

A Hybrid EfficientNet-B0 and Machine Learning Framework for Multi-Class Retinal Disease Classification on RFMiD

Md. Alamgir Hossain¹, *, Paritosh Chandra Ray¹,
Md. Mehedi Hasan¹, Abdul Al Mahin¹, Ferdowsi Khatun²

¹Department of Computer Science and Engineering, Dhaka International University, Dhaka-1212, Bangladesh

²Department of Computer Science and Engineering, Dhaka University of Engineering & Technology, Gazipur-1707, Bangladesh

Abstract- Automated analysis of retinal fundus images can support scalable screening, but multi-disease classification remains difficult because retinal abnormalities may exhibit overlapping visual characteristics and clinically important classes are often underrepresented. This paper investigates a four-class retinal image classification framework using a strictly filtered subset of the Retinal Fundus Multi-Disease Image Dataset (RFMiD). The considered classes are No_Disease_Risk, diabetic retinopathy, age-related macular degeneration, and macular hole. Images with multiple disease labels, unselected diseases, or inconsistent annotations were excluded, resulting in 855 training, 294 validation, and 305 test images. Training data were balanced to 401 entries per class through sampling with replacement and stochastic augmentation. A pretrained EfficientNet-B0 with frozen parameters was used to extract a 1,280-dimensional representation from each image. Support vector machine, Random Forest, K-nearest neighbors, XGBoost, and CatBoost classifiers were subsequently evaluated. On the official test subset, CatBoost attained the highest accuracy of 0.9016, weighted F1-score of 0.9022, and macro-ROC-AUC of 0.9842. SVM achieved the highest macro precision and macro F1-score, at 0.9075 and 0.8628, respectively, while XGBoost produced the highest macro recall of 0.8774. Grad-CAM visualizations were generated for selected test images to examine the retinal regions associated with the model predictions. The results show that fixed deep representations can support effective retinal image classification with conventional learners, although the small number of unique ARMD samples limits the statistical reliability and generalizability of the findings.

Keywords— CatBoost, deep feature extraction, EfficientNet-B0, fundus image analysis, machine learning, retinal disease classification, RFMiD.

I. INTRODUCTION

Retinal disorders are major causes of visual impairment, and several conditions can progress before substantial symptoms are recognized. Color fundus photography provides a noninvasive representation of the retina in which the optic disc, macula, vessels, and disease-associated abnormalities can be examined. Consequently, fundus images have become an important input for computer-assisted retinal screening and image-based disease analysis [1], [2]. Reviews of this field show that deep learning has been investigated for retinal lesion segmentation, biomarker analysis, disease recognition, and image synthesis, while also identifying data imbalance, limited external validation, and model interpretability as continuing

challenges [2]. Interpreting fundus images manually requires significant medical expertise and may present challenges around scaling given the limits of available expertise. Automated classification does not replace the need for a clinical exam but rather serves as a foundation for further research in the area of triage or decision-making. The multi-class type of retinal classification is especially difficult due to the fact that there are minor visible signs in a small part of the picture, the manifestation of diseases overlaps, the conditions for image acquisition are different, and various types of diseases have dramatically different amounts of labeled data. Convolutional neural networks (CNNs) can learn hierarchical image representations directly from pixel data. However, reliable end-to-end CNN optimization generally benefits from sufficiently

large and representative labeled datasets. Transfer learning offers an alternative when the target dataset is comparatively limited. Features learned from a large source dataset can be transferred to a medical imaging task either through network fine-tuning or by retaining the pretrained network as a fixed feature extractor [3], [4]. In the latter setting, the CNN converts each image into a compact numerical representation, after which a separate classifier learns the target decision boundaries. EfficientNet is suitable for this role because its architecture was developed through coordinated scaling of network depth, width, and image resolution [3]. EfficientNet-B0 provides the baseline member of the EfficientNet family and produces a 1,280-dimensional representation after removal of the original classification head. The resulting vectors can be modeled using kernel-based, neighborhood-based, bagging-based, or boosting-based classifiers. Such a design allows the representation model to remain fixed while the behavior of different classification algorithms is compared under the same feature space.

