Enhancing Multiclass Pneumonia Classification with Machine Learning and Textural Features

A. Beena Godbin, S. Graceline Jasmine *
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India
Corresponding author: S. Graceline Jasmine (graceline.jasmine@vit.ac.in)

Abstract The highly infectious and mutating COVID-19, known as the novel coronavirus, poses a substantial threat to both human health and the global economy. Detecting COVID-19 early presents a challenge due to its resemblance to pneumonia. However, distinguishing between the two is critical for saving lives. Chest X-rays, empowered by machine learning classifiers and ensembles, prove effective in identifying multiclass pneumonia in the lungs, leveraging textural characteristics such as GLCM and GLRLM. These textural features are instilled into the classifiers and ensembles within the domain of machine learning. This article explores the multiclass categorization of X-ray images across four categories: COVID-19-impacted, bacterial pneumonia-affected, viral pneumonia-affected, and normal lungs. The classification employs Random Forest, Support Vector Machine, K-Nearest Neighbor, LGBM, and XGBoost. Random Forest and LGBM achieve an impressive accuracy of 92.4% in identifying GLCM features. The network's performance is evaluated based on accuracy, precision, sensitivity and F1-score.

Keywords: COVID-19, chest X-ray, feature extraction, GLCM, GLRLM, Machine Learning, Random Forest, XGB, SVM.

1. Introduction

Pneumonia, an acute respiratory infection caused by bacteria or viruses, can result from a variety of factors and is a leading cause of mortality, particularly affecting vulnerable populations such as the elderly and children. It can present as a mild ailment in individuals of all ages. In March 2020, the World Health Organization (WHO) declared the emergence of the new Coronavirus 2019, widely known as COVID-19, as a global pandemic [32]. This virus, identified as SARS-CoV-2, is linked to Severe Acute Respiratory Syndrome and is frequently associated with respiratory symptoms. Originating in China, the virus rapidly spread to other nations, leading to a global pandemic that has significantly impacted individuals worldwide, resulting in numerous fatalities. Factors such as the recent discovery of the virus, delayed detection, limited testing capabilities, insufficient medical expertise, and its resemblance to pneumonia-related illnesses have collectively impeded the medical community's ability to effectively combat the virus until relatively recently. Pneumonia, characterized by the accumulation of air or pus in the alveoli of the lungs, can be caused by bacterial, fungal, or viral infections, all of which have the potential to trigger severe allergic reactions [36]. This particular instance manifests symptoms such as coughing, respiratory distress, fatigue, fever, and profuse sweating, which are also commonly observed in patients with COVID-19. The chest radiographs of individuals with COVID-19 exhibit patterns reminiscent of those found

Machine GRAPHICS & VISION 32(3/4):83–106, 2023. DOI: 10.22630/MGV.2023.32.3.5 .

in pneumonia cases, indicating the presence of the virus. The radiological findings of COVID-19 on chest X-rays closely resemble those of pneumonia, as reported by imaging departments. Numerous studies have utilized X-ray imaging for the diagnosis and classification of pneumonia [23]. While comprehensive screening of Coronavirus samples through reverse transcription polymerase chain reaction (RT-PCR) may be insufficient in effectively curbing the global spread of the virus, chest X-rays have proven highly useful in triaging patients infected with COVID-19. The rapid and widespread proliferation of the virus has placed a substantial burden on health and medical organizations worldwide. Consequently, the development of technology capable of distinguishing between patients with pneumonia or normal chest X-rays and those with COVID-19 has become imperative. This urgency arises from the inability of chest X-rays, despite thorough analysis, to differentiate between pneumonia patients and individuals suspected of having COVID-19. Researchers worldwide propose the employment of texture-based cognitive strategies to identify COVID-19. Current research is predominantly focused on extracting texture characteristics from chest X-ray images using methods such as GLCM and GLRLM. The advocated strategy for early detection of potential COVID-19 cases involves categorizing X-ray images into four predefined classes: COVID-19, bacterial pneumonia-affected, viral pneumonia-infected, and normal. Given the severity of the current situation, addressing this matter with utmost urgency is paramount. In this paper, the following contributions are made:

- This paper introduces a method enabling accurate differentiation between COVID-19 infection and pneumonia in chest X-ray images, addressing a crucial diagnostic challenge.
- This research pioneers accuracy in disease identification by combining textural features, specifically GLCM and GLRLM. The amalgamation enhances diagnostic precision, marking a notable advancement in medical image analysis.
- This study contributes by systematically evaluating various machine learning classifiers to enhance the accuracy of COVID-19 detection. The paper evaluates system performance using accuracy, sensitivity, and F1 score, offering a comprehensive snapshot of its diagnostic effectiveness.
- To discern between COVID-19, bacterial pneumonia, viral pneumonia, and healthy individuals, we adopted a tailored approach, training each model independently. This meticulous strategy ensures accurate identification across diverse medical conditions.

Section 2 provides a comprehensive review of the literature. Subsequently, in Section 3 the technical matters related to the methodology are presented. Namely, Section 3.1 expounds on the dataset specifics, Section 3.2presents the feature engineering approaches, Section 3.3 discusses the machine learning methodologies, andSection 3.4 presents the cross validation techniques. Section 4 presents the outcomes of our empirical investigations. Finally, Section 5 comprises a discourse on the findings and a conclusive statement.

2. Literature Survey

Ardakani et al. [2] suggested a computer-aided diagnostic (CAD) technique for distinguishing COVID-19 pneumonia patients from non-COVID-19 pneumonia patients. To this end, the authors employed a dataset of 612 CT images of pneumonia patients, where 306 patients were diagnosed with COVID-19 and the remaining 306 were diagnosed with non-COVID-19, of which 376 patients were COVID-19 positive. The researchers extracted 20 imaging features from the dataset and subjected them to classification using five different classifiers, namely, Decision Trees (DT), Naive Bayes (NB), K-nearest neighbors (KNN), Support Vector Machines (SVM), and Ensembles. The authors gained the highest level of accuracy, i.e., 91.94%, through ensemble classification. In another study, al-Karawi et al. [5] developed an automated model for COVID-19 analysis in CT scans utilizing CT scan images. In the dataset utilized, a total of 275 COVID-19 cases were identified as positive, while 195 were negative. The CT images were subjected to a Fast Fourier Transform, followed by the application of a Gabor filter for image manipulation. By employing the SVM technique for classification, a commendable accuracy of 95.37% was obtained. Barstugan et al. [8] incorporated 150 CT scans in their study and extracted four different patches from these scans (16×16, 32×32, 48×48, and 64×64) for comparative purposes. SVM was utilized to classify radiomic features obtained using FOS, GLCM, GLRLM, and GLSZM patches. The study further integrated a 10-fold plus DWT (Discrete Wavelet Transform) feature, and the highest accuracy recorded was 99.64%. The examination carried out by Dey and colleagues [14] scrutinized 400 CT scans of persons afflicted with COVID and 400 of non-afflicted with COVID-19, encompassing 200 CT scans for each cohort. A devised strategy enabled the creation of a system capable of segmenting the COVID-19 infected regions into smaller subsections, subsequently extracting data from each area discretely. Machine learning algorithms offer four distinct techniques for classifying entities into groups: Random Forest, Support Vector Machine, K-Nearest Neighbors and Decision Tree. The utilization of the K nearest neighbor algorithm in the conducted investigation resulted in an 88% accuracy rating.

