Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 7- Issue 1, January 2024
Paper : 87

Deep learning-based recognition of ocular disease in Fundus images.

Sridharan K [1], Vettri Chezhian P [2], Nithish Kumar S [3] and Raj Kumar K [4]
Department of Information Technology, Panimalar Engineering College, Chennai, India
[1]sridharank.p@gmail.com, [2]vettrichezhian217@gmail.com, [3]snithish207@gmail.com, [4]raju313703@gmail.com

Abstract-- This system aims in developing a deep learning model which helps in recognition of ocular diseases present in the fundus images, specifically cataract, diabetic retinopathy and glaucoma. The system utilizes a dataset of fundus images that are labeled with disease categories which are used for training the convolutional neural network (CNN) model. Here, the CNN model contains 4 convolutional layers, 4 max-pooling layers, and 2 fully connected layers, with the rectified linear unit (ReLU) activation are used for the convolutional layers in CNN. Here, Image augmentation purpose is to increase the size of the dataset and also reducing overfitting. System model is trained and validated using the training and testing datasets, respectively, with early stopping used to prevent overfitting. The performance of this system is evaluated using metrics such as F1 score, recall, precision and accuracy. Thus, results shows that the system model achieves a promising level of high accuracy for recognizing the three ocular diseases in the fundus images, demonstrating the deep learning models has high potential to aid in the diagnosis of ocular diseases.

Keywords- Cataract, Computer vision, Centralized Web-Based System, Deep learning, Diabetic retinopathy, Fundus images, Medical image analysis.

I. INTRODUCTION

The system aims to develop a web-based application for diagnosing common eye diseases using deep learning models. The application can detect three common eye diseases: glaucoma, cataract, and diabetic retinopathy. The deep learning models are trained using fundus images of the eye, which are taken by an ophthalmologist. The system uses the Flask web framework to develop a user interface for the application. Users can upload fundus images of the eye and select the type of disease they suspect. The deep learning model corresponding to the selected disease is used in analysing the uploaded image and make a prediction about the disease rate of occurrence. The system is trained in predicting for likelihood of disease occurrence in the image, which is presented to the user as a percentage. The models have been trained using the Keras deep learning library and have been saved in the H5 format for easy loading and use in the application. The user interface has been designed in the form of web based application using HTML and Bootstrap, while the back-end logic has been implemented in Python using the Flask web framework. In this system, CNNs are used to predict three common eye diseases (glaucoma, cataract, and diabetic retinopathy) based on fundus images of the eye. The models are trained on a dataset of thousands of labelled fundus images using a transfer learning approach, here the fine-tuned pre-trained model used the task of classifying eye diseases. The resulting models are then deployed in a Flask web application to allow users to upload their own fundus images and receive predictions of whether they are likely to have one of the three eye diseases.

The system utilizes CNN to analyze fundus images and classify diseases such as glaucoma, diabetic retinopathy, and cataract [1,2]. Deep learning models are trained using fundus images taken by ophthalmologists, with Flask web framework used to develop the user interface [3]. Users can upload fundus images and select a disease, with the corresponding deep learning model analyzing the image and providing a percentage likelihood of disease occurrence. The system uses several metrics to evaluate algorithm performance and compares it to state-of-the-art methods [5,6,7]. The models have been saved in H5 format and implemented in Python using Flask web framework and HTML/Bootstrap for the application design. CNNs are used to predict glaucoma, cataract, and diabetic retinopathy based on fundus images, with high accuracy and specificity demonstrated in comparison to human experts [7].

The models are trained on a large labelled fundus image dataset using transfer learning, with a pre-trained model fine-tuned for classifying eye diseases. The resulting models are integrated into a Flask web

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)
Volume 7- Issue 1, January 2024
Paper : 87

application where users can upload fundus images to receive predictions about their likelihood of having one of the three eye diseases [8,9]. The system has the potential to improve diagnosis and treatment of retinal diseases by enabling more accurate and efficient screening of retinal lesions. This system can be particularly useful for individuals who are at risk of developing these diseases or those who suspect they may be experiencing symptoms [9]. Early detection can lead to prompt intervention, helping to prevent disease progression and improve overall eye health.

II. BACKGROUND STUDY

The proposed system for this process is an automated ocular disease recognition system using deep learning. The system takes fundus images for input and uses the trained convolutional neural network (CNN) for classifying them as either normal or diseased with one of three specific diseases: glaucoma, cataract, or diabetic retinopathy [10, 11, 12]. The system provides an accurate and efficient means of diagnosing ocular diseases, which can help to improve patient outcomes and reduce healthcare costs. This system uses Convolutional Neural Networks (CNNs) one of the deep learning techniques, to recognize ocular diseases [13, 14, 15].