The RFMiD dataset contains 3,200 retinal fundus images with annotations covering normal findings and a broad range of retinal conditions [1]. RFMiD was originally designed as a multi-disease resource and permits more than one disease label for an image. The present experiment does not address the complete RFMiD multilabel task. Instead, the official training, validation, and test partitions were filtered to retain four mutually exclusive categories: No_Disease_Risk, diabetic retinopathy (DR), age-related macular degeneration (ARMD), and macular hole (MH). The principal objective is to determine how five classical machine-learning algorithms behave when trained on a common set of frozen EfficientNet-B0 features. The main contributions are as follows:

- A strict four-class RFMiD subset was constructed by excluding multi-label cases, unselected diseases, and inconsistent label combinations.
- A training-only balancing and augmentation procedure was applied while the official validation and test subsets remained unaugmented.

- Frozen ImageNet-pretrained EfficientNet-B0 features were used to represent every retained image in a 1,280-dimensional feature space.
- SVM, Random Forest, KNN, XGBoost, and CatBoost were compared using accuracy, macro- and support-weighted measures, and one-vs-rest ROC-AUC.
- The aggregate results were complemented by class-wise reports, confusion-matrix analysis, and explicit consideration of the small ARMD subset.

No claim of state-of-the-art or clinical diagnostic performance is made. The experiment is a controlled comparison of classifiers on one strictly filtered RFMiD configuration.

II. RELATED WORK

1. Retinal Fundus Image Classification

CNN-based fundus image analysis has expanded from individual disease-detection tasks toward systems capable of distinguishing multiple retinal abnormalities. Li et al. reviewed more than one hundred applications and reported that CNNs have been used across retinal classification, segmentation, synthesis, and biomarker-analysis tasks [2]. Multi-condition modeling is important because a single-disease detector may not represent the broader range of retinal findings encountered in practice. Research has also demonstrated models that differentiate numerous fundus diseases and conditions, although dataset composition and evaluation protocols vary substantially between studies [5].

Using CNNs for the analysis of fundus images started with single condition tasks and evolved into complex systems capable of differentiating several various retinal problems. Li et al. analyzed over 100 cases of work such as retinal classification, segmentation, synthesis, and others [2]. However, multi-condition modeling is crucial because a single disease model will not be able to fully capture all backgrounds of findings presented in practice [5].

Direct comparison between published retinal-classification results is often inappropriate when the studies use different disease definitions, filtering criteria, partitions, or labeling schemes. This concern is especially relevant for RFMiD, which was released as a multilabel resource with 3,200 images captured using three fundus cameras and annotations for 46 conditions [1]. The present study changes the original problem into an exclusive four-class task. Accordingly, the reported scores characterize the retained subset rather than the complete RFMiD benchmark.

2. Transfer Learning and EfficientNet

In transfer learning, knowledge obtained from a source task can be utilized when learning a similar target task. Due to the fact that finding enough high-quality labeled data in medical image classification is often not possible, pretrained CNNs are often used [4]. Depending on the way the experiments are designed, one can choose to either leave the pretrained layers unchanged or fine-tune them, feeding their outputs into another classifier. Tan and Le developed EfficientNet by employing technique known as compound scaling, which implies varying depth, width and resolution in parallel rather than independently. The EfficientNet series proved to provide good balance between accuracy and efficiency and facilitated transfer learning on different datasets [3]. In the experiment conducted, EfficientNet-B0 is not fine-tuned, is used without the need to make changes. Its parameters from ImageNet are frozen and its classification layer is replaced with identity mapping allowing to operate with the representations of images with 1,280 elements.

3. Classical Classifiers for Deep Representations

SVM establishes a decision-making boundary of maximum distance and allows creating non-linear classifications by application of kernel functions [6]. In the setup of fixed CNN embeds, radial base kernel is applicable as the boundaries of classes are not required to be linearly separable. The Random Forest approach compiles a collection of randomized decision tree models, thus making it independent of any one model due to voting [7]. The ability to formulate non-linear feature representations does

not require standardizing variables. KNN classifies instances based on nearness to exemplar data, but the straightforward nature of this technique can suffer from high dimensionality and variance of features. That is why EfficientNet-B0 features have been standardized for executing a KNN algorithm. For the purposes of classification, XGBoost algorithm uses regularized gradient boosting methodology and some computational trick [8]. Likewise, CatBoost is also a method of gradient boosting which is built around its own features like ordered boosting and symmetric tree construction. In this particular application, the ability to handle categorical variables is insignificant as all variables are presented in the form of continuous CNN features [9].

