Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi
Dokuz Eylul University Faculty of Engineering Journal of Science and Engineering
Elektronik/Online ISSN: 2547-958X

RESEARCH ARTICLE / ARAŞTIRMA MAKALESI

Comparison of Hybrid Models with Multi-Feature Fusion Using Covid-19 Radiography Database

Covid-19 Radyografi Veritabanı Kullanılarak Çok Öznitelikli Füzyona Sahip Hibrit Modellerin Karşılaştırılması

Fatma Günseli Yaşar Çıklaçandır [1*], Gözde Ulutagay [2]

[1] İzmir Katip Çelebi Üniversitesi, Mühendislik ve Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, İzmir, TÜRKİYE
[2] Ege Üniversitesi, Fen Fakültesi, İstatistik Bölümü, İzmir, TÜRKİYE

Corresponding Author / Sorumlu Yazar *: fatma.gunseli.yasar@gmail.com

Abstract

COVID-19, which emerged in 2019 and was subsequently classified as a pandemic, has affected millions of individuals worldwide. Different variations of the illness continue to persist, even though it may seem to have subsided at the moment. Hence, it remains essential to promptly and precisely diagnose COVID-19. Chest imaging has been proven to clearly demonstrate COVID-19 infection even in the early stages of the disease, assisting physicians and radiologists in making quicker and more accurate judgements. This study proposes a hybrid model with feature fusion based on Convolutional Neural Network based models and classifiers to accurately distinguish infected patients from healthy people. The extracted features from two different Convolutional Neural Network based models are concatenated, or added before feature selection. On a publicly accessible radiography database containing 21168 images of the four classes (Covid, Lung_Opacity, Normal, and Viral Pneumonia), extensive tests utilizing five fold cross-validation have been conducted. According to the tests, an accuracy rate of about 96% has been obtained. The findings also demonstrate that the proposed approach can contribute significantly to the rapidly expanding workload in health-care systems.

Keywords: *Feature Extraction, Feature Fusion, Deep Learning*

Öz

COVID-19, dünya çapında milyonlarca kişiyi etkiledi. Şu anda azalmış gibi görünse de hastalığın farklı varyasyonları devam ediyor. Bu nedenle, COVID-19'u hızlı ve kesin bir şekilde teşhis etmek hala hayati önem taşıyor. Göğüs görüntülemenin, hastalığın erken evrelerinde bile COVID-19 enfeksiyonunu açıkça gösterdiği, doktorların ve radyologların daha hızlı ve daha doğru kararlar almasına yardımcı olduğu kanıtlanmıştır. Bu çalışma, enfekte hastaları sağlıklı insanlardan doğru bir şekilde ayırt etmek için Evrişimsel Sinir Ağı tabanlı modellere ve sınıflandırıcılara dayalı özellik füzyonuna sahip hibrit bir model önermektedir. İki farklı Evrişimsel Sinir Ağı tabanlı modelden çıkarılan özellikler birleştirilir veya özellik seçiminden önce eklenir. Dört sınıftan (Covid, Lung_Opacity, Normal ve Viral Pneumonia) 21.168 görüntüyü içeren, kamuya açık bir radyografi veritabanında, beş kat çapraz doğrulamayı kullanan kapsamlı testler yapılmıştır. Yapılan testlere göre yaklaşık %96 oranında doğruluk oranı elde edildi. Bulgular ayrıca önerilen yaklaşımın sağlık sistemlerinde hızla artan iş yüküne önemli ölçüde katkıda bulunabileceğini göstermektedir.

Anahtar Kelimeler: *Öznitelik Çıkarma, Öznitelik Füzyonu, Derin Öğrenme*

1. Introduction

As of February 25, 2024, 7,035,337 fatalities and 774,771,942 confirmed cases of COVID-19 had been reported to WHO (World Health Organization) globally. There have been 279,075,579 cases registered only in Europe. Western Pacific (208,245,005) and America (193,196,825) are the places with the highest number of cases after Europe. Among COVID-19 patients, fever, coughing, exhaustion, dyspnea, and sputum are the most known symptoms [1]. The fast increase in the number of illnesses caused by the epidemic exceeded the capacity of hospitals in countries, both in terms of equipment and personnel, putting the entire healthcare system in peril [2]. As a result, it was critical to be able to detect and isolate affected people in a timely, cost-effective, and dependable manner.

