Ensemble Network for Glaucoma Screening in AIROGS Challenge

Edward Wang[1,*], Ashritha Durvasula[1], Daisy Deng[1], Asaanth Sivajohan[1], Edward Ho[1], and Kevin Lane[1]

[1]Schulich School of Medicine and Dentistry, University of Western Ontario, London, Canada

Glaucoma is a vision impairing disease of the eye with pathological changes that can be seen on colour fundus photography. This paper presents an ensemble method for glaucoma screening used in the Artificial Intelligence for RObust Glaucoma Screening challenge. A YoloV5 model was used for optic disc detection and a subsequent ensemble of convolutional neural network classifiers was used to identify referable glaucoma versus non-referable glaucoma. Two autoencoders were used to estimate ungradability.

Correspondence: ***ewang225@uwo.ca***

Introduction

Glaucoma is a vision impairing disease of the eye caused by increased pressure, which leads to characteristic changes on colour fundus photography. Specifically, the optic cup to disc ratio and the inferior superior nasal temporal rule are commonly used to identify the presence of glaucoma. Both of these findings can be seen on the optic disc, and therefore, it is advantageous to first crop the full retinal image to isolate the optic disc prior to training a classification model [1–3].

Material and Methods

Fig. 1 shows the overall workflow. The full fundus image is preprocessed to isolate the optic disc, which is then passed through an ensemble of classifiers to make the final prediction. The full fundus image is also passed through an autoencoder to determine ungradability.

Dataset. The models were trained on the official training dataset provided by the competition organisers: the Rotterdam EyePACS AIROGS dataset [4]. The dataset consists of 113,893 retinal fundus images, of which 101,442 images were made available to participants as training data and the remainder were used for competition evaluation. Each training image is labelled as referable or non referable glaucoma. For a discussion of the data collection and labelling process, see the competition website [5]. Five percent of the provided training data was randomly reserved as a holdout test set for evaluating classification models. However, the images used to train the optic disc detection model came from both training and testing sets. For the purpose of this paper, the testing data reserved by the competition organizers to evaluate performance is referred to as the submission set. Training, validation and testing data refer to the partitioning of the training data provided to participants.

Optic Disc Detection. As the training labels do not contain information regarding the location of the optic disc, a semi-automated approach was used to generate labels to train an optic disc detection algorithm. First, all fundus images were cropped to 299x299 image size. If necessary, images were padded to ensure a square aspect ratio. Next, a heuristic algorithm based on Otsu thresholding, and exploiting differences in colour channels and a priori estimates about the size and shape of the optic disc was used to draw a loose bounding box (50% of image width) around the optic disc. The implementation was based on previously published approaches [6, 7]. As there was no ground truth mask provided, it was not possible to quantify accuracy, however, the authors estimate that the algorithm adequately detected the optic disc in 80% of images. The authors then manually labelled the optic disc on 277 processed images with a tight bounding box. A modified ResNet34 [8, 9] was trained and used to detect the tight bounding box on 1700 images. The results were visually inspected, and unacceptably labelled images were manually relabelled. The resulting correctly labelled and manually relabelled images were then used to retrain the ResNet34 model. This process was iterated once more. The fully trained ResNet34 was used to detect the bounding box on approximately 4000 images, and the labels were transferred from the processed images to the original images. Finally, a YoloV5 [10] model was trained in 20 epochs on the labelled images for optic disc detection at 288x288 image size. Of the 4088 images used to train YoloV5, 735 were labelled manually. The authors estimate that the YoloV5 model leads to adequate isolation of the optic disc in >99% of images.

