Scope
Volume 14 Number 02 June 2024

Tooth Cavity Detection Using Optimized Fuzzy Neural Network and M-GAN for Orthodontic Diagnosis

Ramyaalakshmi A
Vels Institute of Science, Technology & Advanced Studies, Chennai, India

Dr H Jayamangala
Asst. Prof. Department of Computer Applications-PG, Vels Institute of Science, Technology & Advanced Studies, Chennai,India

Abstract: Tooth cavity detection is a critical aspect of orthodontic diagnosis, facilitating timely intervention to prevent further dental deterioration. Traditional methods for cavity detection, such as visual inspection and radiographic analysis, often suffer from limitations in accuracy and consistency, highlighting the need for advanced computational techniques.Therefore, in this study, we propose a novel approach combining an optimized Fuzzy Neural Network (OptFuzNet) with a modified Generative Adversarial Network (m-GAN) for accurate and efficient cavity detection. First, we pre-process dental images to reduce noise and improve image quality. The m-GAN is then employed to generate synthetic cavity images, augmenting the training dataset and improving model robustness. Then, the GLCM features such as contrast, Homogeneity, Energy, Correlation, and Dissimilarity are extracted from each image. Subsequently, the selected features are given to the OptFuzNet classifier to classify an image as a tooth or cavity. To improve the efficiency of the FuzNet classifier, the weight values are optimally selected using The Lyrebird Optimization Algorithm (LyOA). Our experimental results, validated on a benchmark dataset, demonstrate the superior performance of the proposed approach compared to traditional methods, achieving a high accuracy of 94.29% in cavity detection. The proposed framework offers a promising solution for accurate tooth cavity detection, thereby facilitating early diagnosis and effective orthodontic treatment planning.

Keywords: - Tooth cavity detection, optimized Fuzzy Neural Network, modified Generative Adversarial Network, and Lyrebird Optimization Algorithm.

1. Introduction

In the fields of orthodontics and oral and maxillofacial surgery, diagnosing the necessity for orthognathic surgery is crucial. [1] In addition to being costly, orthodontic surgery carries the danger of general anesthesia. As a result, orthodontic therapy is typically preferred by patients over orthognathic surgery. However, orthodontic therapy is not always sufficient to address cranial structural issues. Only orthognathic surgery can treat an asymmetric jaw, a prominent chin, and a retrusive mandible. [2] When orthodontic therapy is insufficient to achieve the desired cosmetic change, orthognathic surgery is also a viable option. Dental clinics commonly employ many kinds of radiography, each with unique characteristics. For diagnosis and subsequent treatment planning, they visualize several locations of interest. [3] Initially, one of the most popular visualization methods in dentistry was panoramic radiography, which scans a large region at a comparatively low radiation dosage. They make it possible for specialists to diagnose a wide range of abnormalities, illnesses, and lesions. However, diagnosing a case or interpreting a critical situation might be challenging due to complex anatomical features, diseases, and imaging distortions. Systems for computer-assisted diagnosis can support physicians in making decisions.

Computerized dental X-rays are presently widely used, mostly because of their advantages in speed, storage, and image quality. Here are some benefits of computerized radiography: It is faster and easier to use than traditional methods; it needs a lower radiation dosage; and it doesn't employ artificial chemicals to create images. It also produces high-quality images that can identify ailments [4]. The traditional machine learning methods cannot accurately distinguish orthodontic images due to their limited capacity [5]. A subfield of machine learning called deep learning is particularly good at processing high-dimensional data, like text and images. Certain classic machine learning-based computer vision tasks, such as detection, and classification, have been entirely superseded by deep learning.

Deep learning has started to be used in dental research for screening, diagnosis, and decision-making. For instance, one study employed deep learning with a sizable dataset of lateral cephalograms (5890 images) to help orthodontists with skeletal categorization. Those authors claimed that their deep learning model achieved >90% sensitivity, specificity, and accuracy in vertical and sagittal skeleton categorization after training and verifying the model. Support vector machines (SVM), [6] feedforward neural networks (NNs), and sequence alignment algorithms were utilized for tooth classification. All things considered, computerized dental X-rays produced an enhanced image quality than traditional oral X-rays since dental professionals may quickly acquire another image if the one that was procured wasn't up to speed.

To begin with, we assembled a wide range of dental images that included teeth deterioration at different stages. [7] Next, we utilized a modified GAN (m-GAN) architecture designed specifically for dental image augmentation, which increased the amount and variety of the dataset while maintaining the fundamental properties of tooth cavities [8]. This was followed by feature extraction, with an emphasis on obtaining gray-level co-occurrence matrix (GLCM) features. The main distinguishing qualities necessary for cavity identification were identified by extracting these properties [9]. In addition, the Optimized Fuzzy Neural Network (OFNN) model is presented, which uses dental images to categorize them into groups corresponding to the presence or lack of cavities. Additionally, the Lyrebird Optimization Algorithm (LOA) is employed to improve the fuzzy neural network's performance, guaranteeing the efficacy of the classification model while reducing computational cost [10].

2. Literature survey

A deep learning model using DeepID was created by Shihao et al. [11] to categorize and preserve orthodontic images. Orthodontic image classification, archiving, and monitoring were achieved using deep learning techniques more quickly and accurately than manual methods. They demonstrated the improved DeepID-based model's outstanding orthodontic image classification performance. Additionally, deep learning reduced the strain on dentists by improving the efficiency of dental follow-up and treatment. However, their study had various limitations. Firstly, as their strategy relied on orthodontically needed images, the outcomes should be interpreted cautiously. Secondly, many training samples with excellent annotation were required for deep learning to function well.

Jae-Jun et al. [12] developed DCNN techniques for the cephalometric radiograph image-based differential diagnosis of orthognathic surgical indications. The deep learning diagnosis featured differentiating each image versus evaluating everyone's image with a specific measurement. This approach had the advantage of recognizing nuances that measurements could not capture. Grad-CAM visualization demonstrated the capability of descriptive AI and verified the ROI required for classification. They discovered that DCNNs were quite useful in differentiating between orthognathic and surgical reasons. However, their study had limitations, including the need for different deep learning architectures, an adequate amount of datasets, and image sets with validated labels.

Modern deep learning models gave rise to tests that were dependable and robust for clinical decisions, as introduced by Mahmut Emin Celik [13]. This advanced patient treatment planning and physician health management by supporting dentists in their

decision-making. It was demonstrated that the detection of third molars worked for all four models. The application of machine learning in dentistry showed great promise for highly precise and accurate diagnostics. They succeeded in demonstrating how AI-based tools had become prevalent in hospitals and how important they were for making diagnostic recommendations. However, their work was limited in two ways. First, the quantity of data was constrained. Second, since they were more common, mandibular third molars were utilized in this study.

Computer-aided diagnosis created by Dmitry et al. [14] was on par with expert level. Based on these results, the approach for automated dental radiograph analysis might find practical use and require further assessment. Completing digital dental charts was made easier by computer-assisted tooth recognition and numbering. When only assessment and small adjustments were required, a radiologist could utilize its output for automatic charting rather than supplying data, as its efficiency level was nearly at the expert level. They succeeded in automating processes, which might enhance time savings and the accuracy of electronic dental data. However, the study had several drawbacks regarding the use of bounding boxes for tooth isolation.

Al Kheraif et al. [15] developed a dental illness prediction technique based on CNN and the hybrid graph-cut technique. Dental 2D image files were first gathered using an X-ray camera model, which successfully captured about 1500 images. The images were separated into training (800) and testing (700) sets to form a data set. Next, the quality of the images was assessed by examining their pixels. Subsequently, the impacted areas were divided by creating a scribbling line on the image to anticipate the different sections. They effectively analyzed different statistical dental images based on anomalous or disease-related aspects. Finally, an evaluation of the system using dental images was conducted. Their shortcomings did not lie in fewer errors or better dental care.

Leite et al. [16] introduced an AI-driven tool for quick and precise segmentation and tooth recognition on panoramic radiographs, achieved through the refined use of two distinct deep-learning algorithms. Tooth segmentation was found to be more challenging than bone segmentation due to several factors, including the number of teeth in each jaw, the closeness of neighboring tooth structures, the variation in densities within teeth, and dental growth. They achieved the tool attained both sensitivity and accuracy for tooth detection. However, the limitations of a panoramic scan made this approach considerably harder. They were unable to improve the ultimate accuracy or convergence speed of the segmentation task.

