

# Adjustment of Convolutional and Hidden Layers Using Type-1 Fuzzy Logic Applied to Diabetic Retinopathy Classification

Rodrigo Cordero-Martínez, Daniela Sánchez\*, Patricia Melin

Tecnológico Nacional de México, Campus Tijuana,  
Mexico

{rodrigo.cordero201, daniela.sanchez, pmelin}@tectijuana.mx

**Abstract.** Vision problems are common in patients with diabetes mellitus (DM) because they may suffer from diabetic retinopathy (DR). Because the symptoms of this condition are not easy to detect without the intervention of an expert technician, the use of convolutional neural networks (CNN) has been implemented to speed up the process of analyzing retina images. Due to the good results of this technology, efforts have been made to combine it with other technologies. In this paper, we present the use of an intelligent hybrid system that uses CNNs and Fuzzy Logic with the aim of improving the accuracy obtained. The implementation of fuzzy logic to adjust the hyperparameters of the network allowed us to obtain a mean of 0.9526 with a standard deviation of 0.008521158 in the binary case study, while in the multiclass case study we obtained a mean of 0.7299 and a standard deviation of 0.015614013, offering better results when fuzzy logic is combined compared to when not.

**Keywords.** Convolutional neural network, fuzzy logic, image pre-processing.

## 1 Introduction

Individuals afflicted with diabetes mellitus (DM), irrespective of whether it manifests as type 1 or type 2, face a significantly elevated likelihood of developing diabetic retinopathy (DR) [1-2]. Those grappling with this condition may experience visual impairments or, in severe cases, complete blindness [3-4]. A prognostic study indicates that within a mere decade, the prevalence of this ailment is anticipated to double compared to patient counts from a decade prior [5].

Consequently, the incidence of vision impairment is poised to witness a twofold increase. Several factors contribute to the expeditious proliferation of DR cases [6-7]. The foremost factor

is the onset age of this condition, which has been documented in DM patients as young as 20 years old [8].

Another critical concern in the context of DR is the dearth of expert technicians that are adept at diagnosing the ailment [9-10]. In response to this challenge, diverse computer technologies have been harnessed to facilitate prompt detection and informed medical decision-making for patients.

Researchers have leveraged advanced tools like deep learning, particularly neural networks, to expedite the detection process not only for DR but also for other diseases, concurrently diminishing the margin of error [11-14].

Techniques such as convolutional neural networks (CNNs) and deep learning methodologies have proven highly efficacious in DR detection, with optimal outcomes realized through their application [15-17]. Many researchers have expedited their work by utilizing pre-trained CNNs, yet bespoke architectures tailored to specific issues have exhibited enhanced detection accuracy, incorporating techniques such as optimization algorithms and fuzzy logic [18].

This study commenced by meticulously selecting the foundational CNN model and the most appropriate pre-processing technique for the designated database. Exhaustive research ensured the identification of the pre-processing method that exhibited superior performance in prior studies.

Similarly, the base CNN model was chosen based on the favorable outcomes discerned in the research. The subsequent phase involved the integration of a fuzzy inference system to optimize the layers of the CNN architecture for DR detection. Despite the prevalent use of fuzzy logic

**Table 1.** Multiple image distribution of APTOS 2019

Distribution Name	Number of images
Retina without DR	1805
No proliferative mild	370
No proliferative moderate	999
No proliferative severe	193
Proliferative	295
Retina with any proliferative stage of DR	1857

in other studies for classifying diverse databases, its application in the detection and classification of DR remains relatively unexplored.

Section 2 of this paper elucidates the related work conducted by various authors. Section 3 provides comprehensive explanations of fundamental concepts, complemented by illustrative examples, to enhance comprehension of this study.

Section 4 delineates and expounds upon the methods employed in this research. Section 5 presents the results gleaned from the conducted experiments, and finally, Section 6 articulates the conclusions derived from the study.

## 2 Related Work

Increasing the precision of neural networks either for classification or prediction has been a priority for experts in recent years [19-20], due to this many works have been written with different implementations or own methods that seek the same objective.

In the work of [21] we sought to modify the depth of the network, size and number of convolutional filters and number of neurons in the hidden layers. In this work it was concluded that, depending on the data set used, the number of filters used in the convolution layer can be increased or decreased, so increasing the number of filters will not necessarily improve the results.

Do not forget that a convolutional neural network contains the same hidden layers as a traditional neural network. The work of the author [22] shows us the different behaviors that

modifying the number of neurons in each of the hidden layers can result in an increase of precision.

To carry out the modifications to the hyperparameters of the CNNs, the use of optimization algorithms has been implemented that allow experts to speed up the process, one of them being the genetic algorithm. Thanks to this technology, prior work can be carried out with the APTOS 2019 [23] database. Using this algorithm, a CNN model was created that improves the accuracy of diabetic retinopathy classification.

## 3 Basic Concepts

In the previous section, some terms were mentioned that may not be familiar to those who do not work with the use of intelligent hybrid systems. So, for a complete understanding of this work, this section will present the information necessary to understand the work in its entirety.

### 3.1 Artificial Neural Networks

One of the most widely employed machine learning tools for disease detection is the supervised artificial neural network. This specific neural network variant enables expert technicians to train the network using a labeled database [24].

When utilizing images in the learning phase, it becomes imperative to specify pertinent information for accurate future classification. This underscores the necessity for the supervision of an expert technician in the realm of image topics.

However, there exists a type of neural network capable of performing this task with convolutional filters: convolutional neural networks [25]. The design of a convolutional neural network model closely resembles that of a feed-forward neural network, with the differentiating factor becoming evident after the input stage [26].

