



Trish: an efficient activation function for CNN models and analysis of its effectiveness with optimizers in diagnosing glaucoma

Cemil Közkurt² · Aykut Diker¹ · Abdullah Elen¹ · Serhat Kılıçarslan¹ · Emrah Dönmez¹ · Fahrettin Burak Demir¹

Accepted: 8 March 2024 / Published online: 3 April 2024
© The Author(s) 2024

Abstract

Glaucoma is an eye disease that spreads over time without showing any symptoms at an early age and can result in vision loss in advanced ages. The most critical issue in this disease is to detect the symptoms of the disease at an early age. Various researches are carried out on machine learning approaches that will provide support to the expert for this diagnosis. The activation function plays a pivotal role in deep learning models, as it introduces nonlinearity, enabling neural networks to learn complex patterns and relationships within data, thus facilitating accurate predictions and effective feature representations. In this study, it is focused on developing an activation function that can be used in CNN architectures using glaucoma disease datasets. The developed function (Trish) was compared with ReLU, LReLU, Mish, Swish, Smish, and Logish activation functions using SGD, Adam, RmsProp, AdaDelta, AdaGrad, Adamax, and Nadam optimizers in CNN architectures. Datasets consisting of retinal fundus images named ACRIMA and HRF were used within the scope of the experiments. These datasets are widely known and currently used in the literature. To strengthen the test validity, the proposed function was also tested on the CIFAR-10 dataset. As a result of the study, 97.22% validation accuracy performance was obtained. It should be stated that the acquired performance value is at a significant level for the detection of glaucoma.

Keywords Activation function · Artificial learning · Convolutional neural network (CNN) · Glaucoma diagnosis · Optimizer

✉ Serhat Kılıçarslan
skilicarslan@bandirma.edu.tr

¹ Department of Software Engineering, Faculty of Engineering and Natural Sciences, Bandırma Onyedi Eylül University, 10200 Bandırma, Balıkesir, Turkey

² Department of Transportation Engineering, Faculty of Engineering and Natural Sciences, Bandırma Onyedi Eylül University, 10200 Bandırma, Balıkesir, Turkey

1 Introduction

The eye is a vital organ for vision, which is one of our most basic senses to perceive the environment in biological forms. The deterioration in the health status of this organ can lead to the emergence of negative situations, from the decrease in the ability to see to its complete loss. One of the riskiest diseases for the eye is glaucoma. Glaucoma basically represents a group of eye diseases, not a single disease. As a result of glaucoma, eye damage such as loss of vision and complete loss of vision may occur. The main cause of glaucoma is damage to the optic nerves located behind the eye over time. The fact that the symptoms are not very obvious in the diagnosis of glaucoma and the disease progresses very slowly over time can cause people not to be aware of these disorders. In the absence of treatment, it is possible that the vision can be completely lost. Although it is stated that glaucoma-related disorders occur more in old age, it is also possible to encounter this disorder at an earlier age due to heredity. For the early diagnosis of this disease, a comprehensive and detailed examination of the eye is required.

Computer-aided systems have been developed more widely in recent years for the detection of eye diseases and the analysis of health. The most important advantage of these systems is that they assist the health-care professional in the final diagnosis and accelerate the decision process. Glaucoma is diagnosed by analyzing fundus images. In the analysis of these images, besides statistical analysis, artificial learning approaches are also used. It is aimed to model the decision-making process of an expert decision-maker over existing images with artificial learning. Thus, the presence of glaucoma becomes digitally questionable based on the available image data. In recent years, it can be stated that artificial learning-supported systems have become more and more widespread in studies on glaucoma.

Mahdavi and Weiss [1] investigated the measurement of retinal ganglion cells, the most critical components in the human eye, with optical coherence tomography (OCT). Within the scope of their studies, they examined other studies in the literature. In their work, they said, AI-assisted glaucoma diagnosis is still a very new field, and early research has yielded remarkable results. Lee et al. [2] analyzed the thickness of the retinal nerve fiber layer (RNFL) with the ResNet34 deep learning network using a dataset of fundus images. They calculated the risk of developing glaucoma in eyes suspected of having the disease based on RNFL thickness. Wang et al. [3] aimed to present an advanced method that will automatically extract clinical concepts in examining intraocular pressure, visual acuity, and drug results in cataract and glaucoma surgeries using electronic health records. They used a rule-based classifier as the classifier in their method. Wang et al. [4] aimed to predict glaucoma progression requiring surgery by analyzing electronic health records with deep learning methods. They claimed that they successfully predicted glaucoma that requires surgery with deep learning approaches. Hashimoto et al. [5] used deep learning methods to estimate the visual field from tomography images of the spectral region. They compared CNN results with SVM and MLR (multiple linear regression) results. They said that the prediction accuracy of the deep learning method is quite satisfactory. Iqbal et al.

[6] discussed the application of deep convolutional neural networks (DCNN) in medical image analysis, specifically for interpreting endoscopic images of the gastrointestinal tract. Experimental results demonstrate the superiority of the proposed algorithm over recent techniques, showcasing its potential to assist gastroenterologists in classifying gastrointestinal abnormalities and saving time and effort in the diagnostic process. Thakur et al. [7] evaluated the accuracy of a deep learning approach to predict glaucoma from fundus images obtained in the years before the disease was diagnosed. According to their study findings, they suggested that deep learning approaches work at a logical level of accuracy. Chayan et al. [8] proposed a transfer learning approach that could classify glaucoma and compared it with pre-trained convolutional neural network models. They noted in their study that the medical assessment of glaucoma is like a black box, so it is not possible to decide on the key features behind the glaucoma prediction. Li et al. [9] generated color fundus images and visual fields data using longitudinal cohorts. In their experiments, they used artificial intelligence models to predict the presence of glaucoma, the future incidence of glaucoma, and glaucoma progression. Iqbal et al. [10] discussed the use of deep learning in medicine, particularly for diagnosing skin lesions. The proposed approach shows promise for automating and improving the efficiency of skin lesion classification, potentially assisting dermatologists and saving time and effort.

Mirzania et al. [11] conducted a systematic review of the use of deep learning applications for glaucoma detection. They assessed the risk of bias in research that was analyzed around four criteria: dataset, variable identification, algorithm, and output. The ability of deep learning algorithms to detect additional features provides important inputs for glaucoma diagnosis and progression. In the study, it was stated that data quality heavily affects the performance of deep learning systems. Hemelings [12] proposed an active and transfer learning-based convolutional neural network model for the accurate prediction of glaucoma discomfort from color fundus images. They used the ResNet-50 CNN model for transfer learning with reference to optic disk centers. They used the model parameters with the Adam optimizer at a constant learning rate. The binary cross-entropy technique was run as a cost function. Oh et al. [13] developed the glaucoma recognition and explanation system for a given prediction by making machine learning prediction. In their study, they deployed support vector machine, C5.0, Random Forest, and XGboost classifiers on five features. They claimed that their work was the first to address the concept of “explainable artificial intelligence” in eye diseases. Kamal et al. [14] focused on the concept of explicable artificial intelligence in glaucoma predictive analysis to understand risk factors in treatment planning. In the study, modeling of fuzzy membership functions, attributes, and relations between instances was provided with ANFIS architecture. They discussed the use of multi-pattern learning using fundus images and medical records. Huang et al. [15] used probabilistic deep learning to detect glaucoma from multi-model data. They modeled the glaucoma detection process using a probabilistic density layer in three different probabilistic CNN models. According to the results of the study, it was stated that the hybrid probabilistic AI model provides higher accuracy compared to the probabilistic AI models. Some research and studies on glaucoma are summarized and given in Table 1.

Table 1 Comparison of some studies in the literature

References	Method	Accuracy
[1]	Data visualization and deep learning	95%
[2]	Deep learning	95%
[3]	NLP approaches	99–100%
[4]	Deep learning	AUC: 66–73% F1: 34–42%
[5]	CNN, SVM, and multiple linear regression	CNN 74%
[6]	Deep learning	95%
[7]	Local interpretable model-agnostic explanations (LIME)	94.71%
[8]	Deep learning	AUROC: 87–91%
[10]	Deep learning	AUC: 99.5% Sensitivity: 98.0% (CI 95.5–99.4%) Specificity: 91% (CI 84.0– 96.0%)

Thompson et al. [16] stated that glaucoma can lead to progressive visual impairment and visual impairment. Deep learning algorithms were trained with OCT (optical coherence tomography) images to improve the detection of glaucoma damage on fundus images. In the final result of the study, it was said that deep learning models showed satisfactory results in the detection of microstructural damage and glaucoma progression due to glaucoma. Girard and Schmetterer [17] conducted a study on the current status and future prospects of artificial intelligence and deep learning in glaucoma. In their work, they analyzed in depth a number of topics about the role of artificial intelligence in optic nerve analysis, glaucoma diagnosis, glaucoma prediction, cost-effective imaging, and genetics in glaucoma. It should be stated that clinical professionals expect a large number of AI tools to support them in glaucoma analysis in the near future. Tarcoveanu et al. [18] reviewed classification algorithms used to predict glaucoma progression. In this context, Random Tree, C4.5, NNGE (Non-Nested Generalized Exemplars), Random Forest, MLP (Multi-Layer Perceptron), SVM, and KNN classifiers were used. The best results were reported to be obtained with Random Forest and MLP. Madadi et al. [19] introduced domain adaptation-based deep learning models for the prediction and diagnosis of glaucoma. In the new method they introduced, they developed a model that learns domain-invariant and domain-specific representations to extract general and specific attributes. They suggested that their proposed method is generalizable and provides high diagnostic accuracy. Nunez et al. [20] conducted a detailed review of SotA studies and new trends in artificial intelligence systems in glaucoma recognition and monitoring. It was stated that the five main features of such systems will accelerate adoption in clinical settings. These features are testability, compatibility, relative advantage, observability, and complexity. Singh et al. [21] stated that glaucoma can cause irreversible vision loss by damaging

the optic nerves. They used pre-trained CNN models in their work, then tried to refine hyperparameter settings with a transfer learning approach and fine-tuning. Akter et al. [22] conducted a study on the diagnosis of glaucoma by multi-feature analysis and deep learning methods. They performed multi-feature analysis and the introduction of a new cross-sectional optic nerve head feature (ONH) to facilitate existing glaucoma recognition processes. They used the VGG16 and ResNet18 CNN models for deep learning. They expressed the potential of obtaining improved results by combining the newly defined 3D section feature, ONH, with features obtained from the 2D plane. Pham et al. [23] stated that glaucoma is a disease that causes vision loss worldwide, especially in elderly people. In this context, they proposed a multimodal deep learning approach for future visual field prediction in glaucoma patients. They also used features from horizontal and vertical tomogram images in their work.