A. Proposed System Architecture

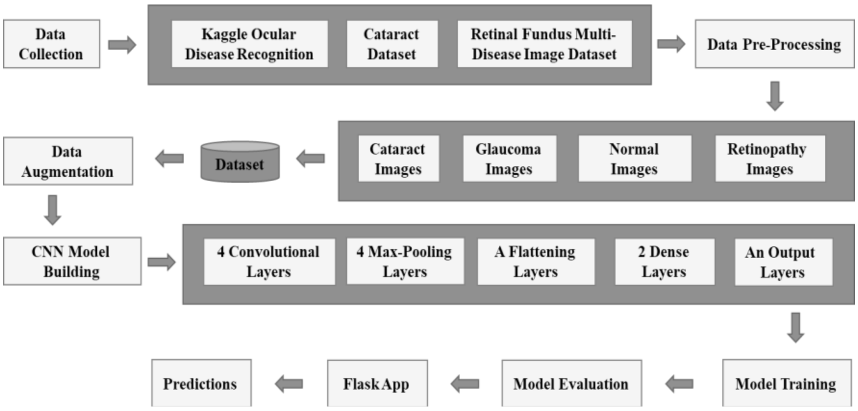

Fig. 1. System Architecture

Ocular Disease Recognition system architecture consists of three main components, namely Data Collection, CNN Model Development and Evaluation, and Web App Development. The Data Collection module collects fundus images of the eye from various sources, including online repositories and hospitals. The CNN Model Development and Evaluation module consists of several sub-modules, including Data Pre-Processing and Image Augmentation. The Data Pre-Processing module preprocesses the fundus images by resizing, normalizing, and applying various filters to enhance the image quality. The Image Augmentation module generates new images from existing images by applying random transformations, such as rotation, flipping, and zooming. The CNN Model Development and Evaluation module trains a CNN model on the preprocessed and augmented images and evaluates its performance on a separate test set. Finally, the Web App Development module deploys the trained CNN model to a web application that allows users to upload fundus images and receive predictions on the presence of ocular diseases.

III. IMPLEMENTATION

The implementation of the web-based application for diagnosing common eye diseases using deep learning models involves the following steps:

A. Data Collection Module

The Data Collection Module of this system involves several steps. First, the scope of the required data is defined, which in this case is fundus images of the eye for detecting ocular diseases. Next, sources for the dataset are identified, such as online repositories, medical institutions, and hospitals. The actual data collection process then takes place, with the dataset being large enough to train the deep learning model. Once collected, the data is verified to ensure its high quality and relevance, followed by cleaning to remove unwanted noise or data. Each image in the dataset is then labeled with the appropriate class label (glaucoma, cataract, or diabetic retinopathy). Finally, the dataset is split into training and validation sets in a specific ratio. By following these steps, the Data Collection Module ensures that the required data is acquired, verified, and prepared for training the deep learning model effectively.

Table 1. Data Collection Module Table

Dataset	Description
Kaggle Ocular Disease Recognition	1560 images mixed
Cataract Dataset	602 images mixed
Retinal fundus multi-disease image dataset	1780 images mixed

B. Data Pre-Processing Module

The Data Pre-Processing Module in this system imports essential libraries, including NumPy, Pandas, OpenCV, Matplotlib, Scikit-learn, Keras, and TensorFlow, and sets a seed for reproducibility. Images are then loaded, read, resized, and converted into arrays. To normalize pixel values, each value is divided by 255 to achieve a range between 0 and 1, improving convergence during neural network training. These pixel values are then scaled to the same range by dividing the NumPy array by 255. Images are separated into lists based on class labels, then transformed into a single NumPy array for both features and labels. By following these steps, the Data Pre-Processing Module effectively loads and pre-processes images for deep learning model training.

C. Image Augmentation Module

The Image Augmentation Module in your system involves several techniques to create new variations of the original images. The first step is random rotation, where the image is rotated by a random angle within a specified range to increase its robustness to misaligned images. The second step is random width and height shifts, where the image is randomly shifted horizontally and vertically within a certain range of pixels while keeping the content intact. The third step is random brightness, where in each pixel of the image a random brightness factor to lighten or darken it. The fourth step is a random horizontal flip, where the image flips horizontally with 0.5 as its probability, creating new training images to improve the model's ability to recognize objects in different orientations. The fifth and final step is random zoom, which simulates getting closer to or further away from the subject while taking an image, making the model more robust to images captured from different distances or with varying levels of zoom. Together, these techniques help the model generalize better and perform well on a wide variety of images.