#### 3.1.1. Convolutional Layer

It is the initial layer in the CNN architecture, facilitates the recognition of key characteristics within the input images [27].

Consequently, the network eliminates the requirement for an expert technician to apply preprocessing methods to the images. To achieve

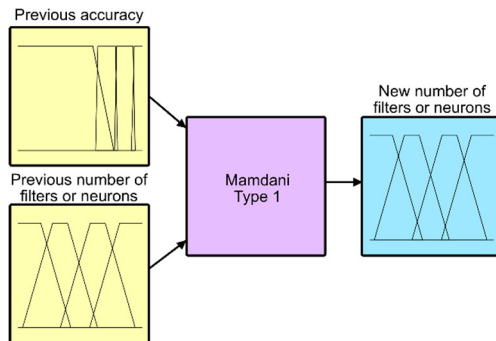


Fig. 1. Graphical representation of FIS

Table 2. Ranges of the new filters number

Convolutional Layer Number	Range
1	[16-32]
2	[32-64]
3	[64-128]
4	[128-256]
5	[256-512]

Table 3. Ranges of the new neurons number

Fully Connected Layer Number	Range
1 (Binary Study Case)	[64 - 128]
2 (Binary Study Case)	[128 - 256]
3 (Binary Study Case)	[256 - 512]
1 (Multiclass Study Case)	[64 - 512]

this, the convolutional layer necessitates a kernel to derive a new matrix.

### 3.1.2. ReLU Function

The Rectified Linear Unit Function (ReLU) plays a crucial role in Convolutional Neural Networks (CNNs). Activation functions are essential after each individual neuron, and one of the widely utilized functions for CNNs is the ReLU function [28].

ReLU permits the activation of every positive number, thereby reducing the time required for experimentation.

### 3.1.3. MaxPooling Layer

The activation function serves to expedite the training time, although the image size remains constant, and not all pixels carry equal significance. To address this, a pooling method is employed, with the MaxPooling method being one of the most prevalent in the domain of convolutional neural networks [29].

### 3.2 Fuzzy Logic

Fuzzy logic is a logic that allows you to reach “reasoned” conclusions based on ambiguous or imprecise information [30]. One of the great contributions of this logic is that it allows us to model situations or behaviors that are vague in themselves, that is, it adapts better to reality than classic logic, where there are only two values to decide [31].

Human reasoning does not react in a classical logic manner, but on the contrary, evaluates the environment and based on the weights of each of the variables decides, so fuzzy logic is more suitable to try to emulate such mental behavior [32].

It has been used for the development of a countless number of applications of all kinds such as medicine and bioinformatics [33]. The primary emphasis of this study lies in hybrid systems, signifying the necessity of incorporating two or more distinct techniques to formulate the proposed method.

The utilization of both CNN and fuzzy logic has been a recurrent theme in previous works, with a noticeable increase in its prevalence over the years [34]. The amalgamation of convolutional neural networks and fuzzy logic typically involves the incorporation of optimization algorithms, such as genetic algorithms or particle swarm optimization, aimed at optimizing the parameters associated with each technology [35].

## 4 Proposed Methods

In this section, we will consider each of the concepts explained in the previous section. First, the architectures of the CNN models from a previous work to which the proposed method will

**Table 4.** Fuzzy rules

Fuzzy Rule Number	Fuzzy Rule
1	If (accuracy is very_bad) and (old_filters is very_few) then (new_filters is a_lot)
2	If (accuracy is bad) and (old_filters is very_few) then (new_filters is many)
3	If (accuracy is good) and (old_filters is very_few) then (new_filters is few)
4	If (accuracy is excellent) and (old_filters is very_few) then (new_filters is very_few)
5	If (accuracy is very_bad) and (old_filters is few) then (new_filters is a_lot)
6	If (accuracy is bad) and (old_filters is few) then (new_filters is many)
7	If (accuracy is good) and (old_filters is few) then (new_filters is very_few)
8	If (accuracy is excellent) and (old_filters is few) then (new_filters is few)
9	If (accuracy is very_bad) and (old_filters is many) then (new_filters is very_few)
10	If (accuracy is bad) and (old_filters is many) then (new_filters is few)
11	If (accuracy is good) and (old_filters is many) then (new_filters is a_lot)
12	If (accuracy is excellent) and (old_filters is many) then (new_filters is many)
13	If (accuracy is very_bad) and (old_filters is a_lot) then (new_filters is very_few)
14	If (accuracy is bad) and (old_filters is a_lot) then (new_filters is few)
15	If (accuracy is good) and (old_filters is a_lot) then (new_filters is many)
16	If (accuracy is excellent) and (old_filters is a_lot) then (new_filters is a_lot)

be applied will be detailed. Afterwards, the data from the APTOS 2019 database, characteristics and the two distributions necessary for the case studies of this work will be presented.

Then the explanation of the preprocessing applied to the database will be gone into detail. And finally, the creation and implementation of a fuzzy inference system that will allow us to modify the number of filters and neurons in the networks.

#### 4.1 Neural Network Models

There are two CNN models obtained on previous work [36]. The first model, focused on binary study case, has an input size of 256x256x3 (width, height, and depth) and 5 convolutional layers, some on them have MaxPooling layer after it, but not all. Also, the model has 3 fully connected layers with different number of neurons.

Finally, the model has a Sigmoid activation function because it is for a binary study case. The second model, focused on multiclass study case, has the same input size and 5 convolutional layers with different values on its hyperparameters (MaxPooling size dropout if applies). Also, the

model has 1 fully connected layer. Finally, the model has a Softmax activation function because it is for multiclass study case.