The relevant experimental gap is therefore not the absence of retinal CNN research, but the need for a controlled comparison in which varied classical classifiers receive exactly the same EfficientNet-B0 representation under an explicitly defined exclusive RFMiD configuration.

III. PROPOSED METHODOLOGY

The complete proposed workflow is illustrated in Fig. 1. The pipeline comprises RFMiD subset construction, training-data augmentation, image resizing and normalization, frozen EfficientNet-B0 feature extraction, classical machine-learning classification, and model evaluation. Retinal fundus images (DR, ARMD, MH, No_Disease) are augmented and resized to $224 \times 224 \times 3$.

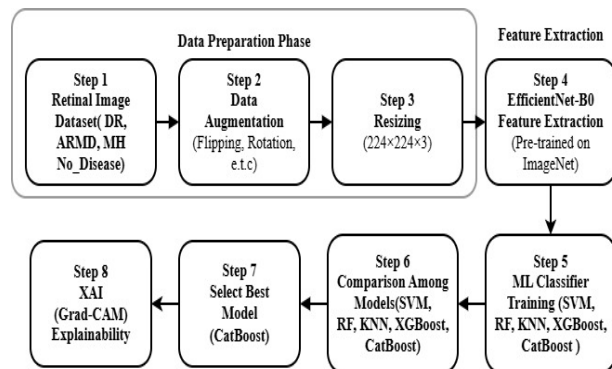


Fig. 1: Proposed workflow for four-class retinal disease classification using frozen EfficientNet-B0 features and classical machine-learning classifiers.

A frozen, ImageNet-pretrained EfficientNet-B0 extracts deep features, which train SVM, RF, KNN, XGBoost and CatBoost classifiers. After comparing these models, CatBoost was selected as the best, and Grad-CAM provides visual explainability.

RFMiD Dataset and Exclusive-Class Construction

RFMiD contains 3,200 retinal fundus images divided into official training, validation, and test sets of 1,920, 640, and 640 images, respectively [1]. Within the label files, there exists a general concept of Disease_Risk and 45 columns for disease indicators. The experiment has kept three types of diseases namely DR, ARMD, and MH, apart from a No_Disease_Risk type.

A row is marked with a No_Disease_Risk only when the state of the Disease_Risk is equal to zero and the 45 illness markers are at a zero level as well. A disease image is preserved only for the cases when the disease state is one while any single disease indicator is marked positively and this indicator belongs to one of DR, ARMD, or MH. Images will be omitted if they possess more than two positive diagnoses, have at least one but not selected disease type, have visible signs of illness but no significant disease indicator, and have normal records which contradict other information. This allowed to change the original dataset of images from 3,200 to 1,454 images. Fig. 2 shows representative retinal fundus images from the four retained classes: No_Disease_Risk, DR, ARMD, and MH. Table I reports the retained distribution.

The physical image and the label files generated had been compared against each other, and it was confirmed that there is no duplicity of assignment between partitions, each image of the dataset is a member of only one class. The original RFMiD partitions were kept intact, and thus no new train-test split was made. The training set obtained is very imbalanced, with 401 distinct images in the No_Disease_Risk class and only 34 in the ARMD class. Thus, the number of images in the ARMD class constitutes approximately 8.5% of the number of images in the class of healthy individuals.

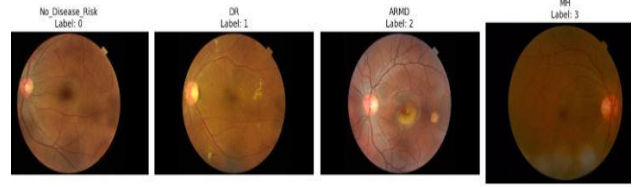


Fig. 2: Representative retinal fundus images from the four retained classes: No_Disease_Risk, diabetic retinopathy (DR), age-related macular degeneration (ARMD), and macular hole (MH).

Table I. Distribution of the Strictly Exclusive Four-Class Subset

Class	Training	Validation	Test	Total
No_Disease_Risk	401	134	134	669
DR	240	81	80	401
ARMD	34	17	18	69
MH	180	62	73	315
Total	855	294	305	1454

Image Preprocessing and Training Augmentation

Every retinal fundus image was transformed to RGB format, and dimensions were resized to 224×224 pixels. To enhance the diversity of training data and boost model robustness, data augmentation was done only on the training data portion. The augmentation process involved altering the images by flipping them horizontally and vertically, doing random rotations by $\pm 15^\circ$, changing brightness, contrast, and saturation. After performing augmentation, images were transformed to tensor and normalized using ImageNet mean and standard deviation. The validation and test subsets underwent only resizing and normalization, with no random augmentation applied, thereby ensuring a consistent and unbiased evaluation protocol.