Artificial intelligence has made significant strides in various medical fields in recent years, particularly in the realm of medical imaging [3-7]. The use of radiological imaging technologies has become increasingly prominent, enabling the rapid identification and isolation of infected individuals. Furthermore, these technologies enhance the precise differentiation of various pneumonia types [8]. Based on the findings, several studies have been done to use image processing and artificial intelligence approaches to autonomously diagnose tomography and x-ray images [9-16]. It is seen that many studies in the literature use deep learning architectures. Studies in the literature will be discussed in more detail in Section 2. The point to be mentioned in this section is that in some of the studies using deep learning architectures, deep learning is used only in the feature extraction step while other methods such as support vector machine (SVM) and k nearest neighbor (KNN) are used in classification [17-18].

DOI: https://doi.org/10.21205/deufmd.2025278020

Geliş Tarihi / Received: 15.09.2024
Kabul Tarihi / Accepted: 13.11.2024

These studies are within the scope of hybrid models. The reason why hybrid models are preferred is that although deep learning architectures work well for feature extraction, they take a very long time to complete classification process [19-20].

In this study, which integrates hybrid models with feature fusion, we conducted a comprehensive comparison of various fusion techniques, feature quantities, and classifiers. This paper's main contributions are as follows:

• Introducing novel models aimed at alleviating the burdens on hospitals induced by the COVID-19 pandemic and aiding clinicians in enhancing the precision of COVID-19 detection through X-ray images.

• Enhancing hybrid models through the incorporation of feature fusion techniques.

• Developing models that offer faster results compared to traditional Convolutional Neural Network (CNN) based classification methods.

The remainder of the study is structured as follows: The literature review is described in Section 2. The materials and methods utilized in this paper are described in Section 3, and the results analysis is covered in Section 4. There is a comparison with the studies in the literature in Section 5. Finally, the conclusions are mentioned in Section 6.

2. Literature Review

A deep learning based model based on MobileNetV2 were suggested by Kaya and Gürsoy [21] to detect COVID-19 infection using a dataset of 1,576 normal, 3,616 COVID-19, and 4,265 pneumonia X-ray images. The accuracy rate of the proposed method is 97.61%.

A CNN approach called LW-CORONet was presented by Nayak et al. and tested on two datasets [22]. 2,250 images made up dataset-1 (750 of pneumonia, 750 of normal, and 750 of COVID-19) used in the study, whereas 15,999 images made up dataset-2 (5,575 of pneumonia, 8,066 of normal, and 2,358 of COVID-19). The custom CNN was trained for 100 epochs, resulting in identification accuracies of 98.67% for dataset-1 and 95.67% for dataset-2 across three categories.

Pneumonia, COVID-19, and typical chest X-ray images were successfully classified from chest X-ray image dataset by Sanida et al.'s CAD system with a hybrid identification technique [23]. Outstanding outcomes were attained by the hybrid Deep Convolutional Neural Networks (DCNN) identification method, including a 99.25% accuracy rate.

In the study of Sanida et al. [24], a CNN-based model was proposed to detect COVID-19. The dataset used includes the samples of normal (10,192 images), COVID-19 (3,616 images), lung opacity (6,012 images), and viral pneumonia (1,345 images). The developed model achieved an classification accuracy rate of 95.80%.

In the study of Ayadi et al. [25], the COVID-AleXception model was introduced, which combines features from two pre-trained CNN-based models (Xception and AlexNet). The dataset used for this study consisted of 15,153 X-ray images, including 1,345 pneumonia, 3,616 COVID-19, and 10,192 normal images. Each CNN model underwent 100 epochs of training. Remarkably, the COVID-AleXception model achieved an impressive identification accuracy rate of 98.68%, surpassing the individual performances of Xception (95.63%) and AlexNet (94.86%).

A bespoke CNN prediction system for chest X-rays was created by Hafeez et al. and compared to two pre-trained CNN approaches, VGG16 and AlexNet [26]. The evaluation involved three categories: normal, COVID-19, and virus bacteria. Their model attained an accuracy rate of 89.855%, while VGG16 achieved 89.015%, and AlexNet achieved 89.155%.

