VER
Vascular & Endovascular Review

Advancing Clinical Outcomes through Diverse Machine Learning Techniques In Pulmonary Nodule Detection

Archaana Randive[1], Pramod Sharma[2]

[1]Research Scholar, E & TC Department, University of Technology, Jaipur, India
[1]Assistant Professor, E & TC Department, AISSM College of Engineering, Pune, Maharashtra, India.
[2]Professor & Dean, School of Engineering& Technology, UOT, Jaipur.
archu.ran06@gmail.com
pramod.sharma@deepshikha.org

ABSTRACT

Pulmonary nodules are a major clinical concern because they are hard to see and could be cancerous. They are often signs of early-stage lung cancer. Early and accurate detection is very important for making things better for patients. New developments in machine learning (ML) have shown that they can change the diagnostic process in a big way by making it easier and more accurate to find pulmonary nodules in medical imaging data. This paper looks at how different machine learning techniques, from simple supervised methods to more advanced deep learning methods, can be used to find and classify pulmonary nodules. We look at how convolutional neural networks (CNNs), support vector machines (SVMs), k-nearest neighbours (k-NN), and ensemble learning models can be used together to find nodules in chest X-rays and computed tomography (CT) scans that are different sizes and shapes. We also talk about the pros and cons of each method, focussing on how training data, feature selection, and the model's readability affect its ability to make diagnoses. It is also talked about how to compare mixed models that use more than one machine learning method to get the best results. The main point of this review is to stress how important it is to use multi-modal learning and big, labelled medical datasets to make these models work better. We also stress how important it is to validate models through thorough clinical studies to make sure that ML models can be used in real-life healthcare situations. In the last part of the paper, future trends in finding pulmonary nodules are talked about.

KEYWORDS: Pulmonary Nodule Detection, Machine Learning Techniques, Convolutional Neural Networks (CNN), Lung Cancer Diagnosis, Medical Imaging Analysis.

How to Cite: Archaana Randive, Pramod Sharma, (2025) Advancing Clinical Outcomes through Diverse Machine Learning Techniques In Pulmonary Nodule Detection, Vascular and Endovascular Review, Vol.8, No.7s, 160-171.

INTRODUCTION

Pulmonary nodules, which are growths or lumps that don't belong in the lungs, are often the first sign of lung cancer, which is the most common type of cancer in the world. Finding and correctly diagnosing lung tumours as soon as possible is very important for better patient results, since getting care right away can greatly increase life rates. The problem with finding lung nodules is that they are not always easy to see because they can be very small, very round, or very dense. This makes it hard to tell them apart from healthy tissue, especially in the early stages. Imaging tools like chest X-rays, computed tomography (CT), and magnetic resonance imaging (MRI) have come a long way, making it much easier to find and see lung tumours. But figuring out what these pictures mean is still hard, and radiologists are the only ones who can do it right. They have to look at a lot of details to decide if a growth is normal or cancerous. The old way of finding nodules depends a lot on the experience and personal opinion of the doctors, which makes the diagnosis less consistent. There is also a pressing need for faster, more reliable, and automated ways to find and classify pulmonary tumours because of the rising rate of lung cancer and the growing amount of medical imaging data. New developments in machine learning (ML) have shown a lot of promise in tackling these problems by making it possible to create automated diagnosis tools that can look at medical pictures more accurately and consistently than humans. Medical picture analysis has changed a lot because of machine learning, especially deep learning. A type of deep learning models called convolutional neural networks (CNNs) have been used a lot to find and sort lung tumours because they can automatically learn hierarchical features from raw picture data.

The performance of these models is better than traditional image processing methods, and they have been shown to cut down on human mistake while making tumour identification more accurate. Other machine learning methods, like Support Vector Machines (SVM), k-Nearest Neighbours (k-NN), and ensemble learning, have also been looked into for finding nodules. Each has its own benefits that rely on the dataset and job. The goal of this study is to give an outline of the different machine learning methods that have been used to find lung nodules. We will talk about the pros and cons of both old and new machine learning techniques, with a focus on deep learning algorithms, which have gotten a lot of attention lately for their ability to handle a lot of medical image data. Figure 1 shows improving pulmonary nodule detection using various machine learning techniques for diagnosis.

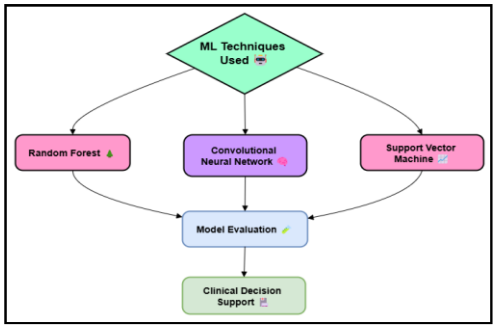

Figure 1: Advancing clinical outcomes through diverse machine learning techniques in pulmonary nodule detection

The study will also talk about how mixed models, which use more than one machine learning method, have been created to improve the finding and grouping of lung tumours. One big problem with using machine learning to find nodules is getting good labelled datasets that are easy to find. Most of the time, medical imaging samples are small and don't fully show the range of tumours that are seen in clinical practice. ML models rely a lot on the quality and variety of the training data they are given. Bad training data can introduce biases and make the models less useful in real life. In healthcare uses, the interpretability of ML models is still a problem because doctors need to be able to trust and understand the decisions these models make. Because of this, methods called explainable AI (XAI) are being added to machine learning systems to make the decision-making process more open and clear.

RELATED WORK

Over the past ten years, a lot of study has been done on how to use machine learning (ML) methods to find pulmonary nodules. This is because early and accurate lung cancer screening is becoming more and more important. Traditional image processing methods, like edge recognition, thresholding, and texture analysis, were used in one of the first attempts to automatically find nodules [1]. These methods were useful in some situations, but they often had trouble with lung tumours because their form, size, and density can change a lot. As more advanced machine learning algorithms came out, these older methods were slowly replaced or added to by ones that could learn patterns straight from the data [2][3]. Implementing Convolutional Neural Networks (CNNs) for finding and categorising lung tumours was a major step forward in the field. CNNs are very good at handling medical pictures because they can easily pull out spatial features and find trends in pixel data. A lot of research has shown that CNN-based models are better than standard algorithms at finding nodules because they are more sensitive and detailed [4]. For instance, Xie et al. showed that a deep learning model could be used to find nodules in CT scans, with a success rate of over 90%.

