Computational and Structural Biotechnology Reports 2 (2025) 100035

Contents lists available at ScienceDirect

Computational and Structural Biotechnology Reports

journal homepage: www.sciencedirect.com/journal/csbr

Research Article

Real-time deep learning for multi-label retinal disease diagnosis with embedded system

Feudjio Ghislain [a,b,*], Saha Tchinda Beaudelaire [a], Romain Atangana [a,c], Tchiotsop Daniel [a]

[a] Research Unit of Automation and Applied Computer (UR-AIA), Electrical Engineering Department of IUT-FV, University of Dschang, P.O. Box: 134, Bandjoun, Cameroon
[b] Research Unit of Condensed Matter of Electronics and Signal Processing (UR-MACETS), Department of Physics, Faculty of Sciences, University of Dschang, P.O. Box 67, Dschang, Cameroon
[c] Department of Computer Science, Higher Teacher Training College, University of Bertoua-Cameroon, P.O.Box 652, Bertoua, Cameroon

ARTICLE INFO

Keywords:
Embedded board
Real-time classification
Lightweight CNN model
Wavelet transforms

ABSTRACT

Background: Ophthalmological consultations are essential for the early detection of retinal pathologies. The development of advanced diagnostic devices that combine portability, ease of access, and energy efficiency offers significant benefits, especially in rural or peri-urban areas, which often struggle with limited medical resources and time constraints.
Method: In this study, we develop a classification platform for various retinal diseases using a Raspberry Pi 4 board. The system relies on a lightweight deep learning model based on convolutional neural networks (CNNs). To enhance the model's efficiency, we integrate biorthogonal wavelet transforms, which facilitate the effective extraction of relevant features from the input images. This approach reduces computational complexity while maintaining the quality of the extracted information. The model, optimized for real-time deployment, was trained on a dataset containing 4303 retinal images, representing four classes of pathologies.
Results: The proposed embedded model achieves an accuracy of 0.9806 across the four classes of pathologies collected from the two public databases ODIR (Ocular Disease Intelligence Recognition) and RFMiD (Retinal Fundus Multi-Disease Image Dataset).
Conclusions: Compared to several state-of-the-art methods published to date, the outstanding performance of the proposed embedded system demonstrates its potential as a valuable tool for clinicians to diagnose various ocular diseases in underprivileged healthcare settings.

1. Introduction

The retina is a key component in the visual processing of information to the brain [1]. Conditions that can potentially lead to blindness, such as glaucoma, cataracts, and diabetic retinopathy, can be detected through fundus photography. Routine examination of retinal fundus images enables early detection of these conditions, thereby facilitating timely intervention [2,3].

Currently, the integration of advanced image processing devices and the use of artificial intelligence (AI) are increasingly common in biomedical imaging, especially in the clinical analysis of ophthalmic data [4,5]. The implementation of these new tools aims to automate the treatment process, facilitate clinicians' tasks, and to improve the quality of medical diagnosis while reducing the duration of patient care [6–9]. Despite this remarkable growth, most middle-income regions face difficulties in accessing the material and human resources necessary for their eye care needs in particular [10–12].

In ophthalmology, many authors have developed algorithms for classifying retinal pathologies based on transfer or ensemble learning approaches, using convolutional neural networks (CNN) [13–18]. Despite the state-of-the-art classification performances typically achieved, these models often require enormous computational loads and are not easy to implement in real time. In addition, their integration into treatment devices combining performance, energy efficiency, ergonomics, and accessibility is often not considered or remains unfinished. The development of new applications meeting these practical

* Corresponding author at: Research Unit of Automation and Applied Computer (UR-AIA), Electrical Engineering Department of IUT-FV, University of Dschang, P. O. Box: 134, Bandjoun, Cameroon.
E-mail address: feudjioghislain@gmail.com (F. Ghislain).

https://doi.org/10.1016/j.csbr.2025.100035
Received 27 November 2024; Received in revised form 12 March 2025; Accepted 13 March 2025
Available online 22 March 2025

constraints therefore remains a permanent challenge.

In this study, we implement a classification algorithm for retinal images using a Raspberry Pi 4 Model B board [19]. The primary goal is to develop and deploy an efficient and automated technique for detecting multiple categories of retinal diseases. Our approach utilizes biorthogonal wavelet transforms, which decompose images into different frequency components while preserving essential information. The resulting coefficient matrices provide a rich and compact feature set, which is then integrated into a lightweight CNN optimized for classification, achieving an optimal trade-off between efficiency and performance.

The Deep Learning (DL) model implemented in this work is built around a lightweight CNN architecture [20], adapted to the needs and the computing power available on the Raspberry Pi 4. The hyperparameters were optimized, and regularization techniques were adjusted to improve the performance of the model. The data necessary for training, validation, and testing our model consists of 4303 images labelled into four categories: a class of healthy images (Healthy), a class of images affected by Diabetic Retinopathy (DR), a class of images showing glaucoma (GL), all of which were collected from the public ODIR database [21]. The fourth class (Others), which includes images of rare pathologies such as tessellation, media haze, and optic disc pallor, sourced from the RFMiD database [22]. To optimize the system in terms of size and execution speed, the trained model was converted into a TensorFlow Lite format compatible with the Raspberry Pi 4. The overall performance of our system is evaluated using measurement parameters such as precision, accuracy, F1-score, recall, and confusion matrix.

The following section of this article explores recent advancements in real-time analysis of retinal diseases and their current integration into low-cost, autonomous embedded boards. 3 describes the developed CNN architecture and explains how it was implemented on the Raspberry Pi 4 board. 4 presents the experimental results obtained, followed by a discussion. The last section concludes the manuscript.

2. Previous works

The World Health Organization estimates that over 500 million people worldwide could be affected by diabetes by 2030. Diabetes is a significant risk factor for eye diseases such as cataracts, glaucoma, and diabetic retinopathy [10,12]. Most eye diseases contribute to vision deterioration and blindness. Routine examinations are strongly recommended by clinicians for early diagnosis, the only guarantee of prevention and rapid treatment that can limit blindness. Unfortunately, recent studies indicate that the rate of access to appropriate health care in poor or developing countries is less than 60 %. This rate is even lower in rural areas [11].

Retinal examinations require specific optical devices for image acquisition, such as fundus cameras, ophthalmoscopes, or Optical Coherence Tomography (OCT). The processing of captured retinal images requires computer tools with appropriate computing power. Despite the superior features of fundus cameras, such as high image quality and large storage capacity, their use remains complex and expensive. These cameras are also impractical for widespread screening due to their bulk, particularly in peri-urban and rural areas where access to equipment and specialists is limited. As a result, recent technologies have focused on developing smaller, more accessible, portable, and efficient retinal imaging systems, particularly using smartphones [7,9,23–27].