In their scrutiny, Liu and colleagues [22] meticulously scrutinized 61 CT scan images of COVID-19 and 27 CT scan images of pneumonia in general, from which they meticulously extracted 35 statistical textural features. An array of models were meticulously evaluated, including but not limited to Support Vector Machine, Linear Regression, k-Nearest Neighbors and Decision Tree. The authors compared the Ensemble of bagged tree with the aforementioned models. The Ensemble model bagged tree classifiers attained the utmost level of accuracy, which yielded a rate of 94.16%. Ozkaya and colleagues employed an identical dataset and partitioned it into two subsets, namely Subset-1 with dimensions of 16×16 and Subset-2 with dimensions of 32×32. To identify distinctive features, they employed a design of convolutional neural network architecture in conjunction with a support vector machine algorithm. The accuracy rate for Subset-2 was 99.28%. With assistance from previously trained deep neural networks, Kassani and colleagues [20] have devised a methodology for extracting features. 117 images of X-ray and 20 images of CT with abnormalities were compared with 117 X-rays and 20 CT scans without abnormalities.

For sorting things into groups, we used XGBoost, Random Forest, Decision Tree, LightGBM, AdaBoost and Bagging algorithm. Bagging tree classifiers are 99% accurate when features are pulled with DenseNet121 and grouped.

CT scans were used in Shi et al.'s study; 1659 COVID-19 and 1028 bacterial pneumonia were classified as negatives. Radiomic and hand-made elements were taken from the infected areas. Linear Regression, Support Vector Machine, Random Forest and Neural Network were compared to a classification method based on Random Forest and LightGBM. In terms of the handmade features, the proposed method gave the best results, with an accuracy level of 89.4% [30].

According to Zheng et al. [39], CT scans could be detected with a 3D deep convolutional neural network (DeCoVNet). 313 participants had COVID-19 while 229 did not. They chose 540 of those to participate in their study. The model has undergone training using a straightforward 2D UNet in an integrated manner. The criteria can be changed to look for COVID-19. There is a best accuracy of 90.8%. The authors developed a three-class system using 618 CT scans with 219 COVID-19, 177 normal persons, and 226 influenza A pneumonia cases. A 3D CNN (Convolutional Neural Network) method was used to divide up the image using a transfer learning model. ResNet-18 classification and location-attention are used to model transfer learning [37]. The precision of their diagnostic outcomes pertaining to COVID-19, IAVP, and uninfected individuals was registered at an impressive 87.7%. Song et al.'s study involved the classification of 88 COVID-19 patients, 100 bacterial pneumonia patients, and 87 healthy individuals through the utilization of deep learning. Specifically the Details Relation neural model was employed, having already undergone training [31]. The researchers achieved a commendable multiclass classification accuracy rate of 94%. Meanwhile, Wang et al. [35] employed a deep learning method, which made use of transfer learning, along with a pre-trained GoogleNet inception model to detect COVID-19 cases. The accuracy of their classifications for 325 COVID-19 positive patients and 740 COVID-19 negative patients was 89.5% for each. A CNN classifier trained on GoogleNet acquired an accuracy rate of 82.14% in an experiment by Alsharman et al. [6]. The study conducted by the researchers included a total of 463 non-COVID-19 images and 349 COVID-19 CT images in their analysis [16].

A team of researchers developed seven Deep CNN models to classify images of pneumonia automatically. A sampling of models included in the table are CNN baselines, VGG16, Xception, DenseNet201, VGG19, InceptionResNetV2, Resnet50, InceptionV3 and MobileNetV2. It was similar to the work by Zhang et al. [38]. Based on deep learning, they created a diagnosis system for COVID-19 based on 3D ResNet18 and four segmentation models. As per the findings of Rajaraman et al. [29], it was determined that a total of five Convolutional Neural Network (CNN) models were employed in the screening process for detecting the outbreak of the COVID-19 virus. The models used were InceptionV3, VGG16, Xception, NasNetmobile and DenseNet201. Each X-ray picture belongs to one of six groups. In 99.26% of cases, the result was accurate. According to Tsiknakis et al. [34], modified deep CNN models could be used for the screening of COVID on chest X-ray images using the InceptionV3 model. Transfer learning models that have already been trained can find COVID-19 cases automatically, as demonstrated by Ahuja et al. [4]. Oh et al. [25] have conducted analogous research, wherein they have proposed the utilization of a CNN based on patches using ResNet18. To diagnose COVID-19 disease, Elasnaoui and Chawki [17] used 7 deep learning models that had already been trained. Chowdhury et al. [13] developed eight deep CNNs to detect COVID-19. The suggested models were tested using 3487 X-ray images with and without image augmentation. In the present set of images, it can be observed that 423 of them depict the COVID-19 virus, while 1485 portray the typhus bacterium, and the remaining 1579 images display normal conditions.

COVID-19 detection and classification could be improved by using a convolutional neural network model, as suggested by Apostolopoulos et al. [7]. An evaluation of MobileNetv2 was conducted using 3905 x-ray pictures, with the results showing its excellent performance in detecting COVID-19. Rahimzadeh and Attar [28] introduced a modified deep Convolutional Neural Network (CNN) leveraging Xception model and ReNet to identify COVID-19 from chest X-ray images. Their experimental findings reveal that the combined method demonstrates an impressive average accuracy of 91.4%, precision of 72.8%, sensitivity of 87.3%, and specificity of 94.2%. Notably, the model achieved outstanding performance with a remarkable 99.18% accuracy, 97.36% sensitivity, and 99.42% specificity. This highlights the effectiveness of their innovative model in accurately detecting COVID-19 cases. According to another study [1], the Decompose Transfer Compose (DeTraC) Convolutional Neural Network model was modified and revised. An accuracy rate of 97.35%, a sensitivity rate of 98.23%, and a specificity rate of 96.34% were successfully attained [3]. Afshar and colleagues constructed a convolutional neural network architecture founded on capsule networks to identify and diagnose COVID-19.

Our inquiry delved into the effectiveness of COVID CAPS via the utilization of two publicly accessible thoracic X-ray databases. Therefore, we achieved 95.8% specificity, 90% sensitivity, and 95.70% accuracy. The deep CNN technique VGG-16 was employed by Brunese et al. [11] to automatically and quickly identify COVID-19 in chest X-ray images. VGG-16 appears to be 97% accurate at diagnosing COVID-19 based on the results of our study. By using deep learning-based AI, Jin et al. [19] have suggested a method for detecting COVID-19 in CT san images. Due to the discoveries, the suggested

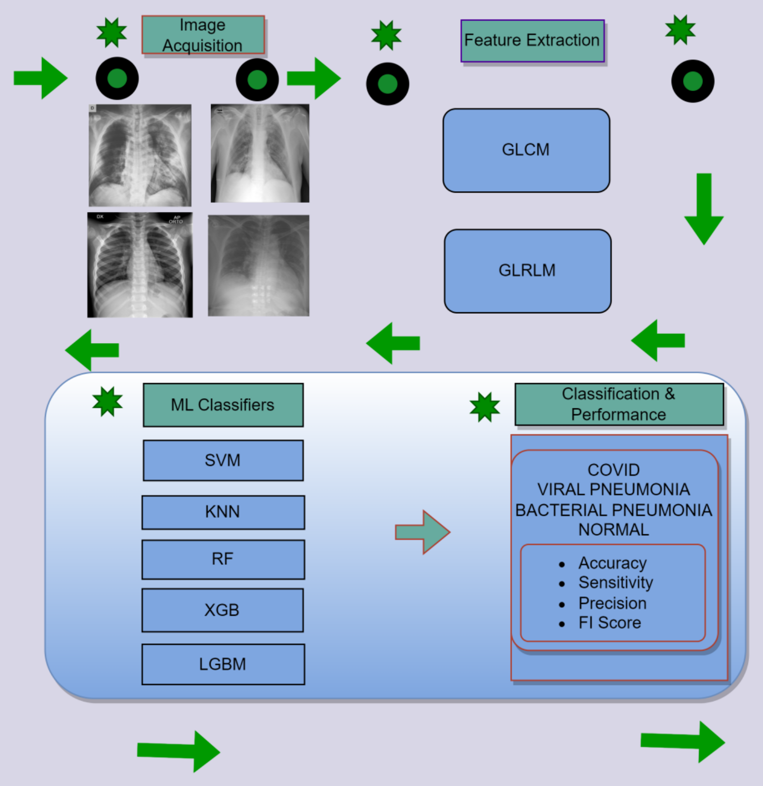


Fig. 1. Workflow of the proposed system.

framework exhibits a 95.8% area under the curve (AUC), a 90.19% level of sensitivity, and a 95.76% level of specificity.

