Privacy-Preserving Multiclass Lung Disorder Classification via CNN with Cosine Similarity in Big Data Framework

Jaya Sharma[1], D. Franklin Vinod[*1], Urvashi Chugh[2]
[1]Department of Computer Science and Engineering, Faculty of Engineering and Technology, SRM Institute of Science and Technology, NCR Campus, Delhi-NCR Campus, Delhi-Meerut Road, Modinagar, Ghaziabad, UP, India.
[2]Department of Information Technology, KIET Group of Institutions, Delhi-NCR, Meerut Road (NH-58), Ghaziabad, UP, India.
E-mail: jayashaa07@gmail.com, datafranklin@gmail.com, urvashimutreja1984@gmail.com
[*]Corresponding author

Keywords: big data, lung diseases, computed tomography scans, privacy-preserving, deep learning, image classification

Received: June 3, 2025

Annotating large-scale medical data manually takes a lot of time and human resources, and it requires specific medical knowledge and experience. Big data and Deep learning are two advancing technologies being widely used in the medical field for improved analysis. Because of the recent advancements in imaging technology, computer vision researchers still have unsolved problems related to automatically identifying medical images. Images, however, could include sensitive information about specific body parts and specifics of diseases. In actuality, sharing medical images that contain extremely sensitive information for each user may expose sensitive information to adversaries. One of the main issues between a user and a databank is privacy, we present in this study a Multi-layered convolutional neural network (MLCNN) integrated with PPCS (Privacy preserved cosine similarity) for feature extraction from large-scale medical image data. The framework that uses fully homomorphic encryption (FHE), CSSK(Cheon-Kim-Kim-Song) scheme to search for safe and to enable the categorization CNN is used for large-scale encrypted images. This study aims to diagnose various lung disorders such as COVID-19, lung cancer, pulmonary tuberculosis, pneumonia, and differentiate them from normal conditions by analyzing computed tomography (CT) images. The model's results included 98.54% F1 score, 97.11% Matthew's correlation coefficient (MCC), 98.89% accuracy (AC), 98.38% recall, and 98.81% precision (PC). We compare and contrast our privacy-preserving method with a CNN-based multiclass classification model that offers quick and effective classification.

Povzetek: Članek predstavi zasebnostno varno večrazredno klasifikacijo pljučnih bolezni iz CT posnetkov z MLCNN in PPCS (kosinusna podobnost) na velikih podatkih (Spark) s FHE/CKKS.

1 Introduction

As data is gathered by devices such as mobile phones, inexpensive and widely used information-sensing Internet of things (IOT) devices, aerial (remote sensing) equipment, software logs, cameras, microphones, radio-frequency identification (RFID) readers, and wireless sensor networks, the size and quantity of available data sets have increased dramatically [1],[2]. Since the 1980s, the global technical capacity per person to store information has approximately doubled every 40 months; as of 2012 [3]. According to a forecast by International Data Corporation (IDC), the worldwide data amount is projected to reach 163 zettabytes by 2025 [4]. Thus, the accumulation of data from many sources leads to high velocity, high volume, and variety of data, which in turn gives rise to the phrase big data has gained popularity recently. The "3V" stands for three major characteristics that best characterize big data: Volume (the total amount of data produced), variety (data spanning numerous categories), and velocity (the speed at which data is produced) [5],[6],[7],[8],[9]. There have been two more "Vs" added: Veracity (the caliber of recorded data) and variability (inconsistency in data) [10].

Numerous scientific fields, including medicine [11] and customized treatment based on genetic data [12], social media [13] and the agriculture areas are using big data applications. One of most and prominent field of big data is in Healthcare. The generation of medical images is growing exponentially as a result of the proliferation of digital devices and the development of camera technology. A digitized image is used by modern hospitals these days to forecast the severity of a patient disorder. Image classification has grown significantly more difficult with the abundance of medical images due to the quick development of digital imaging. In order to allocate these medical images to the most relevant class based on similarity, classification methods must be used. A medical image classification domain contains images of various body organs, including CT scans, X-rays (electromagnetic waves), and positron emission tomography (PET) scans, Magnetic resonance imaging (MRI), and many more

which are a type of imaging test that shows how your body's organs and tissues are functioning. Image classification is now playing a key part with the big data in the medical images because of the development of digital images. Handling with Big data is also a challenging task. In this paper we utilized Lung CT images to diagnose lung disease without disclosing the sensitive information of the patients.

In a variety of machine learning problems [14], deep learning (DL) algorithms have produced exceptional results in recent years. CNN have been shown to be particularly effective at image classification tasks. They have been successfully used in a variety of field's, including the prediction of galaxy morphology [15], the creation of image-guided autonomous vehicles [16], face detection [17],[18], large-scale video classification [19], and many other applications [20],[21],[22]. In the area of pattern identification, CNNs have excelled in particular. In our work, we modify CNN model to classify diseases such as Covid-19, Lung cancer, pulmonary Tuberculosis and Pneumonia.

There may be numerous features in the images of a big data dataset which could have an impact on or worsen our classification model's performance if we include all features. Features can also be related to one another; these features are redundant and inconsequential, which could negatively impact computation time. Feature selecion approaches are presented as a solution to this noise issue, with the potential to eliminate irrelevant characteristics without compromising other pertinent information [23]. The ability to avoid the curse of dimensionality, shorten the training model's runtime, improve data compatibility with the learning model, and enable smoother model interpretation is among the main duties of feature selection [24].

The feature selection procedure can be done by user input interaction with a databank to categorize user disorders; it is most prominent area where sensitive information can be shared or leaked. We may be providing opponents with sensitive information since we are dealing or working with medical images, which are extremely sensitive for each individual. As an illustration, when a user engages with a model by requesting matching features from a databank, the model may learn unique insider information about the person and vice versa. As a result, security and privacy are becoming increasingly important aspects of big data. In this paper, we developed a privacy-preserving system that recognizes similar traits without revealing user's personal information. We accomplish these goals by combining the MLCNN model with privacy preserved cosine similarity.

The paper is divided into the following sections: The literature review is briefly explained in Section 2. Section 3 explains the problem statement. The proposed methodology is presented in Section 4. The experimental findings and results analysis of the proposed investigation are shown in Section 5. Discussion section is presented in Section 6. The conclusion and the scope for the future is presented in Section 7.

2 Literature review

Guo C, et al. [25], the suggested method's security was examined to make sure that no private data from the encrypted images was disclosed. They assess and illustrate the usefulness of the privacy-preserving approach for image searching on four real-world datasets (i.e., chest X-ray images, retinal OCT images, blood cell images, and the Caltech101 image set) as well as CNN-based classification and original image searching. This protocol makes it possible to search for images that are sent to the cloud quickly and accurately without sacrificing the privacy of encrypted data. According to experimental results, PPIS outperforms prior systems in terms of searching time (more than six times faster) and achieves an accuracy rate of over 86% on real-world datasets with the same CNN structure in the plaintext domain.

Qi et al. [26] suggested a ConvMixer model with an adaptive permutation matrix and block-wise encrypted images as a means of classifying images while preserving data privacy in cloud environments. Traditional block-wise scrambled encryption methods usually involve the use of both a classifier and an adaptive network to reduce the impact of image encryption. However, the authors point out that using large-size image with conventional methods that use an adaptation network is difficult because of the significant rise in processing costs. In order to verify that the suggested approach uses fewer computing resources, they also assess the computation cost of cutting-edge privacy-preserving DNNs. The authors conducted an experiment to assess the robustness against different cipher text-only assaults and the classification performance of the suggested method on ImageNet and CIFAR-10 in comparison to other methods. The suggested approach was shown to perform better than traditional approaches in an experiment with regard to classification computation cost, accuracy, and resilience to attack techniques.

