AN INTELLIGENT FRAMEWORK FOR LUNG CANCER DETECTION IN PET IMAGES USING ADVANCED HYBRID GCN-BiLSTM AND U-NET WITH ENHANCED DOVE SWARM OPTIMIZATION ALGORITHM (EDSOA)

[1] M.Vaishnava Priya,
Research scholar, Department of Artificial intelligence and Data Science,
VET institute of Arts and science co education College,
Thindal, Erode.

[2] Dr.R.Tamilselvi,
Assistant professor & Head,
Department of Artificial intelligence and Data Science,
VET institute of Arts and science co education College,
Thindal,Erode.

ABSTRACT: Patients who have lung cancer (LC) have a significant mortality rate from this fatal disease. Accurate LC staging and early diagnosis may preserve patients' lives. An estimated five million deaths occur each year as a result of LC. For both men and women, these mortality rates place it among the world's leading causes of death. Despite the adoption of various image processing (IP), biomarker-based, and machine automation techniques, healthcare providers still struggle to make an accurate and timely diagnosis of LC. This research aims to find malignant lung nodules (LN) on the input lung image and classify LC severity. An intelligent framework for LC detection (LCD) in PET images is offered in the suggested method. The suggested approach integrates modern techniques for feature extraction (FE), classification, segmentation, pre-processing, dataset collection, and hyperparameter tuning (HPT). In order to improve image quality and minimise noise, bilateral filtering (BF) is used for pre-processing. High-quality inputs for additional analysis are guaranteed by BF. Then, Regions of Interest (ROIs) are precisely segmented using U-Net for segmentation. Tumor boundaries are accurately delineated at the pixel level by U-Net. FE then obtains the features, including the Gray Level Run Length Matrix (GLRLM), Gray Level Co-Occurrence Matrix (GLCM), and the suggested Local Vector Pattern (LVP). For LC classification, a hybrid framework that combines attention mechanisms, Bidirectional (LSTM) Long Short-Term Memory (BiLSTM) units, and Graph Convolutional Networks (GCNs) is suggested. A comprehensive spatial and temporal FE are facilitated by this integration. The Enhanced Dove Swarm Optimization Algorithm (EDSOA) is introduced hyperparameter tuning for improve classification accuracy. Evaluated on the PET dataset, the framework EDSOA-HGNN-BiLSTM demonstrates robust accuracy in lung cancer detection, outperforming traditional methods. Accuracy (A), Recall (R) (or Sensitivity (S)), Precision (P), and F-Score (or F-measure) (F1) were used to evaluate the classifiers. Positron Emission Tomography (PET)/Computed Tomography (CT) images collected from publicly available medical images (MI) of cancer from The Cancer Imaging Archive (TCIA) are used to evaluate the applicability of the suggested classifier and compare it with other current methods. According to the study of the outcomes, the suggested model executes better than the other methods currently in use in terms of A.

INDEX TERMS: Lung Cancer (LC), Lung Nodules (LN) , Bilateral Filtering (BF), Regions Of Interest (ROIs), Gray Level Run Length Matrix (GLRLM), Gray Level Co-Occurrence Matrix (GLCM), Local Vector Pattern (LVP), Graph Convolutional Networks (GCNs), Bidirectional Long Short-Term Memory (BiLSTM), Enhanced Dove Swarm Optimization Algorithm (EDSOA), Positron Emission Tomography (PET), The Cancer Imaging Archive (TCIA) and classification.

1. INTRODUCTION

The characteristic of the condition known as cancer is the uncontrollable aberrant growth of bodily cells. One of the most significant illnesses that have ever threatened human health is cancer. In addition to the 18.1 million new cases of cancer worldwide, case

count, almost 9.6 million will be fatal, according to a 2018 study [1].

The growth and expansion of aberrant cells throughout the body that causes cancer. It is currently uncertain what specifically caused this condition. It might be genetic and related to the immune system, or it can be brought on by bad habits like drinking or smoking. Both the incidence of cancer worldwide and the rate of technical development are increasing at the moment. According to medical science, there are over 100 different kinds of cancer [2]. With a rate of 13%, LC is the most prevalent kind of cancer worldwide. LC is one of the most prevalent malignancies in the United States, causing around 225,000 cases, 150,000 deaths, and $12 billion in medical expenses each year [3]. LC is among the worst malignancies; in the United States, just 17% of patients receive a diagnosis and live for 5 years afterward; in developing nations, the survival rate is even lower. Although there are currently less signs in the initial stages of the condition and it is also more difficult to identify, the probability of survival is considerably increased by an LC diagnosis.

To create images that was as precise, useful, and efficient as feasible, contemporary optimisation techniques and IP strategies were required. The technology that has been suggested may help doctors for effectively understanding the internal anatomy and identify LN early. The basic kinds of cancer can be treated early. Compared to healthy cells, malignant cells in the body proliferate more quickly. This will lead to lumps turning into tumours or nodules. An Advanced 3D-Convolution (NN) Neural Network (3DCNN) with Recurrent NN (RNN) is used to analyse and detect cancer cells in order to develop an accurate classifier using methods from DL (Deep Learning) and CV (Computer Vision) [4]. More early detection and better survival could be made possible with a more rapid and cost-effective LC classifier. Computer-Aided Diagnosis (CAD) system that can determine whether a patient has LC based on chest CT scans as input. A CT scan image of the patient with LC is visualized in Figure 1.

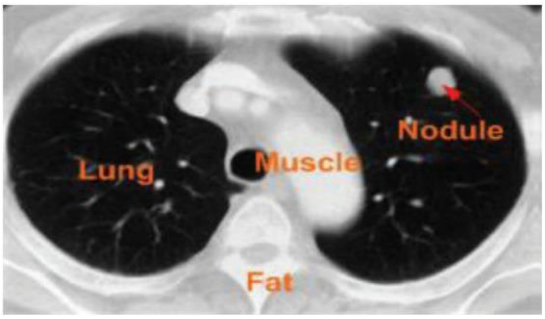

FIGURE 1. IMAGE OF THE AFFECTED PERSON WITH LC FROM CT SCANS

Through the blood vessels (BV), LC cells may spread to various parts of the body. Metastasis or additional LC are terms used to describe this type of cancer. Studies have shown that LC is among the deadliest types of cancer [5]. Every 30 seconds, the LC sufferer passes away somewhere on World. After being diagnosed with LC, the patient has a 20% chance of surviving [6]. Smoking women are more likely to acquire LC than men, according to research. Women had the highest LC rate without smoking. There are two forms of LC: small cell lung cancer (SCLC) and non-SCLC.

Early disease detection is achieved in medical science with the application of CAD technology. Physicians and radiologists employed the CAD approaches to diagnose a wide range of disorders [7]. It is usually difficult to identify several diseases in their early stages, such as prostate, skin, and LC. As a result, early cancer detection is facilitated by the CAD system.

Although DL methods for LC detection and prediction have a lot of clinical value potential, their development and application present a number of difficulties. DL approaches require pre-processing. By removing irrelevant, redundant, or noisy features, the feature space is made reduced by this pre-processing, which additionally assists to identify important characteristics. Finding a suitable representation of the data from all features is one of the main problems in machine learning (ML) tasks. Selection of the best solution is optimisation variety of possible solutions because of the problems in these approaches. The fundamental goal of Feature Selection (FS) strategies is to search across feature subsets and identify the best subset while preserving classification accuracy. It is essential to give the health sector more attention and to create novel early cancer detection methods in order to lower the cancer death rate. Segmentation accuracy is an important consideration when using LC segmentation data. When segmenting images manually, radiologists may provide erroneous results due to inter-observer variability and inconsistent procedures. The tool utilized for classifying patterns or

assesses the extent to which all subset predicts the class output or pattern is called a classification function.
Artificial Intelligence (AI) to the rapid growth of analytics tools and the growing availability of healthcare data, it is revolutionizing the field of healthcare. Like other Artificial NN (ANNs), CNN is made up of neurons that train to maximise themselves [8]. The foundation of ANNs is the concept that each neuron will keep receiving input and performing an action (e.g. producing a scalar product that complies with a Non-Linear Function (NLF)). CNNs are hierarchical, with deeper layers accumulating more complex information and early layers extracting simpler features [9].

By combining GCN with BiLSTM, the research aims to present a novel method known as GCN-BiLSTM, which uses PET scan images to accurately and early detect LC. To improve image quality and minimise noise, BF is used for noise removal. High-quality inputs for additional analysis are guaranteed by this BF. U-Net is used for accurate ROI segmentation. Tumor boundaries are accurately delineated at the pixel level by this U-Net. To improve diagnostic A, the method extracts features from images using LVP. Classification stage, GCNs-BiLSTM units for enabling comprehensive spatial and temporal FE. EDSOA is employed for HPT, optimizing model performance. A, R (or) S, P, and F-Score were the final criteria used to evaluate classifiers. This study's objective is to ascertain the viability of the suggested hybrid DL methodology for the prompt identification of LC and evaluate the results using standard statistical techniques. Evaluated on the PET dataset, the framework demonstrates robust accuracy in LC Detection (LCD).