Once images are acquired, advanced processing techniques must be used to improve image quality and extract relevant features. However, these smartphone-based systems generally do not employ to any automatic technique for analysis, interpretation, and identification of the targeted ocular pathologies. Indeed, an accurate diagnosis relies on processing algorithms using a large number of images. The most widely developed algorithms use DL techniques based on CNNs. The main DL architectures dedicated to classification (such as MobileNet, DenseNet, ResNet, Inception, VGG, GoogLeNet, and AlexNet) require efficient and complex processing tools with considerable computing power [13–18].

Several authors have recently developed efficient and lighter-weight architectures that can be implemented on more accessible and autonomous processing modules. Ghani et al. [8] proposed the use of Field-Programmable Gate Arrays (FPGA) for real-time implementation of classical feed-forward neural network (NN)-based architectures. They obtained satisfactory results in terms of computing power with various preprocessing methods dedicated to the analysis of images that contribute to the diagnosis of glaucoma and diabetic retinopathy.

Much earlier, the performance of FPGAs was successfully demonstrated by Nieto et al. [28], in the extraction of the retinal vascular network, in comparison with other hardware platforms such as the Massively Parallel Single Instruction Multi Data (MP-SIMD) chip and the Massively Parallel Processor Array (MPPA). More recently, the dynamic reliability of the FPGA has been highlighted by Ghanbarpour et al. [29] using the Nyquist-Based Approximation of Retina Rod Cell (NBAoRRC) technique, and by Ghani et al. [30] to map the trained architectures for real-time classification of affected kidney cells. Ruiz-Beltrán et al. [31] configured this integrated module as an on-board material solution for real-time recognition of the iris, with the ability to analyze more than 88 frames per second (FPS).

It turns out that FPGA modules generally have a good adaptability to biomedical imaging applications requiring intensive parallel processing and specific hardware optimization. However, they present a steep learning curve and a potentially higher cost. The Raspberry Pi module is distinguished by its ease of use, its software flexibility, its affordable cost, and its ability to execute a wide variety of applications thanks to its general processor and its easy to handle operating system [32].

Al Jbaar et al. [6] obtained a diagnostic accuracy of 96.35 % by carrying out, using a Raspberry Pi 3 B module, a parallel SIMD implementation of deep CNNs intended for the detection of myopia. Lavanya et al. [33] used the same version of the Raspberry Pi module to implement a DL model to classify DR into five categories. Testing of the model produced a maximum accuracy of 86.6 % with a low overall cost, enabling the diagnosis of DR in multiple primary health centers. Lekshmi et al. [34] used this module to implement a low-cost hybrid automated glaucoma and DR diagnosis algorithm based on an artificial neural network (ANN) classifier. Gibertoni et al. [35] demonstrated that the implementation of classifiers based on machine learning constitutes an easy and appropriate response allowing appreciable performance to be obtained in real time and with reduced computing power on the Raspberry embedded card.

A Raspberry Pi B+ processor was recently used by Rajan et al. [36] to implement an IoT (Internet of Things) system, based on the OCT Deep Net 2 algorithm dedicated to the classification of retinal images, with an accuracy of 98 %. Buzura et al. [37] integrated DL algorithms on the Raspberry Pi 4 to determine the presence or absence of pathologies in retinal images.

It appears that limited access to health care in developing countries, especially in rural areas, hampers the early detection, prevention, and treatment of eye diseases. Additionally, traditional retinal imaging devices are expensive and bulky, making their large-scale deployment difficult. Smartphone-based systems, while more accessible, often lack advanced automated analysis techniques to ensure accurate diagnoses. New technologies based on DL and less expensive hardware platforms like Raspberry Pi modules show promising potential. However, effectively integrating and adapting these technologies to primary health care settings, particularly for multiclass and multilabel classification of ocular pathologies, remains a challenge to be overcome.

3. Methodology and materials

In this section, we present the CNN architecture developed for the classification of various retinal pathologies, as well as its software and hardware implementation on the Raspberry Pi 4 Model B platform.

3.1. Proposed CNN architecture

Typical CNN algorithms for image classification generally consist of two main parts. The first part, called convolution layers, is used to extract essential features, while the second part is dedicated to the actual classification [38]. The first part of the proposed model consists of six convolutional layers, while the second part includes two fully connected layers, followed at the output by a Softmax function. Table 1 shows the tensor sizes and the number of trainable parameters for each layer of the developed architecture.

Each convolutional operation is followed by a Rectified Linear Unit (ReLU) activation function, which introduces the necessary non-linearity for effective modeling of complex relationships between the model's inputs and outputs. The use of the ReLU function as the primary activation function in CNNs offers several advantages: it accelerates learning convergence by limiting potential saturation issues.

A Maxpooling layer with a 2×2 kernel is positioned after each ReLU function to reduce the spatial dimensionality of the data and consolidate essential features [39]. This operation helps to limit the computational load of the model towards the end of the process.

The transition between the convolutional part and the dense part of the architecture is ensured by the flattening layer. This layer applies a flattening technique to transform multidimensional input data into a one-dimensional vector, suitable for the dense layers. The goal of this transition is to facilitate the overall classification process.

Dense layers, also known as fully connected layers, act as classifiers within the CNN architecture. They use the essential features extracted by the convolutional layers to determine the probabilities of a given image belonging to a particular pathology class. Each neuron in a given layer is connected to every neuron in the preceding layer. The model developed in this work includes two dense layers: the first consists of 128 neurons, and the second contains four neurons, corresponding to the number of pathology categories to be classified. This architecture supports the network's learning and optimizes classification based on the extracted features.

The logits from the final dense layer are converted into probabilities using a Softmax function at the system's output, which provides the most probable class for each image. The dropout technique is used in our CNN architecture for regularization. Its choice is justified by its ability to limit overfitting by reducing the interdependencies between different neurons. Another major advantage of dropout is its ability to improve generalization to the entire test dataset, including those data points that were not considered during training.

Table 1
Tensor sizes and number of parameters in the proposed architecture.

