Deep learning for lung Cancer detection and classification

A. Asuntha[1] · Andy Srinivasan[2]

Abstract

Lung cancer is one of the main reasons for death in the world among both men and women, with an impressive rate of about five million deadly cases per year. Computed Tomography (CT) scan can provide valuable information in the diagnosis of lung diseases. The main objective of this work is to detect the cancerous lung nodules from the given input lung image and to classify the lung cancer and its severity. To detect the location of the cancerous lung nodules, this work uses novel Deep learning methods. This work uses best feature extraction techniques such as Histogram of oriented Gradients (HoG), wavelet transform-based features, Local Binary Pattern (LBP), Scale Invariant Feature Transform (SIFT) and Zernike Moment. After extracting texture, geometric, volumetric and intensity features, Fuzzy Particle Swarm Optimization (FPSO) algorithm is applied for selecting the best feature. Finally, these features are classified using Deep learning. A novel FPSOCNN reduces computational complexity of CNN. An additional valuation is performed on another dataset coming from Arthi Scan Hospital which is a real-time data set. From the experimental results, it is shown that novel FPSOCNN performs better than other techniques.

Keywords Lung cancer · Deep learning · Classifiers · Real-time · CNN

1 Introduction

Lung cancer is one of the most important deadly diseases in the world [30, 37]. The recent estimates provided by World Health Organization (WHO) says that around 7.6 million deaths worldwide per year due to lung cancer [21, 22]. Moreover, humanity due to cancer are

✉ A. Asuntha
   asuntha.srm@gmail.com

1   Department of Electronics & Instrumentation Engineering, SRM Institute of Science & Technology, Kattankulathur, Kancheepuram District, Tamil Nadu, India

2   Department of Electronics & Instrumentation Engineering, Valliammai Engineering College, Kancheepuram District, Tamil Nadu, India

supposed to continue rising, to become around 17 million worldwide in 2030 [15]. Discovering lung cancer in the early stage is the only method for its cure [12]. Different methods are available for diagnosis lung cancer, namely, MRI, isotope, X-ray and CT. X-ray chest radiography and Computer Tomography (CT) are the two familiar anatomic imaging modalities that are regularly used in the recognition of different lung diseases [25, 29]. CT images are used by physicians and radiologists to identify and recognize the presence of diseases, directly visualize the morphologic extents of diseases, describe the patterns and severity of diseases, and measure the clinical course of diseases and response to therapy. The volumetric CT technique has introduced spiral scans which shorten the scan time and, when used in thoracic imaging, reduce the artefacts caused by partial volume effects, cardiac motion, and unequal respiratory cycles. As the progress of CT technology, the high-resolution CT test has happen to the imaging modality of choice for the recognition and identification of lung diseases. Even though High-Resolution Computed Tomography (HRCT) recommends images of the lung with progressively more improved anatomic resolution, visual interpretation or evaluation of a large number CT image slices remains as a difficult task.

In CT images, lung cancer disease cannot be identified easily as shown in Fig. 1. For lung cancer screening, Nowadays Low-Dose helical Computed Tomography (LDCT) [18] is being applied as a modality [31]. There are lots of works being done to develop computer assisted diagnosis and detection systems to improve the diagnostic quality for lung cancer detection classification [2]. The necessity for reliable and objective analysis has prompted the development of computer-aided systems. The aim of this work is to extract features for classification [4] and Severity finding.

In this work, first the input image is enhanced by using histogram equalization for image contrast and denoised by using Adaptive Bilateral Filter (ABF). After pre-processing, the next step is to find the lung region extraction. To extract the lung region, Artificial Bee Colony (ABC) segmentation approach is applied. The holes in the lung region are filled by using mathematical morphology technique in the ABC segmented image. After that the texture features are extracted to find the cancerous lung nodules. After finding the location of the cancerous lung nodules the next process is to classify the lung disease name and its severity based on the feature extraction. A new CNN method based on FPSO for reducing the computational complexity of CNN is proposed. FPSOCNN improves the efficiency of CNN.

The main contributions of this paper are summarised as follows:

(i) Deep learning is applied to classify benign and malignant pulmonary nodules.

(a)   (b)
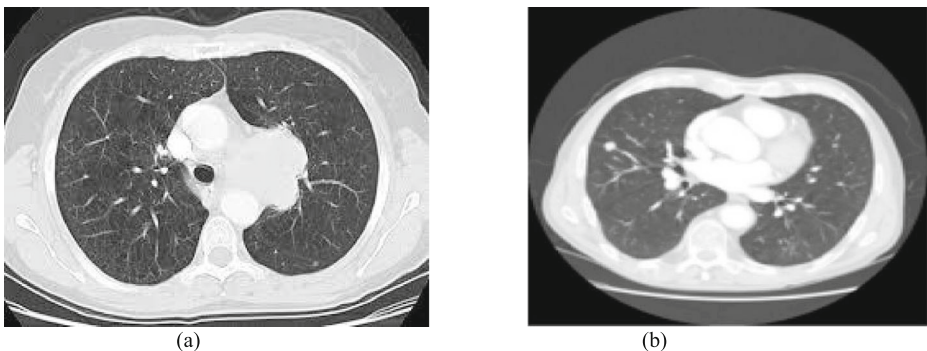
Fig. 1 **a** Represents the clear lung image, **b** Represents the diseased lung image

(ii)   For extracting features, four features are used, namely, texture feature, geometric feature, volumetric feature and intensity features,

(iii)   Best feature extraction techniques such as wavelet transform-based, Local Binary Pattern (LBP), Scale Invariant Feature Transform (SIFT), and Zernike Moment are used.

(iv)   The proposed method provides good results for LIDC data set and real-time data set.

The manuscript of this paper is organised as follows: in Section 2, some related state-of-the-art literatures are reviewed. In Section 3, a detailed description of the proposed architecture is shown. In Section 4, experimental results are shown. Discussion is presented in Section 5. Finally, conclusions and future works are provided in Section 6.

2 Related work

During the most recent decades, prompt progress of pattern recognition and image processing techniques [37], lung cancer detection classification attracts more and more research works. Existing methods in the literature for differentiating a variety of obstructive lung diseases on the basis of textural analysis of thin-section CT images are explained. Chabat et al. [9] have created a 13-dimensional vector of local texture information, which contains statistical moments of CT attenuation distribution, acquisition-length parameters, and co-occurrence descriptors. For feature segmentation, a supervised Bayesian classifier is applied. Here, the dimensionality of the feature vector is reduced using five scalar measurements, namely, maximum, entropy, energy, contrast, and homogeneity that were extracted from each co-occurrence matrix obtained. Yanjie Zhu et al. [38] have presented texture features of Solitary Pulmonary Nodules (SPNs) detected by CT and evaluated. Totally, 67 features were extracted and around 25 features were finally selected after 300 genetic generations. For classification, SVM based classifier is applied. Sang Cheol Park et al. [25] have applied genetic algorithm to select optimal image features for Interstitial Lung Disease (ILD). Hiram et al. [23] have classified lung nodule using Frequency domain and SVM with RBF. Hong et al. [28] have proposed an algorithm for detecting solitary pulmonary nodules automatically. SVM classifier is applied to recognize true nodules and label them on original images. Antonio et al. [13] have classified lung nodules using the LIDC-IDRI image database. Taxonomic Diversity and Taxonomic Distinctness Indexes from ecology are applied with SVM [18] for classification. Results depict a mean accuracy of 98.11%.

In CT examination, only pixels selected by the mesh-grid region growth method were analyzed and classified using ANN to improve computational efficiency. All unselected pixels were classified as negative for ILD. Zhi-Hua et al. [37] have proposed Neural Ensemble-based Detection (NED) which utilised artificial neural network ensemble to identify lung cancer cells. This method provides high accuracy in identification of cancer cells. Hui Chen et al. [10] have provided a computerized scheme for formatting a lung nodule's classification on a thin-section CT scan using a Neural Network Ensemble (NNE). Aggarwal, Furquan and Kalra [1] have proposed a model that classifies normal lung anatomy structure. Optimal thresholding is applied for segmentation. Features are extracted using geometrical, statistical and gray level characteristics. LDA is applied for classification. The results show 84% accuracy, 97.14% sensitivity and 53.33% specificity. Roy, Sirohi, and Patle [26] have developed a system to detect lung cancer nodule using fuzzy inference system for classification. This method uses gray transformation for image contrast enhancement. The resulted image is segmented using

active contour model. Features like area, mean, entropy, correlation, major axis length, minor axis length are extracted to train the classifier. Overall, accuracy of the system is 94.12%. The limitation of this method is, it does not classify the cancer as benign or malignant which is future scope of this proposed model. Hiram et al. [24] have classified lung nodules using wavelet feature descriptor and SVM. Here, wavelet transforms are computed with one and two levels of decomposition. From each wavelet sub-band 19 features are computed. SVM is applied to differentiate CT images with cancerous nodules and not containing nodules.

Bhuvaneswari and Brintha [7] have applied Gabor filter for extracting features. The results of Gabor filter are given to K-NN classifier which is optimized by GA (Genetic Algorithm). The limitation of K-NN classifier is overcome by G-KNN classifier. Sangamithraa, and Govindaraju [27] have segmented the lung using region of interest and analysed the area for nodule detection in order to examine the disease. For extracting features, statistic method called Gray Level Co-occurrence Matrix (GLCM) is applied. For classification, supervised neural network called the Back Propagation Network (BPN) is applied. For removing unwanted artefacts in CT images, Sangamithra et al. [27] have pre-processed using median and wiener filters. Fuzzy K-Means clustering method is applied for segmentation. After that, entropy, contrast, correlation, homogeneity and area are applied for extracting features from the Fuzzy K- Means segmented Image. For feature extraction, statistic method called GLCM is applied. Finally, classification is done by using the supervised neural network called the Back Propagation Network (BPN). Results shows whether the CT Image is a normal Image or cancerous with accuracy of about 90.7%. Suren Makaju et al. [20] have segmented the input image using watershed segmentation. The segmented results show the image with cancer nodules marked. After that features are extracted using area, perimeter, eccentricity, centroid, diameter and pixel mean intensity for the segmented cancer nodules. Finally, classification of cancer nodule has been performed using Support Vector Machine (SVM). Jin, Zhang and Jin [17] have used Convolution Neural Network (CNN) as classifier to detect the lung cancer. The results reported an accuracy of 84.6%, sensitivity of 82.5%, and specificity of 86.7%. Wafaa Alakwaa et al. [3] have applied thresholding for segmentation approach. 3D CNN was used to classify the CT scan as positive or negative for lung cancer. This result in an accuracy of 86.6%. Ignatious and Joseph [16] have applied Gabor filter to enhance the image quality. For segmentation, watershed segmentation is used. The results show an accuracy of 90.1% which is higher than neural fuzzy model and region growing method. Wenqing et al. [30] have used three deep learning algorithms such as CNN, Deep Belief Networks (DBNs) and Stacked Denoising Autoencoder (SDAE) for lung cancer classification. Qing Zeng et al. [29] have applied Deep Learning for Classification of Lung Nodules on CT Images.

2.1 Observation

Even though existing methods provide good classification accuracy, the existing system accuracy is still less. Mostly LIDC public data set alone has been used. Mostly machine learning algorithms are applied for classification. Still, more comparison has to be performed with a real-time data set to choose suitable Deep learning techniques for classification. In this work, accuracy is improved by choosing well-known feature extraction methods and applying deep learning. There is not much work for severity finding.