An effective dental diagnostic method was created by Geetha et al. [17]. They applied adaptive thresholding, morphological operations, and the Laplacian filter for image

enhancement. Sixteen statistics extracted from the segmented image were then applied to a Backpropagation Neural Network (BPNN) for classification, resulting in an effective classifier. The diagnostic system was found to distinguish better between decayed and normal X-ray images. They discovered that the BPNN performed remarkably well in diagnosing decay in dental radiographs. In dental practice, advanced algorithms and large, high-quality datasets could provide superior decay diagnosis. However, the caries depth categorization method has never been improved.

A unique strategy based on the deep learning model YOLOv3 was developed by Almalki, et al. [18] for the detection and classification of the four most prevalent tooth problems: cavities, root canals, dental crowns, and broken-down roots. Unlike other automated tooth issue detection and sorting applications, dental disease datasets are distinct in that access to tagged medical datasets presents substantial difficulty. They developed an automated program capable of recognizing and categorizing dental issues on dental panoramic X-ray pictures (OPG) using deep learning. Because a dataset was not available, they created their collection of dental X-rays. After training, the YOLOv3 model was tested on test images and succeeded. One of the methodology's limitations is that dental anomalies will be detected in realtime.

3. Problem definition with contribution

The rapid advancements in artificial intelligence have catalyzed the development of numerous technologies in the healthcare sector. Among these, tooth cavity detection stands out as a crucial aspect of dental healthcare, essential for timely intervention and decay prevention. Traditional methods of cavity detection rely heavily on visual examination by dentists or dental hygienists. They search for visual cues such as discoloration, soft spots, or visible cavities. Diagnostic aids like dental radiography provide detailed images of tooth structure, aiding in cavity identification. However, these methods are not without limitations. Visual inspection can be subjective and prone to variability between practitioners. Additionally, radiography exposes patients to radiation and may miss early-stage cavities or lesions in certain areas. In recent years, machine learning has emerged as a powerful tool for improving tooth cavity detection, addressing the limitations of traditional methods. Machine learning architecture has demonstrated remarkable capabilities in accurately recognizing patterns and features in dental images relevant to cavity detection. Through training on extensive datasets of dental images, neural networks can efficiently learn complex patterns indicative of tooth cavities, thereby enabling automated and precise detection. Moreover, machine learning techniques allow for the integration of various data sources and modalities pertinent to cavity detection. Our study aims to contribute significantly to the field of tooth cavity detection through cutting-edge machine-learning techniques.

- Firstly, we collected a diverse dataset comprising dental images encompassing various stages of tooth decay.
- Subsequently, we employed a modified GAN (m-GAN) framework tailored for dental image augmentation, enhancing the dataset's size and diversity while preserving the underlying characteristics of tooth cavities.
- Following that, feature extraction was conducted, focusing on the extraction of Gray-level co-occurrence matrix (GLCM) features. These features were extracted to pinpoint the most discriminative characteristics essential for cavity detection.
- Moreover, we introduce the Optimized Fuzzy Neural Network (OptFuzNet) model, to classify dental images indicative of cavity presence or absence.
- Furthermore, the Lyrebird Optimization Algorithm (LyOA) is utilized to enhance the performance of the Fuzzy neural Network, ensuring the classification model's effectiveness while minimizing computational overhead.

4. Proposed Tooth cavity detection using OptFuzNet

The primary objective of this paper is to identify and diagnose areas of decay in teeth accurately and efficiently. By detecting cavities early, dental professionals can initiate timely interventions to prevent further progression of decay, minimize the risk of complications such as infection and tooth loss, and ultimately preserve the patient's oral health and well-being. Additionally, early detection allows for less invasive and more conservative treatment options, reducing patient discomfort and treatment costs. So, in this study, we focused on tooth cavity detection. Figure 1 illustrates the proposed structure of tooth cavity detection. The proposed work consists of three stages namely, pre-processing, data augmentation, and classification. Initially, data are collected pre-processed. Then, the pre-processed data are given to the augmentation process. In this paper, we employed m-GAN for data augmentation, aiming to enhance the diversity and quantity of our dataset, thereby improving the robustness and accuracy of tooth cavity detection. Then, the GLCM features are extracted from newly generated images. After that, the extracted features are given to the OptFuzNet classifier to classify an image as a tooth or cavity. Here, the performance of FuzNet is enhanced using LyOA.

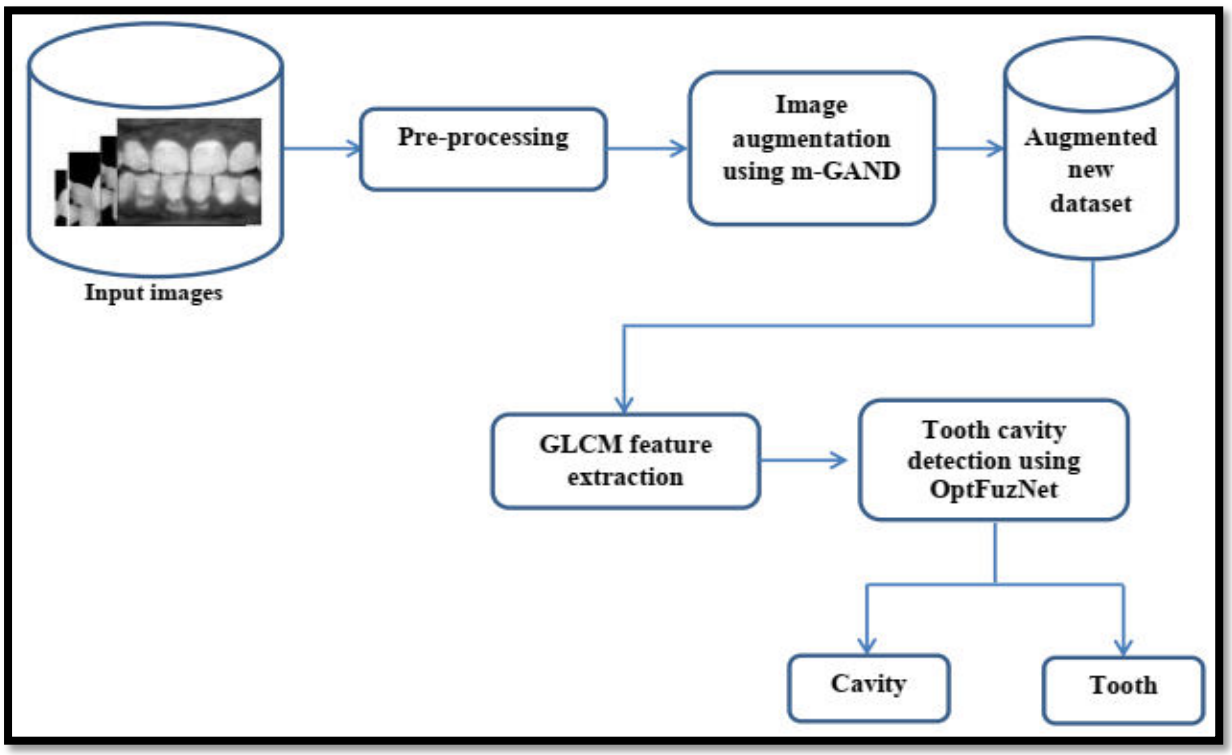

Figure 1: Overall structure of proposed cavity detection

4.1 Pre-processing

Consider the dataset (Zenodo's Dental Caries Dataset) D_j ,$(j = 1,2,...,N)$ which consists of N number of tooth images. Initially, we downloaded the images from the dataset and Dataset and uploaded them to Google Drive for ease of access. Following this, all images are transformed into grayscale to standardize their color representation and subsequently resized to dimensions of 256x256 pixels to ensure uniformity and facilitate efficient processing. These pre-processing steps set the foundation for accurate analysis and detection of cavities in tooth images.

4.1 Image augmentation using m-GAN

Data augmentation is an important technique to improve the classification process, especially when the initial dataset is insufficient. By creating different and artificial variations of existing data, it increases their accuracy and robustness while making predictions. Data augmentation aims to increase the diversity and robustness of a dataset to improve the performance of algorithms in data analysis and prediction tasks. For augmentation, in this paper m-GAN is used. GAN is a popular generative model in deep learning, which is used to generate realistic synthetic data by training two neural networks in opposition: a generator that creates data and a discriminator that evaluates its authenticity. This technique is used to produce high-quality data samples for various applications, such as image generation, data augmentation, and

improving machine learning models. To enhance the performance of GAN, in this paper m-GAN is proposed.

GAN

The original GAN has two components: a generator and a discriminator. The generator's input is random noise, and the discriminator's input is both real data and so-called fake data (generated by the generator). The discriminator's goal is to accurately distinguish between the real data and the generated fake data. The structure of GAN is given in Figure 2.