Moving on to the study of Huang et al. [27], the authors proposed a novel CNN technique for detecting COVID-19 using X-ray images. They tested their approach with seven pre-trained CNN-based systems, including InceptionV3, Xception, ResNet50V2, MobileNetV2, DenseNet121, EfficientNet-B0, and EfficientNetV2. The dataset included the samples of COVID-19 (600 images), normal (600 images), and pneumonia (600 images). Each CNN model underwent 50 epochs of training. The proposed method achieved an accuracy rate of 98.33%, with EfficientNetV2 closely following at 97.73%.

Ghose et al. developed a customized CNN-based automatic diagnosis system using a dataset of 10,293 X-ray images, which included 4,200 pneumonia, 2,875 COVID-19, and 3,218 normal images [28]. Their custom CNN was trained for 25 epochs and they obtained an impressive 98.50% accuracy rate.

In the study of Ibrokhimov et al. [29], a DL diagnosis system was proposed for pneumonia detection using X-ray images. VGG19 and ResNet50 were compared for three lung diseases. The dataset included the samples of pneumonia (11,263 images), COVID-19 (11,956 images), and normal (10,701 images). Both CNN methods were trained for 180 epochs, with the VGG19 method achieving a 96.60% accuracy rate and ResNet50 achieving 95.80%.

In the study of Kong and Cheng [8], chest X-ray image classification was performed with an average accuracy of 98.0% for binary classification and 97.3% for three-category classification. The model utilized ResNet34 for efficient dataset segmentation and feature extraction and incorporated attention mechanisms to improve classification accuracy.

Ji et al. employed four different CNN-based models and two cascaded network models to classify X-ray samples into two categories: individuals with COVID-19 and healthy individuals [30]. Model 2 of the cascade network showed exceptional performance with an accuracy of 96%, significantly aiding in COVID-19 detection.

Narin et al. used chest X-rays to identify patients infected with coronavirus pneumonia by using five CNN-based models (ResNet50, ResNet101, ResNet152, InceptionV3, and Inception-ResNetV2) [31].

Hussain et al. [32] developed a new CNN based model, namely CoroDet. In the study where 2-class, 3-class and 4-class classification results are given, success rates of 99.1%, 94.2% and 91.2% are achieved, respectively.

DarkNet was used in the study of Ozturk et al. [33]. They obtained 98.08% accuracy rate for binary classification, and 87.02% accuracy rate for 3-class classification.

Shaban et al. [17] proposed a hybrid feature selection methodology combining fast selection stage (FSS) and accurate selection stage (ASS). FSS is based on filtering while ASS is based on genetic algorithm (GA). Additionally, they suggest enhanced KNN (EKNN) to overcome the trapping problem of KNN.

In the study of Khan et al. [34], CoroNet was introduced, based on the Xception method. It underwent 80 epochs of training and it achieved an accuracy rate of 89.60% in classifying images into the four classes.

Additionally, in the study of Khan and Aslam [35], four different CNN-based models (VGG16, ResNet50, DenseNet121, and

VGG19) were compared for diagnosing X-ray images as COVID-19 or normal (2020). The dataset consists of the samples for normal (802 images) and COVID-19 (790 images). Each CNN model underwent 30 epochs of training. The accuracy rates of VGG16, ResNet50, DenseNet121 and VGG19 are 99.33%, 97.00%, 96.66% and 96.66%, respectively.

Loey et al. used the GoogLeNet to classify the dataset consisting of the samples for COVID-19, bacterial pneumonia, viral pneumonia, and normal cases [18]. They achieved an accuracy rate of 80.6%.

These studies collectively showcase substantial progress in automated COVID-19 diagnosis using chest X-ray images, with various models consistently achieving high accuracy rates.