3. Materials and methods

As illustrated in Figure 1, the following section provides an overview of the technologies applied. The first step was to extract numerous features (GLCM, GLRLM) from the CXT pictures using a variety of feature extraction techniques. A technique is proposed to study how different characteristics influence COVID-19 illness categorization. Classification accuracy varies depending on the feature set. An individual or a vector group of the retrieved characteristics is employed to assess their influence on the classification outcomes. Feature extracts and feature vectors were prepared and then the dataset was

Tab. 1. Dataset description.

Class	Images
Bacterial pneumonia	2000
Viral pneumonia	1345
COVID-19	2250
Normal	2370
Total	7965

divided into training set and test set. In order to train the machine learning models to categorize characteristics based on the most popular classifiers, we provided them with training data. The performance of the model was assessed by using 10-fold CV (Cross Validation) as a method of evaluation. Test sets with varying characteristics were evaluated ten times. As each step was completed, the classifier output served as the basis for determining the performance results.

3.1. Data set

The datasets used in this work are X-ray chest [24] and Covid19 radiography [33]. In the experiment, chest X-ray pictures were categorized into four classes: Covid19, Pneumonia bacterial, Pneumonia viral, and Normal. The collection contains 7965 chest X-ray pictures divided into four categories: Covid19 (2250 pictures), Normal (2370 pictures), Pneumonia viral class (1345 pictures), and Pneumonia bacterial class (2000 pictures) shown in Table 1. Sample images from the data set are shown in Fig. 2.

3.2. Feature Extraction Techniques

The study of radiological images numerical features is currently undergoing rapid expansion with the use of artificial intelligence techniques. The initial stage of this work involved analyzing the data for features. With the utilization of texture-based features, the process of identifying tissue sections with varying characteristics becomes simplified, as one can easily identify the connections and distinctive attributes between pixels. Further, statistical features can be derived from the matrices generated through the texture-based techniques after the texture feature has been extracted [15]. In order to extract texture-based attributes, matrices such as the Grey Level Run-Length Matrix (GLRLM) and the Grey Level Co-occurrence Matrix (GLCM) are employed.

3.2.1. GLCM

Image processing and analysis use the Gray-Level Co-occurrence Matrix (GLCM) approach to derive texture from pictures. It is possible to describe the spatial associations

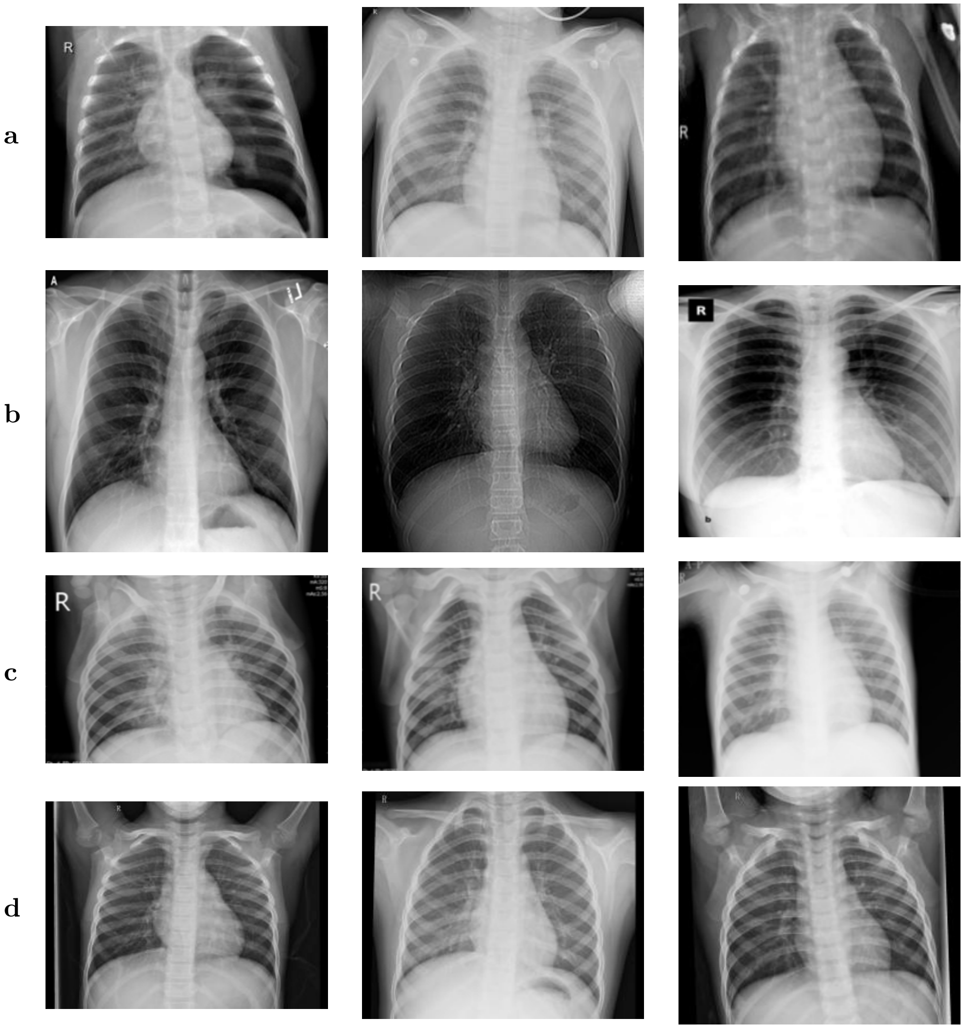


Fig. 2. Samples from the CXT dataset. (**a**) Images depicting bacterial pneumonia affected patients; (**b**) images depicting patients with COVID-19; (**c**) images of viral pneumonia patients; (**d**) images of normal patients.

between pixels with comparable gray-level values in a picture statistically. In the paper authored by Timo Ojala et al. [26], a comparative analysis is conducted on GLCM texture measures and classification methods utilizing featured distributions. GLCM is based on the idea that the distribution of pixel intensities, and their relation, can provide crucial information about a picture's texture. This is done by determining what pairs of gray-level values will appear at various spatial displacements within the picture.

The following stages are involved in producing a GLCM:

Greyscale: transformation in order to simplify the pixel intensity analysis.

Pixel pairing: finding occurrences of gray level pairs. GLCM matrix entries represent gray-level pairs' frequency of occurrence.

Angular Second Moment (ASM): finding the distribution of pixel pairs over the entire image.

Contrast: finding the differences or variations between pixel intensities.

Dissimilarity: finding the average difference between adjacent gray levels.

Energy: calculating the sum of squared elements in the GLCM.

Homogeneity: finding the proximity of the distribution of elements to the GLCM diagonal.

Maximum Probability: finding the most often occurring gray-level pair in the GLCM.

Sum of Squares: finding the GLCM's variance.

Correlation: measuring the linear dependence between gray-level values of neighboring pixels.