3 System methodology

The overall architecture for the lung disease classification has been shown in Fig. 2. In offline process, lung tissue images are trained. In online process, the input lung tissue image is denoised by using the adaptive bilateral filter and the image contrast is enhanced by using the histogram equalization. After pre-processed the input image the next step is to lung region extraction. To extract the lung region the artificial bee colony segmentation approach is applied. The holes in the lung region are filled by using mathematical morphology technique in the output of the ABC segmented image. After that the texture features are used to find the cancerous lung nodules. After finding the location of the cancerous lung nodules the next process is to classify the lung disease name and its severity based on the feature extraction. Among several feature extraction methods this work uses six feature extraction techniques such as the bag of visual-words based on the histogram of oriented gradients, the wavelet transform-based features, the local binary pattern, SIFT, Zernike Moment. After extracting the features the Fuzzy
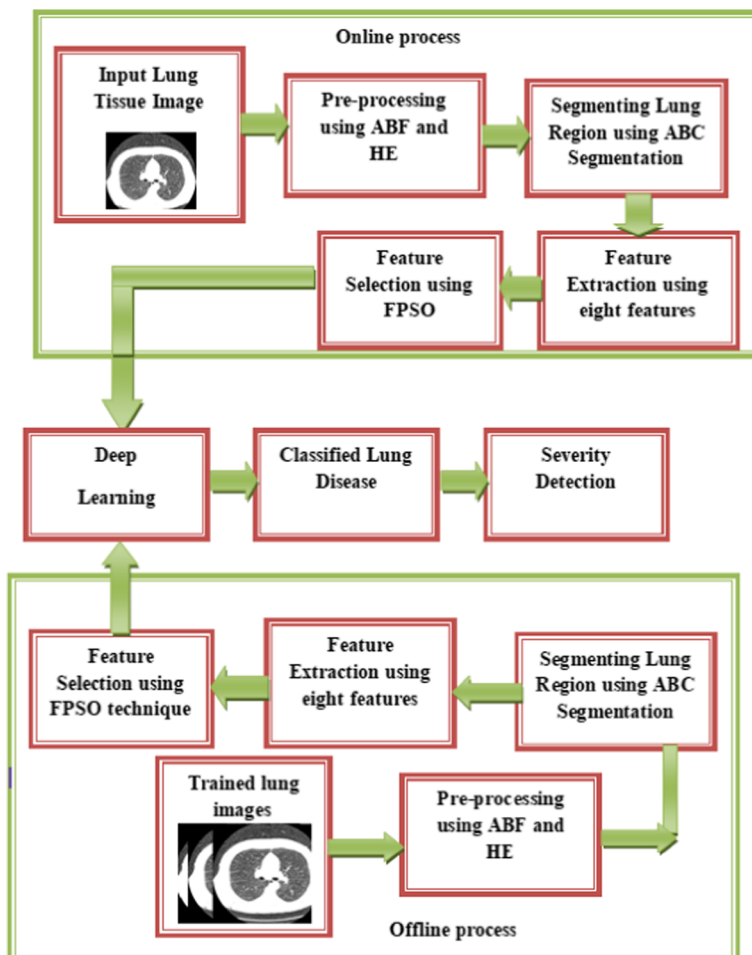

Fig. 2 Overall architecture of the proposed work

Particle Swarm Optimization (FPSO) algorithm is used for select the best feature. Finally these features are classified using Deep learning techniques.

3.1 Pre-processing

Firstly, for pre-processing, the contrast of input CT scan images is enhanced by using the Histogram Equalization (HE) technique. The HE is applied for adjusting image intensities to enhance contrast as shown in Eq. (1). Let I be a given CT scan image represented as a I_x by I_y matrix of integer pixel intensities ranging from 0 to 256. Let N denote the normalized histogram bin of image I for available intensity.

$$I_N = \frac{Number\ of\ pixels\ with\ available\ intensity\ n}{Total\ number\ of\ pixels} \tag{1}$$

Where $n = 0, 1\dots 255$.

It locally recovers the contrast of images by dividing the image into numerous sub regions and by transforming the intensity values of each sub region independently to fulfil with a specified target histogram [5]. The enhanced images are shown in Fig. 3.

Next, for pre-processing, Adaptive Bilateral Filter (ABF) is used on an enhanced CT scan images for de-noising. ABF is an expansion of the traditional bilateral filter [34]. ABF contains some important alterations than bilateral. Range filters used in ABF locally adaptive. By adding a counterbalance to the range filter, the range filter on the histogram is shifted as shown in Eq. (2).

(a)   (b)
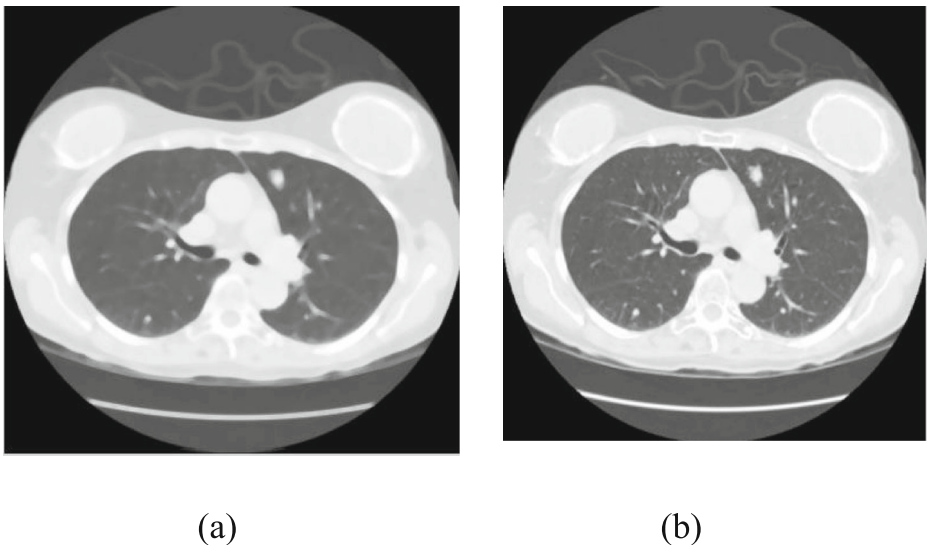
Fig. 3 **a** Represents the input lung tissue image, **b** Represents the enhanced image using histogram equalization

$$ABF_{x_0,y_0} = \sum_{x=x_0-N}^{x_0+N} \sum_{y=y_0-N}^{y_0+N} \exp\left(-\frac{(x-x_0)^2 + (y-y_0)^2}{2\nabla_d^2}\right)$$

$$\times \exp\left(-\frac{(G[x_0,y_0]-G[x_0,y_0]-\delta[x_0,y_0])^2}{2\nabla_r^2[x_0,y_0]}\right) \tag{2}$$

Where x_0 defines the row index of the current pixel is, y_0 defines the column index of the current pixel of the image. x defines the row index of a neighbouring pixel. y defines the column index of a neighbouring pixel. N is a neighbouring window size. Ω_{x_0,y_0} is the centre pixel of a neighbouring window.

If ∇_r and δ is fixed, the ABF will degenerate into a conventional bilateral filter. For ABF, a fixed low-pas Gaussian filter is adopted. The combination of locally adaptive and bilateral filter makes ABF into a much more powerful filter that is capable of both smoothing and sharpening. Moreover, ABF sharpens an image by increasing the slope of the edges. The δ is estimated in ABF using Eq. (3).

$$\delta[x_0,y_0] = \begin{cases} MAXIMUM\left(\beta_{x_0,y_0}\right)-G[x_0,y_0], & if \ \Omega_{x_0,y_0} > 0 \\ MINIMUM\left(\beta_{x_0,y_0}\right)-G[x_0,y_0], & if \ \Omega_{x_0,y_0} < 0 \\ 0, & if \ \Omega_{x_0,y_0} = 0 \end{cases} \tag{3}$$

Where x_0 defines the row index of the current pixel is, y_0 defines the column index of the current pixel of the image.

The window size of input image is represented as $(2W + 1) \times (2W + 1)$. Here each pixel is represented as β_{x_0,y_0} with centre $[x_0,y_0]$. Let MAXIMUM and MINIMUM depicts the operations of taking the value of the data in respectively. The effect of ABF with a fixed domain Gaussian filter and a range filter is effective. Here $\nabla_d = 1$ is fixed and ∇_r value changes as shown in Fig. 4.
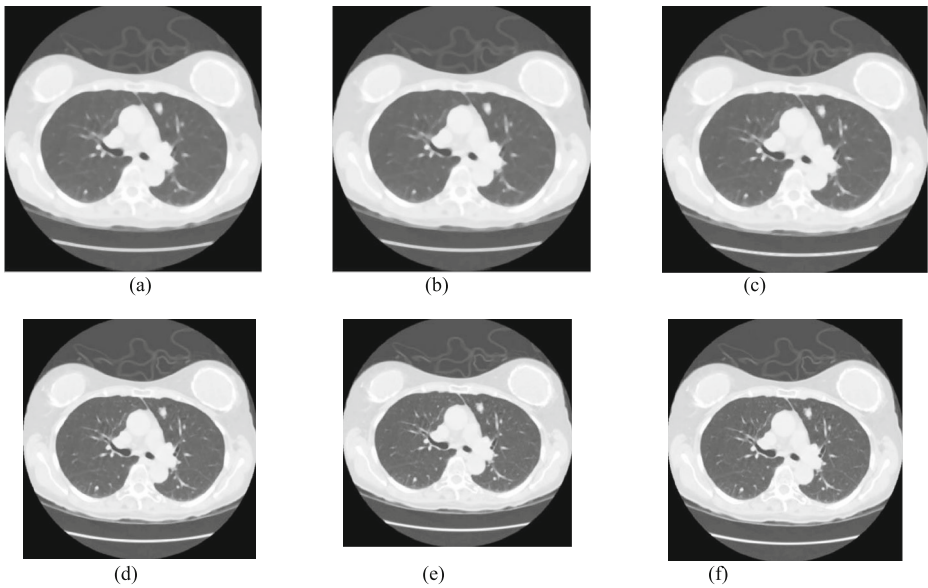
Usually, some noises are surrounded on CT Images at the time of image acquisition process which aids in false detection of nodules. Sometimes noise may be detected as cancer nodules. Therefore, the extra noises have to be removed for accurate detection of cancer. This ABF sharpens CT scan image by increasing the slope of the edges without producing overrun. It is able to smooth the noise, while enhancing edges and textures in the image.

Compared to the conventional filters such as mean and median, ABF provides good results as shown in Fig. 5. The problems in conventional filters such as overshoot undershoot around edges, which causes objectionable ringing or halo artefacts are overcome by ABF. ABF increases the slope of edges in the image without producing overshoot and undershoot which renders clean, crisp, and artefact-free edges, and also improves the overall appearance. Bilateral filter fails in restoring the sharpness of a degraded image. ABF provides good results in both sharpness enhancement and noise removal as shown in Fig. 5.