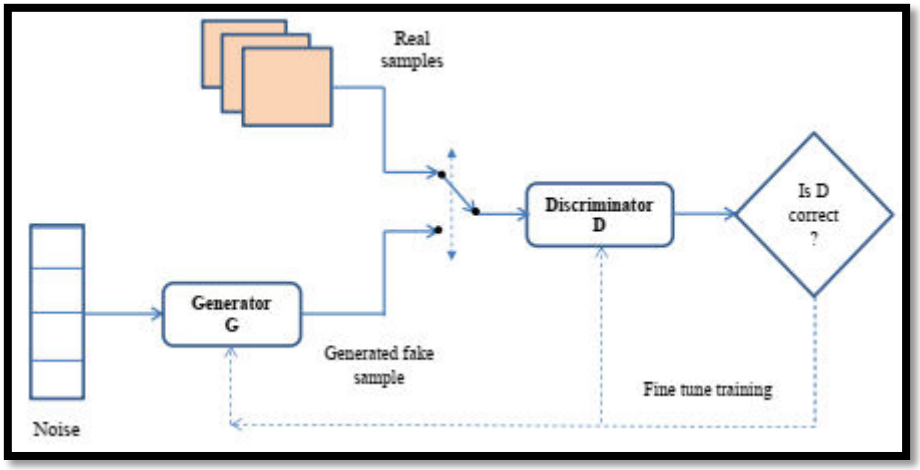

Figure 2: The structure of traditional GAN

In GANs, both the generator and discriminator are trained using feedforward networks and the dropout algorithm. These two components improve each other adversarial: the discriminator keenly learns to distinguish real samples and provides feedback on synthetic samples, while the generator uses this feedback to create increasingly realistic samples. The model converges when the discriminator can no longer tell the difference between real and generated data. To generate the data and classify data, the generator and discriminator, both have various objective functions: the discriminator aims to maximize likelihood, while the generator aims to minimize the error rate of the discriminator [19]. The maximum likelihood function is calculated using equation (1).

$$Lik(\theta) = \prod_{i-1}^{n} f(x_i|\theta) \qquad (1)$$

Where; n represent the total number of images given to the input of the discriminator and sample data created from the generator is represented as x. The comparison of the two samples is utilized to assess their similarity, which ultimately determines the acceptability of the generated sample. The mathematical function of error is given in equation (2).

$$S\tan d\ error\ of\ difference = \sqrt{\frac{(p+q)^2 - (p-q)}{n}} \qquad (2)$$

Where $p\ and\ q$ represent data points used to evaluate the modification between the created data and the original data. GANs have proven useful in balancing datasets, but the use of basic GANs to classify data with small sample sizes has not been extensively studied. Traditional GANs introduce random noise into the original dataset to generate synthetic models, which is not optimal for the sensitive application area of dental cavity detection. Therefore, in this paper, we propose the use of m-GAN for tooth cavity detection.

Modified GAN

We present m-GAN, where based on conventional GAN, the discriminator is trained with the original image.However, diverging from the conventional approach, the generator in our model is fed both the original image and noise (information about the original image) with distribution $P(n)$, as input. The structure of m-GAN is given in Figure 3. This modification aims to enhance the generator's ability to produce more realistic and diverse synthetic samples by incorporating the variability introduced by the noise. Here, the generator creates a synthetic sample which is executed by a function generator, $G_g(A)$. The resultant produced by the generator is used as input for the discriminator. The discriminator has a function called `discriminate`, denoted as $D_d(\ )$.

The resultant produced by the generator is used as input for the discriminator, which has a function called discriminate, denoted as $D_d(\ )$. The discriminate function, $D_d(G_g(A))$, evaluates the likelihood between the generated and original samples to determine the status of the generated sample. If rejected, feedback is passed to the generator, which calculates the error difference and updates it to produce a new synthetic sample.

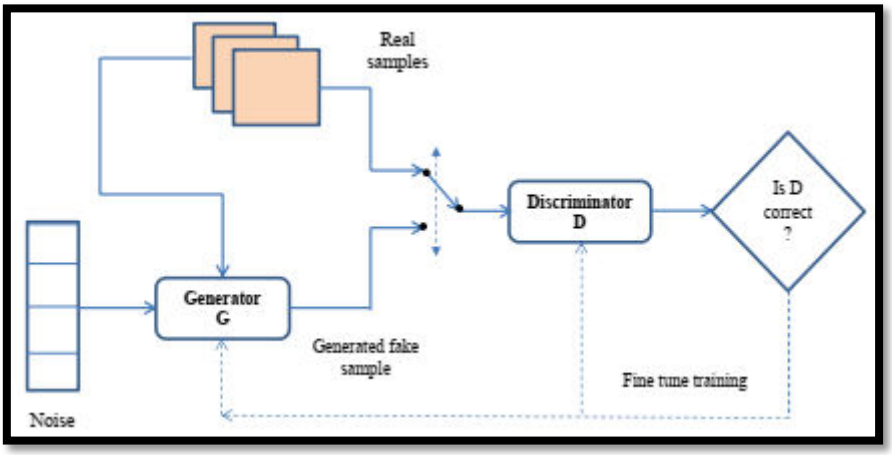

Figure 3: Structure of m-GAN

Algorithm 1: m-GAN
Input: Discriminator: input image, generated image Generator: Gaussian noise and information about the original image **Output:** Generated synthetic image

Begin
Generator
Synthetic sample = Generate (Gaussian noise+ image information)
Do
{
Discriminator
Differentiate (synthetic sample images)
Evaluatethe maximum likelihood factor to distinguish the generated sample from the input samples
Synthetic samples are accepted if the likelihood is high
Otherwise, create feedback and transferit to the generator.
Generator
Synthetic data=Generate (Feedback+ Gaussian noise+ Image information)
The generator evaluates the standard error difference to produce new data
}
While (synthetic sample=accepted)
The accepted generated samples are added to the dataset

4.3 Feature extraction

After the data augmentation process, the newly generated image datasets are fed into the feature extraction stage. For feature extraction, we extract the texture GLCM features such as contrast, Homogeneity, Energy, Correlation, and Dissimilarity from the generated images.

Energy: - It is also known as uniformity or the angular second moment. It is a statistical measure that quantifies the texture uniformity or repeated pair patterns in an image. It is calculated as the sum of the squared elements in the GLCM, providing a measure of textural uniformity. Mathematically, it is expressed as:

$$Energy = \sum_{i,j} P(i,j)^2 \qquad (3)$$

Where, $P(i,j)$ represent the normalized frequency of pixel pairs with gray levels i and j in the GLCM. High energy values indicate less texture complexity and more homogeneity, while lower values indicate more texture variation and complexity.

Contrast: - The "contrast" feature in the GLCM measures the intensity difference between a pixel and its neighbor over the whole image, highlighting the amount of local variation. For tooth cavity detection, this feature helps identify areas with significant differences in pixel intensity, which often correspond to the presence of cavities. Mathematically, it is defined as:

$$Contrast = \sum_{i,j} (i-j)^2 . P(i,j) \qquad (4)$$

Higher contrast values indicate greater differences in intensity, which can be useful for detecting the variations in tooth density and structure associated with cavities.

Homogeneity:- The "homogeneity" feature in GLCM measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal, reflecting the similarity or smoothness of the image texture. For tooth cavity detection, this feature helps identify areas where pixel intensities are similar, as healthy tooth regions typically exhibit more uniform texture compared to the irregularities caused by cavities. Mathematically, homogeneity is defined as:

$$Homogeneity = \sum_{i,j} \frac{P(i,j)}{1+|i-j|} \qquad (5)$$

Higher homogeneity values indicate more uniform regions, which can help distinguish healthy tooth areas from those with cavities, as cavities often disrupt the uniform texture of the tooth surface.

Correlation: The "correlation" feature in the GLCM measures the linear dependency of gray levels between pixels at specified positions relative to each other. For tooth

cavity detection, this feature helps identify the degree to which pixel intensities are correlated, which can reveal structural patterns or disruptions caused by cavities. Mathematically, correlation is defined as:

$$Correlation = \frac{\sum_{i,j}(i.j.P(i,j)) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (6)$$

Where; $P(i,j)$represent the normalized frequency of pixel pairs with gray levels i and j in the GLCM, μ_x and μ_y are the mean value, $\sigma_x and \sigma_y$ are the standard deviation of the marginal distribution of $P(i,j)$.High correlation values indicate a strong linear relationship between the pixel intensities, which typically corresponds to consistent and uniform textures found in healthy tooth regions. Lower correlation values may indicate the presence of cavities, as these areas often disrupt the regularity and introduce more variability in the texture.