D. CNN Model Development and Evaluation:

The CNN model development and evaluation process involves several steps. The input layer takes in 512 x 512 pixel images with 3 color channels. The first convolutional layer applies 16 filters of size 3 x 3 to detect image features, producing an output size of 51 x 510 x 16. This output is then processed by the first max pooling layer, which reduces its size to 255 x 255 x 16. The process is repeated with the second convolutional layer, applying 32 filters of size 3 x 3, and the second max pooling layer, reducing the output to 126 x 126 x 32. Next, the third convolutional layer applies 64 filters of size 3 x 3, producing an output size of 124 x 124 x 64, which is then processed by the third and fourth max pooling layers, resulting in an output size of 30 x 30 x 128. The output is then flattened into a 1D array by the Flatten layer, producing an output size of 115200. The first Dense layer applies 256 neurons with ReLU activation function, and the Dropout layer prevents overfitting. The output size is then reduced to 256. Finally, Dense layer 2 applies 1 neuron with sigmoid activation function to classify whether or not the disease is present.

E. Web App Development Module:

The Web App Development Module of the system has three main stages. First, Frontend Development creates a user-friendly interface for uploading fundus images and receiving predictions. It uses HTML, CSS, and JavaScript technologies. Second, Backend Development creates a web application using Python and Flask technologies to accept user input, pass it to the CNN model for prediction, and return the results to the frontend. Finally, the Deployment stage involves configuring the server environment, installing dependencies, and starting the web application for users to access. This enables the web application to receive fundus images and predict the presence or absence of disease using the CNN model, displaying the results on the user interface.

IV. REFERENCES

1. T. Guergueb and M. A. Akhloufi, "Ocular diseases detection using recent deep learning techniques," in Proceedings of the Annual Int. Conf. IEEE Eng Med Biol Soc, pp. 3336–3339, IEEE, Mexico, 1 November 2021.
2. Application of Ocular Fundus Photography and Angiography. (2014). Ophthalmological Imaging and Applications, 154–175. https://doi.org/10.1201/b17026-12%20.
3. Y. Elloumi, M. Akil, and H. Boudegga, "Ocular diseases diagnosis in fundus images using a deep learning: approaches, tools and performance evaluation," in Proc. Real-Time Image Processing and Deep learning. 109960T, Maryland, USA,2019.
4. Borgen, P. Bours, and S. D. Wolthusen, "Simulating the Influences of Aging and Ocular Disease on Biometric Recognition Performance," International Conference on Biometrics 2009, LNCS 5558, vol. 8(8),pp. 857–867, 2009.
5. Li, N., Li, T., Hu, C., Wang, K., & Kang, H. (2021). A Benchmark of Ocular Disease Intelligent Recognition: One Shot for Multi-disease Detection. Benchmarking, Measuring, and Optimizing, 177–193. https://doi.org/10.1007/978-3-030-71058-3_11.
6. Miranda, E., Aryuni, M., & Irwansyah, E. (2016, November). A survey of medical image classification techniques. In 2016 International Conference on Information Management and Technology (ICIMTech) (pp. 56-61). IEEE.
7. Li, Z., He, Y., Keel, S., Meng, W., Chang, R. T., He, M. (2018). Efficacy of a Deep learning System for Detecting Glaucomatous Optic Neu-ropathy Based on Color Fundus Photographs. Ophthalmology, 125(8), 1199–1206. https://doi.org/10.1016/j.ophtha.2018.01.023.
8. Playout, C., Duval, R., Cheriet, F. (2019). A Novel Weakly Supervised Multitask Architecture for Retinal Lesions Segmentation on Fundus.
9. K. M. Noaman, A. Arafat, and I. A. Alqubati," Diagnosis of Poor Eyesight based on Support Vector Machine and Artificial Neural Networks," Journal of Emerging Trends in Computing and Information Sciences, 2014.
10. MD Abr`amoff, MK Garvin, M Sonka, "Retinal imaging and image analysis," IEEE reviews in biomedical engineering, vol. 3, pp. 169-208, 2014.
11. Carson Lam, Darvin Yi, Margaret Guo, and Tony Lindsey, "Automated detection of diabetic retinopathy using deep learning," AMIA Summits on Translational Science Proceedings 2018, pp. 147-155, 2018.
12. Fu H., Cheng J., Xu Y., Zhang C., Wong D. W. K., Liu J., Cao X., "Disc-Aware Ensemble Network for Glaucoma Screening from Fundus Image," IEEE Trans. Med. Imaging 37(11), pp. 2493-2501, 2018.
13. Tan, J.H., Bhandary, S.V., Sivaprasad, S., Hagiwara, Y., Bagchi, A., Raghavendra, U., Rao, A.K., Raju, B., Shetty, N.S., Gertych, A., et al.: Age-related macular degeneration detection using deep convolutional neural network. Future Generation Computer Systems 87, 127–135 (2018)
14. Decenci`ere, E., Cazuguel, G., Zhang, X., Thibault, G., Klein, J.C., Meyer, F., Marcotegui, B., Quellec, G., Lamard, M., Danno, R., et al.: Teleophta: Machine learning and image processing methods for teleophthalmology. Irbm 34(2), 196–203 (2013)
15. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM international conference on Multimedia. pp. 675–678 (2014)