2 Motivation

Glaucoma is a critical eye disease that can be detected by detailed analyzes at a young age, usually occurs in old ages, and can cause blindness. Early diagnosis of the disease is very important for successful treatment. Considering the increasing average life expectancy, it can be said that this disorder, which usually occurs in old age, will lead to loss of vision, one of the five basic senses, and therefore to a decrease in living standards. The biggest risk factor in this disease is the slow progression of the disease, spreading over time, without showing any external symptoms. Glaucoma is detected by a specialist doctor's analysis of retinal images. Experts can diagnose glaucoma faster with computer-aided systems while making their diagnosis. Computer-assisted machine learning systems can also make important contributions to early diagnosis in the detection of glaucoma and other eye diseases. The study focuses on the methods and techniques used in machine learning-based systems that can be used for early diagnosis of glaucoma. For this purpose, an activation function that can be used in such systems has been developed.

3 Material and methods

3.1 Retinal database (ACRIMA and HRF)

In the study, datasets used in experimental evaluations to examine the effect of activation and optimizer functions on deep learning models are mentioned.

ACRIMA dataset contains 705 fundus pictures (396 glaucomatous and 309 normal images). These were obtained from glaucomatous and healthy patients at the FISABIO Oftalmologica Médica in Valencia, Spain, with their prior agreement and in compliance with the moral guidelines outlined in the 1964 Declaration of Helsinki. Glaucoma experts with years of expertise annotated every image from the ACRIMA database. They were renamed after being clipped around the optic disk [24]. The HRF dataset includes 101 glaucoma retinal images and 300 non-glaucoma

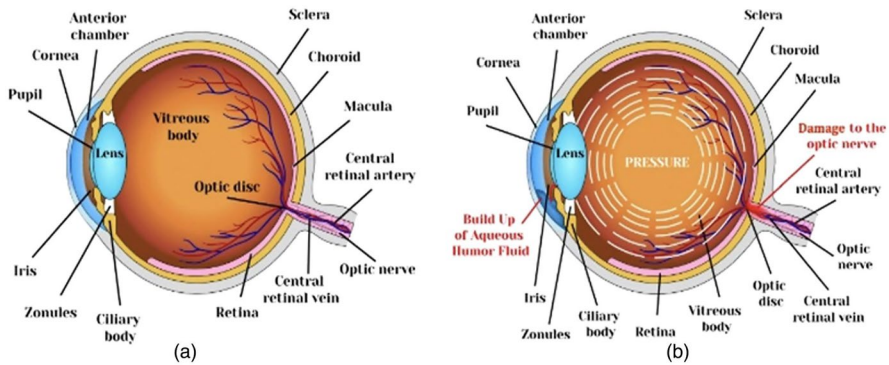


Fig. 1 Human eye structure for **a** normal and **b** glaucoma [26]

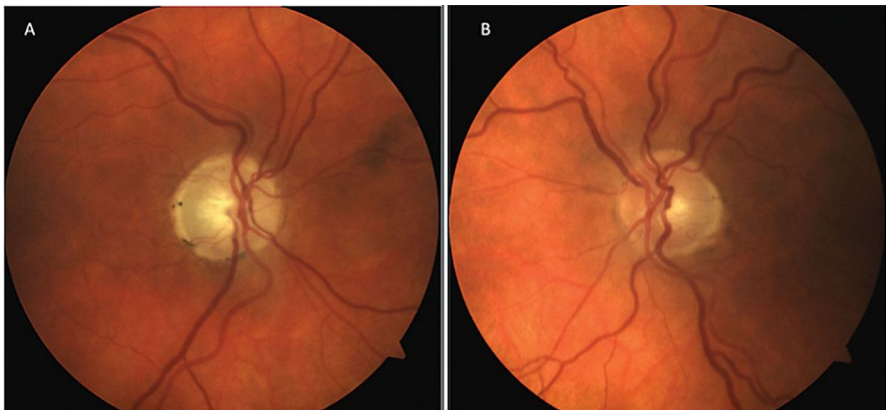


Fig. 2 Fundus images **a** normal eye and **b** glaucoma-affected eye for ACRIMA database [24]

fundus photos with pixel resolutions of 3504×2336 [25]. Figure 2a shows a fundus scan of a normal eye, while Fig. 2b shows a fundus scan of a glaucoma-affected eye. Both images were taken from the ACRIMA dataset. Figure 3a shows a normal fundus scan, while Fig. 3b displays a fundus scan with glaucoma that was taken from the HRF dataset. The optical disk of an eye was the focus for almost all fundus imaging. These photos were gathered with the help of IMAGENet, Topcon (TRC), and the annotations of medical professionals. The CIFAR-10 dataset consists of 60,000 32×32 color images in 10 classes, with 6000 images per class. There are 50,000 training images and 10,000 test images.

The normal and glaucoma-affected eye structures are shown in Fig. 1.

When Figs. 2 and 3 are examined, the differences between the fundus scan images of glaucoma-affected eyes and normal eyes are evident in the effects of glaucoma on the optic disk and retinal structures. In glaucomatous eyes, optic disk fading and an increase in the cup-to-disk ratio are observed, indicative of nerve damage. In addition, thinning of the retinal nerve fiber layer (RNFL), localized defects



Fig. 3 Fundus images **a** normal eye and **b** glaucoma-affected eye for high-resolution fundus (HRF) database [25]

of this thinning, and microbleeds around the optic disk are characteristic signs of glaucoma. As glaucoma progresses, additional findings such as notching at the edge of the optic disk and the appearance of abnormal blood vessels or bypass channels may also occur. These changes are easily recognized on fundus scans and are critical in the diagnosis and monitoring of glaucoma. Fading of the optic disk, changes in cup-to-disk ratio, and thinning and defects in the RNFL are key findings to look for in fundus images of glaucoma, and these findings are the basis for early diagnosis and management of the disease.

3.2 Convolutional neural network

Deep learning, a branch of machine learning, is studied and has a multi-layered structure [27]. Convolutional neural networks are among the most often used deep learning models (CNN). By combining feature extraction and classification, CNNs feedforward neural networks inspired by biological processes are able to discern patterns directly from picture pixels or other data [27–29]. Iqbal et al. [30] compared eight algorithms and fine-tuned their settings before running final tests. The study concludes that AdaGrad outperformed others in terms of accuracy, training time, and memory usage.

In the experimental study, two distinct CNN architectures with three and six layers were tested. The 64 filters and 3×3 filter size in the three-layer CNN architecture are used to construct the feature map. The size of the feature map is then decreased by using the max-pooling layer. The filter size used in the max-pooling layer was 2×2 . In the six-layer CNN architecture, the first convolutional layer has 64 filters and a filter size of 3×3 . The second convolutional layer has 32 filters and a filter size of 3×3 . The third convolutional layer has 16 filters and a filter size of 3×3 . The fourth convolutional layer has eight filters and a filter size of 3×3 . As with the preceding layer, a 2×2 filter size was chosen for the max-pooling layers. The collected data are prepared for the fully connected layer using the flatten layer. The density of 512 neurons was gradually changed in order for the completely connected layer to

connect to every neuron in the preceding layer. The SoftMax layer is then applied to improve classification success.

3.3 Activation functions

The performance of CNN architectures is enhanced by newly developed state-of-the-art activation functions. Since it is more possible to diversify nonlinear activation functions, it is seen that they have been developed and proposed frequently in recent years [28]. A function is considered nonlinear if it has a variable slope. In other words, if the derivative of the function is variable, that function is nonlinear, while if the derivative is constant or zero, that function is linear. Besides, nonlinear activation functions are diversified into two types as monotonic and non-monotonic. If there is a continuous increase or decrease in the min and max input intervals of a function, it has the feature of monotonicity, while if there is no continuity in the increase or decrease, it has the feature of non-monotonicity. This section presents the basic types of nonlinear monotonic and nonlinear non-monotonic activation functions.

Since the logistic function takes a value between 0 and 1, it is a basic nonlinear function suitable for dual-class classification problems. The monotonic hyperbolic tangent function is also used for binary classification where the logistic function is insufficient by solving problems with center zero. However, when the hyperbolic tangent function approaches the boundaries of the range (2, -2) on the y-axis, it encounters the vanishing gradient problem, thus reducing the performance of the architecture as a significant decrease in learning.

Presented in recent studies as a piecewise and non-monotonic nonlinear function, SupEx [31] both eliminates the vanishing gradient problem and is a zero-centered function. However, since non-particle functions can be understood more clearly in terms of mathematical expression, they are often recommended in experimental studies. For example, Swish [32], Mish [33], Logish [34], and Smish [35] functions are non-particle nonlinear non-monotonic activation functions. These functions can be used in basic form as well as with trainable alpha and beta parameters to change nonlinearity and slope.

ReLU ReLU is a nonlinear activation function used in multi-layer or deep neural networks. The formula for this function is $f(x) = \max(0, x)$. Where x is the input value, and the maximum value between zero and the input value is ReLU's output. When the input value is positive, the input value equals the input value, and when the input value is negative, the output equals zero.

LeakyReLU Similar to ReLU, LeakyReLU has a small slope for negative values rather than a flat one. Alpha, another name for this slope coefficient, is a tiny constant number. The network learns the ideal values for weights and other parameters during training, but α stays fixed. LeakyReLU can be expressed as $f(x) = \max(\alpha x, x)$. In this case, α represents the slope coefficient, and x is the activation function's input. If x is positive, the output of the function is x , but if x is negative, the output is αx instead of 0.

Swish A straightforward mathematical formula called the Swish activation function is applied in machine learning to enhance the accuracy of different models, including neural networks. The Swish function formula, defined as $f(x) = x \cdot \text{sigmoid}(\beta x)$, is quite straightforward. But its remarkable qualities and ability to increase learning model accuracy account for its simplicity. Reducing the difference between complex activation functions like repeating units and high-performance ones like ReLU is the primary goal of the Swish function.

Logish This activation function, defined as $f(x) = x \cdot \ln(1 + \text{sigmoid}(x))$, is nonlinear and non-monotonic. In Logish, the variable x is used to guarantee that the negative output has a strong regularization effect after the logarithmic operation is used to reduce the numerical range of $\text{sigmoid}(x) + 1$.

Mish The definition of this activation function, which was first presented in a research paper in 2019, is $f(x) = x \cdot \tanh(\text{Softplus}(x)) = x \cdot \tanh(\ln(1 + e^x))$. The input signal from the preceding neuron is represented by the first part of the formula, x . Two activation functions (hyperbolic tangent and Softplus) are implemented in the second part of the formula, $\tanh(\text{Softplus}(x))$. The Softplus activation function is defined as $\text{Softplus}(x) = \ln(1 + e^x)$. Mish is not monotonic, which sets it apart from other activation functions. Additionally, it has a self-regulation feature that enhances generalization and helps avoid overfitting. In essence, this means that it facilitates better learning by the neural network, preventing it from memorizing patterns in place of learning how to classify data.

