# Employing Neural Networks for Early Detection of Ocular Diseases

DS 6050: Deep Learning

Ana Daley (amd2yc), Evan Mitchell (etm8fs), Cecily Wolfe (cew4pf)

## Introduction

The World Health Organization (WHO) estimates that in 2021, over two billion people were experiencing some form of visual impairment, and over one billion of these cases were preventable or had yet to be treated.[1] Diagnosing and detecting potential visual impairment early can be done using visual acuity tests, retinal exams, or ocular tonometry; these methods are effective, but time-consuming and frustrating for patients. This has motivated the desire to develop new methods that will be able to diagnose ocular diseases more efficiently and during the initial stages of the disease progression, allowing for early treatment. Fundus photography is one such method that has been utilized as an alternative to traditional diagnosis techniques, though it is challenging for humans to detect visual impairment symptoms solely from images without the use of advanced technology. Neural networks can mitigate these issues, using advanced classification algorithms, and enable early detection and correction.

## Hypothesis

As we will discuss in the next section, a variety of individual neural networks have been able to successfully identify a range of ocular diseases. We propose that ensembling these individual models, either through simple or weighted averaging, will have a positive impact on their individual accuracies and be able to outperform any single model.

In addition, we plan to develop a customized neural network. When comparing established neural network frameworks to a custom network of our own design, we expect the transfer learning models to outperform the custom model.

## Literature Review

Various research teams have implemented fundus image classifiers using neural networks, with many looking to classify images with at least one of the following diagnoses: normal, diabetes, glaucoma, cataract, age-related macular degeneration, hypertension, myopia, and other diseases/abnormalities. Jordi et al. (2019) treated the problem as a multi-class classification task and compared neural networks with VGG16 and Inception frameworks, observing that the VGG16 model was more accurate.[2] Gour et al. (2020) also found that the VGG16 model was preferable when comparing it to ResNet, InceptionV3, and MobileNet models.[3] Li et al. (2020) tested a Dense Correlation Network with a ResNet Convolutional Neural Network, a Spatial Correlation Module, and a classifier.[4] Wang et al. (2020) implemented a pre-trained EfficientNet and an ensemble of weak classifiers for a dataset of color fundus images and rescaled gray-scale images, finding that it outperforms all of the aforementioned frameworks on both datasets.[5] More recently, He et al. (2021) implemented a “knowledge distillation-based optimization strategy” to transfer learning from a complex teacher model to a lightweight student model using fundus images tagged with clinical data such as patient age and sex and diagnostic key words.[6] The ideal approach was a R-CNN+LSTM (long short-term memory) with feature selection algorithm NCAR (Neighborhood Components Analysis - ReliefF)

by Demir et. al.[7] This model matched or exceeded the performance of each other model using the AUC, F-score, and accuracy metrics.

**Methods and Data Sources**

The collection of fundus images used for both this project and the literature referenced in the previous section was sourced from the Ocular Disease Intelligent Recognition (ODIR) dataset on Kaggle.[8] The 10,000 color fundus images from 5,000 patients were provided by Peking University and Shanggong Medical Technology Co. Ltd. and show retinal tissue taken with various camera brands, including Canon, Zeis, and Kowa. Pictures of the right and left eyes of patients are labeled with at least one of eight diagnoses – normal (1140 cases), diabetes (1128), glaucoma (215), cataract (212), age-related macular degeneration (164), hypertension (103), myopia (174), and other diseases/abnormalities (979) – as well as the age and sex of patients.[4,8] However, only 7,000 images were labeled in the provided data, so we only used those images for training and evaluation.

Before developing models, we organized the data associated with each patient so that we had the correct classifications for each of the images. Since all of the information for each patient was contained in one row of the data – the right and left image names, the diagnostic keywords, for each image, and a set of dummy variable for the eight conditions, containing "1" if the condition was present, "0" otherwise – we had to parse the data so that each row was associated with a given image and the correct diagnostic keywords and diagnoses.