The remainder of the study is organised in the following way: A review of LCD settings developing models for identifying and classifying cancer cells are offered in Section 2. Section 3 describes the suggested method, called GCNs-BiLSTM. Discuss the application analysis and use particular metrics to assess the suggested framework's attainment in Section 4. In Section 5, summarise the work and make recommendations for future research topics are covered.

2. LITERATURE REVIEW

This section lists and analyses the various LCD models that use ML and DL models. The investigation identifies the core problems with cancer Classification and detection in order to create a new framework.

For LCD, LeNet, AlexNet, VGG-16, CNN, and minimum Redundancy Maximum Relevance (mRMR) were all suggested by Toğaçar et al. [10]. The study employed a publicly available dataset of CT images. CNNs were used in the experiment for both FE and classification. The dataset is subjected to image augmentation methods during model training, including filling, zooming, cutting, and horizontal rotating, in order to improve the classification accuracy rate. The DL training procedure looks into the optimisation methods of RMSprop, ADAM, and Stochastic Gradient Descent (SGD). Because of the remarkable efficacy of the AlexNet model, the features obtained from the last fully-connected (FC) layer were independently used as input to the k-Nearest Neighbour (kNN), Decision Tree (DT), Support Vector Machine (SVM), Linear Regression (LR), Linear Discriminant Analysis (LDA), and Softmax classifiers. A k-NN classifier using AlexNet were used to get the highest classification accuracy. The mRMR FS method was then used to choose the best features from the deep feature set. Using CT scans of the chest, the suggested model provides a reliable diagnosis for LCD. The AlexNet was used with Python and MATLAB (R2017b). A, S, Specificity (Sp), P, and F-score measures are used to evaluate the models' performances.

For CT images, Naseer et al. produced a modified AlexNet and SVM-based automated nodule identification technique for LCDs [11]. The suggested structure consists of two FC layers for FE, three pooling layers (PL), and seven convolutional layers (CL). It classifies nodules as benign or malignant using the SVM classifier. The LN analysis 2016 benchmark dataset, which is openly accessible, is used in the experimental investigation (LUNA16). The maximum A, S, and Sp have been attained using the suggested model. A comparison between the suggested LungNet-SVM model and the state-of-the-art (SOTA) techniques has been carried out for the classification of LC. According to the test findings, the suggested LungNet-SVM model performed remarkably well in terms of accuracy on a LUNA16 dataset. The publicly accessible LUNA16 dataset has been used as a benchmark for training and validating the suggested

LungNet-SVM model. To validate the overall performance, A, Miss Classification rate, S, and Sp are among the statistical performance assessment criteria that are employed.

Utilising Low-Dose CT (LDCTs) from the National Lung Screening Trial (NLST), Mikhael et al. [12] presented a model named Sybil. On a radiological reading station, Sybil might run in the background in real time. Sybil doesn't need clinical information or radiologist comments, and it simply needs one LDCT. Three separate datasets were used to validate Sybil. 8,821 LDCTs from Massachusetts General Hospital (MGH), 12,280 LDCTs from Chang Gung Memorial Hospital (CGMH), this included nonsmokers in addition to the individuals with a range of smoking histories, and 6,282 LDCTs from NLST participants were held out.To further facilitate individualised screening, Sybil can reliably forecast a person's future LC risk from a single LDCT scan. To fully comprehend the clinical uses of Sybil, more research is needed. The annotations and model are openly accessible. To assess performance, the Area Under the Curves (AUC) metric is employed.

For LCD, LR, Multiple LR (MLR), LDA, Quadratic Discriminant Analysis (QDA), KNN, Random Forests (RF), and SVM were developed by Hasan and Al Kabir [13]. CT scans are frequently used to find LC. Thousands of high-resolution lung scans will be used to create algorithms that reliably identify whether or not the lungs are malignant. In comparison to the current technologies, the suggested approach offers improved results. This enhanced outcomes of the suggested method would help radiologists diagnose cancer accurately and early. The technique has shown good results were examined using 198 slices of CT imaging from the Kaggle dataset that showed various stages of cancer. In this dataset, the proposed technique is 72.2% accurate.

A Dual-Tree Complex Wavelet Transform (DTCWT) and DT for LCD were suggested by Dafni Rose et al. [14]. In the initial pre-processing step, denoising is done using the Average Filter (AF), and segmentation is done using region development. In the suggested model, a technique for image fusion (IF) is presented using DTCWT. The results of the fusion are significantly better when the DTCWT and DWT are correlated. Segmentation makes cancer cell position and outline more exact. Predicting LC using DT. Region development is the proposed system segmentation approach. Segmentation is done on fused, CT, and PET images. Additionally, it increases the similarity value, entropy, and PSNR accuracy. Mean Squared Value (MSV), Normalised Cross Correlation (NCC), Peak Signal-to-Noise Ratio (PSNR), and entropy are all evaluated.
An Improved Profuse Clustering Technique (IPCT) and DL were created by Shakeel et al. [15]. Instantaneously trained NN (DINN) to reduce misclassification and improve lung image quality and LC diagnosis. The TCIA dataset's lung CT images are cleaned up of noise using the weighted mean (HE) histogram equalisation technique. The weighted mean (HE) technique efficiently lowers noise from the image. It also enhances its quality by using IPCT to segment the affected area. The impacted area yields a variety of spectral features. In order to forecast LC, these are analysed using DINN. At last, MATLAB-based simulation results are used to analyse the system's efficiency. The technique provides 0.038 classifying errors and 98.42% A. A, R, P, Sp, and f-score were used to assess the effectiveness of the IPCT approach.

For the LCD algorithm, ANN, Haar Wavelet Transforms (HWT), and Median Filters (MF) was suggested by Vas and Dessai [16]. Data collection, pre-processing, segmentation, FE, and classification are the suggested work steps. The first images were taken at the Manipal Hospital in Goa, the V.M. Salgaocar Hospital, and SMRC. The hospital's database provided the lung CT scans. Prior to applying Median Filters (MF), which are essentially used to remove the salt and pepper noise (SPN) from the images, the first step in pre-processing is cropping the image to remove any unwanted parts. In addition to lowering computing complexity and storage problems, converting the images to binary is a prerequisite for lung morphological segmentation. Significant data items that are used as classifier inputs are extracted with the aid of FE. Prior to applying HWT to these images, the image must first be resized into three distinct resolutions. Lung ROI is segmented using an LCD approach that uses mathematical morphological procedures. Haralick features are then retrieved from this data and utilised by ANNs to classify malignancy. To help with the staging of patients with LC, CNN was suggested by El Hamdi et al. [17]. References were taken from the Tumour, Node, and Metastasis (TNM) staging system and the classification of histologic subtypes. The most relevant characteristics are extracted from PET/CT data using the VGG 16 network. To specify the classification of Nodal (N),

Tumor (T), and histologic subtypes, the acquired features are subsequently sent to a three-branch classifier. The CNN model performs well in TN staging and histological categorisation, according to experimental results. Tested on the Lung-PET-CT-Dx dataset, the suggested architecture successfully diagnosed the tumor size with an A of 0.94 and an AUC of 0.97. With an A of 0.98, it has also shown excellent results for N staging. Furthermore, in terms of histological classification, the suggested method outperformed SOTA methods with an A.

To enhance the effectiveness and precision in detecting cancer, a new model for cancer identification and classification is required, according to the report. In order to identify cancer and categorise the disease's severity, this paper develops a hybrid NN GCN-BiLSTM based on cancer cell identification and categorisation. Addressing the difficulties of correctly detecting and diagnosing LC at an early stage and enhancing the precision of differentiating between benign (B) and malignant (M) tumors are its main motivations.

3. PROPOSED METHODOLOGY

By combining modern techniques for dataset collection, pre-processing, segmentation, FE, classification, and HPT, this work suggests an intelligent framework for LCD in PET images. To improve image quality and efficiently eliminate noise, a BF is first applied to PET images. More accurate analysis is ensured by BF. After that, segmentation is done using the U-Net technique. ROIs and tumor boundaries can be accurately delineated at the pixel level due to the U-Net method. The features, such as GLRLM, GLCM, and the suggested LVP, are then obtained for FE. A thorough examination of the temporal and spatial features of PET scans is then made possible by LC classification, a hybrid framework that combines attention mechanisms with the GCN-BiLSTM model. HPT is carried out utilising the EDSOA to optimise model parameters for increased accuracy and resilience in order to improve framework's efficiency. To assess and compare the appropriateness of the suggested hybridisation approaches classifier approaches to existing methods, PET/CT images taken from TCIA medical images (MI) of cancer that are available for public download were used. The suggested work's general procedure is depicted in figure 2.