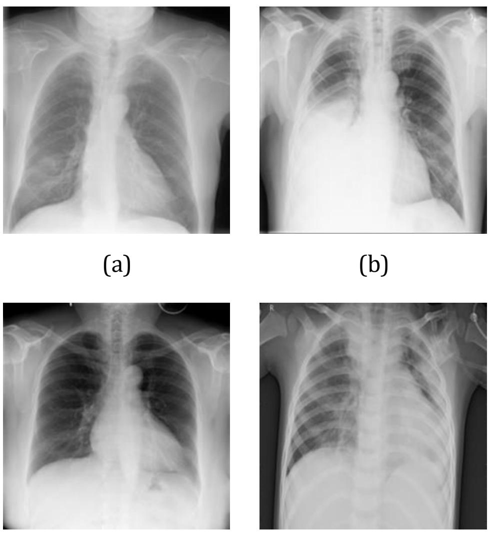
3. Materials and Methods

3.1. Dataset

The dataset 'COVID-19 Radiography Database' utilized in this study comprises of X-Ray images divided into four classes: COVID, Lung Opacity, Normal, and Viral Pneumonia (Table 1) [37]. It is available for download from Kaggle. There are 3619 images belonging to the 'Covid' class, 6012 images belonging to the 'Lung Opacity' class, 10192 images belonging to the 'Normal' class, and 1345 images belonging to the 'Viral Pneumonia' class in the data set. There are 21168 images in total. The samples from the dataset are depicted in Figure 1. These images are gray level and 299×299 sizes.

Table 1. The distribution of the classes in dataset.

Class	The Number of the Images
Covid	3,619
Lung Opacity	6,012
Normal	10,192
Viral Pneumonia	1,345
SUM	21,168



(a) (b) (c) (d)

Figure 1. Image samples from the dataset (a) Covid (b) Lung_Opacity (c) Normal (d) Viral Pneumonia.

3.2. Feature extraction

Deep learning is a type of machine learning that is multi-layered. It is utilized in a variety of applications. For feature extraction and conversion, it employs many layers of nonlinear processing units. Each succeeding layer takes the output from the preceding layer as input. The number of layers in artificial neural networks is increased to construct deep learning systems. One of these architectures is the Convolutional Neural Network (CNN). Table 2 lists the fundamental layers of CNN-based models. In this work, we evaluated four CNN-based models: ResNet18 [37], ResNet50 [38], ResNet101 [39], and GoogLeNet [40].

Table 2. Basic layers of CNN.

Layer	Description
Convolution	It is this layer that extracts the features from the input images. This layer performs the mathematical operations of convolution between the input image and a filter of size MxM. The dot product between the filter and the sections of the input image with regard to the filter size (MxM) is obtained by sliding the filter over the image.
Activation Function	Activation layers give the network nonlinearity by appending an activation function to the output of the layer that came before it. It will take the output of the convolution layer and apply an element-wise activation function. Relu is the most often utilized activation function. In this layer, the input data's negative values are set to 0, which speeds up the network's learning process.
Pooling	The main objective of this layer is to minimize the convolved feature map's size in order to save computational costs. Reducing the links between layers and working separately on every feature map help achieve this. Different approaches result in different sorts of pooling techniques. It basically encapsulates the properties that a convolution layer produces. In this way, it speeds up the computation, saves memory, and prevents overfitting.
Fully Connected Layer	This stage flattens the input pictures from the preceding layers and supplies them to the FC layer. The flattened vector is subsequently routed via a few further FC levels, where the mathematical function operations are often done. At this point, the classification step is performed.

3.3. Feature selection

Feature selection is a crucial step in model creation, involving the careful identification and inclusion of relevant features. When attribute selection is neglected, several challenges may arise:

- Extended model training times can occur, impacting efficiency.
- Model simplicity and interpretability are compromised when dealing with an excessive number of features.
- Overfitting becomes a concern, leading to high performance on training data but poor generalization to test data, especially when the datasets differ.

As a result, prioritizing data cleaning and feature selection as primary and essential steps in the model creation process is imperative for addressing these potential issues.

The maximum relevance and minimal redundancy (mRMR) technique [41] is used to assess the importances of variables in this study. Each characteristic may be rated according to how important it is to the target variable, and the ranking procedure can take into account the duplication of these features at the same time [42]. The mutual information, *I(M, N)*, is used to determine the level of similarity between M and N in this feature selection technique, which treats each feature individually from the dataset (1). The mutual information between features i and j is denoted by $I(F_i, F_j)$. $I(F_i, H)$ represents how similar every feature

i is to the vector of class labels and its related discrete random variable, H. S is the set of features to be chosen and $|S|$ is the number of elements in the set of features. To guarantee that the cluster to be selected is the best cluster that can be selected, the conditions of maximum relevance (2) and minimum redundancy (3) must be met.