Smish The Smish is a mathematical expression employed in deep learning as an activation function. Definition of the expression is $f(x) = x \cdot \tanh(\ln(1 + \sigma(x)))$, where the sigmoid function is indicated by $\sigma(x)$. The sigmoid and hyperbolic tangent functions are used by the Smish activation function to calculate its output given an input of x . Moreover, $f(x) = \alpha x \cdot \tanh(\ln(1 + \sigma(\beta x)))$ is the parameterized form of the Smish function that has been proposed by researchers. The function's performance is enhanced, and its behavior is tuned by adding two new parameters (α and β) in this parameterized version.

3.4 Proposed activation function: Trish

One of the most important criteria in activation functions is their ability to introduce nonlinearity to the neural network. This nonlinearity is crucial for the network to learn and represent complex patterns and relationships within the data. In addition, differentiability, output range, computational efficiency, saturation properties, smoothness and monotonicity, and bias/variance trade-off are important criteria for activation function selection. Inspired by these criteria, which emphasize the importance of the activation function, an approach that can provide a balance between performance and cost is adopted.

The activation function proposed in the study was developed based on the Smish function given in Eq. (1).

$$f(x) = x \tanh(\ln(1 + \text{sigmoid}(x))) \quad (1)$$

The proposed function is achieved by reciprocally transporting the tanh and sigmoid components in the Mish function. That is why it was named Trish. The basic form of the function is given in Eq. (2), and the trainable parameterized form is given in Eq. (3).

$$f(x) = x \text{sigmoid}(\ln(1 + \tanh(x))) \quad (2)$$

$$f(x) = \alpha x \text{sigmoid}(\ln(1 + \tanh(\beta x))) \quad (3)$$

In the equations, the parameters of α and β refer to the slope coefficients. The positive region is controlled by α , and the negative region is controlled with β . The x given in the equations is expressed as input data. Link: <https://github.com/serhatkic/TrishAF>

In activation functions with trainable parameters, the value of α and β allows the model to be adaptively tuned to the characteristics of the data during the training process, which contributes to better generalisation performance and improvement of gradient flow in deep networks. Moreover, thanks to the value of α and β , the activation function can produce nonzero values for negative inputs, which increases the learning capacity of the model and prevents information loss from negative inputs. These properties make α and β a critical factor that directly affects the performance of the model, especially in complex and deep learning models.

Equation (4) is obtained when the basic function is simplified.

$$f(x) = x((\tanh(x) + 1)^{-1} + 1)^{-1} \quad (4)$$

As such, it seems clear that the function is differentiable. So, when we derive it, we get Eq. (5).

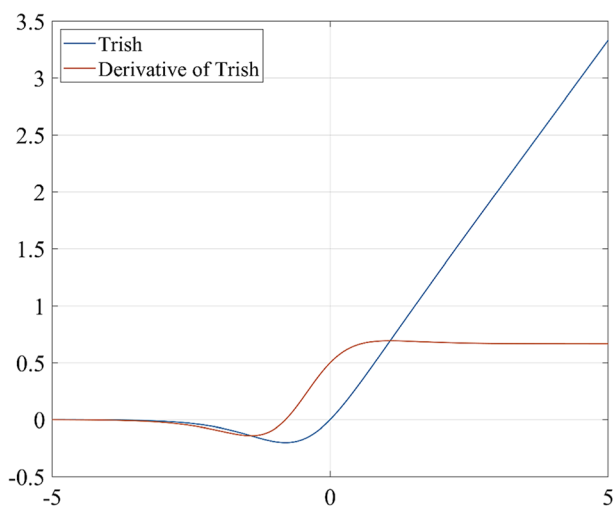


Fig. 4 Graph of proposed Trish activation function and its derivative

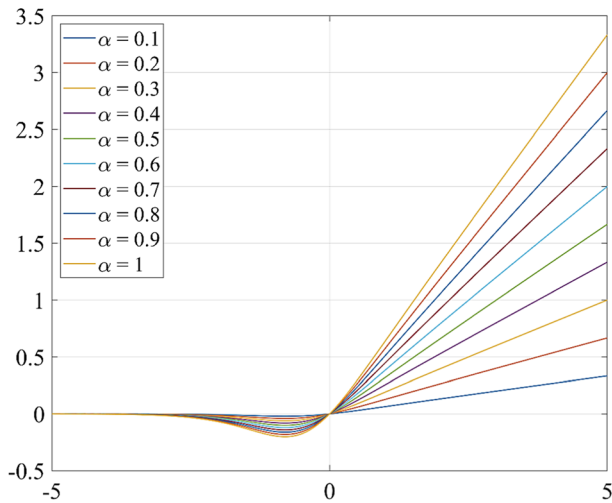


Fig. 5 Graph of proposed Trish activation function with changing α

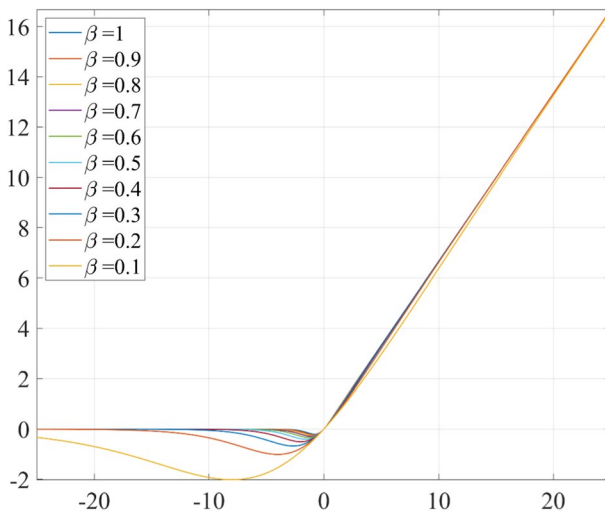


Fig. 6 Graph of proposed Trish activation function with changing β

$$f(x) = \frac{\tanh(x) + 1}{\tanh(x) + 2} + \frac{x \operatorname{sech}^2(x)}{(\tanh(x) + 2)^2} \quad (5)$$

Graphs of Trish's basic equation and its derivative are given in Fig. 4.

Graphs are given in Figs. 5 and 6 to examine the effects of Trish's trainable parameters.

The main reason why the slope values of α and β in Figs. 5 and 6 are between 0.1 and 1 in the proposed activation function is that this range optimizes the

learning process and generalization capability of the model in a balanced way. These values help the model to adapt to different data distributions, achieving both fast learning and high generalization performance. The range between 0.1 and 1 offers a balance obtained as a result of practical experience and experimental validation, which prevents the model from overreacting and gives good results in a wide range of problems. Therefore, tuning these parameters is the key to improving the performance of the model in a balanced way and adapting to the characteristics of the dataset.

Trish has increasing and decreasing monotonic intervals, so it is non-monotonic. Trish also has a variable gradient indicates nonlinearity. Also, since the proposed function does not meet the odd or even conditions, it is neither odd nor even and non-symmetric.

4 Experiments and observations

In the study, a novel non-monotonic activation function named Trish is proposed to improve accuracy of deep learning architectures. In addition, experimental analyses were carried out to examine the effect of optimizers on deep learning architectures. Experiments were performed on three datasets of high-resolution fundus (HRF), ACRIMA, and CIFAR-10. Thus, three-layer and six-layer deep learning architectures in which SGD, Adam, RmsProp, AdaDelta, AdaGrad, Adamax, and Nadam are used as optimizers, ReLU, LReLU, Mish, Swish, Smish, Logish, and the proposed Trish are used as activation functions. The CNN models used in the study were inspired by the simple models used in the literature for testing purposes, which are usually composed of 2–8 layers. The main approach here is to perform the analysis by using system resources more efficiently due to the high number of experiments. On the other hand, this approach is important to see to what extent acceptable success levels are achieved in models with the least complexity.

Thanks to CNN simulations created with HRF and ACRIMA datasets, glaucoma disease detection was performed. In addition, experimental evaluations were made through the CIFAR-10 dataset to further support the proposed activation function. Experiments were obtained by repeating 5 times. In Tables 2 and 3, experimental evaluations ACRIMA and HRF dataset made to examine the effect of parameters of the proposed activation function on deep learning are presented. In experimental studies, each model was trained for 50 epochs, learning rate 0.001, and batch size 64. The proposed model is presented in Fig. 7.

Experimental evaluations with single and double parameter were performed on the ACRIMA and HRM datasets using the Trish activation function in Tables 2 and 3. When Table 2 is examined, the highest performance result was obtained as 0.9750 in Adamax optimizer with $\alpha=1$ slope parameter of single-parameter Trish activation function. In addition, the highest performance result of 0.9642 was obtained in the Adam optimizer with the parameters $\alpha=1$ and $\beta=0.1$ in the two-parameter activation function. Table 3 compares the effect of using single and double parameter on the accuracy of the activation function on the HFR dataset using various optimization algorithms (SGD, RmsProp, Adam, AdaDelta,

Table 2 The effect of single and double parameter on the accuracy results on the activation function on the ACRIMA dataset

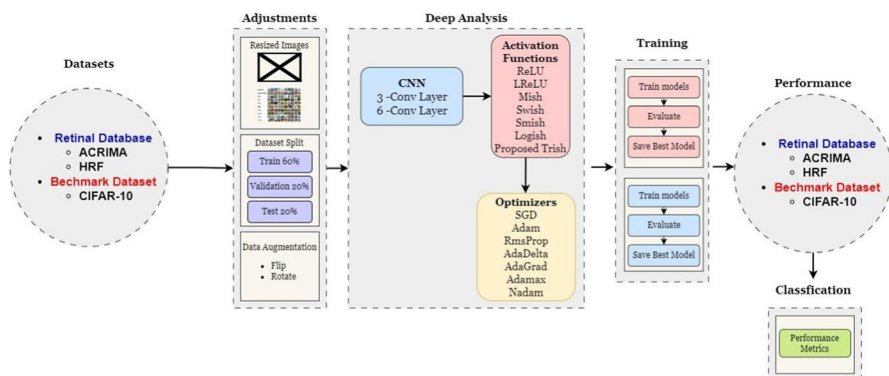
β		SGD	RsmProp	Adam	AdaDelta	AdaGrad	Adamax	Nadam
<i>Single parameter</i>								
0.10	–	0.8012	0.8337	0.8156	0.8078	0.7881	0.7660	0.8216
0.20	–	0.8145	0.8014	0.8154	0.8111	0.7562	0.7376	0.8511
0.30	–	0.8100	0.8723	0.9220	0.7998	0.7456	0.7163	0.7589
0.40	–	0.8227	0.8085	0.8369	0.7684	0.7672	0.8511	0.9362
0.50	–	0.8582	0.9007	0.9220	0.7589	0.7730	0.8723	0.8227
0.60	–	0.8014	0.9220	0.9504	0.8227	0.8440	0.8055	0.8582
0.70	–	0.9149	0.9291	0.9078	0.7872	0.8723	0.9220	0.9078
0.80	–	0.8723	0.9078	0.9291	0.8369	0.8865	0.9291	0.9146
0.90	–	0.9362	0.9291	0.9562	0.8598	0.8898	0.9360	0.9363
1.00	–	0.9362	0.9362	0.9576	0.8723	0.9391	0.9750	0.9645
<i>Double parameter</i>								
0.10	1.00	0.7895	0.9031	0.8511	0.7032	0.6909	0.7589	0.7589
0.20	0.90	0.7888	0.8794	0.9291	0.7037	0.7015	0.8369	0.9220
0.30	0.80	0.7965	0.9007	0.8865	0.7120	0.7145	0.8865	0.9149
0.40	0.70	0.8723	0.9362	0.9220	0.8156	0.7621	0.9433	0.8794
0.50	0.60	0.8936	0.8865	0.9078	0.7142	0.7589	0.9149	0.9291
0.60	0.50	0.9362	0.8227	0.9291	0.8085	0.8794	0.9007	0.9078
0.70	0.40	0.9149	0.9007	0.9362	0.8562	0.9004	0.9362	0.9125
0.80	0.30	0.9291	0.9291	0.9433	0.8511	0.8722	0.9433	0.9362
0.90	0.20	0.9563	0.9362	0.9622	0.8794	0.9220	0.9284	0.9220
1.00	0.10	0.9565	0.9374	0.9642	0.8845	0.9345	0.9365	0.9247