This task required knowledge of the diagnostic keywords particular to each condition. In total, we found 102 key terms, consistent with the analysis by He et al.,[6] which we used to assign diagnostic labels to each image and classify them based on a set of diagnoses. In other words, instead of implementing multiclass classification, we assigned images with unique combinations of diagnoses labels to separate classes. Therefore, our models classified images into one of 39 classes for the predicted label, for instance, class 0, or "N", for normal; class 1, or "D", for diabetes; class 26, or "A" and "O", for age-related macular degeneration and (an)other disease/abnormality (Figures 1, 2).

| | ID | Patient Age | Patient Sex | Left-Fundus | Right-Fundus | Left-Diagnostic Keywords | Right-Diagnostic Keywords | N | D | G | C | A | H | M | O |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 69 | Female | 0_left.jpg | 0_right.jpg | cataract | normal fundus | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 57 | Male | 1_left.jpg | 1_right.jpg | normal fundus | normal fundus | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Figure 1: Two Records (i.e., two patients) in the Original Dataframe Prior to Preprocessing**

| | ID | Image | conditions | N | D | G | C | A | H | M | O | new_label | rank | label_ids | prefix | suffix | new_name |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0_left.jpg | cataract | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 7 | 4 | 0 | 0_left | jpg | 0_left_4_0.jpg |
| 1 | 0 | 0_right.jpg | normal fundus | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0_right | jpg | 0_right_0_0.jpg |
| 2 | 1 | 1_left.jpg | normal fundus | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1_left | jpg | 1_left_0_0.jpg |
| 3 | 1 | 1_right.jpg | normal fundus | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1_right | jpg | 1_right_0_0.jpg |

**Figure 2: Four Records (i.e., four images for each eye of two patients) in the Preprocessed Dataframe**

The imbalance in the original classes carried over to this new classification system, and in fact was exacerbated, due to the specific nature of the 39 new classes. To remedy this issue, we partially balanced the data by randomly oversampling with replacement classes that had fewer than the mean number of images per class (~180 images on average) such that those classes had at least that many images. (As a result, some images were repeated multiple times, a shortcoming we attempted to remedy with data augmentation when developing our neural

networks). While the data was still imbalanced, we elected not to downsample larger classes, as we wanted to preserve all of the original data (Figure 3).

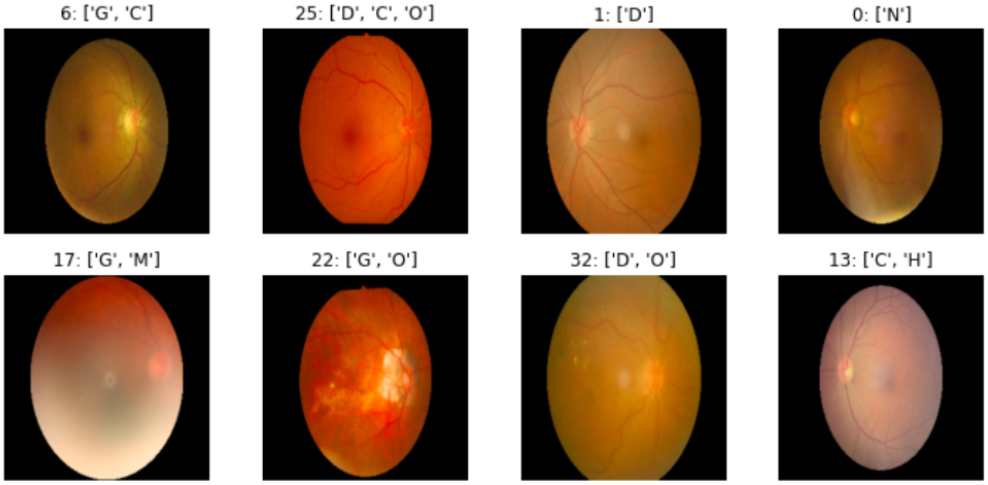


**Figure 3: Sample of Images with New Numeric Class Labels and Corresponding Diagnoses**

Ultimately, our final preprocessed dataset contained 12,370 images divided into 39 classes, with the largest class (label 0, or “N” for normal) with 2818 images, and the smallest classes with 180 images. We separated this dataset using an 80-20 split, placing 9896 images in the training set and 2474 in the validation set.

## Models

We compared a variety of models. We trained MobileNetV2, InceptionNetV3, VGG16, ResNet50V2, and EfficientNetV2-B3 Convolutional Neural Network (CNN) frameworks with transfer learning. Transfer learning is an approach that uses pre-trained models as a starting point; the initial weights and layers are then tuned in order to develop models that can accomplish specific tasks such as ocular disease recognition. For this project, we used weights learned from the ImageNet dataset to initialize the models.

