ISSN: 0067-2904

Classification of COVID-19 Disease Based on Extra Tree Features Selection

Asraa M. Mohammad [1*] , Hussien Attia [1], Yossra H. Ali [2]
[1] *Department of Computer Sciences, College of Science for Women, University of Babylon, Babylon, Iraq*
[2] *Department of Computer Sciences, University of Technology, Baghdad, Iraq*

Received: 25/5/2022 Accepted: 9/9/2023 Published: 30/11/2024

Abstract

The World Health Organization (WHO) has classified coronavirus as a global health emergency. Chest X-rays have been proven to be beneficial in both diagnosing and monitoring various lung diseases, including COVID-19. In this study, a COVID-19 disease detection framework is provided based on the methods used in machine learning and the Extra Tree algorithm to reduce the features extracted from the images. Using the Gray-Level Co-occurrence Matrix (GLCM) algorithm and the Extra Tree algorithm to choose the features of the work, a set of features is extracted from the images. This set of features is then put into the XGBoost algorithm to be classified. The proposed system was evaluated using two different sets of databases: the large database with 9544 images and the small database with 800 images. All image sizes were set to 300 x 300 pixels. The proposed system achieved a classification accuracy score of 90.04% using the large data set and 99.37% using the small set.

Keywords: COVID-19, Classification, Machine Learning, Gray-Level Co-occurrence Matrix GLCM, Feature Selection

تصنيف مرض كوفيد-19 بناءً على اختيار ميزات Extra Tree

اسراء محسن محمد [1]*, حسين عطية لفتة [1] , يسرى حسين علي [2]
[1] قسم علوم الحاسوب ، كلية العلوم للبنات ، جامعة بابل ، بابل ، العراق
[2] قسم علوم الحاسوب ، الجامعة التكنولوجية ، بغداد ، العراق

الخلاصة

صنفت منظمة الصحة العالمية (WHO) الفيروس التاجي على أنه حالة طوارئ صحية عالمية. ثبت أن الأشعة السينية للصدر مفيدة في كل من تشخيص ومراقبة أمراض الرئة المختلفة ، بما في ذلك COVID-19. في هذه الدراسة ، يتم توفير إطار عمل لاكتشاف مرض COVID-19 استنادًا إلى الأساليب المستعملة في التعلم الآلي وخوارزمية Extra Tree لتقليل الميزات المستخرجة من الصور. يتم استخراج مجموعة الميزات من الصور باستعمال خوارزمية GLCM وبناءً على خوارزمية Extra Tree لتحديد ميزات العمل ، ثم يتم إدخالها في خوارزمية XGBoost للتصنيف. تم تقييم النظام المقترح باستعمال مجموعتين مختلفتين من قواعد البيانات: قاعدة البيانات الكبيرة مع 9544 صورة وقاعدة البيانات الصغيرة مع 800 صورة. تم ضبط

*Email: asraa.mohammad.gsci104@student.uobabylon.edu.iq

جميع أحجام الصور على 300 * 300 بكسل. حقق النظام المقترح درجة دقة تصنيف بلغت 90.04٪
باستعمال مجموعة البيانات الكبيرة و 99.37٪ عند استعمال المجموعة الصغيرة.

1. Introduction

According to the World Health Organization (WHO), the initial instances of COVID-19 were discovered in Wuhan City, which is located in China [1] [2]. Within a few weeks, the disease was classified as a pandemic due to its exponential spread throughout the nations, which reported a rise in the number of confirmed cases, which was then followed by deaths. According to the World Health Statistics 2020 report, the present climate of COVID-19 poses a threat to the world [3].

The COVID-19 infection can be diagnosed in many ways, including those based on nucleic acids and using polymerase chain reaction (PCR), next-generation sequencing, computed tomography (CT), and chest X-ray (CXR) [4]–[6]. In general, all these methods can be used. Patients may be required to undergo these pathological tests so that doctors can monitor how their organs are changing as a result of them. Some of the most common of these pathological tests are CT scans and chest X-rays [7], [8]. An experienced observer looks at these photographs and makes a diagnosis based on his or her knowledge of the subject matter after doing an in-depth content analysis. In addition, a normal result on a CT scan or CXR does not always indicate the absence of a COVID-19 infection in the patient. Because of this, it is necessary to have helpers working in the healthcare field to guarantee an accurate diagnosis. Based on the principles of artificial intelligence (AI), new methods have been presented for identifying COVID-19 [9], [10]. Deep learning (DL) and machine learning (ML) are two of these methods. These methods are supposed to give faster and more precise results.

Numerous scholars have developed the traditional ML and DL techniques as tools to assist medical professionals in making correct diagnoses [11]. Classification accuracy will suffer in proportion to the number of features extracted from an image. This is because the images contain features that are not correlated with one another, and as a result, they do not provide accurate information regarding whether the images depict a positive or negative diagnostic situation. The optimization of the features is one of the potential solutions to this problem [12].