AdaGrad, Adamax, and Nadam). In the single-parameter case, the accuracy rate generally increases with increasing parameter value; the highest accuracy rates were generally obtained using Nadam and Adamax. In the double-parameter case, decreasing parameter values from 1.00 to 0.10 generally increased the accuracy rates, with the highest accuracy rates generally obtained at values between 1.00 and 0.10. This shows that the use of double parameters can give better results than the use of single parameter in certain situations. Among the optimization algorithms, in general, Adam, Nadam, and Adamax algorithms provided higher accuracy rates, while the performance of SGD and RmsProp algorithms remained relatively low.

It has been observed that the performance results obtained in the single- and double-parameter experimental results give lower results than the proposed Trish activation function in non-parametric structure. Therefore, all experimental results performed using the non-parametric Trish activation function are presented in Table 4. In the experimental evaluations, only the LeakyReLU activation function was taken with a parameter value of 0.9, and the other activation functions were used non-parametrically. Since the best accuracy in the proposed

Table 3 The effect of single and double parameter on the accuracy results on the activation function on the HFR dataset

α	β	SGD	RsmProp	Adam	AdaDelta	AdaGrad	Adamax	Nadam
<i>Single parameter</i>								
0.10	–	0.6719	0.6591	0.6636	0.6722	0.6802	0.6925	0.6901
0.20	–	0.5305	0.6624	0.6712	0.6766	0.6799	0.6702	0.6725
0.30	–	0.6642	0.6777	0.6699	0.6939	0.7015	0.7010	0.7001
0.40	–	0.7075	0.7019	0.7012	0.7102	0.7202	0.7136	0.7145
0.50	–	0.7035	0.7128	0.7126	0.7199	0.7212	0.7301	0.7314
0.60	–	0.6859	0.7020	0.7101	0.7214	0.7299	0.7322	0.7312
0.70	–	0.6880	0.7273	0.7302	0.7333	0.7432	0.7336	0.7299
0.80	–	0.6707	0.7079	0.6912	0.7000	0.7002	0.7036	0.7010
0.90	–	0.7092	0.7081	0.7099	0.7111	0.7185	0.7202	0.7168
1.00	–	0.7305	0.7365	0.7366	0.7401	0.7436	0.7499	0.7312
<i>Double parameter</i>								
0.10	1.00	0.6883	0.6760	0.6663	0.6931	0.6929	0.7226	0.6809
0.20	0.90	0.5603	0.6609	0.6925	0.6931	0.6935	0.7001	0.6668
0.30	0.80	0.6787	0.7072	0.7136	0.6939	0.6880	0.7135	0.7231
0.40	0.70	0.7163	0.7376	0.7333	0.7234	0.6990	0.7012	0.7092
0.50	0.60	0.7160	0.7192	0.7305	0.7034	0.7106	0.7163	0.7021
0.60	0.50	0.7447	0.7413	0.7413	0.7505	0.7347	0.7418	0.7021
0.70	0.40	0.7305	0.7572	0.7588	0.7356	0.7234	0.7290	0.6817
0.80	0.30	0.7092	0.7288	0.7234	0.7317	0.7163	0.7317	0.7168
0.90	0.20	0.7234	0.7255	0.7447	0.7376	0.7317	0.7653	0.7106
1.00	0.10	0.7543	0.7602	0.7755	0.7731	0.7319	0.7747	0.7207

**Fig. 7** Flowchart of the proposed model

activation function is in the non-parametric structure, α and β values are equal to 1 in the experiments.

Table 4 Three-layer CNN experiments on ACRIMA dataset

Optimizer	Activation function	Train			Val		
		Loss	Accuracy	<i>F</i> -score	Loss	Accuracy	<i>F</i> -score
SGD	ReLU	0.0860	0.9574	0.9598	0.3179	0.8298	0.8409
	LeakyReLU	0.1852	0.9291	0.9251	0.1759	0.9362	0.9346
	Mish	0.0443	0.9811	0.9821	0.2286	0.9433	0.9409
	Swish	0.1479	0.9385	0.9420	0.2977	0.9220	0.9221
	Smish	0.1397	0.9433	0.9464	0.3464	0.9078	0.9187
	Logish	0.2216	0.9125	0.8935	0.3206	0.9078	0.9096
	Trish	0.0916	0.9645	0.9665	0.2482	0.9362	0.9346
RmsProp	ReLU	0.0424	0.9858	0.9866	0.2708	0.9433	0.9500
	LeakyReLU	0.1583	0.9338	0.9295	0.2600	0.8723	0.8601
	Mish	0.1812	0.9598	0.9621	0.2813	0.8865	0.9001
	Swish	0.0838	0.9716	0.9732	0.2316	0.8723	0.8784
	Smish	0.0212	0.9929	0.9933	0.1613	0.9291	0.9284
	Logish	0.0509	0.9811	0.9821	0.2223	0.8652	0.8630
	Trish	0.0602	0.9716	0.9732	0.1934	0.9362	0.9255
Adam	ReLU	0.0911	0.9691	0.9710	0.9772	0.7589	0.7692
	LeakyReLU	0.2008	0.9644	0.9545	0.3517	0.9291	0.9192
	Mish	0.2875	0.9430	0.9464	0.3940	0.8369	0.8288
	Swish	0.1220	0.9596	0.9621	0.1960	0.9433	0.9409
	Smish	0.0146	0.9976	0.9978	0.1677	0.9574	0.9625
	Logish	0.0335	0.9810	0.9821	0.2816	0.9007	0.8942
	Trish	0.0106	1.0000	1.0000	0.1735	0.9576	0.9534
AdaDelta	ReLU	0.3111	0.8812	0.8884	0.4238	0.8782	0.8767
	LeakyReLU	0.2052	0.9264	0.9187	0.3539	0.8723	0.8875
	Mish	0.2791	0.8979	0.8799	0.5112	0.8369	0.838
	Swish	0.3646	0.8527	0.8616	0.5680	0.8014	0.7885
	Smish	0.4289	0.8171	0.8161	0.6002	0.8227	0.8072
	Logish	0.4866	0.7981	0.7982	0.6178	0.7943	0.8005
	Trish	0.2805	0.8884	0.8830	0.4816	0.8723	0.8692
AdaGrad	ReLU	0.0436	0.9905	0.9911	0.2417	0.9362	0.9438
	LeakyReLU	0.0695	0.9715	0.9732	0.2186	0.9291	0.9375
	Mish	0.0788	0.9644	0.9665	0.3284	0.9078	0.9187
	Swish	0.1182	0.9477	0.9509	0.3655	0.8936	0.8697
	Smish	0.2289	0.9002	0.8821	0.4698	0.8511	0.8413
	Logish	0.1860	0.9311	0.9232	0.4459	0.8582	0.8476
	Trish	0.1462	0.9430	0.9464	0.2349	0.9391	0.9475

Table 4 (continued)

Optimizer	Activation function	Train			Val		
		Loss	Accuracy	<i>F</i> -score	Loss	Accuracy	<i>F</i> -score
Adamax	ReLU	0.0125	1.0000	1.0000	0.1566	0.9645	0.9596
	LeakyReLU	0.3757	0.9454	0.9487	0.7826	0.8582	0.8659
	Mish	0.0264	0.9929	0.9933	0.1972	0.9645	0.9505
	Swish	0.0115	1.0000	1.0000	0.1747	0.9645	0.9596
	Smish	0.0295	0.9834	0.9844	0.1864	0.9574	0.9625
	Logish	0.0275	0.9905	0.9911	0.2028	0.9433	0.9500
	Trish	0.0247	0.9952	0.9955	0.1495	0.9716	0.9750
Nadam	ReLU	0.0267	0.9905	0.9911	0.1728	0.9504	0.9471
	LeakyReLU	0.2600	0.9596	0.9621	0.6051	0.8582	0.8385
	Mish	0.0211	0.9905	0.9911	0.2323	0.9291	0.9375
	Swish	0.0512	0.9881	0.9888	0.3116	0.8794	0.8663
	Smish	0.0279	0.9905	0.9911	0.1974	0.9149	0.9159
	Logish	0.0190	0.9952	0.9955	0.1789	0.9574	0.9625
	Trish	0.0294	0.9881	0.9888	0.1759	0.9645	0.9688

The results of the experiments performed for the detection of glaucoma disease on the ACRIMA dataset using the three-layer CNN architecture are given in Table 4. In the experiments, how the optimizer and activation functions affect the accuracy was examined. The best val-loss, val-acc, and val-fscore values in the results of the experimental evaluations for the detection of glaucoma disease on the ACRIMA dataset in the three-layer CNN architecture were obtained with proposed Trish activation function in all optimizers. The highest performance is with the proposed activation function Trish and Adamax optimizer, with 0.1495 loss, 0.9716 accuracy, and 0.9750 *f*-score. As a result, it is seen that the proposed Trish activation function gives better results than different optimizers and other activation functions in the three-layer CNN architecture. In the experimental

Table 5 Confidence interval profile of Trish activation function on three CNN layers for ACRIMA

	95% CI
SGD	0.9362 ± 0.001956
RmsProp	0.9433 ± 0.001851
Adam	0.9576 ± 0.001612
AdaDelta	0.8782 ± 0.002617
AdaGrad	0.9391 ± 0.001914
Adamax	0.9716 ± 0.001329
Nadam	0.9645 ± 0.001481

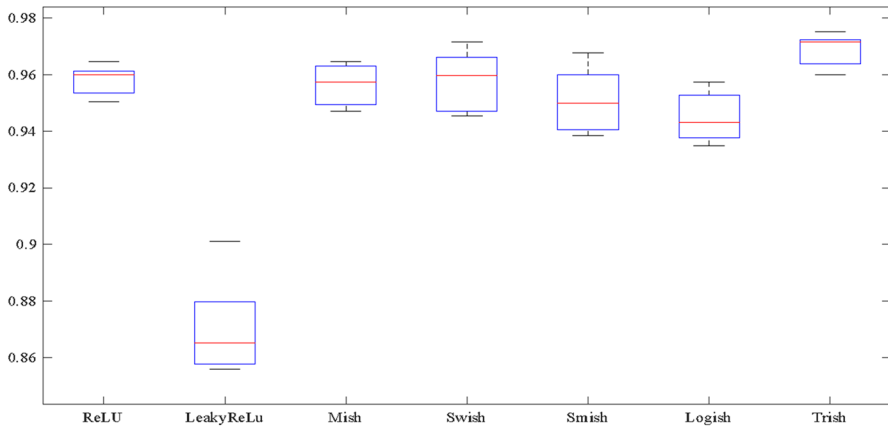


Fig. 8 Accuracy box plot of activation functions in three-layer CNN model of dataset

environment created, the lowest accuracy was obtained with Adam optimizer and non-parametric ReLU activation function as 0.7589 accuracy.