$$I(M,N) = \sum_{n\in N}\sum_{m\in M} p(m,n)\log\left(\frac{p(m,n)}{p_1(m)p_2(n)}\right) \quad (1)$$

$$\max W, W = \frac{1}{|S|}\sum\nolimits_{F_i\in S} I(F_i, H) \quad (2)$$

$$minV, V = \frac{1}{|S|^2}\sum\nolimits_{F_i,F_j\in S} I(F_i, F_j) \quad (3)$$

3.4. Classification

One of the simplest and most intuitive classification technique is the K Nearest Neighbour (K-NN) algorithm [43]. Within this algorithm, the K value is selected to represent the number of neighboring data points to be considered for classification. To calculate these distances, various distance measures such as Euclidean, Manhattan, and Minkowski can be used to measure the similarity between elements in the dataset. The K-NN algorithm operates by identifying the K nearest neighbors of a given data point based on these selected distance measures. It then predicts the class to which the data point belongs by examining the classes of its nearest neighbors. As a result, the data point is assigned to the class that is most common among its nearest neighbors.

Ensemble Learning (EL) is a classification approach that relies on predictions and judgments from several classifiers [44]. It applies each classifier's result by consensus using input from more than one classifier at the same time. In most circumstances, this technique outperforms a single classifier. The fact that the classifiers' errors differ increases ensemble classification performance. To obtain variations in classifier predictions, different subsets of the training dataset are employed. The bootstrap technique is used to generate and train subsets of the training dataset. In this study, AdaBoost algorithm based on boosting approach has been used as the EL algorithm.

The Support Vector Machine (SVM) is a machine learning algorithm developed for classification when the relationships between variables are not well understood Like Decision Trees (DTs), SVM can also be applied to regression analysis. The dataset is usually divided into two subsets: the training set and the test set. Using a labeled training set, the SVM algorithm identifies the ideal hyperplane separating the classes [45]. More than one hyperplane can potentially be used to separate the two classes. The optimal hyperplane is defined in this context as the one that maximizes the distance from the nearest data points of each class.

A multi-variable algorithm called discriminant analysis (DA) ensures the division of N elements into two or more classes according to diverse characteristics and provides the necessary functions. Linear discriminant analysis (LDA) has been used in this study [46]. It can be applied provided that the variance-covariance matrices between groups of randomly drawn sample data matrices from populations with multivariate normal distribution are equal.

3.5. The proposed system

The model proposed in this study is built upon two types of pre-trained CNN models, namely ResNet18 and GoogleNet. This model encompasses four fundamental steps. Firstly, the features are extracted through ResNet18 and GoogleNet after undergoing necessary pre-processing operations. Subsequently, the feature set obtained by merging these features undergoes MRMR feature selection. Finally, the classification step is executed using the refined feature set. The flow chart of the model is given in Figure 4.

This system is based on two types of feature-level fusion: concatenating the features, summing the features. The difference between the two types is the way they combine features drawn from different models.

3.5.1 Concatenating the features

In this type of fusion, *FS=[fs1_1,…,fs1_n,fs2_1,…,fs2_m]* is obtained by combining feature set *FS_1=[fs1_1,…,fs1_n]* taken from a CNN-based architecture with feature set *FS_2=*[*fs2_1,…,fs2_m*] taken from the other CNN-based architecture (Figure 2). The next steps are performed with this new set.

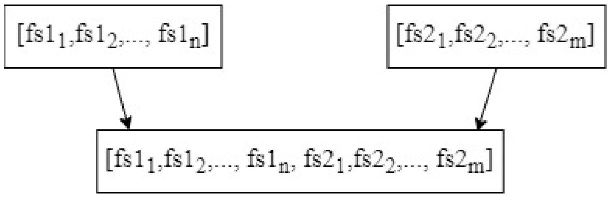


Figure 2. Concantenation of the features.

3.5.2 Summing the features

In this type of fusion, *FS=[fs1_1+fs_1,…, fs1_n+fs2_n]* is obtained by summing the feature set *FS_1=[fs1_1,…,fs1_n]* taken from a CNN-based architecture with the feature set *FS_2=[fs2_1,…,fs2_m]* taken from the other architecture (Figure 3). The next steps are carried out using the new feature set, as in the previous fusion model.