Feature selection classifiers are algorithms that select a subset of relevant features from a larger set of features to use in building a machine-learning model. They include filter methods, wrapper methods, embedded methods, dimensionality reduction methods, and hybrid methods [13]. Filter methods evaluate the relevance of each feature individually based on statistical measures like correlation or mutual information with the target variable [14]. Wrapper methods select features by using a specific machine learning algorithm to evaluate the performance of a subset of features. Embedded methods combine feature selection with the model training process [15]. The feature selection process is integrated with the model training process to optimize the feature subset. Dimensionality reduction methods reduce the number of features by transforming the original feature space into a lower-dimensional space [16]. Finally, hybrid methods combine multiple feature selection classifiers to select the most relevant features [17].

The classification algorithms KNN and XGBoost use wrapper techniques to choose the most useful features, which they then use to create a potent classifier that accurately detects COVID-19.

The primary objective of this case study is to categorize a collection of chest X-ray images obtained from Kaggle.com by making use of very helpful traditional machine learning techniques for predicting the COVID-19 virus, and then to evaluate the outcomes of these techniques in terms of their accuracy, sensitivity, specificity, and F1-score. The implementation of these technological advances is significantly supporting the identification, prediction, and prevention of diseases caused by the coronavirus. Fortunately, one of the most well-known applications of artificial intelligence, i.e., machine learning, has been significantly applied to a variety of COVID-19 datasets in several publications.

The key contributions of this paper are: proposing a COVID-19 classification approach based on the properties of original moment features and feature selection techniques; Creating a new group of descriptions called GLCM to extract features from the COVID-19 images; Using the results of the two COVID-19 X-ray datasets that were used and explained in this paper in Section (3.1), evaluate the performance of the proposed model.

The rest of the paper is organized as follows: Section 2 covers a summary of some of the previous studies' work. Section 3 explains the proposed methodology. Section 4 presents the machine learning results. The final section discusses conclusions and future work for improvements.

2. Related Work

In this section, some of the previous work of a number of researchers who detected COVID-19 disease using various methods is reviewed, including:

Imad et al. [18] have developed a method for detecting COVID-19 through the analysis of chest X-ray images. The suggested method analyzes images and pulls out features from them using HOG. These features are then categorized using machine learning techniques like support vector machines (SVM), K-nearest neighbors, random forests, Naive Bayes algorithms, and decision trees. The findings were best when using the SVM method, which had an accuracy rate of 96%. Sethy et al. [19] An approach suggested, which uses GLCM to extract characteristics and the SVM algorithm to classify 381 images of COVID-19 disease, achieved an accuracy of 93.2%. They also showed off some other feature-extracting strategies, such as HOG+SVM (accuracy: 88.5%), LBP+SVM (accuracy: 93.4%), and ResNet50+SVM (accuracy: 95.33%). Minaee et al. [20] developed an X-ray image classification system based on machine learning in order to locate and identify COVID-19. X-ray images may be categorized according to feature depth using supported vector machines. The accuracy of the suggested classification model, which included ResNet-50 for feature extraction plus SVM, was 95%. Ismael et al. [21] proposed a system based on SVM machine learning. They relied on CNN to extract features and used SVM to classify COVID-19. They used 380 X-ray scans in their study (180 were classified as COVID-19, and 200 were normal classifiers). Their highest accuracy, using the ResNet50 model and SVM, was 94.7%. Mijwil et al. [22] propose using a set of deep and machine learning algorithms to classify COVID-19. The researcher used 1,400 X-ray scans in his study and confirmed through his experiments that the SVM algorithm obtained the highest accuracy of 91.8% and had good performance compared to the other algorithms. Abdulkareem et al. [5] have come up with a plan for a clinical decision support system that uses machine learning and Internet of Things (IOT) devices to find COVID-19 patients. A set of algorithms for machine learning was used (naive Bayes (NB), Random Forest (RF), and support vector machines (SVM) were trained and tested on the basis of laboratory datasets). The results showed that the SVM algorithm was better at detecting COVID-19 disease, with an accuracy of up to 95%. Mahdy et al. [23], This paper recommends a deep learning-based methodology for detecting COVID-19-infected

patients using X-ray images. The support vector machine (SVM) is utilized to classify the affected X-ray images, providing deep features for accurate detection. They achieved a 97.48% success rate with their suggested system.

As a result of the public's interest in this subject, scientists are working hard to develop a reliable diagnostic for COVID-19. All prior studies discussing this subject have done so from a medical perspective, and some have used the symptoms seen in lung CT scans to classify cases of COVID-19 using machine learning. In this study, a COVID-19 disease detection framework is provided based on the methods used in machine learning and the Extra Tree algorithm to reduce the features extracted from the images. Using the Gray-Level Co-occurrence Matrix (GLCM) algorithm and the Extra Tree algorithm to choose the features of the work, a set of features is extracted from the images. This set of features is then put into the XGBoost algorithm to be classified.

3. Methodology

This study aims to develop and validate a model that can effectively and precisely detect COVID-19 disease. This section describes the data set and feature selection, as well as the technical and execution aspects of the model that was proposed.