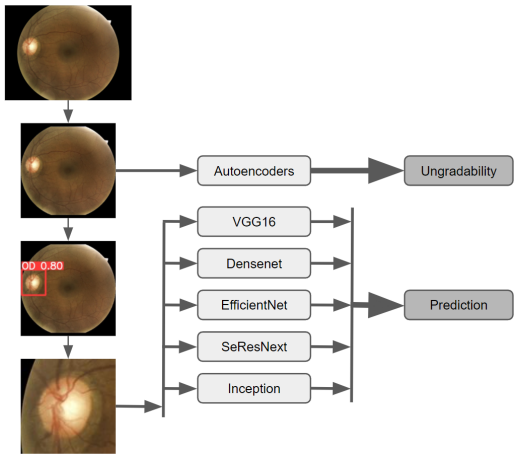

Fig. 1. Workflow

Glaucoma Classifier. Five different convolution neural network classifiers were used: SeResNext50, [11, 12] VGG16 [9, 13], DenseNet161, [9, 14] EfficientNetB5 [15, 16], EfficientNetB7 [15, 16] and InceptionV3 [9, 17]. A dropout layer was inserted before the last linear layer in all models. Weights pretrained on ImageNet [18] were loaded for all models. The models were trained on the optic discs detected by the YoloV5 model. Positive samples were weighted 30:1 compared to the negative samples corresponding to the distribution within the training data. Hyperparameter tuning was used to select the optimal last layer dropout, optimizer learning rate, optimizer weight decay and optimizer momentum for SGD. Models were trained in two stages, and details about the training process can be found in the Appendix. Ninety five percent of the training set was used to train the models and 5% was used for validation (corresponding to 90.3% and 4.75% of the total training data provided by the competition), split randomly. All classifiers were trained in PyTorch [9]. The first submission consisted of an ensemble of SeResNext50, VGG16, DenseNet161 and EfficientNetB5. In the ensemble for the second submission, the EfficientNetB5 was replaced by an EfficientNetB7, and the InceptionV3 model was added to the ensemble. In the third submission, the only change was the removal of the VGG16 model from the ensemble.

Estimating Ungradability. In the first submission, ungradability was set to the standard deviation of ensemble predictions. The cutoff for converting the continuous standard deviation to a binary prediction of ungradability was chosen to be the 95^{th} percentile value (0.3). For the second submission, ungradability (δ) was defined by:

$$\delta = (1-c) * s * \sigma$$

where c is the confidence in detecting the optic disc provided by YoloV5, σ is the standard deviation and s is a scaling factor defined by:

$$s = \begin{cases} 2 * y & \text{if } 0 \leq y \leq 0.5 \\ -2 * (y-1) & \text{Otherwise} \end{cases}$$

in which y is the average of all ensemble model predictions. The cutoff was set to the 99^{th} percentile value (0.075). For the third submission, an autoencoder and a variational autoencoder (VAE) [19] were trained to estimate ungradability. The autoencoder consists of an encoder with 7 convolution layers that iteratively halves the dimensions of the input image (256x256) prior to decoding and was trained with 150 epochs. Sample outputs from the autoencoder are shown in Fig. 2. The VAE consists of a ResNet18 encoder and decoder [20] with weights pretrained on CIFAR-10 [21] and trained for 50 epochs. The input image size to the VAE was 256x256. Ungradability in the third submission was defined by:

$$\delta = (1-c) * s * p_{autoencoder} * p_{vae}$$

where $p_{autoencoder}$ and p_{vae} refer to the mean squared error between the input and output of the autoencoder and VAE respectively.

Evaluation and Results

The results of the ensemble algorithm is summarized in Table 1. The drop in performance metrics in the submission phase relative to the testing phase indicates the presence of model overfitting. The addition of the InceptionV3 and the EfficientNetB7 model increased performance both in the testing phase and the submission phase. The autoencoders outperformed the standard deviation of predictions at estimating ungradability. The performance of individual models and the hyperparameters used to train the final models are shown in the Appendix.

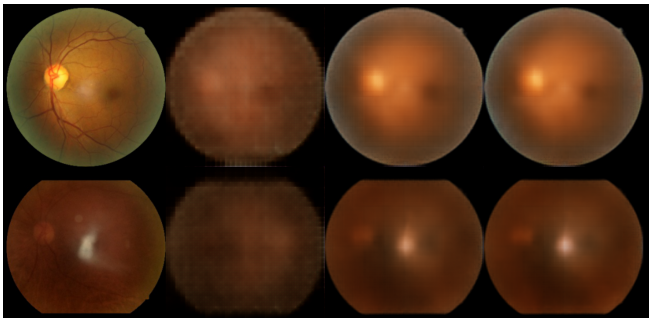

Fig. 2. Autoencoder reconstructions in validation set. Left to right: Original, 1 epoch, 50 epochs, 150 epochs. Validation loss at 1 epoch, 50 epochs and 150 epochs are 0.01102, 0.001437 and 0.001256 respectively.

Table 1. Summary of Performance Metrics

Submission	1	2	3
Testing pAUC	0.9423	0.9637	0.9654
Testing TPR_{95}	0.9396	0.9732	0.9664
Submission pAUC	0.8735	0.8941	0.8935
Submission TPR_{95}	0.8188	0.8750	0.8875
Ungradability Kappa	0.4043	0.4086	0.3712
Ungradability AUC	0.8335	0.8573	0.8707

Discussion and Conclusion

There was little to no overfitting between the validation phase and the test phase, and significant overfitting between the test phase and submission phase. One likely contributing cause to overfitting is the use of images from all of training, validation and test phases to train the optic disc detection model. Computational resources was a limiting factor in this competition. Due to the number of images available in the training data, it was necessary to train the models on images with smaller dimensions (less than 300x300). Using a preprocessing model to detect the optic disc maximizes the amount of relevant information that is contained in a small image. The authors predict that using larger image sizes and training for more epochs will yield better results.