Jia et al. [27] presented a novel method for identifying image characteristics that integrates multilevel homomorphic encryption and image data partitioning. This new partitioning technique reduces computational complexity, minimizes processing load, and improves classification accuracy. To improve data security and privacy, the authors proposed a new encryption method for partitioned images that is entirely homomorphic. To address the intrinsic complexity of encryption, they devised a compound encryption technique that harnesses the full potential of homomorphic computation, with the stated goal of reducing computational and storage overheads. The technology offered significant advantages over traditional methods, including increased computational efficiency, lower storage and transmission costs, and

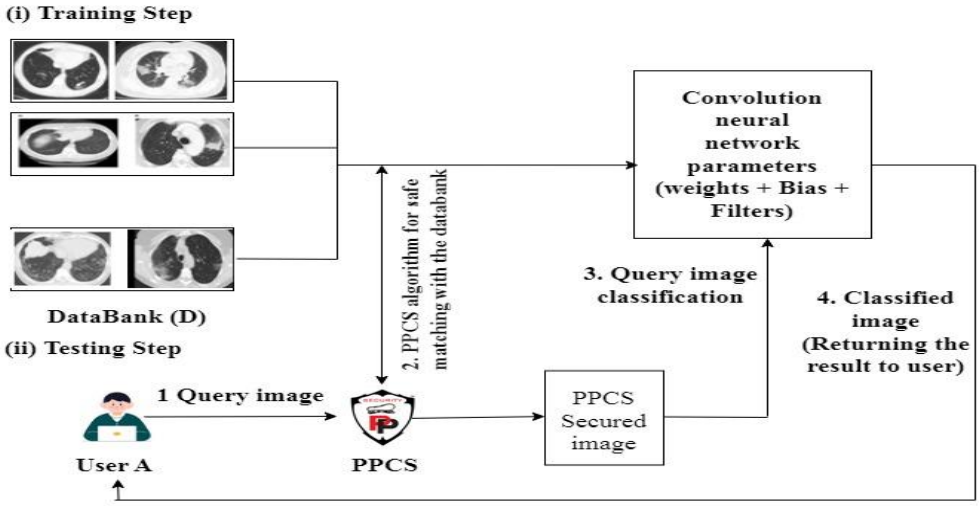

Figure1: Conceptual scheme of proposed work

enhanced security and privacy. This paper's methodology effectively addresses the complex issues of image feature categorization in cloud computing and big data environments.

To determine the most salient features defining a distributed privacy-sensitive dataset, Alishahi et al. [28] developed a unique framework called LDP-FS that combines Local Differential Privacy (LDP) with Feature Selection (FS).The authors presume the features are categorical and independent, and evaluate LDP-FS's performance utilizing two distance measurements: Kendall Tau distance and RMSE, which demonstrate LDP-FS's effectiveness in predicting feature importance scores and ordering features. The experimental findings show how effective and helpful LDP-FS is at detecting irrelevant features in protected data.

Sadoon et al. [29] the author presents the application of cutting-edge techniques, including artificial neural networks, transfer learning, and stacked ensemble models, to create a model with previously unheard-of accuracy, precision, recall, and F1-score. The most dependable classification tool and the best performance are obtained when DenseNet, Xception, and Inception are combined. Additionally, they employ transfer learning to optimize generalization and rapidly train the model, which enables it to detect a variety of pulmonary diseases, including COVID-19. The given research suggested the effective fusion of state-of-the-art deep learning techniques and image processing algorithms with the particularities of the medical imaging sector.

Nemer Z N. et al. [30] given the numerous advantages that sea coral and its classes offer in many facets of our lives, they have developed a system in this study to categorize the sea coral images. Studying four CNN architecture models—AlexNet, SqueezeNet, GoogleLeNet/Inception-v1, and Google Inception-v3—is crucial to this work because it allows researchers to assess each architecture's accuracy and efficiency and identify which one performs best on coral image data. The specifics are provided in the research paragraphs. AlexNet, SqueezeNet, and GoogLeNet demonstrated accuracy of 83.33%, 80.85%, and 93.17%, respectively, according to the results.

Depend on the model architecture and classification difficulty, there are notable differences in accuracy when comparing deep learning models using with secure data in medical image classification. Table 1 illustrates that secure image classification presents difficulties in sustaining high accuracy, when dealing with big data computational resources.

Through the integration of a PPCS and MLCNN, our suggested architecture successfully tackles the pressing issue of data breaches in medical AI classification pipelines. The MLCNN component of the framework uses deeper multi-layer training to enable the network to learn complex hierarchical features that are essential for accurate classification of lung conditions. The architecture is designed to ensure both high diagnostic accuracy and data privacy, which are crucial when handling sensitive medical imaging data like CT scans. PPCS, on the other hand, is essential to preserving data privacy during the search and categorization process. Direct image search and similarity comparison on these encrypted representations are made possible by the PPCS protocol. This is achieved by using a privacy-preserving regularization function that guarantees no identifiable information is leaked in the intermediate or final findings, even during image retrieval or classification. The combination of PPCS and MLCNN allows for a reliable, privacy-conscious medical image analysis system that combines CNNs' deep feature learning capabilities with the advantages of secure computation. The end product is a pipeline that rigorously complies with data privacy regulations while supporting high-accuracy forecasts.

Table 1: Comparisons for related works

Study	Method	Dataset	Privacy	Performance
[25]	Privacy-preserving, image-searching protocol + CNN	Chest X-Ray Images + Blood Cell Images + Caltech101 image +Retinal OCT Image	Secure Multiplication (SMP), Secure Squared Euclidean Distance (SSED), Secure Convolution (SCV), Secure Max Pooling (SMPL)	Precision: Chest X-Ray-89.5%, Retinal OCT - 89.6%, Blood Cell- 86.3%, Accuracy-86%
[26]	Privacy-preserving + ConvMixer	CIFAR-10 + ImageNet	Privacy-preserving wise encryption method	Accuracy CIFAR-10-92.65%, ImageNet - 76.94%
[27]	Homomorphic encryption + CNN	NEU-CLS data set	Multilevel homomorphic encryption	Accuracy - 97.1%
[28]	Differential Privacy + Machine Learning	UCI Repository	Local Differential Privacy Feature Selection	Census- Optimal Local Hashing - 94.1%, Optimized Unary Encoding- 94.1%, Thresholding with Histogram Encoding-94.1%, Mushroom OLH-90.2%, OUE-90.2%, THE-90.2% Obesity OLH-91.2%, OUE-91.2%, THE-91.2%, KDD99 OLH-99.8%, OUE-99.8%,THE -99.8%
[29]	Combination of DenseNet, Xception and Inception	COVIDx, CXR-4 Datasets	No	Precision- 98%, Recall- 98%, F1-score- 98%, Accuracy- 98%
[30]	AlexNet, SqueezeNet,GoogLeNet/ Inception-v1, google Inception-v3	10 classes of sea coral images	No	Accuracy AlexNet -83.34%, SqueezeNet - 80.86%, GoogLeNet - 90.6%, Inception-v3 - 93.18%

3 Problem statement

We investigate a viable solution to the problem outlined in this study. To maintain anonymity, two bodies, A (user) and B (model), want to work together to execute a common image similarity detection algorithm using their private inputs. A represents the user who possesses a private image I. B denotes the model that classifies diseases related to the lungs. D refers to the databank that provides a collection of images. Let D = (I1, I2, I3, I4…….. In) refers to represent the collection of n images that D has. It is possible that additional information for both sides may be sacrificed if a user wishes to transmit a private image I as input to the pre-trained model in order to determine patient disease. To diagnose the disease, we must first determine the cosine similarity between the feature vectors in the database together with the query image I in a secure way before feeding the data straight into the model. Our goal is to find the image in D that most closely resembles Ụ while protecting the patient's and model's privacy.

Our approach, which uses PPCS, is more adaptable to such circumstances. PPCS can assist in safely extracting comparable features from the database. The PPCS method is the first thing the model will utilize if the user wants to enter a query image (CT image) to identify lung-related conditions. By employing cosine similarity with every databank image, PPCS safely locates features and returns a response with a privacy-preserved image. After the classification of the privacy-preserved image by our MLCNN model, the user receives the results (Figure 1).

4 Proposed scheme

Preliminaries

4.1 Examination of the dataset for chest CT

In this study, the chest CT scans have been used to detect Covid-19 inside our dataset. Information was taken from the Moscow Health Care Department's Research and Practical Clinical Center for Diagnostics and Telemedicine Technologies, which provides the labeled and open-source MosMed database. It comprises 1110 unique CT chest images. Comparatively, 684 scans

demonstrate stage#1 COVID-19-related pathology, 125 shows stage#2, 45 shows stage#3, and 2 shows stage#4 disease patterns, which identify different stages of disease progression. The Dataset also includes some more data categorized by nation. The Ning W [31] dataset, which came from two respectable hospitals in China (Liyun Hospital and Union Hospital), was also used. The images in this dataset were categorized into three groups: non-informative images (5705), Covid –ve cases (9979), and Covid +ve cases (4001). Thus, a total of 19,685 CT scans with labels are gathered. The Russia [32] dataset and Brazil [33] dataset is also included in our dataset. The Brazil dataset, which has 2482 CT images from 120 individuals. The data was divided into two groups: 1252 CT images for covid +ve cases and 1230 images for covid -ve cases. A publicly available dataset of Iran has been used that contains 8,439 scans images, which are categorized by 7,495 scans for positive cases and 944 scans for negative.