Proposed Classification Pipeline

Fig. 1 summarizes the experimental sequence. A preprocessed image is passed through EfficientNet-B0, the final ImageNet classification layer is removed, and the resulting representation is supplied to one of five classical classifiers.

Let $\mathbf{x}_i \in \mathbb{R}^{3 \times 224 \times 224}$ denote the preprocessed i -th image. The fixed feature extractor f_θ , with frozen parameters θ , generates

Probability outputs were also needed for calculating the ROC and AUC scores for multiclass classification. 2) *Random Forest*: The Random Forest (RF) classifier consisted of 500 decision trees. Tree depth was left unrestricted, with `min_samples_split = 2` and `min_samples_leaf = 1`. At each split, the number of possible features is defined according to the square root of the total number of features. To deal with class imbalance issues, balanced class weights were taken into account. Predictions were provided through the majority voting rule across the whole ensemble of randomized trees in use.

K-Nearest Neighbors: The K-Nearest Neighbors (KNN) classifier used $k = 5$ neighbors with distance-weighted voting. The Minkowski distance with $p=2$ was used to determine similarity between objects, which is equivalent to Euclidean distance. In order to persist in neighborhood calculations and being sensitive to the scaling of the features of objects, EfficientNet-B0 standardized feature vectors were used for training the classifier.

XGBoost: The Extreme Gradient Boosting (XGBoost) classifier was used with the multi:softprob objective function and multiclass logloss for evaluating multiclass classification. The model was configured with 500 estimators, a learning rate of 0.05, a maximum tree depth of 5, and a minimum child weight of 1. Row and column subsampling ratios were both set to 0.80. The L1 and L2 regularization coefficients were set to 0.0 and 1.0, respectively.

CatBoost: The CatBoost classifier was configured using multiclass loss and evaluation functions. Training was performed for 500 iterations with a learning rate of 0.05, a tree depth of 6, and an L2 leaf-regularization coefficient of 3. File-writing side effects were disabled during training, and the classifier received the unscaled EfficientNet-B0 feature representations as input. Table II summarizes the implementation-specific hyperparameter settings used for all evaluated classifiers.

Table 2. Classifier Configurations

Classifier	Experimental Configuration
SVM	RBF kernel; $C = 10.0$; <code>gamma = scale</code> ; probability estimation enabled; one-vs-rest (OvR) decision function
Random Forest	500 trees; unrestricted depth; square-root feature sampling; balanced class weights
KNN	$k = 5$; distance-weighted voting; Euclidean distance ($p = 2$)
XGBoost	500 estimators; learning rate 0.05; maximum depth 5; row subsampling 0.80; column subsampling 0.80
CatBoost	500 iterations; learning rate 0.05; tree depth 6; L2 leaf regularization coefficient 3

Experimental Environment

The notebook was executed in a Kaggle-hosted Python environment. CUDA availability was detected, and the feature extractor ran on a CUDA device. The random seed was fixed at 42 for Python, NumPy, PyTorch, and the supported classifier parameters.

The implementation used PyTorch and Torchvision for image loading, preprocessing, and EfficientNet-B0 feature extraction; scikit-learn for SVM, Random Forest, KNN, feature scaling, and metrics; and the XGBoost and CatBoost packages for gradient-boosted classifiers.

The uploaded materials do not report:

- the GPU model;
- CPU and RAM specifications;
- operating-system details;
- Python or library versions;
- training and inference time;
- model-size measurements; or
- memory consumption.

These details should be added before submission if reproducibility requirements of the target conference include the complete computing environment.

