



## OPEN Lung cancer detection with machine learning classifiers with multi-attribute decision-making system and deep learning model

T. Meeradevi<sup>1</sup>, S. Sasikala<sup>1</sup>, L. Murali<sup>2</sup>, N. Manikandan<sup>2</sup> & Krishnaraj Ramaswamy<sup>3,4</sup>✉

Diseases of the airways and the other parts of the lung cause chronic respiratory diseases. The major cause of lung disease is tobacco smoke, along with risk factors such as dust, air pollution, chemicals, and frequent lower respiratory infections during childhood. Early detection of these diseases requires the analysis of medical images, which would aid doctors in providing effective treatment. This paper aims to classify lung X-ray images as benign or malignant and to identify the type of disease, such as Atelectasis, Infiltration, Nodule, and Pneumonia, if the disease is malignant. Machine learning (ML) approaches, combined with a multi-attribute decision-making method called Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), are used to rank different classifiers. Additionally, the deep learning (DL) model Inception v3 is proposed. This method ranks the SVM with RBF as the best classifier among the others used in this approach. Furthermore, the results show that the deep learning model achieves the best accuracy of 97.05%, which is 11.8% higher than the machine learning approach using the same dataset.

**Keywords** Machine learning, Deep learning, Lung disease, TOPSIS

Lung disorders arise from diseases affecting the air passages and other components of the respiratory system. Pulmonary diseases, including pneumonia, tuberculosis and COVID-19, affect millions worldwide, with asthma alone impacting about 334 million individuals. These conditions lead to significant mortality, with 1.4 million deaths annually from TB, 1.6 million from lung cancer and millions more from pneumonia. The COVID-19 pandemic has further highlighted the burden on global healthcare systems. Early detection of these diseases is essential for improving outcomes. While traditional diagnostic techniques, including skin tests and X-rays, have been widely utilized, recent advancements in deep learning show great potential in identifying lung diseases through the analysis of medical imaging<sup>21</sup>.

The high mortality rate of lung cancer is primarily attributed to its late detection. This delay occurs because early-stage lung cancer typically presents with non-specific symptoms and is difficult to identify using traditional diagnostic methods. These methods, which primarily rely on radiological imaging such as chest X-rays and CT scans can be limited by the quality of the imaging, the experience of the radiologist, and the subjective nature of image interpretation. As a result, there is a need for more accurate and efficient diagnostic tools that can assist in the early detection of lung cancer, thereby improving patient outcomes and survival rates.

Artificial intelligence (AI), particularly in the fields of machine learning (ML) and deep learning (DL) offer promising solutions to the challenges of lung cancer detection. Machine learning algorithms are developed to analyze data, recognize patterns, and make predictions, thereby aiding in disease diagnosis. These algorithms can process a wide range of patient information, including demographic details, medical histories, and imaging results, to detect subtle signs of lung cancer that might be overlooked by human observers. By identifying patterns linked to lung cancer risk, machine learning serves as an early warning system for clinicians, enhancing early detection and improving patient outcomes.

Deep learning techniques are highly effective in processing and analyzing complex image data. Convolutional Neural Networks (CNNs), in particular, excel at medical imaging tasks due to their ability to automatically learn and extract features from images, such as edges, textures, and shapes indicative of cancerous tissues. By

<sup>1</sup>Department of ECE, Kongu Engineering College, Erode, Tamil Nadu, India. <sup>2</sup>P.A.College of Engineering and Technology, Pollachi, Tamil Nadu, India. <sup>3</sup>Department of Mechanical Engineering, College of Engineering Science, Dambi Dollo University, Dambi Dollo, Ethiopia. <sup>4</sup>Center For Global Health Research, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, India. ✉email: dr.krishnarajdirectorcei@dadu.edu.et

analyzing large volumes of imaging data, deep learning models can detect even minute abnormalities that may indicate the presence of lung cancer. This capability enhances the accuracy of imaging diagnostics, significantly reducing the likelihood of both false negatives and false positives.

This study employs machine learning classifiers, including k-Nearest Neighbors (kNN), Support Vector Machine (SVM), Decision Tree (DT), and Naïve Bayes, along with the deep learning model Inception V3, to classify lung X-ray images. The classification aims to distinguish between benign and malignant cases and, in the case of malignancy, identify the specific type of disease.

### Lung diseases

There are six primary types of lung diseases.

**Atelectasis:** Under certain conditions, the alveoli in the lungs may become deflated, leading to reduced gas exchange due to partial or complete lung collapse. This condition typically affects a portion or the entirety of one lung.

**Cardiomegaly:** Cardiomegaly is an engorged heart not to be considered as a disease. But sometimes it may lead to severity.

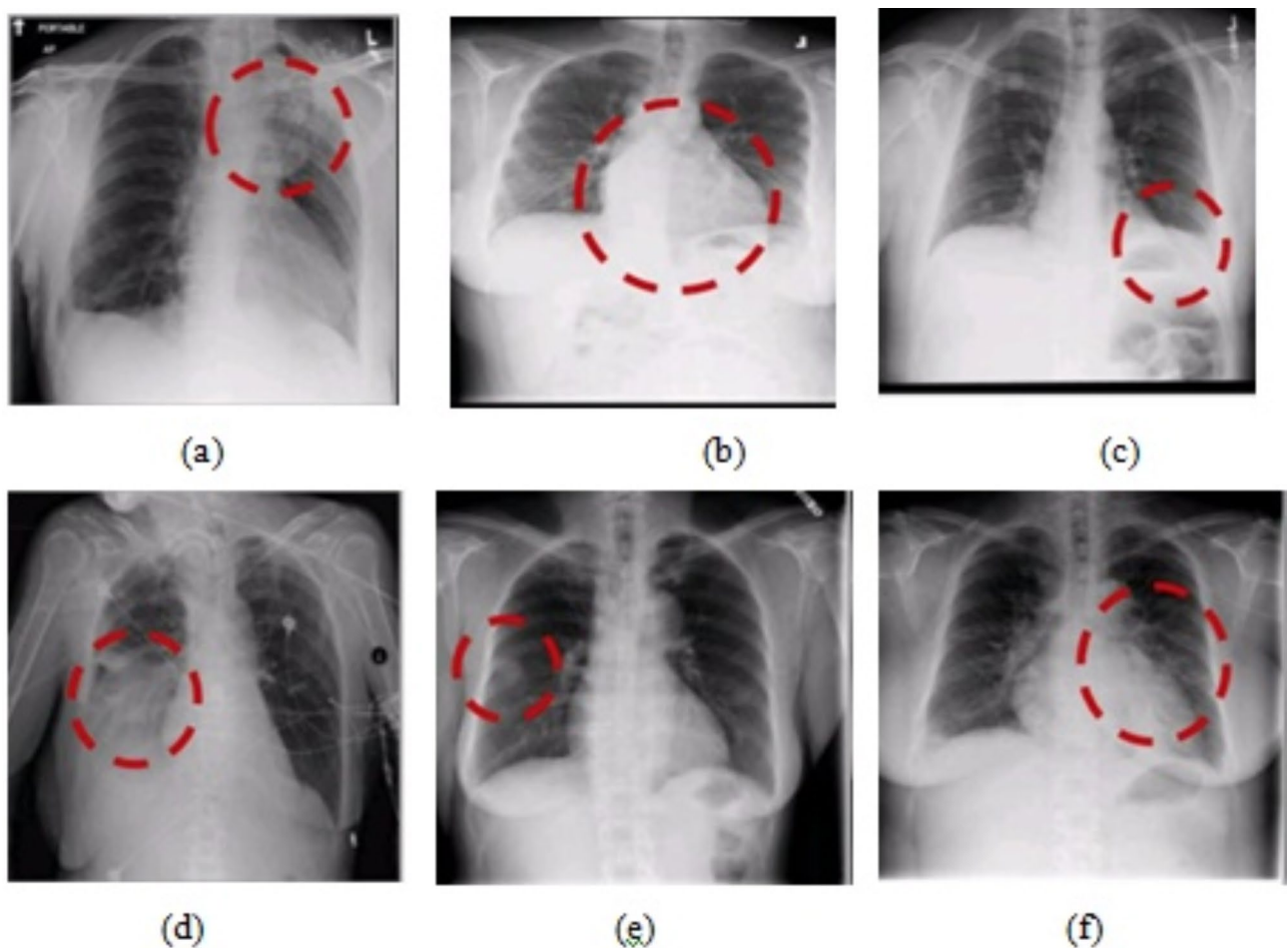
**Effusion:** This is due to the addition of excess fluid among the layers of pleura outside the lungs. Normally, the pleura contains a small amount of blood.

**Infiltration:** This is due to the gradual increase in abnormal substances within cells or body tissues and it accumulates in greater quantity and spreads through the interstices.

**Nodule:** A lung nodule is a small growth on the lung and can be benign or malignant. Benign nodules are usually noncancerous, displaying low aggressiveness and no tendency to spread to other body parts. Conversely, malignant nodules are cancerous and tend to grow rapidly.

**Pneumonia:** When the alveoli fill with fluid, it can lead to an infection in the air sacs, potentially affecting one or both lungs. Figure 1 (a-f) shows the X-ray image of lung infected with different lung diseases<sup>3</sup>.

In this study, Sect. 2 provides a review of relevant literature, while Sect. 3 outlines the materials and methodology used. The results of the study are presented in Sect. 4.



**Fig. 1.** Lung X-ray images diagnosed with (a) atelectasis, (b) cardiomegaly, (c) effusion, (d) infiltration, (e) nodule and (f) pneumonia.

## Literature summary

Geraldo Luis proposed an adaptive active contour model for segmenting lung structures, with discrimination between two types of lung diseases using extreme learning machine neural networks. The experimental results demonstrate an accuracy of 96% in the proposed lung disease classification approach<sup>1</sup>.

Rahib H. Abiyev demonstrated the potential of identifying lung diseases in chest X-rays using both traditional and deep learning techniques. CNNs were employed for detecting and diagnosing various chest diseases, with a comparison made between supervised and unsupervised learning networks<sup>2,7,11–13,29</sup>.

Chunli Qin, Demin Yao have presented several widely used datasets for chest, along with comprehensive image preprocessing techniques including contrast enhancement, segmentation and bone suppression methods commonly applied in this domain<sup>3</sup>. Similarly, Sri Widodo, Ratnasari NurRohmah proposed active shape model morphology and Support Vector Machine volume rendering methods for the detection of lung disease<sup>4,19</sup>.

Subrato Bharati and Prajoy Podder have employed chest X-ray image datasets to identify lung disease properly using a deep learning model. Their study centered on evaluating the efficacy of various neural network architectures including Vanilla neural networks, CNNs, a fusion of CNN and Visual Geometry Group (VGG) based neural networks, Spatial Transformer Networks (STNs) and Capsule networks<sup>5,14,15</sup>. Caixia Liu and Ruibin Zhao presented random forest and multi-scale edge detection algorithms for lung segmentation<sup>6,7</sup>.