Layer name	Tensor sizes	Number of parameters
Conv_1	128×128 x32	1184
Max pooling_1	64×64 x32	0
Conv_2	62×62 x64	18,496
Max pooling_2	31×31 x64	0
Conv_3	29×29 x64	36,928
Max pooling_3	14×14 x64	0
Conv_4	12×12 x64	36,928
Max pooling_4	6×6 x64	0
Conv_5	4×4 x64	36,928
Max pooling_5	2×2 x64	0
Conv_6	2×2 x64	36,928
Max pooling_6	1×1 x64	0
Flatten	64	0
Dropout	64	0
Dense_1	128	8320
Dense_2	4	516
Total learnable parameters		176,228 (688.39 KB)

3.2. Feature extraction based on discrete wavelet decomposition (DWT)

The source retinal images are converted to grayscale and then decomposed using the biorthogonal wavelet transform to extract essential features. The general form of a 2D biorthogonal wavelet can be expressed as follows [40]:

$$\omega_{\alpha_1\alpha_2\beta_1\beta_2}(x,y) = \frac{1}{\sqrt{\alpha_1\alpha_2}}\omega\left(\frac{x-\beta_1}{\alpha_1}, \frac{y-\beta_2}{\alpha_2}\right) \quad (1)$$

Here, α_1 and α_2 represent the scale parameters, and β_1 and β_2 denote the position parameters. The DWT of an image $I(x,y)$ is formulated as [40]:

$$W_s(k,l) = \frac{1}{s}\sum_n\sum_m I(m,n)\omega\left(m-\frac{k}{s}, n-\frac{l}{s}\right) \quad (2)$$

In this equation, (k,l) represents the position of the wavelet, and s is the scale. This framework allows for analysing images similarly to using a magnifying glass, adjusting position and scale to capture different details.

The result of the decomposition is a set of four coefficient matrices: LL (summary image), HL (horizontal details), LH (vertical details), and HH (diagonal details). In the first step, each row of the image is averaged and wavelet-transformed [41]. This produces two reduced images: the summary image $\phi(x,y)$ and the horizontal details $\omega_H(x,y)$, which correspond to:

$$\phi(x,y) = \phi(x)\phi(y) \quad \rightarrow LL \quad (3)$$

$$\omega_H(x,y) = \phi(x)\omega(y) \quad \rightarrow HL \quad (4)$$

Next, the scaling and wavelet transformations are applied to the columns of the two obtained images. This generates quarter-sized images, including the vertical details $\omega_V(x,y)$ and the diagonal details $\omega_D(x,y)$, given by:

$$\omega_V(x,y) = \omega(x)\phi(y) \quad \rightarrow LH \quad (5)$$

$$\omega_D(x,y) = \omega(x)\omega(y) \quad \rightarrow HH \quad (6)$$

In this study, the implementation of feature extraction through discrete decomposition is based on the bior1.3 wavelet. The choice of this biorthogonal version is justified by its symmetry and vanishing moments, which allow for the preservation of important contours and details while minimizing artifacts. The coefficients of the scaling function and the wavelet function of bior1.3 ensure an effective representation of the textures and structures present in retinal images, which is crucial for accurate classification [42].

3.3. Data collection

In this work, four categories of retinal fundus images, labeled as Healthy, Glaucoma (GL), Diabetic Retinopathy (DR), and "Others," were considered. The Healthy, GL, and DR classes, each consisting of approximately 1070 images, were randomly selected from the various groups of ocular diseases in the ODIR dataset [21]. The ODIR dataset includes images from several devices, including Canon, Zeiss, and Kowa. The category labeled "Others" comprises around 1100 retinal fundus images representing various rare or less frequently studied ophthalmological pathologies, including Tessellation, Drusen, Media Haze, Optic Disc Pallor, and Branch Retinal Vein Occlusion. The data for this class were randomly collected from the RFMiD [22], which mainly consists of images from Topcon and Kowa devices. This dataset is known for its high diversity of retinal diseases. Despite the relatively small number of images per class, RFMiD is a suitable choice for developing generalizable tools for comprehensive diagnostic use in clinical settings.

The resulting dataset is divided into a training subset of 2560 images, a validation subset of 452 images used for hyperparameter tuning and

model selection, and a testing subset of 1291 images used for final evaluation of model performance, with the general distribution detailed in Table 2.

3.4. Raspberry Pi 4 model B platform

The Raspberry Pi module is an electronic circuit about the size of a credit card, integrating all the components of a computer. Originally designed for educational purposes, the Raspberry Pi platform achieved great success immediately after its market introduction in 2012.

The Raspberry Pi 4 Model B (RPi4B) is one of the most powerful single-board computers designed by the Raspberry Pi Foundation. The model used in this study is equipped with the Broadcom BCM2711 SoC, which includes a 1.8 GHz quad-core Cortex-A72 64-bit processor (CPU) and 4 GB of RAM. It also features a dual-band WiFi interface operating at 2.4 GHz and 5 GHz in compliance with IEEE 802.11b/g/n/ac standards, a 1 Gb Ethernet interface, 2 micro-HDMI connectors, 4 USB ports (2 USB 2.0 and 2 USB 3.0), and a microSD card slot. This compact, versatile, and powerful platform is suitable for various applications, including mobile projects. As shown in Fig. 1, the RPi4B features multiple interfaces, including RJ45, GPIO, an audio jack (3.5 mm), and various display devices [43,44].

3.5. Display and control peripheral for the RPi4B module

The RPi4B module features an HDMI (High-Definition Multimedia Interface) connector, which allows it to be connected to any display device that supports HDMI. By connecting a keyboard and a mouse through its multiple USB ports, a complete desktop computer setup can be easily achieved. Additionally, it is possible to use displays such as LCD (Liquid Crystal Display) or TFT (Thin-Film Transistor) screens by connecting them to the SPI (Serial Peripheral Interface) bus available on the 26-pin GPIO (General-Purpose Input/Output) port. Furthermore, the RPi4B also features an Ethernet port, enabling a stable wired connection that ensures reliable operation without depending on Wi-Fi, even on low-performance PCs.

To enhance the portability of the system, an alternative is to use an Android device (tablet or smartphone) as a control and display tool for the RPi4B module. For the prototype developed in this work, a smartphone (with 4 GB of RAM, a 6-inch screen, and Android version 14) is used, as illustrated in the diagram of Fig. 2. The smartphone is equipped with a VNC (Virtual Network Computing) application downloaded from the Play Store [45] and configured to remotely access the RPi4B desktop via Wi-Fi. The VNC settings were enabled on the RPi4B. Moreover, the smartphone is configured as a Wi-Fi access point, allowing direct connection to the Raspberry Pi without the need for an external Wi-Fi network or Internet connection. This setup ensures that the system remains functional even in areas with limited network coverage.

