Deep learning ensemble framework for multiclass diabetic retinopathy classification

Mudit Saxena[1], Pratap Narra[1], Mayank Saxena[2], Rakhi Saxena[3]
[1]Department of Computer Science, New York University, New York, USA
[2]Department of Computer Science, Columbia University, New York, USA
[3]Department of Computer Science, Deshbandhu College, University of Delhi, New Delhi, India

Article Info

Article history:

Received Oct 19, 2023
Revised Feb 2, 2024
Accepted Feb 29, 2024

Keywords:

Convolutional neural networks
Explainability
Deep learning
Diabetic retinopathy
Uncertainty

ABSTRACT

Diabetic retinopathy (DR) is the leading cause of blindness among adults and has no visible symptoms. Early detection is the key to prevent vision loss. Computer-aided deep learning using convolutional neural networks (CNN) have recently gained momentum for DR diagnosis as the cost can be significantly reduced while making the diagnosis more accessible. In this work, we present a fully automated framework DR network (DRNET) that fuses both image texture features and deep learning features to train the CNN model. The framework aggregates predictions from three CNN models using ensemble learning for more precise and accurate DR diagnosis when compared to standalone CNN. To strengthen the confidence of medical practitioners in acceptance of automated DR diagnosis, we extend the DRNET framework by producing model uncertainty scores and explainability maps along with the classification results.

Corresponding Author:

Rakhi Saxena
Department of Computer Science, Deshbandhu College, University of Delhi
New Delhi, India
Email: rsaxena@db.du.ac.in

1. INTRODUCTION

Diabetic retinopathy (DR) is a serious condition that can potentially cause vision loss. People with diabetes are susceptible to this condition which makes it the number one cause of blindness among adults and diabetes is projected to affect around 600 million people worldwide by 2040 [1]. However, DR does not have any noticeable symptoms, and thus detecting it early is very important to prevent vision impairment [2]. DR is detected by the appearance of different types of lesions on a retina image. These lesions are micro aneurysms (MA), haemorrhages (HM), and soft and hard exudates (EX) out of which MA is the earliest sign of DR that appears as small red round dots on the retina caused by the weakness of the vessel's walls. Detection of these abnormalities requires the intervention of skilled ophthalmologists or trained clinicians to manually examine retinal images and offer diagnosis. Given the fact that medical diagnosis is usually expensive and requires costly equipment, there is a dire need for a cheap and viable solution [1]. Recently, DR detection using convolutional neural networks (CNN) has gained momentum [2]–[6]. A CNN [7], [8] is a type of neural network that assigns weight and biases to different objects in a particular input image. With the use of appropriate filters, CNNs can capture spatial and temporal connections in an image. CNN architectures generally fit better to image datasets as image data usually has a large number of less-important features. CNNs are composed of one input and output layer with multiple hidden layers. These hidden layers are made of convolutional layers. Three popular CNN-ResNet [9], AlexNet [10], GoogleNet [11] have demonstrated

decent accuracy (≈78%) for DR detection [1]. However, we identify three gaps hindering the implementation and real-world deployment of CNN models for DR diagnosis. We enumerate these:
- Accuracy provided by CNN based DR diagnosis is much less than that acceptable for medical diagnosis. Therefore, medical practitioners are reluctant to believe such diagnosis because of the serious implications of incorrect model predictions.
- In high risk areas, such as medical imaging tasks, it is important to know not only the accuracy of the model's prediction but also how certain the prediction is. If a prediction comes with a high uncertainty, the end user can take that into account while making the diagnosis.
- End-user's confidence and trust in the automated CNNs based prediction is not high. In that context, interpretability and explainability of the model's outcome are crucial. If the model can tell the user which regions of the image impacted the model's prediction, the trust of the medical practioners in the model's prediction will be considerably improved.

In light of the above mentioned gaps in using deep learning strategies for DR diagnosis, we propose a CNN based ensemble framework, named DR network (DRNET), to detect DR that attempts to bridge the gaps for enabling translation of automated DR diagnosis to real-life clinical work.