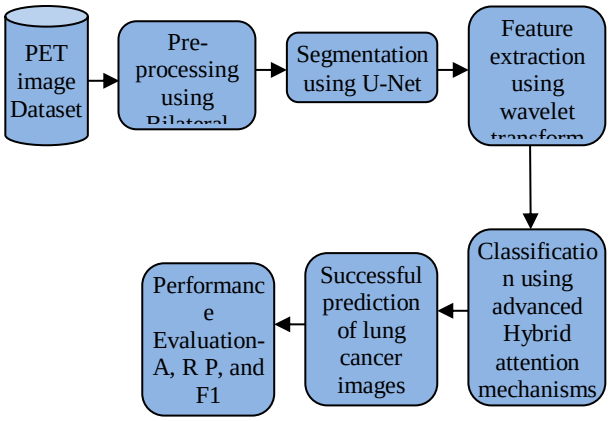

FIGURE 2. THE GENERAL PROCEDURE OF THE SUGGESTED METHOD

3.1. Dataset Collection

It is possible for the public access to download the LC dataset collected from TCIA MI of cancer [18]. LC patients' CT and PET-CT DICOM images with XML annotations files make up the set that use bounding boxes to indicate the tumor's position. The images were obtained after the fact from patients who had PET/CT and a routine lung biopsy and were suspected of having LC. Based on a tissue histopathology diagnosis, the subjects were categorised. Squamous, small, large, and malignant were the types diagnoses for patients whose names or IDs contained the letter "A," "B," "E," and "G." The lung (window width, 1,400 HU; level, –700 HU) and mediastinum (window width, 350 HU; level, 40 HU) parameters were used to analyse the images. Lung setting and 2 mm slice-thick reconstructions were created. 0.625 mm to 5 mm is the range of the CT slice interval. Plain, contrast, and 3D reconstruction are all available in scanning mode.

3.2. Pre-processing

Images are pre-processed to make them useable and intelligible. It enables one to improve particular attributes necessary for CV applications and remove undesired distortions. Images can be cleaned up of undesirable noise by using techniques like filtering, blurring, and smoothing. For noise reduction (NR), the BF approach is employed.

3.2.1. BF

One non-linear (NL) IP method is BF. Images are smoothed by this BF and noise is decreased. The edges are also preserved by this BF. The value of intensity of each pixel are replaced by neighbouring pixel intensity

normalized average. A Gaussian Distribution (GD) can serve as the basis for weight allocation [19]. Importantly, the weights are dependent on both (e.g., range inequalities, such as color intensity, depth distance, etc.) the radiation differences and the Euclidean distance (ED) of pixels. Sharp edges are thus maintained. Equation (1) defines the BF in the following way:

$$I^{filtered}(x) = \frac{1}{W_p}\sum_{x_i\in\Omega} I(x_i) f_r(\|I(x_i) - I(x)\|)\, g_s(\|x_i - x\|) \quad (1)$$

Equation (2) defines the normalisation term, W_p:

$$W_p = \sum_{x_i\in\Omega} f_r(\|I(x_i) - I(x)\|)\, g_s(\|x_i - x\|) \quad (2)$$

The filtered image is represented as $I^{filtered}$

This process is required for the original input image, *I*. *x* represents filtering the properties of the current pixel. $x_i \in \Omega$ is another pixel since Ω is the window centred in x.

For smoothing variations in intensity, the Gaussian function can be used as the range kernel f_r.

A Gaussian function may be used as the spatial (or domain) kernel, g_s , to smooth coordinate discrepancies.

The spatial closeness (g_s) and intensity differential (f_r) are used to determine the weight W_p. Each of the neighbors of the pixel at (i,j) in the image, which is at (k,l), must be used to denoize it. Following that, the weight for pixel (k,l) to denoise pixel (i,j) is given by equation (3), assuming that the f_r and g_s are Gaussian kernels.

$$w(i,j,k,l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2} - \frac{\|I(i,j) - I(k,l)\|^2}{2\sigma_r^2}\right) \quad (3)$$

Here, the smoothing parameters are σ_d and σ_r. The intensity of the (i, j) and (k,l) pixels is denoted by I(i, j)and I(k, l), respectively.

Once the weights have been determined, normalise them using equation (4) in the following way:

$$I_D(i,j) = \frac{\sum_{k,l} I(k,l) w(i,j,k,l)}{\sum_{k,l} w(i,j,k,l)} \quad (4)$$

Here, I_D is used for calculating the denoised intensity of pixel (i,j).

3.3. Segmentation

One of the most important CV task is image segmentation. For a more thorough study, this image segmentation separates an image's boundaries, objects, and structures. In order for computers to observe and comprehend visual data in a way that is similar to how humans view and comprehend it, image segmentation is crucial to the extraction of meaningful information from images. A key method in (DIP) digital IP and CV is image segmentation. It entails dividing a digital image into numerous components (regions or objects) in order to simplify and analyse the image by breaking it down into relevant components. By concentrating on certain ROI, this increases the efficiency of the IP. The steps in a typical image segmentation task are as follows:

1. Based on common features, such as colour, texture, or intensity, pixels are assembled in an image.
2. Gives each pixel a label denoting its affiliation with a particular object or segment.
3. A segmented image, which is frequently displayed as a mask or overlay emphasising the various segments, is the end result.

A collection of segments that together include the full image or a collection of contours taken from the image are the end product of image segmentation. In terms of some attribute or calculated feature, like colour, intensity, or texture, every pixel in a region is comparable. Adjacent regions differ significantly when it comes to the same attributes.

In order to automatically split an image into various segments and extract features for precise analysis and segmentation tasks, DL image segmentation models are a potent technique that makes use of the NN architecture. U-Net: This model effectively segments MI using a U-Shaped network. This methodology offers accurate segmentation and is highly effective when working with tiny amounts of data.

3.3.1. U-Net

Due to its revolutionary performance in image segmentation tasks, U-Net is a remarkable DL architecture that has become extremely popular. One CNN designed for image segmentation is called U-Net. The network's foundation is a Fully CNN, whose architecture was expanded and altered to produce more accurate segmentation and operate with less training images.

Description

The so-called "fully convolutional NN" is the foundation of the U-Net architecture. Enhancing output resolution by adding successive layers with upsampling operators (instead of pooling) in a contracting network is the fundamental concept of this network architecture, which is probably similar to U-Net. These layers assist in learning a more accurate output for the convolutional layer (CL) that follows. In the upsampling, the many feature channels section is one of U-Net's primary characteristics. Contextual information can be sent to higher resolution layers via U-Net. In replicating the contracting route in the expanding path, this design produces a "U-shaped" architecture. The network only uses each convolution's valid section when there are no FC layers. Extrapolating lacking information from the image's border predicts pixels [20].

Network Architecture

The network's u-shaped architecture is a result of its expansive and shrinking paths. While the expanding path returns the image to its initial size and creates the segmentation map, the contracting path collects features from the input image and lowers its resolution. A Rectified Linear Unit (ReLU) and the contracting route is a typical convolutional network as each convolution is followed by a max pooling operation. While feature information increases during the contraction, spatial information decreases. Multiple high-resolution feature up-convolutions and combinations, contracted route are used in the expanding pathway for combining the characteristics and spatial data. In order to get high accuracy in segmentation tasks, U-net may learn both global and local features by combining the two paths [21].

History

Researchers from the University of Freiburg in Germany created U-net in 2015. Originally intended for biological image segmentation, Thomas Brox, Philipp Fischer, and Olaf Ronneberger founded U-Net in 2015. [22].

Applications

The ability of U-net accepts several data types, including colour, grayscale, and multi-channel images, is one of its advantages. Input of different sizes and aspect ratios can also be handled by it. U-Net has served as a backhold for magnetic resonance imaging (MRI), CT scans, and intricate microscope images in the context of biomedical applications. Furthermore, multi-channel images or even 3D volumes can be handled by adapting U-Net.

3.4. FE

By merging or altering the original features, feature extraction creates new features. In order to process raw data and preserve the original data set's information, it is necessary to convert it into numerical features. To limit the count of information, FE can be helpful in this situation for both picking specific variables and combining some related variables. Finding and revealing unique structures inside an image is the task of image FE. The qualities that set one image apart from another are called features. From basic corners and edges to more intricate textures and shapes, these can vary widely. When working with images that feature complex or repeated patterns, texture analysis employing techniques like GLCM, GLRLM, and LVP is crucial for tasks like classification, segmentation, and recognition [23].

3.4.1. GLCM

A statistical technique for capturing the spatial relationships between pixels in an image is called GLCM. It calculates how frequently pairs of pixel values appear in an image at particular orientations and distances. Through the analysis of these pixel pairs, GLCM is able to extract texture features that reveal information about the image's texture attributes, including contrast, correlation, energy, and homogeneity. Textures having distinct patterns and structures are especially well-suited for GLCM analysis [24].