3.1. Dataset

In most instances, using the right data set is crucial when assessing COVID-19 detection systems. The following is a description of the two different datasets that were used to train the categorization model:

The first dataset (moderate) consists of 400 chest X-rays with a confirmed COVID-19 infection and 400 chest X-rays that are not infected. Both image collections were obtained from [24]. The images in the dataset are grayscale in PNG file format.

The large dataset consists of 5500 normal chest X-rays and 4044 confirmed cases of COVID-19 infection derived from [25]. The images are grayscale in JPEG, JPG, and PNG file formats.

3.2. Pre-Processing

During this stage of processing, the images are converted to grayscale, and their dimensions are modified to be 300 x 300 pixels each.

3.3. Feature Extraction

GLCM (Gray-Level Co-occurrence Matrix) is a method used to extract second-degree statistical characteristics from images. It involves analyzing the relationships between the pixel values at different angles within the image. By examining these relationships, we can obtain valuable information about the texture and patterns present in the image, which is useful for various image analysis tasks such as feature extraction and classification.

$$P = [p(i, j|d, \theta)] \quad (1)$$

The co-occurrence matrix is derived from an I image. At this stage, the co-occurrence matrix is utilized to compare the (i) pixel frequency feature with the (j) neighbor pixel frequency feature, taking the angle (θ) direction and d length into consideration [26]. In this paper, the angle [0] is measured, and GLCM is used to determine the dissimilarity, correlation, homogeneity, contrast, energy, and angular second moment (ASM). The GLCM approach generates (1*6) attributes [27].

The *dissimilarity* feature in a gray-level co-occurrence matrix (GLCM) is a measure of the difference between the gray-level values of a pixel and its neighbors. It reflects the variations in gray-level intensity and provides information about the texture of an image. In mathematical terms, the dissimilarity feature is defined as the sum of the absolute differences between the gray-level values of the pixel and its neighbors. The dissimilarity formula is expressed as follows:

$$Dissimilarity = \sum_i \sum_j |i-j| P(i,j) \quad (2)$$

The *correlation* feature in a gray-level co-occurrence matrix (GLCM) is a measure of the relationship between the gray-level values of a pixel and its neighbors. It reflects the linear relationship between the gray-level values of a pixel and its neighbors and provides information about the texture of an image. The correlation formula is as follows:

$$Correlation = \sum_i \sum_j \frac{(i-\mu_i)(j-\mu_j)P(i,j)}{\sigma_i \sigma_j} \quad (3)$$

The *energy* feature in a gray-level co-occurrence matrix (GLCM) is a measure of the texture homogeneity of an image. It reflects the uniformity of the gray-level values in an image and provides information about the texture of the image. The energy formula is given as follows:

$$Energy = \sum_{i,j} P(i,j)^2 \quad (4)$$

The *homogeneity* feature in a gray-level co-occurrence matrix (GLCM) is a measure of the texture uniformity of an image. It reflects the uniformity of the gray-level values in an image and provides information about the texture of the image. The homogeneity formula is given as follows:

$$Homogeneity = \sum_i \sum_j \frac{1}{1+|i-j|^2} P(i,j) \quad (5)$$

The *contrast* feature in a gray-level co-occurrence matrix (GLCM) is a measure of the difference between the gray-level values of a pixel and its neighboring pixels. It reflects the variations in gray-level intensity and provides information about the texture of an image.
In mathematical terms, the contrast feature is defined as the sum of the squared differences between the gray-level values of the pixel and its neighbors. The contrast formula is given as follows:

$$Contrast = \sum (i-j)^2 P(i,j) \quad (6)$$

The *angular second moment* (ASM) feature in a gray-level co-occurrence matrix (GLCM) is a measure of the homogeneity of an image. It reflects the uniformity of the gray-level values in an image and provides information about the texture of the image. The ASM formula is given as follows:

$$ASM = P(i,j)^2 \quad (7)$$

where i and j are the gray-level values of the pixel and its neighbor, respectively, and P(i,j) is the joint probability of the pixel and its neighbor having the gray-level values i and j, respectively [27].

3.4. Feature Selection

The primary purpose of feature selection is to focus on the most informative aspects of a dataset that can be used to accurately predict a dependent variable [28]. To do this, Extra

Trees builds numerous decision trees and uses an average feature importance score to identify which features are more significant.

The extra trees are a useful ensemble learning technique for classification issues that may be used for the task of selecting relevant features. It is superior to both classic decision trees and random forests due to its focus on fixing the issues of overfitting in decision trees and the random nature of feature selection in the latter [29].

Overfitting is prevented, and the variability of feature significance scores is minimized by randomly picking subsets of features at each split throughout the tree-building process. Finally, the average feature significance scores over all trees are used to get the final feature importance scores, with the most essential features being those that regularly receive high scores.