#### 4.2 APTOS 2019 Database

This DR database has 3662 labeled images can be used for training and validation [37]. The database has 5 different classes that represent the damage caused by the disease that are used for multiclass study case [38], but 4 classes can be combined to obtain just 2 total classes for binary study case: healthy retina images and retina with diabetic retinopathy images [39].

The database has images with different kind of noise and no image has the same size. In Table 1, the distribution of the images for this work can be observed.

#### 4.3 Preprocessing Method

This approach involves removing interfering pixels from the background and completely isolating the retina in the image. To achieve this goal, it is essential to transform color images to grayscale.

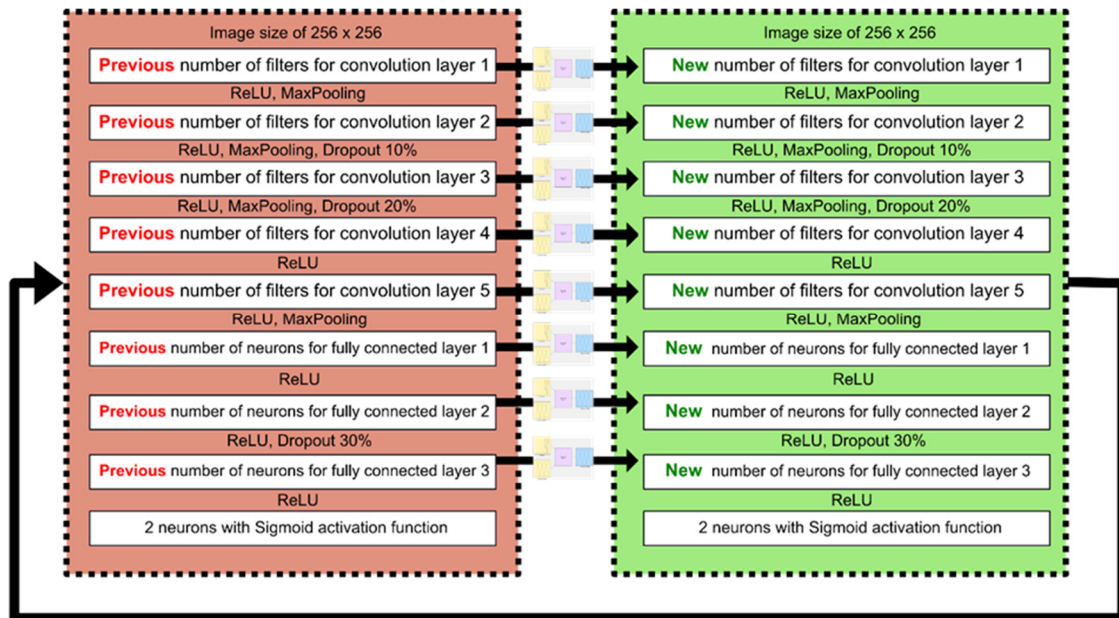


Fig. 2. Graphical representation of the proposed method

By using a grayscale representation, conversion to a binary image becomes feasible. The process of converting the grayscale image to binary involves the selection of each pixel to refine the image.

The amount of luminosity present in the pixel must exceed a defined threshold to avoid considering it as noise and instead take advantage of it for precise retinal extraction. With the resulting binary image, the next step consists of locating and identifying the retina.

Thanks to the binary image, this task is simplified since it only involves identifying the most prominent shape. Once the retina has been detected, its position is extracted and used to isolate the retina from the original images.

Finally, black pixels are inserted as necessary to achieve an image with uniform dimensions in width and height. This method has yielded favorable results compared to other preprocessing techniques [40].

#### 4.4 Fuzzy Inference System Description

To start, both APTOS 2019 study cases require a Mamdani Type-1 fuzzy inference system (FIS). All membership functions are trapezoidal functions.

The FIS comprises two inputs: the accuracy achieved with the current quantity of filters or neurons, and the second input pertains to the current quantity of filters or neurons.

The FIS yields a single output, which signifies the number of filters for the convolutional layer or neurons for the fully connected layer. Each input and output have 4 membership functions. The graphical representation of the FIS is depicted in Fig. 1. The accuracy value is normalized within the range of 0 to 1.

The quantity of filters or neurons is contingent upon the convolutional layer number or fully connected layer, with specified ranges detailed in Table 2 and Table 3 respectively.

The chosen Defuzzification Method is Centroid. In the context of these experiments, the FIS incorporates 16 fuzzy if-then rules obtained by trial and error, which are outlined in Table 4. A graphical representation of the proposed method can be observed on Fig. 2.

Equations of the FIS can be observed on Eq. 1-8 where Eq. 1-4 are for the input of accuracy and Eq. 5-8 are for the input and output of the number of filters or neurons.