The 95% confidence intervals for the optimizers were calculated and presented in Table 5. The Adamax optimizer showed a better confidence interval with a higher mean and a smaller interval width compared to other commonly used optimizers.

In Fig. 8, the performance comparison of the activation functions according to the Adamax optimizer, where the best result was obtained for the detection of glaucoma disease on the ACRIMA dataset, is given. Better validation accuracy value than other non-monotonic functions was obtained by the proposed Trish activation function. In addition, it is observed in Fig. 8 that the proposed Trish and ReLU activation functions show similar behavior. In Fig. 8, it is seen that the Trish activation function except ReLU has a smaller box graph size, and the distance of the whiskers to the box is less compared to other activation functions. Thus, in light of the results obtained, it has been evaluated that the proposed Trish activation function works more stable than other functions. The small whisker lengths in the box plots and the small difference between the lowest and highest performance values obtained in the current experiments are an indication of the stability of the activation functions. The proposed Trish activation function in the results gives more stable results in the detection of glaucoma with the three-layer CNN model compared to other activation functions.

In Table 6, the results of the experiments performed for the detection of glaucoma disease on the ACRIMA dataset using the six-layer CNN architecture are given. The best val_loss, val_acc, and val_fscore values were obtained in all optimizers with the proposed Trish activation function in the results of the experiments performed for the detection of glaucoma disease on the ACRIMA dataset in the six-layer CNN architecture. The highest performance was achieved with the proposed activation function Trish and Adamax optimizer at 0.2406 loss, 0.9722 accuracy, and 0.9842 *f*-score values. As a result, it is seen that the proposed activation function gives better results than all activation functions with different optimizers in the six-layer

Table 6 Six-layer CNN experiments on ACRIMA dataset

Optimizer	Activation	Train			Val		
		Loss	Accuracy	<i>F</i> -score	Loss	Accuracy	<i>F</i> -score
SGD	ReLU	0.2225	0.9145	0.9076	0.3180	0.8652	0.8813
	LeakyReLU	0.2626	0.8955	0.8897	0.3812	0.9007	0.8942
	Mish	0.0700	0.9762	0.9777	0.3726	0.8440	0.8534
	Swish	0.0363	0.9905	0.9911	0.2896	0.8582	0.8659
	Smish	0.0542	0.9789	0.9679	0.3337	0.9007	0.9034
	Logish	0.1215	0.9549	0.9576	0.2400	0.9291	0.9375
	Trish	0.0906	0.9667	0.9688	0.3410	0.9149	0.9159
RmsProp	ReLU	0.1081	0.9667	0.9688	0.382	0.8440	0.8625
	LeakyReLU	0.2283	0.9382	0.9420	0.2022	0.9220	0.913
	Mish	0.1001	0.9715	0.9491	0.3787	0.8440	0.8534
	Swish	0.0875	0.9644	0.9665	0.2096	0.9078	0.9096
	Smish	0.063	0.9857	0.9746	0.4158	0.8936	0.9062
	Logish	0.158	0.9549	0.9576	0.3584	0.9007	0.8942
	Trish	0.0909	0.9667	0.9688	0.3415	0.9291	0.9284
Adam	ReLU	0.0744	0.9739	0.9634	0.4753	0.7730	0.8000
	LeakyReLU	0.1638	0.9739	0.9754	0.3933	0.9007	0.8942
	Mish	0.0385	0.9905	0.9911	0.2568	0.9291	0.9375
	Swish	0.0870	0.9620	0.9522	0.2153	0.9433	0.9500
	Smish	0.0386	0.9810	0.9821	0.3112	0.8936	0.8880
	Logish	0.0109	10.000	10.000	0.3152	0.8865	0.8817
	Trish	0.0227	0.9881	0.9888	0.2282	0.9291	0.9192
AdaDelta	ReLU	0.2085	0.9169	0.9098	0.4959	0.7872	0.7851
	LeakyReLU	0.1209	0.9501	0.9531	0.2373	0.9149	0.9159
	Mish	0.2026	0.9169	0.9219	0.4577	0.9078	0.9005
	Swish	0.2896	0.8860	0.8687	0.6319	0.5390	0.5389
	Smish	0.3643	0.8219	0.8205	0.6388	0.8014	0.7976
	Logish	0.3153	0.8717	0.8795	0.5780	0.8369	0.8380
	Trish	0.2575	0.8955	0.8777	0.4587	0.8440	0.8351
AdaGrad	ReLU	0.0835	0.9662	0.9677	0.4811	0.9204	0.9271
	LeakyReLU	0.0591	0.9786	0.9799	0.3083	0.9149	0.9159
	Mish	0.0744	0.9644	0.9545	0.3825	0.8369	0.8288
	Swish	0.0602	0.9762	0.9656	0.4077	0.8440	0.8534
	Smish	0.1292	0.9501	0.9531	0.5123	0.9078	0.9096
	Logish	0.0874	0.9644	0.9665	0.5134	0.8794	0.8938
	Trish	0.0731	0.9691	0.9710	0.4229	0.9291	0.9284

Table 6 (continued)

Optimizer	Activation	Train			Val		
		Loss	Accuracy	F-score	Loss	Accuracy	F-score
Adamax	ReLU	0.0131	0.9976	0.9978	0.2160	0.9220	0.9221
	LeakyReLU	0.3258	0.9620	0.9522	11.358	0.9220	0.9221
	Mish	0.1025	0.9715	0.9732	24.129	0.5674	0.5365
	Swish	0.0293	0.9857	0.9866	0.2681	0.9007	0.9034
	Smish	0.0450	0.9857	0.9866	0.4783	0.7801	0.7514
	Logish	0.0429	0.9857	0.9866	0.4371	0.8298	0.8135
	Trish	0.0388	0.9952	0.9955	0.2406	0.9722	0.9842
Nadam	ReLU	0.0530	0.9786	0.9799	0.3035	0.8511	0.8231
	LeakyReLU	0.2948	0.9549	0.9455	0.9958	0.8865	0.8909
	Mish	0.0203	0.9952	0.9955	0.2563	0.8723	0.8601
	Swish	0.0264	0.9881	0.9888	0.3161	0.8936	0.8971
	Smish	0.0667	0.9786	0.9799	0.3995	0.8156	0.8192
	Logish	0.1889	0.9501	0.9531	0.3571	0.8440	0.8168
	Trish	0.0073	0.9976	0.9978	0.2223	0.9362	0.9346

Table 7 Confidence interval profile of Trish activation function on six CNN layers for ACRIMA

	95% CI
SGD	0.9291 ± 0.002054
RmsProp	0.9291 ± 0.002054
Adam	0.9433 ± 0.001851
AdaDelta	0.8440 ± 0.002903
AdaGrad	0.9291 ± 0.002054
Adamax	0.9716 ± 0.001329
Nadam	0.9362 ± 0.001956

CNN architecture. In the experimental environment created, the lowest accuracy was obtained with the AdaDelta optimizer, with an accuracy of 0.5390 with the non-parametric Swish activation function, while an accuracy of 0.8440 with the Trish activation function suggested in the same optimizer.

The 95% confidence intervals calculated for the optimizers are given in Table 7. The proposed Trish activation function showed a better confidence interval with a higher mean and a smaller interval width in the Adamax optimizer compared to other common optimizers used in the study.

In Fig. 9, the performance comparison of the activation functions according to the Adamax optimizer, where the best result was obtained for the detection of glaucoma disease on the ACRIMA dataset, is given. The proposed Trish activation function has a higher validation accuracy value than other non-monotonic activation

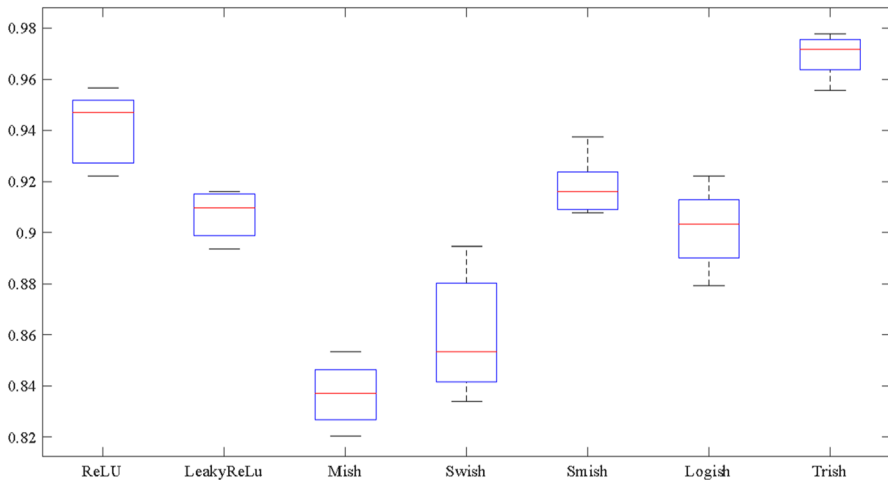


Fig. 9 Accuracy box plot of activation functions in six-layer CNN model of ACRIMA dataset with Adamax optimizer

functions. In addition, it is observed in Fig. 9 that the Trish activation functions proposed with LeakyReLU show similar behavior. It is seen that the box graph size is small, and the distance of the whiskers to the box is less compared to other activation functions except LeakyReLU. Thus, the distance between the extreme values in the box plot is very small. Also, the small difference between the accuracy rates shows that the proposed activation function is stable. In light of the results obtained, it is observed that the proposed Trish activation function works more stable than other functions. It is seen that the suggested Trish activation function box plot lengths are shorter than the other box plot lengths, and the distance of the whiskers to the box is short. The small whisker lengths in the box plots and the small difference between the lowest and highest performance values obtained in the current experiments are an indication of the stability of the activation function. The proposed Trish activation function in the results gives more stable results in the detection of glaucoma with the three-layer CNN model compared to other activation functions. When the other activation functions are examined, it is seen that especially the lower whiskers are quite far from the box plot.