3.2.2. GLRLM

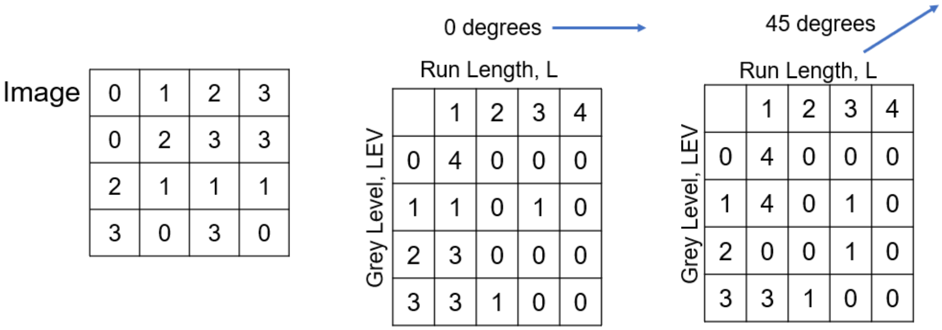
In their paper, Haralick et al. [18] introduced the GLRLM model, an acronym for Gray-Level Run Length Matrix. In the field of image processing feature analysis, the GLRLM is an approach which quantifies the distribution of consecutive gray-level pixels with the same intensity of an image, or the distribution of the gray-level runs. It provides statistical information on the lengths and frequency of these runs, and having this information can help characterize and distinguish textures in a picture much better. GLRLM is a process that is based on the idea that the organization and distribution of runs of comparable gray-level values within a picture may provide key textural information about this picture (see Fig.3). As a result of analyzing these runs, we can extract features that characterize the texture of the picture and identify aspects of its textural quality.

These are some of the most commonly computed GLRLM features:

Short Run Emphasis (SRE): represents how many short runs are there in the picture.

Long Run Emphasis (LRE): shows how many long runs are there in the picture.

Gray-Level Non-Uniformity (GLN): represents comparability, or consistency of the gray level values among runs.



Image

0	1	2	3
0	2	3	3
2	1	1	1
3	0	3	0

0 degrees

	1	2	3	4
0	4	0	0	0
1	1	0	1	0
2	3	0	0	0
3	3	1	0	0

45 degrees

	1	2	3	4
0	4	0	0	0
1	4	0	1	0
2	0	0	1	0
3	3	1	0	0

Fig. 3. Calculation of GLRLM (according to [27], license: CC BY 3.0).

Run Length Non-Uniformity (RLN): measures the variation or consistency in run lengths.

Run Percentage (RP): shows the share of the picture area where runs are present.

Run Entropy (RE): represents uncertainty associated with the run length.

3.3. Machine Learning Algorithms

In conjunction with the process of feature extraction, the retrieved features underwent training by machine learning models, and were subsequently evaluated on the test dataset. In light of their unyielding strength, we opted to employ the most formidable and extensively utilized machine learning methodologies. Noteworthy among these techniques for classification were the Support Vector Machine, K-Nearest Neighbor, XGBoost, LGBM and Random Forest.

3.3.1. Support Vector Machine

The SVM is a machine learning technique used for classification and regression. The method is particularly useful when data cannot be linearly separated or when the decision boundary is complex. Corinna et al. [9] presented a SVM training algorithm for optimal margin classifiers in their paper. Data points are divided into multiple groups using SVM's ideal hyperplane. In order to maximize the margin, the hyperplane is chosen to be as near as possible to the nearest data points from each class. Support vectors consists of the data points nearest to the given class.

Here is the outline of the SVM procedure:

Data representation: Selecting the optimal hyperplane: The SVM algorithm seeks the hyperplane with the highest margin. While minimizing classification error, the hyperplane should divide data points of various classes. This is accomplished by resolving an optimization problem.

Handling non-linear data: In cases where the data aren't linearly separable, SVM uses the "kernel trick". It transcends the data points to a realm of heightened dimensionality. A linear kernel, a polynomial kernel, and an RBF kernel are three common kernel functions.

Training: Through optimization of a cost function, the SVM algorithm learns the hyperplane's parameters and penalizes misclassifications. In order to accomplish this, a convex optimization problem needs to be solved.

Classification: The SVM possesses the remarkable ability to apprehend novel, indiscernible data points through its astute discernment of the side on which they lie on the established hyperplane. A newly-introduced data point is affiliated with a specific class contingent upon its placement on either side of the hyperplane; one side denotes one class and the other side denotes the other one.

3.3.2. K-Nearest Neighbor

In the field of machine learning, the K-nearest neighbor algorithm assumes a prominent role as a supervised decision tree algorithm that possesses the unique ability to tackle classification problems as well as regression problems. Richard et al. presented KNN algorithm in their paper, as reported by Duda et al. in their monograph [15]. The KNN method uses a distance calculation between the test data and all of the training points in order to try and predict which class the test data belongs to based on the data points in the test set. As a result, the number of K points closest to the test data is the number of points that should be chosen. As a result of implementing the KNN algorithm, we shall endeavor to compute the likelihood that the test data is affiliated with the classifications of K training data points. Subsequently, we shall elect the classification that possesses the most elevated probability of belonging to the test data set. As regards the scrutiny of regression analysis, the ascertained value is the mean of the data points from K arbitrarily picked training points. The aforementioned data points are then utilized for the evaluation of the regression.

3.3.3. Random Forest

The Random Forest, a captivating machine learning algorithm, has gained widespread popularity for its exceptional ability to perform classification and regression tasks in machine learning is shown in Figure 4. Breiman proposed the random forest algorithm in his publication, documented as [10]. By combining the predictions from many individual trees, a decision tree system can produce an outcome that is as accurate and reliable as possible.

Random Forest works as follows (see also the Algorithm 1):

As part of the training phase, Random Forest constructs decision trees. The decision trees are constructed using a subset of the original training data as well as the available features. During the process, randomization and bagging take place.

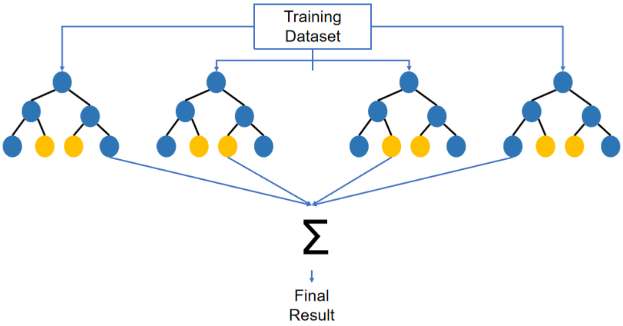


Fig. 4. Random Forest concept scheme.

Rather than considering all features at each node, the Random Forest evaluates the best split based on a random selection of features. In this way, randomness increases the diversity of the ensemble and reduces the correlation between trees.

Training trees: The training data for each decision tree is derived from a different bootstrap sample. As a result, each tree in the ensemble sees a slightly different subset of the original data.

As part of the prediction phase, Random Forest makes predictions based on input features. During classification tasks, the class receiving the most votes is chosen. For regression tasks, it is customary to derive the ultimate prediction through the computation of the mean or median of the individualized predictions.

3.3.4. XGBoost

A gradient boosted trees algorithm is implemented in an efficient and open-source manner using eXtreme Gradient Boosting (XGBoost). In their publication referenced as [12], Tianqi Chen and co-authors put forward a model that employs the XGBoost algorithm. Gradient boosting is a algorithm for supervised learning that combines simple and weaker models to produce accurate predictions of a target variable. With its capacity to manage an extensive range of data categories, connections, spreads, and hyperparameters that are adaptable, the XGBoost algorithm excels in machine learning competitions. You can use it for regression and classification.