Evaluation Measures

For a multiclass test set containing N samples, classification accuracy was computed as

$$\text{Accuracy} = \frac{1}{N} \sum_{i=1}^N \mathbb{I}(y_i = \hat{y}_i), \quad (5)$$

where $\mathbb{I}(\cdot)$ denotes the indicator function, y_i is the true label, and $\hat{y}_i$ is the predicted label. For each class c , precision, recall, and F1-score were calculated as

$$P_c = \frac{TP_c}{TP_c + FP_c}, \quad (6)$$

$$R_c = \frac{TP_c}{TP_c + FN_c}, \quad (7)$$

$$F1_c = \frac{2P_c R_c}{P_c + R_c}, \quad (8)$$

where TP_c , FP_c , and FN_c denote the true positives, false positives, and false negatives associated with class c , respectively. Macro-averaged performance assigns equal importance to each of the $C = 4$ classes and is defined as

$$M_{\text{macro}} = \frac{1}{C} \sum_{c=1}^C M_c, \quad (9)$$

where M_c represents the metric value for class c . Support-weighted averaging accounts for the number of samples, n_c , in each class and is computed as

$$M_{\text{weighted}} = \frac{\sum_{c=1}^C n_c M_c}{\sum_{c=1}^C n_c}. \quad (10)$$

Receiver Operating Characteristic Area Under the Curve (ROC-AUC) was evaluated using a one-vs-rest strategy. For each class, the corresponding predicted probability was treated as the score for distinguishing that class from all remaining classes.

Macro ROC-AUC was obtained by averaging the class-specific AUC values with equal weight. In addition to accuracy, precision, recall, and F1-score, confusion matrices and ROC-AUC scores were used to provide a comprehensive assessment of classifier performance on the filtered RFMiD subset.

IV. EXPERIMENTAL RESULTS

Validation-Set Performance

Table III reports performance on the 294-image official validation subset. CatBoost achieved the highest validation accuracy, macro F1-score, and weighted F1-score. XGBoost produced the highest validation macro-ROC-AUC. The difference between macro and weighted results reflects the unequal amount of validation support. No_disease_risk has 134 validation instances compared to only 17 in ARMD. The weighted F1 score of SVM was 0.8390, while the macro F1 score was 0.7628. The reason for the lower macro result can be attributed to the ARMD F1 score of 0.5185. CatBoost and XGBoost achieved the ARMD recall of 0.8235; however, their ARMD precisions in validation were 0.6087 and 0.5833, respectively. KNN performed the worst in all the validation measures. Its ARMD precisions, recalls, and F1 scores were found to be 0.3571, 0.2941, and 0.3226. This indicates that using local Euclidean neighborhoods in the normalized feature space was not helpful compared to the use of ensemble models.

Table 3. Performance on the Official Validation Subset

Model	Acc.	Macro P	Macro R	Macro F1	Weighted P	Weighted R	Weighted F1	Macro AUC
SVM	0.8435	0.8043	0.7415	0.7628	0.8408	0.8435	0.8390	0.9501
Random Forest	0.8435	0.7848	0.7829	0.7800	0.8493	0.8435	0.8435	0.9628
KNN	0.7653	0.6726	0.6520	0.6603	0.7603	0.7653	0.7609	0.8665
XGBoost	0.8605	0.8076	0.8351	0.8148	0.8676	0.8605	0.8613	0.9708
CatBoost	0.8707	0.8193	0.8453	0.8273	0.8760	0.8707	0.8715	0.9677

Test-Set Performance

Table IV reports the final test results. CatBoost attained the highest accuracy, weighted precision,

weighted recall, weighted F1-score, and macro-ROC-AUC. SVM attained the highest macro precision and macro F1-score, while XGBoost produced the highest macro recall.

Table 4. Performance on the Official Test Subset

Model	Acc.	Macro P	Macro R	Macro F1	Weighted P	Weighted R	Weighted F1	Macro AUC
SVM	0.8754	0.9075	0.8351	0.8628	0.8830	0.8754	0.8746	0.9835
Random Forest	0.8656	0.8190	0.8318	0.8168	0.8781	0.8656	0.8659	0.9760
KNN	0.7967	0.7509	0.7115	0.7197	0.8175	0.7967	0.7940	0.9216
XGBoost	0.8918	0.8580	0.8774	0.8610	0.9000	0.8918	0.8917	0.9827
CatBoost	0.9016	0.8599	0.8661	0.8590	0.9089	0.9016	0.9022	0.9842

CatBoost correctly classified 275 of the 305 test images. XGBoost, SVM, Random Forest, and KNN correctly classified 272, 267, 264, and 243 images, respectively. The ranking depends on the selected evaluation criterion. CatBoost produced the strongest aggregate and support-weighted performance, but SVM's macro F1-score of 0.8628 slightly exceeded the values of 0.8610 and 0.8590 obtained by XGBoost and CatBoost. XGBoost's macro recall of 0.8774 was the highest, indicating the strongest average sensitivity across the four classes. Therefore, describing any model as universally superior would oversimplify the observed trade-offs. The absolute difference between CatBoost's macro F1-score and weighted F1-score was 0.0432. The corresponding difference for XGBoost was 0.0307. These disparities indicate that aggregate scores weighted by class support are increased by strong performance on the larger normal, DR, and MH categories.