Dissimilarity: The "dissimilarity" feature in theGLCM quantifies the average difference in intensity between a pixel and its neighbor over the entire image. For tooth cavity detection, this feature helps identify areas where there are significant variations in pixel intensity, potentially indicating the presence of cavities or irregularities in tooth structure. Mathematically, dissimilarity is defined as:

$$Dissimilar\,ity = \sum_{i,j} |i-j|.P(i,j) \quad (7)$$

Higher dissimilarity values indicate greater intensity differences between neighboring pixels, which can be indicative of areas with irregularities or cavities. Therefore, dissimilarity is a useful feature for detecting textural variations associated with tooth cavity detection.

4.4 Tooth cavity detection using Optimized Fuzzy Neural Network (OptFuzNet)

After the feature extraction process, the extracted features are given to the OptFuzNet for the cavity detection process. The proposed OptFuzNet classifier is a combination of a fuzzy inference system (FIS) and a neural network (NN). To enhance the performance of FuzNet, the parameter present in the classifier is optimally selected using**the** Lyrebird Optimization Algorithm (LyOA).In this work, we consider the problem of multi-attribute decision-making and build a multi-input, single-output (MISO) FuzNet, as shown in Figure 3. We use different features of the tooth image as input vectors and the cavity as the output vector. The proposed FuzNet consists of four layers namely, input, membership function, rule, and output layers.The first layer of nodes represents input variables, with individual nodes representing linguistic descriptions of contrast, Homogeneity (Hm), Energy (E), Correlation (Co), and Dissimilarity (Di).Layer 2 computes membership values based on the linguistic terms associated with the variables specified in Layer 1. Layer 3 generates the fuzzy rules

based on the membership values (Takes from layer 2). Layer 3 takes the input and applies fuzzy rules to determine the rule firing strength (outputs), which are then used in subsequent layers for further processing. Finally, Layer 4, acting as the output layer, integrates the information representing the output of the cavity detection.

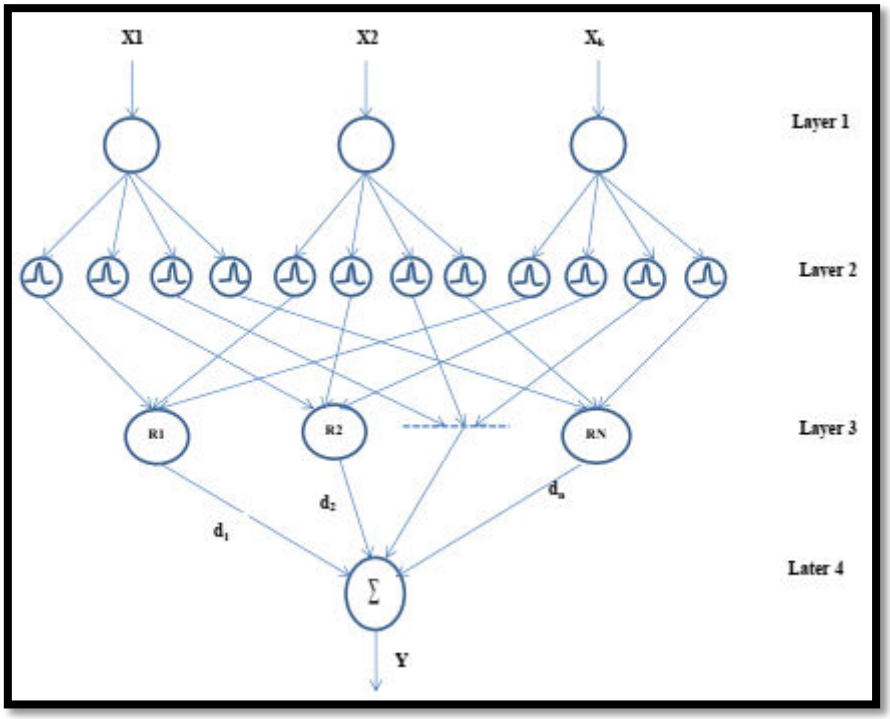

Figure 4: Structure of the Fuz Net model

The FIS uses the bell membership function (MF) in the second layer layer. However, to improve the performance of the rule generation process, the proposed method utilizes the Gaussian kernel MF. Besides, to improve the performance of FuzNet, weight values are optimally selected by adaptive LyOA. The two basic rules of Fuzzy are given in below equations;

Rule R_n: If F_1 is Q_{1n} and ... F_m is Q_{mn} .... and F_K is Q_{Kn} Then O is d_n, (8)

Where n=1, 2,..., N and Q_{mn} are the input variables fuzzy sets, F_m, i=1,2,...K and d_n are the consequent parameter of O. To simplify the analysis, a fuzzy rule o is introduced.

Rule o: If F_1 is Q_{10} and ... F_m is Q_{m0} .... and F_K is Q_{K0} Then O is d_0, (9)

Where Q_{K0} is a universal fuzzy set, whose fuzzy degree is 1 for any input value F_m, i=1,2,...K, and d_0 is the consequent parameter of Oin the fuzzy rule o. Define $O^{(K)}$ and $X^{(K)}$ as theoutput and input variables of a node in layer K, respectively.

Layer 1: Input layer: -Initially, the extracted features (Hm, E,C, and Di) are given to the input of the input layer. The input layer is in the form of crisp variables (L, H, M).

There is no computation in this layer. Each node in this layer, directly transmits the corresponding input variable to the next layer.

$$O^{(1)} = X_m^{(1)} = F_m \quad (10)$$

Where F_m, i=1,2,...K are the input variables of the FuzNet.

Layer 2: MF layer: - Every node in this layer performs MF that resembles one linguistic label of one of the input variables in Layer 1. In this paper, the Gaussian kernel membership function is used.

$$O^{(2)} = \mu_m^{(n)}(X_m^{(2)}) \quad (11)$$

Where $\mu_m^{(n)}(\cdot)$ is anMF. With the use of GMF, the operation is performed. The GMF is calculated using equation (12).

$$O^{(2)} = e^{-((X_m^{(2)} - c_{mn})^2 / \sigma_{mn}^2)} \quad (12)$$

Where c_{mn} represents the center (mean) and σ_{mn} represents the width (or variance) of the GMF of the n^{th}term of the m^{th}input variable f_m.

Step 3: Rule layer: Links are used in this layer to match fuzzy logic rules' preconditions, and each rule node has a single antecedent link that comes from a linguistic variable. To fuzzify the inputs, fuzzy operators are used, such as AND.

$$O^{(3)} = \prod_{m=1}^{K} X_m^{(3)} = e^{-[Dia_n(z-c_n)]^T [Dia_n(z-c_n)]} \quad (13)$$

Where $Dia_m = diag(1/\sigma_{1n},...,1/\sigma_{Kn})$, $c_n = [c_{1n}, c_{2n}, ...c_{Kn}]^T$, $F = [F_1, F_{2,} F_3, ...F_K]^T$ is the FNN input vector.

Step 4-Output layer: The single node $O^{(4)}$ in this layer is labeled with $\sum$, and it computes the overall output by summing all the input signals.

$$O^{(4)} = \sum_{m=1}^{N} d_m \times X_m^{(4)} + d_0 \quad (14)$$

Where, d_m represents the output action strength of Layer 4 and the scalar d_0 is a bias. Therefore, the FuzNet mapping can be reformulated in the subsequent input-output structure:

$$O^{(4)} = \sum_{n=1}^{N} d_n \times X_n^{(4)} + d_0 = \sum_{n=1}^{N} d_n \prod_{m=1}^{K} \mu_m^{(n)}(F_m) + d_0 \quad (15)$$

The following illustration shows how FuzNet processes input data through layers, interprets fuzzy rules, and produces output representing cavity detection.

- The input layer represents input variables such as Hm, E, C, and Di, and each node transmits linguistic descriptions {Low, Medium, High} to the next layer.
- Layer 2 computes membership values based on linguistic terms from Layer 1 using Gaussian MFs. For example, if Di is high, the MF in Layer 2 calculates the degree of membership in the 'high' fuzzy set using a Gaussian function with specific mean and variance parameters.

- In Layer 3, the nodes create the fuzzy logic rules and perform precondition matching using the AND operation for each Layer 2 node. The model rule is presented below;

  RULE:"If $\mathbf{H_m}$ is Low AND **E** is Medium AND **C** is High AND $\mathbf{D_i}$ is Medium, THEN $\mathbf{I_{in}}$ is Cavity,"

  Table 1: Sample Fuzzy Rules

INPUTS				**OUTPUT**
$\mathbf{H_m}$	**E**	**C**	$\mathbf{D_i}$	$\mathbf{I_{in}}$
High	Low	Medium	Low	Cavity
Low	Medium	Medium	Medium	Non-Cavity
High	High	Medium	High	Cavity
Medium	Low	High	Low	Cavity
Low	High	High	High	Non-Cavity
High	Low	Low	Low	Non-Cavity

- In Layer 4, a singular node computes the total output by aggregating all inputs, considering connecting weights and biases. The resulting overall output is represented in binary form, where 0 denotes the presence of a cavity and 1 indicates its absence.