Algorithm 1 Extra Tree Feature Selection

Input:
- X: Training dataset features
- Y: Target variable
- n_trees: Number of trees to be used in the Extra Trees model
- m_features: Number of randomly selected features at each split
- scoring_metric: Scikit-learn metric to evaluate feature importance

Output: - Sorted list of feature importance

Begin

1. feature_importances ← an empty list
2. For i in 1 to n_trees:
   a. random_features ← select m_features features from X without replacement
   b. model ← an Extra Trees classifier with default parameters
   c. model.fit(X[:,random_features], y)
   d. importance_scores ← compute feature importance for each feature based on the chosen scoring metric
   e. Append importance_scores to feature_importances
3. mean_importances ← compute mean of feature importance across all trees
   std_importances ← compute standard deviation of feature importance across all trees
   feature_importances ← [(mean_importances[i], std_importances[i]) for i in range(X.shape[1])]
4. sorted_features ← sort feature_importances in descending order based on mean_importances
5. Return sorted_features

End

3.5. Machine Learning

3.5.1. k-NN (k-nearest neighbors' algorithm)

KNN is a simple and effective classification algorithm that sorts items into groups based on their proximity to a given object [30]. The implementation involves the following steps:

- Parameter: 'k' represents the number of nearest neighbors to consider during classification. It is essential to choose an appropriate value for 'k' to ensure accurate predictions.
- Distance Metric: The Euclidean distance is commonly used to measure the proximity between data points in feature space.
- Classification: To classify a new data point, the algorithm finds the "k" closest data points from the training set and assigns the class label based on the majority class among these neighbors [31].

Algorithm 2 pseudocode for the k-NN algorithm

Input: train data T, test data x, label data C, K
Output: class for test C_x

Begin
For each x do
Calculate distance between T and x: D $(T, x) = \sqrt{(T_1 - x_1)^2 - (T_2 - x_2)^2}$

End for
Arrange the values in descending order
Select the top K value
Classify x
End

3.5.2. XGBoost

XGboost is a powerful gradient boosting algorithm known for its superior performance and fast execution speed [30]. The implementation involves the following aspects:
- Parameters: XGboost offers a wide range of parameters to tune for optimal performance. Some of the key parameters include the number of boosting rounds (n_estimators), maximum tree depth (max_depth), learning rate (eta), and subsample ratio (subsample).
- Parallel Computation: XGboost employs parallel computation to build trees across all CPUs during training, which significantly speeds up the process.
- Tree Pruning: Instead of traditional stopping criteria, XGboost uses the "max depth" parameter for tree pruning, which contributes to improved model generalization [32].

Algorithm 3 pseudocode for XGBoost algorithm

Input:
- Training dataset {(x1, y1), (x2, y2), ..., (xn, yn)}
- Number of trees T
- Maximum depth of each tree d

Output:
- XGBoost model

Begin
1. Initialize the F_0(x) to be the mean of the training labels y.
2. For t in 1 to T:
   a. Compute the negative gradient of the loss function w.r.t. F_{t-1}(x), denoted by r_{it} = -[∂L(yi, F_{t-1}(xi)) / ∂F_{t-1}(xi)]
   b. Fit a regression tree with maximum depth d to the negative gradients r_{it}, i.e., the tree will minimize the objective function J = ∑(r_{it} - F_t(xi))^2
   c. Compute the tree weights w_t by minimizing the objective function J = ∑(r_{it} - w_t * h_t(xi))^2
   d. Update the prediction F_t(x) = F_{t-1}(x) + η * w_t * h_t(x), where η is the learning rate.
4. Return the final prediction F_T(x).
End

Table 1 provides the parameter values used in the two classification methods. During the classification process using KNN, the algorithm considers the three nearest neighbors to the data point being classified to make the decision. The XGBoost Classifier indicates that 80%

of the available data is used for training the model, while the remaining 20% is used for testing the model's performance. This division helps assess the model's ability to generalize to new, unseen data.

Table 1: Parameter values for used methods

Methods	Parameters Values
XGboost classifier	Training set:80% Testing set: 20 %
KNN classifier	Number of neighbors:3

3.6. Proposed System

The general structure of the proposed system and its function are divided into phases corresponding to various work stages, as depicted in Figure 1.