4. Results and discussions

Python frameworks such as TensorFlow were employed to train our model for 80 epochs with a batch size of 20. Once the model was trained, it was converted to TensorFlow Lite. This conversion is crucial because TensorFlow Lite is specifically designed to optimize models for efficient operation on resource-limited devices, such as the Raspberry Pi 4B. The use of TensorFlow Lite helps shrink the model size and boost its execution speed, which is essential for performance on the Raspberry Pi platform, given its limitations in computational power and memory. In addition, the graphical interface was developed to facilitate the use of the prototype from the Android terminal, as shown in Fig. 3. This approach ensures that our model operates optimally on the target platform while maintaining high performance.

Table 2
Distribution of retinal image classes across the training, validation, and testing subsets.

No	Class	Training set	Validation set	Test set	TOTAL
1	Healthy	637	113	324	1074
2	GL	607	109	321	1037
3	DR	660	115	323	1098
4	Others	656	115	323	1094
TOTAL		2560	452	1291	4303

4.1. The performance measures

The overall classification performance of the proposed model was evaluated based on various statistical parameters, including accuracy, precision, recall, and the F1 score.

Precision, or Positive Predictive Value (PPV), determines the ratio of true positive predictions to all positive predictions made by the model. Recall, or sensitivity, evaluates the number of true positive predictions relative to all actual positives for a given class. The F1 score represents the harmonic mean of recall and precision, providing a single metric that balances these two measures. Accuracy (Acc) determines the ratio of correctly classified items to the total number of targeted samples. Accuracy is therefore an important measure of the model's overall ability to predict the correct class for all samples. Table 3 provides the mathematical expressions for these various performance parameters [46–48].

In Table 3, TP (True Positives) refers to the set of positive samples correctly identified as positive, while TN (True Negatives) represents negative samples correctly classified as negative. FP (False Positives) denotes the number of negative samples incorrectly classified as positive, and FN (False Negatives) refers to positive samples incorrectly classified as negative by the model.

We also introduced a graphical parameter to evaluate the prototype's performance using the ROC (Receiver Operating Characteristic) curve. The area under the ROC curve (AUC) was then calculated to assess the system's ability to differentiate between the various classes [3,49,50].

4.2. Result analysis

The confusion matrix presented in Table 4 shows the performance of the classification model for the four classes: Healthy, GL, DR, and Others. The results indicate that the model correctly predicted 316 examples of the Healthy class, 305 for the GL class, 322 for the DR class, and 323 for the Others class, demonstrating excellent overall performance with few errors.

However, a few minor errors are observable, primarily between the Healthy and GL classes. Specifically, 8 examples from the Healthy class were misclassified as GL, and 15 examples from the GL class were misclassified as Healthy. Additionally, 1 example from the GL class was misclassified as DR. The Others class recorded no confusion, thus highlighting the model's effectiveness in classifying the different categories.

The performance report presented in Table 5 highlights the model's performance, achieving an overall accuracy of 0.98064, indicating that it correctly classified nearly all instances in the test data. The individual average metrics also show strong results, with precision, recall, and the F1 score at 0.98073, 0.98064, and 0.98062, respectively, suggesting that the model is reliable in its positive predictions and effective in identifying true instances of each class.

In particular, the DR class exhibits perfect performance with both precision and recall at 1.0. Similarly, the "Others" class also shows perfect scores, demonstrating the model's effectiveness in classifying these categories. However, the Healthy and GL classes show slight disparities: Healthy has a precision of 0.95 and a recall of 0.98, while GL has a precision of 0.97 and a recall of 0.95. This indicates that some instances from these classes were not identified, but these differences are minimal and do not significantly affect the model's overall performance.

The prediction results shown in Fig. 4 for the specific images

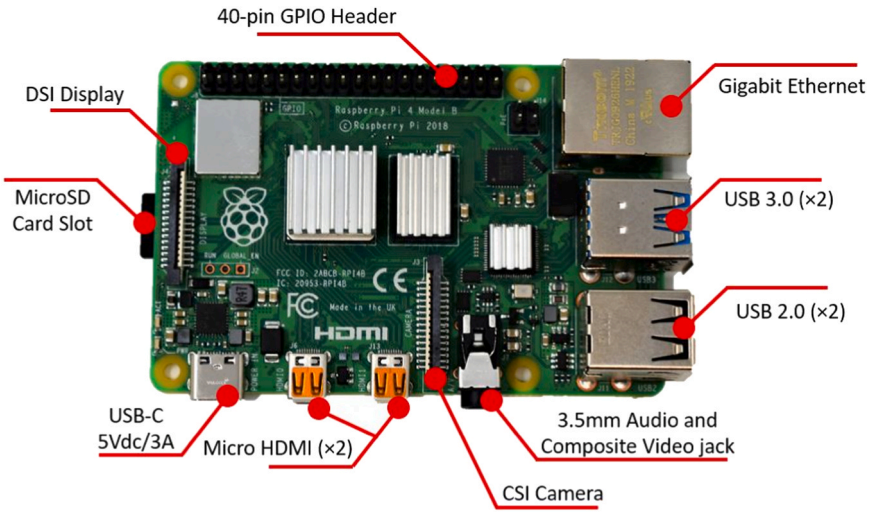

Fig. 1. Raspberry Pi 4 Model B.

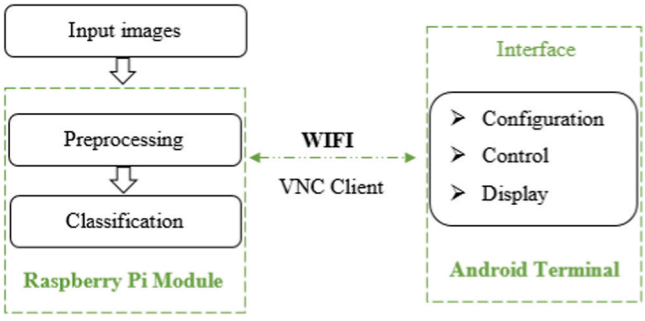

Fig. 2. Control and display of the RPi4B module via smartphone.

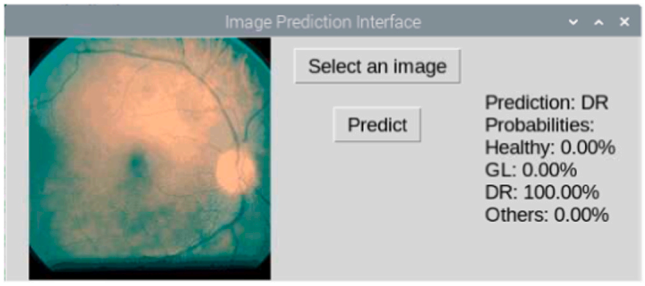

Fig. 3. Image prediction interface.