This worked a lot better than standard machine learning models like Support Vector Machines (SVMs) or k-Nearest Neighbours (k-NN), which were used before in this field. Hybrid models, which blend several machine learning methods to improve performance, have led to even more progress [5]. Ensemble learning methods have shown promise in finding lung nodules. These methods mix the outputs of several models to make predictions more accurate. Additionally, Rathore et al. (2018) showed how using a group of CNNs along with SVM models can help find small spots that single-model methods often miss [6]. More and more people are using these mixed methods because they work around the problems with single models and make the most of their combined strengths. There is growing interest in explainable AI (XAI) methods to improve the interpretability of machine learning (ML) models, which is very important in medical uses [7]. Deep learning models like CNNs have produced amazing outcomes, but their "black-box" nature has caused worry among doctors who need clear and understandable ways to make decisions. Recent research has looked into how to use XAI techniques to find lung nodules. These studies show how models make their findings [8]. Grad-CAM (Gradient-weighted Class Activation Mapping), for example, has been suggested as a way to see the parts of the CT scan that were most important to the model's choice [9].

METHODOLOGY

The method starts with getting CT scans of lung tumours and then doing some basic processing like shrinking, greyscale conversion, and normalisation. For classifying nodules, different machine learning methods are used, such as Random Forest, SVM, Logistic Regression, CNN, and Hybrid CNN-LSTM. Accuracy, F1-score, and model stability measured by cross-validation are used to judge performance.

1. Load Dataset

DATASET 1: IQ-OTH/NCCD - LUNG CANCER DATASET

The Figure 2 looks like a collection of CT scan images that have been separated into three groups: Normal, Benign, and Malignant. The IQ-OTH/NCCD Lung Cancer Dataset has these pictures. This dataset is often used to teach machine learning models how to find lung cancer. In the first row, "Benign" pictures show lung tumours that are not deadly. Most of the time, these lumps have smooth sides and might not be a threat right away [14]. Even though they are harmless, they still need to be watched to make sure they don't change over time. The pictures in the second row are called "malignant," and they include lumps that look like they might be tumour growths. Nodules that are cancerous often have edges that are jagged, uneven, or irregular [15]. This could mean that cancer cells are present and need quick medical attention. The pictures in the third row are Normal, which means they don't show any problems or tumours. These are used as a starting point to help tell the difference between lung tissue that is healthy and tissue that has normal or cancerous problems [16]. This collection is very important for the development of diagnostic

tools because it helps machine learning models learn to tell the difference between these groups, which is very important for finding lung cancer early and making patients' lives better.

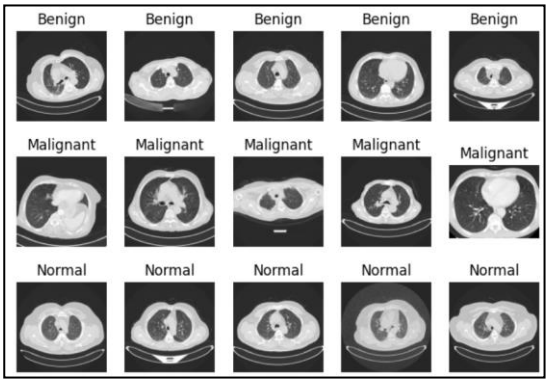

Figure 2: Dataset Sample

The Figure 3 shows bar chart you gave shows how the data points in the IQ-OTH/NCCD Lung Cancer Dataset are spread out among three groups: Normal, Benign, and Malignant. There is a different colour for each type: blue for benign, red for malignant, and green for normal. It's clear from the chart that the Malignant group has the most cases, with over 500 of them. This means that the dataset likely has a lot of cases of cancer, which is important for teaching machine learning models how to correctly find dangerous tumours.

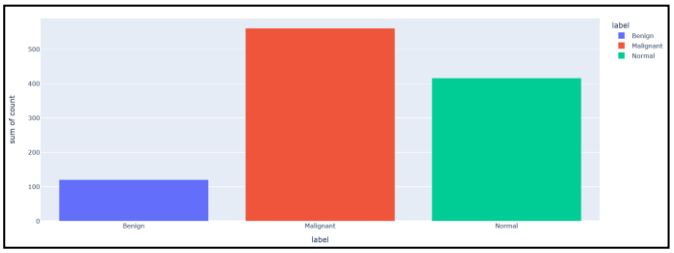

Figure 3: Dataset Sample Distribution

The next group, "Normal," has a modest number of cases and shows lungs that are healthy and have no problems. There are just over 100 cases in the Benign group, which is the fewest. This might be because of the way the dataset naturally spreads out normal lung problems or because it's hard to get data for tumours that aren't deadly. This distribution shows how important it is to use fair data when creating a model, since too much of one type could cause estimates to be wrong. A fair sample is necessary to make the model more general and better at all types of tasks.

DATASET 2: LUNG CANCER PREDICTION USING EFFICIENTNET B5

Normal and Adenocarcinoma lung CT scan pictures are shown next to each other. This Figure 4 shows the changes between healthy lung tissue and adenocarcinoma, which is one of the most common types of lung cancer [17]. In the first row, you can see pictures of normal lungs, which don't have any problems or tumours obvious. These pictures are the starting point. They show lung structure and tissue that are healthy, with no growths or tumours present. Adenocarcinoma is a type of non-small cell lung cancer shown in the bottom row [18]. Usually, adenocarcinoma shows up in the lungs as a lump or mass with rough edges. In some pictures, you can see growths or tumours that look like adenocarcinoma.

These pictures show that the lung tissue isn't working right because they show lumps that look like dangerous growths. This dataset is useful for teaching deep learning models how to tell the difference between healthy and dangerous tissue in the context of predicting lung cancer using EfficientNet B5. A deep learning framework called EfficientNet B5 is known for how quickly it can process high-resolution pictures like CT scans.