3.2 Segmentation

In pre-processed image, segmentation process locates objects or boundaries which help in acquiring the region of interest in the image. It partitions the image into regions to identify the meaningful information. In lung cancer classification it is important to segment the cancer

Fig. 4 Impact of ∇_r and ∇_d. 4. **a** $\nabla_d = 1,4$. **b** $\nabla_r = 1, 4$. **c** $\nabla_r = 5, 4$. **d** $\nabla_r = 10, 4$. **e** $\nabla_r = 25, 4$. **f** $\nabla_r = 50$

nodule from the pre-processed CT scan image. The pre-processed image is first segmented using Artificial Bee Colony (ABC) segmentation algorithm as shown in Fig. 6.
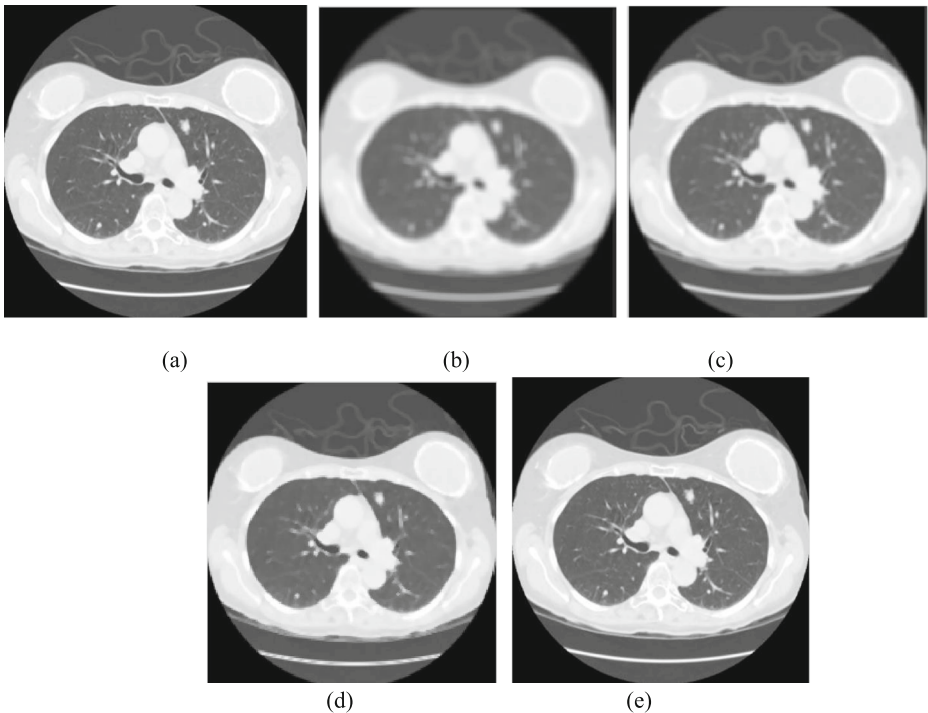
Fig. 5 **a** Input image, **b** Mean filter, **c** Median filter, **d** Bilateral filter and **e** ABF
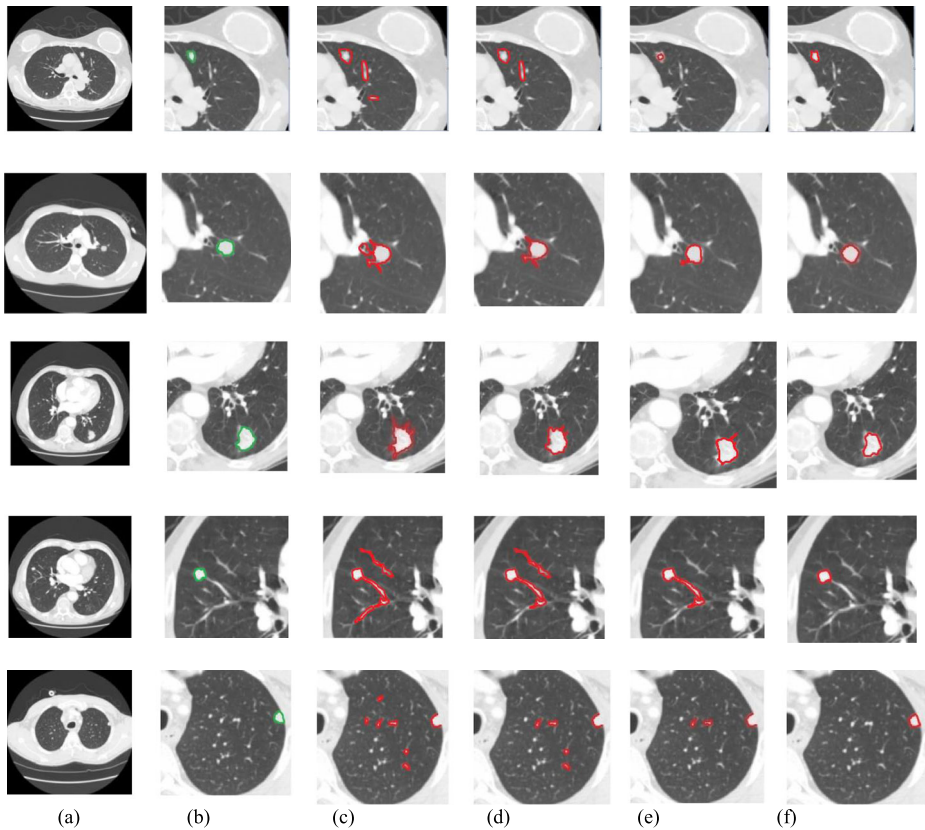
Fig. 6 **a** Respresents the pre-processed image, **b** Ground Truth, **c** Represents the segmentation using kmeans, **d** Represents the segmentation using FCM, **e** Represents the segmentation using ant colony, **f** Represents the segmentation using ABC

For segmenting the lung cancer nodules, several techniques such as K-Means, FCM and Ant Colony algorithms are applied. FCM algorithm has long computational time. It is sensitive to speed, local minima and noise. K-means algorithm has difficulty in predicting the number of clusters [8]. In Ant Colony Approach, the probability distribution changes by iteration and it is independent on the earlier decision to find the best solution. It takes lot of time to convergence uncertain. These drawbacks are overcome by ABC. ABC is Simple, flexible and robust. Its implementation is easy. It has fewer control parameters to explore local solutions and to handle objective cost.

The two important functions in ABC segmentation are.

$$PV_i = \frac{FV_i}{\sum_{m=1}^{SN} FV_n} \qquad (4)$$

$$C_{ij} = I_{ij} + \varnothing_{ij}\left(I_{ij} - I_{kj}\right) \qquad (5)$$

Where PV_i is the probability value associated with i^{th} food source that calculated by the Eq. (4).

An onlooker bee selects a food source relying on PV_i. In this equation, FV_i represents ith food source's nectar amounts, which is measured by employed bees and SN is the number of food source which is equal to the number of employed bees. Fitness is calculated by Eq. (6).

$$FV_i = \begin{cases} {1}/{(1+C_{ij})}, & C_{ij} \geq 0 \\ 1 + abs(C_{ij}), & C_{ij} < 0 \end{cases} \tag{6}$$

Where C_{ij} is the cost function of the quality of source which is calculated from the Eq. (5) and abs is the absolute value of C_{ij}. Greedy selection is applied to select the best source. In the real-world problems, I_{ij} and I_{kj} represent the different old food source positions. The difference between these two positions is the distance from one food source to the other one. $\varnothing_{ij}$ is a random number between $[-1, 1]$ and controls the distance of a neighbour food source position around I_{ij}.

3.3 Feature extraction

In this step, the features are extracted from the segmented lung image. Here, four types of ROI features are used for extracting features. They are volumetric features, texture features, intensity features and geometric features. For extracting texture features, LBP and wavelet techniques are used. This paper extracts 96 LBP features, 26 wavelet features, 18 HOG features, 1 Eccentricity feature, 1 Curvature feature, 18 SIFT Features and 20 Zernike moment features. Finally, 180 features are extracted. Four feature categories are explained in more details in the following sections.

3.3.1 Texture features

Wavelet features Wavelets are important and commonly used feature descriptors for texture feature extraction. Wavelet features show their effectiveness in capturing localized spatial frequency information and multiresolution characteristics. The wavelet signal passes successively through pairs of low pass and high pass filters, the analysis filters, which produce the transform coefficients. Here in this image, a H level decomposition is performed resulting in 3H + 1 different frequency bands. The frequency sub-bands LH is used to constitute the vertical details of the image, HL is used to constitute the horizontal details of the image, HH is used to constitute the diagonal details of the image. The LL sub-band is combined with discrete wavelet transform to obtain more level of decomposition as shown in Fig. 7.

It generates another four sub-bands. Sub-band LL represent the approximate element of image, LH represent the vertical element of image, HL represent the horizontal element of image and HH represent the diagonal element of image. Thus the information of image is stored in decomposed form in these sub-bands. Here, the ROIs are decomposed to four levels by using 2-D symlets wavelet because the symlets wavelet has better symmetry than other wavelets. The horizontal, vertical, and diagonal detail coefficients are extracted from the wavelet decomposition structure. Finally, the wavelet features by calculating the mean and variance of these wavelet coefficients. Some texture features combined with this wavelet function is shown in Table 1.

LBP features The LBP feature is a compact texture descriptor in which each comparison result between a centre pixel and one of its surrounding neighbors is encoded as a bit. The LBP

Fig. 7 Wavelet feature sub-band

LL	HL
LH	HH

operator is a unified approach to statistical and structural texture analysis. Their values are compared with the value of the centre pixel. For each neighbouring pixel, the result will be set to one if its value is no less than the value of the centre pixel, otherwise the result will be set to zero. The LBP code of the centre pixel is obtained by multiplying the results with weights given by powers of two.

3.3.2 Intensity features

Intensity feature is frequently used as the most important source of image information in CT images. Intensity features and their equation descriptors are described in more details in Table 2. The HOG feature is also considered in intensity feature extraction.

HOG features The HOG feature is a texture feature descriptor describing the distribution of image gradients in different orientations. In the HOG features, histogram of gradient directions and edge orientations are accumulated over the pixels of the cell. After that the gradient values are computed. The centred and point discrete derivative mask is applied in both horizontal and vertical directions. The filter kernels $[-1, 0, 1]$ and $[-1, 0, 1]^T$ are applied.

After applying kernels, each pixel is estimated with a weighted vote for an edge orientation histogram channel and the votes are accumulated into orientation bins over

Table 1 Texture features used with equations

Texture features used	Equation with description
Difference variance	$\sum_{k=0}^{l-1} k^2 P_{x-y}(k)$
Sum average	$\sum_{k=0}^{l-1} k^2 P_{x+y}(k)$
Maximum probability	$max_{i,j} \, X(i,j)$
Contrast	$\sum_{i=0}^{l-1} \sum_{j=0}^{l-1} |i-j|^2 \, X(i,j)$
Auto-correlation	$\sum_{i=0}^{l-1} \sum_{j=0}^{l-1} |i-j| \, X(i,j)$

Table 2 Intensity features used with equations

Intensity features used	Equation with description
Mean $\bar{x}$	$\frac{1}{P^2} \sum\limits_{k=1}^{P} \sum\limits_{l=0}^{P} I_{(k,l)}$
Standard Deviation σ	$\sqrt{\frac{1}{P} \sum\limits_{k=0}^{P} I_k - \bar{x}^2}$
Skewness	$\frac{\sum\limits_{k=0}^{PP} (I_k - x)^3}{(P-1)\sigma^2}$
Entropy	$\frac{1}{2} \left(\sum\limits_{k=0}^{255} \sum\limits_{l=0}^{255} G_H(k,l).\log(G_H(k,l)) + \sum\limits_{k=0}^{255} \sum\limits_{l=0}^{255} G_V(k.l).\log(G_V(k,l)) \right)$
Max-Intensity	$Max[I(k,l)]$

local spatial regions. Here, ROI is divided into smaller rectangular blocks of 8×8 pixels and further divide each block into four cells of 4×4 pixels. An orientation histogram which contains nine bins covering a gradient orientation range of 0–180° is computed for each cell. Here, a block is represented by linking of the orientation histograms of cells in it. This means a 36 Dimension HOG feature vector is extracted for each block in the segmented lung image.

3.3.3 Volumetric features

Zernike moment features The Zernike moments are the descriptors of mass shapes in extracting features. Here, the input pre-processed image is subjected to histogram equalization at first, which shows the mass margins of images more visible. Zernike moments are dependent on the translation and scaling of masses in ROIs. In other words, the Zernike moments of two similar images that are not equally scaled and translated are different. Two processes have been employed to resolve the dependency problems in Zernike moments. The centroid of each image mass is translated into the centre of corresponding ROI. This process eliminates the dependency of Zernike moments to object translation.

SIFT features SIFT is called a volumetric feature because it calculates the edges of an image using keypoints. SIFT features are applied because they are invariant to small illumination changes, scale changes, image rotation, and viewpoint changes. The SIFT algorithm have four main steps: (1) Scale-Space Peak Selection (SSPK), (2) Keypoint Localization, (3) Orientation Assignment, (4) Keypoint Descriptor Computation, and (5) Keypoint Matching. In the first phase, SSPK has been used by constructing a Gaussian Pyramid (GP). GP is calculated by searching the extreme local peaks in a series of Difference-of-Gaussian (DoG) images. DoG is estimated as the difference of Gaussian blurring of an image with two different values.