In our experiment, we used the LIDC-IDRI [34] dataset, which is open to the public. Prior studies on lung nodule identification solely took into account the nodule's diameter, which had to be at least 3 mm in order to ensure consistency for the assessment and subsequent comparison with the suggested structure. The size of the nodule and its edge, which is equally crucial for diagnosing lung cancer, are two significant metrics that we have examined in our work. In essence, a spherical nodule has a lower risk of cancer than a circular one (which has sharp edges).

Some helpful datasets, like the Radiological Society of North America (RSNA), and Radiopaedia, have medical and public datasets that are thought to be the most common. These collections of common pneumonia CT images, which total roughly 4846 images [35], [36], can be used to train our proposed deep MLCNN to differentiate COVID-19 from pneumonia.

There are pulmonary tuberculosis samples available for study and advancement. These include the multi-category segmentation dataset DeepPulmoTB, as well as datasets from the National Institutes of Health (NIH), Hugging Face etc [37]. These datasets provided useful tools to examine and comprehend pulmonary tuberculosis using CT images. This study's final dataset, which includes 2500 CT images, is of normal images.

4.2 Data storing phase

Due to the rapid increase in lung diseases, including, COVID-19, Lung cancer, pulmonary Tuberculosis and Pneumonia, it is essential to analyze diseases using a multi-classification approach. In order to accomplish this, we are working with extensive datasets from multiple sources, which are challenging due to their varying conventional shapes, sizes, and formats.It is critical to handle this massive amount of data in accordance to process and storing. Apache Spark, one of the most popular frameworks, efficiently stores large amounts of data and distributes the processing of the data, either inside the same system or in combination with other distributed computing technologies.

In our approach, we build end-to-end deep learning pipelines that operate on the Spark platform are built using the Python packages PySpark, TensorFlow, and Elephas. We utilized PySpark to access the Python API in Spark [38], which expands Apache Spark's capabilities and allows for Python operations.

The Apache Spark architecture is seen in Figure 2 and includes a driver programme that aids in running the system's primary purpose and creating Spark Context objects. To operate the processes on a cluster, Spark Context connects with many kinds of cluster managers.

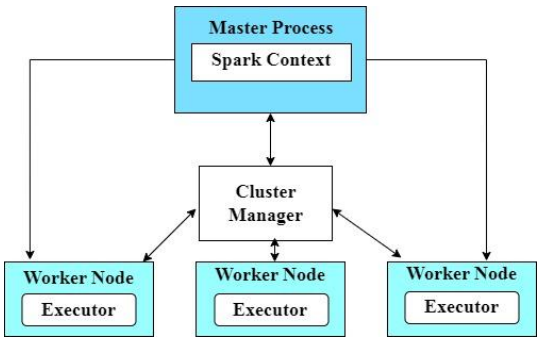

Figure 2: Overview of the apache spark cluster

It obtains the executor from the cluster nodes and then passes our system jobs to the executor so that it can execute the job. The cluster manager is responsible for allocating the resource across the systems. Cluster managers are available in various types, including Apache Mesos, Hadoop YARN, and standalone schedulers. The application running on the cluster is supported by the worker or slave nodes.

To complete our task, we must first decode the images into a Spark dataframe. To do this, SparkSession is set up to access the fundamental Spark features for DataFrames with dispersed master clusters. We loaded all images from folders into the DataFrame using the Apache Spark SQL data source API, utilizing a single column named "image". Image augmentation has been used to carry out operations like rotation, skewing, and reversal. After the image size of 224 x 224 x 3 has been achieved, each block of the image can be distributed to RDD in order to create a faster and more efficient map and minimize operations.

4.3 Training the model

4.3.1 Feature extraction

The feature extration procedure reduces dimensionality during the predictive model construction process by removing superfluous and irrelevant features from the input image. There are algorithms in deep learning that

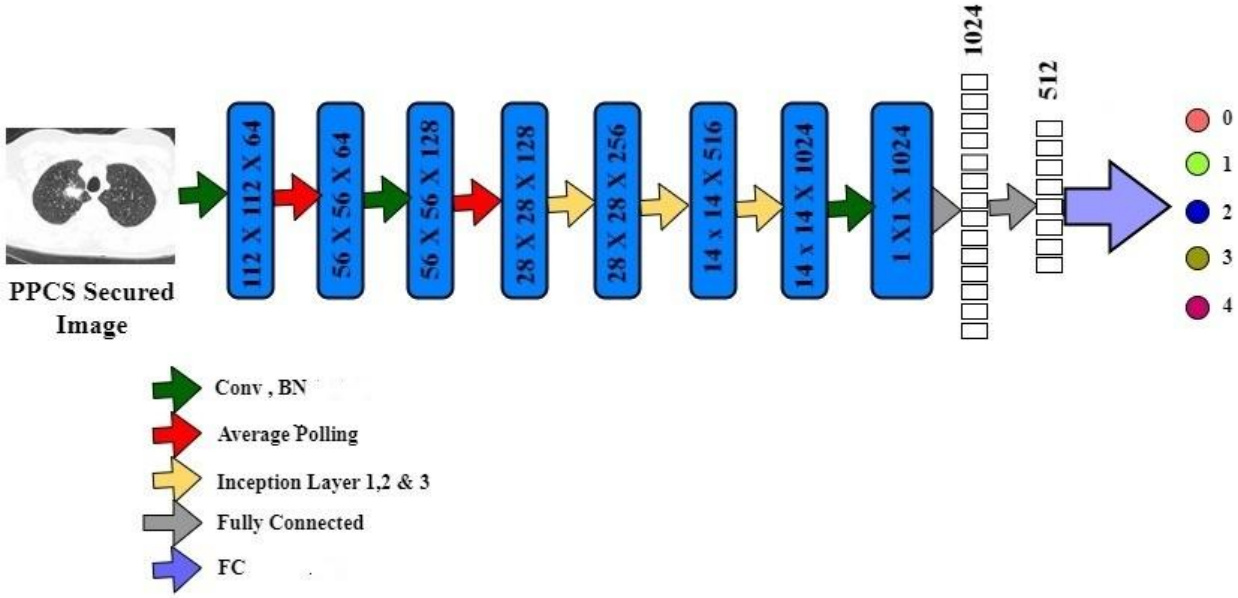

Figure 3: MLCNN Architecture to classify Lung images

choose features automatically as part of the model-learning process. Convolutional neural networks, or CNNs, have been widely used in medical image processing and have produced significant benefits for public health in the field of medicine [39].

There are three primary layers in the proposed MLCNN. Extracting hierarchical characteristics from the encrypted input image is the goal of these layers.

The first layer is convolutional layer receives data from the input layer and processes it. The convolutional layer, the central layer of the neural network, has the ability to improve feature information and eliminate irrelevant information. The output of neurons connected to the input is calculated by each convolutional layer using a dot product of the weights of the neurons and a small portion of the input to which it is connected. Equation (1) provides the output of the neurons at the first convolutional layer, where x is an input feature map and w is the weights .

$$\gamma^k = f\left(\sum_{k=0}^{k} x^k .\ w^k + b^k\right) \tag{1}$$

Where dot denotes the 2D convolution operation between input feature map and weight w. γ^k is the convolutional layer's output feature map for the k^{th} input, with b serving as the bias term. With 64 kernels, the convolutional layer-1 generates output in a volume of 56 x 56 x 64.

While CNNs are built for encryption image contexts, polynomial activation functions are crucial because they allow for secure computation by substituting operations that can be carried out on encrypted data (such as additions and multiplications) for conventional non-linearities. This ensures interoperability with privacy-preserving frameworks while maintaining the model's capacity to learn intricate patterns. By using simply addition and scalar multiplication, it offers effective spatial downsampling that is compatible with homomorphic encryption. It also produces smoother and more stable feature aggregation without the need for non-linear operations like max pooling.

The Conv and Poly activation functions are separated by batch normalization (BN), which expedites training and reduces network initialization sensitivity [40]. The internal covariate shift is intended to be eliminated by BN. This is accomplished by normalizing the batch-wise mean (μ) and standard deviation (σ).