2. MATERIALS AND METHODS

The proposed framework DRNET for multiclass DR classification provides high accuracy firstly, by combining image texture features with deep learning features and secondly, by constructing best ensemble models from three CNN-ResNet50, AlexNet, and GoogleNet. Further, the pipeline incorporates both uncertainty estimation and explainability maps. A misdiagnosis is flagged when the uncertainty score is high and/or the explainability map is unsatisfactory. *To the best of our knowledge, this is the first attempt to fuse image texture and deep learning features and to include uncertainity scores and explainability maps for DR diagnostics.*

2.1. Datasets used

The APTOS 2019 blindness detection dataset [12] is used for the training experiments. The dataset is a part of the Kaggle competition, it has 35126 retinal images. The images have been graded by medical practitioners into 5 classes of DR, namely, No-DR, Mild-DR, Moderate-DR, Severe-DR, and Proliferative-DR.

2.2. Proposed method

This section describes the proposed fully automated deep learning framework called DRNET to diagnose DR. A summary of the method:
- The first step of the DRNET pipeline begins with improving accuracy of standalone CNN models by preprocessing (pp) the images and applying contrast enhancement algorithms to accentuate the lesions on the images [13]. Subsequently, feature extraction is carried out to determine image texture features (using GCLM) and deep learning features (using transfer learning). Three CNN models are trained after determining the best hyperparameters (batch size, learning rate, and the optimizer) for the CNN models using algorithms proposed in [14], [15]. Cosine annealing is further deployed to improve model performance by tweaking the learning rate. The resultant probabilities from three CNN for the five DR classes are combined using ensembling techniques.
- Considering the severity of DR, there is a need to minimise the number of false positives the model produces. For this purpose, DRNET exploits "uncertainty" to measure the uncertainty of the predictions [16]. Predictions paired with uncertainty scores would permit diagnostics with a high degree of confidence and trust. If the model is not sure i.e the degree of confidence is quite low for a prediction, flagging the prediction as 'not sure' is more responsible and safer compared to an incorrect prediction.
- We incorporate explainability in the framework so that users can understand the prediction of the framework better. The explainability maps produced by the DRNET framework will enable users to figure out which regions of the input images are given more importance than others by different architectures and make appropriate decision. Figure 1 shows the DRNET framework with the steps explained in detail in the following subsections.

2.3. Image preprocessing

As a first step of image preprocessing, the images are trimmed to remove the uninformative backspace. For this row of pixels in which a specified fraction of pixels are above a threshold are removed. This results in mainly eliminating the black/dark segments around the eye. The images are then resized into 227×227 while maintaining the aspect ratio so as to reduce the training overhead of the deployed model. Gaussian filtering is used for noise reduction by convolving the two dimension Gaussian distribution function given below [17] with the image:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)} \tag{1}$$

where σ is the standard deviation of the distribution.

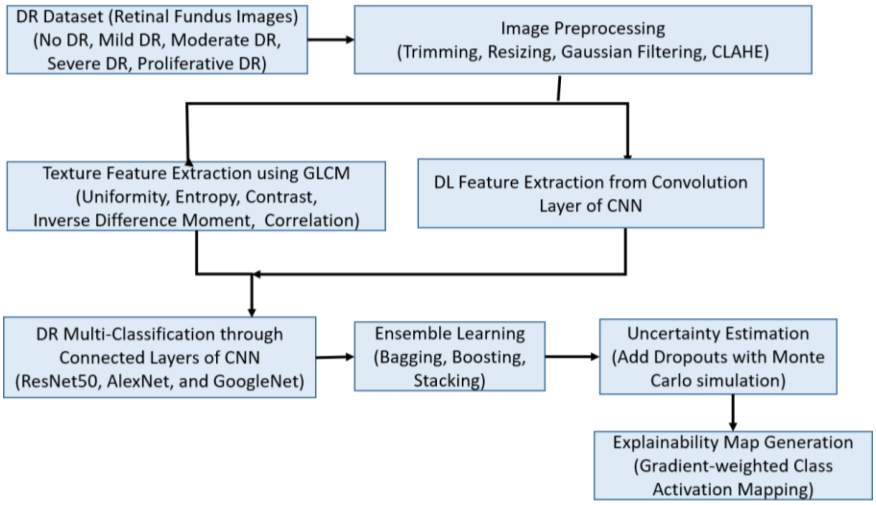


Figure 1. Block diagram of the proposed DRNET framework

Medical images such as retinal fundus images and x-rays generally have a low-contrast, this makes it difficult for CNNs to identify the features properly in order to classify an image. We use the contrast limited adaptive histogram equalization (CLAHE) algorithm to enhance the retinal images and use the enhanced images to train our baseline models. CLAHE was proposed to enhance the contrast of images and has proven to be a good preprocessing technique for medical images. The image is partitioned into contextual regions, then histogram equalization is applied on the each region by the CLAHE algorithm [13].