3.3.4 Geometric features

Eccentricity Eccentricity is defined in the equation below,

$$ECC = \sqrt{1 - \frac{x^2}{y^2}} \tag{7}$$

where x and y are semi-major axis and semi-minor axis lengths of nodule region of interest, respectively.

Curvature descriptor Curvature descriptor is estimated with respect to intensity inside nodule region of interest, which depends on the intensity variation.

$$Cdd = tan^{-1} \left(\frac{\sqrt{\alpha_1^2 + \alpha_2^2}}{1 + f(x,y)} \right) \tag{8}$$

Where α_1 and α_2 ($\alpha_1 \leq \alpha_2$) are two Eigen values of Hessian matrix.

3.4 Feature selection

In order to achieve good classification results, generally several types of features are applied at the same time. Since the different types of features may contain complementary information, it could bring better classification performance through selecting discriminative features from various feature spaces. The advantage of feature selection is to determine the importance of original feature set [35, 36]. For feature selection, Fuzzy Particle Swam Optimization (FPSO) is applied. A FPSO [32] is composed of a knowledge base, that includes the information given by the expert in the form of linguistic control fuzzy rules, a fuzzification interface, which has the effect of transforming crisp data into fuzzy sets, an inference system, that uses them together with the knowledge base to make inference by means of a reasoning method, and a defuzzification interface, that translates the fuzzy control action thus obtained to a real control action using a defuzzification method. FPSO is based on the following equation,

$$FPSO\ (p) = \sum_{k=1}^{n} \sum_{i=1}^{c} [\tau_{ik}]^m \ \|y_k - W_i\|^2 \tag{9}$$

where, m > 1 is a real number, W_i is the cluster centre of i, y_k is the vector part of k.

$$W_i = \frac{\sum_{k=1}^{n} [\tau_{ik}]^m y_k}{\sum_{k=1}^{n} [\tau_{ik}]^m} \tag{10}$$

$$[\tau_{ik}]^{(t+1)} = \left[\sum_{j=1}^{c} \left(\frac{\|y_k - W_i^{(t)}\|^2}{\|y_k - W_i^{(t)}\|^2} \right)^{\frac{1}{m-1}} \right]^{-1} \tag{11}$$

Where c = 1, 2, ...n, $\|y_k - W_i\|^2$ represents the Euclidean distance between y_k and W_i, and $[\tau_{ik}]^{(t+1)}$ is the membership degree of part k in group i.

For feature selection process several existing algorithms such as PSO, DE, GA are available. GA does not guarantee an optimal solution. This problem is solved by using PSO. Both GA and DE have high computational cost. But PSO is computationally less expensive. In PSO, the best particle in each neighbourhood exerts its influence over other particles in neighbourhood. To overcome these problems, several particles in each neighbourhood can be allowed to influence others to a degree by a fuzzy variable. The selected features using DE,GA, PSO and FPSO using various test rounds are shown in Table 3.

3.5 Classification

3.5.1 Bag classifier

The Bag classifier is mainly used in natural language processing and information retrieval (IR). Recently, this classifier has also been used for computer vision. In computer vision application, it is applied to image classification regarding image feature data sets. In this classifier, each image is treated as a document, and it is characteristically symbolized by a histogram generated from the training images. Here, the classes of the images are selected and labelled in advance. Here, classifier makes a decision that which class name from given training classes to be the class name of a test image.

3.5.2 Naive Bayes classifier

Naive Bayes classifier depends on a probability model and allocates the specific class, which has the maximum estimated posterior probability to the feature vector. The posterior probability $P(C_v/FV)$ of a specific class C_v is given by a feature vector FV is determined using Bayes' theorem is given in equation below,

Table 3 Number of Selected Features using DE, GA, PSO and FPSO with various Test Rounds

Test Rounds	Feature Selection Approaches			
	DE	GA	PSO	FPSO
1	115	108	99	94
2	141	131	127	122
3	159	150	144	140
4	164	159	151	145
5	168	160	155	150

$$P(C_v/FV) = \frac{P(FV/C_v)\,P(C_v)}{P(FV)} \tag{12}$$

Naive Bayes Classifier [6] technique is based on the Bayesian theorem and is particularly suited when the dimensionality of the inputs is high. Despite its simplicity, Naive Bayes can often outperform more sophisticated classification methods.

3.5.3 K-NN classifier

The k-Nearest Neighbors algorithm is a non-parametric method [5] used for classification and regression. The input consists of the k closest training examples in the feature space. In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k-nearest neighbors. If k = 1, the object is simply assigned to the class of that single nearest neighbour.

3.5.4 Adaboost classifier

Adaboost classifier is trained on more training examples. It provides a good fit to training examples by producing low training error. It is very simple. Adaboost improves the classification results by combining weak predictors together [6].

Adaboost algorithm maintains a set of weight over the training images. Let us consider the training set of images be $(t_1, x_1), (t_2, x_2), \ldots (t_n, x_n)$ where t_i belongs to some domain space of T. After that each label of x_i is in the label set of X = {−1, +1} is shown in equation below,

$$(t_1, x_1), \ldots (t_n, x_n); t_i \in T, x_i \in \{-1, +1\} \tag{13}$$

The weight on the training example i on round r is depicted as $W_k(i)$. The weights are initialized using the next equation below,

$$W_k(i) = 1/N, i = 1 \ldots N \tag{14}$$

Where k = 1...K, which represents the series of rounds. The images are trained along with their assigned weights.

3.5.5 SVM classifier

SVM [5] is a binary classification method that takes as input labelled data from two classes and outputs a model file for classifying new unlabelled/labelled data into one of two classes. It group items that have similar feature maps into groups. SVM constructs a hyperplane that maximizes the margin between negative and positive samples. Finally, classification is performed by the decision based on the value of the linear combination of the features.

SVM is trained by feeding known data with previously known decision values, by forming a finite training set. It is from the training set that an SVM gets its intelligence to classify unknown data. In SVM, for two class classification problem, input data is mapped into higher

dimensional space using RBF kernel [5]. Here, a hyper plane linear classifier is applied in this transformed space utilizing those patterns vectors that are closest to the decision boundary.

The estimation for the classification using SVM with N support vectors $g_1, g_2, \ldots g_n$ and weights $\tau_1, \tau_{2, \ldots} \tau_n$ is given by:

$$SVM = \sum_{i=1}^{n} \tau_i \left(g_i, x\right) + b \qquad (15)$$

Where x represents a feature vector and b represents a bias.

3.5.6 ELM classifier

The existing classifiers encounter several problems while training such as local minima, not proper learning rate and over fitting, differentiable activation functions etc. To overcome these problems, ELM has enhanced generalization result. ELM will provide the results directly without such difficulties. ELM classifier can also be used to train SLFNs with many non-differentiable activation functions.

ELM are feed forward neural networks for classification, regression, clustering, sparse approximation, compression and feature learning with a single layer or multiple layers of hidden nodes, where the parameters of hidden nodes need not be tuned. In most cases, the output weights of hidden nodes are usually learned in a single step, which essentially amounts to learning a linear model as shown in Fig. 8.

ELM contains a Single Hidden Layer Feed-Forward Neural Networks (SLFNs) which will randomly select the input weights and analytically establish the output weights of SLFNs. This algorithm provides the best generalization performance at extremely fast learning speed. Usually, ELM contains an input layer, hidden layer and an output layer. The training process of ELM can be carried out in seconds or less than seconds for many applications. For all the existing classifiers, the training performed by feed forward network will take a huge chunk of time even for straightforward applications. ELM classifier can be represented as following below
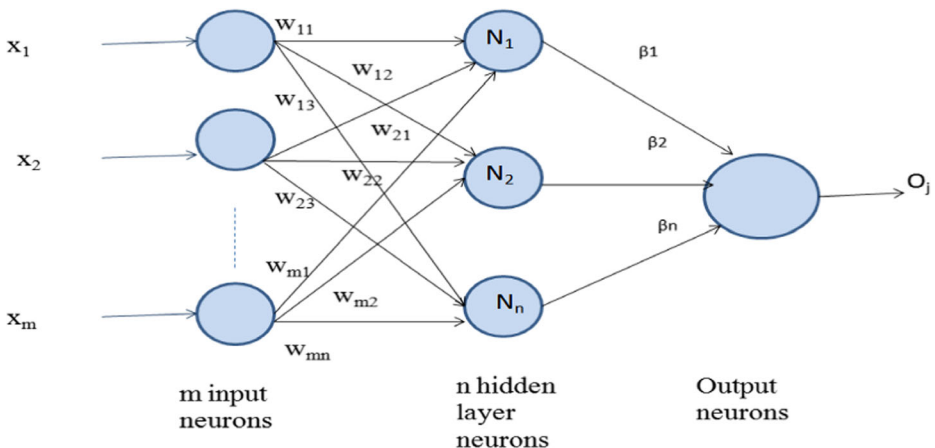

Fig. 8 ELM Architecture

$$q = W2\sigma(W1X) \tag{16}$$

where W2 is the weight matrix between the hidden layer and the output layer, W1 is the weight matrix between the input and the hidden layer, σ is the activation function. $q(x) = [q_1(x)q_2(x)\ldots q_n(x)]$ is the vector obtained from the hidden layer output for x, where n is the number of neurons in the hidden layer. $q(x)$ maps the input 'x' onto ELM feature space. W_{mn} gives the weight between the input and the hidden layer. β provides the weight between the hidden and the output layer. Gaussian radial basis activation function is used in ELM. It helps to distinguish diagonal elements of the matrix.

3.5.7 Fuzzy particle swarm optimization convolution neural network (FPSOCNN)

Convolutional Neural Networks are very similar to ordinary Neural Networks. They are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. A CNN consists of one or more convolutional layers and pooling layers. Pooling layers are also called sub sampling layers. Normally CNN are used for classification purpose. Here, CNN is used to classify the lung cancer disease. The pooling layer is used to perform down sampling. It is used to reduce the amount of computation time by reducing the extracted features in convolution layer as shown in Fig. 9.

There are two kinds of pooling layers, max pooling and average pooling. In max pooling, the value of the largest pixel is considered in the receptive field of the filter. In average pooling, the average of all the values is considered in the receptive field. The output of the pooling layer is given as input to the next convolution layer. CNN has very high computational cost for large feature maps. CNN [33] is slow to train large feature maps. To overcome the drawback of CNN, Fuzzy Particle Swarm Optimization Convolution Neural Network (FPSOCNN) is proposed. This reduces high computation cost and improves speed. The dimension reduction of image space is realized by vector of features that is created by FPSO from multidimensional image space to low dimensional feature space. This approach radically reduces the number of features for lung cancer disease classification. Instead of using max and average pooling concept in CNN, PSO and GA are applied. FPSOCNN method is compared with CNN, PSOCNN and GACNN. Figure 10 shows the architecture of FPSOCNN.
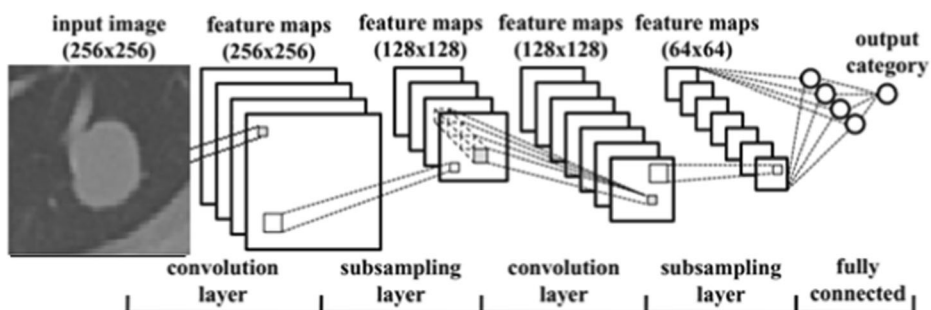
Fig. 9 Lung Cancer Detection using CNN
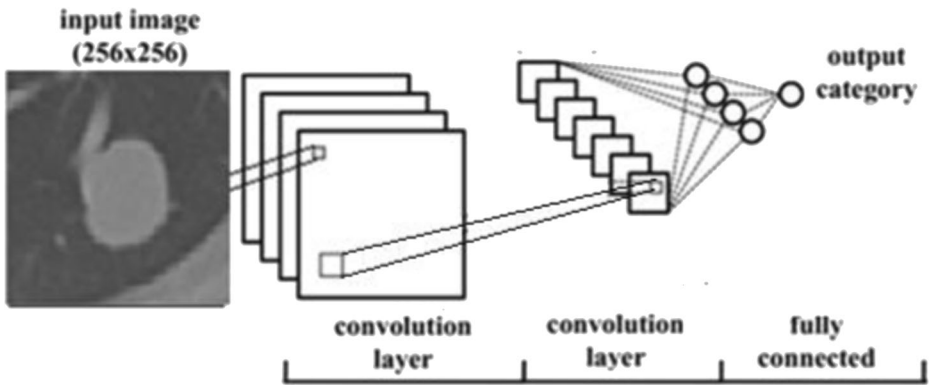
Fig. 10 Lung Cancer Detection using Proposed FPSOCNN

3.5.8 Severity finding

From the lung cancer segmented image, five severity finding parameters such as Area, Longest Diameter, Shortest Diameter, Perimeter and Elongation are depicted in Table 4. After extracting the severity parameter values, it is compared with range values of the Benign and Malignant stages to find out the severity of the input lung cancer image. The range values of these four stages are shown in Table 5. The severity results of the Lung cancer dataset are shown in Table. 6.