An approach to identify pneumonia in chest X-rays based on image processing techniques has been suggested by W. Khan et al. In this study, chest X-ray images were re-dimensioned, and histogram equalization was applied. Various thresholding techniques were used to detect cloudy regions in the lungs<sup>8–11</sup>.

Erdem Aydin have proposed deep online sequential extreme learning machine for the detection of pneumonia<sup>15</sup>. S. Chavda and M. Goyani have attempted to use SVM and LBP features to detect influential image regions and fold data in order to reduce the influence of insignificant regions. Thus, narrow picture areas were used to compute LBP features and to define SVM<sup>18</sup>.

Recently, deep learning algorithms have gained recognition for their effectiveness in detecting lung cancer on chest radiographs, demonstrating strong diagnostic performance, especially in disease-rich environments. Reports indicate that a deep learning algorithm attained specificity ranging from 93 to 100%, sensitivity from 71 to 91% and an area under the receiver operating characteristic curve (AUC) between 0.92 and 0.99 in validation datasets with lung cancer prevalence approximately between 60% and 68%. Zeba Ghaffar evaluated CNNs mobilenet, efficientnet and inception v3 models for detecting covid-19 infection automatically and obtained 95% accuracy<sup>33</sup>. Pir Masoom proposed CNN model to extract features and a Gated Recurrent Unit (GRU) to identify viral diseases from chest X-rays. The study involved 424 X-ray images categorized into three classes: COVID-19, Pneumonia, and Normal<sup>34</sup>.

## Materials and methods

National Institutes of Health (NIH) Chest X-ray dataset is used in this work, serves as a substantial resource primarily for training various image processing systems. It comprises 1,12,120 frontal-view X-ray images annotated with labels for fourteen distinct diseases from 32,717 unique patients. This dataset is extracted from the clinical Picture Archiving and Communication Systems (PACS) at the NIH Clinical Center. The dataset utilized in this study comprises images that are exclusively associated with a single label and uses 8,000 images from NIH database which includes 3,000 benign images and 5,000 malignant images of four classes namely Atelectasis (1328), Infiltration (1410), Nodule (852) and Pneumonia (1410). 80% and 20% of data are used for training and testing respectively.

This proposed work uses two levels of classifiers; one is a binary classifier for classifying the image into benign and malignant and the second level is a multiclass classifier for classifying the malignant image into four different lung diseases. Figure 2 represents the process flow for lung disease classifier. Further, preprocessing techniques such as resizing and gray scaling are employed before feature extraction. Additionally, 20 Gy Level Co-occurrence Matrix (GLCM) features and a total of 59 features related to LBP are individually extracted from each image in the first level classifier, after preprocessing the image. These features are trained using SVM, NB and KNN classifiers. Since it is a binary classifier, it classifies the images as benign or malignant. If the output is classified as malignant, it is then passed to the second-level classifier. In this step, features such as GLCM (Gray-Level Co-occurrence Matrix) and LBP (Local Binary Patterns) are extracted and trained both separately and as a combined feature set using machine learning classifiers. Further, DL model InceptionV3 is trained with malignant images for 4 level disease classification.

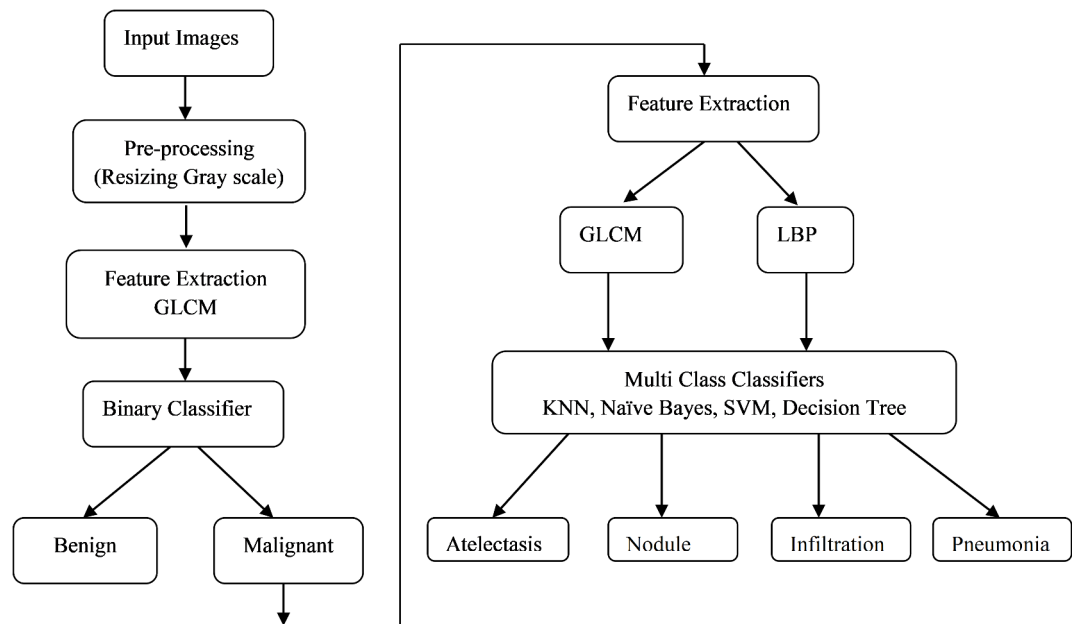
## Feature extraction

The input images are pre-processed by resizing them to uniform dimensions and converting them to grayscale. Feature extraction is a dimensionality reduction technique that creates a compact feature vector, highlighting the most relevant parts of the images. This approach is particularly useful when large image dimensions need to be reduced for tasks such as fast image matching and retrieval.

### *Gy level concurrence matrix (GLCM)*

GLCM provide details related to pixel locations having identical gray levels. The texture characteristics are determined from the statistical distribution of the intensity combinations observed at specified places relative to one another. The GLCM is an  $N_g$ -dimensional square matrix, where  $N_g$  equates to the amount of gray in the frame. Each matrix element in the co-occurrence matrix represents the number of pixel pairs with specific values,  $i$  and  $j$ . A co-occurrence matrix is a 2D array where both rows and columns correspond to potential image values. From each image, 20 GLCM features have been extracted.

The following are the features extracted from the GLCM.



**Fig. 2.** Proposed lung disease binary classifier and multi classifier using machine learning models.

(1) Auto-Correlation: Autocorrelation quantifies the extent to which the values of the same variable are correlated across consecutive time intervals. It evaluates the connection between a variable's present value and its previous values within a time series. It can be formulated as seen in Eq. (1).

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (P_x - \mu_x)(P_y - \mu_y) / \sigma_x \sigma_y \quad (1)$$

Where  $P_x$  and  $P_y$  refers the values of a time series, a data vector,  $\mu_x$  and  $\mu_y$  refers mean for the datasets and  $\sigma_x$  and  $\sigma_y$  represent the standard deviations of the variables  $x$  and  $y$ .

(2) Contrast: It refers to the distinction between elements, emphasizing differences when compared to others. In visual arts, it is the difference in color or hue that makes objects stand out from each other. This difference can create depth, focus and interest in a composition. By highlighting these variations, contrast enhances the visual impact and clarity of an object or image and is given in Eq. (2).

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j)(i - j)^2 \quad (2)$$

(3) Correlation: It quantifies the strength and direction of the association between two variables. A positive correlation signifies that as one variable rises, the other tends to rise as well, whereas a negative correlation indicates that as one variable increases, the other typically decreases. In the next pixels, the gray values are determined by the similarity value uniformly. It can be formulated as shown in Eq. (3).

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j) \times (i \times j) - \mu_x \times \mu_y / \sigma_x \sigma_y \quad (3)$$

(4) Cluster Prominence : It refers to the degree of asymmetry exhibited by clusters within an image. Higher prominence indicates greater asymmetry, resulting in reduced image symmetry. Conversely, lower prominence corresponds to more symmetrical clusters with the GLCM matrix showing peak values around the median values. It can be expressed as in Eq. (4)

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j)(i + j - \mu_x - \mu_y)^4 \quad (4)$$

(5) Cluster shade: It is a metric used to assess the skewness of a matrix, reflecting how data points are distributed around the mean. It indicates asymmetry in an image with higher values suggesting greater imbalance. When cluster shade is high, the image may appear uneven or lopsided. This measure helps to identify texture

variations which contributes in image analysis and pattern recognition. It can be formulated as provided in Eq. (5).

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j) (i + j - \mu_x - \mu_y)^3 \quad (5)$$

(6) Dissimilarity: The comparison and distinction between the two data sets are almost identical, though theoretical frameworks differ. Both sets exhibit similar data patterns, but their value sets diverge. When calculating weights through comparison, they increase exponentially, whereas weights determined by variance exhibit a different expansion rate. It can be expressed as given in Eq. (6).

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} |i - j| P(i, j) \quad (6)$$

(7) Energy: The energy of the Gray-Level Co-occurrence Matrix (GLCM) quantifies texture uniformity by summing the squares of the matrix's elements. Its value ranges from 0 to 1, with 0 indicating high variability and 1 reflecting perfect uniformity. For a constant image, where pixel intensities are uniform, the GLCM energy is maximized at 1, signifying no variation. It can be formulated as shown in Eq. (7).