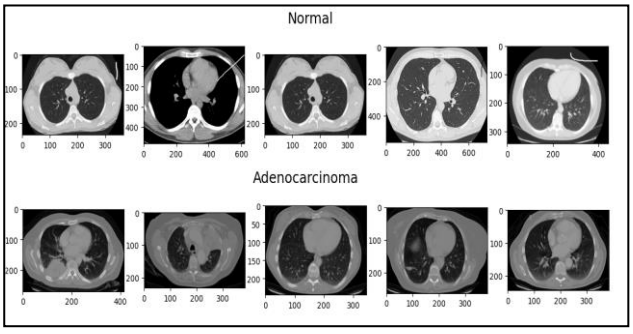

Figure 4: Dataset Sample

2. Data Pre-Processing

A. Read Image

Reading the picture is the first step in data pre-processing. This is a very important step when working with image-based datasets, especially medical photos like CT scans. Many people use tools like OpenCV, PIL (Python Imaging Library), or TensorFlow and PyTorch to handle images and read images [19]. When reading pictures, it's important to make sure that the type and style are right for processing. The system can read images in forms like PNG, JPEG, and TIFF as groups or tensors, which makes it easy to do more work on them [20]. In this step, the picture is loaded into memory so that it can be changed, edited, and put into machine learning models. For deep learning uses, pictures are often turned into arrays where each pixel is represented by a number that corresponds to a colour intensity (RGB or greyscale values).

Let the input image be denoted by I.

I = Read(path) (1)

I_array=Convert(I, array) # Convert the image to an array of pixel value (2)

B. Resize

Another important part of picture pre-processing is resizing. This is especially true for deep learning tasks, since neural networks need inputs that are all the same size. Medical pictures like CT scans can have different resolutions, so shrinking them to a common size makes sure that the model can easily process all of them. Changing the size of pictures to standard sizes for many convolutional neural networks (CNNs), like 224x224 or 256x256 pixels, is a popular method. The process of resizing should keep the original picture's aspect ratio, but it may also crop or pad the image to get it to the right size [21]. It is important to resize pictures because big ones might have more data than they need, which makes training models take longer and use more computing power. In addition to making the computer work faster, lowering the size of the picture helps the model work better by focussing on important features instead of small details that aren't needed.

Desired size= (hnew, wnew) (3)

Resize operation: I resized = Resize(I,(hnew, wnew)) (4)

C. Gayscale Conversion

When working with medical pictures, changing them to greyscale is an important step, especially when the main focus is on structure features rather than colour information. For example, CT scan pictures are usually greyscale, but they may be shown in RGB colour bands to make them easier to work with. By going from three channels (RGB) to one (intensity), greyscale conversion makes the data simpler. This can improve model performance and make computations simpler. Individual pixels in a greyscale picture have values that show how bright the light is. The values of these pixels range from black (lowest intensity) to white (highest intensity). In greyscale conversion, colour information is thrown away and only the light strength is kept. This is enough for jobs like finding objects, dividing them into sections, and putting lung lumps into groups. This makes the data simpler, which lowers its complexity and speeds up the learning process without giving up the data needed to make accurate medical assessments. Using simple methods that average or combine the RGB channels into a single intensity channel, tools like OpenCV or PIL make it easy to change RGB pictures to greyscale.

- Let the original image be I_rgb with 3 channels (RGB).
- Compute the grayscale value using:

*Igray= 0.2989 * IR + 0.5870 * IG+ 0.1140 * IB* (5)

- The output image I_gray is now a single-channel grayscale image.

3. Feature Visualization

A. PCA

The Figure 5 shows the Support Vector Machine (SVM) classifier's decision limit after Principal Component Analysis (PCA) was used to reduce the number of dimensions. This is a common way to make complex data easier to understand by reducing its number of dimensions while keeping its range.

The plot shows three groups: Normal (green), Benign (blue), and Malignant (orange). These groups show different lung cancer conditions. PCA takes the data from its original high-dimensional space and divides it into two parts, PCA Component 1 and PCA Component 2. These parts are then drawn on the X and Y axes. In the picture, the SVM decision border that separates the different types of data is shown.

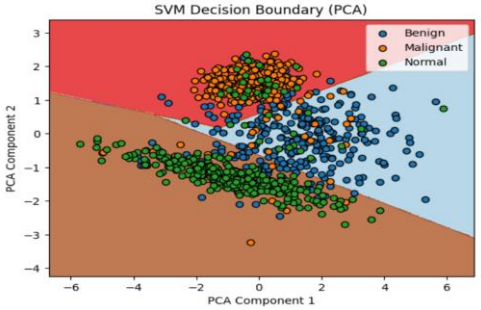

Figure 5: SVM Decision Boundary

4. Apply Machine Learning And Deep Learning Algorithm

A. Random Forest

Random Forest is an ensemble learning method that uses more than one decision tree to make a model work better and be more accurate overall. It works by teaching several decision trees with random parts of the training data. At each split, a different random feature group is used to build the tree. The final prediction is made by adding up the results of all the trees, either by voting (for classification) or averaging (for regression). Compared to a single decision tree, this method cuts down on overfitting and improves generalisation. Random Forest is very good at predicting lung cancer because it can work with big, complicated datasets like CT scan images or other medical data and can find connections between traits that don't follow a straight line. It also has the benefit of estimating the value of features, which helps figure out which features make the most difference in the results.

- Given a dataset X = {x_1, x_2, ...,x_n} with labels y = {y_1, y_2, ..., y_n}, }, where each x_i∈ℝ^m.
- Bootstrap sampling: For each tree, randomly sample the data X* ⊂ X with replacement.
- At each node, select the best split based on a criterion like Gini impurity or Information Gain:

$$Gini(t) = 1 - \sum_{\{k=1\}}^{\{K\}} p_k^2 \quad (6)$$

Where pk is the proportion of class k in node t.