4 Results and discussions

4.1 Data set used

4.1.1 Real-time data set

Lung cancer images are collected from Aarthi Scan Hospital, Tirunelveli, Tamilnadu, India. Aarthi Scan Hospital dataset contains nearly 1000 lung images. The original dataset has taken from patients in Digital Imaging Communication Medicine (DICOM) images. The resolution of every image is 256×256. Here the training and testing process are performed as shown in Table 7. TrTeD1 contains high number of

Table 4 Severity Finding Parameter with equations

Intensity features used	Equation with description
Area	$\sum_{i=1}^{m} \sum_{j=0}^{n} SI_{(i,j)}$
Longest Diameter	$\sqrt{(x_1-x_2)^2 + (y_1-y_2)^2}$
Shortest Diameter	$\sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$
Perimeter	$2\pi r$
Elongation	$\frac{Longest\ Diameter}{Shortest\ Diameter}$
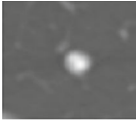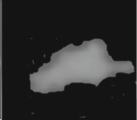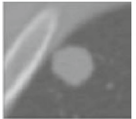
Table 5 Lung Cancer Severity Ranges of Benign and Malignant

S. No	Stage Finding Parameters	Benign	Malignant
1	Area	100–1000	1000–10,000
2	Longest Diameter	20–70	70–150
3	Shortest Diameter	10–40	40–120
4	Elongation	0.1–0.3	0.3–1.5
5	Perimeter	20–70	70–500

malignant images from the total training and testing images. TrTeD9 contains high number of benign images from the total training and testing images.

Figure 11a shows the benign pulmonary nodule with the rank of malignancy '1'. Figure 11b shows the benign pulmonary nodules with the rank of malignancy '2'. Figure 11c shows the malignant pulmonary nodules with the rank of malignancy '4'. Figure 11d shows the malignant pulmonary nodules with the rank of malignancy '5'.

Table 6 Stages of Lung Cancer Images in Dataset

S.No	FLAIR	Input Image	Lung Cancer Region	Stage Finding Parameter		Lung Cancer Stage
1	Img 1			Area	- 551	
				Longest Diameter	- 43	
				Shortest Diameter	- 44	Benign
				Elongation	- 1.0233	
				Perimeter	- 17.777	
2	Img 2			Area	- 108	
				Longest Diameter	- 40	
				Shortest Diameter	- 20	Benign
				Elongation	- 0.2704	
				Perimeter	- 11.876	
3	Img 3			Area	- 2610	
				Longest Diameter	- 77	
				Shortest Diameter	- 52	Malignant
				Elongation	- 0.6753	
				Perimeter	- 241.78	
4	Img 4			Area	- 1626	
				Longest Diameter	- 69	
				Shortest Diameter	- 49	Malignant
				Elongation	- 0.71014	
				Perimeter	- 216.66	
5	Img 5			Area	- 7123	Malignant
				Longest Diameter	- 158	
				Shortest Diameter	- 132	
				Elongation	- 0.83544	
				Perimeter	- 496.12	

Table 7 Data set details

Data set names	Taken from	Training (Tr)	Testing (Te)	Total Images
TrTeD1	Real – Time	250 (150 Malignant, 100 Benign)	750 (500 Malignant,250 Benign)	1000
TrTeD2	Real – Time	500 (350 Malignant, 150 Benign)	500 (350 Malignant, 150 Benign)	1000
TrTeD3	Real – Time	750 (500 Malignant, 250 Benign)	250(150 Malignant, 100 Benign)	1000
TrTeD4	Real – Time	900 (700 Malignant, 200 Benign)	100 (70 Malignant, 30 Benign)	1000
TrTeD5	LIDC	250 (150 Malignant, 100 Benign)	750 (500 Malignant, 250 Benign)	1000
TrTeD6	LIDC	500 (350 Malignant, 150 Benign)	500 (350 Malignant, 150 Benign)	1000
TrTeD7	LIDC	750 (500 Malignant, 250 Benign)	250 (150 Malignant, 100 Benign)	1000
TrTeD8	LIDC	900 (700 Malignant, 200 Benign)	100 (70 Malignant, 30 Benign)	1000
TrTeD9	Real – Time	250 (150 Benign, 100 Malignant)	750 (500 Benign,250 Malignant)	1000
TrTeD10	Real – Time	500 (350 Benign, 150 Malignant)	500 (350 Benign, 150 Malignant)	1000
TrTeD11	Real – Time	750 (500 Benign, 250 Malignant)	250 (150 Benign, 100 Malignant)	1000
TrTeD12	Real – Time	900 (700 Benign, 200 Malignant)	100 (70 Benign, 30 Malignant)	1000
TrTeD13	LIDC	250(150 Benign, 100 Malignant)	750 (500 Benign, 250 Malignant)	1000
TrTeD14	LIDC	500 (350 Benign, 150 Malignant)	500 (350 Benign, 150 Malignant)	1000
TrTeD15	LIDC	750 (500 Benign, 250 Malignant)	250(150 Benign, 100 Malignant)	1000
TrTeD16	LIDC	900 (700 Benign, 200 Malignant)	100 (70 Benign, 30 Malignant)	1000

4.1.2 B LIDC data set

LIDC dataset of thoracic CT scans is considered to evaluate the performance of the ELM and various classifiers for the classification of benign and malignant pulmonary nodules. LIDC [19] data set is the largest library of thoracic CT scans publicly available, which contains 1018 CT thoracic scans. Here, pulmonary nodule appears in several slices of a CT scan. The semantic rating is used for testing and training the classifier ranges from 1 to 5 by four

(a)	(b)	(c)	(d)
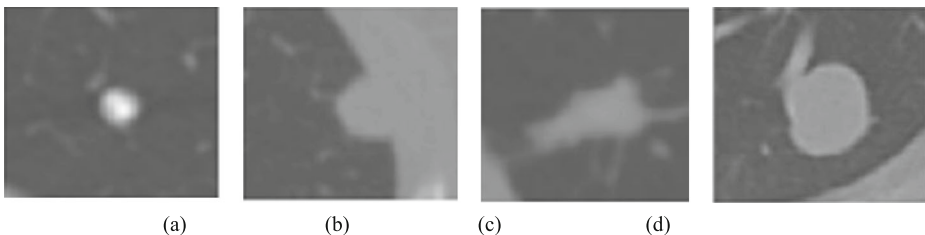
Fig. 11 Shows the examples of benign and malignant pulmonary nodules from the real-time data set.

| (a) | (b) | (c) | (d) |
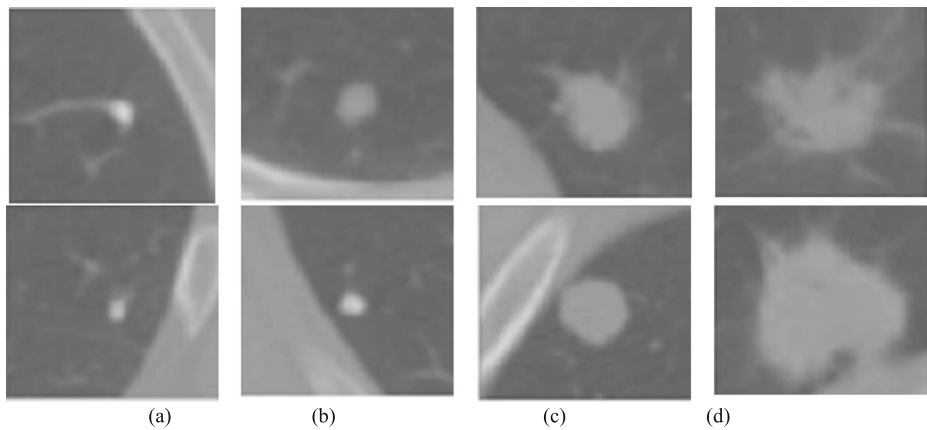
Fig. 12 Examples of rated benign and malignant pulmonary nodules from the LIDC radiologist's marks

experienced thoracic radiologists, which indicates an increasing degree of the manifestation of nodule characteristics. The ground truth data for LIDC dataset are collected the website (https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI).

In this paper, all training images are classified into malignant and benign nodules. A malignancy nodule will have scored lower than 3 are called as a benign nodule and a malignancy nodule will have scored higher than 3 are called as a malignant nodule. The pulmonary nodules with a score of 3 in malignancy are removed to avoid the ambiguousness of nodule samples. In this paper, 1000 nodules are randomly selected per class to train ELM classifier as shown in Table 7. Figure 12 shows examples of benign and malignant pulmonary nodules with different ranks of malignancy from the LIDC dataset. Figure 12a depicts the benign pulmonary nodules with the rank of malignancy '1'. Figure 12b depicts the benign pulmonary nodules with the rank of malignancy '2'. Figure 12c depicts the benign pulmonary nodules with the rank of malignancy '4'. Figure 12d depicts the benign pulmonary nodules with the rank of malignancy '5'.

4.1.3 C parameter setting

In our experiments, two dimension sizes were chosen in FPSO: d = 10 and d = 30. The number of iterations was set to 1000 and 2000 corresponding to the dimensions 10 and 30 in FPSO. The number of particles was equal to 30 and the number of trials was equal to 30 in all experiments in FPSO. The parameters used for ABC segmentation approach is shown in Table 8 and parameters used for ELM Training and Testing process is shown in Table 9.

Table 8 Test parameters for Artificial Bee Colony (ABC)

Parameters used	Value
Colony Size	100
ABC Cycles (Max Number)	5000
Employed Bees	50
Onlooker Bees	49
Random Scouts	1

Table 9 ELM Training and Testing parameters

Parameters used	Value
Number of Layers	3
Number of Input Neurons in the Layer	8
Number of Hidden Layer	10, ….50
Number of Output Layer	1
Activation Function	Radial Basis
Sum Squared Error	1

4.1.4 D performance metrics used

4.1.5 D a overlap measure (OM)

$$OM = \frac{|Se_A \cap Se_M|}{|Se_A \cup Se_M|} \tag{17}$$

Where Se_A and Se_M represents the segmentation results. $|Se_A|$ and $|Se_M|$ represents the numbers of pixels in Se_A and Se_M. $|Se_A \cap Se_M|$ represents the number of pixels in both Se_A and Se_M. $|Se_A \cap Se_M|$ represents the number of pixels in either Se_A and Se_M.