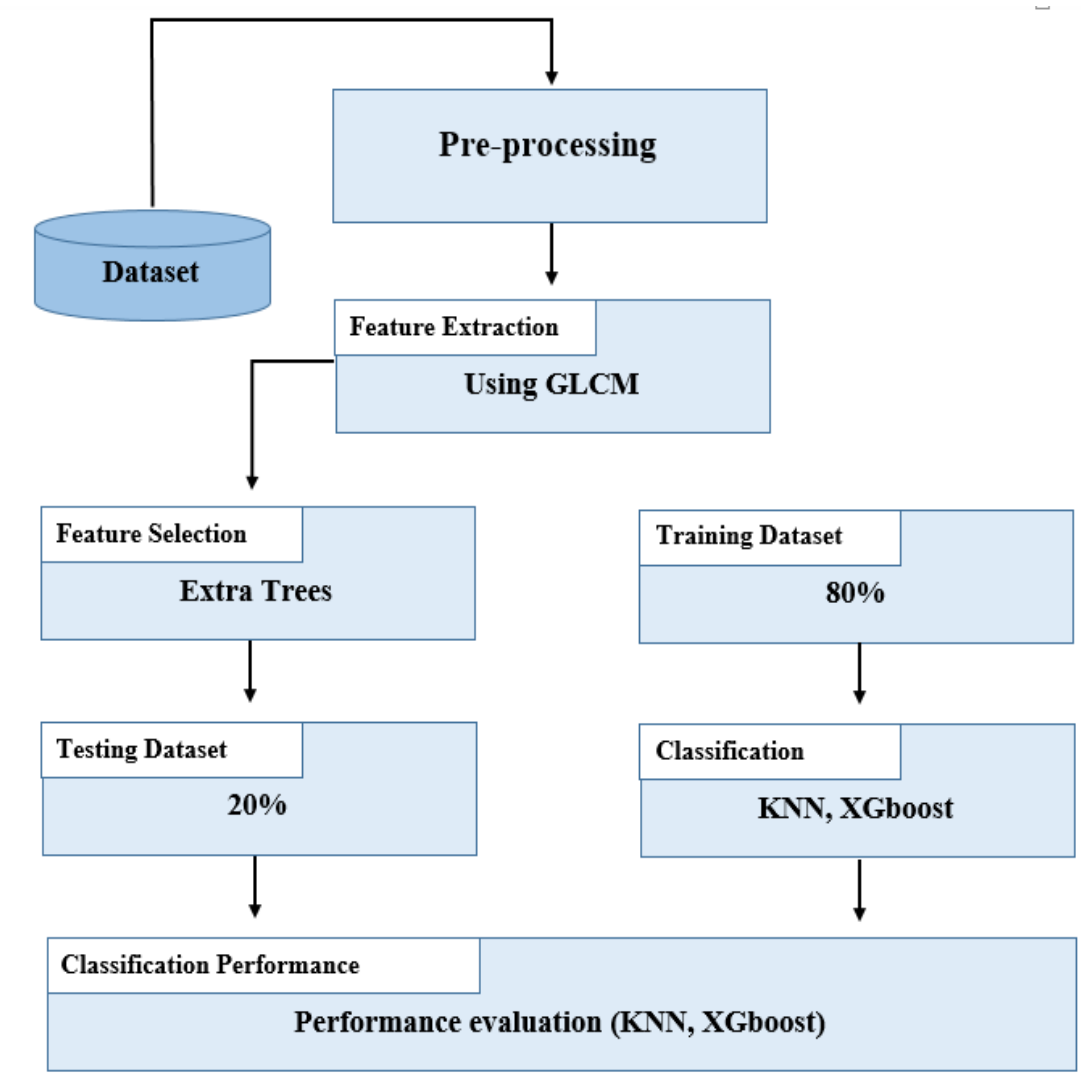

Figure 1: Proposed system structure

As shown in Figure 1, the proposed system consists of the following:

i. Pre-processing: this is the starting point for changing the size of the images and preparing them for feature extraction.

ii. Feature extraction: refers to extracting features from images, which is considered the most important step because it is used in the classification process.
iii. Feature selection: refers to selecting the best and most relevant feature to obtain a strong classifier and reduce time and effort during the classification process.

3.7. Performance Evaluation

A confusion matrix is a table that is utilized in the process of assessing the effectiveness of a classifier. It shows the number of accurate positive results, accurate negative results, incorrect positive results, and false negative results that a classification algorithm produced. The matrix offers a summary of the right and wrong predictions generated by the classifier, and it is frequently utilized to evaluate the performance of the classifier in terms of accuracy, recall, precision, and F1 score.

The confusion matrix is organized in such a manner that each column in the matrix represents a predicted class, while each row represents an actual class. The entries in the matrix reveal the number of occurrences from the actual class that were properly or wrongly identified as belonging to a certain predicted class. This indicates how well or inaccurately the matrix was able to predict the real class.

A true positive, sometimes abbreviated as TP, is an incident that was appropriately labeled as positive. A true negative, sometimes abbreviated as TN, refers to an occurrence that was appropriately labeled as negative. A situation in which a positive result was wrongly assigned to it is referred to as a "false positive" (FP). A situation that was wrongly labeled as negative is an example of a false negative, abbreviated as FN.

$$ACC = \frac{(TN+TP)}{(TN+TP+FN+FP)} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 - Score = 2\left(\frac{Precision * Recall}{Precision + Recall}\right) \quad (4)$$

4. Results And Discussion

Both the small database with only 800 images and the big database with 9544 X-rays were utilized in this study to determine whether or not a person was infected with COVID-19. Python programming was used on the calculator to construct the model (an HP laptop with a Core i5 processor and 4 GB of RAM to perform experiments). On each of the two datasets, training and testing were performed on two distinct types of algorithms (k-NN and XGBoost), with 20% for testing and 80% for training. Once the image size has been modified and the images have been converted to grayscale, the GLCM technique is used to extract properties from the images. Following that, a classification procedure was carried out on the data without first extracting the relevant characteristics from the database to determine the degree of accuracy before this stage, as demonstrated in Table 2.