- The prediction for each tree is given by:

$$\hat{y} = Majority\ Vote(Tree\ Predictions) \quad (7)$$

B. SVM

Support Vector Machine (SVM) is a strong guided machine learning method that is often used to sort things into groups. To use SVM, you need to find the hyperplane in a high-dimensional feature space that best separates the data points of different classes. Its goal is to find the largest gap between data points from different classes. The data points closest to the decision line are called support vectors. When it comes to finding lung cancer, SVM is very useful because it can sort complicated and non-linear data by using various kernel functions, such as linear, polynomial, or radial basis function (RBF). SVM works well even in places with a lot of dimensions because it is based on strong theory. This means it can be used to do things like sorting CT scans or pathology pictures. For SVMs, however, the kernel you choose and how you tune hyperparameters like the regularisation parameter and kernel parameters can make a difference.

Given training data , Where x_i∈ℝ^m and y_i∈ {-1, 1}.

- The decision function is:

$$f(x) = w^T x + b \quad (8)$$

where w is the weight vector and b is the bias term

- The optimization problem:

$$\min_{\{w,b\}}^{2}\left(\frac{1}{2}\right)||w|| \quad (9)$$

subject to the constraints y_i (w^Tx_i + b) ≥ 1.

- The final decision is:

$$\hat{y} = sign(f(x)) \quad (10)$$

C. Logistic Regression

Logistic Regression is a type of statistical model used to classify things into two groups. By using a logistic function, which is also called the sigmoid function, to describe the chance of a class name (like "Malignant" vs. "Benign"), it works. The logistic regression model gives you a number between 0 and 1 that tells you how likely it is that a data point belongs to a certain class. Logistic regression works best with simple datasets where classes can be separated linearly. This is because it assumes that the independent factors have a linear relationship with the log chances of the dependent variable. When looking for lung cancer, logistic regression can be used to divide patients into three groups: those with benign, cancerous, or normal conditions. These groups are based on information from CT scans, patient traits, and clinical data. Logistic regression is simple to understand and quick to run on computers, but it might not work well on large datasets with connections that don't follow a straight line.

- Given training data
- The logistic function is:

$$P(y = 1 \mid x) = \sigma(w\text{^}T\ x + b) = 1 / (1 + e\text{^}\{-(w\text{^}T\ x + b)\}) \quad (11)$$

- The loss function (logistic loss) is:

$$L(w,b) = -\sum_{\{i=1\}}^{\{n\}}\left[y_i \log\left(\sigma(w^T x_i + b)\right) + (1 - y_i)\log\left(1 - \sigma(w^T x_i + b)\right)\right] \quad (12)$$

- The optimization objective is to minimize the loss function:

$$\hat{y} = argmin_{\{w,b\}L(w,b)} \quad (13)$$

D. CNN

Convolutional Neural Networks (CNNs) are a type of deep learning models that are intended to handle and analyse data that looks like a grid, like pictures. CNNs are made up of layers that use convolutional processes to naturally learn how to arrange features in space. What makes CNNs special is that they can pull out important patterns from pictures, like edges, colours, and shapes, without having to do it by hand. CNNs have been shown to be very good at looking at medical pictures like CT scans to find lung cancer. CNNs have many layers, including convolutional layers, pooling layers, and fully linked layers. These layers

work together to improve the picture data and learn more abstract representations. The convolutional layers take an input picture and apply different filters (called "kernels") to it to find basic features. The pooling layers help lower the size of the image and focus on more important features. CNNs can learn very complicated patterns and have reached the top of the field in jobs that use images to classify them.

- Given an input image

where h is the height, w is the width, and c is the number of channels.

- The convolution operation with filter K ∈ ℝ^{f_h × f_w × c} is:

$$I_{conv(x,y)} = \sum_{\{i=1\}}^{\{f_h\}} \sum_{\{j=1\}}^{\{f_w\}I} (x + i, y + j)K(i,j) \quad (14)$$

- . Apply non-linearity using activation function σ (e.g., ReLU):

$$I_{out(x,y)} = \sigma\left(I_{conv(x,y)}\right) \quad (15)$$

Model: "sequential_5"

Layer (type)	Output Shape	Param #
conv2d_15 (Conv2D)	(None, 126, 126, 32)	320
max_pooling2d_15 (MaxPooling2D)	(None, 63, 63, 32)	0
conv2d_16 (Conv2D)	(None, 61, 61, 64)	18,496
max_pooling2d_16 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_17 (Conv2D)	(None, 28, 28, 128)	73,856
max_pooling2d_17 (MaxPooling2D)	(None, 14, 14, 128)	0
flatten_5 (Flatten)	(None, 25088)	0
dense_10 (Dense)	(None, 512)	12,845,568
dropout_5 (Dropout)	(None, 512)	0
dense_11 (Dense)	(None, 3)	1,539

Total params: 38,819,339 (148.08 MB)
Trainable params: 12,939,779 (49.36 MB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 25,879,560 (98.72 MB)

E. Hybrid CNN- LSTM

A Hybrid CNN-LSTM model takes the best parts of both Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks and puts them together. This makes it a useful design for jobs that need to deal with both spatial and temporal data. CNNs are famous for being able to pull out spatial data from pictures. This makes them perfect for jobs like medical image analysis, object recognition, and classifying images. LSTMs, on the other hand, are a type of recurrent neural network (RNN) that is intended to work with sequential data and understand how things change over time and how they are related to each other over time. When CNNs are put together, they first use convolutional layers to handle the picture data and pull out important spatial elements like lines, colours, and patterns. The LSTM layers then get these separated features and learn how the features are related to each other over time.

- Given input sequences X = {x_1, x_2, ..., x_n}

where each x_i∈ℝ^m.

- Apply convolutional layers to extract spatial features:

$$I_{conv} = Conv2D(X) \quad (16)$$

- Pass the features through LSTM layers to capture temporal dependencies:

$$h_t = LSTM\left(I_{conv}, h_{\{t-1\}}\right) \quad (17)$$

- The final output is obtained after a dense layer:

$$\hat{y} = softmax(W \cdot h_T + b) \quad (18)$$

whereh_T is the final hidden state from the LSTM.