We also evaluated an ensemble model using each of the transfer learning models. Initially, we allotted equal weights to each of the models, averaged the softmax predictions, and used those averaged predictions to make a final classification. We then tuned the model weights by evaluating different combinations on the validation set and retained the weights that resulted in the highest accuracy within the available time. (All of the models were trained using slurm scripts due to the length of time required to train them – ranging from several hours to nearly a day – meaning the time and computing resources allocated to them were limited.)

Finally, we created a custom Convolutional Neural Network (CNN) that begins by rescaling the data from the range [0, 255] to [0, 1] and augmenting the data by randomly flipping, rotating, and translating images in the training set. A series of 2D convolution layers interspersed with max pooling layers comes next, and is followed by a flattening layer and several densely connected and regularization (batch normalization, dropout) layers prior to a softmax classification layer (Figure 4).

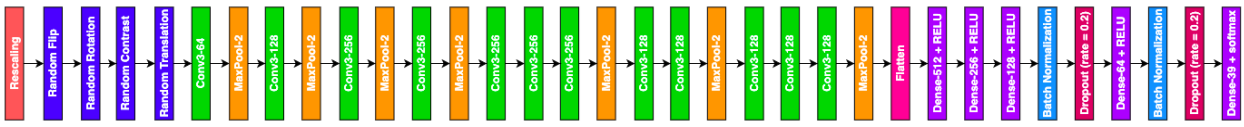


**Figure 4: Custom Convolution Neural Network (CNN) architecture**

During the training process for each model, we used an Adam optimizer to minimize sparse categorical cross-entropy loss and an initial learning rate of 0.001 (except for the custom CNN, which had a base learning rate of 0.0001). We also included class weights to place more emphasis on smaller classes. We adapted an equation to calculate class weights provided in the TensorFlow tutorials created by researchers at the Google Brain Team[9] for our multiclass classification problem as follows:

$$w_k = \frac{1}{n_k} * \frac{N}{K}$$

where $w_k$ is the weight for class $k$; $n_k$ is the number of images in class $k$; $N$ is the total number of images; and $K$ is the total number of classes. Therefore, classes with fewer images have larger weights.

**Analysis and Interpretations**

In terms of individual models, EfficientNetV2 achieved the highest validation accuracy (~77%), and in the smallest number of epochs, while InceptionNetV3 reported the lowest loss (0.9922). On the other hand, EfficientNetV2 had the highest sparse categorical cross-entropy loss (2.0642), while VGG16 performed the worst (~60% validation accuracy), even after training for a greater number of epochs.

The metrics for custom CNN, surprisingly, are comparable to a number of the transfer learning models: a loss of 1.2555, lower than any other individual model besides InceptionNetV3, and a validation accuracy of nearly 70%, similar to InceptionNetV3 and ResNet50V2 and better than VGG16. However, the required number of epochs to attain this validation accuracy far outstripped any other individual model (400 compared to a maximum of 270 for the transfer learning models). (A validation accuracy of greater than 60% was achieved within 176 epochs, but the values continued to oscillate during training.) These results may be due in part to the larger architecture of the custom CNN, which contains a number of trainable parameters (~3.86 million compared to MobileNetV2's ~1.91 million), but also to its relative simplicity, since it does not incorporate more advanced features, such as inception modules or skip connections, that might better capture the complexity of the data.

When considering all of the models, the ensemble models performed the best with regards to both sparse categorical cross-entropy loss and validation accuracy. Ensemble models that place the most weight on InceptionNetV3 and EfficientNetV2, the model with the lowest loss and the model with the highest validation accuracy, respectively, and contain a combination of all of the transfer learning models outperformed ensemble models composed of only a subset of the transfer learning models. Although the number of combinations tested for the weighted average of model predictions was limited by time and computing constraints, the weights that maximized the validation accuracy and minimized the loss were $\frac{1}{9}$ each for MobileNetV2, VGG16, and ResNet50V2, and $\frac{1}{3}$ each for InceptionNetV3 and EfficientNetV2. When adjusting the weights manually after the automated evaluation process, increasing the weight for InceptionNetV3 and EfficientNetV2 to $\frac{1}{3}$ for each (and decreasing the weights for the other transfer learning models to $\frac{1}{81}$ for each) slightly improved model performance. This pattern only held true up to a certain point, for ensemble models with even larger weights on these two transfer learning models ($\frac{120}{243}$ for each of the best performing models and $\frac{1}{243}$ for the

others, or $\frac{1}{2}$ for the best performing models and $0$ for the others) had higher loss and lower validation accuracy.