3.3.5. LGBM

Guolin Ke et al. [21] proposed the Light Gradient Boosting Model (LGBM). By using lightGBM, gradient boosting models can be trained efficiently and at high performance. LightGBM improves training speed and model accuracy by using a gradient boosting algorithm. Gradient boosting allows LightGBM to combine multiple weak prediction

Algorithm 1 Random Forest

```
Data:
Training set $B$ with $m$ instances, $p$ features, and target value.
Total number of classes $K$; total number of classifiers $C$ in RF.
Procedure:
For $c = 1$ to $C$
    Generate a bootstrapped sample $B_c$ from the training set $B$.
    Construct a tree using a random feature subset from bootstrapped sample $B_c$.
For each node $t$ in the tree
    Select randomly $n \approx \sqrt{p}$ or $n \approx \frac{p}{3}$ features.
    Identify the best split features and cutpoints using the random feature subset.
    Send down the data using the best split features and cutpoints.
Develop trained classifier $D_c$:
    Group the $C$ trained classifier models using majority vote:
    Predicted label $D_c$: $D_c(x) = \arg\max_j \sum_{B_c} I(D_c(x) = j)$, for $j = 1, \ldots, K$.
```

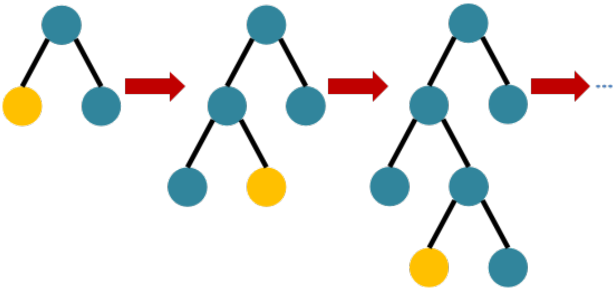


Fig. 5. LGBM-leafwise concept scheme.

models into one strong model (typically decision trees). A sequential correction is made for each successive tree based on the error of the previous tree as symbolically depicted in Figure 5 and outlined in Algorithm 2.

3.4. Cross validation

An important part of machine learning algorithms is the use of cross validation as one of the statistical methods used to evaluate the methods. A significant component of this procedure is the division of the data into two types: training set and validation set. The preeminent approach is the k-fold technique, which is unequivocally the most widely employed methodology. This method involves dividing the dataset into k equally divided parts (folds) for the purpose of training and validating the model of a dataset. During

Algorithm 2 LightGBM Classifier

```
Input:
$x\_train$: training data features
$y\_train$: set of training data labels
$x\_test$: set of test data features
$y\_test$: set of test data labels
Output:
Trained LightGBM classifier model
Initialization:
learning_rate = 0.02
max_depth = 8
random_state = 422
Fit the Model:
evaluation_set = $[(x\_test, y\_test), (x\_train, y\_train)]$
verbose = 20
evaluation_metric = 'logloss'
Output:
Trained LightGBM classifier model
```

the validation and training process, the validation of the model is executed through the utilization of distinct folds for each iterations in the training and validation process. Once all the folds have been averaged, the overall performance of the product can be obtained. A k-fold cross validation is shown in Figure 6.

4. Results

This part presents the outcomes of the classification based on CXT images of multiclass pneumonia based on the results of the investigation. The acquisition of all training and test outcomes was achieved through the utilization of a computer equipped with Windows 10 as the operating system and a memory capacity of 8 GB. As part of the analysis, Python 3.7.10 was used in conjunction with Scikit-learn 0.23.19. The types of algorithms used for classification include SVM, RF, KNN, XGBoost, and LGBM. It is possible to tune the hyperparameters of these classifiers in order to control the process of learning. For each method of classification, a number of parameters are controlled in order to achieve the best results. In addition, the hyperparameters for each classification methods were determined through a rigorous grid search coupled with a ten-fold cross-validation over the training set. The attainment of SVM results necessitated the consideration of several pertinent parameters. A conclusion was reached that the XGBoost and RF classifiers can produce similar outcomes for certain parameter variables, contingent upon

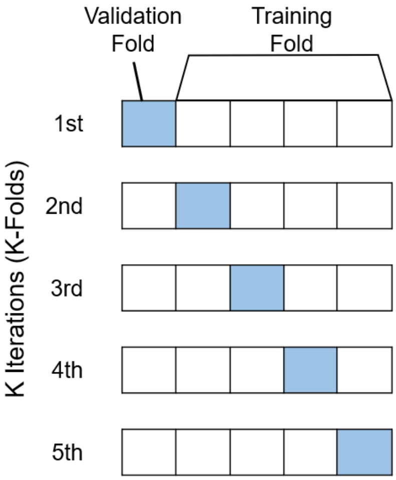


Fig. 6. The concept of k-fold cross validation.

the parameters on which they are trained. The parameters selected for these models are therefore from the range of parameters specified by the authors. A 10-fold Cross Validation technique was used in our study to obtain more reliable and realistic results. The performance of the model was evaluated using accuracy, sensitivity, precision and F1 score. As a proportion, accuracy can be delineated as the quotient of the number of accurate predictions and the aggregate number of forecasts. As a heuristic, precision can be characterized as the ratio of correct number of positive class predictions to the total number of positive class predictions. The assessment of prognostications is appraised through the ratio of accurate affirmative projections and erroneous negatory prognoses. Precision and sensitivity are averaged to calculate F1. Here are the formulas for each metric in terms of TP, TN, FP, and FN for each class i in a multi-class confusion matrix shown in equations (1–8).

$$\text{accuracy} = \text{ACC}_i = \frac{\text{TP}_i + \text{TN}_i}{\text{TP}_i + \text{TN}_i + \text{FP}_i + \text{FN}_i}\,, \tag{1}$$

$$\text{precision} = \text{PRE}_i = \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}\,, \tag{2}$$

$$\text{sensitivity} = \text{SNS}_i = \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}\,, \tag{3}$$

$$\text{F1-score} = \text{F1}_i = 2 * \left(\frac{\text{PRE}_i \times \text{SNS}_i}{\text{PRE}_i + \text{SNS}_i}\right)\,, \tag{4}$$

where the measures for class i are:
TP_i: number of True Positives for class i,
TN_i: number of True Negatives for class i (sum of all non-class i values in the confusion matrix),
FP_i: number of False Positives for class i (sum of all values in the row for class i, excluding the diagonal element),
FN_i: number of False Negatives for class i (sum of all values in the column for class i, excluding the diagonal element).

The macro-measures for all the classes $i = 1, \ldots, N$ are simply the average values:

$$\text{macro-accuracy} = \mathrm{MACC} = \frac{\sum_{i=1}^{N} \mathrm{ACC}_i}{N} \tag{5}$$

$$\text{macro-precision} = \mathrm{MPRE} = \frac{\sum_{i=1}^{N} \mathrm{PRE}_i}{N} \tag{6}$$

$$\text{macro-sensitivity} = \mathrm{MSNS} = \frac{\sum_{i=1}^{N} \mathrm{SNS}_i}{N} \tag{7}$$

$$\text{macro-F1 score} = \mathrm{MF1} = \frac{\sum_{i=1}^{N} \mathrm{F1}_i}{N} \tag{8}$$

The quantification of accurately predicted positive class instances is denoted as TP. Additionally, TN is also a constituent of the aforementioned metric, or true negative, is the number of examples of a class that are predicted correctly as negatives. False positives can be defined as negative examples that are misinterpreted as positive examples. As a result of false negative, there are examples of positive classes predicted to be negative. Several feature sets were tested to see how well the proposed method performed for different feature sets. The combination of features was tested in many different ways here as well.

Our approach to extracting GLCM or GLRLM features was based on a Python package. The computation is performed to determine the numerical worth of every characteristic attribute with regards to every individual angular measurement, then returns the mean value for each angle degree [1,28]. In order to calculate our features, we calculate them for the angles of 0, 45, 90 and 135 degrees. Results of using various features are shown in Tables 2 and 3. We incorporate an element of experimentation by varying the distance values used in the analysis. This approach yielded insignificant alterations in the outcome of our trials. The results of Table 2, which rely on GLCM characteristics, reveal that the LGBM classifier demonstrated the highest accuracy at 92.43%, whereas the random forest classifier produced the best F1 score of 96.02%. On the basis of GLRLM features, Table 3 shows that for LGBM classification, the best accuracy

Tab. 2. Results for GLCM features.

GLCM Classifier	Accuracy (%)	Precision (%)	Sensitivity (%)	F1-Score (%)
SVM	91.13	92.21	93.23	93.14
RF	92.04	94.03	99.02	96.02
KNN	90.01	93.12	91.43	92.41
XGB	91.14	95.54	95.27	95.07
LGBM	92.43	96.02	95.25	96.62

Tab. 3. Results for GLRLM features.