4.4. 1 Weight optimization using LyOA

The Lyrebird Optimization Algorithm is a nature-inspired computational technique modeled after the behavior of the lyrebird, known for its exceptional mimicry skills [20]. This algorithm leverages the lyrebird's ability to adapt and imitate various sounds to find optimal solutions in complex problem spaces. By simulating the mimicry and adaptive behaviors of the lyrebird, the algorithm explores and exploits the search space efficiently, balancing exploration and exploitation. It is particularly useful for solving optimization problems where traditional methods may struggle, providing a robust and flexible approach to finding high-quality solutions. The steps involved in weight optimization using LyOA are given below;

Step 1: Solution initialization

In the initialization stage of the optimization algorithm, we first define the solution space and constraints. An initial population of candidate solutions is generated randomly to ensure diversity. In this study, we initialize weight parameters ($w_1, w_2, ...w_n$) present in the FuzNet. The solution can be represented as follows;

$$Y_i = [y_1, y_2, y_3, ...y_n] \quad (16)$$

Where the solution y_1 can be represented as follows,

$$y_1 = \lfloor w_1, w_{2,...w_n} \rfloor \quad (17)$$

Step 2: Opposite solution generation: Then, the opposite solutions are created for each initialized solution. The opposite solution is given in equation (18)

$$\overline{Y}_i = a + b - Y_i \quad (18)$$

Where, $Y_i \in [a,b]$ is a real number

Step 3: Fitness calculation: After initializing the solutions, we calculate the fitness of each one. In this paper, accuracy is used as the fitness function. A good system should aim to achieve the maximum fitness value. The fitness function is defined in Equation (19).

$$Fitness = Max(Accuracy) \quad (19)$$

$$Accuracy = \frac{Tr^P + Tr^N}{Tr^P + Tr^N + Fa^P + Fa^N} \quad (20)$$

Where; Tr^P represents the true positive value, Tr^N represents the true negative value, Fa^P represents the false positive value and Fa^N represents the false negative value.

Step 4: Updation process

In the LyOA approach design, the location of population members is updated in each iteration based on the lyrebird's strategy for sensing danger. In the LyOA design, the population optimization process mimics the lyrebird's behavior, consisting of two phases: (i) escape and (ii) hiding. Using Equation (21), the decision-making process between these strategies is simulated. Thus, in each iteration, each LyOA member's state is updated based on either escape or hiding.

$$Updatedsolution\ \ Y_i = \begin{cases} Based\ on\ phase\ 1,\ Ra_p \le 0.55 \\ Based\ \ on\ \ phase\ 2, else \end{cases} \quad (21)$$

Where; R_p represent the random value [0,1].

Phase 1: Escaping strategy

In the LyOA, the Escaping Strategy involves updating the position of each population member to simulate a lyrebird moving from a dangerous area to a safe one. This movement results in significant positional changes, allowing the algorithm to explore different regions of the search space effectively. Each member considers the positions of others with better objective function values as secureplaces. Thus, the set of secure places for each member is calculated using equation (22), enhancing LyOA's global search exploration capability.

$$SA_i = \{Y_k,\ \ Fit_k < Fit_i \ \ and \ \ k \in \{1,2,...,N\}\},\ \ Where\ i = 1,2,..,N, \quad (22)$$

Where, SA_i represent the set of secure places for the i^{th} lyrebird and Y_k represent the k^{th} row of Y the matrix, which has a better objective function value (i.e., F_k) than the i^{th} LyOA member (i.e, $F_k < F_i$).

In the LyOA design, each lyrebird arbitrarily escapes to one of the identified safe areas. A new location for each member is evaluated using a specific displacement model using Equation (23). If this new location results in an improved objective function value, it replaces the member's previous position according to Equation (24).

$$y_{i,j}^{P1} = y_{i,j} + r_{i,j}.\left(SSA_{i,j} - I_{i,j}.y_{i,j}\right) \quad (23)$$

$$Y_i = \begin{cases} Y_i^{P1}, & F_i^{P1} \le F_i \\ Y_i, & else \end{cases} \quad (24)$$

Here, SSA_i represent the selected secure area for i^{th} lyrebird, $SSA_{i,j}$ represent the j^{th} dimension, Y_i^{P1} represent the new locationevaluated for the i^{th} lyrebird based on the escaping strategy of the proposed LOA, $y_{i,j}^{P1}$ represent the j^{th} dimension, F_i^{P1} represent the objective function value, $r_{i,j}$ represent the arbitrary value [0.1], and $I_{i,j}$ are random value which is selected as for this paper.

Phase 2: Hiding Strategy (Exploitation Phase)

In this LyOA phase, each population member's position is updated by emulating a lyrebird's strategy of hiding in nearby safe areas. This involves small, precise movements to scan the environment, showcasing LyOA's local search exploitation ability. A new location for each member is calculated using Equation (25) and replaces the old location if it improves the objective function value based on Equation (26).

$$y_{i,j}^{P2} = y_{i,j} + \left(1 - 2r_{i,j}\right).\frac{UB_j - LB_j}{t} \quad (25)$$

$$Y_i = \begin{cases} Y_i^{P2}, & F_i^{P2} \le F_i \\ Y_i, & else \end{cases} \quad (26)$$

Where, Y_i^{P2} represent the new locationevaluated for the i^{th} lyrebird based on the hiding strategy of the proposed LyOA, $x_{i,j}^{P2}$ represent the j^{th} dimension, F_i^{P2} represent the objective function value, $r_{i,j}$ represent the arbitrary value [0,1], and t represent the iteration counter.

Step 5: Termination criteria: The process of updating the population is repeated until the final iteration of the algorithm. If not, the algorithm will terminate when the optimal fitness is attained.Upon completion of adaptive LyOA, the best solution

(weight) found is output as the problem's solution. The updated weight value is given to the OptFuzNet.

Algorithm 2: Pseudocode of Adaptive LyOA

Input: population size (N), iteration (T), number of weights, parameters of LyOA

Output: optimal solution (weight)

Start LyOA
Create the initial population using weight values
Evaluate the opposite solution (18)
Evaluate fitness for each solution using equation (19)
For t=1 to T
For i=1 to N
Equation (21) can be used to select the updating strategy
If $Ra_p \leq 0.55$ (select phase 1)
Evaluate secureregions for i^{th} lyrebird using equation (22)
evaluate the new location of the i^{th} lyrebird using equation (23)
Update i^{th} lyrebird member using equation (24)
Else (select phase 2)
Evaluate the new location of the i^{th} LOA member using equation (25)
Update i^{th} LOA member using equation (26)
End if
End for
Store the best solution
End for
Select the optimal best solution.
End LyOA

5.Results and discussion

In this section, the findings of the study are presented and explained in the context of the research objectives. This section clarifies the significance of the results and explores their implications for the wider field of study. The proposed methodology is implemented using Python and the experimental used system has Windows 10 with an Intel Core i5 processor and 6GB of RAM. The efficiency of the presented technique is discussed using accuracy, precision, recall, F-measure, and positive predictive value (PPV). For experimental analysis, Zenodo's Dental Caries Dataset is utilized.