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j)^2 \quad (7)$$

(8) Entropy: It measures the amount of unpredictability or randomness in an image's texture by quantifying the pixel value distribution. A higher entropy value indicates greater complexity and more texture detail reflecting more variation in pixel values. Conversely, lower entropy suggests more uniform and predictable textures. This numerical indicator helps in analyzing and distinguishing various texture patterns within an image. It can be expressed as in Eq. (8).

$$= - \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j) \log(P(i, j)) \quad (8)$$

(9) Homogeneity: Homogeneity in GLCM assesses the distribution of pixel values by focusing on how closely elements cluster along the matrix's diagonal. This measure evaluates the uniformity or rigidity of pixel distribution, with higher homogeneity indicating a more consistent and less varied texture. Essentially, it quantifies how similar neighboring pixel values are, reflecting the degree of texture smoothness. It can be expressed as shown in Eq. (9).

$$= - \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{P(i, j)}{1 + |i - j|} \quad (9)$$

(10) Maximum Probability: Maximum Probability measures the most frequent occurrence of a particular pixel value pair. It identifies the highest value in the probability distribution matrix, reflecting the most likely pair of gray levels occurring in the image. A higher maximum probability suggests more dominant and repetitive texture patterns. This metric provides insight into the texture's regularity and predictability. The concept can be mathematically expressed as follows (3.10).

$$= \max(i, j) P(i, j) \quad (10)$$

(11) Sum of Squares (Variance): The variance varies when it places comparatively high weights than the average values. It is the heterogeneity indicator. As the values of the gray scale fluctuate, the variance increases. Mathematically, it can be expressed as shown in Eq. (11).

$$= \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j) (i - \mu)^2 \quad (11)$$

(12) Sum Average: Sum Average in the GLCM quantifies the average of the sum of pixel value pairs' intensities. It provides insight into the overall level of gray scale intensity present in the image. It can be formulated as shown in Eq. (12).

$$= \sum_{i=0}^{2G-2} iP_{x+y}(i) \quad (12)$$

(13) **Sum Variance:** Sum Variance measures the dispersion or variability of the sum of pixel value pairs' intensities. It calculates the summed intensities which deviate from their average, reflecting the texture's complexity and variability. Higher Sum Variance indicates greater texture irregularity and more pronounced differences in intensity sums, while lower values suggest a more uniform texture. It can be formulated as shown in Eq. (13).

$$= \sum_{i=0}^{2G-2} (i - sa)^2 P_{x+y}(i) \quad (13)$$

(14) **Sum Entropy:** Sum Entropy measures the unpredictability or randomness of the sum of pixel value pairs' intensities. It quantifies the level of disorder or complexity in the summed intensity distribution. Higher Sum Entropy values indicate more complex and less predictable textures, while lower values suggest more uniform and predictable patterns in the image. This metric helps in understanding the texture's overall randomness and information content. It can be expressed as shown in Eq. (14).

$$= - \sum_{i=0}^{2G-2} P_{x+y}(i) \log(P_{x+y}(i)) \quad (14)$$

(15) **Difference Variance:** Difference measures the dispersion of pixel intensity differences. It calculates the difference between pixel values that deviates from its mean, reflecting the variability in texture patterns. Higher Difference Variance indicates greater texture complexity and more pronounced differences in intensity, while lower values suggest a more uniform texture with less variation. This calculation is based on the default function in the MATLAB.

(16) **Difference Entropy:** Difference Entropy quantifies the randomness or unpredictability of pixel intensity differences. It measures the degree of disorder in the distribution of these differences, with higher values indicating more complex and varied textures. Lower Difference Entropy suggests a more predictable and uniform texture. This metric helps assess the level of texture irregularity and information content related to pixel intensity difference. It can be expressed as shown in Eq. (15).

$$= \sum_{i=0}^{G-1} P_{x+y}(i) \log(P_{x+y}(i)) \quad (15)$$

(17) **Information Measure of Correlation 1:** It evaluates the correlation between the pixel values and their spatial positions. It measures the intensity values of pixels that are related to their positions in the matrix, reflecting the degree of linear dependency between them. Higher IMC1 values indicate stronger linear relationships and more regular patterns, while lower values suggest less correlation and more variability in the texture. It can be formulated as shown in Eq. (16).

$$IMC1(X; Y) = \frac{I(X : Y)}{\min \{H(X), H(Y)\}} \quad (16)$$

Where  $I(X : Y)$  is the mutual information between the random variables  $X$  and  $Y$  and  $H(X)$  and  $H(Y)$  are the entropies respectively.

(18) **Information Measure of Correlation 2:** It assesses the correlation between the joint distribution of pixel values and their marginal distributions. It quantifies the degree of dependency between pixel values and their spatial positions and it is given in Eq. (17). Higher IMC2 values indicate stronger associations and more structured textures, while lower values reflect weaker relationships and greater randomness in the texture patterns.

$$IMC2(X; Y) = \frac{I(X : Y)}{\sqrt{H(X) H(Y)}} \quad (17)$$

(19) **Inverse Difference Normalized (IDN):** The Inverse Difference Normalized (IDN) is a metric used to assess texture features in image analysis. It is calculated by normalizing the inverse difference of pixel intensities across an image. The IDN is given by the Eq. (18).

$$IDN = \frac{\sum_{i,j} \frac{1}{1+|i-j|} P(i, j)}{\sum_{i,j} P(i, j)} \quad (18)$$

Where  $P(i, j)$  represents the joint probability distribution of pixel intensities  $i$  and  $j$  and  $|i - j|$  is the absolute difference between these intensities.

(20) **Inverse Difference Moment Normalized:** The Inverse Difference Moment Normalized (IDMN) is a texture feature used in image analysis to quantify the uniformity or smoothness of a texture. It is derived from the co-occurrence matrix of pixel intensities and is defined as shown in Eq. (19).



$$IDMN = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{P(i, j)}{1 + |i - j|^2} \quad (19)$$

#### Local binary pattern

The local binary patterns are highly efficient texturation operators that mark the pixels of the image by thresholding the neighborhoods of each pixel and display the effects of the screening as a binary number. The feature separates the image into cells that do not overlap.

The LBP function vector is produced as follows:

- Divide the screen into cells as illustrated in Fig. 3.
- Compare the pixels to one of their 8 neighbors as shown in Fig. 3 for each pixel inside a cell. Follow the pixels in a circle, i.e., in the right or in the opposite direction.
- Write down “0” when the value of the center pixel is greater than that of the neighbor. If not, type “1”. There’s a binary number of 8 digits.
- Calculate the histogram and it can be seen as a vector of 256 dimensions.
- Use the histogram to normalize.

#### Machine learning classifiers

To make predictions and decisions without directly programming, machine learning algorithms construct a sample data based mathematical model<sup>17</sup>. Its key purpose is to classify the type or class that may contain a new data. An algorithm is used for each classifier to map the input data to a given group.

#### KNN classifier

The K-Nearest Neighboring (KNN) is a type of ML algorithm used for predictive problems in both classification and regression.

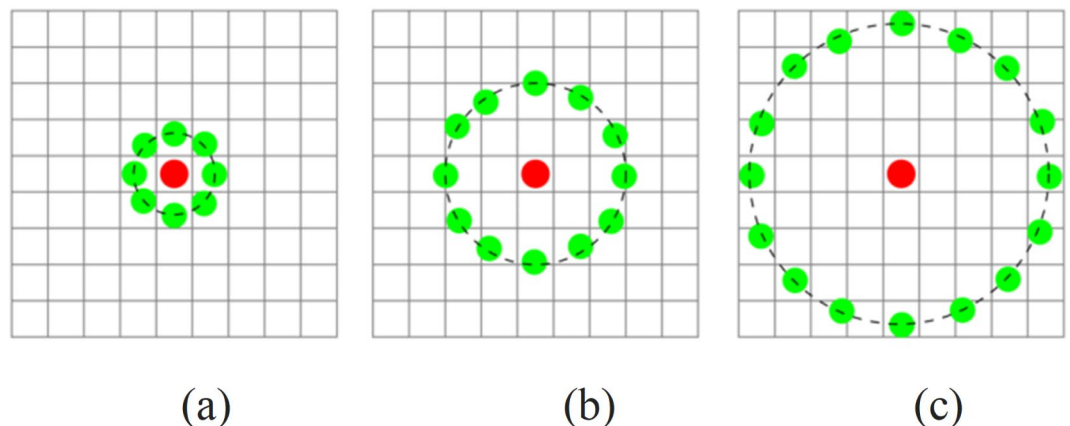
The KNN algorithm involves the following steps.

- Step 1 – Divide the dataset into training and test data.
- Step 2 – K must be picked, as the closest datasets. K may be any whole value.
- Step 3 – The following steps must be utilized at each point in the test results.
- Calculate Distances: Compute the Euclidean distance between the test data and each row of training data.
- Sort Distances: Arrange the distances in ascending order according to their values.
- Select Top K: Select the top K rows from the array after sorting.
- Determine Class: Assign the class of the test point based on the most common class among the selected K rows.
- Step 4 – End.

The standardized Euclidean metric shown in Eq. (19) is employed to calculate the distance between A and B.

$$dist(A, B) = \sqrt{\frac{\sum_{i=1}^m (x_i - y_i)^2}{m}} \quad (20)$$

In k-NN, the hyperparameter k is often initially set to  $\sqrt{n}$ , where n is the number of data points in the training set. The selection of k is crucial for balancing bias and variance in the model. A small k can lead to high variance,



**Fig. 3.** LBP with neighbors obtained in different radius (R) and the number of neighbors (P): (a)  $R=1$  &  $P=8$ , (b)  $R=2.5$  &  $P=12$ , (c)  $R=4$  &  $P=16$ <sup>32</sup>.

causing the model to fit noise in the training data (overfitting) and conversely, a large  $k$  can result in high bias, as the model averages over more points and may miss important patterns (underfitting). In this study,  $k$  was initially tested over the range from 1 to 10. Although there are no pre-defined statistical methods to determine the most favorable  $k$ , plotting error rates against different  $k$  values allowed us to select  $k=5$ , which minimized the error.

#### *SVM classifier*

SVM algorithm is commonly used for both classification and regression tasks. However, SVM is primarily utilized for classification problems. The primary objective of SVM is to separate datasets into classes through the identification of a marginal hyperplane. The surface area for decision making is the hyperplane of the form shown in Eq. (20).

$$W^T X + b = 0 \quad (21)$$

where  $W$  refers to the weight vector,  $X$  represents an input vector and  $b$  is the bias.

The SVM classifier with a Gaussian kernel can be expressed as a weighted linear combination of the kernel functions between a given data point and the support vectors. The Gaussian Kernel may be expressed as shown in Eq. (21).

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (22)$$

The Radial Basis Function shown in Eq. (22) is a commonly used kernel.

$$K(x, x') = \frac{e^{(-x-x'^2)}}{2\sigma^2} \quad (23)$$

where  $\|x - x'\|^2$  stands for the squared Euclidean distance between the two data points  $x$  and  $x'$ .

In SVM, the hyperparameter  $C$  determines the strength of regularization, balancing the trade-off between creating a smooth decision boundary and accurately classifying the training data. A smaller  $C$  emphasizes a smoother decision boundary, potentially allowing for more misclassifications, while a larger  $C$  focuses on classifying training points correctly, possibly at the cost of overfitting. Conversely, the parameter  $\Gamma$  in SVM affects the curvature of the decision boundary, determining its flexibility or rigidity. A lower  $\Gamma$  value results in a smoother, less intricate boundary, while a higher  $\Gamma$  leads to a more complex boundary with greater detail. This study uses the value of  $C$  &  $\Gamma$  as 1 as per<sup>31</sup>.

#### *Decision tree classifier*

Decision tree algorithms can be used for both regression and classification tasks. A decision tree has root nodes and terminal nodes. It can be generated using the following steps after creating the root node.

Step 1: Creation of terminal nodes: One important item is to choose when to avoid generating terminal nodes or when to create new terminal nodes.

Step 2: Recursive division is a tree-building process. In this process, after a node has been created, the child nodes on each data group are generated by splitting the dataset. It can be created repetitively by calling the same function.

Step 3: Estimation and assumption is important to make a forecast of this after a decision tree has been developed. Prediction is the navigation of the decision tree with the specific data row.

Step 4: Pruning is a process of machine learning and searching algorithms that reduce the size of decision making bodies by eliminating the parts of the tree that have low classification capacity. Taping decreases the finishing classification complexity and thereby increases the predictive performance by eliminating overfitting.