4.1.6 D b sensitivity (Sn)

Sensitivity (Sn) is defined as the fraction of malignant nodules predicted perfectly as shown in Eq. (18).

$$Sn = \frac{T_r P}{T_r P + F_a N} \tag{18}$$

4.1.7 D c specificity (Sp)

Specificity (Sp) is defined as the fraction of benign nodules predicted perfectly as shown in Eq. (19).

$$Sp = \frac{T_r n}{T_r n + F_a p} \tag{19}$$

4.1.8 D d classification accuracy (CA)

$$CA = \frac{T_r P + T_r n}{T_r P + T_r n + F_a p + F_a N} \tag{20}$$

Where T_rP represents the number of malignant nodules perfectly predicted. F_aN represents the number of malignant nodules imperfectly predicted. T_rn represents the number of benign nodules perfectly predicted. F_ap represents the number of benign nodules imperfectly predicted.

4.1.9 Error rate (ER)

$$ER = \frac{F_ap + F_aN}{T_rP + T_rn + F_ap + F_aN} \tag{21}$$

4.2 Experimental analysis

4.2.1 Experiment no 1: Analysis of segmentation approaches

In this experiment, the contribution of each segmentation approaches has been evaluated. The segmentation methods used in the work are K-Mean, FCM, Ant Colony and ABC. To evaluate the performance of this segmentation approach, the performance metric called overlap measure is used. Ideally, a good segmentation approach is expected to have a high overlap measure. Table 10 lists the overlap measures of segmentation approaches such as K-Mean, FCM, Ant Colony and ABC.

As observed from Table 10, the mean of the overlap measures obtained by the ABC segmentation method is 0.933, which is higher than that of the other existing segmentation methods. Next to ABC segmentation methods Ant colony provides efficient result with 0.931 for TrTeD1. For TrTeD7, ABC has obtained 0.95 which is more than other methods. For TrTeD16, ABC has obtained 9.959 which is 2% – 8% more than other methods. Next to ABC, Ant colony provides good results. K-means provides good results for TrTeD12 data set with 0.841 values and for TrTeD4

Table 10 Overlap measures obtained by K-Mean, FCM, Ant Colony and ABC

Data sets	Segmentation Approaches			
	K-Means	FCM	Ant Colony	ABC
TrTeD1	0.805	0.884	0.913	0.933
TrTeD2	0.821	0.9	0.929	0.949
TrTeD3	0.832	0.911	0.94	0.96
TrTeD4	0.85	0.929	0.958	0.978
TrTeD5	0.795	0.874	0.903	0.923
TrTeD6	0.811	0.89	0.919	0.939
TrTeD7	0.822	0.901	0.93	0.95
TrTeD8	0.84	0.919	0.948	0.968
TrTeD9	0.796	0.875	0.904	0.924
TrTeD10	0.812	0.891	0.92	0.94
TrTeD11	0.823	0.902	0.931	0.951
TrTeD12	0.841	0.92	0.949	0.969
TrTeD13	0.786	0.865	0.894	0.914
TrTeD14	0.802	0.881	0.91	0.93
TrTeD15	0.813	0.892	0.921	0.941
TrTeD16	0.831	0.91	0.939	0.959

Table 11 Analysis of Average accuracy, sensitivity, specificity and Error Rate of Real-time Dataset for Feature Extraction Approaches

Metrics	Accuracy	Sensitivity	Specificity	Error Rate	Accuracy	Sensitivity	Specificity	Error Rate
	TrTeD1				TrTeD2			
Intensity	82.23	91.13	94.35	17.77	82.232	91.132	94.352	17.772
HOG	83.14	92.15	93.52	16.86	83.142	92.152	93.522	16.862
Wavelet	83.66	90.74	94.74	16.34	83.662	90.742	94.742	16.342
LBP	82.58	91.4	93.91	17.42	82.582	91.402	93.912	17.422
SIFT	81.12	90.51	92.85	18.88	81.122	90.512	92.852	18.882
Zernike	82.71	89.82	93.8	17.29	82.712	89.822	93.802	17.292
Eccentricity	81.6	90.53	92.62	18.4	81.602	90.532	92.622	18.402
Curvature	80.28	89.64	91.28	19.72	80.282	89.642	91.282	19.722
Proposed	97.47	98.11	97.77	2.53	97.472	98.112	97.772	2.532
	TrTeD3				TrTeD4			
Intensity	82.252	91.152	94.372	17.792	82.273	91.173	94.393	17.813
HOG	83.162	92.172	93.542	16.882	83.183	92.193	93.563	16.903
Wavelet	83.682	90.762	94.762	16.362	83.703	90.783	94.783	16.383
LBP	82.602	91.422	93.932	17.442	82.623	91.443	93.953	17.463
SIFT	81.142	90.532	92.872	18.902	81.163	90.553	92.893	18.923
Zernike	82.732	89.842	93.822	17.312	82.753	89.863	93.843	17.333
Eccentricity	81.622	90.552	92.642	18.422	81.643	90.573	92.663	18.443
Curvature	80.302	89.662	91.302	19.742	80.323	89.683	91.323	19.763
Proposed	97.492	98.132	97.792	2.552	97.513	98.153	97.813	2.573
	TrTeD5				TrTeD6			
Intensity	82.033	90.933	94.153	17.967	81.853	90.753	93.973	18.147
HOG	82.943	91.953	93.323	17.057	82.763	91.773	93.143	17.237
Wavelet	83.463	90.543	94.543	16.537	83.283	90.363	94.363	16.717
LBP	82.383	91.203	93.713	17.617	82.203	91.023	93.533	17.797
SIFT	80.923	90.313	92.653	19.077	80.743	90.133	92.473	19.257
Zernike	82.513	89.623	93.603	17.487	82.333	89.443	93.423	17.667
Eccentricity	81.403	90.333	92.423	18.597	81.223	90.153	92.243	18.777
Curvature	80.083	89.443	91.083	19.917	79.903	89.263	90.903	20.097
Proposed	97.273	97.913	97.573	2.727	97.093	97.733	97.393	2.907
	TrTeD7				TrTeD8			
Intensity	81.623	90.523	93.743	18.377	81.783	90.683	93.903	18.217
HOG	82.533	91.543	92.913	17.467	82.693	91.703	93.073	17.307
Wavelet	83.053	90.133	94.133	16.947	83.213	90.293	94.293	16.787
LBP	81.973	90.793	93.303	18.027	82.133	90.953	93.463	17.867
SIFT	80.513	89.903	92.243	19.487	80.673	90.063	92.403	19.327
Zernike	82.103	89.213	93.193	17.897	82.263	89.373	93.353	17.737
Eccentricity	80.993	89.923	92.013	19.007	81.153	90.083	92.173	18.847
Curvature	79.673	89.033	90.673	20.327	79.833	89.193	90.833	20.167
Proposed	96.863	97.503	97.163	3.137	97.023	97.663	97.323	2.977
	TrTeD9				TrTeD10			
Intensity	81.873	90.773	93.993	18.127	81.863	90.763	93.983	18.137
HOG	82.783	91.793	93.163	17.217	82.773	91.783	93.153	17.227
Wavelet	83.303	90.383	94.383	16.697	83.293	90.373	94.373	16.707
LBP	82.223	91.043	93.553	17.777	82.213	91.033	93.543	17.787
SIFT	80.763	90.153	92.493	19.237	80.753	90.143	92.483	19.247
Zernike	82.353	89.463	93.443	17.647	82.343	89.453	93.433	17.657
Eccentricity	81.243	90.173	92.263	18.757	81.233	90.163	92.253	18.767
Curvature	79.923	89.283	90.923	20.077	79.913	89.273	90.913	20.087
Proposed	97.113	97.753	97.413	2.887	97.103	97.743	97.403	2.897
	TrTeD11				TrTeD12			
Intensity	81.893	90.793	94.013	18.107	81.953	90.853	94.073	18.047
HOG	82.803	91.813	93.183	17.197	82.863	91.873	93.243	17.137
Wavelet	83.323	90.403	94.403	16.677	83.383	90.463	94.463	16.617
LBP	82.243	91.063	93.573	17.757	82.303	91.123	93.633	17.697

Table 11 (continued)

Metrics	Accuracy	Sensitivity	Specificity	Error Rate	Accuracy	Sensitivity	Specificity	Error Rate
SIFT	80.783	90.173	92.513	19.217	80.843	90.233	92.573	19.157
Zernike	82.373	89.483	93.463	17.627	82.433	89.543	93.523	17.567
Eccentricity	81.263	90.193	92.283	18.737	81.323	90.253	92.343	18.677
Curvature	79.943	89.303	90.943	20.057	80.003	89.363	91.003	19.997
Proposed	97.133	97.773	97.433	2.867	97.193	97.833	97.493	2.807
	TrTeD13				TrTeD14			
Intensity	81.973	90.873	94.093	18.027	81.903	90.803	94.023	18.097
HOG	82.883	91.893	93.263	17.117	82.813	91.823	93.193	17.187
Wavelet	83.403	90.483	94.483	16.597	83.333	90.413	94.413	16.667
LBP	82.323	91.143	93.653	17.677	82.253	91.073	93.583	17.747
SIFT	80.863	90.253	92.593	19.137	80.793	90.183	92.523	19.207
Zernike	82.453	89.563	93.543	17.547	82.383	89.493	93.473	17.617
Eccentricity	81.343	90.273	92.363	18.657	81.273	90.203	92.293	18.727
Curvature	80.023	89.383	91.023	19.977	79.953	89.313	90.953	20.047
Proposed	97.213	97.853	97.513	2.787	97.143	97.783	97.443	2.857
	TrTeD15				TrTeD16			
Intensity	81.933	90.833	94.053	18.067	81.973	90.873	94.093	18.027
HOG	82.843	91.853	93.223	17.157	82.883	91.893	93.263	17.117
Wavelet	83.363	90.443	94.443	16.637	83.403	90.483	94.483	16.597
LBP	82.283	91.103	93.613	17.717	82.323	91.143	93.653	17.677
SIFT	80.823	90.213	92.553	19.177	80.863	90.253	92.593	19.137
Zernike	82.413	89.523	93.503	17.587	82.453	89.563	93.543	17.547
Eccentricity	81.303	90.233	92.323	18.697	81.343	90.273	92.363	18.657
Curvature	79.983	89.343	90.983	20.017	80.023	89.383	91.023	19.977
Proposed	97.173	97.813	97.473	2.827	97.213	97.853	97.513	2.787

data set with 0.929 values. ABC has obtained 0.923 value for TrTeD5 data set while K-means, FCM and Ant colony provides less results.

4.2.2 Experiment no 2: Analysis of feature extraction approaches

In this experiment, the contributions of each feature extraction approaches which are used in this work are evaluated. To evaluate the performance of these feature extraction approaches, the performance metrics, namely, accuracy, sensitivity, specificity and error rate measures are used. Ideally, a good feature extraction approach is expected to have a high accuracy, high sensitivity, high specificity and low error rate. Table 11 depicts accuracy, sensitivity, specificity and error rate measures of various feature extraction approaches. For TrTeD1 to TrTeD16, the proposed method depicts good results in accuracy, sensitivity, specificity and error rate.

As observed from Table 11, accuracy of the proposed features is 97.47 for TrTeD1, which is higher than that of the individual feature extraction methods. As well as, in the proposed feature extraction approach, the error rate is 2.53 for TrTeD1 which is lower than that of the traditional individual feature extraction methods. For TrTeD16, Accuracy obtained is 97.213, Sensitivity is 97.853, Specificity is 97.513 and error rate is 2.787 which is very high compared to other methods.

The accuracy has been improved from (7–11) % compared to the other feature extraction methods. Accuracy, Sensitivity and specificity are more compared to the other existing methods. Next to the proposed, HOG provides good results in accuracy and Sensitivity. For specificity, Wavelet provides good result next to the proposed method.