In Table 8, the results of the experiments performed to examine the effect of optimizer and activation functions for the detection of glaucoma disease over the HRF dataset using the six-layer CNN architecture are given. The highest performance was achieved with the recommended activation function Trish and Adamax optimizer at 0.6429 loss, 0.7538 accuracy, and 0.7874 *f*-score values. When the experimental results are examined, it can be said that the proposed Trish activation function works actively in all optimizers, so it works both consistently and stably. As a result, it is seen that the proposed activation function gives good results in three-layer CNN architecture, from different optimizers and all activation functions. In addition, in the experimental evaluations performed with the HRF dataset, it was observed that

Table 8 Three-layer CNN experiments on high-resolution fundus (HRF) dataset

Optimizer	Activation	Train			Val		
		Loss	Accuracy	<i>F</i> -score	Loss	Accuracy	<i>F</i> -score
SGD	ReLU	0.4446	0.8096	0.8180	0.6463	0.6692	0.6374
	LeakyReLU	0.5511	0.7269	0.7224	0.5901	0.5384	0.69374
	Mish	0.4408	0.8038	0.8014	0.6125	0.6923	0.7499
	Swish	0.425	0.8096	0.818	0.5678	0.7153	0.7687
	Smish	0.4647	0.7942	0.7922	0.5468	0.7153	0.7687
	Logish	0.4733	0.7673	0.7665	0.5612	0.6923	0.7499
	Trish	0.4656	0.7807	0.7904	0.5734	0.7384	0.7499
RmsProp	ReLU	0.5231	0.7711	0.7647	0.7212	0.7076	0.6687
	LeakyReLU	0.4103	0.8038	0.8069	0.7609	0.723	0.6812
	Mish	–	–	–	–	–	–
	Swish	0.5002	0.7769	0.7867	0.6153	0.7153	0.7687
	Smish	0.4197	0.8249	0.8327	0.7289	0.723	0.7062
	Logish	0.4768	0.7711	0.7757	0.7009	0.7384	0.6937
	Trish	0.3654	0.823	0.8253	0.7043	0.7361	0.7574
Adam	ReLU	0.4838	0.773	0.783	0.5984	0.7461	0.7937
	LeakyReLU	0.4448	0.7788	0.772	0.8721	0.6692	0.6374
	Mish	–	–	–	–	–	–
	Swish	–	–	–	–	–	–
	Smish	0.3776	0.8134	0.8106	0.6633	0.7307	0.6874
	Logish	0.4384	0.7942	0.8033	0.6884	0.7153	0.7437
	Trish	0.3997	0.823	0.8143	0.6355	0.7537	0.7871
AdaDelta	ReLU	0.5218	0.748	0.7536	0.5975	0.7153	0.7687
	LeakyReLU	0.5306	0.7403	0.7463	0.5852	0.7384	0.7874
	Mish	–	–	–	–	–	–
	Swish	0.5477	0.7423	0.7536	0.6165	0.7284	0.7674
	Smish	–	–	–	–	–	–
	Logish	–	–	–	–	–	–
	Trish	0.3925	0.8111	0.8144	0.5941	0.7308	0.7677
AdaGrad	ReLU	0.4315	0.7749	0.7794	0.6232	0.6538	0.7187
	LeakyReLU	0.421	0.8019	0.8051	0.6412	0.6999	0.6624
	Mish	0.433	0.8038	0.7959	0.5622	0.6769	0.7249
	Swish	–	–	–	–	–	–
	Smish	–	–	–	–	–	–
	Logish	–	–	–	–	–	–
	Trish	0.4568	0.7769	0.7757	0.5719	0.7307	0.7562

Table 8 (continued)

Optimizer	Activation	Train			Val		
		Loss	Accuracy	<i>F</i> -score	Loss	Accuracy	<i>F</i> -score
Adamax	ReLU	0.413	0.8192	0.8162	0.7467	0.6153	0.5937
	LeakyReLU	0.6629	0.7692	0.7738	0.8203	0.6846	0.6499
	Mish	0.4055	0.8192	0.8161	0.6045	0.7076	0.6687
	Swish	0.3946	0.8115	0.8143	0.5939	0.7307	0.7687
	Smish	0.4148	0.8	0.7977	0.6201	0.7076	0.7187
	Logish	0.3815	0.8173	0.8088	0.6618	0.7076	0.7062
	Trish	0.4123	0.7903	0.7886	0.6429	0.7538	0.7874
Nadam	ReLU	0.4382	0.798	0.8014	0.5958	0.6923	0.7499
	LeakyReLU	0.3927	0.8134	0.8216	0.6381	0.7153	0.7687
	Mish	–	–	–	–	–	–
	Swish	–	–	–	–	–	–
	Smish	0.256	0.8865	0.8915	0.7422	0.6538	0.6562
	Logish	–	–	–	–	–	–
	Trish	0.3918	0.8307	0.8419	0.5693	0.7184	0.7499

Table 9 Confidence interval profile of Trish activation function on three CNN layers for HRF

	95% CI
SGD	0.7384 ± 0.003517
RmsProp	0.7361 ± 0.003527
Adam	0.7538 ± 0.003448
AdaDelta	0.7308 ± 0.003549
AdaGrad	0.7307 ± 0.003550
Adamax	0.7538 ± 0.003447
Nadam	0.7184 ± 0.003599

some activation functions could not produce accuracy in all optimizer functions due to the characteristics of the dataset.

The 95% confidence intervals calculated for different optimizers are given in Table 9. The proposed Trish function showed a better confidence interval with a higher mean and a smaller interval width with the Adamax optimizer than the other common optimizers used in the study.

In Fig. 10, the performance comparison of the activation functions with the Adamax optimizer, where the best result was obtained for the detection of glaucoma disease on the HRF dataset, is given. It is seen that the proposed Trish activation function achieves higher results in validation accuracy value than other non-monotonic activation functions. When the figure is examined, it is seen that the box graph size of the proposed Trish activation function is smaller, and the

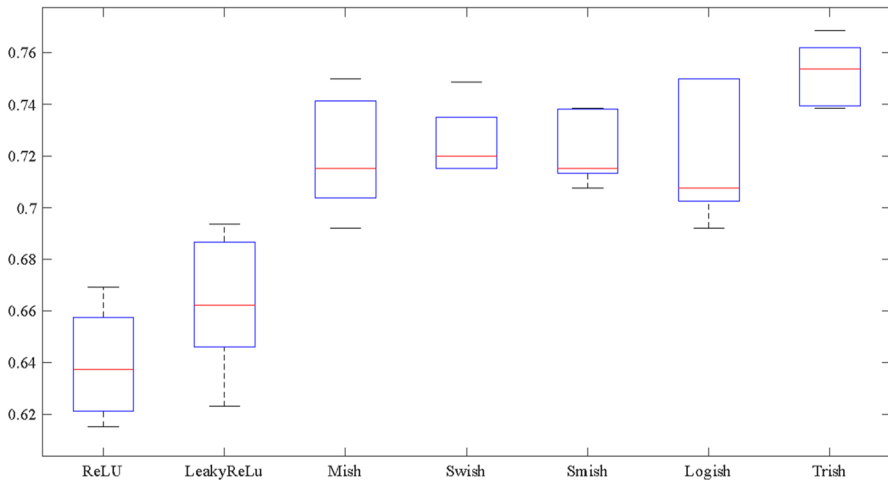


Fig. 10 Accuracy box plot of activation functions in three-layer CNN model of HRF dataset with Adamax optimizer

distance of the whiskers to the box of the proposed Trish activation function is less compared to the other activation functions. However, since the box size of the proposed Trish activation function is close to the box size of the Smish and Swish activation functions, they exhibit similar behavior. Also, the small difference between the accuracy values shows that the proposed activation functions are stable. Thus, in light of the results obtained, it is observed that the proposed Trish activation function works more stable than other functions. It is seen that the length of the box plot of the proposed Trish activation function is shorter than the other box plot lengths, and the distance of the whiskers to the box is short. The small whisker lengths in the box plots and the small difference between the lowest and highest performance values obtained in the current experiments are an indication of the stability of the activation function. The proposed Trish activation function in the results gives more stable results in the detection of glaucoma with the three-layer CNN model compared to other activation functions (Table 10).

The 95% confidence intervals calculated for different optimizers are given in Table 11. The proposed Trish activation function gave a better confidence interval with a higher mean and a smaller interval width with the Adamax optimizer compared to other common optimizers used in the study.

In Fig. 11, the performance comparison of the activation functions used with the Adamax optimizer, where the best result was obtained to measure the performance of the proposed activation function over the CIFAR-10 dataset, is given. The validation accuracy value of the proposed Trish activation function was higher than other non-monotonic activation functions. When the box plot of the proposed Trish activation function is examined, it is smaller than the box plot size of the other activation functions, and the distance of the whiskers to the box is less. It is also seen that there is a small difference between the accuracy values. In addition, the smallest

Table 10 Three-layer CNN experiments on CIFAR-10 dataset

Optimizer	Activation	Train			Val		
		Loss	Accuracy	<i>F</i> -score	Loss	Accuracy	<i>F</i> -score
SGD	ReLU	0.4852	0.8292	0.8284	11.559	0.6693	0.6680
	LeakyReLU	0.8085	0.7215	0.7101	11.401	0.6357	0.6232
	Mish	0.4606	0.8377	0.8376	11.824	0.6677	0.6694
	Swish	0.4612	0.8356	0.8345	11.823	0.6714	0.6731
	Smish	0.4710	0.8352	0.8342	11.537	0.6608	0.6626
	Logish	0.4771	0.8304	0.8300	11.849	0.6673	0.6687
	Trish	0.4886	0.8268	0.8249	11.152	0.6757	0.6758
RmsProp	ReLU	0.4506	0.8396	0.8383	13.571	0.6529	0.6544
	LeakyReLU	0.7684	0.7352	0.7271	10.824	0.6456	0.6398
	Mish	0.3742	0.8654	0.8668	15.048	0.6642	0.6671
	Swish	0.3845	0.8614	0.8621	15.071	0.6493	0.6526
	Smish	0.3877	0.8620	0.8619	14.548	0.6616	0.6636
	Logish	0.3824	0.8635	0.8637	15.026	0.6549	0.6554
	Trish	0.7321	0.7460	0.7395	10.106	0.6615	0.6676
Adam	ReLU	0.4895	0.8238	0.8235	11.989	0.6603	0.6624
	LeakyReLU	0.7634	0.7331	0.7262	11.265	0.6407	0.6351
	Mish	0.3652	0.8673	0.8677	14.185	0.6561	0.6591
	Swish	0.3617	0.8677	0.8676	13.178	0.6682	0.6735
	Smish	0.3686	0.8661	0.8661	13.542	0.6660	0.6668
	Logish	0.3579	0.8712	0.8709	13.471	0.6722	0.6722
	Trish	0.4063	0.8519	0.8513	12.561	0.6797	0.6850
AdaDelta	ReLU	18.001	0.3574	0.0840	17.799	0.3666	0.0883
	LeakyReLU	16.504	0.4164	0.2221	16.306	0.4259	0.2355
	Mish	16.630	0.4075	0.1662	16.479	0.4137	0.1802
	Swish	16.866	0.4032	0.1439	16.791	0.4028	0.1582
	Smish	17.318	0.3843	0.1076	17.187	0.3864	0.1166
	Logish	17.169	0.3885	0.1238	16.999	0.3943	0.1220
	Trish	16.847	0.4012	0.1534	16.703	0.4065	0.1641
AdaGrad	ReLU	12.832	0.5457	0.4590	12.952	0.5369	0.4576
	LeakyReLU	12.920	0.5519	0.4624	13.030	0.5433	0.4648
	Mish	12.796	0.5536	0.4651	12.925	0.5440	0.4646
	Swish	12.897	0.5477	0.4583	13.047	0.5384	0.4544
	Smish	13.154	0.5337	0.4359	13.161	0.5368	0.4372
	Logish	12.978	0.5396	0.4434	12.905	0.5422	0.4496
	Trish	12.889	0.5511	0.4629	12.876	0.5406	0.4637