The mean and variance (σ^2) are computed using the following formulas for batch normalization computation.

$$Mean(\mu) = \frac{1}{n}\sum_i^n x_i \tag{2}$$

$$Variance\ (\sigma^2) = \frac{1}{n}\ x \sum_i^n (x_i\ - \mu_i) \tag{3}$$

Here n is the mini-batch size of the x_i input feature element, while μ and σ are the mini-batch mean and standard deviation, respectively. In our case, n is set of 64

$$\gamma_i = \frac{(x_i - \mu_i)}{\sqrt{\sigma 2 + \varepsilon}}\gamma + \beta \tag{4}$$

Where the initial values of each output's learnable parameters are γ and β.

Lastly, the pooling layer divides the neurons of the previous layer into a set of non-overlapping rectangles and utilizes a down sampling technique to extract the value of one neuron from each sub-area in current layer. In average pooling the output is selected as the average value within each sub-region. With stride 2, a 2 x2 avg-pooling is used for down-sample the feature map produced by the previous sub-layer in our work. Each successive convolutional layer's feature map is immediately pooled

by computing its maximum polynomial activation output, as indicated in Equation (5).

$$\gamma_{PooL} = AVG_{PooL}\left(Poly\left(BL\left(Conv(x, \text{ẇ})\right)\right)\right) \quad (5)$$

Table 2 illustrates that the input provided to the proposed MLCNN model is of size 224 × 224 × 1. In the first stage, the model employs a convolutional layer (Conv_1) with 64 filters of size 7×7 to extract initial features. In the subsequent stage, a second convolutional layer (Conv_2) is applied, using 128 filters of size 3×3 to capture more detailed and higher-level features essential for accurate disease identification. Since we are dealing with homomorphically encrypted images, it is crucial to make sure that every CNN operation is compatible with the restrictions of homomorphic encryption techniques. The ReLU activation function and max pooling are examples of standard non-linear processes that depend on conditional logic and comparisons, which homomorphic encryption does not natively enable. Polynomial activation functions, which can be easily calculated using addition and multiplication, were used in place of ReLU in order to overcome this. Likewise, we replaced max pooling with average pooling since it works with addition and scalar multiplication, which are fully supported by homomorphic encryption. After each Convolution layers (Conv_1 and Conv_2), polynomial functions and average pooling layers (AP1, AP2, AP3 and AP4) have been applied. With these modifications, MLCNN inference is safe and private without disclosing computational feasibility. After that Inception layers (I1, I2, and I3) are employed which is the combination of Convolution layers followed by polynomial function and average pooling. In order to reduce dimensions utilizing stacked 1x1 convolutions and enable more efficient computation in the deeper networks, convolutional neural networks use inception modules. These modules are designed to address a number of issues such as computational cost and overfitting. The final two fully connected layers (FC1 and FC2) are added to classify the diseases . To prevent overfitting, a dropout layer is added after the last pooling layer for regularization. Figure 3 shows the architecture of the proposed model.

4.3.2 Classification

The feature map obtained from the Inception layer is flattened into a one-dimensional array, and a dense fully connected layer with the poly activation function is then added in order to categorize the diseases into five groups (COVID-19, Lung Cancer, pulmonary Tuberculosis, Pneumonia, or Normal). To anticipate the result, the dense layer uses the features that were recovered from the processed images; the poly activation is applied as the final layer.

Algorithm 1: Fast identification and classify large size images

Input: Lung CT image (including all diseases images) Training_Set δ1 and Testing_Set δ2 Learning rate – μ, iteration step – ξ;

Step1- import required libaries
import pyspark // Setup a pyspark session
import tensorflow as tf
import elephas // DL receives tensorflow. keras to spark from the deep learning pipeline.
from tensorflow.keras.models import Sequential as Seq
from tensorflow.keras.layers import Conv2d,Flatten, Dense_d,AvgPooling2D, Dropout_Lr
from tensorflow.keras layers import Dense, Dropout, Conv2d, AvgPool2d, Fattern, Activation, Batch Normalization.

Step2- Create a Spark dataframe with all of the images loaded, and then divides it into training and testing sets. // the capacity to study data in multiple analytical ways is provided by Spark Data Frame.

Step3- // to categorize lung_images

L#3.1: (Conv2d) layer_Conv1= Conv2d (filters_f= 64,(filter_size_L1 = 7,7);input_shape=(224,224,3) stride_s=2; padding_p=same; activation_act=’poly’) (ɣ)

L#3.2: (Conv2d) layer_Conv2= Conv2d (filters_f= 128,(filter_size_L2 = 3,3); stride_s=1; padding_p=same; Activation_act=’poly’) Avg_Pooling2d(2,2)

L#3.3,3.4,3.5: def incp_mod(a, F_1x1, F_3x3,F_avgpool, name= None):
Output = concatenate ([Conv2D_1x1, Conv2D_3x3,AvgPool_1, Droupout]
Dense layer // auxiliary _outputs

Step4– Fit and analyze the model

Table 2: A tabular representation of the proposed model that includes the kernals, output shape, and learnable parameters used

Layer (type)	**Kernel size**	**Output shape**	**Learnable Parameters**
Conv_1	7x7, s=2 f#64	112, 112 , 64	9.4k
AP1(AVG P_1)	2 x2/2	56, 56, 64	0
Conv_2	3 x 3,s=1f#128	56,56, 128	75 k
AP2(AVG P_2)	2 x2/2	28,28, 128	0
InceptionM(I_1)	1 x1conv, 3x3 conv, AVG_Pool	28,28, 256	300k
AP3(AVG P_3)	2 x2 /2	14, 14, 256	0
InceptionM(I_2)	1 x1conv, 3x3 conv, AVG_Pool	14,14, 512	76.6k
InceptionM (I_3)	1 x1conv, 3x3conv, AVG_Pool	14,14, 1024	160k
AP4(AVG P_4)	2 x2/2	7 ,7, 1024	0
FC1	Linear	1,1, 1024	5M
FC2	Linear	(None, 5)	5k
Total parameters: 51,763,845			
Trainable parameters: 51,763,743			
Non trainable paramerters: 102			

4.4 Testing the model

4.4.1 Querying image

The suggested model's basic architecture is depicted in Figure 1. The user provides a CT scan as input to diagnose illnesses during the testing phase. The security of the user, model, and databank must be considered. Using the PPCS method, we first safely extract lung features from the databank to confirm whether the image belongs to the lungs and to detect harmful or noisy data. A compression level of M=0.6N is then applied to the image, where M represents the compressed measurements and N represents the original dimension. In order to protect sensitive user and databank information, PPCS makes sure that an encrypted and protected image is shared with the MLCNN model.

4.4.2 The functionality of PPCS

By comparing the input image to every other image in the database, the PPCS method determines similarities without jeopardizing model security or user personal information. Based on certain fully homomorphic encryption [41], privacy-preserving aggregate is a widely used method of gathering data for event statistics. If a public key encryption method E(*) is homomorphic, then many sources can encrypt their own data (m1, m2, m3…….. mn) using the same public key to create ciphertexts (ct1 = E(m1), ct2 = E(m2), …, ctn = E(mn). For instance, these ciphertexts can be accumulated as follows: $C = \prod_{i=1}^{n} ct_n = \sum_{i=1}^{n} m_i$ using the sum aggregation. It is possible to recover the accumulated result $\sum_{i=1}^{n} m_i$ from ciphertexts using the matching private key. To ascertain commonalities, we compute the dot (*) product of two images represented as vectors and divide it by the magnitudes of each feature vector map. All of the image properties in the database are compared to the input feature's cosine values. The cosine similarity range is between -1 and 1. Two vectors with the same orientation will have a high cosine similarity when their angle is 0.

PPCS Algorithm 2

The steps involved in the cosine similarity computation process are as follows:

P_A = Image of Patient A

And feature vector $a = (a_1, a_2, \ldots\ldots\ldots a_m)$

D_B = Database Image B

And feature vector $b = (b_1, b_2, \ldots\ldots\ldots b_m)$

Step1: Generation of keys (α, β):

i) Determine that by choosing two different prime numbers, α and β.

ii) $\eta = \alpha * \beta$ and Security parameter

$\lambda = \text{lcm}(\alpha\text{-}1, \beta\text{-}1)$, where the least common multiple is shown by the lcm operation. Assume that the private key is λ.

iii) Select an integer at random $g \varepsilon Z^2$.

iv) Make sure the following modular multiplicative inverse exists in order to verify that η divides g's order.