Classifier	Accuracy [%]	Precision [%]	Sensitivity [%]	F1-Score [%]
SVM	80.21	85.21	82.18	88.00
RF	89.12	100.00	99.11	100.00
KNN	82.19	99.04	98.90	99.00
XGB	86.80	98.08	97.35	100.00
LGBM	91.20	100.00	100.00	99.21

is 91.2%, whereas for RF classifiers and XGBs, the best F1 score is 100%. GLRLM-based machine learning models produce a higher degree of accuracy when compared to GLCM-based models.

Figure 7a shows an example of a confusion matrix for an SVM classifier. A TP of 215 is calculated, a TN of 35, and a FP of 47 is calculated. Figure 7b shows the confusion matrix for a simple LGBM classifier test. In the LGBM classifier for bacterial pneumonia, there are 376 TPs, 0 TNs, and 12 FPs.

The XGB classifier for COVID shown in Figure 8a has 384 TP, 47 TN, and 61 FP. The confusion matrix for Random Forest classifiers is illustrated in Figure 8b. For LGBM classifier for bacterial pneumonia, TP is 447, TN is 60, and FP is 49.

The F1 score was computed using precision and sensitivity as the basis. Figure 9 illustrates the percentage overall performance metrics achieved by all GLCM stride combinations considered.

In Figure 10, the percentage values of performance measures obtained for every GLRLM stride within a multiclass classification methods are presented using the Covid19 identification method. This accuracy value reaches 91.44 percent at its maximum. In a test X-ray images dataset for GLCM stride combinations, the proposed method of multiclass pneumonia identification with machine learning classifiers showed impressive results. Model performance is commonly visualized using ROC curves.

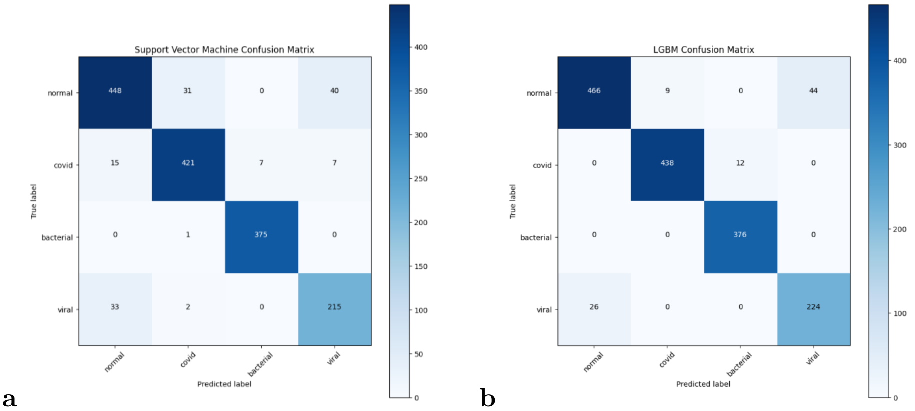


Fig. 7. Confusion matrix for GLCM features. (**a**) SVM; (**b**) LGBM.

5. Discussion and conclusion

In the initial step, the present study aims to classify multiclass pneumonia diseases using images from a CXT scanner. Furthermore, the purpose of this study is to examine whether a variety of feature extraction methods can improve classification accuracy. Medical images are characterized by gray levels of intensity, as opposed to sophisticated algorithms and features used in the previous methods. Using intensity-based features, one can analyze important properties of images. COVID-19 requires rapid detection of the diagnosis in order to be successful. CXT used datasets collected from several papers and collected in different ways to develop the proposed method. GLCM classifier has the best results for LBGM and RF with 92.4% and 91.5% accuracy respectively, while XGBoost has 91.5% accuracy for individual feature vectors. Compare GLRLMs with GLCMs, and GLRLMs give lower results. In GLCM and GLRLM, each feature is calculated separately for angles 0, 45, 90, and 135 degrees. Angle-based trials don't matter here. In this paper, we show that gray levels with high gray levels have more value than gray levels with low gray levels. With RF or LGBM classifiers, we get over 92% on GLCM features. GLRLM improved LGBM, XGBoost, and RF by 91.2%, 89.1%, and 86.8%, respectively. Ten-fold cross-validation provided reliable and robust results. According to this study, SVM and KNN classifiers perform less well than RF, XGB and LGBM classifiers. In most cases, RF classifiers will provide the best results. In addition, LGBM and XGB provide significant results as well. Our recommendation is that RF and LGBM classifiers for the purpose of classify COVID-19 classification. In the future, we intend to test our model on a variety of datasets and to improve our model performance to provide an improved diagnosis of COVID-19 from CT and CXT images.

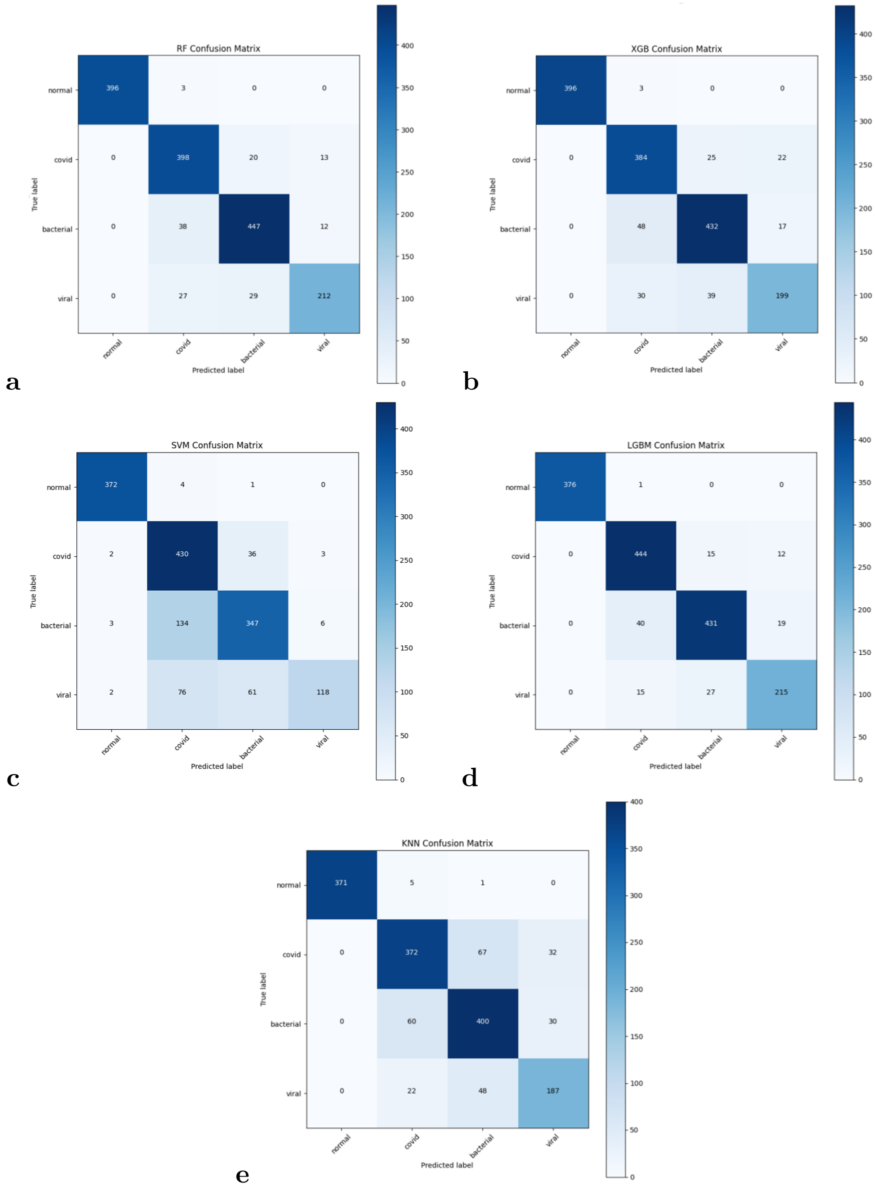


Fig. 8. Confusion matrix for GLRLM features. (**a**) RF; (**b**) XGB; (**c**) SVM; (**d**) LGBM; (**e**) KNN.