RESULTS AND ANALYSIS

With an accuracy of 99.4% and an F1-score of 99.1%, the data show that Hybrid CNN-LSTM does better than other models. CNN also did very well, but there was some overfitting. It was harder to use SVM and Logistic Regression. These results show that deep learning methods, especially mixed models, make it much easier to find lung nodules and make clinical decisions.

1. DATASET 1 RESULTS:

A. Accuracy and Loss Curve

The accuracy and loss curves of a Hybrid CNN-LSTM model over 50 epochs are shown in Figure 6. The Model Accuracy curve (left) shows that the training accuracy starts at about 0.5 and quickly rises to almost 1 (100%). It reaches a peak of 1.0 around the 20th generation. The orange line shows the confirmation accuracy, which follows a similar path but is a little behind the training accuracy. It levels off at about 0.9 near the end of training. This shows that the model is learning well and is able to apply what it has learnt to the evaluation data.

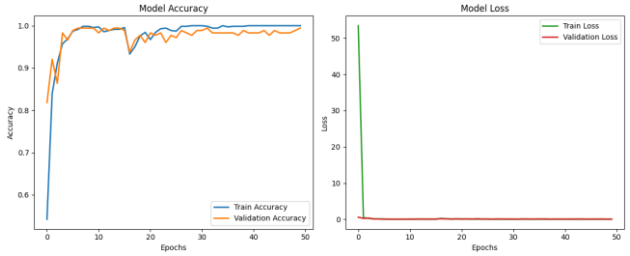

Figure 6: Accuracy and Loss Curve of CNN

The training loss (green) starts out pretty high, around 50, but drops very quickly after the first few epochs, getting very close to 0 by epoch 20. This is shown in the Model Loss plot (right). In the same way, the confirmation loss (red) starts out big and slowly goes down until it's almost zero. This means that the model learnt from both the training and test datasets and quickly reached an optimal state with little error.

Figure 7 displays the Hybrid CNN-LSTM model's accuracy and loss curves over 50 iterations. The Model Accuracy plot (left) shows that the training accuracy starts at about 0.5 and steadily rises until it reaches 1.0 around time 40. This shows that the model is learning well. Similar trends can be seen in the orange confirmation accuracy, which rises from about 0.5 and levels off at about 0.9 by epoch 50.

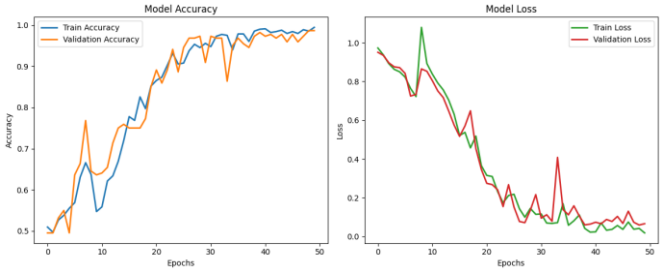

Figure 7: Accuracy and Loss Curve of Hybrid CNN –LSTM

The validation accuracy curve stays pretty close to the training accuracy, which means that the model is doing well with data it hasn't seen before and isn't overfitting too much. On the right, the Model Loss map shows that the training loss (green) starts out high, around 1.0, and drops quickly, getting close to 0.1 by the end of the training. A similar pattern can be seen in the confirmation loss (red), which goes from 1.0 to about 0.2. These falling loss values show that the model is doing a good job of reducing mistakes. Also, both the training and validation losses have levelled off, which means that the model is well-trained and has good generalisation performance.

B. Confusion Matrix

The given confusion matrix shows how well a Random Forest classifier works by putting guesses into three groups: Class 0, Class 1, and Class 2. Figure 8 shows the confusion matrix for the Random Forest model. The labels in the columns are the expected labels, and the labels in the rows are the real labels. In Class 0, the model got 14 of the predictions right, but got 3 wrong as Class 1 and 12 wrong as Class 2. This shows that Class 0 and the other two classes are not always clear. The model did a good job with Class 1, properly identifying 108 cases as Class 1. However, it wrongly labelled one case as Class 2 and none as Class 0, showing that it did a good job of finding Class 1. In Class 2, the model got 76 of the predictions right as Class 2, but it got 6 wrong as Class 1 and 0 wrong as Class 0. In general, the model is pretty accurate, especially for Class 1. However, it makes some mistakes when deciding between Classes 0 and 2, which could be fixed with more data or fine-tuning.

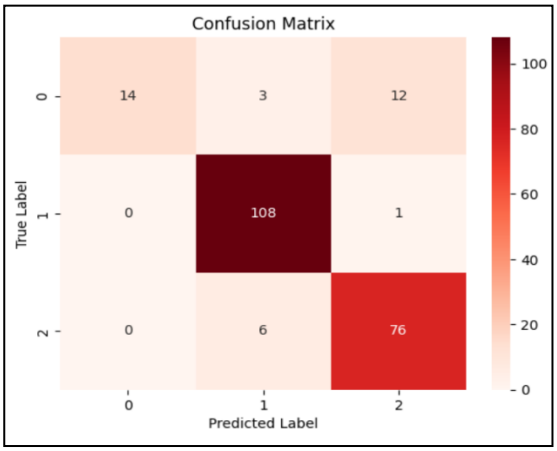

Figure 8: Confusion Matrix of Random Forest

A Support Vector Machine (SVM) model sorts cases into three groups: Class 0, Class 1, and Class 2. The confusion matrix shows how well this model works. The program got 59 of the predictions for Class 0 right, but got 5 wrong for Class 1 and 13 wrong for Class 2. It looks like there is some confusion between Class 0 and the other two classes. In Class 1, the model did a good job; it properly identified 84 cases as Class 1, misclassified none to Class 0, and only one to Class 2. Figure 9 shows the confusion matrix for the SVM model. The algorithm got 79 of the predictions for Class 2 right, but it got 5 wrong as Class 1 and 0 wrong as Class 0. Overall, the SVM model is very accurate, especially for Class 1. It doesn't make many mistakes when putting things into different groups.