2.4. Feature extraction

Two types of features are extracted from the processed images.

– Texture features encapsulate the spatial variation of pixel values within the image. Gray level co-occurrence matrix (GLCM) method [18] is used for extracting second order statistical texture features of the image. A GLCM matrix (G) is a nXn matrix where n is the number of gray levels in an image. The gij element of G is the relative frequency with which a pair of pixels, separated by a specified spatial relationship, occur within a given neighbourhood, one with value 'i' and the other with value 'j'. Four statistical texture features, namely, uniformity, entropy, contrast, inverse difference moment, and correlation as identified in [19] are computed from the GCLM.
– Deep learning features are extracted using transfer learning from the convolution layer of pre-trained CNN models-AlexNet, GoogleNet, and ResNet. This process takes the models trained on a very large dataset (ImageNet [20]) and transfers the learnings to the smaller dataset (DR APTOS dataset).

2.5. Multiclass classification using CNN

The texture and deep learning features are fused and fed into the fully connected layers of the CNN to obtain the predicted probabilities of the five DR classes using techniques given in [21]. For training the CNN, optimal model hyperparameters and learning rate were identified as explained:

– Hyperparameter tuning: to train any deep learning model there are a number of hyperparameters that have to be set beforehand, but trying out all combinations of hyperparameters to find the best set can be compute-intensive as it requires training a lot of deep learning models. In this work, we use the asynchronous successive halving algorithm (ASHA) [15] to perform hyperparameter tuning. ASHA is designed to exploit asynchrony and maximize the parallelism on the successive halving algorithm (SHA) [15]. SHA starts with allocating a uniform budget to all the candidate hyperparameter configurations, i.e if learning rate and momentum are to be picked from the following sets 1e-1,1e-2 and 0.7,0.9 respectively, then there are a total of 4 initial candidate hyperparameter configurations. The model performance is evaluated on all the candidate configurations. Each round of promotion is called a rung.

Later the top half candidate configurations will be promoted to the next rung for further evaluation and their budget is doubled, this process is repeated until one configuration remains. In SHA the model must be evaluated on the entire set of configurations to select the next half. ASHA removes this bottleneck [15] by assigning configurations to workers, when a worker finishes a job instead of waiting for all the workers to finish. The algorithm identifies the configurations in the top half of each rung that can be promoted to the next one, else a configuration from the lowest rung is added to the worker [15].

– Cosine annealing: since cosine annealing [14] is known to work well with SGD optimisers, to further enhance the performance of our model, we have used cosine annealing learning rate scheduler. This scheduler constantly varies the learning rate such that it starts with a large learning rate and then reduces it to a certain minimum value before rapidly increasing it again.

2.6. Ensemble learning for prediction

The actual application in a clinical setting depends on the ability of the model to predict with high accuracy, which decides whether the framework is effective. By combining several individual models, it is possible to obtain much more precise performance [22], [23] as it enables us to yield the individual advantages of all the models. In this work we experimented with various ensembling methods-bagging, boosting and stacking. Best results were obtained by bagging models. We bag models in two ways, through hard-voting, and soft-voting. In hard voting, we select the predicted class as the class predicted by maximum number of models. On the other hand, in soft voting, we sum the probabilities (i.e the output of the last layer of each model just before the argmax), and choose the class that has the highest probability.

2.7. Incorporating explainability

In this work, we have used gradcam [24] to find the salient regions which impacted a model's decision. It is important for us to not treat any deep learning model as a black box and understand the functioning of our model. Gradcam, short for "gradient-weighted class activation mapping", is a tool that produces visual explanations of the decision model of any CNN-based architectures [24]. Gradcam allows us to see what the model considers as important regions in an input image that is used for prediction. In the context of our problem, we use gradcam to identify what our CNN networks consider as the important features in the image [24]. Since our input images are fundus retinal images, gradcam allows us to visually understand which part of the retina the model focuses more on. It gives clear reasoning for the model's decision hence making the model more dependable for diagnosis.