Validation-to-Test Behavior

All five models showed improved accuracy on the test subset compared to the validation subset. The improvements were measured at 0.0319 for SVM, 0.0221 for Random Forest, 0.0314 for KNN, 0.0313 for XGBoost, and 0.0309 for CatBoost. Macro ROC-AUC also experienced growth for all models. However, the improvements cannot be assumed to imply that the test subset was easier in nature, which will require further analysis. There is a considerable difference between validation and test images with respect to their specifics and class representations; in ARMD, there are 17 validation and 18 test samples. Considering that, we can say that the change in the small minority classes may significantly affect the overall macro results.

Class-Wise and Error Analysis

CatBoost Classification Report

CatBoost achieved the highest test accuracy and weighted F1-score. Its class-level results are presented in Table V. In total, CatBoost identified 130 of the 134 No_Disease_Risk photographs correctly. In the case of normal-class mistakes, three ended up in the category of MH and two to ARMD. For DR, 65 out of 80 photographs were identified correctly. No_Disease_Risk turned out to be the most frequent error (eight cases), followed by ARMD (six cases). 14 out of the 18 images of ARMD were identified. Three of them were assigned to No_Disease_Risk and one to DR. ARMD had the precision of 0.6364 but quite high recall of 0.7778. This can be explained by the misclassification of certain images that belonged to other categories, particularly DR. CatBoost has correctly identified 66 of the 73 images of MH. Five images were assigned to No_Disease_Risk, one to DR and one to ARMD.

Table 5. CatBoost Class-Wise Performance on the Test Subset

Class	Precision	Recall	F1-score	Support
No_Disease_Risk	0.8904	0.9701	0.9286	134
DR	0.9701	0.8125	0.8844	80
ARMD	0.6364	0.7778	0.7000	18
MH	0.9429	0.9041	0.9231	73

Confusion-Matrix Analysis

Fig. 4 presents the confusion matrices of all five classifiers on the official test subset. CatBoost correctly classified 275 of the 305 test images, followed by XGBoost, SVM, Random Forest, and KNN with 272, 267, 264, and 243 correct predictions, respectively. Across the evaluated models, disease

images were most frequently misclassified as No_Disease_Risk. KNN exhibited this tendency most strongly, whereas CatBoost produced the largest diagonal total. XGBoost correctly identified 16 of the 18 ARMD images, representing the highest ARMD recall among the evaluated classifiers.

ROC-AUC Analysis

Fig. 5 presents the one-vs-rest ROC curves of the SVM classifier on the official test subset. The class-specific AUC values were 0.9801 for No_Disease_Risk, 0.9793 for DR, 0.9855 for ARMD, and 0.9891 for MH. Their arithmetic mean is

$$\frac{0.9801+0.9793+0.9855+0.9891}{4} = 0.9835, \quad (11)$$

which matches the reported SVM macro-ROC-AUC.

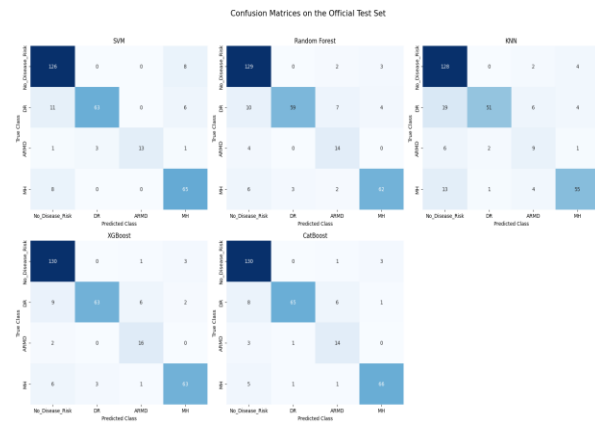


Fig. 4: Confusion matrices of SVM, Random Forest, KNN, XGBoost, and CatBoost on the 305-image official test subset.