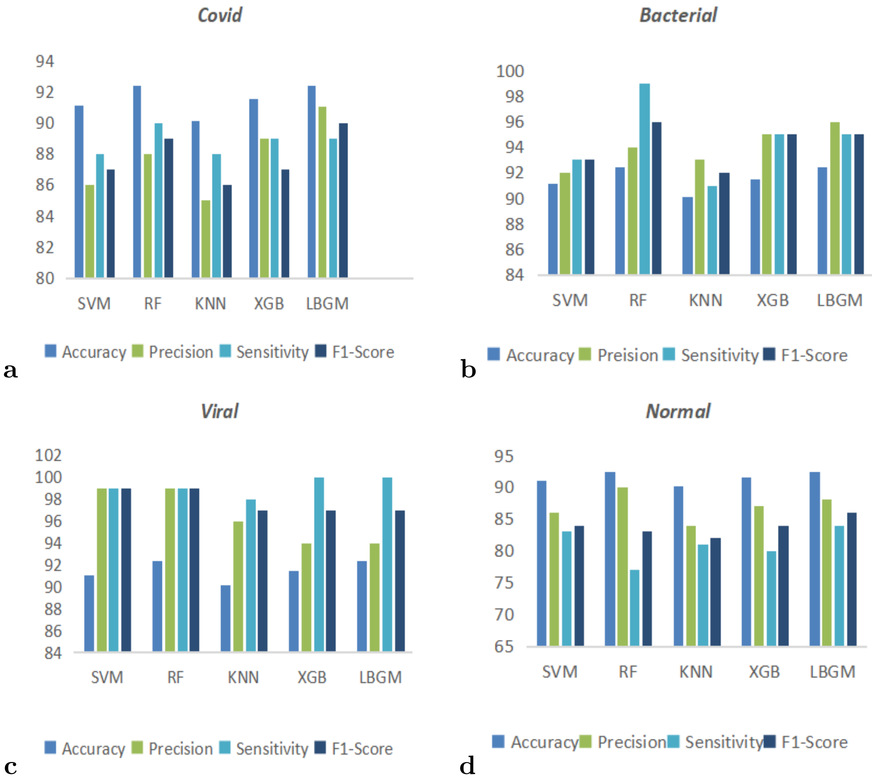


Fig. 9. Performance of classifiers for GLCM features. (**a**) COVID; (**b**) bacterial; (**c**) viral; (**d**) normal.

References

[1] A. Abbas, M. M. Abdelsamea, and M. M. Gaber. Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. *Applied Intelligence*, 51:854–864, 2021. doi:10.1007/s10489-020-01829-7.

[2] A. Abbasian Ardakani, U. R. Acharya, S. Habibollahi, and A. Mohammadi. COVIDiag: a clinical CAD system to diagnose COVID-19 pneumonia based on CT findings. *European Radiology*, 31:121–130, 2021. doi:10.1007/s00330-020-07087-y.

[3] P. Afshar, S. Heidarian, F. Naderkhani, A. Oikonomou, K. N. Plataniotis, et al. COVID-CAPS: A capsule network-based framework for identification of COVID-19 cases from X-ray images. *Pattern Recognition Letters*, 138:638–643, 2020. doi:10.1016/j.patrec.2020.09.010.

[4] S. Ahuja, B. K. Panigrahi, N. Dey, V. Rajinikanth, and T. K. Gandhi. Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices. *Applied Intelligence*, 51:571–585, 2021. doi:10.1007/s10489-020-01826-w.

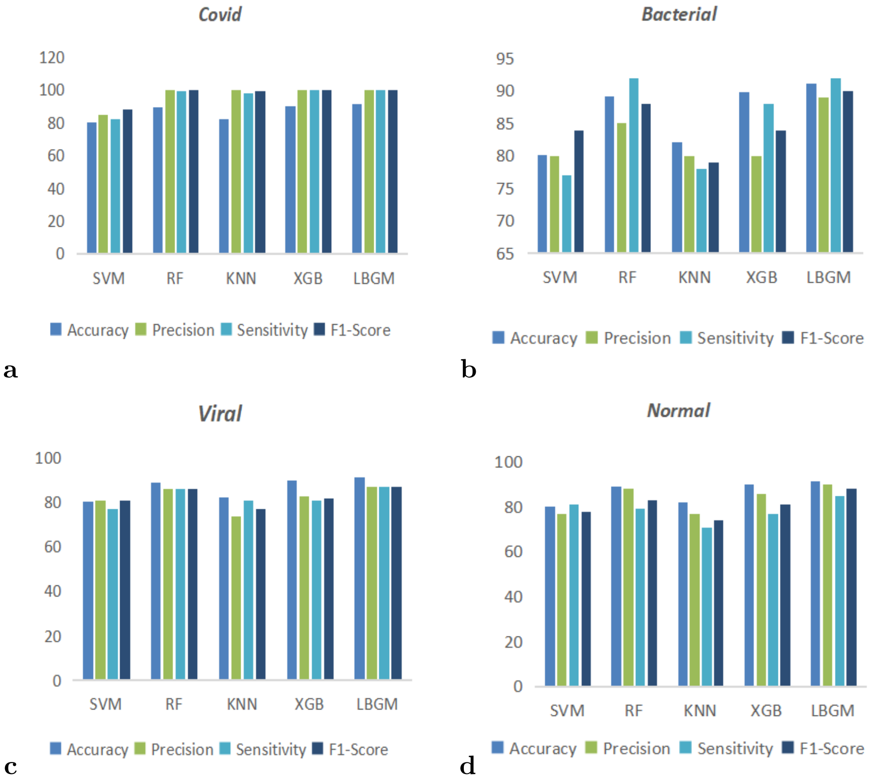


Fig. 10. Performance of classifiers for GLRLM features. (**a**) COVID; (**b**) bacterial; (**c**) viral; (**d**) normal.

[5] D. Al-Karawi, S. Al-Zaidi, N. Polus, and S. Jassim. Machine learning analysis of chest CT scan images as a complementary digital test of coronavirus (COVID-19) patients. *MedRxiv*, 2020. MedRxiv.2020.04.13.20063479. doi:10.1101/2020.04.13.20063479.

[6] N. Alsharman and I. Jawarneh. GoogleNet CNN neural network towards chest CT-coronavirus medical image classification. *Journal of Computer Science*, 16(5):620–625, 2020. doi:10.3844/jcssp.2020.620.625.

[7] I. D. Apostolopoulos, S. I. Aznaouridis, and M. A. Tzani. Extracting possibly representative COVID-19 biomarkers from X-ray images with deep learning approach and image data related to pulmonary diseases. *Journal of Medical and Biological Engineering*, 40:462–469, 2020. doi:10.1007/s40846-020-00529-4.

[8] M. Barstugan, U. Ozkaya, and S. Ozturk. Coronavirus (COVID-19) classification using CT images by machine learning methods. *arXiv*, 2020. ArXiv:2003.09424. doi:10.48550/arXiv.2003.09424.

[9] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In: *Proc. 5th Annual Workshop on Computational Learning Theory*, COLT ’92, pp. 144–152. Association for Computing Machinery, 1992. doi:10.1145/130385.130401.

[10] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. doi:10.1023/A:1010933404324.

[11] L. Brunese, F. Mercaldo, A. Reginelli, and A. Santone. Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays. *Computer Methods and Programs in Biomedicine*, 196:105608, 2020. doi:10.1016/j.cmpb.2020.105608.

[12] T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In: *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 785–794. Association for Computing Machinery, San Francisco, California, USA, 2016. doi:10.1145/2939672.2939785.