Decision trees are built with varying maximum depths ranging from 1 to 8 as part of hyperparameter tuning. The accuracy remained around 85% for depths of 7 and higher, so a depth of 7 is selected to prevent overfitting.

#### *Naïve Bayes classifier*

Naïve Bayes algorithms are classification technologies based on the theorem of Bayes's implementation. All predictors are firmly expected to be autonomous. It can forecast the likelihoods of class inclusion like the likelihood that a certain number is in a given class.

There are two types of probabilities.

- Posterior Probability  $P\left(\frac{H}{X}\right)$
- Prior Probability  $P(H)$

where  $X$  means data tuple and  $H$  stands for some hypotheses. The Bayes Theorem is given in Eq. (23).

$$P\left(\frac{H}{X}\right) = \frac{P\left(\frac{X}{H}\right) P(H)}{P(X)} \quad (24)$$

The hyperparameter  $\alpha$  in a Naive Bayes classifier plays a critical role in achieving the right balance between bias and variance. A small  $\alpha$  value can lead to overfitting by making the model too sensitive to the training data,

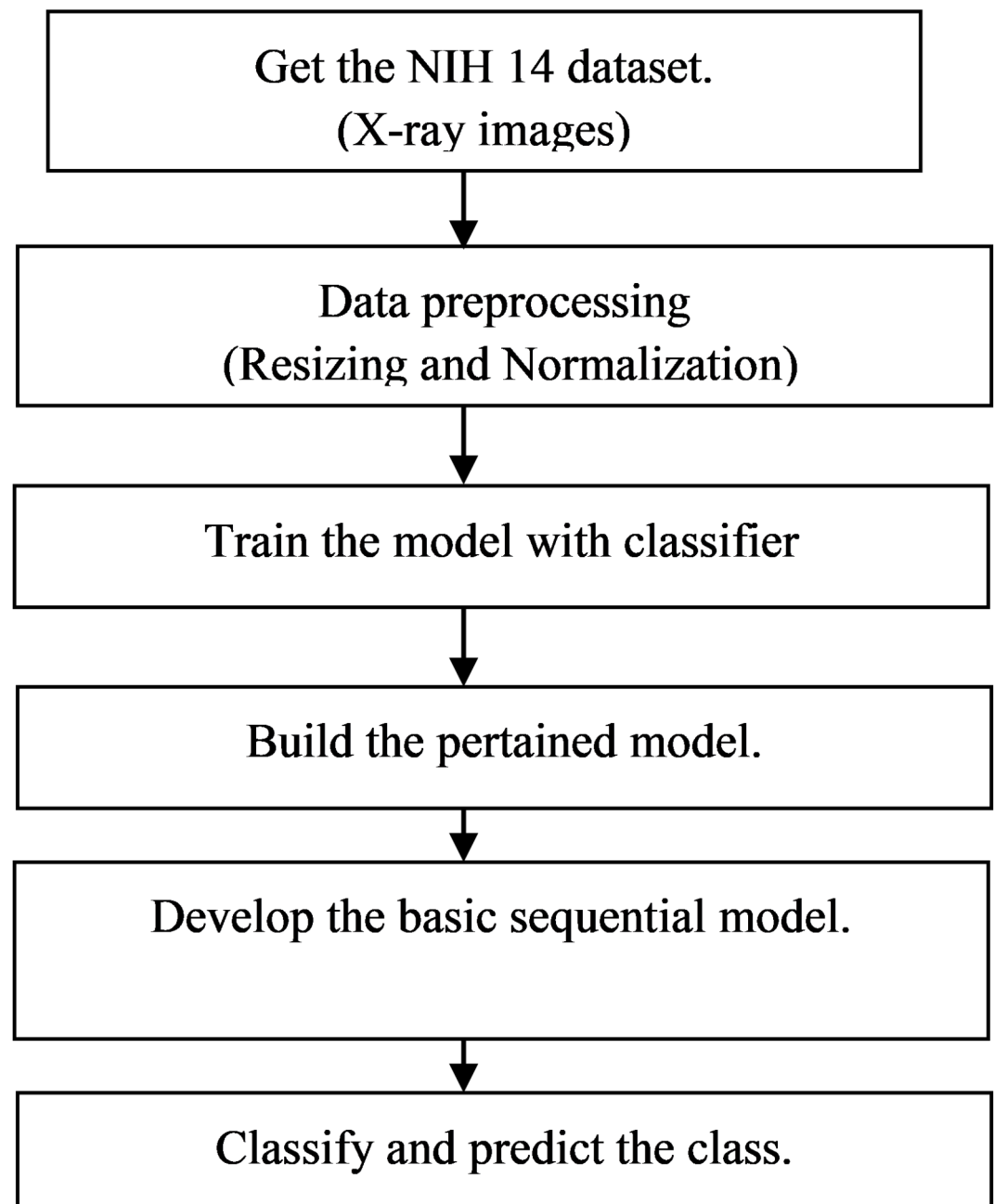


while a large  $\alpha$  can result in underfitting, causing the model to oversimplify the data patterns. In this study, we selected an  $\alpha$  value of 0.01 for the Multinomial Naive Bayes classifier, which is lower than the typical default of 1 to improve model performance.

#### Deep learning model

The process classifying four lung diseases using deep learning model is given in Fig. 4. Deep Neural Network (DNN) is a class of algorithms that employ several layers to steadily retrieve higher input levels. The right mathematical manipulation is detected in order to translate the input into the output, be it linear or non-linear.

It takes input as a 2-dimensional sequence and operates on images directly instead of relying on function extraction as the key distinction among some other ordinary neural networks. In the input data, the CNN uses spatial correlations. A neural network binds those input neurons with each concurrent layer. This particular environment is referred to as the receptive area. The secret neurons are the local receptive field and they process the data in the above region, so that adjustments do not arise beyond the particular frontier.



**Fig. 4.** General flowchart of DNN classifier for lung disease classification.

### Inception V3

The Keraslibrary consist of 10 pre-trained models of application. InceptionV3 has 48 layers of depth with an image dimension of  $299 \times 299 \times 3$ . The pre-trained model can classify the images upto 1,000 categories. Figure 5 gives the architecture of inception v3 model. Batch normalization stabilizes the study phase and significantly decreases the number of training cycles. This helps each layer to learn about consistent input propagation, which can thus accelerate network preparation. The fundamental reason for using this normalization is to reduce the covariable change by normalizing the activations for each layer.

Within the various layers of a neural network, the initial layer is typically the convolutional layer, designed to extract features from input images. By processing input data through small squares, this layer effectively preserves the spatial relationships between pixels. During this process, a filter traverses the image, examining a subset of pixels at each step and generating a feature map based on these localized patterns<sup>30</sup>. The next layer is pooling layer which reduces the size of the filter layer, thereby allowing the model to train faster. It also prevents overfittingby eliminating unwanted values in the filter tensor. There are several pooling types such as Max, Min and Avg pooling. One commonly used pooling technique is Max Pooling, which essentially disregards the exact position of a feature within the input volume. Instead, it focuses on the presence of features, rather than their precise locations in relation toothers. As a result, the spatial dimensions of the input volume are significantly reduced by this layer, while the depth remains unchanged.

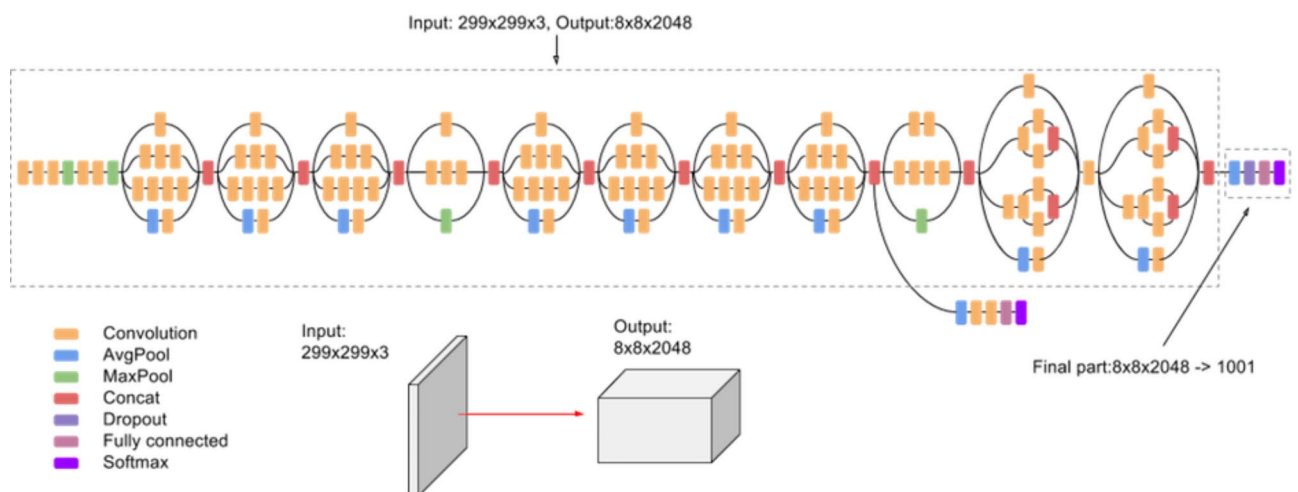
Furthermore, the activation function determines whether or not the weighted total should be triggered and adds extra bias to the neuron. The activation mechanism involves the application of non-linearity to neuron production. Among the array of activation functions, including linear, sigmoid, tanh, ReLU and softmax, the pre-trained InceptionV3 model utilizes ReLU and softmax functions. ReLU is typically employed in intermediate layers, while softmax is often used in the final layer for multi-class classification tasks. Additionally, flattening is a process that transforms the data between the convolutional layer and the fully connected layer into a one-dimensional array, facilitating its input into subsequent layers. In V3, at the beginning, a tensor (2048.1) appears from the final convolutionary layer. If this outcome is flattened, the next layer will have an input (2048 ). 2048 inputs originates from  $2048 \times 1 \times 1$  multiplication which is the size of the previous tensor production. Flattening helps to examine each pixelif it has a neuron in the picture. Moreover, loss functions are used to measure the quantity that a model needs to minimize during the training process. An error must be calculated repeatedly for the existing model status. This includes a function called the failure function, which is traditionally used to measure the weight loss of the model, in order to upgrade the weights to minimize losses during the next assessment. The loss function utilized in this scenario is the sparse categorical cross-entropy loss function. In a dense layer, also known as a fully connected layer, every input and output is connected to each neuron within the layer. This layer would have the class numbers and their images while classification is done and used softmax function as a classifier in these layers.