Table 12 Analysis of Average accuracy, sensitivity, specificity and Error Rate of Real-time Dataset for Classifier Approaches

Metrics	Accuracy	Sensitivity	Specificity	Error Rate	Accuracy	Sensitivity	Specificity	Error Rate
	TrTeD1				TrTeD2			
SVM	94.23	95.61	94.91	5.77	94.25	95.63	94.93	5.75
Bagging	89.26	90.75	89.95	10.74	89.28	90.77	89.97	10.72
Naive Bayes	85.71	86.63	86.42	14.29	85.73	86.65	86.44	14.27
KNN	84.23	85.21	84.96	15.77	84.25	85.23	84.98	15.75
AdaBoost	91.76	92.82	92.43	8.24	91.78	92.84	92.45	8.22
ELM	97.14	98.39	97.87	2.86	97.16	98.41	97.89	2.84
CNN	98.18	98.43	98.76	1.82	98.21	98.46	98.79	1.79
GACNN	98.76	98.88	98.93	1.24	98.79	98.91	98.96	1.21
PSOCNN	98.91	98.95	98.99	1.09	98.94	98.98	99.02	1.06
FPSOCNN	99.23	99.31	99.43	0.77	99.13	99.24	99.36	0.87
	TrTeD3				TrTeD4			
SVM	94.272	95.652	94.952	5.728	94.291	95.671	94.971	5.709
Bagging	89.302	90.792	89.992	10.698	89.321	90.811	90.011	10.679
Naive Bayes	85.752	86.672	86.462	14.248	85.771	86.691	86.481	14.229
KNN	84.272	85.252	85.002	15.728	84.291	85.271	85.021	15.709
AdaBoost	91.802	92.862	92.472	8.198	91.821	92.881	92.491	8.179
ELM	97.182	98.432	97.912	2.818	97.201	98.451	97.931	2.799
CNN	98.2	98.45	98.78	1.8	98.18	98.43	98.76	1.82
GACNN	98.78	98.9	98.95	1.22	98.76	98.88	98.93	1.24
PSOCNN	98.93	98.97	99.01	1.07	98.91	98.95	98.99	1.09
FPSOCNN	99.12	99.23	99.35	0.88	99.1	99.21	99.33	0.9
	TrTeD5				TrTeD6			
SVM	94.161	94.971	94.841	5.839	94.051	94.861	94.731	5.949
Bagging	89.191	90.011	89.881	10.809	89.081	89.901	89.771	10.919
Naive Bayes	85.641	86.481	86.351	14.359	85.531	86.371	86.241	14.469
KNN	84.161	85.021	84.891	15.839	84.051	84.911	84.781	15.949
AdaBoost	91.691	92.491	92.361	8.309	91.581	92.381	92.251	8.419
ELM	97.071	97.931	97.801	2.929	96.961	97.821	97.691	3.039
CNN	98.05	98.76	98.63	1.95	97.94	98.65	98.52	2.06
GACNN	98.63	98.93	98.8	1.37	98.52	98.82	98.69	1.48
PSOCNN	98.78	98.99	98.86	1.22	98.67	98.88	98.75	1.33
FPSOCNN	99.18	99.33	99.2	0.82	99.07	99.22	99.09	0.93
	TrTeD7				TrTeD8			
SVM	94.121	94.931	94.801	5.879	94.171	94.981	94.851	5.829
Bagging	89.151	89.971	89.841	10.849	89.201	90.021	89.891	10.799
Naive Bayes	85.601	86.441	86.311	14.399	85.651	86.491	86.361	14.349
KNN	84.121	84.981	84.851	15.879	84.171	85.031	84.901	15.829
AdaBoost	91.651	92.451	92.321	8.349	91.701	92.501	92.371	8.299
ELM	97.031	97.891	97.761	2.969	97.081	97.941	97.811	2.919
CNN	98.01	98.72	98.59	1.99	98.06	98.77	98.64	1.94
GACNN	98.59	98.89	98.76	1.41	98.64	98.94	98.81	1.36
PSOCNN	98.74	98.95	98.82	1.26	98.79	99	98.87	1.21
FPSOCNN	99.14	99.29	99.16	0.86	99.19	99.34	99.21	0.81
	TrTeD9				TrTeD10			
SVM	94.291	95.101	94.971	5.709	94.221	95.031	94.901	5.779
Bagging	89.321	90.141	90.011	10.679	89.251	90.071	89.941	10.749
Naive Bayes	85.771	86.611	86.481	14.229	85.701	86.541	86.411	14.299
KNN	84.291	85.151	85.021	15.709	84.221	85.081	84.951	15.779
AdaBoost	91.821	92.621	92.491	8.179	91.751	92.551	92.421	8.249
ELM	97.201	98.061	97.931	2.799	97.131	97.991	97.861	2.869
CNN	98.18	98.89	98.76	1.82	98.11	98.82	98.69	1.89
GACNN	98.76	99.06	98.93	1.24	98.69	98.99	98.86	1.31
PSOCNN	98.91	99.12	98.99	1.09	98.84	99.05	98.92	1.16
FPSOCNN	99.31	99.46	99.33	0.69	99.24	99.39	99.26	0.76

Table 12 (continued)

Metrics	Accuracy	Sensitivity	Specificity	Error Rate	Accuracy	Sensitivity	Specificity	Error Rate
	TrTeD11				TrTeD12			
SVM	94.171	94.981	94.851	5.829	94.141	94.951	94.821	5.829
Bagging	89.201	90.021	89.891	10.799	89.171	89.991	89.861	10.799
Naive Bayes	85.651	86.491	86.361	14.349	85.621	86.461	86.331	14.349
KNN	84.171	85.031	84.901	15.829	84.141	85.001	84.871	15.829
AdaBoost	91.701	92.501	92.371	8.299	91.671	92.471	92.341	8.299
ELM	97.081	97.941	97.811	2.919	97.051	97.911	97.781	2.919
CNN	98.06	98.77	98.64	1.94	98.03	98.74	98.61	1.94
GACNN	98.64	98.94	98.81	1.36	98.61	98.91	98.78	1.36
PSOCNN	98.79	99	98.87	1.21	98.76	98.97	98.84	1.21
FPSOCNN	99.19	99.34	99.21	0.81	99.16	99.31	99.21	0.81
	TrTeD13				TrTeD14			
SVM	94.141	94.951	94.821	5.859	94.141	94.951	94.821	5.859
Bagging	89.171	89.991	89.861	10.829	89.171	89.991	89.861	10.829
Naive Bayes	85.621	86.461	86.331	14.379	85.621	86.461	86.331	14.379
KNN	84.141	85.001	84.871	15.859	84.141	85.001	88.876	15.859
AdaBoost	91.671	92.471	92.341	8.329	91.671	92.471	92.341	8.329
ELM	97.051	97.911	97.781	2.949	97.051	97.911	97.781	2.949
CNN	98.03	98.74	98.61	1.97	98.03	98.74	98.61	1.97
GACNN	98.61	98.91	98.78	1.39	98.61	98.91	98.78	1.39
PSOCNN	98.76	98.97	98.84	1.24	98.76	98.94	98.84	1.24
FPSOCNN	99.16	99.31	99.18	0.84	99.16	99.31	99.19	0.84
	TrTeD15				TrTeD16			
SVM	94.121	94.931	94.801	5.879	94.141	94.951	94.821	5.859
Bagging	89.151	89.971	89.841	10.849	89.171	89.991	89.861	10.829
Naive Bayes	85.601	86.441	86.311	14.399	85.621	86.461	86.331	14.379
KNN	84.121	84.981	84.851	15.879	84.141	85.001	84.871	15.859
AdaBoost	91.651	92.451	92.321	8.349	91.671	92.471	92.341	8.329
ELM	97.031	97.891	97.761	2.969	97.051	97.911	97.781	2.949
CNN	98.01	98.72	98.59	1.99	98.03	98.74	98.61	1.97
GACNN	98.59	98.89	98.76	1.41	98.61	98.91	98.78	1.39
PSOCNN	98.74	98.95	98.82	1.26	98.76	98.97	98.84	1.24
FPSOCNN	99.14	99.29	99.16	0.86	99.16	99.31	99.18	0.84

4.2.3 Experiment no 3: Analysis of classifier approaches

In this experiment, the contributions of various classifiers which are used in this work are evaluated. To evaluate the performance of these classifiers, the performance metrics used are accuracy, sensitivity, specificity and error rate. Ideally, a good classifier is expected to have a

Table 13 Analysis of Average accuracy, sensitivity and specificity for existing works for Whole LIDC data set

Existing works	Average Accuracy (%)	Average Sensitivity (%)	Average Specificity (%)
Suren et al. [20]	92.24	93.65	91.23
Taruna et al. [1]	97.12	97.14	96.33
Hiram Madero et al. [24]	89.52	90.90	88.15
Hiram Madero et al. [23]	95.66	96.15	97.32
Hong Shao et al. [28]	90.35	89.47	90.52
Silva et al. [11]	94.78	94.66	95.14
Costa et al. [14]	91.81	93.42%	91.21
Proposed	95.62	97.93	96.32

Table 14 Analysis of Average accuracy, sensitivity and specificity for existing works for Whole Real Time data set

| Existing works	Average Accuracy (%)	Average Sensitivity (%)	Average Specificity (%)
Suren et al. [20]	90.42	91.27	89.14
Taruna et al. [1]	94.27	95. 42	93.41
Hiram Madero et al. [24]	87.42	88.48	87.38
Hiram Madero et al. [23]	92.27	93.25	91.74
Hong Shao et al. [28]	88.64	85.56	84.37
Silva et al. [11]	93.14	93.53	94.28
Costa et al. [14]	90.74	92.13	90.57
Proposed	94.97	96.68	95.89

high accuracy, high sensitivity, high specificity and low error rate. Table 12 lists the accuracy, sensitivity, specificity and error rate measures of various classifiers.

As observed from Table 12, accuracy, sensitivity, error rate and specificity for 16 data sets are provided. For TrTeD1, the proposed method got an accuracy of 99.23, for sensitivity the value obtained is 99.31, for specificity the result value is 99.43 and error rate is 0.77. For TrTeD16, the proposed method got an accuracy of 99.16, for sensitivity the value obtained is 99.31, for specificity the result value is 99.18 and error rate is 0.84.

4.2.4 Experiment no 4: Analysis of proposed method with existing works

In this experiment, the contribution of proposed method is evaluated with the existing works. To evaluate the performance of this proposed approach, the performance metrics used are accuracy, sensitivity, specificity and error rate. Ideally, a good proposed approach is expected to have a high accuracy, sensitivity, and specificity. Table 13 depicts the results with the state-of-the-art methods for LIDC data set. The proposed method shows 95.62% accuracy, 97.93% sensitivity and 96.32% specificity for LIDC data set.

As observed from Table 14, the average accuracy value of the proposed obtained is 94.97, which is higher than of the all existing works. As well as, average sensitivity of the proposed method obtained is 96.68, which is higher than of the all existing works. The specificity value of the proposed method obtained is 95.89 which are higher than of the all existing works. So, from the Table 14, it is concluded that the proposed method provides good results than other existing works.
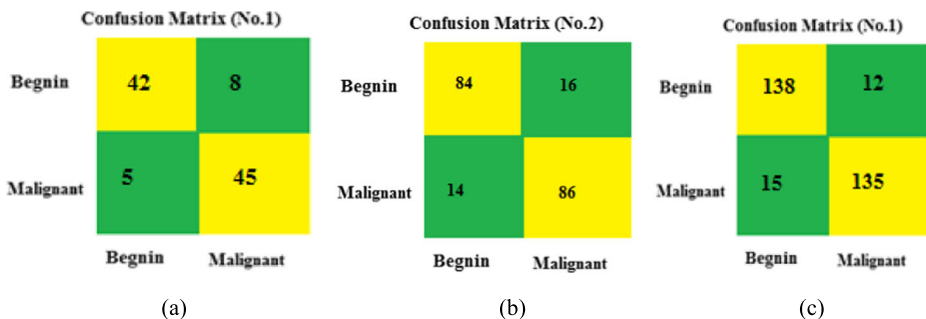
Figure 13 Results of confusion matrices on different numbers of testing samples **a** Confusion matrix on 100 testing samples, **b** Confusion matrix on 200 testing samples, **c** Confusion matrix on 300 testing samples

4.2.5 Experiment no 5: Analysis of proposed method with confusion matrix

To verify the performance of the proposed classification method, the confusion matrix [6, 58, 59] is used as the metric. Here there are two classes: benign nodules and malignant nodules. Therefore, the confusion matrix with a size of 2×2 is used. Figure 13 depicts the confusion matrices on different numbers of testing samples.