$$\mu_{\text{very\_bad}}(x) = \begin{cases} 0, & x \leq -0.2 \\ \frac{x - (-0.2)}{-0.1 - (-0.2)}, & -0.2 \leq x \leq -0.1 \\ 1, & -0.1 \leq x \leq 0.7 \\ \frac{0.8 - x}{0.8 - 0.7}, & 0.7 \leq x \leq 0.8 \\ 0, & 0.8 \leq x \end{cases}, \quad (1)$$

$$\mu_{\text{bad}}(x) = \begin{cases} 0, & x \leq 0.65 \\ \frac{x-0.65}{0.72-0.65}, & 0.65 \leq x \leq 0.72 \\ 1, & 0.72 \leq x \leq 0.83 \\ \frac{0.9-x}{0.9-0.83}, & 0.83 \leq x \leq 0.9 \\ 0, & 0.9 \leq x \end{cases}, \quad (2)$$

$$\mu_{\text{good}}(x) = \begin{cases} 0, & x \leq 0.8 \\ \frac{x - 0.8}{0.85 - 0.8}, & 0.8 \leq x \leq 0.85 \\ 1, & 0.85 \leq x \leq 0.95 \\ \frac{1 - x}{1 - 0.95}, & 0.9 \leq x \leq 1 \\ 0, & 1 \leq x \end{cases}, \quad (3)$$

$$\mu_{\text{excellent}}(x) = \begin{cases} 0, & x \leq 0.9 \\ \frac{x - 0.9}{0.96 - 0.9}, & 0.9 \leq x \leq 0.96 \\ 1, & 0.96 \leq x \leq 1.042 \\ \frac{1.375 - x}{1.375 - 1.042}, & 1.042 \leq x \leq 1.375 \\ 0, & 1.375 \leq x \end{cases}, \quad (4)$$

$$\mu_{\text{very\_few}}(x) = \begin{cases} 0, & x \leq -0.375 \\ \frac{x - (-0.375)}{-0.04 - (-0.375)}, & -0.375 \leq x \leq -0.04 \\ 1, & -0.04 \leq x \leq 0.1 \\ \frac{0.4 - x}{0.4 - 0.1}, & 0.1 \leq x \leq 0.4 \\ 0, & 0.4 \leq x \end{cases} \quad (5)$$

$$\mu_{\text{few}}(x) = \begin{cases} 0, & x \leq -0.005 \\ \frac{x - (-0.005)}{0.3 - (-0.005)}, & -0.005 \leq x \leq 0.3 \\ 1, & 0.3 \leq x \leq 0.4 \\ \frac{0.7 - x}{0.7 - 0.4}, & 0.4 \leq x \leq 0.7 \\ 0, & 0.7 \leq x \end{cases} \quad (6)$$

$$\mu_{\text{many}}(x) = \begin{cases} 0, & x \leq 0.3 \\ \frac{x - 0.3}{0.6 - 0.3}, & 0.3 \leq x \leq 0.6 \\ 1, & 0.6 \leq x \leq 0.7 \\ \frac{1 - x}{1 - 0.7}, & 0.7 \leq x \leq 1 \\ 0, & 1 \leq x \end{cases} \quad (7)$$

$$\mu_{\text{a\_lot}}(x) = \begin{cases} 0, & x \leq 0.6 \\ \frac{x - 0.6}{0.9 - 0.6}, & 0.6 \leq x \leq 0.9 \\ 1, & 0.9 \leq x \leq 1.015 \\ \frac{1.315 - x}{1.315 - 1.015}, & 1.015 \leq x \leq 1.315 \\ 0, & 1.315 \leq x \end{cases} \quad (8)$$

## 5 Experimental Results

In this section, we are going to bring together each of the concepts and methods proposed for distributed experimentation in two study cases: binary and multiclass.

### 5.1 Experiments for APTOS 2019 Binary Study Case

Two experiments were conducted; the initial one involved employing the CNN model derived through the hierarchical genetic algorithm as documented in prior research [41]. The second experiment utilized the CNN model acquired through the previously explained FIS.

Each experiment was iterated 30 times, maintaining consistent hyperparameters: 10 epochs, utilization of the APTOS 2019 database, and the Adam optimizer algorithm.

In the first experiment, the mean accuracy recorded was 0.9021, accompanied by a standard deviation of 0.108434797. Conversely, for the second experiment, the mean accuracy achieved was 0.9526, with a standard deviation of 0.008521158. Detailed results for each iteration are presented in Table 5.

#### 5.1.1. Box Plot for Binary Study Case

One box plot was made to observe the comparison of the values. Box plot for the binary study case can be observed on Fig. 3.

#### 5.1.2. Hypothesis Testing for Binary Study Case

Based on the results observed in Table 5, the hypothesis testing will be between mean accuracy and standard deviation obtained by the first experiment and the second one. The experiment of this present work got a higher mean accuracy, so, our statement is that the experiment with the CNN

**Table 5.** Results for binary study case experimentation

Experiment Number	Genetic Algorithm Accuracy	Fuzzy Logic Accuracy	Experiment Number	Genetic Algorithm Accuracy	Fuzzy Logic Accuracy
1	0.938608468	0.934515715	16	0.927694380	0.960436583
2	0.929058671	0.956343770	17	0.889495254	0.938608468
3	0.911323309	0.960436583	18	0.931787193	0.954979539
4	0.507503390	0.952251017	19	0.950886786	0.949522495
5	0.920873106	0.956343770	20	0.960436583	0.956343770
6	0.930422902	0.961800814	21	0.953615308	0.952251017
7	0.939972699	0.961800814	22	0.897680759	0.972714841
8	0.937244177	0.942701221	23	0.931787193	0.949522495
9	0.924965918	0.957708061	24	0.904502034	0.954979539
10	0.938608468	0.956343770	25	0.934515715	0.935879946
11	0.937244177	0.960436583	26	0.916780353	0.950886786
12	0.937244177	0.960436583	27	0.929058671	0.946793973
13	0.939972699	0.952251017	28	0.934515715	0.945429742
14	0.946793973	0.952251017	29	0.507503390	0.954979539
15	0.930422902	0.939972699	30	0.933151424	0.948158264