The relevance of each pixel is calculated using a 2D histogram and divided by the image's d_{occ} distance. An image of $\times M$ with L gray levels is denoted as $I(i,j)$. Then, $I(i_1,j_1)$ and $I(i_2,j_2)$ be two pixels with x_1 and x_2 grey level intensities, by assumption . The direction of the connecting line, θ, is same to $arctan\left(\frac{\Delta j}{\Delta i}\right)$. Here, $\Delta i = i_2 - i_1$ is taken in the i direction and $\Delta j = j_2 - j_1$ in the j direction. Equation (5) illustrates the generalised co-occurrence matrix $C_{\theta,d}$.

$$C_{\theta,d}(x_1,x_2) = \left(Num\left\{\frac{\left((i_1,j_1),(i_2,j_2)\right)\varepsilon(N\times M)}{A}\right\}\right)/K \quad (5)$$

In this framework, a condition, such as $d_{occ}sin(\theta), \Delta j = d_{occ}cos(\theta)$, defines A. K indicates the total count (5), while Num displays the values of the co-occurrence matrix's components. For estimations, $d_{occ} = 1,2$ and $\theta = 0^{\circ}$, $45^{\circ}, 90^{\circ}, 135^{\circ}$ have often been used.

3.4.2. GLRLM

For a particular ROI, grey level (GL) value pairs and their run periods are the GLRLM statistic. The term "GL run" (GLR) refers to a group of pixels with the

same GL value that are distributed inside the ROI in a successive and collinear manner in certain directions. The amount of pixels in the collection determines its length. Thus, such a set is characterised by a GL value and its length of a GLR. In a ROI for a certain direction, a GLRLM logs the occurrence of each conceivable combination of GL values and GLR. Essentially, a 2D histogram is represented by a matrix in GLRLM [25].

Traditionally, the matrix's rows and columns are represented by the keys for GL values and GLR. As a result, the number of combinations with a GL value of *i* and a run length of *j* is indicated by the (*i*, *j*)-th entry in the matrix. In practice, four primary directions: horizontal (0°), anti-diagonal (45°), vertical (90°), and diagonal (135°) are typically taken into account. P is typically used to indicate a GLRLM. The GLRLM's (*i*, *j*)-th entry is P_{ij}.

Additionally, the actual ROI run durations is indicated by N_r. The set of unique GL that are present in the ROI is N_g. The ROI's total number of pixels is N. The following equation (6), it serves as an easy way to confirm the correctness of a GLRLM.

$$N = \sum_{i \in N_g} \sum_{j \in N_r} j P_{i,j} \tag{5}$$

The GLCM technique is characterised by the incredible association between the Haralick characteristics and the matrix's enormous dimensionality; translation, rotation, affine transformation, and perspective transformation do not alter the performance of some of them. A LVP was designed to address these problems.

3.4.3. LVP

The LVP vector representation is computed by computing the different distances in different directions. The 1D orientation and structural details of the cancer texture are represented by the LVP vector. The LVP strengthens the structure of micropatterns by encoding different pairwise vector directions as a descriptor based on the vector representation. The 1D direction and structural information of the local texture may be obtained by calculating the values between the referenced pixel (RP) and neighboring pixels that are spaced differently from different directions. Two-dimensional micropattern structures with varied RP vector pairing directions and its neighbouring areas are provided by the LVP descriptor based on the vector representation [26].

A local sub-region I vector's direction value is $V_{\beta,D}(G_c)$.The red-marked RP in I is represented by G_c. For the variation direction, let β be the index angle . Let D be the separation along the β direction between the RP and its neighbouring pixels. To provide an example, the colors yellow, green, and blue, respectively, represent the distances D = 1, D = 2, and D = 3. A vector's direction value at the RP G_c is specified by equation (6).

$$V_{\beta,D}(G_c) = (I(G_{\beta,D}) - I(G_c)) \tag{6}$$

Consider the vector as the LDP's first-order derivative values at *D*=1. When D>1, 1D direction information's implicit attributes may be obtained.

The $LVP_{P,R,\beta}(G_c)$ is encoded in the β direction of the vector at G_c by equation (7).

$$\begin{aligned} &LVP_{P,R,\beta}(G_c) \\ &= \{s_5(V_{\beta,D}(G_{1,R}), V_{\beta} \\ &+ 45°,D(G_{1,R}), V_{\beta,D}(G_c), V_{\beta} \\ &+ 45°,D(G_c)), s_5(V_{\beta,D}(G_{2,R}), V_{\beta} \\ &+ 45°,D(G_{2,R}), V_{\beta,D}(G_c), V_{\beta} \\ &+ 45°,D(G_c)), \ldots, s_5(V_{\beta,D}(G_{p,R}), V_{\beta} \\ &+ 45°,D(G_{p,R}), V_{\beta,D}(G_c), V_{\beta+45°,D}(G_c))\}|_{p} \\ &= 1,2,\ldots,P; R = 1 \end{aligned} \tag{7}$$

$s_5(.,,)$ modifies the β-direction value of neighborhoods using a transform ratio that is computed using a pairwise direction (PWD) of the RP's vector $(\beta + 45°)$ -direction. Micropatterns' binary pattern is then identified by comparing this value to the initial $(\beta + 45°)$ -direction value of neighbourhoods. Thus, the following equation (8) can be used to officially define $s_5(\cdot, \cdot)$.

$$\begin{aligned} &s_5\left(V_{\beta,D}(G_{p,R}), V_{\beta+45°,D}(G_{p,R}), V_{\beta,D}(G_c), V_{\beta} \right. \\ &\left. + 45°,D(G_c)\right) \\ &= \begin{cases} 1, if V_{\beta+45°,D}(G_{p,R}) - \left(\frac{V_{\beta+45°,D}(G_c)}{V_{\beta,D}(G_c)} \times V_{\beta,L}\right. \\ 0, else \end{cases} \end{aligned} \tag{8}$$

Equation (9) defines the LVP, $LVP_{P,R}(G_c)$, at the referred pixel G_c. Four 8-bit binary LVP patterns are concatenated to produce this.

$$LVP_{P,R}(G_c) = \{LVP_{P,R,\beta}(G_c) | \beta = 0°, 45°, 90°, 135°\} \quad (9)$$

In particular, dynamics linear decision function weight vector is produced using the transform ratio that is produced by each PWD of the RP vector.

3.5. Classification

The process of classifying involves grouping or classifying things according to predetermined standards. Groups of pixels or vectors within an image are categorised and labelled using certain rules in the process of image classification. The categorization control might employ spectral or textural properties 'Unsupervised' and 'supervised' are two general classification techniques. HGCN-BiLSTM is a new architecture that combines the robustness of hybrid GCN and Bi-LSTM units. Comprehensive analysis of both spatial and temporal features in PET images is made possible by HGCN-BiLSTM. The EDSOA is used for HPT, which optimises model parameters for increased accuracy and resilience in order to improve model performance.

3.5.1. Hybrid GCN-BiLSTM (HGCN-BiLSTM) GCN

In recent times, GCN have become more and more common in a variety of fields. The major development in the field of graph analysis is made possible by GCN ability to model the dependencies among nodes in a network. One kind of NN that directly manipulates the graph structure is the GCN. Node classification is a common application for GCN. Since each node in the network is given a label, the goal is to predict the labels of the nodes in the graph without the use of ground truth.The neighborhood of every node is a discrete computation graph Utilize data propagation throughout the graph to calculate embeddings for nodes. All of the data transferred from a node's neighbors, who have been notified by their own neighbors , and it is effectively aggregated.

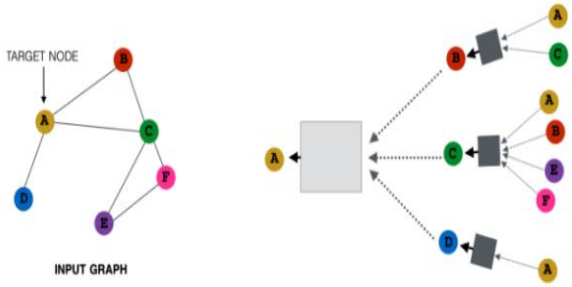

FIGURE 3. STRUCTURE OF INPUT GRAPH AND TARGET NODE IN GCN

The computation graph of node A depending on its neighborhood is presented on the right side of the graph in Figure 3, which depicts the input graph and the target node A. It is logical, A receives all of the messages from the nodes [B, C, D] in its neighborhood and handles them in some way. The nodes [B, C, D] then handle data from their neighbors. To understand the computational graph, look at the edge directions. The original node features of nodes [A,C] are transmitted to node BBB in this computational graph of depth 2, where they are altered before being passed back to A node. As, dynamic recursive programming was employed, node embeddings of every node at every tier of the CG are simultaneously computed and sent into the subsequent layer. Using this concept, the disease classification is detected.