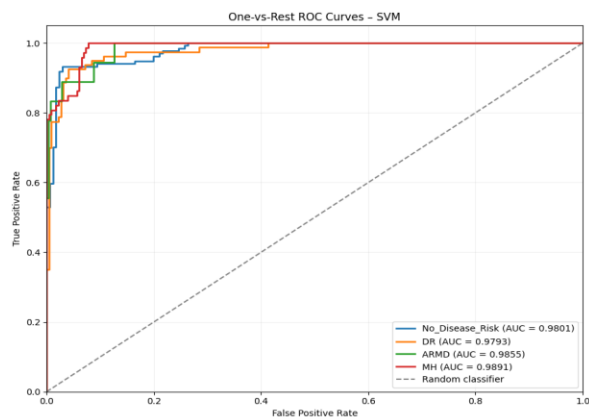


Fig. 5: One-vs-rest ROC curves of the SVM classifier on the official test subset.

The ROC curves remain well above the chance diagonal. However, AUC evaluates ranking across possible thresholds, whereas the confusion matrix reflects a single multiclass decision rule. Thus, the ARMD AUC of 0.9855 and recall of 0.7222 are not inconsistent. The SVM probability scores can rank ARMD cases effectively while final multiclass predictions still assign some ARMD images to competing labels. At the aggregate level, CatBoost achieved the highest macro-ROC-AUC of 0.9842, followed by SVM at 0.9835 and XGBoost at 0.9827. The differences among these three values are small and were not subjected to statistical significance testing. Random Forest achieved 0.9760, while KNN was lower at 0.9216.

Explainability Analysis

Grad-CAM was applied to examine the spatial regions contributing to the model's prediction. Fig. 6 presents an example from the MH class, including the original fundus image, the generated activation heatmap, and the heatmap overlaid on the original image. The classification of the sample as MH was accurate, with a predicted likelihood of 0.9558. High activation occurs within a localized central retina area, and less activity occur around that area. This finding offers information about regions of the image related to the prediction of the model. Nevertheless, the area presented should not be treated as conclusive proof of the presence of a clinically relevant lesion. Grad-CAM shows the properties that have influenced the model output instead of providing a definitive clinical diagnosis.

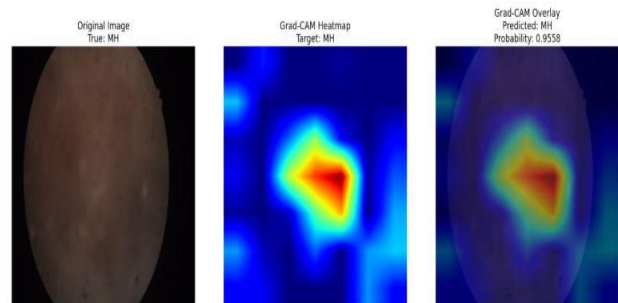


Fig. 6: Grad-CAM visualization for a correctly classified MH test image: original fundus image, class-specific activation heatmap, and heatmap overlay. The predicted class was MH with a probability of 0.9558.

Comparative Model Interpretation

The nonlinear RBF SVM produced the highest macro precision and macro F1-score. Its ARMD precision of 1.0000 contributed to the high macro precision, although the estimate is based on only 13 ARMD predictions. Random Forest performed competitively on the majority categories but achieved an ARMD precision of 0.5600 and a macro F1-score of 0.8168. Random feature selection and ensemble voting provided nonlinear modeling capacity, although this implementation did not match the two boosted tree models in overall accuracy. KNN showed the largest tendency to predict disease cases as normal. The high feature dimensionality, residual overlap between disease representations, and sensitivity of neighborhood methods to local sample structure may be associated with this behavior. These explanations remain hypotheses because no dimensionality-reduction or neighborhood-sensitivity ablation was performed. XGBoost attained the highest macro recall and test ARMD recall. This result indicates a greater tendency to recover minority-class cases, but its ARMD precision of 0.6667 shows that the increased sensitivity was accompanied by false-positive ARMD assignments. CatBoost achieved the strongest accuracy and support-weighted measures. The observed outcome may be associated with the model's ability to form nonlinear boosted-tree partitions over the deep feature vectors. However, the experiment did not isolate which difference between CatBoost and XGBoost generated the performance gap. No conclusion about inherent algorithmic superiority can therefore be drawn.