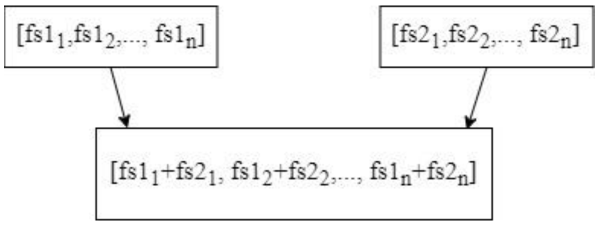


Figure 3. Summation of the features.

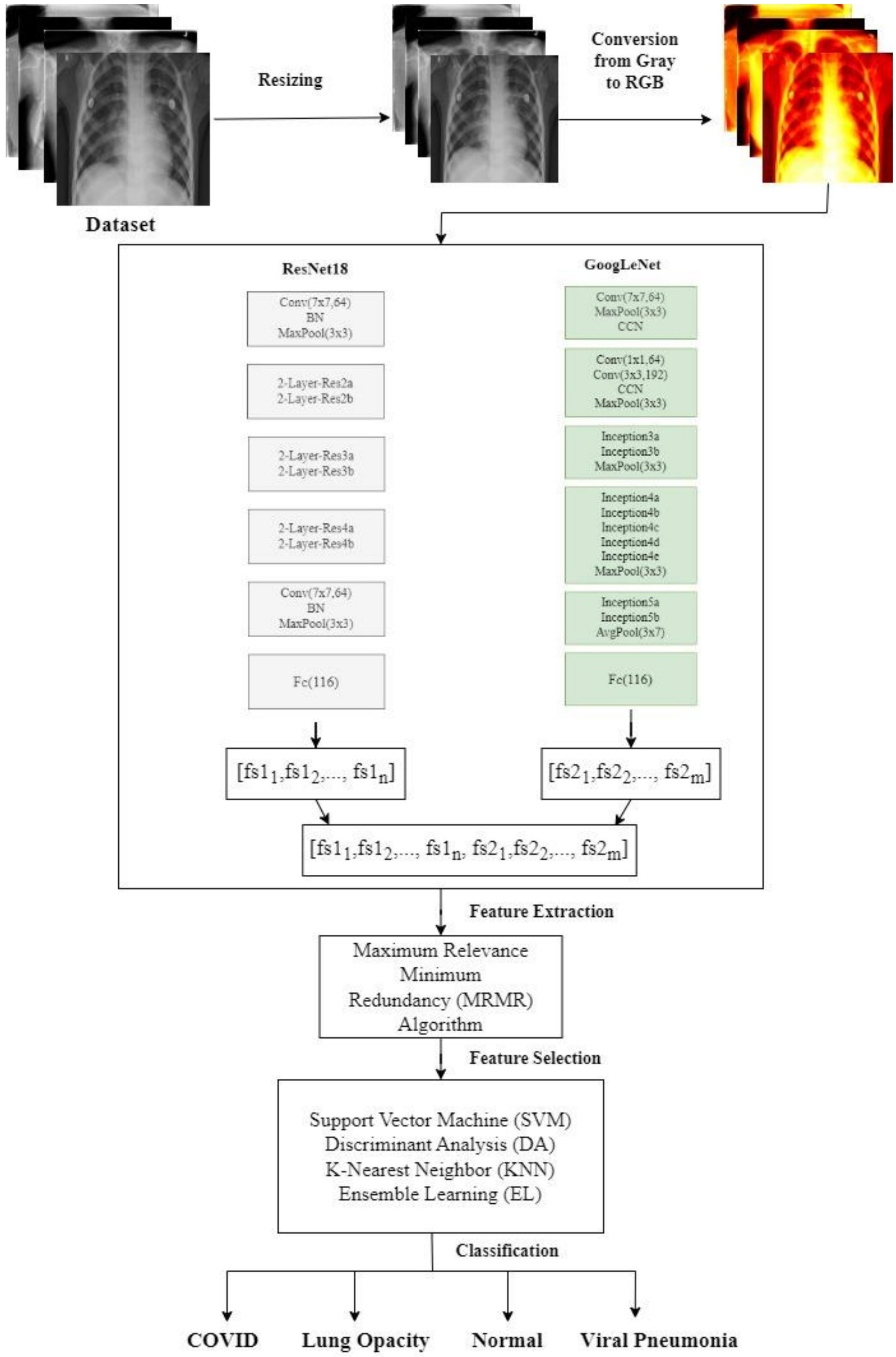


Figure 4. Flow chart of the model developed using concatenating the features.