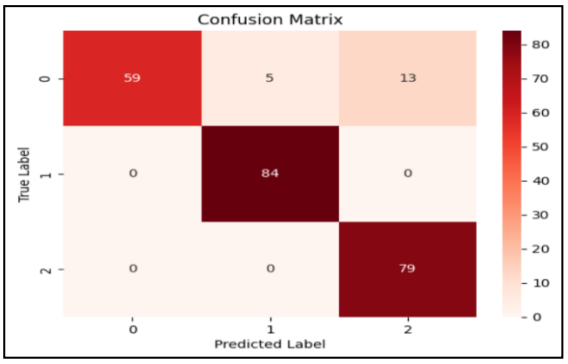

Figure 9: Confusion Matrix of SVM

The given confusion matrix displays how well a Logistic Regression model with three groups—Class 0, Class 1, and Class 2—works. The program got 59 of the predictions for Class 0 right, but got 11 wrong for Class 1 and 7 wrong for Class 2. This shows that there is some misunderstanding between Class 0 and the other classes. In figure 10, In Class 1, the model did a good job; it correctly identified 84 instances as Class 1, misclassifying none to Class 0 and just one to Class 2. For Class 2, the model properly guessed 76 cases as Class 2, mistakenly putting three cases in Class 0 and none in Class 1. Overall, the Logistic Regression model works well, especially for Class 1. However, there is some misunderstanding between Class 0 and Class 2, which could be fixed by adding more traits or fine-tuning the model.

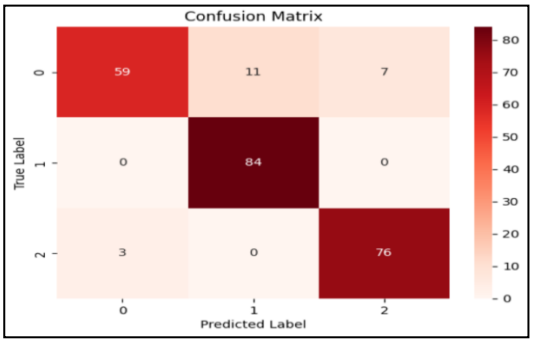

Figure 10: Confusion Matrix of Logistic Regression

The given confusion matrix shows how well a Hybrid CNN-LSTM model works at sorting results into three groups: Normal, Benign, and Malignant. For Benign, the model got 28 of the predictions right as Benign, but got one wrong as Normal. Figure 11 shows the confusion matrix for the Hybrid CNN-LSTM model. No cases were wrongly labelled as Malignant, which means the Benign group did a good job.

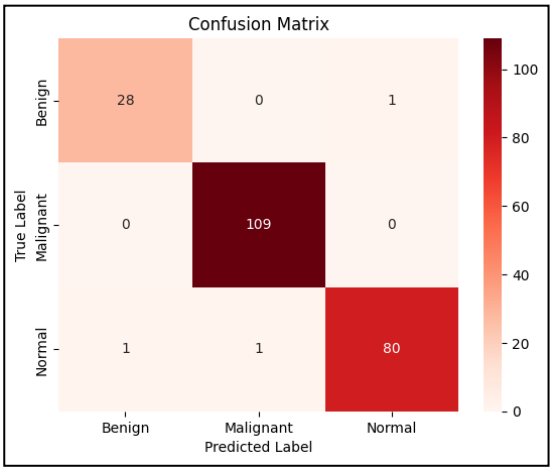

Figure 11: Confusion Matrix of Hybrid CNN – LSTM

For Malignant, the model did very well; it properly identified 109 cases as Malignant and mistakenly labelled one case as Normal. This means that the model is likely to be very good at finding cases of cancer. Eighty of the predictions for Normal were right. One was wrongly labelled as Benign, and one was wrongly labelled as Malignant.

C. Comparative Analysis

The precision and F1-scores for five machine learning models are shown in Table 2. These are Random Forest (RF), Support Vector Machine (SVM), Logistic Regression (LR), Convolutional Neural Network (CNN), and Hybrid CNN-LSTM. Starting with Random Forest (RF), it got an accuracy of 90%, but the F1-score was a little lower at 83%. This showed that the model did pretty well overall, but there were big differences in how well it did with precision and memory for some classes. With a 93% accuracy rate and a very high F1-score of 92%, the SVM model did a good job, showing that it is good at combining precision and memory.

Table 2: Performance Comparison of Machine Learning Models for Pulmonary Nodule Detection

Model	Accuracy (%)	F1-Score (%)
RF (Random Forest)	90	83
SVM (Support Vector Machine)	93	92
LR (Logistic Regression)	91	91
CNN (Convolutional Neural Network)	99	98
Hybrid CNN-LSTM	99.4	99.1

Logistic Regression (LR) did well, with an accuracy rate of 91% and an F1-score of 91%, which suggests it was good at separating positive and negative cases. Figure 12 shows comparative analysis of models using Dataset 1.

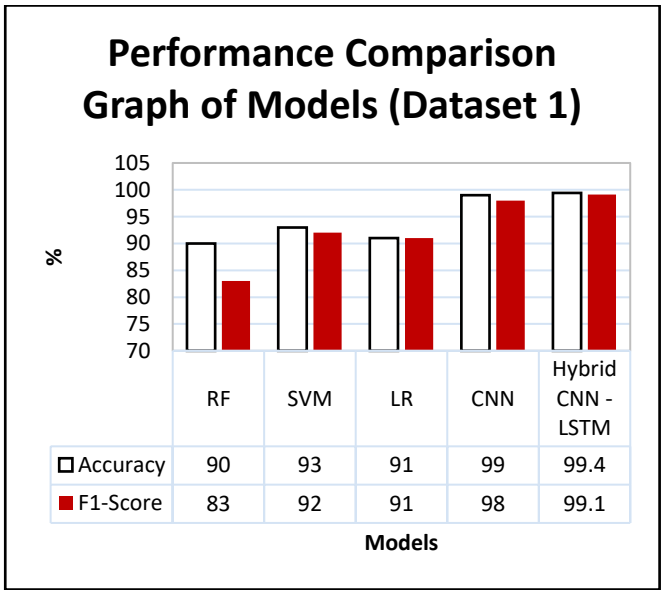

Figure 12: Comparative Analysis of Models (Dataset 1)

CNN did really well, getting an F1-score of 98% and a 99% accuracy rate, which shows how well it can pull out important traits from data. The Hybrid CNN-LSTM model did the best, with a 99.4% accuracy rate and a 99.1% F1-score, which means it was better at predicting the future and doing the job overall.