$\mu = L(g^{\lambda} \bmod (\eta^2))^{-1} \bmod \eta$

The function can then be defined = $\frac{\mu-1}{\eta}\lambda$;

The public key pk will be (g, η); transmit it to UA for additional calculation.

Step2: Encryption (a, pk) : Computation on UA

i) Choose an integer at random $r \in \dot{Z} * \eta^2$ (A number between 1 and η^2) for everya_i, i =1,2,...,m

ii) Calculate $C_i = g^{a_i} r^{\eta} \bmod \eta^2$

where $a_i = (a_1, a_2, \ldots\ldots\ldots a_m)$

iii) Transmit (g, η, C_i) toD_B

Examine $A = \sum_{i=1}^{m} a_i^2$

Step3:D_B Calculation (calculated for each input database image)

For each bi , i= 1,2,.. m $D_i = g^{a_i} r^{\eta} \bmod \eta^2$

$B = \sum_{i=1}^{m} b_i^2$ and $D = \sum_{i=1}^{m} D_i$

Transmit (B, D) to UA

Step4: Calculation by UA:

Find out $E = r^{-\eta}.D \bmod \eta$

$\vec{a}.b = \sum_{i=1}^{m} a_i b_i$

$$= \frac{E - (E \bmod r^2)}{r^2}$$

$$\cos(\vec{a}.\vec{b}) = \cos \frac{\vec{a} * \vec{b}}{\sqrt{A}\sqrt{B}}$$

Now, without revealing the model or user data, the subset is produced. Consequently, the private information of the model and its users is safe, and there is no chance of a security breach from either side.

We can efficiently compute the cosine similarity cos (a,b), given $a = (a_1, a_2, \ldots\ldots\ldots a_m)$ and $b = (b_1, b_2, \ldots\ldots\ldots b_m)$. However, Inter-big data processing would make each other's privacy visible through the direct cosine similarity evaluation. We used FHE, CSSK scheme, to execute the privacy-preserving cosine similarity computation by first computing $\vec{a}.b$ and then $\cos(\vec{a}.\vec{b})$ using the values of $\vec{a}.\vec{b} = \sum_{i=1}^{m} a_i b_i$. Since each a_i i = 1, 2,…, m is randomly masked with $C_i = g^{a_i} r^{\eta} \bmod \eta^2$, Algorithm guarantees that each $a_i$is privacy-preserving for the purposes of PPCS. It should be noted that the purpose of adding $a_{n+1} = a_{n+2} = b_{n+1} = b_{n+2} = 0$ is to guarantee that D contain a arbitrary integer, ŕ. The vector $b = (b_1, b_2, \ldots\ldots\ldots b_m)$can be prevented from being guessed by vector a.

Table 3: Findings of lung CT image classification in various rounds

		MLCNN					MLCNN with PPCS				
		CP	**LC**	**PT**	**PN**	**H**	**CP**	**LC**	**PT**	**PN**	**H**
20 rounds	CP	**2847**	809	547	691	96	**4747**	689	215	896	67
	LC	353	**3785**	479	275	450	367	**3993**	354	537	518
	PT	722	200	**3265**	198	283	868	419	**4215**	749	317
	PN	443	176	325	**3673**	205	653	299	134	**2478**	191
	H	506	516	274	163	**3966**	765	475	82	340	**3910**
50 rounds	CP	**4926**	410	769	78	164	**5726**	595	67	68	279
	LC	902	**5888**	379	25	93	925	**5925**	83	98	92
	PT	707	1325	**7515**	46	177	1360	2136	**10335**	111	240
	PN	821	1008	176	**8834**	145	1356	1056	121	**9723**	45
	H	675	259	413	42	**9566**	843	288	223	123	**9389**
100 rounds	CP	**15098**	565	136	67	337	**14565**	1065	153	131	370
	LC	1387	**17271**	267	133	154	1986	**17226**	241	43	162
	PT	1455	478	**21665**	243	870	1723	298	**22465**	126	549
	PN	1381	616	312	**19800**	132	947	486	183	**19775**	231
	H	1589	970	139	121	**21331**	1889	925	142	180	**21186**

4.4.3 Classification for querying image

As illustrated in Figure 3, we provide the private, encrypted analysis of the parameters to calculate the querying image's feature vector in order to build a forward propagation structure. Thus, the tensor variable v indicates the feature mappings of each layer.

5 Results and discussion

5.1 Experimental parameters

The Tensorflow 2.10.1 framework and Python 3 were used to create the models. Apache Spark 2.4 with a P100 GPU processor, 2 TB of storage, and 32 GB of RAM was used to run these. Using an API Keras Augmentor, the images of the input classes were enhanced to increase the amount of images in each class in order to achieve the statistical findings. To carry out pre-processing tasks like augmentation, scaling, and normalization, all original and augmented images were sent to Keras Image Data Generator class [42].

5.2 Performance metrics

For all training images, we determine effectiveness and accuracy of the model with the help of the confusion matrix (COM). It has 4 variables: True positive ($True_P$), False Positive ($False_P$), False Negative ($False_N$), and True Negative ($True_N$). To measure or evaluate the effectiveness of a model, the rows in the matrix represent the original class values, while the columns represent the estimated class values. Furthermore, according to Equation (6), Accuracy (AC) is calculated using a confusion matrix that takes into account the proportion of correctly evaluated specimens to all specimens. Equation (7) defines precision (PC) as the positive predicted value, and recall or sensitivity, as used to classify as positive corrected specimens that yielded a positive result shown in Equation (8). While specificity (SF), as shown in

Equation (9), refers to the true negative rate. In addition, the Matthew's Correlation Coefficient (MCC) is also mentioned in Equation (10), and F1-score is specified in Equation (11).

$$Accuracy\ (AC) = \frac{True_P + True_N}{True_P + True_N + False_P + False_N} \quad (6)$$

$$Precision(PC) = \frac{True_P}{True_P + False_P} \quad (7)$$

$$Recall\ (TPR\ or\ Sensitivity) = \frac{True_P}{True_P + True_N} \quad (8)$$

$$Specificity(SP) = \frac{True_N}{True_N + False_P} \quad (9)$$

F1-score illustrates how Recall and Precision are used to generate the F1 score. Compared to other scores, such as MCC, the F1 score provides a more balanced perspective; nonetheless, it may produce a biased result by ignoring the number of True Negatives. As indicated by Equation (9), the Matthew's correlation coefficient [43] takes into account each and every cell in the confusion matrix.

$$MCC = \frac{(\boldsymbol{True_P} * \boldsymbol{True_N}) - (\boldsymbol{False_P} * \boldsymbol{False_N})}{\sqrt{(\boldsymbol{True_P} + \boldsymbol{False_P})(\boldsymbol{True_P} + \boldsymbol{False_N})(\boldsymbol{True_N} + \boldsymbol{False_P})\ \boldsymbol{X}\ (\boldsymbol{True_N} + \boldsymbol{False_N})}} \quad (10)$$

$$F1 - score = \frac{2 * PC * TRP}{PC + TRP} \quad (11)$$

MCC and F1-score have been performed on the experiments result which evaluates the system model's performance. Initially, the suggested deep learning model was trained to recognize CT images based on five categories: COVID-19, lung cancer, pulmonary Tuberculosis, Pneumonia, and normal. The CT images were then tested using the deep learning model. The suggested deep learning model's potential is assessed using a technique for 5-fold cross-validation for the classification problem.

For every fold, multi-class classification and average prototype classification are assessed. The suggested

framework classified COVID-19, lung cancer, pulmonary Tuberculosis, Pneumonia, and normal categories which provide excellent level of classification accuracy. The input CT images with dimensions 224 x 224 x 3 were split into 80% training and 20% testing data.To reduce overfitting, the MLCNN implementation includes dropout layers with a rate of 0.5 following each dense layer. Early stopping was utilized to train the model based on validation loss for up to 300 epochs. The Adam [44] optimizer was employed to maintain a parameter learning rate of 0.05 utilizing the AdaGrad. It improves our model's performance.