2.8. Uncertainty estimation

Bayesian neural networks are commonly used to measure the uncertainty of the prediction. In bayesian neural networks weights are not just numbers, they are probability distributions, and which aid in throwing some light on model uncertainty. However, bayesian neural networks are compute-intensive. An alternative method to estimate the uncertainty score of the prediction is to add dropouts to a model [25]. Then during the testing phase, an image is passed multiple times through the network, and the dropouts which are activated during the testing phase drop different neurons randomly and the model's predictions variations are observed [25]. This is mathematically equivalent to measuring the uncertainty using bayesian neural networks [25]. Passing the image multiple times is called monte carlo simulation. The shannon entropy equation defined below is used to measure uncertainty.

$$E = -\sum\left(p(i) \times log_2\left(p(i)\right)\right) \qquad (2)$$

3. RESULTS AND DISCUSSION

We perform our experiments with the three baseline models viz. AlexNet, GoogleNet, and ResNet and an optimised ensemble of the three models. In the following subsections, we report the results.

3.1. Data preprocessing using CLAHE and hyperparameter tuning

We apply CLAHE as pre-processing techniques to the training dataset (APTOS) as described in section 2.3. Table 1 depicts the difference between test accuracies of models trained with and without CLAHE preprocessing, it is clear that the preprocessing played a key role in improving the model performance. As described earlier in section 2.5., we start the experiments by hyperparameter tuning on our individual baseline models on the test dataset. For hyperparameter tuning, we used grid search to find the best set of hyperparameters-batch size, learning rate and the optimizer-for all 3 architectures (AlexNet, ResNet, and GoogleNet). Table 2 contains the best set of hyperparameters selected by the ASHA algorithm.

Table 1. Model test accuracies without and with pp

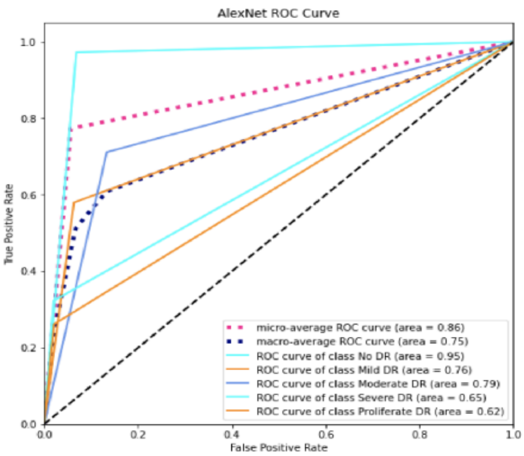
Model	Test accuracy (without pp) (%)	Test accuracy (with pp) (%)
AlexNet	71.48	77.48
GoogleNet	74.89	73.48
ResNet50	76.39	78.17

Table 2. Best hyperparameters selected by the ASHA algorithm

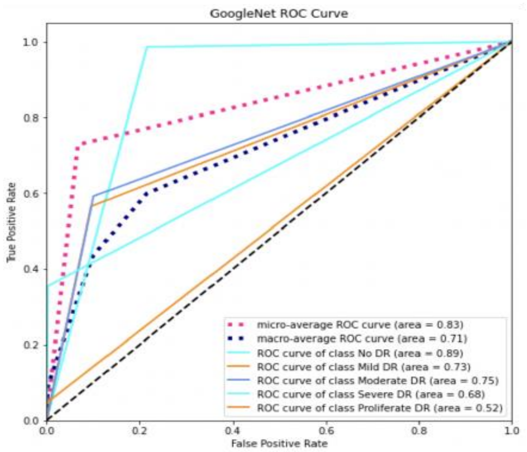
Model	Learning rate	Batch size	Optimizer
AlexNet	2.63994e-05	128	Adam
GoogleNet	0.0008695	64	Adagrad
ResNet50	0.011097376	32	Adam

3.2. Model accuracy with standalone CNN and ensemble model

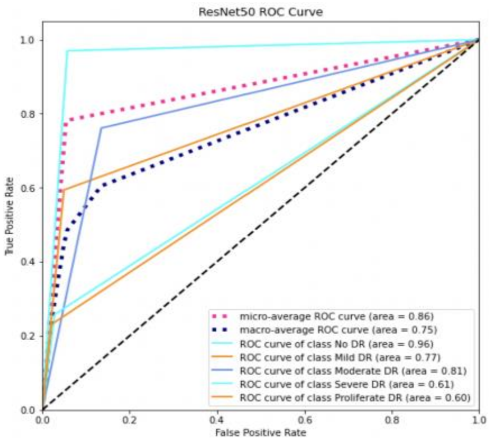
Using the best set of hyperparameters for each architecture, and enhancing images using CLAHE, each model was trained further for 200 epochs. The ROC curve graphs in Figure 2 shows the performance of tested CNN models. The area under the ROC curve measures the performance across all classification threshholds and the values for each DR class as well as micro and macro averages are mentioned in the figure. Figure 2(a) shows the performance of the AlexNet model. Figure 2(b) shows the performance of the GoogleNet model and Figure 2(c) shows the performance of the ResNet50 model. While comparing baseline single models, it’s clear that ResNet50 has superior performance compared to the other two models. The models were combined using ensemble techniques (boosting, bagging, and stacking) as described in section 2.6. The results are displayed in Table 3. We obtained significant improvement in accuracy by using ensembles. Most effective prediction was achieved using bagging.