5.1 Experimental results

In the Experimental Results section, we present the outcomes of our experiments. We describe how we conducted the tests and explain what we observed. These results will help us understand our findings better in the upcoming discussion.The confusion matrix for cavity detection is shown below, where the values represent the percentage of instances in each category:

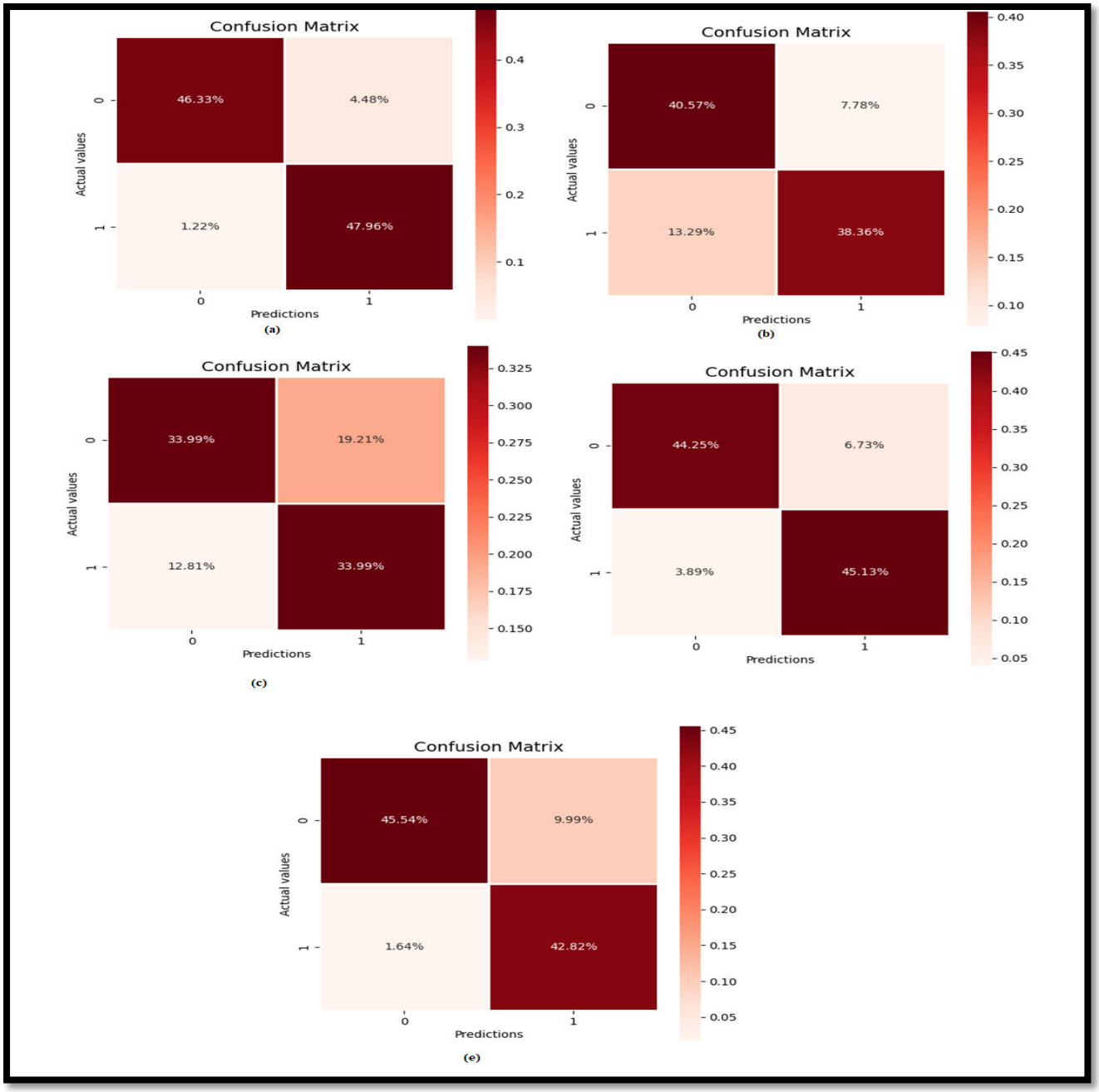

Figure 5: Confusion matrix (a) OptFuzNet, (b) neural network, fuzzy logic, fuzzy-neural network, and SVM

In Figure 5(a),the matrix indicates that 46.33% of the instances were correctly identified as cavities (True Positives), while 47.96% were correctly identified as non-cavities (True Negatives). Misclassification rates include 1.22% of non-cavity instances incorrectly labeled as cavities (False Positives) and 4.48% of cavity instances incorrectly labeled as non-cavities (False Negatives). The high percentages of True Positives and True Negatives demonstrate the model's effectiveness in accurately detecting cavities, with minimal error rates, indicating strong overall performance in distinguishing between cavity and non-cavity instances.

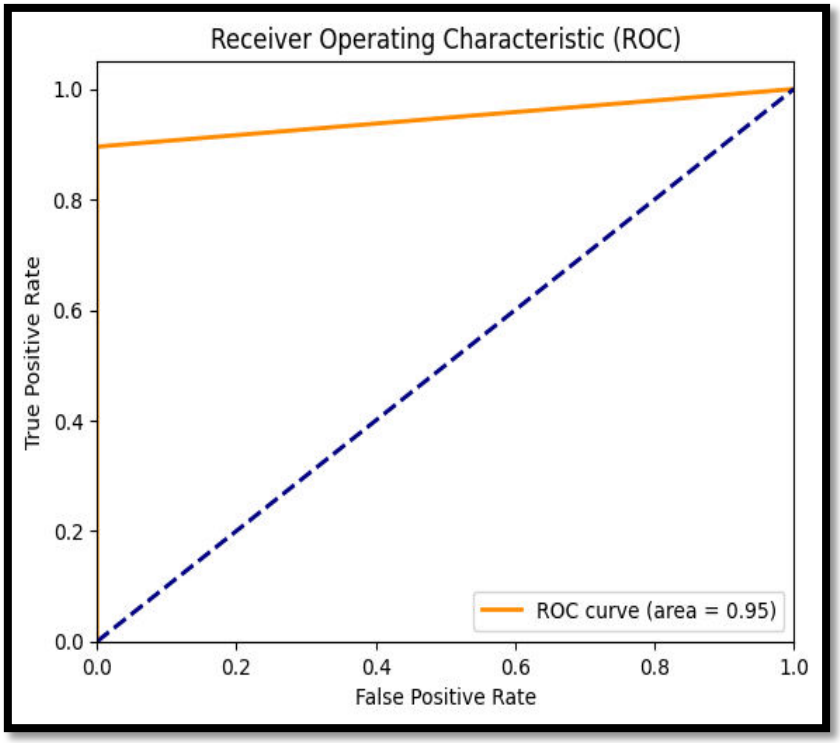

Figure 6: ROC curve of the proposed method

Figure 6 illustrates the ROC curve, this is a graphical representation used to evaluate the performance of a binary classifier by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) across various thresholds. The area under the ROC curve (AUC-ROC) quantifies the overall ability of the classifier to distinguish between positive and negative classes, with values ranging from 0 to 1. In our work, we attained an AUC-ROC of 0.95;which indicates excellent model performance, meaning there is a 95% chance that the classifier will correctly differentiate between a positive instance and a negative instance. This high value reflects the model's strong ability to minimize both false positives and false negatives, making it highly reliable for tasks requiring accurate classification.Moreover, figure 7 illustrates the existing methods of ROC cure.