Table 3
Mathematical expressions of classification performance metrics.

Measures	Equation
Precision	TP/(TP + FP)
Recall	TP/(TP+FN)
F1-Score	2 x PPV x Recall/(PPV + Recall)
Acc	**(TP + TN)/(TP+ FP + TN + FN)**

randomly selected from different classes demonstrate a high level of confidence from the model in its classifications. For class 0, the model assigns a 99.97 % certainty that the image belongs to the "Healthy" category, indicating a strong confidence in this classification. In contrast, the probabilities for the other classes are very low, with 0.03 % for the GL class and 0.00 % for both the DR and "Others" classes.

For class 1, the model shows absolute confidence with a 100 % certainty that the image belongs to the GL class, while the probabilities for the other classes remain 0.00 %. In class 2, the model again displays complete certainty at 100 % for the DR class, indicating that it recognizes the image as belonging to this category, with all other class probabilities at 0.00 %. Finally, in class 3, the model is again 100 % certain that the image belongs to the "Others" class, with no probability assigned to the other classes. Overall, these results demonstrate that the model is extremely confident in its classifications, with high certainty for specific classes and negligible probabilities for others.

Table 4
Confusion matrix of proposed CNN.

Predicted class	Targeted class			
	Healthy	GL	DR	Others
Healthy	316	8	0	0
GL	15	305	1	0
DR	0	1	322	0
Others	0	0	0	323

Table 5
Classification performance report of the proposed CNN.

No	Class	Accuracy	Precision	Recall	F1 Score	AUC
0	Healthy	0.95	0.95	0.98	0.96	1.0
1	GL	0.97	0.97	0.95	0.96	1.0
2	DR	1.0	1.0	1.0	1.0	1.0
3	Others	1.0	1.0	1.0	1.0	1.0
Average		0.98064	0.98073	0.98064	0.98062	1.0

The AUC values presented in Fig. 5 offer valuable insights into the model's performance for classifying each class. An AUC of 1.00 indicates perfect performance, meaning the model can identify instances of each class without any error. This is particularly encouraging, as the model demonstrates a remarkable ability to effectively distinguish the associated images.

These AUC values reflect a high level of robustness and reliability in the model's predictions, suggesting that the model will perform consistently in real-world applications. These results are particularly promising for clinical applications, where accurate classification is crucial.

4.3. Comparative analysis of performance with existing methods

In Tables 6 and 7, we present a comparative analysis of the

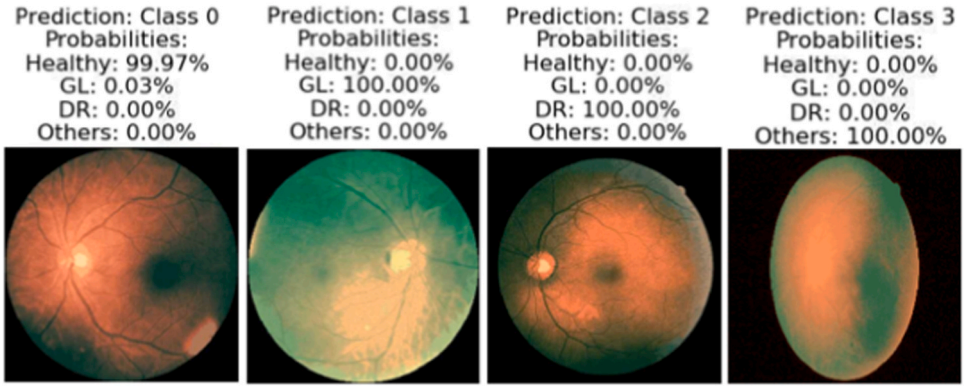

Fig. 4. Predictions for the specific images in different classes.

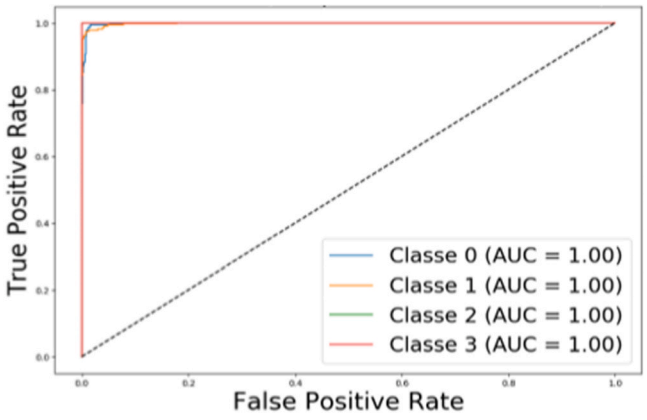

Fig. 5. Roc curves.

performance of the proposed ocular disease detection model against existing methods and other deep learning models suitable for embedded systems. This analysis is based on statistical measures such as recall, precision, F1 score, accuracy (Acc), and AUC, as well as computational time and the architecture used for implementation. It references the classification models developed by Ouda et al. [3], Al Jbaar et al. [6], Thomas et al. [7], Rajan et al. [36], Yadav et al. [38], Scarpa et al. [48], Gour et al. [49], Wang et al. [50], Zamir et al. [51], Kristiani et al. [52], James et al. [53], Jeny et al. [54], Bhati et al. [55], Al-Naji et al. [56], and Jayanthy et al. [57]. All highlighted results are directly sourced from the original literature. The symbol "/" indicates values not provided by the authors.

In Table 6, the proposed model, with an average accuracy (Acc) of 0.9806, outperforms several top approaches listed. However, the model by Yadav et al. [38] achieves a slightly higher accuracy of 0.9944. It is important to note that their model uses a pre-trained architecture, which, while effective, is not optimal for real-time applications due to the significant processing power required. In comparison, our model, with an average predictive positive value (PPV) of 0.9807, and both recall and F1 score at 0.9806, demonstrates robustness in detecting ocular diseases while being designed for efficient operation on embedded systems. Furthermore, the highest AUC value further underscores the model's ability to perfectly distinguish the targeted classes.

Table 7 shows that the developed deep learning model achieves the previously mentioned performance, with an execution time of only 0.53 seconds on a Raspberry Pi 4 B. This relatively low processing time places it among the best existing models, such as those by Al Jbaar et al.

Table 6
Performance comparison with existing ocular disease detection methods.

