Amgad - **Salah** Mohamed - **Somaia** Ahmed - **Ayat** Tarek - **Abdullah** Mahmoud - **Hasnaa** Hossam - **Mohamed** Ahmed

ahmed.abdelfarage03 ahmed.elsharkawy03 salah.ismael03 somaia.fakhreldin03 ayat.al-ruby02 abdallah.abdelghaphar03 Hassnaa.hussein03 mohamed.hassan034