The findings of the algorithms, together with an analysis of how accurately they classified the two databases, are presented in Table 2. In comparison to the KNN algorithm, the accuracy of XGBoost was significantly higher.

Table 2: Results without feature selection

	Algo.	Precision	Recall	F1-score	ACC	TN	TP	FN	FP
Small data	XGBoost	0.99	0.99	0.99	0.9937	88	71	1	0
	k-NN	0.98	0.97	0.97	0.975	74	82	1	3
Big data	XGBoost	0.94	0.93	0.93	0.9041	233	710	55	45
	k-NN	0.96	0.91	0.93	0.904	238	705	73	27

To evaluate how well the model can detect COVID-19 disease, the results were analyzed and displayed using the confusion matrix that is depicted in Figure 2.

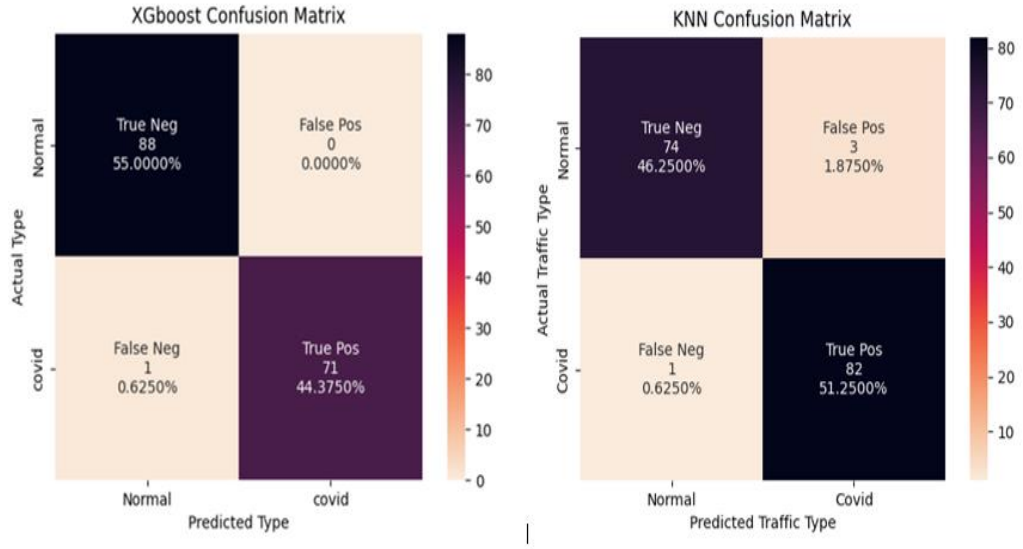

Figure 2: Confusion matrix

The next step required to be taken in order to implement the described system is to choose the qualities that would make the best classifiers. In the experiments that were conducted, the Extra Trees algorithm was utilized, and the fifth attribute, ASM, was disregarded in both databases that were utilized, as shown in Figure 3.

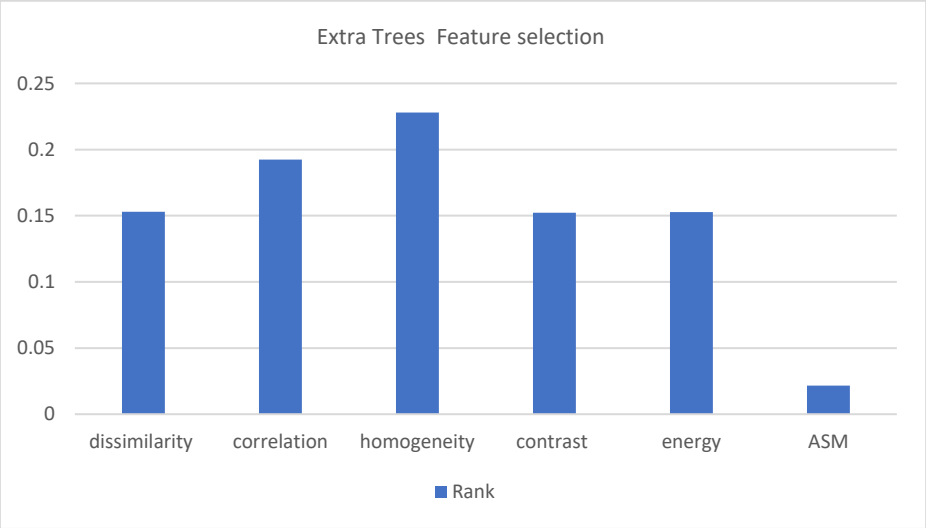

Figure 3: Feature selection ranking

After obtaining these results, the second experiment was conducted with feature selection applied before running the classification algorithms (Table 3). The purpose of feature selection is to choose the most relevant characteristics from the database, aiming to improve classification performance.