[13] M. E. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, et al. Can AI help in screening viral and COVID-19 pneumonia? *IEEE Access*, 8:132665–132676, 2020. doi:10.1109/ACCESS.2020.3010287.

[14] N. Dey, V. Rajinikanth, S. J. Fong, M. S. Kaiser, and M. Mahmud. Social group optimization–assisted kapur's entropy and morphological segmentation for automated detection of COVID-19 infection from computed tomography images. *Cognitive Computation*, 12:1011–1023, 2020. doi:10.1007/s12559-020-09751-3.

[15] R. O. Duda, P. E. Hart, et al. *Pattern Classification and Scene Analysis*, vol. 3. Wiley New York, 1973.

[16] K. El Asnaoui and Y. Chawki. Using X-ray images and deep learning for automated detection of coronavirus disease. *Journal of Biomolecular Structure and Dynamics*, 39(10):3615–3626, 2021. doi:10.1080/07391102.2020.1767212.

[17] K. El Asnaoui, Y. Chawki, and A. Idri. Automated methods for detection and classification pneumonia based on X-ray images using deep learning. In: *Artificial Intelligence and Blockchain for Future Cybersecurity Applications*, pp. 257–284. Springer, 2021. doi:110.1007/978-3-030-74575-2_14.

[18] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(6):610–621, 1973. doi:10.1109/TSMC.1973.4309314.

[19] C. Jin, W. Chen, Y. Cao, Z. Xu, Z. Tan, et al. Development and evaluation of an artificial intelligence system for COVID-19 diagnosis. *Nature Communications*, 11(1):5088, 2020. doi:10.1038/s41467-020-18685-1.

[20] S. H. Kassania, P. H. Kassanib, M. J. Wesolowskic, K. A. Schneidera, and R. Detersa. Automatic detection of coronavirus disease (COVID-19) in X-ray and CT images: A machine learning based approach. *Biocybernetics and Biomedical Engineering*, 41(3):867–879, 2021. doi:10.1016/j.bbe.2021.05.013.

[21] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, et al. LightGBM: A highly efficient gradient boosting decision tree. In: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, et al., eds., *Advances in Neural Information Processing Systems – Proc. NIPS 2017*, vol. 30. Curran Associates, Inc., 2017. https://proceedings.neurips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html.

[22] C. Liu, X. Wang, C. Liu, Q. Sun, and W. Peng. Differentiating novel coronavirus pneumonia from general pneumonia based on machine learning. *Biomedical Engineering Online*, 19(1):1–14, 2020. doi:10.1186/s12938-020-00809-9.

[23] H. Mohammad-Rahimi, M. Nadimi, A. Ghalyanchi-Langeroudi, M. Taheri, and S. Ghafouri-Fard. Application of machine learning in diagnosis of COVID-19 through X-ray and CT images: A scoping review. *Frontiers in Cardiovascular Medicine*, 8:638011, 2021. doi:10.3389/fcvm.2021.638011.

[24] P. Mooney. Chest X-Ray Images (pneumonia) Dataset, 2018. https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia, Kaggle dataset.

[25] Y. Oh, S. Park, and J. C. Ye. Deep learning COVID-19 features on CXR using limited training data sets. *IEEE Transactions on Medical Imaging*, 39(8):2688–2700, 2020. doi:10.1109/TMI.2020.2993291.

[26] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996. doi:10.1016/0031-32032895.2900067-4.

[27] K. Preetha and Dr. S. K. Jayanthi. GLCM and GLRLM based feature extraction technique in mammogram images. *International Journal of Engineering and Technology*, 7:266, 2018. doi:10.14419/IJET.V7I2.21.12378.

[28] M. Rahimzadeh and A. Attar. A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. *Informatics in Medicine Unlocked*, 19:100360, 2020. doi:10.1016/j.imu.2020.100360.

[29] S. Rajaraman and S. Antani. Weakly labeled data augmentation for deep learning: a study on COVID-19 detection in chest X-rays. *Diagnostics*, 10(6):358, 2020. doi:10.3390/diagnostics10060358.

[30] F. Shi, L. Xia, F. Shan, B. Song, D. Wu, et al. Large-scale screening to distinguish between COVID-19 and community-acquired pneumonia using infection size-aware classification. *Physics in Medicine & Biology*, 66(6):065031, 2021. doi:10.1088/1361-6560/abe838.

[31] Y. Song, S. Zheng, L. Li, X. Zhang, X. Zhang, et al. Deep learning enables accurate diagnosis of novel coronavirus (COVID-19) with CT images. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 18(6):2775–2780, 2021. doi:10.11092FTCBB.2021.3065361.

[32] A. Tahamtan and A. Ardebili. Real-time rt-pcr in covid-19 detection: issues affecting the results. *Expert Review of Molecular Diagnostics*, 20(5):453–454, 2020. doi:10.1080/14737159.2020.1757437.

[33] R. Tawsifur, C. D. Muhammad, and K. Amith. COVID-19 radiography database, 2019. doi:https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database, Kaggle dataset.

[34] N. Tsiknakis, E. Trivizakis, E. E. Vassalou, G. Z. Papadakis, D. A. Spandidos, et al. Interpretable artificial intelligence framework for COVID-19 screening on chest X-rays. *Experimental and Therapeutic Medicine*, 20(2):727–735, 2020. doi:10.3892/etm.2020.8797.

[35] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, et al. A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19). *European Radiology*, 31:6096–6104, 2021. doi:10.10072Fs00330-021-07715-1.

[36] M. Xu, D. Wang, H. Wang, X. Zhang, T. Liang, et al. COVID-19 diagnostic testing: Technology perspective. *Clinical and Translational Medicine*, 10(4):e158, 2020. doi:10.1002/ctm2.158.

[37] X. Xu, X. Jiang, C. Ma, P. Du, X. Li, et al. A deep learning system to screen novel coronavirus disease 2019 pneumonia. *Engineering*, 6(10):1122–1129, 2020. doi:10.1016/j.eng.2020.04.010.

[38] K. Zhang, X. Liu, J. Shen, Z. Li, Y. Sang, et al. Clinically applicable AI system for accurate diagnosis, quantitative measurements, and prognosis of COVID-19 pneumonia using computed tomography. *Cell*, 181(6):1423–1433, 2020. doi:10.1016/j.cell.2020.04.045.

[39] C. Zheng, X. Deng, Q. Fu, Q. Zhou, J. Feng, et al. Deep learning-based detection for COVID-19 from chest CT using weak label. *MedRxiv*, 2020. MedRxiv.2020.03.12.20027185. doi:10.1101/2020.03.12.20027185.

A. Beena Godbin received the B.Tech. degree in 2005 in information technology and engineering from Anna University, Chennai, Tamil Nadu, and the M.E. degree in computer science and engineering from Anna University, Coimbatore, Tamil Nadu, in 2019. She is currently pursuing the Ph.D. degree with the Vellore Institute of Technology, Chennai. Her research interests include medical image processing, machine learning and deep learning.

Dr. S. Graceline Jasmine, an accomplished professional with a Master's in Computer Application and a Ph.D. in Remotely Sensed Image Processing from VIT, serves as an Associate Professor at the School of Computer Science and Engineering, VIT, Chennai Campus. With 15 years of extensive experience in teaching and research, she has published around 50 papers in international journals and conferences. Currently, she is the research group chair of Imaging and Computer Vision Research Group at VIT Chennai and is engaged in diverse roles, including industrial consultancy and overseeing projects related to Image Processing, Computer Vision, and Remote Sensing. Dr. Jasmine is actively involved in numerous professional bodies and holds certifications such as NASSCOM certified Master Trainer of Associate Analytics and Remote Sensing and Digital Image Analyst certified by ISRO. Her recent projects include AI-based Smart Agriculture, Computer Vision for ST Microelectronics, and 3D weather reconstruction for the Indian Meteorological Department, showcasing her expertise in cutting-edge technologies and applications.