Authors	Implementation platform	Trainable parameters	Acc	PPV	Recall	F1 Score	AUC
Ouda el al. [3]	NVIDIA VGA	29.6 M	0.943	0.915	0.80	/	0.967
Al Jbaar et al. [6]	Cortex-A53	3 M	0.9635	0.96	0.96	0.96	/
Thomas et al. [7]	/	/	0.9583	0.9750	0.93	/	/
Yadav et al. [38]	Nvidia Quadro RTX 4000	/	**0.9944**	/	**0.9926**	/	/
Scarpa et al. [48]	/	11 M	0.964	/	/	/	/
Gour et al. [49]	NVIDIA RTX5000	15.2 M	/	/	/	0.8557	0.8493
Wang et al. [50]	NVIDIA RTX2080Ti	/	0.89	0.63	0.58	/	0.73
Jeny et al. [54]	Nvidia GeForce RTX 2070	1.749 M	0.9696	0.9277	0.8738	/	0.9590
Bhati et al. [55]	Nvidia GeForce RTX 2070	27 M	0.9381	/	/	0.9428	0.9608
Proposed Model	Cortex-A72	**176228**	0.9806	0.9807	0.9806	**0.9806**	**1.0**

Table 7
Performance comparison with existing deep learning methods on embedded systems.

Authors	Implementation platform	Deep learning models	Developed application	Acc	Time(s)
Al Jbaar et al. [6]	Raspberry Pi 3 B	Deep CNN	Myopia Ocular disease detection	0.9635	3.25
Rajan et al. [36]	Raspberry Pi B+	OCT Deep Net2	Retinal images classification	0.98	/
Zamir et al. [51]	Raspberry Pi 3	ResNet	Face Detection & Recognition from Images	**0.9824**	30
Kristiani et al. [52]	Raspberry Pi 4	VGG16	Image classification	0.91	50
James et al. [53]	Raspberry Pi 4 B	MobileNetV3	Distributed Deep Learning Inference Using Raspberry Pi	/	20.67
Al-Naji et al. [56]	Raspberry Pi 4 B	InceptionResNetV2	Computer vision for eye diseases detection	0.93	/
Jayanthy et al. [57]	Raspberry Pi 4 B	MobileNetV2	Otoscopy Image Classification	0.97	/
Proposed Model	Raspberry Pi 4 B	Deep CNN	Multi-Label Retinal Diseases Classification	0.9806	**0.53**

[6] (3.25 s), Zamir et al. [51] (30 s), and James et al. [53] (20.67 s). This improvement in inference time is crucial for real-time applications, such as those in ophthalmology. In summary, the developed prototype stands out for its efficiency and speed, making it particularly well-suited for practical use on embedded systems.

4.4. Study limitations

A notable limitation of this study is the lack of an in-depth analysis of the impact of the different devices used to capture retinal fundus images in the two datasets. Indeed, the images employed were captured using distinct traditional ophthalmic imaging devices, such as the Kowa VX-10, the TOPCON 3D OCT-2000, and the TOPCON TRC-NW300 for the RFMiD dataset [22], and the Canon, Zeiss, and Kowa for the ODIR dataset [21]. We did not study in detail how the differences between these devices, such as varying resolutions and other technical features, might influence the overall performance of the model. This variation in image quality could affect the model's ability to generalize its predictions, particularly in contexts where the types of devices used present very diverse characteristics.

Furthermore, the exclusive use of datasets with images captured solely by traditional devices represents another limitation of this study. While these traditional devices provide high-quality images, their high cost and complexity of use can limit their accessibility, especially in resource-constrained environments, which are the focus of this work. In this context, smartphone-captured fundus images could offer an interesting alternative to make ophthalmic imaging more affordable and accessible. Further research is needed not only to optimize the development of image datasets captured with these devices, but also to assess how the quality of these images might impact diagnostic accuracy, and to identify best practices to minimize potential artifacts.

5. Conclusion

A real-time diagnostic system using a Raspberry Pi 4 B module has been implemented to automatically identify four distinct classes of pathologies. The classification algorithm developed is based on a lightweight CNN architecture, enabling fast, efficient execution with minimal computing power. Testing of the system yielded impressive performance scores, with an average accuracy, recall, and F1 score of 0.9806, and an average precision of 0.9807. The overall performance surpasses that of several state-of-the-art methods reported in the literature. The advantages of the proposed CNN prototype go beyond its simple implementation; it also demonstrates exceptional precision in real-time identification of a wide range of eye conditions. Its successful integration into a portable, accessible, and lightweight device makes it an ideal support tool for ophthalmology services in health centers located in peri-urban or rural areas with limited resources. This is particularly significant in real clinical settings, where this low-cost embedded processor can perform classifications autonomously, using a portable Android phone for control and display, without relying on an external PC. Furthermore, configuring a portable camera to capture retinal fundus images, and appropriately adapting it to the proposed system, could further enhance its autonomy and automation in future developments.

Ethics approval

This article does not contain any studies with human participants and/or animals performed by any of the authors.

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Consent to participate

Not applicable.

Consent to publish

Not applicable.

CRediT authorship contribution statement

Daniel Tchiotsop: Formal analysis, Funding acquisition, Project administration, Supervision, Validation, Visualization, Writing – review & editing. **Atangana Romain:** Formal analysis, Validation, Visualization, Writing – review & editing. **Beaudelaire Saha Tchinda:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Ghislain Feudjio:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors sincerely thank the editor and the reviewers for their constructive comments to improve the quality of the manuscript.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.csbr.2025.100035.

Data availability

Data will be made available on request.

References

[1] Navarro R. The optical design of the human eye: a critical review. J Optom 2009;2(1):3–18. https://doi.org/10.3921/joptom.2009.3.
[2] Cen LP, Ji J, Lin JW, Ju ST, Lin HJ, Li TP, et al. Automatic detection of 39 fundus diseases and conditions in retinal photographs using deep neural networks. Nat Commun 2021;12(1):4828. https://doi.org/10.1038/s41467-021-25138-w.
[3] Ouda O, AbdelMaksoud E, Abd El-Aziz AA, Elmogy M. Multiple ocular disease diagnosis using fundus images based on multi-label deep learning classification. Electronics 2022;11(13):1966. https://doi.org/10.3390/electronics11131966.
[4] Senapati A, Tripathy HK, Sharma V, Gandomi AH. Artificial intelligence for diabetic retinopathy detection: a systematic review. Inform Med Unlocked 2024:101445. https://doi.org/10.1016/j.imu.2024.101445.
[5] Ghislain F, Beaudelaire ST, Daniel T. An accurate unsupervised extraction of retinal vasculature using curvelet transform and classical morphological operators. Comput Biol Med 2024;178:108801. https://doi.org/10.1016/j.compbiomed.2024.108801.
[6] Al Jbaar MA, Dawwd SA. SIMD implementation of deep CNNs for myopia detection on a single-board computer system. East-Eur J Enterp Technol 2023;125(9). https://doi.org/10.15587/1729-4061.2023.289007.
[7] Thomas SA, Titus G. Design of a portable retinal imaging module with automatic abnormality detection. Biomed Signal Process Control 2020;60:101962. https://doi.org/10.1016/j.bspc.2020.101962.
[8] Ghani A, See CH, Sudhakaran V, Ahmad J, Abd-Alhameed R. Accelerating retinal fundus image classification using artificial neural networks (ANNs) and reconfigurable hardware (FPGA). Electronics 2019;8(12):1522. https://doi.org/10.3390/electronics8121522.
[9] Bourouis A, Feham M, Hossain MA, Zhang L. An intelligent mobile based decision support system for retinal disease diagnosis. Decis Support Syst 2014;59:341–50. https://doi.org/10.1016/j.dss.2014.01.005.
[10] Bechange S, Schmidt E, Ruddock A, Khan IK, Gillani M, Roca A, et al. Understanding the role of lady health workers in improving access to eye health