(a)



(b)



(c)

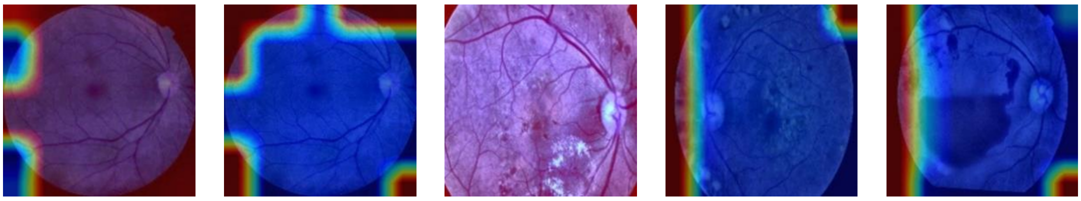
Figure 2. Receiver operating characteristic (ROC) curves for each model; (a) AlexNet, (b) GoogleNet, and (c) ResNet50

Table 3. Accuracy of ensemble of CNN models

Ensemble of CNN models	Accuracy (%)
Boosting ensemble	90.2
Stacking ensemble	88.1
Bagging ensemble	97.3

3.3. Generation of explainable images

We generate images using gradcam and show one image from each of the classification class in Figure 3 for our best single model which is ResNet50. The red areas in these figures represent the salient regions. In case of the No-DR retinal image (Figure 3(a)), the key area indicating the absence of DR is almost the entire image. For the Mild-DR (Figure 3(b)) and Moderate-DR images (Figure 3(c)), the salient regions that enabled the classification are present the image edges. However, as we can see that in two of the images (Figure 3(d) and Figure 3(e) specifically), the salient regions are outside the retinas which makes the model's prediction more untrustworthy. Evidently, through the gradcam images, the model's behaviour becomes clearer as it demarcates parts of the image the prediction is focusing more on and thus bolster confidence in the results.

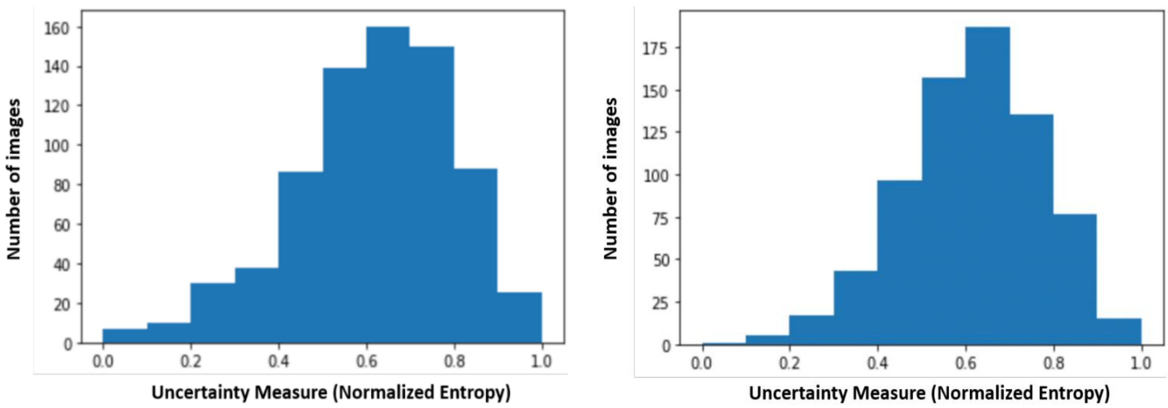


(a) (b) (c) (d) (e)

Figure 3. Gradcam images for all classes for the ResNet50 models, the red areas are the salient regions for the models; (a) No-DR, (b) Mild-DR, (c) Moderate-DR, (d) Severe-DR, and (e) Proliferative-DR