Figure 4 represents the average accuracy and loss over the number of epochs during the training and validation stages for the suggested models. We can see how model performance increases throughout training for various communication rounds configurations (20, 50, and 100) in the accuracy graph, the X-axis shows the total effective epochs, which are determined by multiplying the number of rounds by the number of local epochs per round. The validation accuracy percentage is displayed on the Y-axis. As learning stabilizes, the accuracy increases in a smooth, logistic-like fashion before plateauing close to 98.89%, suggesting that the model is doing well. The 20-rounds setup may run the risk of overfitting or instability, but it tends to converge more quickly due to its greater local epochs each round. The rise in the 100-rounds design is more gradual, indicating smaller local updates but more frequent communication. The Loss graph tracks the model's loss (such as cross-entropy) as training goes on, the X-axis shows the total number of effective epochs, and the Y-axis shows the validation loss value, which normally ranges from 1.0 to 0.04. As the model gains insight

Table 4: Confusion matrix for the proposed model

	Normal	Lung Cancer	Covid-19	Pneu-monia	Tuber-culois
Normal	0.9712	0.0561	0.0083	0.0005	0.0011
Lung Cancer	0.0267	0.9712	0.0925	0.0003	0.0001
Covid-19	0.0111	0.0036	0.9790	0.0824	0.0130
Pneu-monia	0.0110	0.0271	0.0212	0.9812	0.1210
Tuber-culosis	0.0121	0.0129	0.0218	0.0250	0.9822

from the data, each curve gradually drops from its starting greater loss value (about 1.0), which indicates bad initial predictions. Because to data noise and stochastic optimization, the loss exhibits an exponential decline pattern with slight oscillations. A loss value near 0.04, which denotes a well-trained model, is the goal of all setups. While the smooth drop in many rounds represents slower but steady development, the steeper curves in fewer rounds suggest faster local convergence.

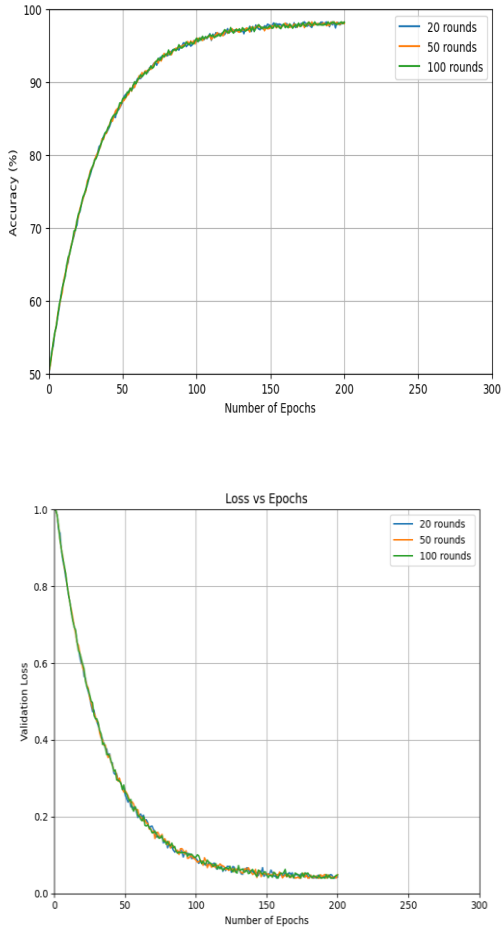

Figure 4: Average accuracy and loss, over the number of epochs in the training and validation stages

The accuracy (AC), loss, recall (Sensitivity), positive predictive value (PPV), also referred to as precision (PC), specificity (SF), negative predictive value (NPV), F1-score, Matthew's correlation coefficient (MCC), and area under the curve (AUC) were among the metrics used to assess the effectiveness of the lung disease classification models.

Before sending the query image to the model, this method offers a better technique to remove irrelevant (private) features. Consequently, categorization performance is one metric that can be used to assess our approach. Here, classification and feature similarity searching are done using the PPCS technique, with MLCNN and without PPCS serves as the baseline for classification. The categorization results with MLCNN with and without PPCS for the medical datasets are shown in Table 3, Covid, lung cancer, pulmonary Tuberculosis, Pneumonia, and normal patients are denoted by the terms CP, LC, PT, PN and H respectively

Table 5:Measures of evaluation for various models

Model & Authors	**Dataset**	**Accuracy**	**Recall**	**Precision**	**F1-Score**
Salih Ahmed et al. [45]	Kaggle CXR	98.72%	Covid - 96.27%, Non findings- 99.3%, Pneumonia- 99.66%, Tuberculosis - 98.10%	Covid - 97.00% Non findings - 98.72%, Pneumonia - 99.89%, Tuberculosis - 98.90%	Covid- 96.63%, Non findings- 99.04%, Pneumonia- 99.77%, Tuberculosis- 98.50%
Jia et al. [27]	NEU-CLS	97.1%	-	-	-
Nemer et al.[30]	sea coral images	AlexNet (AC)- 83.34% , SqueezeNet (AC) - 80.86% , GoogLeNet (AC) - 90.6% Inception-v3 (AC)- 93.18%	-	-	-
Kulkarni et al. [46]	Multiclass Kaggle dataset and the NIH dataset (COVID-19, Pneumonia, TB)	97%	98%	97%	Covid - 97%, Pneumonia - 96%, TB- 0.98%
PPCS+ MLCNN	Covid-19, Lung cancer, pulmonary Tuberculosis, Pneumonia, or Normal	98.89%	98.38%	98.89%	98.54%

in this context. Several classification rounds are carried out using different test sets of the same size following the model training procedure. The Table displays the number of accurate and inaccurate predictions for each class. This facilitates comprehension of the classes that the model is mistaking for another class. The PPCS+MLCNN model outperformed standard MLCNN when it came to lung medical images. The five explained lung disorders can be reliably and automatically identified by our proposed procedure.

In deep learning, some stochastic training effects are required to demonstrate the stability or dependability of the outcomes such as Confidence interval (CI), which indicates that results are trustworthy and that the true accuracy is near our reported mean and Standard deviation (SD). SD indicates that our model performs consistently throughout training runs. We computed this for our model using five-fold validation and 100 rounds. Since the SD is approximately 0.04(SD = 0.1%, which is quite low and indicates strong stability), the values are closely grouped. A 95% confidence interval [98.01% - 98.03%] is attained, indicating a closely defined confidence interval.

As seen in Table 4, we constructed the confusion matrix for the suggested PPCS+MLCNN model. Having the greatest proportion to the standard images (0.9712), lung cancer (0.9712), COVID-19 (0.9790), pneumonia (0.9812), and Tuberculosis (0.9822), the table demonstrates that the PPCS+MLCNN model can successfully classify the five patient statuses (COVID-19, Pneumonia, Lung Cancer, Tuberculosis and Normal). This outcome guarantees that the five statuses are accurately classified.

We also demonstrated disease-wise comparisons between our results and the pre-existing model that identifies certain diseases in Table 6, 7, 8 and 9.

Table 6: Measures for the Covid-19 evaluation

Model and Reference	Disease	Dataset	Accuracy	F1 score	Precision	Recall	Privacy
(VGG19 + Attention CNN) [47]	Covid-19	MosMed database+ curated dataset of COVID-19 lung CT scans	97.52	-	95.71	98.53	Yes
PPCS + MLCNN	Five Lung diseases	Covid-19, Lung cancer, pulmonary Tuberculosis, Pneumonia, and Normal lung Dataset	98.89	98.54	98.89	98.38	Yes

Table 7: Measures for the lung cancer evaluation

Model and Reference	Disease	Dataset	Accuracy	F1 score	Precision	Recall	Privacy
CNN (GoogleN-et + Vgg16) [48]	Lung Cancer	Sathybama Hospital, Chennai CT images	98	98	99	99	No
PPCS + MLCNN	Five Lung diseases	Covid-19, Lung cancer, pulmonary Tuberculosis, Pneumonia, and Normal lung Dataset	98.89	98.54	98.89	98.38	Yes

Table 8: Measures for the pneumonia evaluation

Model and Reference	Disease	Dataset	Accuracy	F1 score	Precision	Recall	Privacy
VGG19 + CNN [49]	Pneumonia	COVID-19, pneumonia, and lung cancer Dataset	98.05	98.24	98.43	99.5	No
PPCS + MLCNN	Five Lung diseases	Covid-19, Lung cancer, pulmonary Tuberculosis, Pneumonia, and Normal lung Dataset	98.89	98.54	98.89	98.38	Yes

Table 9: Measures for the Tuberculosis evaluation

Model and Reference	Disease	Dataset	Accuracy	F1 score	Precision	Recall	Privacy
VGG19 + CNN [50]	Tuberculosis	Pulmonary TB CT image dataset	96.08	-	95	96	No
PPCS + MLCNN	Five Lung diseases	Covid-19, Lung cancer, pulmonary Tuberculosis, Pneumonia, and Normal lung Dataset	98.89	98.54	98.89	98.38	Yes

6 Discussion

An experiment comprising combinations of Networks VGG19, GoogLeNet, Inception V3, AlexNet, and the proposed network were carried out to determine the best performing model. The metrics of performance made by many well-known CNN pre-existing models to diagnose lung disorders in terms of Loss, accuracy, precision, recall, F1-score are displayed in Table 5.