The second approach with feature selection (Table 3) yielded slightly better results for both XGBoost and k-NN algorithms on the big dataset compared to the first approach without feature selection (Table 2). The results were similar for XGBoost on the small dataset, but k-NN showed slightly better performance after feature selection.

Table 3: Results after feature selection

	Algo.	**Precision**	**Recall**	**F1-score**	**ACC**	**TN**	**TP**	**FN**	**FP**
Small data	XGBoost	0.99	0.99	0.99	0.9937	88	71	1	0
	k-NN	0.98	0.98	0.98	0.981	76	82	1	1
Big data	XGBoost	0.95	0.93	0.94	0.9104	233	710	55	45
	k-NN	0.95	0.93	0.94	0.9108	261	686	54	93

The outcomes of the suggested approach were also compared with those of earlier studies that selected a fixed number of images (Table 4).

The comparison table highlights the performance of the proposed system in relation to earlier studies using different feature extraction methods and algorithms. The proposed approach shows impressive accuracy results, outperforming the previous methods on the dataset containing 800 images and achieving an accuracy of 0.9937. This indicates the superiority of the proposed system in image classification compared to the methods tested in the previous studies.

Table 4: A comparison of the results

Researcher	**Feature extraction method**	**Algo.**	**No.image**	**ACC**
[19]	GLCM	SVM	386	0.934
	HOG			0.885
	LBP			0.934
	ResNet50			0.953
[21]	Resnet50	SVM	380	0.947
[22]	-	SVM	1400	0.918
[23]	multi-level thresholding	SVM	40	0.974
proposed approach	GLCM	XGBoost	800	0.9937

Conclusion

In the current investigation, X-rays were used in two different ways to identify and detect the COVID-19 disease. The method that was developed relied on the characteristics that were derived from images based on GLCM. After that, the extra trees were used to extract the characteristics that were deemed to be the most important. The first method, which utilized the XGBoost algorithm, fared better than the second method, which utilized the KNN algorithm. The XGBoost method has an accuracy of 0.9937 for small data and 0.9104 for large data. Future work will include increasing the number of images captured, trying to

collect as many as possible, and developing the proposed system to be a real-time application that can be used in e-medicine applications.

References

[1] Y. R. Guo *et al.*, "The origin, transmission and clinical therapies on coronavirus disease 2019 (COVID-19) outbreak- A n update on the status," *Mil. Med. Res.*, vol. 7, no. 1, pp. 1–10, 2020.

[2] H. A. Sameer, S. K. Gharghan, and A. H. Mutlag, "Hybridization of particle swarm optimization algorithm with neural network for COVID-19 using computerized tomography scan and clinical parameters," *J. Eng.*, vol. 2023, no. 2, p. e12226, 2023.

[3] C. S. Benson *et al.*, "The effect of iron deficiency and anemia on women's health," *Anesthesia*, vol. 76, no. S4, pp. 84–95, 2021.

[4] M. A. Elaziz, K. M. Hosny, A. Salah, M. M. Darwish, S. Lu, and A. T. Sahlol, "New machine learning method for image based diagnosis of COVID-19," *PLoS One*, vol. 15, no. 6, pp. 1–18, 2020.

[5] K. H. Abdulkareem *et al.*, "Realizing an Effective COVID-19 Diagnosis System Based on Machine Learning and IoT in Smart Hospital Environment," *IEEE Internet Things J.*, vol. 8, no. 21, pp. 15919–15928, 2021.

[6] Z. A. A. Alyasseri *et al.*, "Review on COVID-19 diagnosis models based on machine learning and deep learning approaches," *Expert Syst.*, vol. 39, no. 3, pp. 1–32, 2022.

[7] A. Ahmad, S. Garhwal, S. K. Ray, G. Kumar, S. J. Malebary, and O. M. Barukab, "The Number of Confirmed Cases of Covid-19 by using Machine Learning: Methods and Challenges," *Arch. Comput. Methods Eng.*, vol. 28, no. 4, pp. 2645–2653, 2021.

[8] Y. Pan *et al.*, "Initial CT findings and temporal changes in patients with the novel coronavirus pneumonia (2019-nCoV): a study of 63 patients in Wuhan, China," *Eur. Radiol.*, vol. 30, no. 6, pp. 3306–3309, 2020.

[9] G. Pinter, I. Felde, A. Mosavi, P. Ghamisi, and R. Gloaguen, "COVID-19 pandemic prediction for Hungary; A hybrid machine learning approach," *Mathematics*, vol. 8, no. 6, pp. 1–20, 2020.

[10] M. Jamshidi *et al.*, "Artificial Intelligence and COVID-19: Deep Learning Approaches for Diagnosis and Treatment," *IEEE Access*, vol. 8, no. December 2019, pp. 109581–109595, 2020.

[11] M. Barstuğan, U. Özkaya, and Ş. Öztürk, "Coronavirus (Covid-19) classification using CT images by machine learning methods," *CEUR Workshop Proc.*, vol. 2872, no. 5, pp. 29–35, 2021.