### Preprocessing

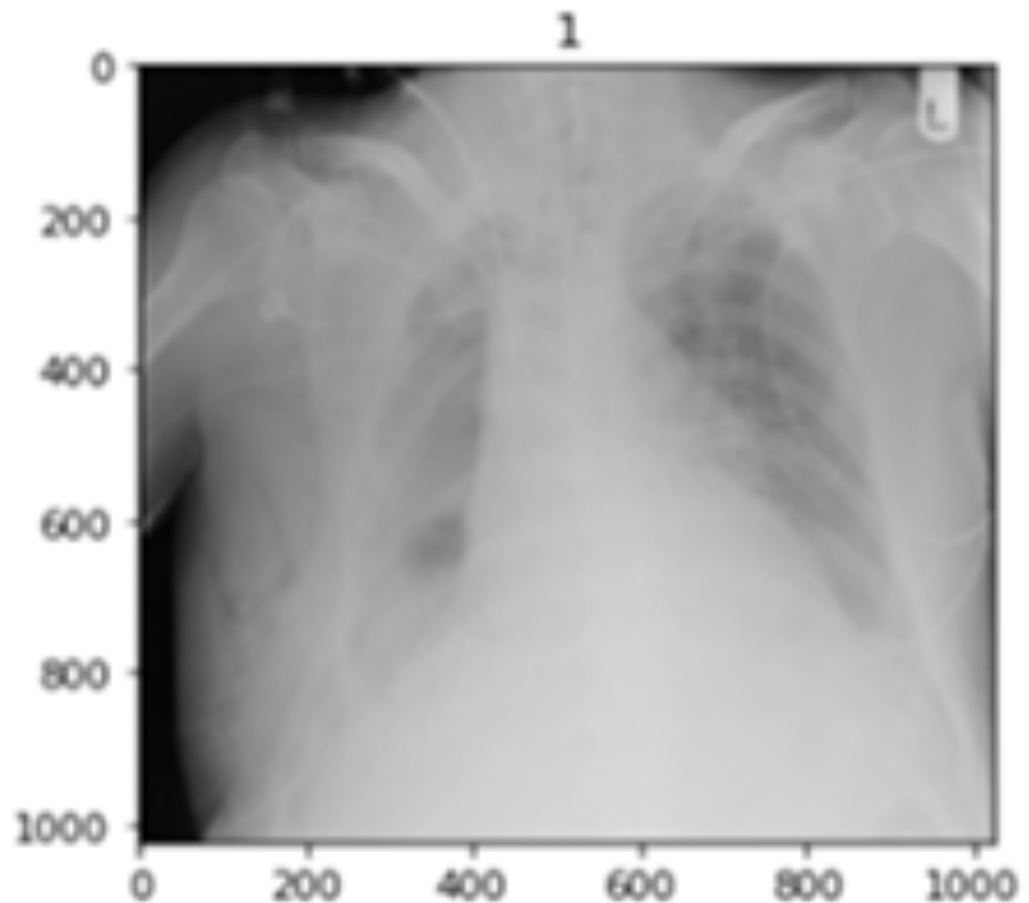
The key purpose of the CNN is to reduce the computational sophistication of the model in most image recognition tasks. To minimize the computations and for the quicker processing, the initial 3 channel images are resized from  $1024 \times 1024 \times 3$  in Fig. 6 to  $229 \times 229 \times 3$  in Fig. 7.

### Build the pre-trained model

Keras contains 10 pre-trained models for image classification. All the pre-trained network models are available in the application module of Keras. These pre-trained models are capable of classifying any image that falls into 4 categories of images.



**Fig. 5.** Architecture of Inception V3 model.



**Fig. 6.** X-ray image of size  $1024 \times 1024$  before preprocessing.

#### *Train the model*

The proposed system uses a dataset consisting of four types of diseases or classes. The images are trained to evaluate the accuracy and determine whether the pre-trained model provides better classification results. Before training the model, it is essential to incorporate an optimizer, loss function, and evaluation metrics to ensure the model runs with the maximum number of epochs, optimizing its performance in classifying the given images. After training, the model reports parameters such as accuracy, loss, validation accuracy, and validation loss.

#### **Multi attribute decision making method- TOPSIS**

The Technique for Order of Preference by Similarity to Ideal Solution (*TOPSIS*) is a multi-criteria decision analysis method. Based on the attributes, this method ranks the alternative. Based on the ranking, the best alternative may be chosen out of all the alternatives.

Given a set of alternatives,  $A = \{A_k | k = 1, \dots, n\}$ , and a set of criteria  $= \{C_j | j = 1, \dots, m\}$ , where  $X = \{x_{kj} | k = 1, \dots, n; j = 1, \dots, m\}$  denotes the set of performance ratings and  $w = \{w_j | j = 1, \dots, m\}$  is the set of weights. The initial step involved in TOPSIS is to calculate normalized ratings by using Eq. (24).

$$r_{kj}(x) = \frac{x_{kj}}{\sqrt{\sum_{k=1}^n x_{kj}^2}}, \quad k = 1, \dots, n; j = 1, \dots, m \quad (25)$$

For benefit criteria.

$r_{kj}(x) = (x_{kj} - x_j^-) / (x_j^* - x_j^-)$ , where  $x_j^* = \max_k x_{kj}$  and  $x_j^- = \min_k x_{kj}$ . The Positive Ideal Point (PIS) and Negative Ideal Point (NIS) are calculated using Equations (3.25) and (3.26).

$$\begin{aligned} PIS &= A^+ = \{v_1^+(x), v_2^+(x), \dots, v_j^+(x), \dots, v_m^+(x)\} \\ &= \left\{ (\max_k v_{kj}(x) | j \in J_1), (\min_k v_{kj}(x) | j \in J_2) | k = 1, \dots, n \right\} \end{aligned} \quad (26)$$



**Fig. 7.** X-ray image of size  $299 \times 299$  after preprocessing for DNN model.

$$\begin{aligned}NIS &= A^- = \{v_1^-(x), v_2^-(x), \dots, v_j^-(x), \dots, v_m^-(x)\} \\ &= \left\{ \left( \max_k v_{kj}(x) \mid j \in J_1 \right), \left( \min_k \mid j \in J_2 \right) \mid k = 1, \dots, n \right\}\end{aligned}\quad (27)$$

where  $v_{kj}(x) = w_j r_{kj}$ ,  $k=1$  to  $n$ ,  $j=1$  to  $m$ .

From the distinction of alternatives between the PIS and NIS., the separation values are calculated using the Euclidean distance which is given in 3.27 and 3.28

$$D_k^+ = \sqrt{\sum_{j=1}^m [v_{kj}(x) - v_j^+(x)]^2}, k = 1, \dots, n \quad (28)$$

$$D_k^- = \sqrt{\sum_{j=1}^m [v_{kj}(x) - v_j^-(x)]^2}, k = 1, \dots, n \quad (29)$$

The similarities to PIS can be.

$$C_k^* = D_k^- / (D_k^+ + D_k^-), k = 1, \dots, n \quad (3.29)$$

Finally, the best alternative is selected based on the  $C_k$  value.

## Results and discussion

### Binary classifier

In the first level classification, KNN, SVM and Naïve Bayes classifiers are proposed and its classification accuracy and confusion matrix are given in Table 1. In the dataset, 80% is used for training and 20% is used for testing.

Classifier	Confusion Matrix	Accuracy	Precision
K-NN Classifier	$\begin{bmatrix} 498 & 144 \\ 113 & 845 \end{bmatrix}$	83.9	81.5
Binary SVM Classifier	$\begin{bmatrix} 586 & 17 \\ 42 & 955 \end{bmatrix}$	96.3	93.31
Naïve Bayes Classifier	$\begin{bmatrix} 562 & 30 \\ 46 & 962 \end{bmatrix}$	95.25	90.93
Decision Tree [35]	-	74	-
Linear Regression [35]	-	75	-

**Table 1.** Classification accuracy for binary class classification.

Classifier	Confusion Matrix	Sensitivity	Specificity	Precision	F1 Score Value	Accuracy	Training loss	Testing Loss
KNN	$\begin{bmatrix} 104 & 76 & 29 & 49 \\ 62 & 169 & 50 & 41 \\ 29 & 48 & 171 & 19 \\ 33 & 40 & 22 & 58 \end{bmatrix}$	0.4869	0.8311	0.4849	0.4853	0.751	0	0.5230
Weighted KNN with k= 5	$\begin{bmatrix} 128 & 77 & 23 & 30 \\ 100 & 151 & 56 & 15 \\ 31 & 50 & 181 & 5 \\ 66 & 46 & 19 & 22 \end{bmatrix}$	0.4467	0.8217	0.4535	0.4413	0.741	0.3575	0.5180
SVM with RBF	$\begin{bmatrix} 109 & 76 & 27 & 46 \\ 63 & 174 & 55 & 30 \\ 21 & 49 & 185 & 12 \\ 55 & 41 & 23 & 34 \end{bmatrix}$	0.4695	0.8302	0.4670	0.4670	0.751	0	0.4980
Decision Tree	$\begin{bmatrix} 106 & 78 & 15 & 59 \\ 85 & 176 & 22 & 39 \\ 17 & 32 & 211 & 7 \\ 54 & 38 & 15 & 46 \end{bmatrix}$	0.5121	0.8441	0.5137	0.5129	0.7695	0.1218	0.4610
Pruned Tree	$\begin{bmatrix} 110 & 79 & 16 & 53 \\ 87 & 170 & 30 & 35 \\ 12 & 28 & 225 & 2 \\ 54 & 37 & 15 & 47 \end{bmatrix}$	0.5260	0.8480	0.5224	0.5237	0.776	0.1570	0.4480
Decision Tree (all predictors)	$\begin{bmatrix} 115 & 84 & 11 & 48 \\ 88 & 171 & 24 & 39 \\ 14 & 29 & 218 & 6 \\ 45 & 44 & 14 & 50 \end{bmatrix}$	0.5300	0.8484	0.5316	0.5307	0.777	0.3023	0.4460
Naïve Bayes	$\begin{bmatrix} 95 & 33 & 64 & 66 \\ 70 & 90 & 124 & 38 \\ 11 & 26 & 219 & 11 \\ 43 & 14 & 27 & 69 \end{bmatrix}$	0.4797	0.8240	0.4664	0.4509	0.7365	0.5255	0.5270
Naïve Bayse using Kernal	$\begin{bmatrix} 85 & 34 & 61 & 78 \\ 81 & 88 & 110 & 43 \\ 21 & 14 & 219 & 13 \\ 39 & 10 & 29 & 75 \end{bmatrix}$	0.4783	0.8233	0.4650	0.4450	0.7335	0.5028	0.5330

**Table 2.** Results for the classification of abnormal images using GLCM features.

For the value of k=5, KNN gives the accuracy of 83.9%, Naïve Bayes algorithm provides 95.25% and SVM classifier give the better classification accuracy of 96.3%. These results are compared with decision tree and logical regression model<sup>35</sup> and it is given in Table 1.