services in rural Pakistan–findings from a qualitative study. Arch Public Health 2021;79:1–2. https://doi.org/10.1186/s13690-021-00541-3.
[11] Bechange S, Jolley E, Virendrakumar B, Pente V, Milgate J, Schmidt E. Strengths and weaknesses of eye care services in sub-Saharan Africa: a meta-synthesis of eye health system assessments. BMC Health Serv Res 2020;20:1–8. https://doi.org/10.1186/s12913-020-05279-2.
[12] Palmer JJ, Chinanayi F, Gilbert A, Pillay D, Fox S, Jaggernath J, et al. Mapping human resources for eye health in 21 countries of sub-Saharan Africa: current progress towards VISION 2020. Hum Resour Health 2014;1:1–6. https://doi.org/10.1186/1478-4491-12-44.
[13] Mall PK, Singh PK, Srivastav S, Narayan V, Paprzycki M, Jaworska T, et al. A comprehensive review of deep neural networks for medical image processing: Recent developments and future opportunities. Healthc Anal 2023:100216. https://doi.org/10.1016/j.health.2023.100216.
[14] Ilesanmi AE, Ilesanmi T, Gbotoso AG. A systematic review of retinal fundus image segmentation and classification methods using convolutional neural networks. Healthc Anal 2023:100261. https://doi.org/10.1016/j.health.2023.100261.
[15] Zakaria N, Hassim YM. A review study of the visual geometry group approaches for image classification. J Appl Sci, Technol Comput 2024;1(1):14–28. https://doi.org/10.30880/jastec.2024.01.01.003.
[16] Badar M, Haris M, Fatima A. Application of deep learning for retinal image analysis: A review. Comput Sci Rev 2020;35:100203. https://doi.org/10.1016/j.cosrev.2019.100203.
[17] Kim HE, Cosa-Linan A, Santhanam N, Jannesari M, Maros ME, Ganslandt T. Transfer learning for medical image classification: a literature review. BMC Med Imaging 2022;22(1):69. https://doi.org/10.1186/s12880-022-00793-7.
[18] Muchuchuti S, Viriri S. Retinal disease detection using deep learning techniques: a comprehensive review. J Imaging 2023;9(4):84. https://doi.org/10.3390/jimaging9040084.
[19] Karthikeyan S, Raj RA, Cruz MV, Chen L, Vishal JA, Rohith VS. A systematic analysis on raspberry pi prototyping: uses, challenges, benefits, and drawbacks. IEEE Internet Things J 2023;10(16):14397–417. https://doi.org/10.1109/JIOT.2023.3262942.
[20] Gayathri S, Gopi VP, Palanisamy P. A lightweight CNN for diabetic retinopathy classification from fundus images. Biomed Signal Process Control 2020;62:102115. https://doi.org/10.1016/j.bspc.2020.102115.
[21] Peking University. Peking University International Competition on Ocular Disease Intelligent Recognition (ODIR-2019). [Online]. Available from: ⟨https://odir2019.grand-challenge.org/dataset/⟩.
[22] Pachade S, Porwal P, Thulkar D, Kokare M, Deshmukh G, Sahasrabuddhe V, et al. Retinal fundus multi-disease image dataset (RFMiD): a dataset for multi-disease detection research. Data 2021;6(2):14. https://doi.org/10.3390/data6020014.
[23] Karakaya M, Hacisoftaoglu RE. Comparison of smartphone-based retinal imaging systems for diabetic retinopathy detection using deep learning. BMC Bioinforma 2020;21(4):259. https://doi.org/10.1186/s12859-020-03587-2.
[24] Neto A, Camara J, Cunha A. Evaluations of deep learning approaches for glaucoma screening using retinal images from mobile device. Sensors 2022;22(4):1449. https://doi.org/10.3390/s22041449.
[25] Soliz P, Zamora G, Benson J, Nemeth S, Maynard J, Barriga S, et al. Impact of retinal image quality: software aid for a low-cost device and effects on disease detection. Ophthalmic Technologies XXIX, 10858. SPIE; 2019. p. 137–48. https://doi.org/10.1117/12.2510125.
[26] Martinez-Perez ME, Hughes AD, Thom SA, Parker KH, Witt NW. Evaluation of a portable retinal imaging device: towards a comparative quantitative analysis for morphological measurements of retinal blood vessels. R Soc Open Sci 2023;10(6):230065. https://doi.org/10.1098/rsos.230065.
[27] Penha FM, Priotto BM, Hennig F, Przysiezny B, Wiethorn BA, Orsi J, et al. Single retinal image for diabetic retinopathy screening: performance of a handheld device with embedded artificial intelligence. Int J Retin Vitr 2023;9(1):41. https://doi.org/10.1186/s40942-023-00477-6.
[28] Nieto A, Brea V, Vilariño DL, Osorio RR. Performance analysis of massively parallel embedded hardware architectures for retinal image processing. EURASIP J Image Video Process 2011;2011:1–7. https://doi.org/10.1186/1687-5281-2011-10.
[29] Ghanbarpour M, Haghiri S, Hazzazi F, Assaad M, Chaudhary MA, Ahmadi A. Investigation on vision system: Digital FPGA implementation in case of retina rod cells. IEEE Trans Biomed Circuits Syst 2023. https://doi.org/10.1109/TBCAS.2023.3323324.
[30] Ghani A, Hodeify R, See CH, Keates S, Lee DJ, Bouridane A. Computer vision-based Kidney's (HK-2) damaged cells classification with reconfigurable hardware accelerator (FPGA). Electronics 2022;11(24):4234. https://doi.org/10.3390/electronics11244234.
[31] Ruiz-Beltrán CA, Romero-Garcés A, González M, Pedraza AS, Rodríguez-Fernández JA, Bandera A. Real-time embedded eye detection system. Expert Syst Appl 2022;194:116505. https://doi.org/10.1016/j.eswa.2022.116505.
[32] Bhowmik D, Appiah K. Embedded vision systems: A review of the literature (Lecture Notes in Computer Science). Applied Reconfigurable Computing. Architectures, Tools, and Applications. ARC, 10824. Cham: Springer; 2018. https://doi.org/10.1007/978-3-319-78890-6_17.