| **Model** | **Loss** | **Validation Accuracy** | **Number of Epochs** |
|---|---|---|---|
| MobileNetV2 | 1.4334 | 0.7352 | 120 |
| InceptionNetV3 | 0.9922 | 0.7017 | 90 |
| VGG16 | 1.3468 | 0.5982 | 270 |
| ResNet50V2 | 1.3820 | 0.7061 | 90 |
| EfficientNetV2 | 2.0642 | 0.7676 | 90 |
| Overall Best Ensemble Model<br>• $\frac{1}{81}$ * MobileNetV2<br>• $\frac{39}{81}$ * InceptionNetV3<br>• $\frac{1}{81}$ * VGG16<br>• $\frac{1}{81}$ * ResNet50V2<br>• $\frac{39}{81}$ * EfficientNetV2 | 0.7726 | 0.7773 | 660*<br>*Sum of individual model epochs |
| Best Ensemble Model from automated testing<br>• $\frac{1}{9}$ * MobileNetV2<br>• $\frac{1}{3}$ * InceptionNetV3<br>• $\frac{1}{9}$ * VGG16<br>• $\frac{1}{9}$ * ResNet50V2<br>• $\frac{1}{3}$ * EfficientNetV2 | 0.7450 | 0.7769 | 660*<br>*Sum of individual model epochs |
| Ensemble Model with higher weights<br>• $\frac{1}{243}$ * MobileNetV2<br>• $\frac{120}{243}$ * InceptionNetV3<br>• $\frac{1}{243}$ * VGG16<br>• $\frac{1}{243}$ * ResNet50V2<br>• $\frac{120}{243}$ * EfficientNetV2 | 0.7844 | 0.7765 | 660*<br>*Sum of individual model epochs |
| Ensemble Model with select models<br>• 0 * MobileNetV2<br>• $\frac{1}{2}$ * InceptionNetV3<br>• 0 * VGG16<br>• 0 * ResNet50V2<br>• $\frac{1}{2}$ * EfficientNetV2 | 0.8030 | 0.7765 | 660*<br>*Sum of individual model epochs |
| Custom CNN | 1.2555 | 0.6968 | 400 |

**Table 1: Sparse Categorical Cross-Entropy Loss, Validation Accuracy, Number of Epochs for Models Tested**

**Discussion**

The best individual model was EfficientNetV2, while VGG16 was the worst with regards to validation accuracy, which disagreed with the findings of Jorid et al.[2] and Gour et al.[3] but confirmed those of Wang et al.[4] InceptionNetV3 had the lowest loss, though, whereas EfficientNetV2 had the highest loss, indicating that models with the highest validation accuracy do not necessarily have the lowest loss.

As predicted, the ensemble model composed of a weighted average of all of the transfer learning models performed better than any individual model. The ensemble model that combined all five of the transfer learning models outperformed an ensemble model with only the two best performing models in terms of loss and validation accuracy, which was somewhat unexpected. Considering the time required to train each of the transfer learning models and the improvement from ensembling them (~0.2 reduction in loss compared to InceptionNetV3 and ~0.1 increase in validation accuracy compared to EfficientNetV2), though, the final ensemble model may not be a particularly useful method in practice.

In contrast, our prediction that the transfer learning models would have a higher validation accuracy than our custom CNN did not prove correct. VGG16 had both a lower validation accuracy and higher loss than the custom CNN, and InceptionNetV3 and ResNet50V2 both reported only slightly higher validation accuracies, along with higher loss values. This finding may be confounded by the discrepancies in epochs used to train each of the models and base learning rates.

We encountered several challenges throughout this project. First, the process of cleaning the data was tedious and, admittedly, somewhat imprecise, due to the idiosyncrasies in terms used by physicians to describe the same conditions; the overlap in certain terms across diagnoses, such as different types of retinopathy associated with diabetes, myopia, and other conditions; and multiple diagnoses for a given eye. In addition, assigning the diagnoses for each patient to the correct eye, since some patients had eyes with distinct conditions, was nearly impossible without any ground truth with which to compare the resulting diagnoses. Differences in preprocessing workflows across research papers investigated during the literature review led to different counts for the number of images in each class, meaning we could not easily confirm the viability of our method.

Second, while the process of multiclass classification was more familiar and easier to implement than multi-label classification, if these models were used in a clinical setting, they would fail to correctly diagnose novel combinations of diagnoses not present in the training dataset. For example, there are no patients with both cataracts and myopia in the dataset, but according to the Myopia Institute, cataracts are more common in people with myopia than those without it.[10] As such, multiclass classification constrained our models and made them more dependent on the training data than with multi-label classification.