The probability of diagnosing leaf l with disease *m* can then be estimated by calculating the degree of correspondence among any diseases and leaf image's embeddings. The Graph Encoder (GE) framework, a low-dimensional embedding at each node in any system, z ∈ Rd, is described at the beginning of this section [27].

The first step is to randomly characterize a disease, sign, or node for a graph $G = \{C, P\}$ as $v \in G$ in order to make the representation simple. Afterward, the embedding h_v^l of node v is computed at the *l*-th information Transmission Layer (TL) as follows equations (10-11),

$$h_{N_{(v)}}^l = AGGREGATE\left(\left\{h_{v'}^{l-1}, \forall v' \in N(v)\right\}\right) \quad (10)$$

$$h_v^l = \sigma\left(W^l.\left[h_v^{l-1}; h_{N_{(v)}}^l\right]\right) \quad (11)$$

In this case, node v's embedding at the previous layer can be demoted as h_v^{l-1}. W^l can be used to represent a low-dimensional embedding, z ∈ Rd, at each node in an arbitrary network The entire layer size is denoted by *L*. For instance, a real-valued dense Feature Vector (FV) can be used to initialise a leaf node. Thus, given the available picture demographics and features in the data, each digit in

h0*v* indicates the FV's actual number (e.g., age). The synergic representation that emerges when the embeddings of node v's neighbours at the (l−1)-th layer are aggregated using the aggregation function is represented as $h^l_{N(v)}$. The aggregator may be mean, max pooling, or another option if σ is a non-linear (NL) Activation Function (AF) (such as tanh). Use *mean*(.) by default to aggregate the data in the model created. Then proceed to the final embedding for every node at final layer L after a normalization stage as follows equation (12),

$$h_v = \frac{h_v^L}{\|h_v^L\|_2}, \qquad \forall_v \in \mathcal{G} \tag{12}$$

Notably, the models learnt for the symptoms nodes *p* in leaf graph *P* and concept network *C* share a same embedding space. In other words, every symptom *p* has similar embeddings in both networks, which effectively acts as a link among the disease and leaf nodes from graph that is distinct. As various node types: diseases, leaves, and symptoms, for example learn from 3 distinct embedding spaces, projected all node embeddings into one space, and then perform a non-linear activation to further align their contexts as follows equation (13),

$$z_v = \sigma(W h_v), \forall_v \in \mathcal{G} \tag{13}$$

Here, z_v is the node v final embedding and W is the weight of learnable projection. A knowledge graph's different edges may represent various node relationships and it is unable to handle edge data. Because GCNs may record intricate structural connections between nodes in a graph, they have attracted a lot of interest in a variety of fields. GCN enables effective feature propagation through local neighbourhoods by extending conventional convolution (Conv) procedures to graph-structured data. One common application of GCNs is node classification, where the goal is to infer the label of each node using features of its neighbouring nodes and the graph topology.

In the proposed framework, each node (representing either a disease, symptom, or image) is connected in a heterogeneous graph. G=(V,E)G = (V, E)G=(V,E) is the definition of the graph. Here, the set of edges is EEE and the set of nodes is VVV.

The GCN layer computes node representations using the normalized graph Laplacian as follows (equation 14):

$$\mathrm{H}^{(\mathrm{l}+1)} = \sigma\left(\tilde{\mathrm{D}}^{-1/2}\tilde{\mathrm{A}}\tilde{\mathrm{D}}^{-1/2}\mathrm{H}^{(\mathrm{l})}\mathrm{W}^{(\mathrm{l})}\right) \tag{14}$$

In this case, A stands for recurrent self-loops in the adjacency matrix. The matrix of degrees for A is D. The feature matrix at layer l is denoted by H. σ is the AF (such as ReLU), and the trainable weight matrix at layer l is W.

Each node starts with an initial feature vector X includes real-valued features, through two stacked GCN layers, the final node embedding is produced as in equation 15:

$$Z = H^{(L)} = \mathrm{GCN}(X, A) \tag{15}$$

The resulting node embeddings Z are then passed to a Bidirectional LSTM (Bi-LSTM) to model temporal or sequential dependencies. To improve performance, an attention mechanism is also applied after Bi-LSTM to focus on the most informative nodes or time steps during classification.

Bidirectional LSTM (Bi-LSTM)

To determine the Hidden Vector (HV) sequence, $h = (h1, \ldots, hT)$, and the Output Vector (OV) sequence, $y = (y1, \ldots, yT)$, a basic RNN receives an input sequence, $x = (x1, \ldots, xT)$, and iteratively solves the following equations (16–17) from t = 1 to T.

$$h_t = \mathcal{H}(W_{xh}h_t + W_{hh}h_{t-1} + b_h) \tag{16}$$

$$y_t = W_{hy}h_t + b_y \tag{17}$$

Here, the Hidden Layer (HL) function can be denoted as $\mathcal{H}$. The b terms are used to represent bias vectors. The hidden bias vector is denoted by b_h. For example, the input-hidden weight matrix is expressed as W_{xh}. The W stand for weight matrices.

The Sigmoid Function (SF) is often applied element wise to H. The LSTM framework, on the other hand, has proven to be more effective in identifying and utilizing long range context [28]. It performs this by storing data in particular memory cells. One memory cell of an LSTM. The following Composite Function (CF) implements H for the version of LSTM utilized as follows equations (18-22),

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{18}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{19}$$

$$c_t = f_t c_{t-1} + i_t tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (20)$$

$$O_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (21)$$

$$h_t = O_t \tanh(c_t) \quad (22)$$

The forget vector can be denoted as f, output vector can be denoted as o, cell activation vector can be denoted as c, and the input vector can be denoted as i. Here, σ represents the logistic SF.

The diagonally organized weight matrices from the cell to GV only input element m of each Gate Vector (GV). (e.g., W_{si})). Through the utilization of 2 separate HL for processing the data in both ways and then input the results back to the same (OL) Output Layer, bidirectional RNNs, or BRNNs, accomplish this.

To compute as equations (23-25) in a BRNN, the Forward Layer (FL) and Backward Layer (BL) iterate from t = 1 to T, respectively.

$$\vec{h}_t = \mathcal{H}(W_{x\vec{h}}x_t + W_{\vec{h}\vec{h}}\vec{h}_{t-1} + b_{\vec{n}}) \quad (23)$$

$$\overleftarrow{h}_t = \mathcal{H}(W_{x\overleftarrow{h}}x_t + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{t-1} + b_{\overleftarrow{n}}) \quad (24)$$

$$y_t = W_{\vec{h}y}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t + b_y \quad (25)$$

Bi-LSTM has access to long-range information in both input directions, is created by combining BRNNs with LSTM. Using deep structures to create increasingly complex models for acoustic information is a key component of the recent achievement of hybrid structures. By piling several RNN HL on top of one another, deep RNNs can be produced, with the input sequence for every layer being formed by the previous layer with an output series. Assuming that the identical HL function is applied HV sequences h n are determined from n = 1 to N and t = 1 to T for all N layers in the stack using equation (26).

$$h_t^n = \mathcal{H}(W_{h^{n-1}h^n}h_t^{n-1} + W_{h^n h^n}h_{t-1}^n + b_h^n) \quad (26)$$

Here $h^0 = x$ is defined. y_t is the network output as follows equation (27),

$$y_t = W_{h^N y}h_t^N + b_y \quad (27)$$

Deep Bi RNNs (DBi-RNNs) consists of the forward and backward sequences substituted for each hidden sequence h_n, $\vec{h}^n$and $\overleftarrow{h}^n$, and make certain that the FL and BL at the lower level provide input to each HL. Figure 4 displays the Bi-LSTM structure.

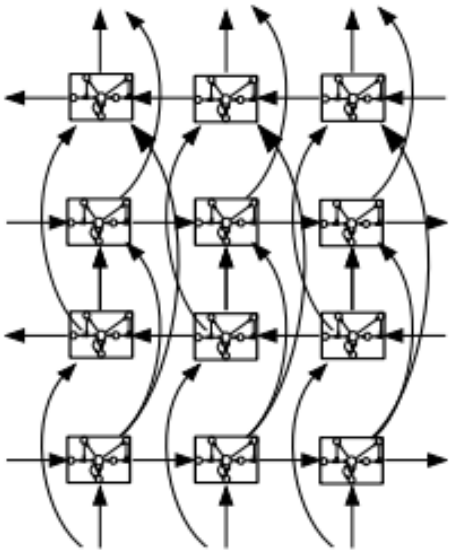

FIGURE 4. STRUCTURE OF BI-LSTM

While Bi-LSTM processes sequences bidirectionally and captures rich temporal information, not all-time steps contribute equally to the final prediction. Therefore, an attention mechanism is applied to the Bi-LSTM output to dynamically focus on the most informative hidden states.