### Machine learning method

In the second level of classification, 5000 malignant images such as Atelectasis=1328, Infiltration=1410, Nodule=852 and Pnemonia=1410 are trained using the multi class classifiers KNN, SVM, NB and Decision Tree classifier for both GLCM features, LBP features and its combined set of features<sup>22</sup>. The results of these classifiers are compared and shown in Tables 2, 3 and 4 respectively.

After the first level classification, if the result is malignant, then the image will be applied to the next level classifier to find out the class of disease. The confusion matrix for the SVM classifier with GLCM and LBP feature is shown in Fig. 8.

GLCM is indeed a statistical feature extraction method. It captures and quantifies the statistical properties of pixel relationships in an image and the features derived from the GLCM provide valuable information for analyzing and interpreting textures. Table 2 shows that the classification accuracy is ranging from 73 to 77% and 77% for decision tree classifiers using GLCM features. Specificity is ranging 82 to 84, which shows that the system correctly identifying the negative cases, specifically, it indicates that the model has a low rate of false positives. Sensitivity in the range of 50% in all cases indicates that the model is performing moderately well but

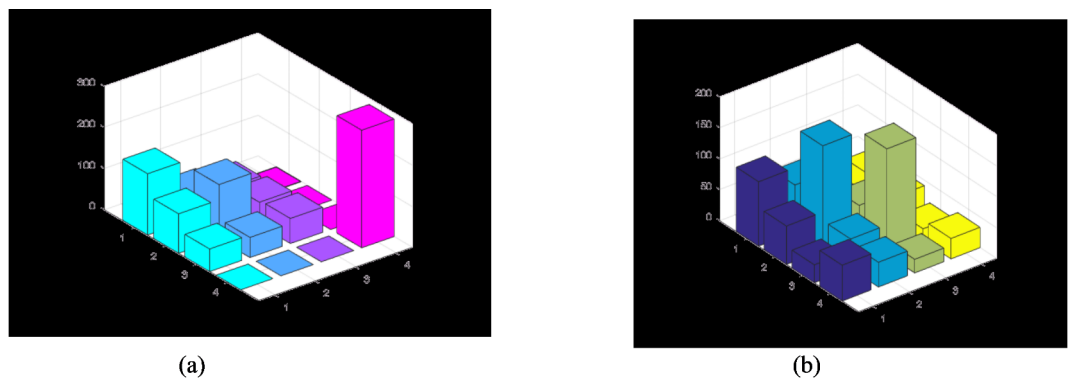
Classifier	Confusion Matrix	Sensitivity	Specificity	Precision	FI Score Value	Accuracy	Training Loss	Testing Loss
KNN	$\begin{bmatrix} 129 & 76 & 57 & 1 \\ 92 & 131 & 62 & 0 \\ 57 & 50 & 58 & 0 \\ 0 & 0 & 0 & 286 \end{bmatrix}$	0.5754	0.8694	0.5745	0.5744	0.8025	0	0.3953
Weighted KNN with k=5	$\begin{bmatrix} 153 & 73 & 36 & 1 \\ 114 & 133 & 38 & 0 \\ 69 & 48 & 48 & 0 \\ 0 & 0 & 0 & 286 \end{bmatrix}$	0.5848	0.8731	0.5922	0.5843	0.8105	0.2568	0.3792
SVM with RBF	$\begin{bmatrix} 149 & 65 & 48 & 1 \\ 96 & 133 & 56 & 0 \\ 55 & 49 & 61 & 0 \\ 0 & 0 & 0 & 286 \end{bmatrix}$	0.6007	0.8774	0.6003	0.5993	0.815	0	0.3702
Decision Tree	$\begin{bmatrix} 136 & 74 & 50 & 3 \\ 105 & 123 & 55 & 2 \\ 41 & 58 & 65 & 1 \\ 2 & 2 & 1 & 281 \end{bmatrix}$	0.5813	0.8691	0.5792	0.5797	0.803	0.0775	0.3943
Pruned Tree	$\begin{bmatrix} 137 & 73 & 50 & 3 \\ 105 & 123 & 55 & 2 \\ 41 & 58 & 65 & 1 \\ 2 & 2 & 1 & 281 \end{bmatrix}$	0.5822	0.8695	0.5801	0.5806	0.8035	0.0805	0.3933
Decision Tree (all predictors)	$\begin{bmatrix} 143 & 81 & 37 & 2 \\ 99 & 140 & 42 & 4 \\ 35 & 66 & 63 & 1 \\ 3 & 1 & 0 & 282 \end{bmatrix}$	0.6007	0.8757	0.6041	0.6017	0.8145	0.2233	0.3712
Naïve Bayes	$\begin{bmatrix} 124 & 65 & 70 & 4 \\ 97 & 115 & 68 & 5 \\ 33 & 28 & 101 & 3 \\ 0 & 10 & 1 & 275 \end{bmatrix}$	0.6122	0.8741	0.5987	0.5989	0.808	0.3970	0.3843
Naïve Bayse using Kernal	$\begin{bmatrix} 118 & 69 & 75 & 1 \\ 83 & 125 & 72 & 5 \\ 27 & 35 & 102 & 1 \\ 0 & 12 & 2 & 272 \end{bmatrix}$	0.6141	0.8751	0.6044	0.6023	0.809	0.3885	0.3823

**Table 3.** Results for the classification of abnormal images using LBP features.

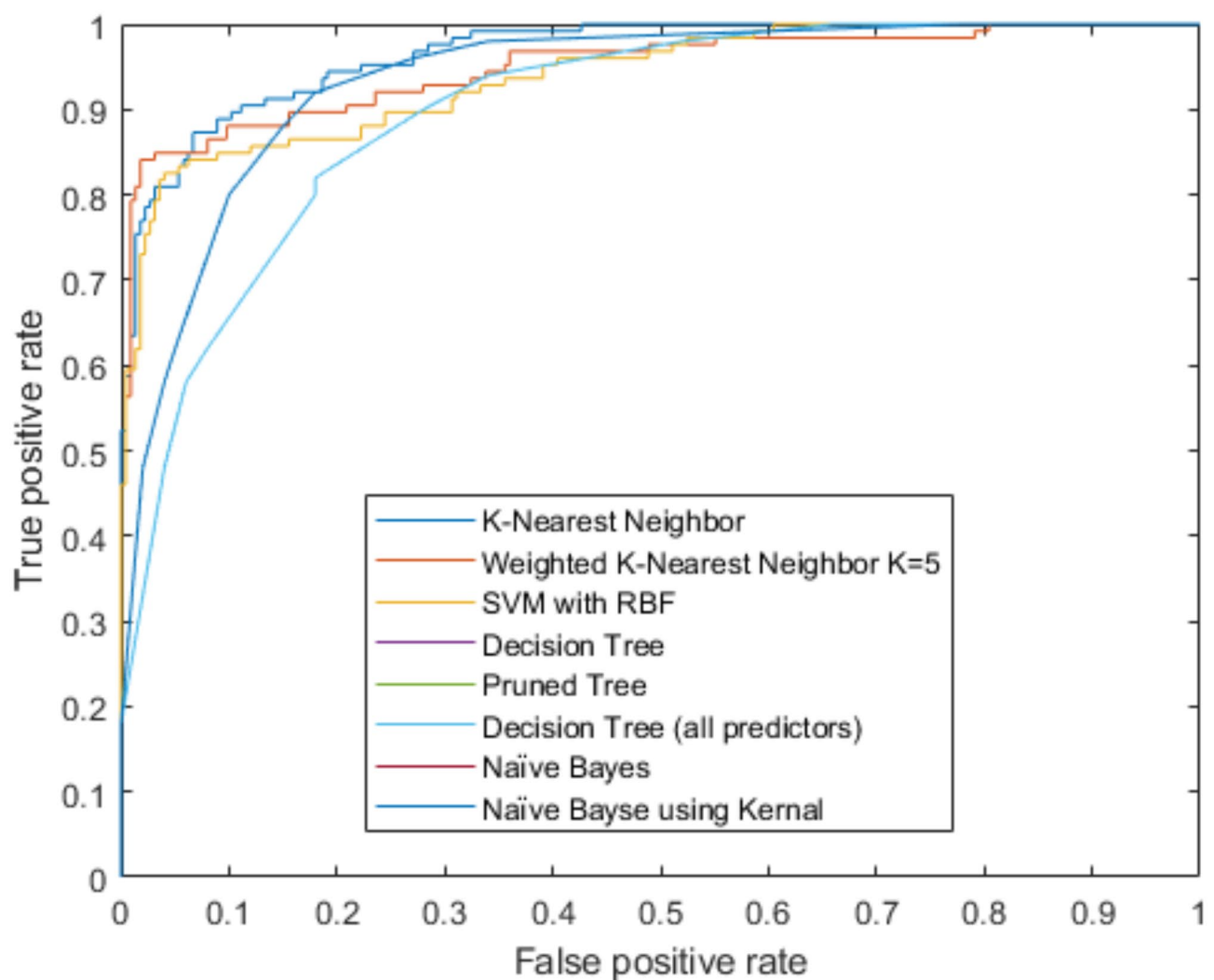
Classifier	Confusion Matrix	Sensitivity	Specificity	Precision	FI Score Value	Accuracy	Training Loss	Testing Loss
KNN	$\begin{bmatrix} 147 & 52 & 57 & 1 \\ 81 & 142 & 62 & 0 \\ 47 & 41 & 73 & 4 \\ 0 & 0 & 0 & 286 \end{bmatrix}$	0.6249	0.8846	0.6217	0.6217	0.8245	0	0.3953
Weighted KNN with k=5	$\begin{bmatrix} 167 & 73 & 36 & 1 \\ 114 & 133 & 38 & 0 \\ 69 & 48 & 48 & 0 \\ 0 & 0 & 0 & 286 \end{bmatrix}$	0.6318	0.8865	0.6421	0.6328	0.8305	0.2568	0.3792
SVM with RBF	$\begin{bmatrix} 149 & 65 & 36 & 1 \\ 8 & 159 & 43 & 2 \\ 36 & 45 & 81 & 3 \\ 0 & 0 & 0 & 286 \end{bmatrix}$	0.6814	0.9020	0.6800	0.6796	0.8525	0	0.3702
Decision Tree	$\begin{bmatrix} 165 & 45 & 50 & 3 \\ 92 & 134 & 57 & 2 \\ 34 & 54 & 78 & 2 \\ 2 & 2 & 1 & 281 \end{bmatrix}$	0.6337	0.8865	0.6297	0.6298	0.8280	0.0775	0.3943
Pruned Tree	$\begin{bmatrix} 156 & 64 & 42 & 1 \\ 91 & 136 & 56 & 2 \\ 3 & 54 & 72 & 2 \\ 1 & 2 & 1 & 282 \end{bmatrix}$	0.6232	0.8829	0.6206	0.6212	0.8235	0.0805	0.3933
Decision Tree (all predictors)	$\begin{bmatrix} 161 & 61 & 37 & 4 \\ 71 & 169 & 41 & 4 \\ 32 & 58 & 75 & 0 \\ 2 & 0 & 0 & 284 \end{bmatrix}$	0.6632	0.8966	0.6637	0.6632	0.8450	0.2233	0.3712
Naïve Bayes	$\begin{bmatrix} 159 & 45 & 55 & 4 \\ 85 & 131 & 64 & 5 \\ 27 & 24 & 112 & 2 \\ 0 & 4 & 2 & 280 \end{bmatrix}$	0.6805	0.8964	0.6679	0.6662	0.8415	0.3970	0.3843
Naïve Bayse using Kernal	$\begin{bmatrix} 161 & 47 & 54 & 1 \\ 69 & 148 & 64 & 4 \\ 22 & 31 & 111 & 1 \\ 2 & 4 & 0 & 280 \end{bmatrix}$	0.6958	0.9024	0.6853	0.6850	0.8505	0.3885	0.3823

**Table 4.** Results for the classification of abnormal images using GLCM and LBP features.





**Fig. 8.** Representation of Classification matrix of four lung disease classes for SVM classifier using (a) GLCM features and (b) LBP features.