**Table 6.** Results for multiclass study case experimentation

Experiment Number	Genetic Algorithm Accuracy	Fuzzy Logic Accuracy	Experiment Number	Genetic Algorithm Accuracy	Fuzzy Logic Accuracy
1	0.728512943	0.717598915	16	0.688949525	0.740791261
2	0.706684828	0.738062739	17	0.703956366	0.720327437
3	0.714870393	0.733969986	18	0.731241465	0.736698508
4	0.712141871	0.712141871	19	0.710777640	0.744884014
5	0.710777640	0.731241465	20	0.703956366	0.727148712
6	0.714870393	0.733969986	21	0.735334218	0.713506162
7	0.727148712	0.723055959	22	0.728512943	0.720327437
8	0.736698508	0.743519783	23	0.739427030	0.731241465
9	0.740791261	0.739427030	24	0.739427030	0.729877234
10	0.750341058	0.729877234	25	0.718963146	0.725784421
11	0.724420190	0.742155552	26	0.690313756	0.727148712
12	0.725784421	0.710777640	27	0.712141871	0.727148712
13	0.727148712	0.724420190	28	0.706684828	0.721691668
14	0.688949525	0.725784421	29	0.720327437	0.735334218
15	0.716234624	0.736698508	30	0.717598915	0.753069580

model obtained by the fuzzy system inference offers a bigger mean accuracy than the offered by the experiment with the CNN model obtained by the hierarchical genetic algorithm for binary study case. Using an Alpha value of 0.05, the critical

value obtained must be more than 1.96 to reject the null hypothesis. The score of the statistic test is 2.5240, meaning that the null hypothesis is rejected and there is enough evidence to support the claim.

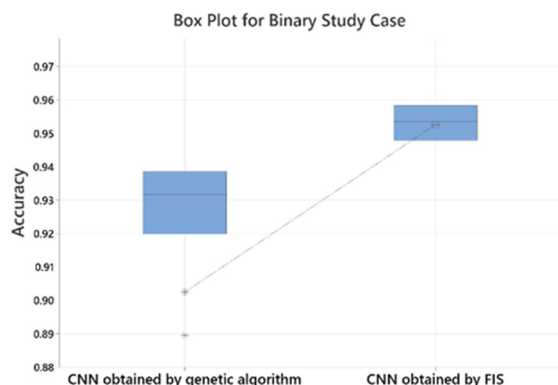


Fig. 3. Box plot for binary study case

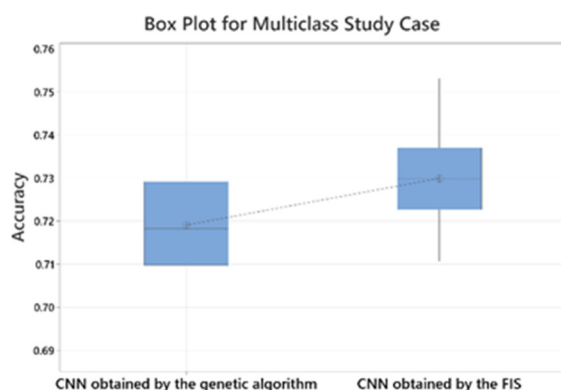


Fig. 4. Box plot for multiclass study case

## 5.2 Experiments for APTOS 2019 Multiclass Study

In the same way as previous experimentation, two experiments were conducted; the first one involved employing the CNN model obtained through the hierarchical genetic algorithm [42] and the second experiment utilized the CNN model acquired through the FIS. Each experiment was iterated 30 times, maintaining consistent hyperparameters: 10 epochs, utilization of the APTOS 2019 database, and the Adam optimizer algorithm.

In the first experiment, the mean accuracy recorded was 0.7191, accompanied by a standard deviation of 0.010199619. Conversely, for the second experiment, the mean accuracy achieved was 0.7299, with a standard deviation of 0.015614013. Detailed results for each iteration are presented in Table 6.

### 5.2.1. Box Plot for Multiclass Study Case

In the same way as the previous experiment, one box plot was made to observe the comparison of the values. Box plot for the multiclass study case can be observed on Fig. 4.

### 5.1.2. Hypothesis Testing for Multiclass Study Case

Based on the results observed in Table 6, the hypothesis testing will be between mean accuracy and standard deviation obtained by the first experiment and the second one.

The experiment of this present work got a higher mean accuracy, so, our statement is that the experiment with the CNN model obtained by the fuzzy system inference offers a bigger mean accuracy than the offered by the experiment with the CNN model obtained by the hierarchical genetic algorithm for multiclass study case.

Using an Alpha value of 0.05, the critical value obtained must be more than 1.96 to reject the null hypothesis. The score of the statistic test is 3.1786, meaning that the null hypothesis is rejected and there is enough evidence to support the claim.

## 6 Conclusions

In this study, the focus was on employing a Mamdani Type 1 fuzzy inference system to determine the filter numbers based on the previous filter values and the obtained accuracy. Before implementing the proposed method, the mean accuracy and standard deviation of the base CNN model were calculated for comparative analysis. Subsequently, the proposed method was integrated into a pre-existing CNN model.

After the generation of the new CNN model, the FIS was used iteratively to refine the CNN model, obtaining the best CNN model, mean precision, and standard deviation. There is room for improvement in the construction of the FIS such as the number of variables, rules and membership functions, so as future work the current work can be taken and implemented the respective improvements and seek a higher average precision with a reduced standard deviation.

Finally, APTOS 2019 serves as a valuable database in addressing real-world problems, yet it is not the exclusive dataset where the proposed



method could find application. As future work, we plan to consider different metaheuristics for optimizing the method, as in [43-48]. Also, elevate the use of fuzzy logic to type-2, like it is done in several recent works [49-54].

## Acknowledgments

We would like to express our gratitude to the CONAHCYT and Tijuana Institute of Technology for the facilities and resources granted for the development of this research.