Table 10 (continued)

Optimizer	Activation	Train			Val		
		Loss	Accuracy	<i>F</i> -score	Loss	Accuracy	<i>F</i> -score
Adamax	ReLU	0.6115	0.7849	0.7800	10.772	0.6578	0.6616
	LeakyReLU	0.7677	0.7345	0.7250	11.606	0.6301	0.6199
	Mish	0.4705	0.8351	0.8329	13.574	0.6290	0.6314
	Swish	0.4944	0.8236	0.8227	10.645	0.6814	0.6814
	Smish	0.5319	0.8151	0.8115	10.327	0.6784	0.6795
	Logish	0.5141	0.8177	0.8156	10.058	0.6895	0.6904
	Trish	0.5792	0.7972	0.7938	0.9523	0.6964	0.6958
Nadam	ReLU	0.4874	0.8238	0.8233	10.908	0.6890	0.6906
	LeakyReLU	0.7493	0.7411	0.7317	10.805	0.6530	0.6425
	Mish	0.7488	0.7381	0.7318	11.229	0.6380	0.6319
	Swish	0.7556	0.7374	0.7300	11.306	0.6413	0.6334
	Smish	0.7563	0.7392	0.7308	11.105	0.6513	0.6421
	Logish	0.8466	0.7083	0.6938	10.711	0.6522	0.6383
	Trish	0.4920	0.8256	0.8235	10.558	0.6898	0.6910

Table 11 Confidence interval profile of Trish activation function on three CNN layers for CIFAR-10

	95% CI
SGD	0.6757 ± 0.003746
RmsProp	0.6615 ± 0.003786
Adam	0.6797 ± 0.003734
AdaDelta	0.4065 ± 0.003930
AdaGrad	0.5406 ± 0.003988
Adamax	0.6964 ± 0.003679
Nadam	0.6898 ± 0.003701

value of Trish activation function is above LeakyReLU, Mish, and Smish activation functions. Thus, in light of the results obtained, it is observed that the proposed Trish activation function works more stable than other functions. It is seen that the CIFAR-10 dataset gives different results in different activation functions and exhibits an unstable behavior. As a result, it is seen that the proposed Trish activation function exhibits a more stable structure on the CIFAR-10 dataset than other activation functions. The model was successfully optimized using an Adamax optimizer with $\text{beta}_1 = 0.9$, $\text{beta}_2 = 0.99$, $\text{epsilon} = 1\text{e}-07$, and a beginning learning rate of 0.001.

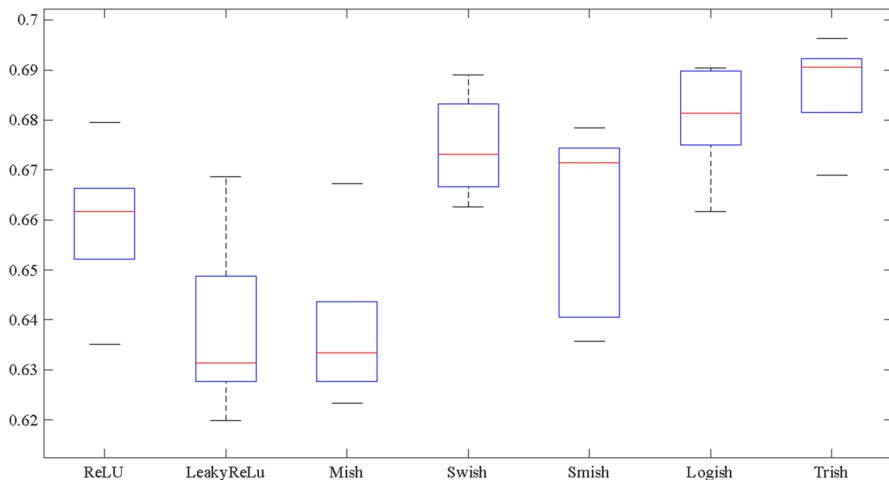


Fig. 11 Accuracy box plot of activation functions in three-layer CNN model of CIFAR-10 dataset with Adamax optimizer

5 Discussion

Glaucoma is a chronic eye disorder characterized by damage to the optic nerve, which leads to progressive loss of vision, typically in the peripheral field. It is caused by an increase in intraocular pressure (IOP) within the eye, which can lead to damage to the nerve fibers that carry visual information to the brain. The most common form of glaucoma is primary open-angle glaucoma, which is characterized by a slow and painless loss of peripheral vision. Preventing vision loss caused by glaucoma, which is a leading cause of blindness, can be achieved through early diagnosis and treatment [36].

Computer-aided systems are widely used in glaucoma classification by using techniques such as machine learning, deep learning, and image processing to analyze and classify the images. In ref [37] is a study that proposed a method for classifying glaucoma using color fundus images and a CNN with transfer learning. The study used a dataset of color fundus images of eyes with glaucoma and healthy eyes. The CNN model was pre-trained on a large dataset and then fine-tuned on the dataset of color fundus images. The study reported that the proposed method achieved an accuracy of 96.6% in classifying glaucoma and healthy eyes, demonstrating the potential of using color fundus images and CNNs with transfer learning for automatic glaucoma classification.

In [38] used a dataset of retinal images of eyes with glaucoma and healthy eyes, and a CNN as the classification network. They also used methods such as data augmentation and transfer learning to increase the performance of the network. This study reported that the proposed network achieved an accuracy of 96.5% in classifying glaucoma and healthy eyes, demonstrating the potential of using deep learning-based approaches and CNN method for automatic glaucoma classification in retinal images. In another study [39] that proposed an improved swarm

optimization method for glaucoma classification using deep features. The study used a combination of a swarm optimization algorithm called SEGSO and a deep learning model called VGGNet, to optimize the features extracted from eye fundus images for glaucoma classification. The authors used a dataset of eye fundus images and trained the model on this dataset. They then evaluated the performance of the model on a separate test dataset and reported that the proposed method achieved high accuracy in glaucoma classification, with an accuracy of 99.67%. The study demonstrates that the suggested method is an efficacious method for glaucoma classification using deep features and can be a good option for glaucoma diagnosis. Other ref [40] that suggested an hybrid feature space composed of texture information and transfer learning for glaucoma classification. The authors used a dataset of eye fundus images and trained a deep learning model using the texture features extracted from the images, and transfer learning technique. They then evaluated the performance of the model on a separate test dataset and reported that the proposed method achieved high accuracy in glaucoma classification, with an accuracy of 93.61%. The study demonstrates that the suggested method is a powerful system for glaucoma classification using texture information and transfer learning and can be a good option for glaucoma diagnosis.

Optimizers and activation functions, which are components of complex deep learning architectures, significantly affect network accuracies. For this reason, this study analyzes how optimizers and activation functions improve the performance of three-layer and six-layer CNN architectures designed for glaucoma classification. Architectural variations with several existing optimizers and activation functions were created and tested. In addition, a new activation function, Trish, is proposed and compared with architectures using other activation functions. This architecture, designed specifically for glaucoma classification, was found to be highly successful. This study by intervening in the internal structure of deep learning architectures, especially the activation function, is scientifically seminal. Comparison of performance regarding glaucoma classification is given in Table 12.

It has been observed that the proposed activation function satisfies the performance in deep learning architectures with the appropriate optimizer. The six-layer CNN activation function proposed with Adamax optimizer in ACRIMA dataset obtained the highest values of 0.2406 error, 0.9722 validation accuracy, and 0.9842 *f*-score, respectively. The experimental studies have shown that satisfactory results have been obtained. In this study, the activation function with optimizer improve to performance of the CNN architecture was analyzed.

The proposed Trish activation function can outperform ReLU and other traditional activation functions in certain tasks and problems, especially in the classification of glaucoma disease. This is due to the fact that it offers an adaptive approach and can better manage the complexity of deep learning models. The ability to generate nonzero values for negative inputs helps to avoid the dead neuron problem and increases the learning capacity of the model, allowing it to achieve non-saturation. It also provides an advantage against the problem of gradient loss in deep networks by allowing gradients to flow better during model training. However, the computational cost of Trish can be higher than that of ReLU, and it needs an optimal choice of parameters to maximize its performance. Determining these parameters may require

Table 12 Performance results for glaucoma classification

References	Data	Feature	Technique	Performance
Valverde et al. [37]	2313 images	Lesion-based features	CNN	AUC of 94%
Juneja et al. [38]	Four datasets	Deep features	CNN	Accuracy: 93.5%
Balasubramanian et al. [39]	7280 images with data augmentation	Deep features	VGGNet	Overall accuracy of 99.6%
Claro et al. [40]	1675 images	Texture, morphology, and transfer learning	SVM, MLP, and RF	Accuracy: 93.6%
Jun et al. [41]	1022 fundus images	Deep features	TRk-CNN Opt	Accuracy: 92.96%
Haider et al. [42]	1060 images	Deep features	SLSR-Net	Accuracy: 99.6%
Proposed model	1106 fundus images	Deep features	CNN + Activation function with optimizer	Accuracy: 97.22%

additional experiments and tuning. Furthermore, although Trish performs well on certain tasks and datasets, there is no guarantee that it is the best choice in all cases; different problems and data structures may require different activation functions.

6 Conclusion

Network accuracies are greatly influenced by optimizers and activation functions, which are parts of complex deep learning architectures. This study examines ways that optimizers and activation functions can improve the effectiveness of CNN architectures with two different numbers of layers for classifying glaucoma. A new activation function called Trish is proposed, and various existing optimizers and activation functions are used to create different architectural variations for comparison. It was discovered that this architecture, created specifically for the classification of glaucoma and using Trish, was quite effective. This discovery is scientifically groundbreaking as it diversifies the activation function, an important component of the deep learning system.