**Fig. 9.** Receiver Operating Characteristic (ROC) of different Machine learning models with GLCM features.

is missing a significant portion of positive instances. F1 score is a single metric that balances both aspects of performance, which is in the range of 50% indicates a model that has a balanced but moderate performance in terms of both precision and recall.

Figure 9 illustrates the Receiver Operating Characteristic (ROC) curve also known as the performance curve for the classifier output based on the GLCM features.

From Table 3, it is clear that the SVM and decision tree classifier with all predictors using LBP features gives 81.5% classification accuracy than others with small variations. This also gives higher range of specificity values which indicates that the classification for negative case is greater than the positive classes. Since LBP effectively detects fine-grained textural details due to its local comparison approach, this ML classifiers with LBP features gives better classification accuracy of 4% approximately more than with GLCM features.

Figure 10 displays the Receiver Operating Characteristic (ROC) curve for the classifier output based on the LBP features.

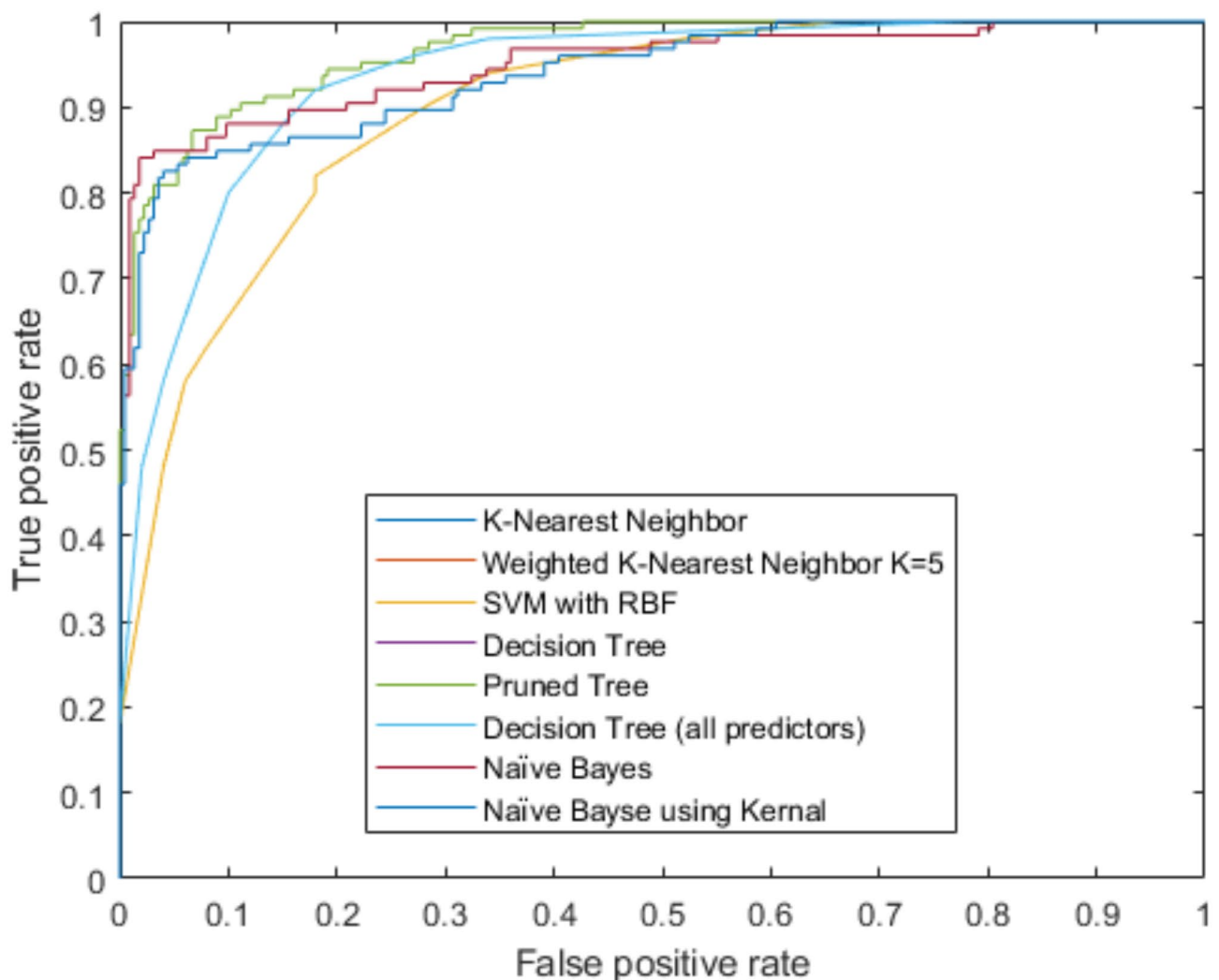
ML classifiers are also tested with both GLCM and LBP features and the results are shown in Table 4. The classification results are increased approximately 4–5% than the results with LBP features and around 8% more than GLCM features. By combining GLCM and LBP features, classification accuracy improves through the integration of global texture characteristics like contrast and correlation with detailed local texture patterns. This approach results in a comprehensive feature set that captures both broad structural information and fine textures, enhancing the model's ability to differentiate between classes. SVM classifier with RBF kernel provides 85% accuracy with the dataset size of 5000 images.

In this machine learning algorithm, the accuracy level may be improved by increasing the number of dataset. Increased number of datasets would increase the complexity of the network as well as computation time. Rather than expanding the dataset, this study proposes a deep learning model for improving performance of multiclass classification.

In both feature sets, it's observed that the SVM classifier with radial basis function kernel achieves superior classification accuracy compared to other classifiers utilized.

### TOPSIS decision making

Since a number of attributes are involved in predicting the classification accuracy, ranking is provided to the classifiers used in this machine learning approach. The multi attribute decision making method TOPSIS is applied to the results obtained for the classifiers with GLCM and LBP features individually<sup>20</sup>. The similarity



**Fig. 10.** Receiver Operating Characteristic (ROC) of different Machine learning models with LBP features.

No. of images used from Dataset	No. of Images in each class	No. of training images	No of Validation images	No. of Epoch	Accuracy	Loss	Val Accuracy	Val Loss
1000	Atelectasis = 250 Infiltration = 250 Nodule = 250 Pneumonia = 250	800	200	25	0.7825	0.482	0.825	0.526
2000	Atelectasis = 500 Infiltration = 500 Nodule = 500 Pneumonia = 500	1600	400	25	0.8375	0.390	0.8625	0.414
3000	Atelectasis = 750 Infiltration = 750 Nodule = 750 Pneumonia = 750	2400	600	25	0.9183	0.328	0.87	0.402
4000	Atelectasis = 1050 Infiltration = 1049 Nodule = 852 Pneumonia = 1049	3200	800	25	0.9421	0.266	0.92	0.281
4500	Atelectasis = 1216 Infiltration = 1216 Nodule = 852 Pneumonia = 1216	3600	900	50	0.9611	0.103	0.9288	0.182
5000	Atelectasis = 1328 Infiltration = 1410 Nodule = 852 Pneumonia = 1410	4000	1000	50	0.9705	0.083	0.9570	0.096

**Table 5.** Results obtained using deep learning model (InceptionV3).

Reference	Methods	Data set	No of images	Accuracy
Rahaman et al. (23)	deep convolutional neural networks	COVID-19 benchmark dataset	860 images (260 COVID-19 cases, 300 healthy and 300 pneumonia cases)	89.3%
RajaramanS et al. <sup>24</sup>	convolutional neural networks	This dataset contained 238 abnormal CXRs and 729 healthy controls	This dataset contained 238 abnormal CXRs and 729 healthy controls	99.1%
Yu, Gordienko et al. <sup>25</sup>	Convolutional neural network	JSRT and BSE-JSRT data sets	247 + 247	96%
StefanusKieu Tao Hwa et al. <sup>26</sup>	Ensemble deep learning	Montgomery and Shenzhen datasets	Montgomery – 138 (80 normal and 58 images abnormal) Shenzhen – 662 (326 normal and 336 TB images)	89.77%
Antony, B <sup>27</sup>	K-nearest neighbor (KNN) & Sequential Minimal Optimization (SMO)	National Library of Medicine	662 (326 normal and 336 abnormal.)	KNN – 80% & SMO – 75%
Chouhan V <sup>28</sup>	Convolutional Neural Networks	Guangzhou Women and Children's Medical Center	5232 (1346 Normal 3883 Pneumonia)	96.40%
Purposed Method	Deep learning	National Institutes of Health	8,000 (3,000 benign & 5,000 malignant) Atelectasis (1328), Infiltration (1410), Nodule (852) and Pneumonia (1410)	97.03%

**Table 6.** Comparison of proposed method with existing studies.

index obtained for the 8 attributes is  $C_1^* = 0.708$ ,  $C_2^* = 0.461$ ,  $C_3^* = 0.712$ ,  $C_4^* = 0.625$ ,  $C_5^* = 0.591$ ,  $C_6^* = 0.572$ ,  $C_7^* = 0.483$ ,  $C_8^* = 0.323$  and the optimal sequence of the options can be established as  $A_3 > A_1 > A_4 > A_5 > A_6 > A_7 > A_2 > A_8$ . From this ranking, the third attribute SVM classifier with RBF kernel gives the better classification result than the other classifiers. The same method is also applied for the results obtained for LBP features. The similarity index obtained for the 8 attributes is  $C_1^* = 0.3319$ ,  $C_2^* = 0.562$ ,  $C_3^* = 0.356$ ,  $C_4^* = 0.8617$ ,  $C_5^* = 0.582$ ,  $C_6^* = 0.5801$ ,  $C_7^* = 0.42$ ,  $C_8^* = 0.726$  and the optimal sequence of the options can be established as  $A_4 > A_8 > A_5 > A_6 > A_2 > A_7 > A_1 > A_3$ . With the LBP features, the fourth attribute decision tree gives better results compared with other classifiers.