After the Bi-LSTM layers produce their output sequences, the attention mechanism computes a weighted combination of hidden states (HS). These weights are learned during training and indicate how each time step affects output. The attention score αt for time step t is computed as in equation 28:

$$\alpha_t = \frac{\exp(\text{score}(h_t))}{\sum_{t'}\exp(\text{score}(h_{t'}))} \quad (28)$$

Here score(.) is a trainable function (e.g., dot product, additive attention) and h_t is the Bi-LSTM HS at time t.The final context vector c is calculated as (equation 29):

$$c = \sum_t \alpha_t \cdot h_t \quad (29)$$

This context vector summarizes the most relevant information from the sequence and is passed to the OL for final classification. The performance of the attention-enhanced Bi-LSTM is still influenced by the selection of hyperparameters like weights and biases,

even though it is effective at capturing important temporal patterns. Manual tuning or random search often leads to suboptimal results. To address this, the Enhanced Dove Swarm Optimization Algorithm (EDSOA) is employed to optimally adjust these parameters. This metaheuristic strategy mimics dove foraging behaviour and ensures better convergence toward global optima, thereby boosting classification accuracy.

3.5.2. Hyperparameter Tuning using EDSOA

EDSOA imitates doves' foraging behaviours [24]. When crumbs are scattered around in plazas, it is evident that doves search for them. Not all doves are satisfied, but maybe some are.

Disappointed doves are observed flying forward to areas where additional crumbs may be found. Over time, it becomes clear that areas with the most crumbs may be occupied by consuming doves.
The dove's foraging behaviour has served as the model for the optimal approach. The optimisation objective function is denoted by $f(W)$. The number of crumbs in each classed image pattern (W) in a dataset is $f(W)$ crumbs. Each classified image pattern is thought to include crumbs. The best option is the area with the largest number of crumbs. Hybrid GCN-Bi-LSTM algorithm parameters weight (W) and bias (b) tuning using EDSOA for improve classification accuracy.
Step 1: Deploy the doves on the LCD space after deciding regarding their number. After determining how many doves are needed in solution spaces. Assuming N stands for predetermined dove counts, they are placed randomly, but spreading them out evenly throughout a rectangle area is recommended.

Step 2: For dove d, $d = 1,\ldots,N$, take the satiety level and epochs (e = 0) (s_d^e). There are two methods to initialise the node position vector $W_d \subset R^M$ of dove d. Starting W_d at random and centred on the LCD is the simplest way. The alternative is to initialize a method that uses a lattice. The actions are depicted as follows,

There are two efficient weight initialisation techniques that can speed up the method used to train a node map with topological order. $[lo_1, up_1], \ldots, [lo_M, up_M]]$ is the parameter space's smallest hyper-rectangle. It includes each value that is permitted for each parameter. The a-dimension lower and upper bounds in the LCD space are denoted by the letters lo_a and up_a. There are some steps given below:
Phase 2-1. Using equations (30–33) to initialise the network's 4 edge cells,

$$we_{1,1} = (lo_1, lo_2, \ldots., lo_M)^T \tag{30}$$

$$we_{A,B} = (up_1, up_2, \ldots., up_M)^T \tag{31}$$

$$we_{1,B} = \left(lo_1, lo_2, \ldots., lo_{\lfloor \frac{M}{2} \rfloor}, up_{\lfloor \frac{M}{2} \rfloor + 1}, \ldots, up_M\right)^T \tag{32}$$

$$we_{A,1} = \left(up_1, up_2, \ldots., up_{\lfloor \frac{M}{2} \rfloor}, lo_{\lfloor \frac{M}{2} \rfloor + 1}, \ldots, lo_M\right)^T \tag{33}$$

Phase 2-2. Using equations (34–38), the 4 edge cells are initialised.

$$we_{1,j} = \frac{we_{1,B} - we_{1,1}}{B-1}(j-1) + we_{1,1}(W,b) * Ew_{1,1} + for\ j = 2, \ldots, B-1 \tag{34}$$

$$we_{A,j} = \frac{we_{A,B} - we_{A,1}}{B-1}(j-1) + we_{A,1}(W,b) * Ew_{A,1}\ for\ j = 2, \ldots, B-1 \tag{35}$$

$$we_{i,1} = \frac{we_{A,1} - we_{1,1}}{A-1}(i-1) + we_{1,1}\ (W,b) * Ew_{1,1}\ for\ i = 2, \ldots, A-1 \tag{36}$$

$$we_{i,B} = \frac{we_{A,B} - we_{1,B}}{A-1}(i-1) + we_{1,B}(W,b) * Ew_{1,B}\ for\ i = 2, \ldots, A-1 \tag{37}$$

$$ew_A(x) = \begin{cases} 0, x < a \\ \dfrac{x-a}{b-a}, a \le x \le b \\ \dfrac{c-x}{c-b}, b \le x \le c \\ 0, x > c \end{cases} \tag{38}$$

In equation (34), $Ew_{1,1}, Ew_{A,1}, Ew_{1,1}\&Ew_{1,B}$ have been introduced enhanced weight generation ew_A . The enhanced membership function does the greatest task of describing the node. Three parameters, $a, b, and\ c$, can be used to create the triangle

membership function that enhances the input. In the equation (34), a specifies the lower bound, b *and* c specify the middle, higher bounds for each node. For each node, $A = \{a, b, c\}$, weight x is produced at random.

Step 2-3. Steps two and three initialise the network edges' weight vectors for the four neurons. Start the remaining neurons from top to bottom and left to right, as indicated by equation (39).

$$we_{i,j} = \frac{we_{A,j} - we_{1,j}}{A-1}(i-1) + we_{1,j} \quad (39)$$

Choose a variety of dimensions and evaluate the result. To assess the training outcomes, the difference value and the winning numbers of each neuron should be counted. Equation (40) shows the learning rate's rate of fall, with an initial value of 0.1.

$$\eta(n) = \eta_0 * \left(1 - \frac{t}{T}\right) \quad (40)$$

Here, t is the iteration count and the initial learning rate is denoted as η_0.

Step 3: Using the all dove fitness function $f(we_j^e), j = 1, \ldots, N$ epoch, determine how many crumbs there are overall at the d[th] dove's location.

Step 4: Equation (41) may be used to determine which d_j^e is closest to the most crumbs using the highest requirements at epoch e.

$$d_j^e = \arg\max\{E(we_j^e)\}, \text{for j} = 1, \ldots, N \quad (41)$$

Step 5: To update the state of satiety for each dove, use equation (42).

$$S_j^e = \lambda S_j^{e-1} + e^{\left(E(we_j) - E\left(we_{d_f}\right)\right)}, for\ j = 1, \ldots, N \quad (42)$$

Step 6: Utilizing the maximal criterion, select the dove, d_s^e, with the highest level of satiety ,

$$d_s^e = \underset{1 \le j \le N}{\text{argmax}}\{S_j^e\}, \text{for j} = 1, \ldots, N \quad (43)$$

Equation (43) indicates that the dove with the best foraging behaviour, d_s^e, should serve as an example for the other doves in the flock.

Step 7: For every dove position vector, apply the maximum criterion,

$$we_j^{e+1} = we_j^e + \eta\beta_j^e(we_{d_s}^e - we_j^e) \quad (44)$$

$$\beta_j^e = \left(\frac{S_{b_s}^e - S_j^e}{S_{b_s}^e}\right)\left(1 - \frac{\left\|we_j^e - we_{d_s}^e\right\|}{maxDistance}\right) \quad (45)$$

$$maxDistance: \max_{1 \le j \le N}\left\|we_j - we_i\right\| \quad (46)$$

The parameter (η) represents the learning rate used to update the dove position vector. The updating is explained in full in the next step using equations (44-46).

Step 8: To satisfy the terminate condition, add one additional epoch by returning back to step 3. ($e = e + 1$). The terminate condition can be found in equation (47).

$$\left|E_{d_s}^e - T(e)\right| \le \varepsilon \text{ or } e \le \text{the set max epoch} \quad (47)$$

The number of e, and the count of doves N all contribute to the DSO algorithm order of complexity which is $O(NN_d\ e)$. Equations (48-49) can be changed to equations (44-45) if the optimization is based on the criterion is the best way to locate the minimum (we_j^e).

$$d_j^e = \arg\min\{E(we_j^e)\}, \text{for j} = 1, \ldots, N \quad (48)$$

$$S_j^e = \begin{cases} \lambda S_j^{e-1} + e^{\left(E(we_j) - E(we_{df})\right)}, if\ E(we_{d_f}) \neq \\ \lambda S_j^{e-1} + 1, if\ E\left(we_{d_f}\right) = 0, \text{for j} = 1, \ldots, \end{cases} \quad (49)$$

The dove with the highest degree of satiety is the one that the other doves go to search more food. When a dove is more satiated, it is more likely to become cautious and be reluctant to alter the way it now forages. However, a less satisfied dove would probably be more inclined to adapt its foraging strategy and replicate the LCD's. Figure 5 shows the general flowchart of the EDSOA.