By providing a single point of detection instead of requiring multiple programs to identify each disease independently, the proposed work can be considered the most recent attempt to introduce a single deep learning model to identify various underlying chest ailments. This shortens the decision-making process and eases the burden on the user.

State-of-the-Art comparison: Privacy-preserving image search (PPIS) [25] obtained an average 86% accuracy rate (Chest X-Ray, Retinal OCT, Blood Cell). Our model performs better than this since it continuously maintains high accuracy in increasingly intricate multi-class scenarios. The reason for this growth is that our approach may jointly optimize features from MLCNN integrated inception layer architecture. The author employed a basic CNN model while providing an effective security approach.

Qi et al. [26], images encrypted using the suggested encryption technique were satisfactorily classified with comparatively less processing by the ConvMixer model with an adaptive permutation matrix. However, there are still gaps to obtain accurate classifications, although the authors employed two networks for the datasets: ConvMixer-512/16 obtained 89.14% accuracy and ConvMixer-512/16+ achieved 92.65% accuracy for the CIFAR-10 dataset. However, compared to a model trained on unencrypted images, which was ground breaking work in this field, the strategy did not significantly increase the calculation cost.

Jia H. et al. [27] the study offers a new method for classifying images that protects data privacy by using a modified CNN and encrypted image chunks. They use CNN to carry out the encrypted classification system after first dividing the input images into image chunking encryption. To keep the spatial links between various areas of the image, the encrypted sections were then combined into chunk groups while retaining their original order. However, classifying images might take a long time, and the accuracy rate was 97.1%; which the authors considered unsatisfactory.

Alishahi et al. [28] examined how classifiers accuracy changes as they are trained on subset of features chosen by the suggested LDP-FS. To achieve this, a selection of datasets is used to train four well-known classification algorithms: Random Forest, k-Nearest Neighbors (kNN), AdaBoost, and Support Vector Machine (SVM) both before and after 20% of the irrelevant characteristics are eliminated. After applying the Local Differential Privacy Feature Selection technique to various datasets such as Census, Obesity, KDD99 and many more dataset, they achieved slightly better performance compare to previous approaches.

As an implicit regularizer, PPCS is used. PPCS forces the network to ignore noise and redundant information and concentrate only on the most important features by compressing and sparsifying the input data. This enhances generalization to new data and lowers the possibility of overfitting.The deeper and richer feature hierarchies created by the conception modules are one way they contribute. They improve the network's ability to identify both fine and coarse patterns in the input data by enabling it to capture multi-scale characteristics at once. In contrast to conventional CNN designs, this aids the model in learning more robust and discriminative features.This work can leads to longer processing times and more memory usage than conventional plaintext-based models. Real-time deployment and scalability may be impacted by these additional expenses. Given the vital significance of protecting patient data privacy in medical AI applications, we do, however, think this expense is acceptable. Additional optimization methods can be used to lessen the computational load without sacrificing security, including model pruning, hardware acceleration (such as GPUs/TPUs), and partial decryption approaches.

7 Conclusion

Detecting image similarity is crucial for many real-world applications. Existing approaches presume that the images that need to be matched are openly accessible. However, in many cases, patients are unwilling to share their collections with one another. In this study, we developed a privacy preserving image-searching technique. Quick and accurate image searches in feature maps are made possible by the protocol; it blends completely homomorphic encryption's semantic security with supervised learning. On real-world datasets with the same CNN structure, the experimental results demonstrate that PPCS outperforms earlier systems in terms of accuracy rate and searching time (more than six times faster). The MLCNN model performs better than others due to the deeper, multi-scale feature learning provided by the inception modules and the regularization function of PPCS, which protects against overfitting and improves generalization.

Acknowledgement

Contributions from the MedSeg Lung CT Dataset, MosMedData, and other publicly accessible datasets are much appreciated by the authors, as they were crucial in the creation of the database used in this investigation. Medical imaging and disease diagnose research has benefited greatly from their availability

References

[1] Hilbert, M. & López, P. (2011). The world's technological capacity to store, communicate, and compute information (PDF). Science 332, 60(6025), 1095-9203. https://doi.org/10.1126/science.1200970.

[2] Lindsey, L.B. How to grow faster. Atlantic Economic Journal 25, 7–17 (1997). https://doi.org/10.1007/BF02298473.

[3] Hellerstein, Joe. (2012). Parallel Programming in the Age of Big Data. Gigaom Blog. Archived from the original on 7 October 2012.

[4] Alabdullah B., Beloff N. and White M. Rise of Big Data – Issues and Challenges. 2018 21st Saudi Computer Society National Computer Conference (NCC), Riyadh, Saudi Arabia, 2018, pp. 1-6. 10.1109/NCG.2018.8593166.

[5] Oussous A. et al. (2018). Big Data technologies: A survey. Journal of King Saud University - Computer and Information Sciences, Volume 30, Issue 4,2018, 431-448,ISSN 1319-1578. https://doi.org/10.1016/j.jksuci.2017.06.001.

[6] Patgiri, Ripon & Ahmed, Arif. Big Data: The V's of the Game Changer Paradigm. In Proceedings of the 18th IEEE High Performance Computing and Communications, Sydney, NSW, Australia, pp. 17–24, December 2016. 10.1109/HPCC-SmartCity-DSS.2016.0014.

[7] Hukkeri, G. et al. (2019). Handling 3vs of Big Data Through Swarm Intelligence. 2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), Bangalore, India, 2019, pp. 589-595. 10.1109/RTEICT46194.2019.9016846.

[8] Belle, A., Fouad H. Awad & Murtadha M. Hamad (2023). Big Data Clustering Techniques Challenges and Perspectives: Review. Informatica, volume 47, issue ,str. 203-218. https://doi.org/10.31449/inf.v47i6.4445.

[9] Khan N, Yaqoob I, Hashem IA, Inayat Z, Ali WK, Alam M, Shiraz M, Gani A, " Big data: survey, technologies, opportunities,and challenges," ScientificWorldJournal, vol. 2014:712826, pages. 1-18, Jul 17, https://doi.org/10.1155/2014/712826.

[10] Kouanou, AT., Tchiotsop, D., Kengne, R., Zephirin, DT., Armele, NM., Tchinda R. (2018). An optimal big data workflow for biomedical imageanalysis. Inf Med Unlocked,11,68–74. https://doi.org/10.1016/j.imu.2018.05.001.

[11] Andreu-Perez, J., Poon, C, Merrifield, R., Wong, S., Yang, G.(2015). Big data for health. IEEE J biomedical health Inf,19(4),1193–208. https://doi.org/10.1109/JBHI.2015.2450362.

[12] Cirillo, D. & Valencia, A.(2019). Big data analytics for personalized medicine. Curr Opin Biotechnol,58,161–7. https://doi.org/10.1016/j.copbio.2019.03.004.

[13] Pääkkönen, P. & Pakkala, D. (2015). Reference architecture and classification of technologies, products and services for big data systems.Big data research,2(4),166–86. https://doi.org/10.1016/j.bdr.2015.01.001.

[14] Schmidhuber, J. (2015). Deep learning in neural networks: an overview, Neural Netw. 61,85–117. https://doi.org/10.1016/j.neunet.2014.09.003.

[15] Dieleman, S., Willett, Dambre, J.(2015). Rotation-invariant convolutional neural networks for galaxy morphology prediction, Mon. Notices R. Astron. Soc. 450,1441–1459. https://doi.org/10.48550/arXiv.1503.07077.

[16] Huval, B., Wang, T., Tandon, S., Kiske, J., Song, W., Pazhayampallil, J., Andriluka, M., Cheng-Yue, Mujica, F. (2015). An Empirical Evaluation of Deep Learning on Highway Driving. https://doi.org/10.48550/arXiv.1504.01716.