### Deep learning method

The results obtained for the abnormal images, namely Atelectasis, Infiltration, Nodule and Pneumonia using the pretrained InceptionV3 with the image size of  $299 \times 299 \times 3$  and the results obtained are shown in Table 5.

In this model also, 80% of dataset is used for training and 20% is used for testing. The classification accuracy of the InceptionV3 model is increased when the number of dataset is increased and is shown in Table 5. For the dataset of 5000 malignant images, this model gives the classification accuracy of 97.05%.

The comparison of proposed method with existing studies is given in the Table 6. Considering the number of images the proposed method shows the better accuracy than the previous studies.

# Conclusion

In this paper, a comprehensive machine learning algorithms and deep learning model Inception V3 is presented for lung disease classification into benign and malignant categories, as well as into four distinct disease classes. Lung X-ray images are tested with various possibilities like various size of image dataset and with various machine learning algorithms. Decision tree classifier with GLCM features provides highest accuracy of 77% and specificity of 84.8%. With LBP features, SVM with RBF kernel gives the specificity of 87.7% and accuracy of 81.5%. When both statistical and textural features are considered, the SVM classifier achieves an accuracy of 85.25%, with all other parameters showing higher performance compared to the other classifiers. Given the availability of multiple attributes, a multi-criteria decision-making model, TOPSIS, is applied in this study. It results SVM model with RBF as top ranking model with GLCM feature features and decision tree in LBP features. In both cases, SVM and DT provides approximately the same performance. The entire image dataset is tested with InceptionV3 model. In this study, the deep learning classifier achieves an accuracy of 97.05% with 5,000 images, which is 18.41% higher than the accuracy obtained with 1,000 images. Additionally, it outperforms the highest accuracy achieved by machine learning approaches by 11.8%, while the loss is 40% lower when compared to the results with fewer images. Moving ahead, as research and refinement persist, the utilization of deep learning for lung disease detection is poised to become even more central in healthcare, reshaping approaches to diagnosing and managing lung conditions.

# Data availability

Data availability The datasets used and analyzed during the current study are available from the corresponding author on request.

Received: 17 May 2024; Accepted: 24 January 2025

Published online: 12 March 2025

# References

1. Geraldo & Luis Lung disease detection using feature extraction and extreme learning machine. *Rev. Bras. Eng. Bioméd* (2014). 303.
2. Rahib, H. Abiyev Deep Convolutional Neural Networks for Chest Diseases Detection, Vision Based Computing Systems for Healthcare Applications, *J. Healthc. Eng.* **2018**, 4168538 (2018).
3. Qin, C. & Yao, D. Computer-aided detection in chest radiography based on artificial intelligence: a survey, *BioMedical Engineering Online*, **17**, 113 (2018).
4. Sri Widodo RatnasariNurRohmah, Lung diseases detection caused by smoking using support vector machine, *TELKOMNIKA*, **17**(3), 1256–1266 (2019).
5. SubratoBharati, P. P. Disease Detection from Lung X-ray images based on Hybrid Deep Learning, *Informatics in Medicine* 2020
6. Caixia Liu, R. Lung segmentation based on random forest and multi-scale edge detection. *IET Image Proc.* **13** (Issue 10), 1745–1754 (2019). Zhao MingyongPang.
7. Li, X. X. Bin Li · Lian-Fang Tian, Li Zhang, Automatic benign and malignant classification of pulmonary nodules in thoracic computed tomography based on RF algorithm, *IET Image Processing*, **12**(7), 1253–1264 (2018).
8. Khan, W., Zaki, N. & Ali, L. Intelligent Pneumonia Identification from chest X-Rays: A systematic literature review. *IEEE Access.* **9**, 51747–51771 (2021).
9. GhostanKhatchatoorian, A. & Jamzad, M. Architecture to improve the accuracy of automatic image annotation systems, *IET Computer Vision*, **14**(5), 214–223, Apr. (2020).
10. Paris, M. T. & Mourtzakis, M. Muscle composition analysis of ultrasound images: A narrative review of texture analysis, *Ultrasound Med. Biol.*, **47**(4), 880–895, Apr. (2021).
11. Al-Shakarchy, N. D. Drowsy Detection based on Spatiotemporal Feature Extraction of Video Using 3D-CNN, *J. Adv. Res. Dyn. Control Syst.*, **11**(10)-SPECIAL ISSUE, 742–751, Oct. (2019).
12. Elshennawy, N. M. & Ibrahim, D. M. Deep-Pneumonia Framework Using Deep Learning Models Based on Chest X-Ray Images, *Diagnostics*, **10**(9), 649, Aug. (2020).
13. Fauzan, A. R., Wahyuddin, M. I. & Ningsih, S. Pleural effusion classification based on chest X-ray images using convolutional neural network, *JurnalIlmuKomputerdanInformasi*, **14**(1), 9–16, (2021).
14. Magar, H., Patil, S. J., Waykole, S. R., Sandikar, S. D. & Parakh, N. D. Pneumonia detection using X-Ray images with deep learning. *Int. J. Comput. Appl. Technol. Resarc.* **9**(5), 183–185 (May 2020).
15. Erdem, E. & Aydin, T. *Detection of Pneumonia with a Novel CNN-based Approach* ( Sakarya University Journal of Computer and Information Sciences, Dec. 2020).
16. Gupta, P. Pneumonia detection using convolutional neural networks, *Int. J. Mod. Trends Sci. Technol.*, **7**(01), 77–80, Jan. (2021).
17. Raut, S. Pneumonia detection using deep learning approach, *Int. J. Res. Appl. Sci. Eng. Technol.*, **9**(4), 1625–1638, (2021).
18. Chavda, S. & Goyani, M. Hybrid approach to content-based image retrieval using modified multi-scale LBP and color features, *SN Comput. Sci.*, **1**(6), (2020).
19. NamanGupta, D. G. AshishKhanna, Pedro P.Rebouças, FilhoVictor Hugo C.de Albuquerque, Evolutionary algorithms for automatic lung disease detection. *Measurement* **140**, 590–608 (2019).
20. Krohling, R. A. & Pacheco, A. G. C. A-TOPSIS – an Approach based on TOPSIS for ranking Evolutionary algorithms. *Procedia Comput. Sci.* **55**, 308–317 (2015).
21. Kieu, S. T. H., Bade, A., Mohd, H. A., Hijazi & HoshangKolivand A Survey of Deep Learning for Lung Disease Detection on Medical images: state-of-the-Art, taxonomy, issues and future directions. *J. Imaging* **6**(12), 131. <https://doi.org/10.3390/jimaging6120131> (2020).
22. Lee, H. et al. Performance of a Deep Learning Algorithm Compared with Radiologic Interpretation for Lung Cancer Detection on Chest Radiographs in a Health Screening Population, *Radiology*, **297**(3), 687–696, Dec. (2020).
23. Rahaman, M. M. et al. 'Identification of COVID-19 samples from chest X-Ray images using deep learning: A comparison of transfer learning approaches'. 1 Jan 821–839. (2020).
24. Rajaraman, S. et al. A novel stacked generalization of models for improved TB detection in chest radiographs. In Proceedings of the 2018 40th Annual International Conference the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, USA, 17–21 July, 718–721. (2018).
25. Gordienko, Y. et al. Deep Learning with Lung Segmentation and Bone Shadow Exclusion techniques for chest X-Ray analysis of Lung Cancer. *Adv. Intell. Syst. Comput.* 638–647. (2019).

26. Kieu, S. T. H., Hijazi, M. H. A., Bade, A., Yaakob, R. & Jeffree, S. Ensemble deep learning for tuberculosis detection using chest X-Ray and canny edge detected images. *IAES Int. J. Artif. Intell.* **8**, 429–435 (2019).
27. Antony, B. & Nizar Banu, P. K. Lung tuberculosis detection using x-ray images. *Int. J. Appl. Eng. Res.* **12**, 15196–15201 (2017).
28. Chouhan, V., Singh, S. K., Khamparia, A., Gupta, D. & Albuquerque, V. H. C.D. A novel transfer learning based Approach for Pneumonia detection in chest X-ray images. *Appl. Sci.* **10**, 559 (2020).
29. Manoj SenthilKailasam, M. D. & Thiagarajan Detection of lung tumor using dual tree complex wavelet transform and co-active adaptive neuro fuzzy inference system classification approach. *Int. J. Imaging Syst. Technol.* **31**, 2032–2046 (2021).
30. Manoj, K. S. & Meeradevi, T. Performance analysis of feature-based lung tumor detection and classification. *Curr. Med. Imaging* **13**(3) (2017).
31. Nabeel, S. M. et al. Optimizing lung cancer classification through hyperparameter tuning. *Digit. Health* **10**, 20552076241249661. <https://doi.org/10.1177/20552076241249661> (2024). PMID: 38698834; PMCID: PMC11064752.
32. Tekin, R. New local binary pattern approaches based on color channels in texture classification. *Multimedia Tools Appl.* **79**, 43–44 (2020).
33. Zeba Ghaffar, P. M. et al. Comparative analysis of state-of-the-art deep learning models for detecting COVID-19 lung infection from chest X-Ray Images, 2022, arXiv:2208.01637 [eess.IV].
34. Shah, P. M. et al. Deep GRU-CNN model for COVID-19 detection from chest X-Rays data. *IEEE Access.* **10**, 35094–35105 (2021).
35. Kandhro, I. A. et al. Performance evaluation of E-VGG19 model: Enhancing real-time skin cancer detection and classification. *Heliyon* **10**(10), e31488. <https://doi.org/10.1016/j.heliyon.2024.e31488> (2024)

## Author contributions

Author contributions Conceptualization, T.M, S.S, L.M, N.M, and K.R.; Data curation, T.M, S.S, L.M, N.M, and K.R.; Analysis and Validation, T.M, S.S, L.M and K.R.; Formal analysis, T.M, S.S, L.M, N.M, and K.R.; Investigation, T.M, S.S, L.M, N.M, and K.R.; Methodology, T.M, S.S, L.M, N.M, and K.R.; Project administration, T.M, S.S, L.M, N.M, and K.R.; Resources, T.M, S.S, L.M, N.M, and K.R.; Software, T.M, S.S, L.M and K.R.; Supervision, T.M, S.S and K.R.; Validation, T.M, S.S, L.M, N.M, and K.R.; Visualization, T.M, S.S, L.M, N.M, and K.R.; Writing—original draft, T.M, S.S, L.M, N.M, and K.R, Data Visualization, Editing and Rewriting, T.M, S.S, L.M, N.M, and K.R.

## Declarations

### Competing interests

The authors declare no competing interests.

### Additional information

**Supplementary Information** The online version contains supplementary material available at <https://doi.org/10.1038/s41598-025-88188-w>.

**Correspondence** and requests for materials should be addressed to K.R.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

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

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

© The Author(s) 2025