[33] Lavanya RV, Sumesh EP, Jayakumari C, Isaac R. Detection and classification of diabetic retinopathy using raspberry PI. 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA). IEEE; 2020. p. 1688–91. https://doi.org/10.1109/ICECA49313.2020.9297408.
[34] Lekshmi SA, Rajathi GM. Detection of glaucoma and diabetic retinopathy using image processing technique by Raspberry pi. Indian J Sci Technol 2019;12(29):1–6. https://doi.org/10.17485/ijst/2019/v12i29/146968.
[35] Gibertoni G, Borghi G, Rovati L. Vision-based eye image classification for ophthalmic measurement systems. Sensors 2022;23(1):386. https://doi.org/10.3390/s23010386.
[36] Rajan R, Kumar SN. IoT based optical coherence tomography retinal images classification using OCT Deep Net2. Meas: Sens 2023;25:100652. https://doi.org/10.1016/j.measen.2022.100652.
[37] Buzura L, Groza G, Papara R, Galatus R. Assisted OCT diagnosis embedded on Raspberry Pi 4. 2021 IEEE 27th International Symposium for Design and Technology in Electronic Packaging (SIITME). IEEE; 2021. p. 281–6. https://doi.org/10.1109/SIITME53254.2021.9663686.
[38] Yadav S, Murugan R, Goel T. Opti-EN-net: optimized ensemble deep neural network for the classification of retinal detachment through fundus images. Biomed Signal Process Control 2024;91:105999. https://doi.org/10.1016/j.bspc.2024.105999.
[39] Raiaan MA, Fahad NM, Mukta MS, Shatabda S. Mammo-light: a lightweight convolutional neural network for diagnosing breast cancer from mammography images. Biomed Signal Process Control 2024;94:106279. https://doi.org/10.1016/j.bspc.2024.106279.
[40] Hess-Nielsen N, Wickerhauser MV. Wavelets and time-frequency analysis. Proc IEEE 1996;84(4):523–40. https://doi.org/10.1109/5.488698.
[41] Zhang D, Zhang D. Wavelet Transform. Fundamentals of Image Data Mining. Texts in Computer Science. Cham: Springer; 2019. https://doi.org/10.1007/978-3-030-17989-2_3.
[42] Mazumder B, Khan MS, Uddin KM. Biorthogonal wavelet based entropy feature extraction for identification of maize leaf diseases. J Agric Food Res 2023;14:100756. https://doi.org/10.1016/j.jafr.2023.100756.
[43] Gamess E, Hernandez S. Performance evaluation of different Raspberry Pi models for a broad spectrum of interests. Int J Adv Comput Sci Appl 2022;13(2).
[44] Raspberry Pi Foundation. Raspberry Pi 4 Model B [Online]. Available from: ⟨https://www.raspberrypi.com/products/raspberry-pi-4-model-b⟩. Accessed September 4, 2024.
[45] VNC Viewer. Google Play Store. Available from: ⟨https://play.google.com/store/apps/details?id=com.realvnc.viewer.android⟩. Accessed September 4, 2024.
[46] Tchinda BS, Tchiotsop D, Noubom M, Louis-Dorr V, Wolf D. Retinal blood vessels segmentation using classical edge detection filters and the neural network. Inform Med Unlocked 2021;23:100521. https://doi.org/10.1016/j.imu.2021.100521.
[47] Dai H, Yang Y, Yue X, Chen S. Improving retinal OCT image classification accuracy using medical pre-training and sample replication methods. Biomed Signal Process Control 2024;91:106019. https://doi.org/10.1016/j.bspc.2024.106019.
[48] Scarpa F, Berto A, Tsiknakis N, Manikis G, Fotiadis DI, Marias K, et al. Automated analysis for glaucoma screening of retinal videos acquired with smartphone-based ophthalmoscope. Heliyon 2024;10(14). https://doi.org/10.1016/j.heliyon.2024.e34308.
[49] Gour N, Khanna P. Multi-class multi-label ophthalmological disease detection using transfer learning based convolutional neural network. Biomed Signal Process Control 2021;66:102329. https://doi.org/10.1016/j.bspc.2020.102329.
[50] Wang J, Yang L, Huo Z, He W, Luo J. Multi-label classification of fundus images with efficientnet. IEEE Access 2020;8:212499–508.
[51] Zamir M, Ali N, Naseem A, Ahmed Frasteen A, Zafar B, Assam M, et al. Face detection & recognition from images & videos based on CNN & Raspberry Pi. Computation 2022;10(9):148. https://doi.org/10.3390/computation10090148.
[52] Kristiani E, Yang CT, Huang CY. iSEC: An optimized deep learning model for image classification on edge computing. IEEE Access 2020;8:27267–76.
[53] James N, Ong LY, Leow MC. Exploring distributed deep learning inference using raspberry pi spark cluster. Future Internet 2022;14(8):220. https://doi.org/10.3390/fi14080220.
[54] Jeny AA, Junayed MS, Islam MB. Deep neural network-based ensemble model for eye diseases detection and classification. Image Anal Stereol 2023 Jul 10;42(2):77–91. https://doi.org/10.5566/ias.2857.
[55] Bhati A, Gour N, Khanna P, Ojha A. Discriminative kernel convolution network for multi-label ophthalmic disease detection on imbalanced fundus image dataset. Comput Biol Med 2023;153:106519. https://doi.org/10.1016/j.compbiomed.2022.106519.
[56] Al-Naji A, Khalid GA, Mahmood MF, Chahl J. Computer vision for eye diseases detection using pre-trained deep learning techniques and raspberry Pi. J Eng 2024;2024(7):e12410. https://doi.org/10.1049/tje2.12410.
[57] Jayanthy S, Abinaya G, Abinayasri K. Otoscopy Image Classification Using Embedded AI. Jun 28. 2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS). IEEE; 2024. p. 1–6. https://doi.org/10.1109/ICITEICS61368.2024.10625385. Jun 28.