[12] V. Ravi, H. Narasimhan, C. Chakraborty, and T. D. Pham, "Deep learning-based meta-classifier approach for COVID-19 classification using CT scan and chest X-ray images," in *Multimedia Systems*, vol. 28, pp. 1401–1415, 2022.

[13] J. Lee, W. Seo, H. Han, and D. Kim, "Evolutionary Multilabel Feature Selection Using Promising Feature Subset Generation," *J. Sensors*, vol. 2018, p. 3419213, 2018.

[14] M. Ghosh and G. Sanyal, "Performance Assessment of Multiple Classifiers Based on Ensemble Feature Selection Scheme for Sentiment Analysis," *Applied Computational Intelligence and Soft Computing*, vol. 2018, p. 8909357, 2018.

[15] Y. Mao and Y. Yang, "A Wrapper Feature Subset Selection Method Based on Randomized Search and Multilayer Structure," *Biomed Res. Int.*, vol. 2019, p. 9864213, 2019.

[16] H. B. Bisheh, G. G. Amiri, and E. Darvishan, "Ensemble Classifiers and Feature-Based Methods for Structural Damage Assessment," *Shock Vib.*, vol. 2020, p. 8899487, 2020.

[17] D. Endalie and G. Haile, "Hybrid Feature Selection for Amharic News Document Classification," *Math. Probl. Eng.*, vol. 2021, p. 5516262, 2021.

[18] M. Imad, N. Khan, F. Ullah, M. A. Hassan, A. Hussain, and Faiza, "COVID-19 Classification based on Chest X-Ray Images Using Machine Learning Techniques," *J. Comput. Sci. Technol. Stud.*, vol. 2, no. 2, pp. 01–11, 2020.

[19] P. K. Sethy, S. K. Behera, P. K. Ratha, and P. Biswas, "Detection of coronavirus disease (COVID-19) based on deep features and support vector machine," *Int. J. Math. Eng. Manag. Sci.*, vol. 5, no. 4, pp. 643–651, 2020.

[20] S. Minaee, R. Kafieh, M. Sonka, S. Yazdani, and G. Jamalipour Soufi, "Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning," *Med. Image Anal.*, vol. 65, p. 101794, 2020.

[21] A. M. Ismael and A. Şengür, "Deep learning approaches for COVID-19 detection based on chest X-ray images," *Expert Syst. Appl.*, vol. 164, p. 114054, 2021.

[22] M. M. Mijwil, "Implementation of Machine Learning Techniques for the Classification of Lung X-Ray Images Used to Detect COVID-19 in Humans," *Iraqi J. Sci.*, vol. 62, no. 6, pp. 2099–2109, 2021.

[23] L. N. Mahdy, K. A. Ezzat, H. H. Elmousalami, H. A. Ella, and A. E. Hassanien, "Automatic X-ray COVID-19 Lung Image Classification System based on Multi-Level Thresholding and Support Vector Machine," *medRxiv*, vol. 2020, p. 215782177, 2020.

[24] Kaggle, "Chest X-Ray Images (Pneumonia) | Kaggle," *Kaggle's chest X-ray images (Pneumonia) dataset*, 2020. https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia

[25] W. El-Shafai and F. E. Abd El-Samie, "Extensive COVID-19 X-Ray and CT Chest Images Dataset," *Mendeley Data*, 2020, [Online]. Available: https://data.mendeley.com/datasets/8h65ywd2jr/3

[26] S. Bakheet and A. Al-Hamadi, "Automatic detection of COVID-19 using pruned GLCM-Based texture features and LDCRF classification," *Comput. Biol. Med.*, vol. 137, no. June, p. 10, 2021.

[27] P. K. Mall, P. K. Singh, and D. Yadav, "GLCM Based Feature Extraction and Medical X- RAY Image Classification using Machine Learning Techniques," *2019 IEEE Conf. Inf. Commun. Technol.*, vol. 19533775, no. April, pp. 1–6, 2019.

[28] V. Bolón-canedo and A. Alonso-Betanzos, "Ensembles for feature selection : A review and future trends," *Inf. Fusion*, vol. 52, no. May 2018, pp. 1–12, 2019.

[29] E. Nasarian *et al.*, "Association between work-related features and coronary artery disease : A heterogeneous hybrid feature selection integrated with balancing approach," *Pattern Recognit. Lett.*, vol. 133, no. February, pp. 33–40, 2020.

[30] N. Rastin, M. Z. Jahromi, and M. Taheri, "A generalized weighted distance k-Nearest Neighbor for multi-label problems," *Pattern Recognit.*, vol. 114, p. 107526, Jun. 2021.

[31] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A brief review of nearest neighbor algorithm for learning and classification," *2019 Int. Conf. Intell. Comput. Control Syst. ICCS 2019*, no. January, pp. 1255–1260, 2019.

[32] A. Konstantinov, L. Utkin, and V. Muliukha, "Gradient boosting machine with partially randomized decision trees," *Conf. Open Innov. Assoc. Fruct*, vol. 2021-Janua, no. January, pp. 1–14, 2021.