[17] H. Li et al. (2015). A convolutional neural network cascade for face detection, In, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,5325–5334. https://doi.org/10.1109/CVPR.2015.7299170.

[18] Farfade, S. et al. (2015).Multi-view face detection using deep convolutional neural networks, In, Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, ACM, 643–650. https://doi.org/10.48550/arXiv.1502.02766.

[19] Karpathy, A. et al. (2014). Fei-Fei, Large-scale video classification with convolutional neural networks, In, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1725–1732. https://doi.org/10.1109/CVPR.2014.223.

[20] Zbontar, J. & LeCun, Y. (2015). Computing the stereo matching cost with a convolutional neural network, In, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1592–1599. https://doi.org/10.48550/arXiv.1409.4326.

[21] Ji S. et al. (2013). 3D convolutional neural networks for human action recognition, IEEE Trans. Pattern Analysis Mach. Intell. 35,221–231. https://doi.org/10.1109/TPAMI.2012.59.

[22] Sudholt, S. & Fink, G. (2016). Phocnet: A Deep Convolutional Neural Network for Word Spotting in Handwritten Documents. https://doi.org/10.48550/arXiv.1604.00187.

[23] Tang, et al. (2014). Feature Selection for Classification: A Review. CRC Press, 37–64. https://doi.org/10.1201/b17320.

[24] Guyon, I. &Elisseeff, A. (2003). An introduction to variable and feature selection. J. Mach. Learn. Res. 3 (null), 1157–1182. https://doi.org/10.1162/153244303322753616.

[25] Guo C, et al. (2020).Privacy-preserving image search (PPIS): Secure classification and searching using convolutional neural network over large-scale encrypted medical images.Computers & Security,Volume 99,2020,102021,ISSN 0167-4048,

https://doi.org/10.1016/j.cose.2020.102021

[26] Qi, Z., MaungMaung, A., Kiya, H.(2023). Privacy-Preserving Image Classification Using ConvMixer with Adaptative Permutation Matrix and Block-Wise Scrambled Image Encryption. J. Imaging 2023, 9, 85. https://doi.org/10.3390/jimaging9040085.

[27] Jia, H. et al. (2023) .Efficient and privacy-preserving image classification using homomorphic encryption and chunk-based convolutional neural network. J Cloud Comp 12, 175. https://doi.org/10.1186/s13677-023-00537-0.

[28] Alishahi, M. &Moghtadaiee V &Navidan, Hojjat. (2022). Add Noise to Remove Noise: Local Differential Privacy for Feature Selection. Computers & Security. 123. 10.1016/j.cose.2022.102934.

[29] Sadoon, Ruaa N., Chaid, Adala M., Nini, Brahim (2024). Classification of pulmonary diseases using a deep learning stacking ensemble model. Informatica (Ljubljana), volume 48, issue 14, str. 43-63. URN:NBN:SI:DOC-5JN6KCP4. https://doi.org/10.31449/inf.v48i14.6145.

[30] Nemer Z N. Et al. (2023). Implementation of Multiple CNN Architectures to Classify the Sea Coral Images.Informatica, volume 47, issue ,str. 43-50. https://doi.org/10.31449/inf.v47i1.4429.

[31] Ning W, Lei S, Yang J, et al, "iCTCF: an integrative resource of chest computed tomography images and clinical features of patients with COVID-19 pneumonia," in Research Square; 2020. 10.21203/rs.3.rs-21834/v1.

[32] Suham Mukherjee, September 2020, accessed in June 2021. https://www.kaggle.com/soham1024/chest-ct-scans-with-covid19.

[33] E. Soares, P.Angelov, S. Biaso, M. H. Froes, and D.K.Abe, "SARS-CoV-2 CT-scan dataset :A large dataset of real patients CT scans for SARS-CoV-2 identification," medRxiv, 2020, [Online] Available: https://www.medrxiv. org/content/10.1101/2020.04.24.20078584v3.

[34] S. G. Armato III et al., "The lung image database consortium (LIDC) and image database resource initiative (IDRI): A completed reference database of lung nodules on CT scans," Med. Phys., vol. 38, no. 2, pp. 915–931, 2011.

[35] C.D. KW, Y.P.S.M. Cheng, K.P. Hui, P. Krishnan, Y. Liu, D.Y. Ng, Deep-learning framework to detect lung abnormality–a study with chest x-ray and lung ct scan images, Clin. Chem. 66 (2020) 549–555. 10.1016/j.patrec.2019.11.013.

[36] Hacking C, Knipe H, Bell D, et al. Pneumonia. Reference article, Radiopaedia.org (Accessed on 11 Jun 2025). https://doi.org/10.53347/rID-39216.

[37] Gaillard F, Silverstone L, Walizai T, et al. Tuberculosis (pulmonary manifestations). Reference article, Radiopaedia.org (Accessed on 11 Jun 2025) https://doi.org/10.53347/rID-8631.

[38] Shaikh, Eman et al., "Apache Spark: A Big Data Processing Engine," in 2nd IEEE Middle East and North Africa COMMunications Conference (MENACOMM), pp. 1-6, 2019. 10.1109/MENACOMM46666.2019.8988541.

[39] Ioffe, S. & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift,arXiv:1502.03167. https://arxiv.org/abs/1502.03167.

[40] Choe, J., Lee, SM. , Do K-H. , Lee G. ,Lee J-G. , Lee SM. et al.(2019). Deep learning–based image conversion of CT reconstructionkernels improves radiomics reproducibility for pulmonary nodules or masses. Radiology ,292(2),365–73, 2019. https://doi.org/10.1148/radiol.2019181960.

[41] Cheon, J.H., Kim, A., Kim, M., Song, Y. (2017). Homomorphic Encryption for Arithmetic of Approximate Numbers. In: Takagi, T., Peyrin, T.(eds) Advances in Cryptology – ASIACRYPT 2017. ASIACRYPT 2017. Lecture Notes in Computer Science(), vol 10624. Springer, Cham. https://doi.org/10.1007/978-3-319-70694-8_1.

[42] Gulli, A. & Pal, S.(2017). Deep Learning with Keras. Packt Publishing Ltd. https://dl.acm.org/doi/10.5555/3153803.

[43] Ozkaya U.et al. (2020). Coronavirus (Covid-19) Classification Using Deep Features Fusion and Ranking Technique.In Big Data Analytics and Artificial Intelligence Against COVID-19: Innovation Vision and Approach, Vol. 78, pp. 281-295, 2020. https://doi.org/10.48550/arXiv.2004.03698.

[44] Kingma et al. (2015).Adam: A Method for Stochastic Optimization. CoRR, Vol. abs/1412.6980, 2015. https://doi.org/10.48550/arXiv.1412.6980.

[45] Ahmed MS et al. (2023). Joint Diagnosis of Pneumonia, COVID-19, and Tuberculosis from Chest X-ray Images: A Deep Learning Approach. Diagnostics (Basel). 2023 Aug 1;13(15):2562. PMID: 37568925; PMCID: PMC10417844. 10.3390/diagnostics13152562.

[46] Kulkarni, Aditya, et al. "Advancing diagnostic precision: Leveraging machine learning techniques for accurate detection of covid-19, pneumonia, and tuberculosis in chest x-ray images." arXiv preprint arXiv:2310.06080 (2023).

[47] Chowa, S.S. et al. (2025). An automated privacy-preserving self-supervised classification of COVID-19 from lung CT scan images minimizing the requirements of large data annotation. Sci Rep 15, 226 (2025). https://doi.org/10.1038/s41598-024-83972-6.

[48] Pandian R. et al. (2022).Detection and classification of lung cancer using CNN and Google net,Measurement: Sensors,Volume 24,2022,100588,ISSN 2665-9174, https://doi.org/10.1016/j.measen.2022.100588.

[49] Dina M. Ibrahim et al.(2021).Deep-chest: Multi-classification deep learning model for diagnosing COVID-19, pneumonia, and lung cancer chest

diseases. Computers in Biology and Medicine, Volume 132,2021,104348,ISSN 0010-4825. https://doi.org/10.1016/j.compbiomed.2021.104348

[50] Yang, Z., & Wang, X. (2024). A Study on Tuberculosis CT Image Classification Based on Federated Learning Methods. International Journal of Computer Science and Information Technology, 2(2), 369-379. https://doi.org/10.62051/ijcsit.v2n2.43.