4.2.6 Computational complexity

The computational complexity of CNN is calculated by using big o notation. The computational complexity of CNN is calculated using Eq. (22)

$$\text{CNN Time Complexity} = n_{layers} \text{Convolution filer size.Pooling size} \qquad (22)$$

$$\text{CNN} = O\left(n^2 . n^4 . n^2\right) = O\left(n^8\right) \qquad (23)$$

The computational complexity of PSOCNN is found by using Eq. (24)

$$\text{PSOCNN Time Complexity} = n_{layers}. \text{ Convolution filer size} \qquad (24)$$

$$\text{PSOCNN} = O\left(n^1 . n^4\right) = O\left(n^5\right) \qquad (25)$$

The computational complexity of GACNN is found by using Eq. (26)

$$\text{GACNN Time Complexity} = n_{layers}. \text{ Convolution filer size} \qquad (26)$$

$$\text{GACNN} = O\left(n^1 . n^4\right) = O\left(n^5\right) \qquad (27)$$

The computational complexity of FPSOCNN is found by using Eq. (28)

$$\text{FPSOCNN Time Complexity} = n_{layers}. \text{Convolution filer size} \qquad (28)$$

$$\text{FPSOCNN} = O\left(n^1 . n^4\right) = O\left(n^5\right) \qquad (29)$$

5 Conclusion

This work is to detect the cancerous lung nodules from the given input lung image and to classify the lung cancer and its severity. To detect the location of the cancerous lung nodules, this work uses novel Deep learning methods. Here, features are classified using Deep learning. A novel FPSOCNN is proposed which reduces computational complexity of CNN. This work uses best feature extraction techniques such as Histogram of oriented Gradients (HoG), wavelet transform-based features, Local Binary Pattern (LBP), Scale Invariant Feature

Transform (SIFT) and Zernike Moment. After extracting texture, geometric, volumetric and intensity features, Fuzzy Particle Swarm Optimization (FPSO) algorithm is applied for selecting the best feature. An additional valuation is performed on another dataset coming from Arthi Scan Hospital which is a real-time data set. From the experimental results, it is shown that novel FPSOCNN performs better than other techniques.

In future, further improvement will be performed in the classification performance of pulmonary nodules and optimise the proposed model. In addition, the further work will be grading the images based on the degree of the malignancy of pulmonary nodules, which is of valuable significance for the diagnosis and treatment of lung cancer in clinical applications.

References

1. Aggarwal T, Furqan A, Kalra K (2015) Feature extraction and LDA based classification of lung nodules in chest CT scan images. IEEE, International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1189–1193
2. Akram S, Javed MY, Hussain A, Riaz F, Usman Akram M (2015) Intensity-based statistical features for classification of lungs CT scan nodules using artificial intelligence techniques. Journal of Experimental & Theoretical Artificial Intelligence 27(6):737–751. https://doi.org/10.1080/0952813X.2015.1020526
3. Alakwaa W, Nassef M, Badr A (2017) Lung Cancer detection and classification with 3D convolutional neural network (3D-CNN). International Journal of Advanced Computer Science and Applications (IJACSA) 8(8):409–417
4. Ani Brown Mary N, Dejey D (2018) 'Classification of coral reef submarine images and videos using a novel Z with tilted Z local binary pattern (Z⊕TZLBP)', springer. Wirel Pers Commun 98(3):2427–2459. https://doi.org/10.1007/s11277-017-4981-x
5. Ani Brown Mary N, Dharma D (2017) 'Coral reef image classification employing improved LDP for feature extraction', Elsevier. J Vis Commun Image Represent 49(C):225–242. https://doi.org/10.1016/j.jvcir.2017.09.008
6. Ani Brown Mary N, Dharma D (2018) A novel framework for real-time diseased coral reef image classification', Springer. Multimed Tools Appl:1–39. https://doi.org/10.1007/s11042-018-6673-2
7. Bhuvaneswari BT (2015) Detection of Cancer in lung with K-NN classification using genetic algorithm', Elsevier. Procedia Mater Sci 10:433–440
8. Brown A, Mary N, Dejey D (2018) 'Classification of coral reef submarine images and videos using a novel Z with tilted Z local binary pattern (Z⊕TZLBP)', springer. Wirel Pers Commun 98(3):2427–2459. https://doi.org/10.1007/s11277-017-4981-x
9. Chabat F, Yang G-Z, Hansell DM (2003) Obstructive lung diseases: texture classification for differentiation at CT1. Radiology 228(3):871–877
10. Chen H, Xu Y, Ma Y, Ma B (2010) Neural Network Ensemble-Based Computer-Aided Diagnosis for Differentiation of Lung Nodules on CT Images. Acad Radiol 17(5)
11. Da Silva GLF, da Silva Neto OP, Silva AC, de Paiva Marcelo Gattass AC (2017) "Lung nodules diagnosis based on evolutionary convolutional neural network", springer. Multimed Tools Appl 76(18):19039–19055. https://doi.org/10.1007/s11042-017-4480-9
12. da Silva GLF, de Carvalho Filho AO, Silva AC, de Paiva AC, Gattass M (2016) Taxonomic indexes for differentiating malignancy of lung nodules on CT images. Research on Biomedical Engineering 32(3):263–272
13. de Carvalho Filho AO, Silva AC, de Paiva AC, Nunes RA, Gattass M (2016) Lung-nodule classification based on computed tomography using taxonomic diversity indexes and an SVM. Springer, Journal of Signal Processing Systems, DOI 87:179–196. https://doi.org/10.1007/s11265-016-1134-5
14. de Sousa Costa RW, da Silva GLF, de Carvalho Filho AO, Silva AC, de Paiva Marcelo Gattass AC (2018) "Classification of malignant and benign lung nodules using taxonomic diversity index and phylogenetic distance", springer. Med Biol Eng Comput 56(11):2125–2136
15. Dhaware BU, Pise AC, (2016) Lung Cancer Detection Using Bayasein Classifier and FCM Segmentation. IEEE, International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), pp. 170–174
16. Ignatious S, Joseph R (2015) Computer Aided Lung Cancer Detection System. IEEE, Proceedings of 2015 Global Conference on Communication Technologies (GCCT 2015), pp. 555–558.

17. Jin X-Y, Zhang Y-C, Jin Q-L (2016) Pulmonary nodule detection based on CT images using Convolution neural network. IEEE, 9th International Symposium on Computational Intelligence and Design, pp. 202–204.
18. Kumar D, Wong A, Clausi DA (2015) "Lung nodule classification using deep features in CT images", IEEE, 12th conference on computer robot vision, pp 133-138. DOI. https://doi.org/10.1109/CRV.2015.25
19. Li X-X, Li B, Tian L-F, Zhang L (2018) Automatic benign and malignant classification of pulmonary nodules in thoracic computed tomography based on RF algorithm. IET Image Process. https://doi.org/10.1049/iet-ipr.2016.1014
20. Makaju S, Prasad AA, Elchouemi S (2018) Lung Cancer detection using CT scan images. Elsevier, Procedia Computer Science 125:107–114
21. Nie L, Wang M, Zhang L, Yan S, Zhang B, Chua T-S (2015) Disease inference from health-related questions via sparse deep learning. IEEE Trans Knowl Data Eng 27(8):2107–2119
22. Nie L, Zhang L, Yang Y, Wang M, Hong R, Chua T-S (2015) Beyond doctors: future health prediction from multimedia and multimodal observations. proceedings of the 23rd ACM international conference on multimedia.
23. Orozco HM, Villegas OOV, Maynez LO, Sanchez VGC, de Jesus Ochoa Dominguez H (2012) Lung Nodule CLASSIFICATION in Frequency Domain Using Support Vector Machine. IEEE, In international conference on information science, signal processing and their application.
24. Orozco HM, Villegas OOV, Sánchez VGC, de Jesús Ochoa Domínguez H, de Jesús Nandayapa Alfaro M (2015) Automated system for lung nodules classification based on wavelet feature descriptor and support vector machine. Biomed Eng 14(9):1–20. https://doi.org/10.1186/s12938-015-0003-y
25. Park SC, Tan J, Wang X, Lederman D, Leader JK, Kim SH, Zheng B (2011) Computer-aided detection of early interstitial lung diseases using low-dose CT images', Iop Publishing. Phys Med Biol 56:1139–1153. https://doi.org/10.1088/0031-9155/56/4/016
26. Roy TS, Sirohi N, Patle A (2015) Classification of Lung Image and Nodule Detection Using Fuzzy Inference System. IEEE, International Conference on Computing, Communication and Automation (ICCCA2015), pp. 1204–1207
27. Sangamithraa, Govindaraju (2016) Lung tumour detection and classification using EK-mean clustering. IEEE WiSPNET
28. Shao H, Cao L, Liu Y (2012) A detection approach for solitary pulmonary nodules based on CT images. IEEE, 2nd international conference on computer science and network technology.
29. Song QZ, Zhao L, Luo XK, Dou XC (2017) Using deep learning for classification of lung nodules on computed tomography images. Journal of healthcare engineering. https://doi.org/10.1155/2017/8314740
30. Sun W, Zheng B, Qian W (2016) "computer aided lung cancer diagnosis with deep learning algorithms" International Society for Optics and Photonics, medical imaging : computer-aided diagnosis. Vol. 9785
31. Suzuki K, Li F, Sone S, Doi K (2005) Computer-aided diagnostic scheme for distinction between benign and malignant nodules in thoracic low-dose CT by use of massive training artificial neural network. IEEE Trans Med Imaging 24(9):1138–1150
32. Dong-ping Tian and Nai-qian Li, 2009, 'Fuzzy Particle Swarm Optimization Algorithm', IEEE, International Joint Conference on Artificial Intelligence, pp. 263–267.
33. Van Ginneken B, Setio AAA, Jacobs C, Ciompi F (2015) Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans. IEEE 12th International Symposium on Biomedical Imaging (ISBI). doi:10.1109/isbi.2015.7163869
34. Zhang B, Allebach JP (2008) Adaptive Bilateral Filter for Sharpness Enhancement and Noise Removal. IEEE Trans Image Process 17(5)
35. Zhang L, Zhang Q, Du Member B, Huang X, Tang YY, Tao D (2016) Simultaneous spectral-spatial feature selection and extraction for Hyperspectral images. IEEE Transactions on Cybernetics
36. Zhang L, Zhang Q, Zhang L, Tao D, Huang X, Bo D (2014) Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding", Elsevier. Pattern Recogn. https://doi.org/10.1016/j.patcog.2014.12.016
37. Zhou Z-H, Jiang Y, Yang Y-B, Chen S-F (2002) Lung cancer cell identification based on artificial neural network ensembles', Elsevier. Artif Intell Med 24:25–36
38. Zhu Y, Tan Y, Hua Y, Wang M, Zhang G, Zhang J (2010) Feature selection and performance evaluation of support vector machine (SVM)-based classifier for differentiating benign and malignant pulmonary nodules by computed tomography. J Digit Imaging 23(1):51–65

A. Asuntha is an Assistant Professor in the department of Electronics & Instrumentation Engineering, SRM Institutem of Science and Technology, Tamilnadu,India. The author has completed B.E and M.E from Annamalai University. She has many publications and is presently workings on more papers in image processing. Her other areas of specialization includes Nonlinear system design, control system and process control applications.

Andy Srinivasan received the Ph.D. degree in Anna University, Chennai, India. He is currently working as a Professor of Electronics and Instrumentation Department in Valliammai Engineering College, Kanchipuram, Tamilnadu, India. Under his guidance, many young researchers have completed their research works. His research areas include process control, system identification, neural network, fuzzy logic, image processing and machine learning.