## References

1. **Alex, S. A., Nayahi, J. J. V., Shine, H., Gopirekha, V. (2021).** Deep convolutional neural network for diabetes mellitus prediction. *Neural Computing and Applications*, Vol. 34, No. 2, pp. 1319–1327. DOI: 10.1007/s00521-021-06431-7.
2. **Milluzzo, A., Maugeri, A., Barchitta, M., Sciacca, L., Agodi, A. (2021).** Epigenetic mechanisms in type 2 diabetes retinopathy: a systematic review. *International Journal of Molecular Sciences*, Vol. 22, No. 19, pp.10502. DOI: 10.3390/ijms221910502.
3. **Singh, S., Rathore, B., Das, H., Agrawal, S., Bhutia, D., Maan, V., Jangir, S. K., Kumar, M. (2022).** An improved convolutional neural network for classification of type-2 diabetes mellitus. *Algorithms for Intelligent Systems*, pp.417–424. DOI: 10.1007/978-981-19-2065-3\_45.
4. **Flaxel, C. J., Adelman, R. A., Bailey, S. T., Fawzi, A., Lim, J. I., Vemulakonda, G. A., Ying, G. (2020).** Diabetic retinopathy preferred practice pattern®. *Ophthalmology*, Vol. 127, No. 1, pp. 66–145. DOI: 10.1016/j.ophtha.2019.09.025.
5. **Rocha, D. A. D., Ferreira, F. M. F., Peixoto, Z. M. A. (2022).** Diabetic retinopathy classification using VGG16 neural network. *Research on Biomedical Engineering*, Vol. 38, No. 2, pp. 761–772. DOI: 10.1007/s42600-022-00200-8.
6. **Correia, J. L., Moreira-Bessa, W. (2022).** Intelligent control with artificial neural networks for automated insulin delivery systems. *Bioengineering*, Vol. 9, No. 11, pp. 664. DOI: 10.3390/bioengineering9110664.
7. **Ranjan, R., Gupta, S., Sharma, S., Alatba, S. R., Sreeram, A., Fernandez, S. (2023).** Artificial neural network based categorization of diabetes mellitus and its comparison with k-nearest neighbors algorithm. *Proceedings of the 3rd International Conference on Advance Computing and Innovative Technologies in Engineering*, pp. 1099–1101 DOI: 10.1109/icacite57410.2023.10182961.
8. **Prabhu, P., Selvabharathi, S. (2019).** Deep belief neural network model for prediction of diabetes mellitus. *3rd International Conference on Imaging, Signal Processing and Communication*, pp. 138–142 DOI: 10.1109/icispc.2019.8935838.
9. **Fung, T. H., Patel, B., Wilmot, E. G., Amoaku, W. M. (2022).** Diabetic retinopathy for the non-ophthalmologist. *Clinical Medicine*, Vol. 22, No. 2, pp. 112–116. DOI: 10.78 61/clinmed.2021-0792.
10. **Hatmal, M. M., Abderrahman, S. M., Nimer, W., Al-Eisawi, Z., Al-Ameer, H. J., Al-Hatamleh, M. A. I., Mohamud, R., Alshaer, W. (2020).** Artificial neural networks model for predicting type 2 diabetes mellitus based on VDR gene FoKI polymorphism, lipid profile and demographic data. *Biology*, Vol. 9, No. 8, pp. 222. DOI: 10.3390/biology9080222.
11. **Hosseinzadeh, M., Ahmed, O. H., Ghafour, M. Y., Safara, F., Hama, H. K., Ali, S., Vo, B., Chiang, H. (2020).** A multiple multilayer perceptron neural network with an adaptive learning algorithm for thyroid disease diagnosis in the internet of medical things. *The Journal of Supercomputing*, Vol. 77, No. 4, pp. 3616–3637. DOI: 10.1007/s11227-020-03404-w.
12. **Janghel, R., Rathore, Y. (2021).** Deep convolution neural network based system for early diagnosis of alzheimer's disease. *IRBM*, Vol. 42, No. 4, pp. 258–267. DOI: 10.1016/j.irbm.2020.06.006.
13. **Irmak, E. (2020).** A novel deep convolutional neural network model for covid-19 disease