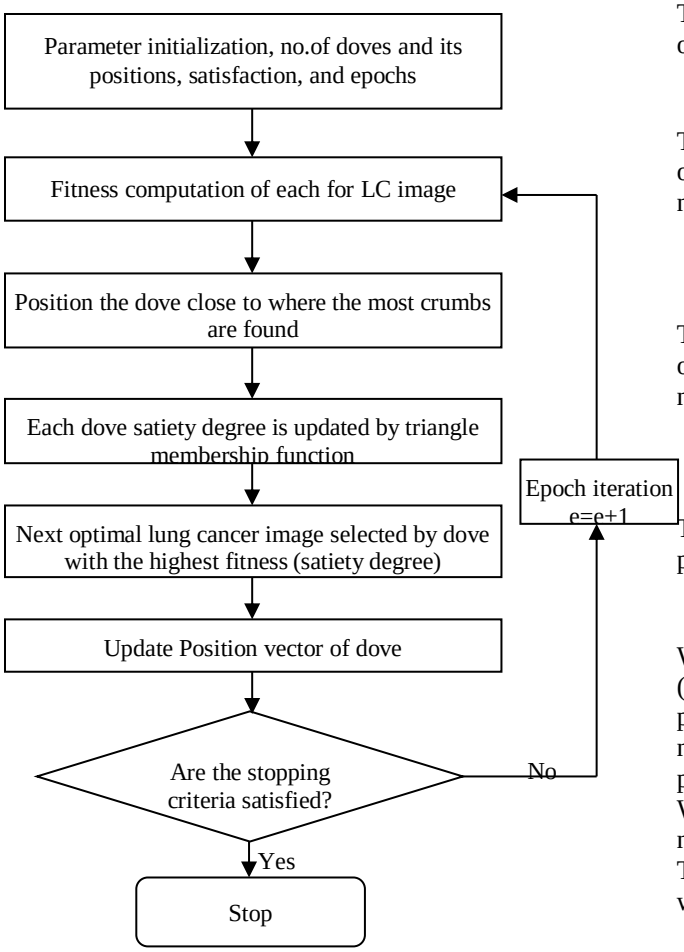

FIGURE 5. FLOW CHART OF EDSOA ALGORITHM

4. RESULTS AND DISCUSSION

The results of the classifier analysis of the LC dataset are shown in this section. PET/CT images were acquired from TCIA [18]. The suggested HGCN-BiLSTM classification algorithm for lung image recognition has been evaluated in this research using images acquired from the PET dataset. Then, various existing techniques are ANN [16], 3D-CNN-RNN [30], HGNN-BiLSTM, have been suggested for results comparison of suggested EDSOA-HGNN-BiLSTM.

To assess the classifiers in this work, performance metrics such as A, R (or) S, P, and F-Score (or) F-measure (F1) were employed. Table 1 displays the confusion matrix's (CM) outline.

TABLE 1. CM IN THIS ANALYSIS

Actual Class	**Prediction Class**	
	P	**N**
P	TP	FN
N	FP	TN

The predicted positive cases that turn out to be actual or real positives are measured by P.

$$\mathrm{P} = \frac{TP}{TP + FP} \tag{50}$$

The analysis of the overall count of real or true positive occurrences as influenced by the total number of false negative examples is evaluated by R.

$$\mathrm{Recall(R)/Sensitivity} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{51}$$

The F-measure is represented by the harmonic mean of P and R (F1). A balance between P and R is necessary.

$$\mathrm{F-Score/F-measure(F1)} = 2.\frac{\mathrm{P} * \mathrm{R}}{\mathrm{P} + \mathrm{R}} \tag{52}$$

The accuracy or precision of ML or classifier model predictions is evaluated by A.

$$\mathrm{A} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{53}$$

When a positive sample is classified as true positive (TP), it means that the diagnosis was correct. When positive samples are misdiagnosed, the result is a false negative (FN). An incorrect diagnosis of the non-positive samples is known as a false positive (FP). When a non-positive sample is classified as true negative (TN), it means that the diagnosis was correct. Table 2 discusses the suggested classifier's comparison with current techniques.

TABLE 2. COMPARATIVE ANALYSIS OF THE SUGGESTED CLASSIFIERS WITH CURRENT APPROACHES

Evaluation Metrics/Classifiers	**CNN**	**3D-CNN-RNN**	**HGNN-BiLSTM**	**EDSOA-HGNN-BiLSTM**
Accuracy	85.12	93	95.69	98.118
Precision	81	90.09	96.984	99.321
Recall	84	92.2	92.499	94.918
F-measure	83	91.12	94.688	97.069

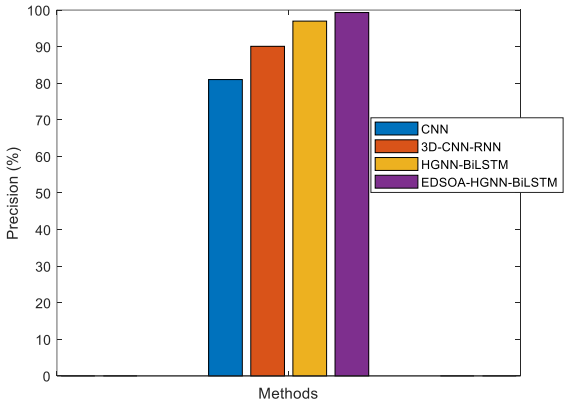

FIGURE 6. COMPARISON OF PRECISION (P) WITH CLASSIFIERS

The precision (P) comparison of classifiers such as ANN, 3D-CNN-RNN, HGNN-BiLSTM, and the suggested classifier EDSOA- HGNN-BiLSTM is displayed in Figure 6. EDSOA- HGNN-BiLSTM classifier has produces highest precision results of 94.4%. ANN, 3D-CNN-RNN, and HGNN-BiLSTM has lowest precision of 79.1%, 88.5% and 92.2% (Refer Table 2).

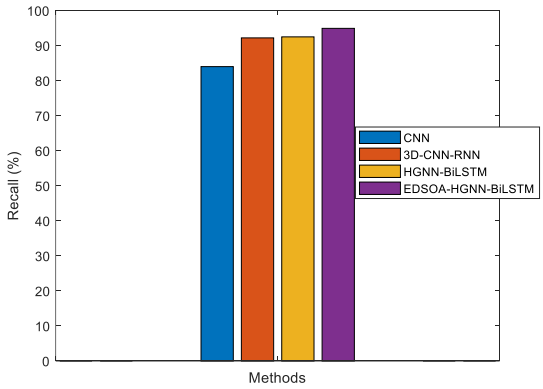

FIGURE 7. COMPARISON OF R WITH CLASSIFIERS

Recall (R) comparison of classifiers like ANN, 3D-CNN-RNN, HGNN-BiLSTM, and proposed classifier EDSOA- HGNN-BiLSTM, are presented in figure 7. EDSOA- HGNN-BiLSTM classifier has produces highest R outcomes of 95.2%. ANN, 3D-CNN-RNN, and HGNN-BiLSTM has lowest recall of 82.4%, 86.2%, and 93.4% (Refer Table 2).

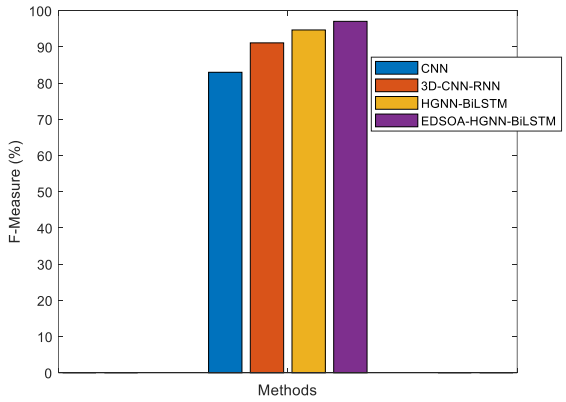

FIGURE 8. F-MEASURE (F1) COMPARISON VS. CLASSIFIERS

F-measure (F1) comparison of classifiers like ANN, 3D-CNN-RNN, HGNN-BiLSTM, and proposed classifier EDSOA- HGNN-BiLSTM are presented in figure 8. EDSOA- HGNN-BiLSTM classifier has delivered highest F-measure outcomes of 94.8%. ANN, 3D-CNN-RNN, and HGNN-BiLSTM has lowest F-measure of 80.8%, 87.4%, and 92.8% (Refer Table 2).