DATASET 2 RESULTS

A. Accuracy and Loss Curve

The Figure13 shows how a CNN model's accuracy and loss changed over 50 epochs. The blue line on the Model Accuracy plot shows that the training accuracy starts at about 0.3 and slowly rises until it reaches about 0.7 by the 50th epoch. Starting at about 0.4 and getting close to 0.6, the validation accuracy (orange) also shows improvement. This means that the model gets better at classifying data with each epoch. There isn't much difference in accuracy between training and validation, which means the model works well in real life and doesn't overfit. The training loss (green) in the Model Loss plot starts out high and peaks at 40. It then drops quickly and is almost zero by epoch 5. In the same way, the confirmation loss (red) stays around 0.2 after the initial drop. The fast drop in loss shows that the model is learning the job well and making few mistakes while training, with not too much or too little overfitting or underfitting.

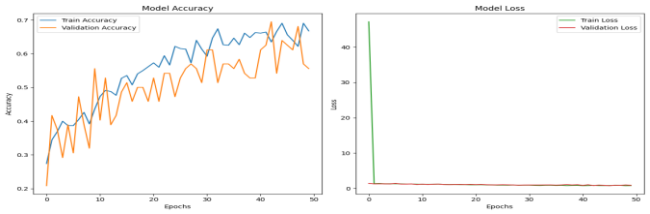

Figure 13: Accuracy and Loss Curve of CNN

The Figure 14 shows graph displays the Hybrid CNN-LSTM model's accuracy and loss curves over 50 iterations. On the left is a picture of the Model Accuracy. The training accuracy (blue) starts at about 0.4 and quickly rises until it reaches 1.0 in the 45th episode.

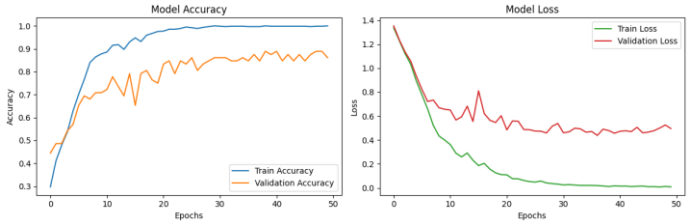

Figure 14: Accuracy and Loss Curve of Hybrid CNN – LSTM

The orange number for confirmation accuracy starts out lower at 0.5 and slowly rises to around 0.9 by the end of training. The difference in accuracy between training and validation stays pretty small, which means the model is generalising well and not fitting too well.

B. Comparative Analysis of Models

The precision and F1-scores for five different machine learning models are shown in Table 3. These are Random Forest (RF), Support Vector Machine (SVM), Logistic Regression (LR), Convolutional Neural Network (CNN), and Hybrid CNN-LSTM. Starting with Random Forest (RF), it got a 65% accuracy score and a 64% F1-score, which means it did pretty well. The difference between accuracy and F1-score, on the other hand, shows that precision and memory are not balanced.

Table 3: Baseline vs. Advanced Model Performance in Pulmonary Nodule Detection

Model	Accuracy (%)	F1-Score (%)
RF (Random Forest)	65	64
SVM (Support Vector Machine)	60	59
LR (Logistic Regression)	52	54
CNN (Convolutional Neural Network)	56	55
Hybrid CNN-LSTM	86.5	86.1

With an accuracy of 60% and an F1-score of 59%, the SVM model did a little worse, showing that it has limits on this dataset. Figure 15 shows comparative analysis of models using Dataset 2. The Logistic Regression (LR) model did the worst, with an accuracy score of 52% and an F1-score of 54%.

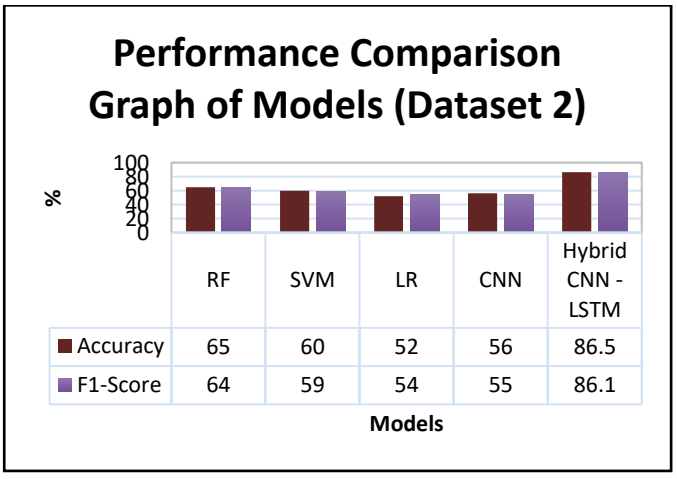

Figure 15: Comparative Analysis of Models (Dataset 2)

This means that it has a harder time classifying the data than the other models. The CNN model did about average, too. It got 56% accuracy and a 55% F1-score, which means it could learn from the data, but it could be even better. With an 86.5% accuracy rate and an 86.1% F1-score, the Hybrid CNN-LSTM model does much better than the others. This shows that it can better identify the data and combine precision and memory, especially in a more complicated dataset.

CONCLUSION

Over the past few years, study and development in using machine learning to find lung nodules have come a long way. Several methods have shown promise in improving the accuracy of diagnostics. The table above shows a summary of the most important studies in this area. It shows the different methods used to find lung tumours on medical images, mostly CT scans and X-rays. Among these are basic machine learning methods like SVM and Random Forest, as well as more complex ones like CNN, Hybrid CNN-LSTM, and deep learning models that use both spatial and time data. Out of all the ways we talked about, Convolutional Neural Networks (CNN) constantly showed high accuracy in finding lung tumours. Some models even reached 96% accuracy. CNN is able to easily pull out hierarchical features from raw picture data, which means that feature engineering doesn't have to be done by hand as much. Additionally, models that combine CNN with SVM or LSTM have been shown to perform better than single models. This is especially true when dealing with noisy datasets and improving the understanding of temporal sequences, which makes the model more useful across a wider range of imaging modalities. Even with these improvements, it is still hard to deal with the limits of datasets, especially the small amount of labelled data and the fact that tumour traits vary from patient to patient. Even though many of the models are showing great results, they still need big datasets with lots of labels to work at their best. Future improvements should also focus on adding explainable AI (XAI) methods so that doctors can trust the model and understand how it makes decisions.