- detection. Medical Technologies Congress, pp. 1–4 DOI: 10.1109/tiptekno50054.2020.9299286.
14. **Alharbi, A., Alghahtani, M. (2018).** Using genetic algorithm and ELM neural networks for feature extraction and classification of type 2-diabetes mellitus. *Applied Artificial Intelligence*, Vol. 33, No. 4, pp. 311–328. DOI: 10.1080/08839514.2018.1560545.
  15. **Kandel, I., Castelli, M. (2020).** Transfer learning with convolutional neural networks for diabetic retinopathy image classification. a review. *Applied Sciences*, Vol. 10, No. 6, pp. 2021. DOI: 10.3390/app10062021.
  16. **Saeed, F., Hussain, M., Aboalsamh, H. A. (2021).** Automatic diabetic retinopathy diagnosis using adaptive fine-tuned convolutional neural network. *IEEE Access*, Vol. 9, pp. 41344–41359. DOI: 10.1109/access.2021.3065273.
  17. **Wen, D., Li, P., Zhou, Y., Sun, Y., Xu, J., Liu, Y., Li, X., Li, J., Bian, Z., Wang, L. (2020).** Feature classification method of resting-state EEG signals from amnesic mild cognitive impairment with type 2 diabetes mellitus based on multi-view convolutional neural network. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 28, No. 8, pp. 1702–1709. DOI: 10.1109/tnsre.2020.3004462.
  18. **Abdolrasol, M. G. M., Hussain, S. M. S., Ustun, T. S., Sarker, M. R., Hannan, M. A., Mohamed, R., Ali, J. A., Mekhilef, S., Milad, A. (2021).** Artificial neural networks based optimization techniques: a review. *Electronics*, Vol. 10, No. 21, pp. 2689. DOI: 10.3390/electronics10212689.
  19. **Baiioletti, M., Bari, G. D., Milani, A., Poggioni, V. (2020).** Differential evolution for neural networks optimization. *Mathematics*, Vol. 8, No. 1, pp. 69. DOI: 10.3390/math8010069.
  20. **Sinha, T., Verma, B., Haidar, A. (2017).** Optimization of convolutional neural network parameters for image classification. *IEEE Symposium Series on Computational Intelligence*, pp. 1–7 DOI: 10.1109/ssci.2017.8285338.
  21. **Uzair, M., Jamil, N. (2020).** Effects of hidden layers on the efficiency of neural networks. *IEEE 23rd International Multitopic Conference*, pp. 1–6 DOI: 10.1109/inmic50486.2020.9318195.
  22. **Melin, P., Sánchez, D., Cordero-Martínez, R. (2023).** Particle swarm optimization of convolutional neural networks for diabetic retinopathy classification. *Studies in Computational Intelligence*, pp. 237–252. DOI: 10.1007/978-3-031-22042-5\_14.
  23. **Singh, P., Singh, P., Farooq, U., Khurana, S. S., Verma, J. K., Kumar, M. (2023).** Cottonleafnet: cotton plant leaf disease detection using deep neural networks. *Multimedia Tools and Applications*, Vol. 82, No. 24, pp. 37151–37176. DOI: 10.1007/s11042-023-14954-5.
  24. **Bala, D., Hossain, M. S., Hossain, M. A., Abdullah, M. I., Rahman, M. M., Manavalan, B., Gu, N., Islam, M. S., Huang, Z. (2023).** Monkeynet: A robust deep convolutional neural network for monkeypox disease detection and classification. *Neural Networks*, Vol. 161, pp. 757–775. DOI: 10.1016/j.neunet.2023.02.022.
  25. **Gadekallu, T. R., Khare, N., Bhattacharya, S., Singh, S., Maddikunta, P. K. R., Srivastava, G. (2020).** Deep neural networks to predict diabetic retinopathy. *Journal of Ambient Intelligence and Humanized Computing*, Vol. 14, No. 5, pp. 5407–5420. DOI: 10.1007/s12652-020-01963-7.
  26. **Jiwani, N., Gupta, K., Afreen, N. (2022).** A convolutional neural network approach for diabetic retinopathy classification. *IEEE 11th International Conference on Communication Systems and Network Technologies*, pp. 357–361. DOI: 10.1109/csnt54456.2022.9787577.
  27. **Chen, C., Min, F., Zhang, Y., Bao, H. (2023).** Relu-type Hopfield neural network with analog hardware implementation. *Chaos, Solitons and Fractals*, Vol. 167, pp. 113068. DOI: 10.1016/j.chaos.2022.113068.
  28. **Zhang, J., Zhang, Z., Wang, L., Zhou, L., Zhang, X., Liu, M., Wu, W. (2023).** Kernel-based feature aggregation framework in point cloud networks. *Pattern Recognition*, Vol. 139,