Third, multiclass classification only exacerbated the class imbalances in the original dataset, because certain combinations of diagnoses were quite rare. We attempted to use random oversampling with replacement (and class weights when training models) to combat this imbalance, but in order to do so, we had to duplicate certain images many times. This strategy likely limited the generalizability of our models, even when data augmentation was added to our neural networks to increase robustness.

Finally, training and evaluating models was time consuming and required significant computational resources. We were forced to use slurm scripts to develop our models, and even then, this process could take hours, and also, unfortunately, lacked any visualization of loss and

accuracy metrics in the output files. As such, we largely had to monitor changes in performance metrics manually, and could not train certain models, such as the ensemble models, for as long as desired.

**Conclusion**

Ensembling can improve upon transfer learning methods used in this and previous studies by combining predictions from high-performing models to accurately diagnose various ocular diseases. Designing new CNNs also shows promise for tackling this challenge. The time, computational load, and cost of creating and deploying models must be considered, though, especially in clinical settings with limited resources. In short, while neural networks can play an important role in informing diagnoses, it may not be feasible for all physicians.

Future work would include training all models for more epochs and further refining weight values for the ensemble model. Custom CNNs with different, more advanced architectures are another avenue for exploration, and one that has not been discussed as much in the literature. Additional evaluation metrics, such as class-specific area under the curve (AUC), F-score, and Cohen's kappa, will also help assess model performance on individual classes and provide insights into new ways to combat the persistent issue of class imbalances.[7]

**For more information, please visit our GitHub and view our presentation linked below.**

**GitHub link**: https://github.com/cew4pf/DS6050-eye-project.git

**Presentation**:
https://drive.google.com/file/d/1SifDwj3arTktgzgqaCDH87w46-26s6aQ/view?usp=sharing

**Sources**

1. "Vision Impairment and Blindness." World Health Organization. World Health Organization, 2021. https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment.

2. JordiCorbilla. "Jordicorbilla/Ocular-Disease-Intelligent-Recognition-Deep-Learning: Odir-2019. Ocular Disease Intelligent Recognition through Deep Learning Architectures." GitHub. Accessed March 6, 2022. https://github.com/JordiCorbilla/ocular-disease-intelligent-recognition-deep-learning.

3. Gour, Neha, and Pritee Khanna. "Multi-Class Multi-Label Ophthalmological Disease Detection Using Transfer Learning Based Convolutional Neural Network." *Biomedical Signal Processing and Control* 66 (2021): 102329. https://doi.org/10.1016/j.bspc.2020.102329.

4. Li, Cheng, Jin Ye, Junjun He, Shanshan Wang, Yu Qiao, and Lixu Gu. “Dense Correlation Network for Automated Multi-Label Ocular Disease Detection with Paired Color Fundus Photographs.” *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, 2020. https://doi.org/10.1109/isbi45749.2020.9098340.

5. Wang, Jing, Liu Yang, Zhanqiang Huo, Weifeng He, and Junwei Luo. “Multi-Label Classification of Fundus Images with EfficientNet.” *IEEE Access* 8 (2020): 212499–508. https://doi.org/10.1109/access.2020.3040275.

6. He, Junjun, Cheng Li, Jin Ye, Yu Qiao, and Lixu Gu. “Self-Speculation of Clinical Features Based on Knowledge Distillation for Accurate Ocular Disease Classification.” *Biomedical Signal Processing and Control* 67 (2021): 102491. https://doi.org/10.1016/j.bspc.2021.102491.

7. Demir, Fatih, and Burak Taşcı. “An Effective and Robust Approach Based on R-CNN+LSTM Model and NCAR Feature Selection for Ophthalmological Disease Detection from Fundus Images.” *Journal of Personalized Medicine* 11, no. 12 (2021): 1276. https://doi.org/10.3390/jpm11121276.

8. Larxel. “Ocular Disease Recognition.” Kaggle, September 24, 2020. https://www.kaggle.com/andrewmvd/ocular-disease-recognition-odir5k?select=preprocessed_images.

9. “Classification on imbalanced data.” TensorFlow, April 15, 2022. https://www.tensorflow.org/tutorials/structured_data/imbalanced_data#train_a_model_with_class_weights

10. Myopia Institute. “The Link Between Myopia and Cataracts.” July 5, 2018. https://www.myopiainstitute.com/eye-care/the-link-between-myopia-and-cataracts/