3.4. Uncertainty computation using entropy

Shannon entropy is used to measure the uncertainty of the models. Figures 4(a) and 4(b) depict the uncertainty distribution of both the top models, namely, AlexNet, and ResNet50 respectively where the x-axis indicates the entropy and the y-axis is the number of images. From the figures, it is clear that the images classified by AlexNet are more uncertain compared with those classified ResNet50 and we conclude that ResNet50 is the more certain model. We also show in Table 4 the uncertainty measure of 3 test images, it can be observed that wrong predictions have higher uncertainty compared to the correct predictions. Hence when a model's prediction has higher uncertainty measure then there is a higher chance that it is a incorrect prediction and is flagged as such by DRNET.



(a) (b)

Figure 4. Uncertainty distribution for; (a) AlexNet and (b) ResNet50

Table 4. Uncertainty measure of 3 test images

Image	Entropy	Prediction	Ground truth
Test image A	1.51	0	1
Test image B	0.81	2	2
Test image C	1.47	1	0

4. CONCLUSION

In this work we provide an end to end solution for DR diagnosis, along with uncertainty estimation and explainability. We have studied the affect of preprocessing and hypermarameter tuning on the model performance and conclude that models trained on images preprocessed with CLAHE perform better. We demonstrate significant improvement in accuracy of DR classification by using ensembles of the three baseline models with the bagging technique turning out to be the ideal solution to build the ensemble. Then we have used shannon entropy to measure the uncertainty of the test set, our observations show that images which were wrongly predicted have higher uncertainty compared to the ones which were rightly predicted. The gradcam maps were used to highlight the salient regions of the images, these give insight about the regions of the image with impacted the model's decision. We conclude that automated CNN can be deployed in real life settings if explainability maps and uncertainity scores are incorporated in the pipeline.

ACKNOWLEDGEMENT

We acknowledge the support received from Deshbandhu College, University of Delhi for carrying out research work of this paper.

REFERENCES

[1] J. Caicho *et al.*, "Diabetic Retinopathy: Detection and Classification Using AlexNet, GoogleNet and ResNet50 Convolutional Neural Networks," in *Communications in Computer and Information Science*, pp. 259–271, 2022, doi: 10.1007/978-3-030-99170-8_19.

[2] N. M. Al-Moosawi and R. S. Khudeyer, "ResNet-n/DR: Automated diagnosis of diabetic retinopathy using a residual neural network," *Telecommunication Computing Electronics and Control*, vol. 21, no. 5, pp. 1051–1059, 2023, doi: 10.12928/TELKOMNIKA.v21i5.24515.

[3] M. Z. Atwany, A. H. Sahyoun, and M. Yaqub, "Deep Learning Techniques for Diabetic Retinopathy Classification: A Survey," *IEEE Access*, vol. 10, no. 28, pp. 28642–28655, 2022, doi: 10.1109/ACCESS.2022.3157632.

[4] N. M. A. Tajudin *et al.*, "Deep learning in the grading of diabetic retinopathy: A review," *IET Computer Vision*, vol. 16, no. 8, pp. 667–682, 2022, doi: 10.1049/cvi2.12116.

[5] M. Saxena, P. Narra, and R. Saxena, "DRNET: Improved Deep Learning Solution to detect Diabetic Retinopathy by incorporating Explainability, Uncertainty and Generalisation," 2022, doi: 10.21203/rs.3.rs-1954769/v1.

[6] R. Kommaraju and M. S. Anbarasi, "Diabetic retinopathy detection using convolutional neural network with residual blocks," *Biomedical Signal Processing and Control*, vol. 87, p. 105494, 2024, doi: 10.1016/j.bspc.2023.105494.

[7] and Y. B. Y. LeCun, H. Patrick, L. Bottou, "Object recognition with gradient-based learning," *in Shape, Contour and Grouping in Computer Vision. Berlin, Heidelberg: Springer-Verlag*, p. 319, 1999.

[8] D. A. Anggoro, A. A. T. Marzuki, and W. Supriyanti, "Classification of Solo Batik patterns using deep learning convolutional neural networks algorithm," *Telecommunication Computing Electronics and Control*, vol. 22, no. 1, pp. 232–240, 2024, doi: 10.12928/telkomnika.v22i1.24598.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017, doi: 10.1145/3065386.

[11] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 1–9, 2015, doi: 10.1109/CVPR.2015.7298594.