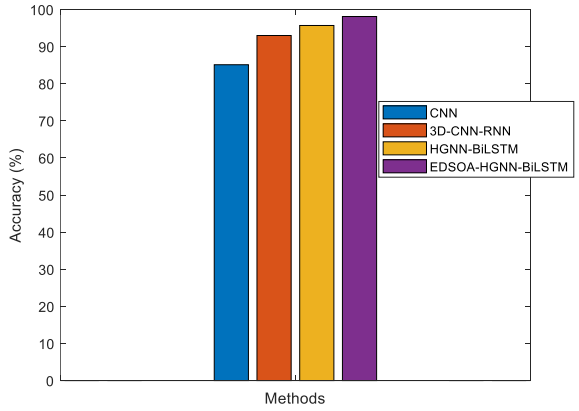

FIGURE 9. COMPARISON OF A WITH CLASSIFIERS

Figure 9 shows a comparison of classifiers such as ANN, 3D-CNN-RNN, HGNN-BiLSTM, and the suggested classifier EDSOA-HGCN-BiLSTM. At 97.8%, the EDSOA-HGCN-BiLSTM classifier yields the best A outcome. ANN, 3D-CNN-RNN, and HGNN-BiLSTM has lowest accuracy of 85.4%, 92.4%, and 95.6% (Refer Table 2).

5. CONCLUSION AND FUTURE WORK

A sophisticated framework for LCD in PET images using HGCN-BiLSTM that incorporates cutting-edge

methods for dataset collection, pre-processing, segmentation, FE, classification, and HPT.Initially, BF was used to improve image quality and lower noise in PET images. High-quality inputs for additional analysis are guaranteed by BF. Secondly segmentation, U-Net is utilized for precise segmentation of ROIs, delineating tumor boundaries with pixel-level accuracy. The features, such as GLRLM, GLCM, and the suggested LVP, are then obtained using FE. Then LC classification, hybrid framework combining GCN and BiLSTM units for enabling comprehensive spatial and temporal feature extraction. The EDSOA is introduced hyperparameter tuning for improve classification accuracy. The proposed framework is evaluated using the PET Dataset, demonstrating its efficiency in LCD accurately in PET images. Evaluated on the PET dataset, the framework demonstrates robust accuracy in LCD, outperforming traditional methods. From the simulation outcomes it is identified that the suggested model has improved accuracy as 97.8%. As the future work the ensemble model is introduced for improving the classifier accuracy.

REFERENCES

1. Guo, Y., Shang, X. and Li, Z., 2019. Identification of cancer subtypes by integrating multiple types of transcriptomics data with deep learning in breast cancer. *Neurocomputing*, *324*, pp.20-30.
2. Thomas, A., Pattanayak, P., Szabo, E. and Pinsky, P., 2018. Characteristics and outcomes of small cell lung cancer detected by CT screening. *Chest*, *154*(6), pp.1284-1290.
3. Taher, F., Prakash, N., Shaffie, A., Soliman, A. and El-Baz, A., 2021. An overview of lung cancer classification algorithms and their performances. *IAENG International Journal of Computer Science*, *48*(4), pp.1-7.
4. Kaur, M., Sakhare, S.R., Wanjale, K. and Akter, F., 2022. Early Stroke Prediction Methods for Prevention of Strokes. *Behavioural Neurology*, *2022*(1), pp.1-9.
5. Siegel, R.L., Miller, K.D. and Jemal, A., 2015. Cancer statistics, 2015. *CA: a cancer journal for clinicians*, *65*(1), pp.5-29.
6. Martin, M.D., Kanne, J.P., Broderick, L.S., Kazerooni, E.A. and Meyer, C.A., 2017. Lung-RADS: pushing the limits. *Radiographics*, *37*(7), pp.1975-1993.
7. Song, Q., Zhao, L., Luo, X. and Dou, X., 2017. Using deep learning for classification of lung nodules on computed tomography images. *Journal of healthcare engineering*, *2017*(1), pp.1-7.
8. Rodriguez-Canales, J., Parra-Cuentas, E., &Wistuba, I. I. (2016). Diagnosis and molecular classification of lung cancer. *Lung Cancer: Treatment and Research*, pp.25-46.
9. Schabath, M. B., & Cote, M. L. (2019). Cancer progress and priorities: lung cancer. *Cancer epidemiology, biomarkers & prevention*, *28*(10), 1563-1579.
10. Toğaçar, M., Ergen, B. and Cömert, Z., 2020. Detection of lung cancer on chest CT images using minimum redundancy maximum relevance feature selection method with convolutional neural networks. *Biocybernetics and Biomedical Engineering*, *40*(1), pp.23-39.
11. Naseer, I., Masood, T., Akram, S., Jaffar, A., Rashid, M. and Iqbal, M.A., 2023. Lung Cancer Detection Using Modified AlexNet Architecture and Support Vector Machine. *Computers, Materials & Continua*, *74*(1), pp.2039-2054.
12. Mikhael, P.G., Wohlwend, J., Yala, A., Karstens, L., Xiang, J., Takigami, A.K., Bourgouin, P.P., Chan, P., Mrah, S., Amayri, W. and Juan, Y.H., 2023. Sybil: a validated deep learning model to predict future lung cancer risk from a single low-dose chest computed tomography. *Journal of Clinical Oncology*, *41*(12), pp.2191-2200.
13. Hasan, M.R. and Al Kabir, M., 2020. Lung cancer detection and classification based on image processing and statistical learning. *Journal of Emerging Trends in Engineering and Applied Sciences*, *11*(6), pp.229-236.
14. Dafni Rose, J., Jaspin, K. and Vijayakumar, K., 2021. Lung cancer diagnosis based on image fusion and prediction using CT and PET image. *Signal and image processing techniques for the development of intelligent healthcare systems*, pp.67-86.
15. Shakeel, P.M., Burhanuddin, M.A. and Desa, M.I., 2019. Lung cancer detection from CT image using improved profuse clustering and deep learning instantaneously trained neural networks. *Measurement*, *145*, pp.702-712.
16. Vas, M. and Dessai, A., 2017. Lung cancer detection system using lung CT image processing. In *2017 international conference on computing, communication, control and automation (ICCUBEA)*, pp.1-5.

17. El Hamdi, D., Elouedi, I. and Slim, I., 2024. Computer-Aided Classification of Cell Lung Cancer Via PET/CT Images Using Convolutional Neural Network. *International Journal of Image and Graphics*, *24*(04), pp.2450040.
18. Li, P., Wang, S., Li, T., Lu, J., HuangFu, Y. and Wang, D., 2020. A Large-Scale CT and PET/CT Dataset for Lung Cancer Diagnosis (Lung-PET-CT-Dx). *The Cancer Imaging Archive*.
19. Chen, B.H., Tseng, Y.S. and Yin, J.L., 2020. Gaussian-adaptive bilateral filter. *IEEE Signal processing letters*, *27*, pp.1670-1674.
20. Zunair, H. and Hamza, A.B., 2021. Sharp U-Net: Depthwise convolutional network for biomedical image segmentation. *Computers in biology and medicine*, *136*, pp.1-13
21. Ma, H., Zou, Y. and Liu, P.X., 2021. MHSU-Net: A more versatile neural network for medical image segmentation. *Computer Methods and Programs in Biomedicine*, *208*, pp.1-10.
22. Weng, W. and Zhu, X., 2021. U-Net: convolutional networks for biomedical image segmentation. *IEEE Access*, *9*, pp.16591-16603.
23. Devi, E.A., Nasir, A.W., Devi, E.A., Mahesh, N., Pavithra, G. and Ramkumar, M.S., 2021. Texture based feature extraction and classification of retinal fundus image for glaucoma detection. In *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, pp.1662-1671.
24. Abbas, H.K., Fatah, N.A., Mohamad, H.J. and Alzuky, A.A., 2021. Brain tumor classification using texture feature extraction. In *Journal of Physics: Conference Series*, *IOP Publishing*, vol.1892, no.1, pp.1-11.
25. Shankar, K., Perumal, E., Tiwari, P., Shorfuzzaman, M. and Gupta, D., 2022. Deep learning and evolutionary intelligence with fusion-based feature extraction for detection of COVID-19 from chest X-ray images. *Multimedia Systems*, *28*(4), pp.1175-1187.
26. Fan, K.C. and Hung, T.Y., 2014. A novel local pattern descriptor-local vector pattern in high-order derivative space for face recognition. *IEEE transactions on image processing*, *23*(7), pp.2877-2891.
27. Almasan, P., 2019. Towards network optimization using graph neural networks. *Master's thesis, Universitat Politècnica de Catalunya)*, pp.1-18.
28. Ravulaplli, L.T., 2023. A Novel Bi-LSTM Based Automatic Image Description Generation. *Ingénierie des Systèmes d'Information*, *28*(2), pp.1-19.
29. Su, M.C., Chen, J.H., Utami, A.M., Lin, S.C. and Wei, H.H., 2022. Dove swarm optimization algorithm. *IEEE Access*, 10, pp.46690-46696.
30. Wankhade, S. and Vigneshwari, S., 2023. A novel hybrid deep learning method for early detection of lung cancer using neural networks. *Healthcare Analytics*, *3*, pp.1-13.