REFERENCES

[1] Siddiqui, E.A.; Chaurasia, V.; Shandilya, M. Detection and classification of lung cancer computed tomography images using a novel improved deep belief network with Gabor filters. Chemom. Intell. Lab. Syst. 2023, 235, 104763.

[2] El Hamdi, D.; Elouedi, I.; Slim, I. Computer-Aided Classification of Cell Lung Cancer Via PET/CT Images Using Convolutional Neural Network. Int. J. Image Graph. 2023, 2450040.

[3] Raza, R.; Zulfiqar, F.; Khan, M.O.; Arif, M.; Alvi, A.; Iftikhar, M.A.; Alam, T. Lung-EffNet: Lung cancer classification using EfficientNet from CT-scan images. Eng. Appl. Artif. Intell. 2023, 126, 106902.

[4] Thakur, S.K.; Singh, D.P.; Choudhary, J. Lung cancer: Detection and classification of malignancies. In Proceedings of the Artificial Intelligence and Sustainable Computing: Proceedings of ICSISCET 2020; Springer: Berlin/Heidelberg, Germany, 2022; pp. 451–463.

[5] Shafi, I.; Din, S.; Khan, A.; Díez, I.D.L.T.; Casanova, R.d.J.P.; Pifarre, K.T.; Ashraf, I. An effective method for lung cancer diagnosis from ct scan using deep learning-based support vector network. Cancers 2022, 14, 5457.

[6] Mohamed, T.I.; Oyelade, O.N.; Ezugwu, A.E. Automatic detection and classification of lung cancer CT scans based on deep learning and ebola optimization search algorithm. PLoS ONE 2023, 18, e0285796.

[7] Xu, Y.; Wang, S.; Sun, X.; Yang, Y.; Fan, J.; Jin, W.; Li, Y.; Su, F.; Zhang, W.; Cui, Q.; et al. Identification of benign and malignant lung nodules in CT images based on ensemble learning method. Interdiscip. Sci. Comput. Life Sci. 2022, 14, 130–140.

[8] AR, B. A deep learning-based lung cancer classification of CT images using augmented convolutional neural networks. ELCVIA Electron. Lett. Comput. Vis. Image Anal. 2022, 21, 130–142.

[9] Bishnoi, V.; Goel, N.; Tayal, A. Automated system-based classification of lung cancer using machine learning. Int. J. Med. Eng. Inform. 2023, 15, 403–415.

[10] Damayanti, N.P.; Ananda, M.N.D.; Nugraha, F.W. Lung cancer classification using convolutional neural network and DenseNet. J. Soft Comput. Explor. 2023, 4, 133–141.

[11] Vaiyapuri, T.; Liyakathunisa; Alaskar, H.; Parvathi, R.; Pattabiraman, V.; Hussain, A. Cat swarm optimization-based computer-aided diagnosis model for lung cancer classification in computed tomography images. Appl. Sci. 2022, 12, 5491.

[12] Yadlapalli, P.; Bhavana, D.; Gunnam, S. Intelligent classification of lung malignancies using deep learning techniques. Int. J. Intell. Comput. Cybern. 2022, 15, 345–362.

[13] Guo, Z.; Yang, J.; Zhao, L.; Yuan, J.; Yu, H. 3D SAACNet with GBM for the classification of benign and malignant lung nodules. Comput. Biol. Med. 2023, 153, 106532.

[14] Wu, R.; Liang, C.; Li, Y.; Shi, X.; Zhang, J.; Huang, H. Self-supervised transfer learning framework driven by visual attention for benign–malignant lung nodule classification on chest CT. Expert Syst. Appl. 2023, 215, 119339.

[15] Qiao, J.; Fan, Y.; Zhang, M.; Fang, K.; Li, D.; Wang, Z. Ensemble framework based on attributes and deep features for benign-malignant classification of lung nodule. Biomed. Signal Process. Control. 2023, 79, 104217.

[16] Bhende, M.; Thakare, A.; Saravanan, V.; Anbazhagan, K.; Patel, H.N.; Kumar, A. Attention layer-based multidimensional feature extraction for diagnosis of lung cancer. BioMedRes. Int. 2022, 2022, 3947434.

[17] He, W.; Li, B.; Liao, R.; Mo, H.; Tian, L. An ISHAP-based interpretation-model-guided classification method for malignant pulmonary nodule. Knowl.-Based Syst. 2022, 237, 107778.

[18] Jiang, H.; Shen, F.; Gao, F.; Han, W. Learning efficient, explainable and discriminative representations for pulmonary nodules classification. Pattern Recognit. 2021, 113, 107825.

[19] Zheng, B.; Yang, D.; Zhu, Y.; Liu, Y.; Hu, J.; Bai, C. 3D gray density coding feature for benign-malignant pulmonary nodule classification on chest CT. Med. Phys. 2021, 48, 7826–7836.

[20] Saied, M.; Raafat, M.; Yehia, S.; Khalil, M.M. Efficient pulmonary nodules classification using radiomics and different artificial intelligence strategies. Insights Imaging 2023, 14, 91.

[21] Park, H.; Yun, J.; Lee, S.M.; Hwang, H.J.; Seo, J.B.; Jung, Y.J.; Hwang, J.; Lee, S.H.; Lee, S.W.; Kim, N. Deep learning-based approach to predict pulmonary function at chest CT. Radiology 2023, 307, e221488.
[22] Riya Sharma; Dr.Pramod Sharma. Machine Learning Based 5G Network Channel Quality Prediction, International Research Journal of Engineering and Technology (IRJET), Volume: 09 Issue: 12 | Dec 2022.