[12] G. Quellec, M. Lamard, P. M. Josselin, G. Cazuguel, B. Cochener, and C. Roux, "Optimal Wavelet Transform for the Detection of Microaneurysms in Retina Photographs," *IEEE Transactions on Medical Imaging*, vol. 27, no. 9, pp. 1230–1241, 2008, doi: 10.1109/TMI.2008.920619.

[13] K. Zuiderveld, "Contrast Limited Adaptive Histogram Equalization," *Graphics Gems*, pp. 474–485, 1994, doi: 10.1016/b978-0-12-336156-1.50061-6.

[14] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," *5th International Conference on Learning Representations, ICLR 2017-Conference Track Proceedings*, 2017.

[15] L. Li *et al.*, "A System for Massively Parallel Hyperparameter Tuning," *in Proceedings of Machine Learning and Systems, I. Dhillon, D. Papailiopoulos, and V. Sze*, vol. 2, pp. 230–246, 2018, [Online]. Available: http://arxiv.org/abs/1810.05934

[16] C. Leibig, V. Allken, M. S. Ayhan, P. Berens, and S. Wahl, "Leveraging uncertainty information from deep neural networks for disease detection," *Scientific Reports*, vol. 7, no. 1, p. 17816, 2017, doi: 10.1038/s41598-017-17876-z.

[17] S. Dasari, B. Poonguzhali, and M. S. Rayudu, "An efficient machine learning approach for classification of diabetic retinopathy stages," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 1, pp. 81–88, 2023, doi: 10.11591/ijeecs.v30.i1.pp81-88.

[18] R. M. Haralick, I. Dinstein, and K. Shanmugam, "Textural Features for Image Classification," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973, doi: 10.1109/TSMC.1973.4309314.

[19] P. Mohanaiah, P. Sathyanarayana, and L. Gurukumar, "Image Texture Feature Extraction Using GLCM Approach," *International Journal of Scientific and Research Publication*, vol. 3, no. 5, pp. 1–5, 2013.

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009*

IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 248–255, 2010, doi: 10.1109/cvpr.2009.5206848.
[21] G. Bogacsovics, J. Toth, A. Hajdu, and B. Harangi, "Enhancing CNNs through the use of hand-crafted features in automated fundus image classification," *Biomedical Signal Processing and Control*, vol. 76, p. 103685, 2022, doi: 10.1016/j.bspc.2022.103685.
[22] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan, "Ensemble deep learning: A review," *Engineering Applications of Artificial Intelligence*, vol. 115, p. 105151, 2022, doi: 10.1016/j.engappai.2022.105151.
[23] A. Francis and I. A. Pandian, "Ensemble learning approach for multi-class classification of Alzheimer's stages using magnetic resonance imaging," *Telecommunication Computing Electronics and Control*, vol. 21, no. 2, pp. 374–381, 2023, doi: 10.12928/TELKOMNIKA.v21i2.23352.
[24] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," in *Proceedings of the IEEE International Conference on Computer Vision*, IEEE, 2017, pp. 618–626, doi: 10.1109/ICCV.2017.74.
[25] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," *33rd International Conference on Machine Learning, ICML 2016*, vol. 3, pp. 1651–1660, 2016.

BIOGRAPHIES OF AUTHORS

Mudit Saxena is a Software Engineer at Imprint, New York. He completed his Masters in Computer Science in 2022 from the New York University, New York, USA. His areas of interest include machine learning, big data analytics, and deep learning. He can be contacted at email: ms12768@nyu.edu.

Pratap Narra is a Software Engineer at APi Group, New York. He completed his Masters in Computer Science in 2023 from the New York University, New York, USA. His areas of interest include machine learning, natural language processing, and deep learning. He can be contacted at email: fpn2158@nyu.edu.

Mayank Saxena is a Software Engineer at Amazon, New York, USA. He completed his Masters in Computer Science in 2018 from the Columbia University, New York. His areas of interest include social network analysis, machine learning, big data analytics, and deep learning. He can be contacted at email: ms5736@columbia.edu.

Rakhi Saxena is an Associate Professor in the Department of Computer Science, at Deshbandhu College, University of Delhi, India. She received her Ph.D. from the Department of Computer Science, University of Delhi, in 2020. Her research areas of interest include social network analysis, graph mining, recommender systems, multilayer networks, and big data analytics. She can be contacted at email: rsaxena@db.du.ac.in.