V. CONCLUSION AND FUTURE WORK

Conclusion

This paper assessed five traditional classifiers employing fixed EfficientNet-B0 representations for extraordinary four-class retinal image classification on RFMiD dataset. Rigid label filtering preserved images from only the classes No_Disease_Risk, DR, ARMD, and MH, while multi-labeled, unprovided dials, and inconsistent cases were excluded. After training-only augmentation and oversampling, the pretrained network on ImageNet gave rise to a 1,280-dimensional feature vector to SVM, Random

Forest, KNN, XGBoost, and CatBoost. CatBoost has gained the best testing accuracy of 0.9016, weighted F1-score of 0.9022, and macro-ROC-AUC of 0.9842. The SVM classifier attained the best macro precision of 0.9075 and macro F1-score of 0.8628, while the XGBoost algorithm had the best macro recall of 0.8774 and demonstrated correct classification of 16 ARMD images from 18 images. KNN showed the worst results since it frequently labeled diseased images as No_Disease_Risk.

The findings indicate that frozen EfficientNet-B0 representations can support effective nonlinear classification without end-to-end CNN training. Nevertheless, the very small ARMD subset, oversampling of repeated image paths, exclusive-label filtering, and absence of external validation limit the conclusions. From the results, it seems that frozen EfficientNet-B0 representations can be used for reliable nonlinear classification without the need for end-to-end CNN training. However, the small ARMD subset, the oversampling of repetitions in image paths, the filtering of exclusive labels, and the lack of external validation restrict the conclusions.

Future Work

Further research will look into the existing underlying mutually exclusive four-class architecture towards multi-disease prediction where one retinal fundus image can receive several disease labels. This advancement is crucial because multiple retinal disorders can be found together in one image, while the current study does not consider multi-label cases. The multi-label model will utilize the complete RFMiD annotation structure regarding the probability of retinal diseases and will not limit each image to one category. The experiments will conduct research on multi-label family of losses, class-aware sampling, and thresholds optimization, and evaluation methods such as macro- and micro-F1 score, mean average precision, and per-class ROC-AUC. The architecture can also be improved due to partial or end-to-end tuning of EfficientNet-B0, the use of larger and externally verified datasets, and other techniques to detect diseases like those faced through the current study. Finally, external validation and reliability testing can lead to the development of web or smartphone application, that will take in a

retinal fundus image, go through the same pre-processing approach, and return probabilities of diseases including many possible ones. The interface could additionally present confidence scores, image-quality warnings, and visual explanations.

REFERENCES

1. S. Pachade, P. Porwal, D. Thulkar, M. Kokare, G. Deshmukh, V. Sahasrabuddhe, L. Giancardo, G. Quéllec, and F. Mériaudeau, "Retinal fundus multi-disease image dataset (RFMiD): A dataset for multi-disease detection research," *Data*, vol. 6, no. 2, p. 14, Feb. 2021. <https://doi.org/10.3390/data6020014>
2. T. Li, W. Bo, C. Hu, H. Kang, H. Liu, K. Wang, and H. Fu, "Applications of deep learning in fundus images: A review," *Medical Image Analysis*, vol. 69, p. 101971, Apr. 2021. <https://doi.org/10.1016/j.media.2021.101971>
3. M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, vol. 97, 2019, pp. 6105–6114. <https://doi.org/10.48550/arXiv.1905.11946>
4. A. W. Salehi, S. Khan, G. Gupta, B. I. Alabdullah, A. Almjally, H. Alsolai, T. Siddiqui, and A. Mellit, "A study of CNN and transfer learning in medical imaging: Advantages, challenges, future scope," *Sustainability*, vol. 15, no. 7, p. 5930, 2023. <https://doi.org/10.3390/su15075930>
5. J. Cen et al., "Automatic detection of 39 fundus diseases and conditions in retinal photographs using deep neural networks," *Nature Communications*, vol. 12, 2021. <https://doi.org/10.1038/s41467-021-25138-w>
6. C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995. <https://doi.org/10.1007/BF00994018>
7. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. <https://doi.org/10.1023/A:1010933404324>
8. T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794. <https://doi.org/10.1145/2939672.2939785>
9. L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased boosting with categorical features," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 6638–6648. <https://doi.org/10.48550/arXiv.1706.09516>