This study introduces a new non-monotonic activation function to improve the training success of deep learning classifiers. Three different datasets, fundus (HRF), ACRIMA, and CIFAR-10, were used. For this purpose, optimization method (SGD, Adam, RmsProp, AdaDelta, AdaGrad, Adamax, and Nadam), ReLU, LReLU, Mish, Swish, Smish, Logish, and proposed activation function and three-layer and six-layer CNN from different deep learning architectures were used. Thanks to the simulation, glaucoma disease detection was carried out. Moreover, experimental evaluation was made through the CIFAR-10 dataset to further support the proposed activation function.

Finally, the effectiveness of the proposed method was compared to the performance of other activation functions with different optimizers in a variety of measures, including training loss, training accuracy, training f -score, validation loss, validation accuracy, and validation f -score. The highest performance obtained for the detection of glaucoma disease over the HRF dataset using the proposed activation function and Adamax optimizer with a six-layer CNN architecture has 0.6429 error, 0.7538 validation accuracy, and 0.7874 f -score values, respectively. The proposed activation function, using the ACRIMA dataset, has the highest performance obtained with six-layer CNN and Adamax optimizer, with 0.2406 error, 0.9722 validation accuracy, and 0.9842 f -score, respectively. The advantage of the proposed activation function with an optimizer is the efficiency is high.

Author contributions The authors contributed equally to the study.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK).

Declarations

Conflict of interest The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Nouri-Mahdavi K, Weiss RE (2021) Detection of glaucoma deterioration in the macular region with optical coherence tomography: challenges and solutions. *Am J Ophthalmol* 222:277–284. <https://doi.org/10.1016/j.ajo.2020.09.026>
2. Lee T, Jammal AA, Mariottoni EB, Medeiros FA (2021) Predicting glaucoma development with longitudinal deep learning predictions from fundus photographs. *Am J Ophthalmol* 225:86–94. <https://doi.org/10.1016/j.ajo.2020.12.031>
3. Wang SY, Pershing S, Tran E, Hernandez-Boussard T (2020) Automated extraction of ophthalmic surgery outcomes from the electronic health record. *Int J Med Inf* 133:104007. <https://doi.org/10.1016/j.ijmedinf.2019.104007>
4. Wang SY, Tseng B, Hernandez-Boussard T (2022) Deep learning approaches for predicting glaucoma progression using electronic health records and natural language processing. *Ophthalmol Sci* 2(2):100127. <https://doi.org/10.1016/j.xops.2022.100127>
5. Hashimoto Y et al (2021) Deep learning model to predict visual field in central 10° from optical coherence tomography measurement in glaucoma. *Br J Ophthalmol* 105(4):507–513. <https://doi.org/10.1136/bjophthalmol-2019-315600>
6. Iqbal I, Walayat K, Kakar MU, Ma J (2022) Automated identification of human gastrointestinal tract abnormalities based on deep convolutional neural network with endoscopic images. *Intell Syst Appl* 16:200149. <https://doi.org/10.1016/j.iswa.2022.200149>
7. Thakur A, Goldbaum M, Yousefi S (2020) Predicting glaucoma before onset using deep learning. *Ophthalmol Glaucoma* 3(4):262–268. <https://doi.org/10.1016/j.ogla.2020.04.012>
8. Chayan TI, Islam A, Rahman E, Reza MT, Apon TS, Alam MGR (2022) Explainable AI based glaucoma detection using transfer learning and LIME. Preprint <http://arxiv.org/abs/2210.03332> (Erişim 20 Şubat 2023)
9. Li F et al (2022) A deep-learning system predicts glaucoma incidence and progression using retinal photographs. *J Clin Invest* 132(11):e157968. <https://doi.org/10.1172/JCI157968>
10. Iqbal I, Younus M, Walayat K, Kakar MU, Ma J (2021) Automated multi-class classification of skin lesions through deep convolutional neural network with dermoscopic images. *Comput Med Imaging Graph* 88:101843. <https://doi.org/10.1016/j.compmedimag.2020.101843>
11. Mirzania D, Thompson AC, Muir KW (2021) Applications of deep learning in detection of glaucoma: a systematic review. *Eur J Ophthalmol* 31(4):1618–1642. <https://doi.org/10.1177/1120672120977346>
12. Hemelings R et al (2020) Accurate prediction of glaucoma from colour fundus images with a convolutional neural network that relies on active and transfer learning. *Acta Ophthalmol*. <https://doi.org/10.1111/aos.14193>
13. Oh S, Park Y, Cho KJ, Kim SJ (2021) Explainable machine learning model for glaucoma diagnosis and its interpretation. *Diagnostics* 11(3):510. <https://doi.org/10.3390/diagnostics11030510>
14. Kamal MS, Dey N, Chowdhury L, Hasan SI, Santosh K (2022) Explainable AI for glaucoma prediction analysis to understand risk factors in treatment planning. *IEEE Trans Instrum Meas* 71:1–9. <https://doi.org/10.1109/TIM.2022.3171613>
15. Huang X et al (2022) Detecting glaucoma from multi-modal data using probabilistic deep learning. *Front Med* 9:923096. <https://doi.org/10.3389/fmed.2022.923096>

16. Thompson AC, Falconi A, Sappington RM (2022) Deep learning and optical coherence tomography in glaucoma: bridging the diagnostic gap on structural imaging. *Front Ophthalmol* 2:937205. <https://doi.org/10.3389/fopht.2022.937205>
17. Girard MJA, Schmetterer L (2020) Artificial intelligence and deep learning in glaucoma: current state and future prospects. *Progress Brain Res* 257:37–64. <https://doi.org/10.1016/bs.pbr.2020.07.002>
18. Tarcoveanu F, Leon F, Curteanu S, Chiselita D, Bogdanici CM, Anton N (1831) Classification algorithms used in predicting glaucoma progression. *Healthcare* 10(10):2022. <https://doi.org/10.3390/healthcare10101831>
19. Madadi Y, Abu-Serhan H, Yousefi S (2022) Domain adaptation-based deep learning models for forecasting and diagnosis of glaucoma disease. <https://doi.org/10.36227/techrxiv.21391551.v2>
20. Nunez R et al (2022) Artificial intelligence to aid glaucoma diagnosis and monitoring: state of the art and new directions. *Photonics* 9(11):810. <https://doi.org/10.3390/photonics9110810>
21. Singh LK, Pooja, Garg H, Khanna M (2022) Deep learning system applicability for rapid glaucoma prediction from fundus images across various data sets. *Evol Syst* 13(6):807–836. <https://doi.org/10.1007/s12530-022-09426-4>
22. Akter N, Fletcher J, Perry S, Simunovic MP, Briggs N, Roy M (2022) Glaucoma diagnosis using multi-feature analysis and a deep learning technique. *Sci Rep* 12(1):8064. <https://doi.org/10.1038/s41598-022-12147-y>
23. Pham QTM, Han JC, Shin J (2022) A multimodal deep learning model for predicting future visual field in glaucoma patients. In: Review. <https://doi.org/10.21203/rs.3.rs-1236761/v1>
24. Diaz-Pinto A, Morales S, Naranjo V, Köhler T, Mossi JM, Navea A (2019) CNNs for automatic glaucoma assessment using fundus images: an extensive validation. *Biomed Eng Online* 18(1):29. <https://doi.org/10.1186/s12938-019-0649-y>
25. Budai A, Bock R, Maier A, Horneegger J, Michelson G (2013) Robust vessel segmentation in fundus images. *Int J Biomed Imaging* 2013:e154860. <https://doi.org/10.1155/2013/154860>
26. Akbar S, Hassan SA, Shoukat A, Alyami J, Bahaj SA (2022) Detection of microscopic glaucoma through fundus images using deep transfer learning approach. *Microsc Res Tech* 85(6):2259–2276. <https://doi.org/10.1002/jemt.24083>
27. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):7553. <https://doi.org/10.1038/nature14539>
28. Kiliçarslan S, Celik M (2021) RSigELU: a nonlinear activation function for deep neural networks. *Expert Syst Appl* 174:114805. <https://doi.org/10.1016/j.eswa.2021.114805>
29. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324. <https://doi.org/10.1109/5.726791>
30. Iqbal I, Odesanmi GA, Wang J, Liu L (2021) Comparative investigation of learning algorithms for image classification with small dataset. *Appl Artif Intell* 35(10):697–716. <https://doi.org/10.1080/08839514.2021.1922841>
31. Kiliçarslan S, Közkurt C, Baş S, Elen A (2023) Detection and classification of pneumonia using novel superior exponential (SupEx) activation function in convolutional neural networks. *Expert Syst Appl* 217:119503. <https://doi.org/10.1016/j.eswa.2023.119503>
32. Ramachandran P, Zoph B, Le QV (2017) Searching for activation functions. <https://doi.org/10.48550/arXiv.1710.05941>
33. Misra D (2020) Mish: a self regularized non-monotonic activation function. Preprint <http://arxiv.org/abs/1908.08681>
34. Zhu H, Zeng H, Liu J, Zhang X (2021) Logish: a new nonlinear nonmonotonic activation function for convolutional neural network. *Neurocomputing* 458:490–499. <https://doi.org/10.1016/j.neucom.2021.06.067>
35. Wang X, Ren H, Wang A (2022) Smish: a novel activation function for deep learning methods. *Electronics*. <https://doi.org/10.3390/electronics11040540>
36. Weinreb RN, Aung T, Medeiros FA (2014) The pathophysiology and treatment of glaucoma: a review. *Clin Rev Educ Am Med Assoc*
37. Gómez-Valverde JJ, Antón A, Fatti G, Liefers B, Herranz A, Santos A, Sánchez CI, Ledesma-Carbayo MJ (2019) Automatic glaucoma classification using color fundus images based on convolutional neural networks and transfer learning. *Biomed Opt Express* 10
38. Juneja M, Thakur S, Uniyal A, Wani A, Thakur N, Jindal P (2022) Deep learning-based classification network for glaucoma in retinal images. *Comput Electr Eng* 101

39. Balasubramanian K, Ramya K, Devi KG (2022) Improved swarm optimization of deep features for glaucoma classification using SEGSO and VGGNet. *Biomed Signal Process Control* 77
40. Claro M, Veras R, Santana A, Araujo F, Silva R, Almeida J, Leite D (2019) An hybrid feature space from texture information and transfer learning for glaucoma classification. *J Vis Commun Image R* 64
41. Jun TJ, Eom Y, Kim D, Kim C, Park JH, Nguyen HM, Kim YH, Kim D (2021) TRk-CNN: transferable ranking-CNN for image classification of glaucoma, glaucoma suspect, and normal eyes. *Expert Syst Appl* 182:115211
42. Haider A, Arsalan M, Lee MB, Owais M, Mahmood T, Sultan H, Park KR (2022) Artificial Intelligence-based computer-aided diagnosis of glaucoma using retinal fundus images. *Expert Syst Appl* 207

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.