Bibliography

1. Javier Civit-Masot, Manuel J. Domínguez-Morales, Saturnino Vicente-Díaz, and Anton Civit. Dual machine-learning system to aid glaucoma diagnosis using disc and cup feature extraction. *IEEE Access*, 8:127519–127529, 2020. doi: 10.1109/ACCESS.2020.3008539.
2. Rutuja Shinde. Glaucoma detection in retinal fundus images using u-net and supervised machine learning algorithms. *Intelligence-Based Medicine*, 5:100038, 2021. ISSN 2666-5212. doi: https://doi.org/10.1016/j.ibmed.2021.100038.
3. Ruben Hemelings, Bart Elen, João Barbosa-Breda, Matthew B Blaschko, Patrick De Boever, and Ingeborg Stalmans. Deep learning on fundus images detects glaucoma beyond the optic disc. *Scientific Reports*, 11(1):1–12, 2021.
4. Coen de Vente, Koenraad A. Vermeer, Nicolas Jaccard, Bram van Ginneken, Hans G. Lemij, and Clara I. Sánchez. Rotterdam eyepacs airogs train set, December 2021. The previous version was split into two records. This new version contains all data and the second record is deprecated.
5. Airogs - grand challenge. https://airogs.grand-challenge.org/.
6. M Elena Martinez-Perez, Nicholas Witt, Kim H Parker, Alun D Hughes, and Simon AM Thom. Automatic optic disc detection in colour fundus images by means of multispectral analysis and information content. *PeerJ*, 7:e7119, 2019.
7. Behdad Dashtbozorg, Ana Maria Mendonça, and Aurélio Campilho. Optic disc segmentation using the sliding band filter. *Computers in biology and medicine*, 56:1–12, 2015.
8. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.
9. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
10. Glenn Jocher, Alex Stoken, Jirka Borovec, NanoCode012, ChristopherSTAN, Liu Changyu, Laughing, tkianai, Adam Hogan, lorenzomammana, yxNONG, AlexWang1900, Laurentiu Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, Francisco Ingham, Frederik, Guilhen, Hatovix, Jake Poznanski, Jiacong Fang, Lijun Yu, changyu98, Mingyu Wang, Naman Gupta, Osama Akhtar, PetrDvoracek, and Prashant Rai. ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements, October 2020.
11. Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks, 2019.
12. Pretrained models for pytorch. https://github.com/Cadene/pretrained-models.pytorch.
13. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.
14. Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018.
15. Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.
16. Efficientnet pytorch. https://github.com/lukemelas/EfficientNet-PyTorch.
17. Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision, 2015.
18. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
19. Diederik P Kingma and Max Welling. Auto-encoding variational bayes. 2014.
20. William Falcon and Kyunghyun Cho. A framework for contrastive self-supervised learning and designing a new approach. *arXiv preprint arXiv:2009.00104*, 2020.
21. Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research).

Appendix 1

Table A1. Performance of Individual Models

	Submission	1	2	3
pAUC	SeResNext	0.8921	0.9348	0.9348
	DenseNet	0.9489	0.9489	0.9489
	VGG16 [a]	0.9107	0.9107	-
	EfficientNet [b]	0.8957	0.9547	0.9547
	Inception [c]	-	0.9355	0.9355
	Mean [d]	0.9423	0.9637	0.9654
TPR_{95}	SeResNext	0.8523	0.9195	0.9195
	DenseNet	0.9530	0.9530	0.9530
	VGG16 [a]	0.9060	0.9060	-
	EfficientNet [b]	0.8523	0.9597	0.9597
	Inception [c]	-	0.9262	0.9262
	Mean [d]	0.9396	0.9732	0.9664

[a] VGG16 was removed in the last submission.
[b] The B5 variant was used for the first submission and the B7 for subsequent submissions.
[c] Inception was not included in the first submission.
[d] Mean is determined by calculating the metric on the average of individual predictions.

Table A2. Hyperparameters

		SeResNext	DenseNet	VGG16	EfficientNetB7	Inception [a]
Stage 1	Image Size	120	120	120	120	299
	Epochs	4	4	4	4	10
	Optimizer	AdamW	AdamW	AdamW	AdamW	SGD
	Learning Rate	0.000016	0.000018	0.000008	0.000043	0.000003
	Dropout	0.0373	0.0828	0.3034	0.1090	0.4383
	Weight Decay	0.0014	0.0001	0.0002	0.0008	0.0101
	Momentum	-	-	-	-	0.776716
Stage 2	Image Size	224	-	-	224	-
	Epochs [b]	5		-	5	-
	Optimizer	AdamW	-	-	AdamW	-
	Learning Rate	0.000029	-	-	0.000036	-
	Dropout	0.6702	-	-	0.6743	-
	Weight Decay	0.0013	-	-	0.0032	-
	Momentum	-	-	-	-	-

[a] Inception was trained in a single stage of 10 epochs.
[b] The number of epochs in this row refers to the number of subsequent epochs.