- pp. 109439. DOI: 10.1016/j.patcog.2023.109439.
29. **Kacimi, M. A., Guenounou, O., Brikh, L., Yahiaoui, F., Hadid, N. (2020).** New mixed-coding PSO algorithm for a self-adaptive and automatic learning of Mamdani fuzzy rules. *Engineering Applications of Artificial Intelligence*, Vol. 89, pp. 103417. DOI: 10.1016/j.engappai.2019.103417.
  30. **Kaur, J., Khehra, B. S. (2021).** Fuzzy logic and hybrid based approaches for the risk of heart disease detection: State-of-the-art review. *Journal of The Institution of Engineers (India): Series B*, Vol. 103, No. 2, pp. 681–697. DOI: 10.1007/s40031-021-00644-z.
  31. **Ouifak, H., Idri, A. (2023).** On the performance and interpretability of Mamdani and Takagi-Sugeno-Kang based neuro-fuzzy systems for medical diagnosis. *Scientific African*, Vol. 20, pp. e01610. DOI: 10.1016/j.sciaf.2023.e01610.
  32. **Kamel, A. A., Zahran, F. (2020).** A fuzzy decision support system for diagnosis of some liver diseases in educational medical institutions. *International Journal of Fuzzy Logic and Intelligent Systems*, Vol. 20, No. 4, pp. 358–368. DOI: 10.5391/ijfis.2020.20.4.358.
  33. **Deng, Y., Ren, Z., Kong, Y., Bao, F., Dai, Q. (2017).** A hierarchical fused fuzzy deep neural network for data classification. *IEEE Transactions on Fuzzy Systems*, Vol. 25, No. 4, pp. 1006–1012. DOI: 10.1109/tfuzz.2016.2574915.
  34. **Melin, P., Sánchez, D., Monica, J. C., Castillo, O. (2021).** Optimization using the firefly algorithm of ensemble neural networks with type-2 fuzzy integration for COVID-19 time series prediction. *Soft Computing*, Vol. 27, No. 6, pp. 3245–3282. DOI: 10.1007/s00500-020-05549-5.
  35. **Cordero-Martínez, R., Sánchez, D., Melin, P. (2023).** Optimizing a convolutional neural network with a hierarchical genetic algorithm for diabetic retinopathy detection. *Studies in Computational Intelligence*, pp. 199–208. DOI: 10.1007/978-3-031-22042-5\_11.
  36. **Chetoui, M., Akhloufi, M. A. (2020).** Explainable diabetic retinopathy using efficientnet. *42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 1966–1969. DOI: 10.1109/embc44109.2020.9175664.
  37. **Bodapati, J. D., Shaik, N. S., Naralasetti, V. (2021).** Composite deep neural network with gated-attention mechanism for diabetic retinopathy severity classification. *Journal of Ambient Intelligence and Humanized Computing*, Vol. 12, No. 10, pp. 9825–9839. DOI: 10.1007/s12652-020-02727-z.
  38. **Macsik, P., Pavlovicova, J., Goga, J., Kajan, S. (2022).** Local binary CNN for diabetic retinopathy classification on fundus images. *Acta Polytechnica Hungarica*, Vol. 19, No. 7, pp. 27–45. DOI: 10.12700/aph.19.7.2022.7.2.
  39. **Cordero-Martínez, R., Sánchez, D., Melin, P. (2022).** Comparison of image pre-processing for classifying diabetic retinopathy using convolutional neural networks. *Lecture Notes in Networks and Systems*, pp. 194–204. DOI: 10.1007/978-3-030-96305-7\_18.
  40. **Cordero-Martínez, R., Sánchez, D., Melin, P. (2022).** Hierarchical genetic optimization of convolutional neural models for diabetic retinopathy classification. *International Journal of Hybrid Intelligent Systems*, Vol. 18, No. 1-2, pp. 97–109. DOI: 10.3233/his-220004.
  41. **Amador-Angulo, L., Ochoa, P., Peraza, C., Castillo, O. (2023).** Fuzzy dynamic adaptation of an artificial fish swarm algorithm for the optimization of benchmark functions. *Studies in Computational Intelligence*, pp. 99–114. DOI: 10.1007/978-3-031-28999-6\_6.
  42. **Castillo, O., Lizárraga, E., Soria, J., Melin, P., Valdez, F. (2015).** New approach using ant colony optimization with ant set partition for fuzzy control design applied to the ball and beam system. *Information Sciences*, Vol. 294, pp. 203–215. DOI: 10.1016/j.ins.2014.09.040.
  43. **Amador-Angulo, L., Mendoza, O., Castro, J., Rodríguez-Díaz, A., Melin, P., Castillo, O. (2016).** Fuzzy sets in dynamic adaptation of parameters of a bee colony optimization for controlling the trajectory of an autonomous mobile robot. *Sensors*, Vol. 16, No. 9, pp. 1458. DOI: 10.3390/s16091458.

44. **Valdez, F., Vazquez, J. C., Melin, P., Castillo, O. (2017).** Comparative study of the use of fuzzy logic in improving particle swarm optimization variants for mathematical functions using co-evolution. *Applied Soft Computing*, Vol. 52, pp. 1070–1083. DOI: 10.1016/j.asoc.2016.09.024.
45. **Sánchez, D., Melin, P., Castillo, O. (2017).** A grey wolf optimizer for modular granular neural networks for human recognition. *Computational Intelligence and Neuroscience*, Vol. 2017, pp. 1–26. DOI: 10.1155/2017/4180510.
46. **González, B., Valdez, F., Melin, P., Prado-Arechiga, G. (2015).** Fuzzy logic in the gravitational search algorithm for the optimization of modular neural networks in pattern recognition. *Expert Systems with Applications*, Vol. 42, No. 14, pp. 5839–5847. DOI: 10.1016/j.eswa.2015.03.034.
47. **Tai, K., El-Sayed, A., Biglarbegian, M., Gonzalez, C., Castillo, O., Mahmud, S. (2016).** Review of recent type-2 fuzzy controller applications. *Algorithms*, Vol. 9, No. 2, pp. 39. DOI: 10.3390/a9020039.
48. **Ontiveros, E., Melin, P., Castillo, O. (2020).** Comparative study of interval type-2 and general type-2 fuzzy systems in medical diagnosis. *Information Sciences*, Vol. 525, pp. 37–53. DOI: 10.1016/j.ins.2020.03.059.
49. **Melin, P., Castillo, O. (2003).** A new method for adaptive model-based control of non-linear plants using type-2 fuzzy logic and neural networks. *The 12th IEEE International Conference on Fuzzy Systems*, Vol. 1, pp. 420–425. DOI: 10.1109/FUZZ.2003.1209400.
50. **Moreno, J. E., Sanchez, M. A., Mendoza, O., Rodríguez-Díaz, A., Castillo, O., Melin, P., Castro, J. R. (2020).** Design of an interval type-2 fuzzy model with justifiable uncertainty. *Information Sciences*, Vol. 513, pp. 206–221. DOI: 10.1016/j.ins.2019.10.042.
51. **Guerrero, M., Valdez, F., Castillo, O. (2022).** Comparative study between type-1 and interval type-2 fuzzy systems in parameter adaptation for the cuckoo search algorithm. *Symmetry*, Vol. 14, No. 11, pp. 2289. DOI: 10.3390/sym14112289.
52. **Cuevas, F., Castillo, O., Cortés-Antonio, P. (2022).** Generalized type-2 fuzzy parameter adaptation in the marine predator algorithm for fuzzy controller parameterization in mobile robots. *Symmetry*, Vol. 14, No. 5, pp. 859. DOI: 10.3390/sym14050859.

*Article received on 30/01/2024; accepted on 23/04/2024.  
Corresponding author is Daniela Sánchez.*