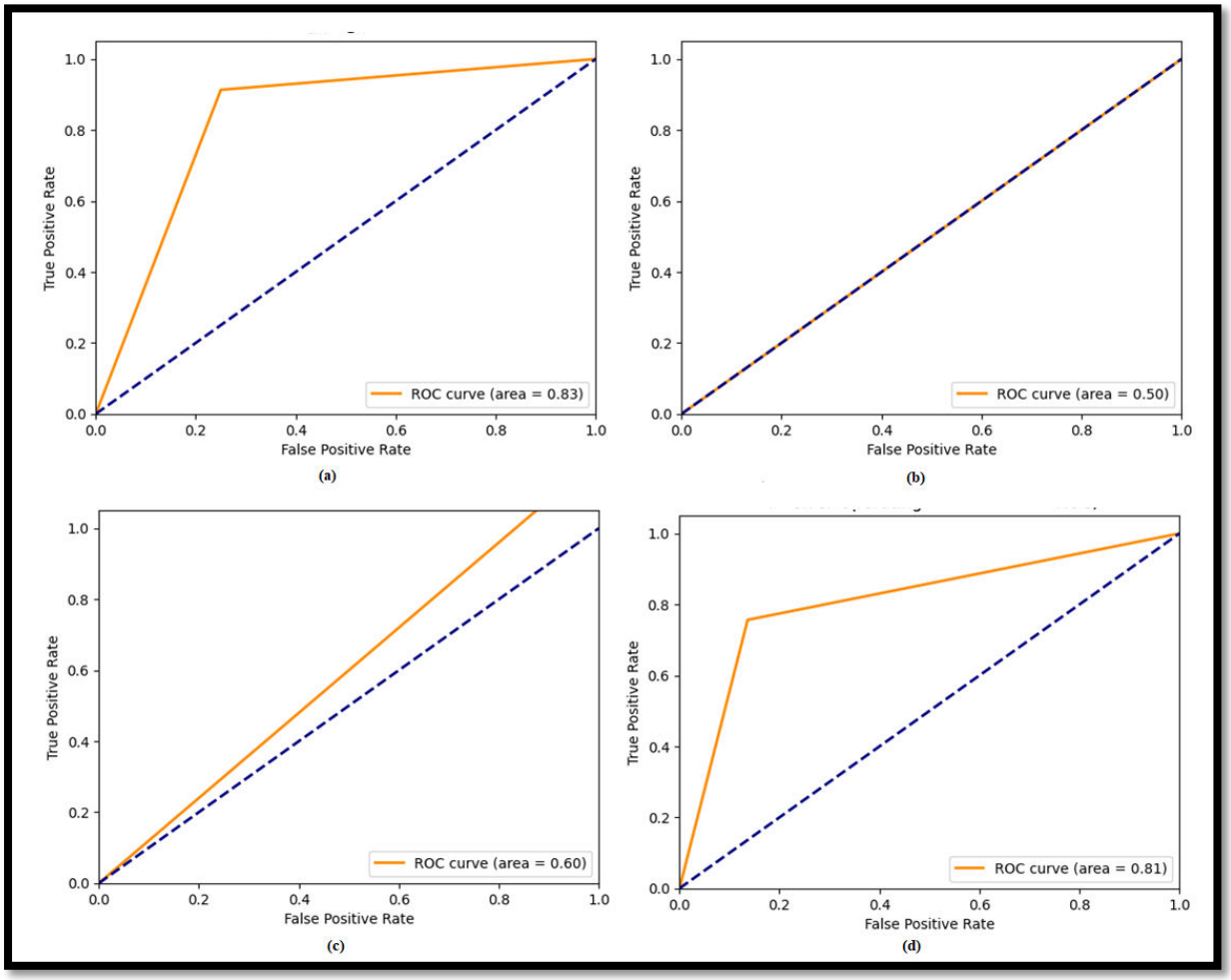

Figure 7: ROC curve of exiting model (a) Fuzzy neural network, (b) fuzzy logic system, (c) Neural network, and (d) SVM

Figure 7 illustrates the ROC curve performance for different classification models.The obtained ROC curve values for different models indicate varying levels of performance in distinguishing between positive and negative classes: the fuzzy neural network achieves an AUC-ROC of 0.83, suggesting good discrimination ability; the fuzzy logic model has an AUC-ROC of 0.50, indicating performance equivalent to random guessing; the neural network attains an AUC-ROC of 0.60, reflecting marginally better than random performance; and the SVM (Support Vector Machine) shows strong performance with an AUC-ROC of 0.81. The proposed model, with an AUC-ROC of 0.95, significantly outperforms all other models, demonstrating exceptional classification ability and a robust capability to correctly identify positive and negative instances, thus representing a substantial improvement over the existing methods.

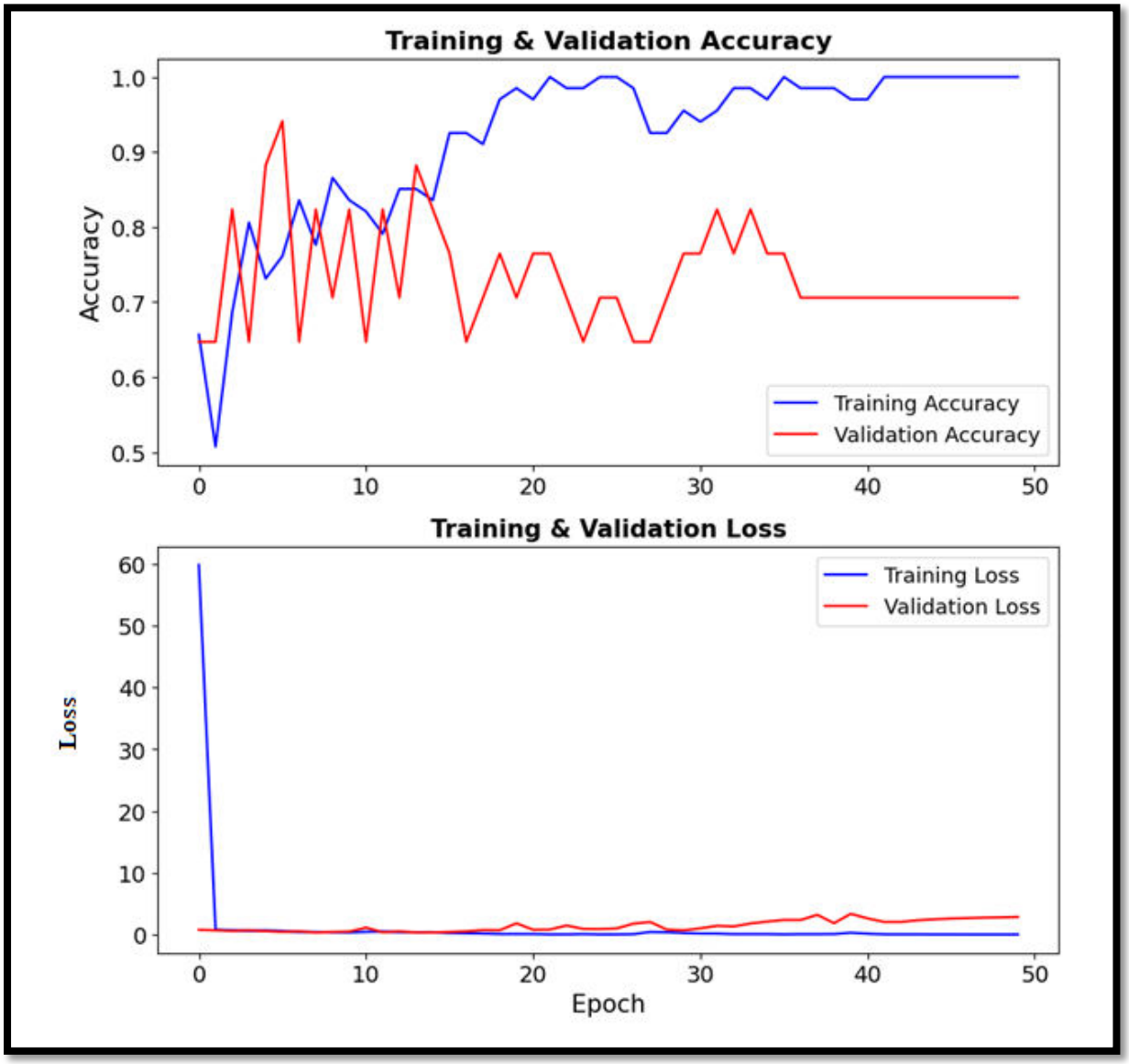

Figure 8: Epochs vs. loss and Accuracy plot for training and validation

A plot of epochs versus loss and accuracy for the proposed model is shown in Figure 8, which indicates a steady decrease in training and validation loss, indicating effective learning and over-fitting over time. Both the training and validation accuracy curves exhibit an upward trend, which improves the performance and generalization capabilities of the model. This visualization underscores the robustness and efficiency of the model in achieving high classification accuracy while maintaining low error rates across epochs.

Measures	Proposed OptFuzNet	Fuzzy neural network	Fuzzy logic system	Neural network	SVM
Accuracy	94.2973	89.3805	69.2404	78.9308	86.8497
Precision	91.1826	86.8055	60.6866	83.9024	85.3837
Recall	95.58359	91.9117	70.2933	75.3284	85.2573
F-measure	94.20289	89.2857	69.1046	79.38461	87.0738
PPV	90.9909	86.7977	60.6866	83.9024	85.9695

Table 2: proposed tooth cavity detection performance

Table 2 presents the performance metrics of different methods for tooth cavity detection, including Proposed OptFuzNet, Fuzzy Neural Network, Fuzzy Logic System, Neural Network, and Support Vector Machine (SVM). The metrics evaluated are Accuracy, Precision, Recall, F-measure, and Positive Predictive Value (PPV).OptFuzNet achieved the highest accuracy at 94.2973%, significantly outperforming the Fuzzy Neural Network (78.9308%), Fuzzy Logic System (69.2404%), Neural Network (89.3805%), and SVM (86.8497%). In terms of precision, OptFuzNet also leads with 91.1826%, followed by the SVM at 87.3837%. The recall is highest for OptFuzNet at 95.58359%, indicating its robust ability to correctly identify true positives, in contrast to the lower recall rates of the other models. The F-measure, a balance between precision and recall, is also greatest for OptFuzNet at 94.20289%, reinforcing its overall effectiveness. Additionally, the Positive Predictive Value (PPV) for OptFuzNet is 90.9909%, again surpassing the other methods. These results underscore the significant performance improvement of OptFuzNet over existing techniques in tooth cavity detection.

The reason behind the excellent results is that, firstly, OptFuzNet combines the advantages of fuzzy logic and neural networks, using fuzzy rules to handle uncertainty and neural networks to learn complex patterns in data. This hybrid approach allows for more accurate and robust cavity detection. Additionally, the optimization process used in OptFuzNet, fine-tunes model parameters to increase performance, resulting in better results compared to other methods. Overall, the combination of fuzzy logic, neural networks, and optimization techniques in the Proposed OptFuzNet method enables it to achieve the maximum output in tooth cavity detection, showcasing its potential for clinical application in dental diagnostics.

4.3. Comparison with published work

In this section, we compare our findings with those reported in the existing literature to contextualize and validate our results. By exploring similarities and contrasts, we aim to contribute to the ongoing dialogue in the field and highlight the novelty or significance of our contributions.

Ref No	Authors	Algorithm	Accuracy	Precision	Recall	F-Score
[11]	Li et al. (2022)	DeepID	92.4	-	-	-
[12]	Lee et al. (2020)	DCNN techniques	83.8	-	75.0	-
[13]	Celik (2022)	Modern deep learning models	76.7	88	93	-
[14]	Tuzoff et al. (2019)	Deep CNNs	89	90.98	91.80	-
[15]	Al Kheraif, et al. (2019)	CNN and the hybrid graph-cut technique	91.2	89	92	94
[16]	Leite, et al. (2021)	Deep learning algorithms	86.9	90	92	93.1
[17]	Geetha, et al. (2020)	Backpropagation Neural Network (BPNN)	92.1	90.2	92.1	93.1
[18]	Almalki, et al. (2021)	Deep learning model YOLOv3	91.3	92	91.7	92.5
Proposed		OptFuzNet	94.2973	91.1826	95.58359	94.20289

Table 3: Comparison with already published work

Table 3 compares the performance of the proposed OptFuzNet model with several existing models in terms of accuracy, precision, recall, and F-score. The proposed OptFuzNet model achieves the highest accuracy of 94.30%, outperforming previous models such as the DeepID by Li et al. (92.4%) and the Deep CNNs by Tuzoff et al. (89%). It also demonstrates superior precision (91.18%), recall (95.58%), and F-score

(94.20%) compared to other methods. For instance, the Backpropagation Neural Network (BPNN) by Geetha et al. reports an accuracy of 92.1% with a precision of 90.2% and recall of 92.1%, while the CNN and hybrid graph-cut technique by Al Kheraif et al. show an accuracy of 91.2%, precision of 89%, and recall of 92%. The proposed OptFuzNet's balanced and high scores across all metrics indicate its robustness and effectiveness, making it a significant improvement over the existing methods.

6. Conclusion

The study on tooth cavity detection using the OptFuzNet and m-GAN for orthodontic diagnosis demonstrates significant advancements in the field. The purpose of OptFuzNet in this study is to leverage optimized fuzzy neural network techniques to accurately classify and detect tooth cavities, improving diagnostic precision. The m-GAN enhances data augmentation, ensuring the robustness and reliability of the diagnostic process in orthodontics. The proposed OptFuzNet model achieved an outstanding accuracy of 94.30%, surpassing various existing models such as DeepID, DCNN techniques, and YOLOv3. Additionally, it exhibited superior precision of 91.18%, recall of 95.58%, and F-score of 94.20%, indicating a high level of reliability and effectiveness in detecting tooth cavities. The integration of the optimized fuzzy neural network with m-GAN techniques contributes to enhanced feature extraction and improved classification performance, making it a robust tool for orthodontic diagnostics. These results highlight the potential of the proposed method to significantly improve the accuracy and efficiency of tooth cavity detection, ultimately contributing to better orthodontic care and diagnosis. Future work could explore real-time implementation and scalability of the model in clinical settings, and investigate its performance across different demographic groups and varying dental conditions to further validate its efficacy.

References

1. Gonano, C., Leitgeb, U., Sitzwohl, C., Ihra, G., Weinstabl, C. and Kettner, S.C., 2006. Spinal versus general anesthesia for orthopedic surgery: anesthesia drug and supply costs. Anesthesia & Analgesia, 102(2), pp.524-529.
2. Ko, E.W.C., Huang, C.S. and Chen, Y.R., 2009. Characteristics and corrective outcome of face asymmetry by orthognathic surgery. Journal of oral and maxillofacial surgery, 67(10), pp.2201-2209.
3. Pauwels, R., 2020. History of dental radiography: Evolution of 2D and 3D imaging modalities. Med Phys Int, 8(1), pp.235-77.
4. Zhao, Y., Li, P., Gao, C., Liu, Y., Chen, Q., Yang, F. and Meng, D., 2020. TSASNet: Tooth segmentation on dental panoramic X-ray images by Two-Stage Attention Segmentation Network. Knowledge-Based Systems, 206, p.106338.

5. Alalharith, D.M., Alharthi, H.M., Alghamdi, W.M., Alsenbel, Y.M., Aslam, N., Khan, I.U., Shahin, S.Y., Dianišková, S., Alhareky, M.S. and Barouch, K.K., 2020. A deep learning-based approach for the detection of early signs of gingivitis in orthodontic patients using faster region-based convolutional neural networks. International Journal of Environmental Research and Public Health, 17(22), p.8447.
6. Zafar, S. and Zachar, J.J., 2020. Evaluation of HoloHuman augmented reality application as a novel educational tool in dentistry. European Journal of Dental Education, 24(2), pp.259-265.
7. Lin, S.Y. and Chang, H.Y., 2021. Tooth numbering and condition recognition on dental panoramic radiograph images using CNNs. IEEE Access, 9, pp.166008-166026.
8. Zhang, X., Liang, Y., Li, W., Liu, C., Gu, D., Sun, W. and Miao, L., 2022. Development and evaluation of deep learning for screening dental caries from oral photographs. Oral diseases, 28(1), pp.173-181.
9. Fooladpanjeh, S., Dadrasi, A., Gharahbagh, A.A. and Parvaneh, V., 2021. Fuzzy neural network and coupled gene expression programming/multivariate non-linear regression approach on mechanical features of hydroxyapatite/graphene oxide/epoxy: Empirical and optimization study. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 235(23), pp.7169-7179.
10. Dehghani, M., Bektemyssova, G., Montazeri, Z., Shaikemelev, G., Malik, O.P. and Dhiman, G., 2023. Lyrebird optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems. Biomimetics, 8(6), p.507.
11. Li, S., Guo, Z., Lin, J. and Ying, S., 2022. Artificial intelligence for classifying and archiving orthodontic images. BioMed Research International, 2022.
12. Lee, K.S., Ryu, J.J., Jang, H.S., Lee, D.Y. and Jung, S.K., 2020. Deep convolutional neural networks based analysis of cephalometric radiographs for differential diagnosis of orthognathic surgery indications. Applied Sciences, 10(6), p.2124.
13. Celik, M.E., 2022. Deep learning based detection tool for impacted mandibular third molar teeth. Diagnostics, 12(4), p.942.
14. Tuzoff, D.V., Tuzova, L.N., Bornstein, M.M., Krasnov, A.S., Kharchenko, M.A., Nikolenko, S.I., Sveshnikov, M.M. and Bednenko, G.B., 2019. Tooth detection and numbering in panoramic radiographs using convolutional neural networks. Dentomaxillofacial Radiology, 48(4), p.20180051.
15. Al Kheraif, A.A., Wahba, A.A. and Fouad, H., 2019. Detection of dental diseases from radiographic 2d dental image using hybrid graph-cut technique and convolutional neural network. Measurement, 146, pp.333-342.
16. Leite, A.F., Gerven, A.V., Willems, H., Beznik, T., Lahoud, P., Gaêta-Araujo, H., Vranckx, M. and Jacobs, R., 2021. Artificial intelligence-driven novel tool for

tooth detection and segmentation on panoramic radiographs. Clinical oral investigations, 25, pp.2257-2267.

17. Geetha, V., Aprameya, K.S. and Hinduja, D.M., 2020. Dental caries diagnosis in digital radiographs using back-propagation neural network. Health Information Science and Systems, 8, pp.1-14.
18. Almalki, Y.E., Din, A.I., Ramzan, M., Irfan, M., Aamir, K.M., Almalki, A., Alotaibi, S., Alaglan, G., Alshamrani, H.A. and Rahman, S., 2022. Deep learning models for classification of dental diseases using orthopantomography X-ray OPG images. Sensors, 22(19), p.7370.
19. Gharakhanian, A. (2017). Generative Adversarial Networks–Hot Topic in Machine Learning.
20. Dehghani, M., Bektemyssova, G., Montazeri, Z., Shaikemelev, G., Malik, O. P., & Dhiman, G. (2023). Lyrebird optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems. Biomimetics, 8(6), 507.