4. Experimental Results

In this paper, deep learning-based models have been incorporated to classify the images. The performances of the systems are validated by computing the evaluation metrics of accuracy, sensitivity (recall), specificity, precision, and *F1* score. These metrics are calculated by using the values of true positive (*TP*), true negative (*TN*), false positive (*FP*), and false negative (*FN*). When we consider a dataset with two classes, covid and non-covid (normal), accuracy is the division of the number of correctly classified images (*TP+TN*) by the total number of images (2). It assesses the overall correctness of the models. Sensitivity is the division of the number of correctly classified images for covid class by the sum of TP and *FN* (3). Sensitivity measures the model's capacity to capture all positive cases while avoiding false negatives. The ratio of correctly identified non-covid images to all non-covid images in the dataset gives specificity value (4). Specificity refers to the number of negative cases correctly predicted. To get around the limitations of accuracy metric, the precision metric is employed. It investigates how many people that are actually infected with covid are classified as infected (5). *F1* calculates the harmonic mean of precision and recall (6).

Accuracy=(TP+TN)/(TP+TN+FP+FN) (2)

Sensitivity (Recall)=TP/(TP+FN) (3)

Specificity=TN/(TN+FP) (4)

Precision=TP/(TP+FP) (5)

F1=(2×Precision×Recall)/(Precision+Recall) (6)

The dataset has been split into two subsets for training and testing. Specifically, 70% of the dataset has been allocated for training the proposed model, while the remaining 30% has been reserved for testing the model's performance.

Information regarding the layers from which features are extracted can be found in Table 3. Specifically, 1,000 features are extracted from the 'fc1000' layer of ResNet18, while 1,024 features are extracted from the 'pool5-drop_7x7_s1' layer of GoogLeNet. These extracted features are then used in the subsequent stages of the model.

Table 3. Information about the layers.

Model	Layer	Feature Number
ResNet18	fc1000	1000
ResNet50	fc1000	1000
ResNet101	fc1000	1000
GoogLeNet	pool5-drop_7x7_s1	1024

4.1. xperimental results for hybrid models

The findings in Table 4 show the classification performance of the features drawn from ResNet18, ResNet50, ResNet101, and GoogLeNet without applying any feature selection method. It is evident that the highest success is achieved when the features pulled from ResNet50 are classified by SVM (0.9113). The worst success is seen when the hybrid model of ResNet101&EL is used (0.7764). When the accuracy averages of feature extractors are examined, the averages of ResNet18, ResNet50, ResNet101, and GoogLeNet are 0.8347, 0.8619, 0.8462, and 0.8268, respectively. On the other hand, the average accuracy values of the classifiers are 0.809, 0.7888, 0.8868, 0.885 for KNN, EL, SVM and DA, respectively.

Table 5 provides insights into the classification performance of features extracted from the architectures after applying the mRMR feature selection method. Here, the effect of different feature numbers on performance is also examined. Therefore, the results for 250 features have been tested first, and then the results for 500 features have been tested. It is seen that the highest success is obtained when 500 of the features drawn from ResNet50 are selected with the help of mRMR and classified with SVM (0.9067). The average accuracy rate obtained when the number of features is 250 is 0.8360, while the average accuracy rate obtained when the number of features is 500 is 0.8413. From this, it can be deduced that the results obtained for 500 features are higher.

Table 4. Results for hybrid models without using feature selection.

MODEL	Classifier	ACC	SENS	SPEC	PREC	F1
RESNET18	KNN	0.7936	0.9344	0.9899	0.8812	0.9070
	EL	0.7996	0.9467	0.9791	0.7475	0.8354
	SVM	0.8728	0.9309	0.9948	0.9332	0.9320
	DA	0.8728	0.9659	0.9931	0.9109	0.9376
RESNET50	KNN	0.8370	0.9697	0.9907	0.8852	0.9256
	EL	0.7922	0.9228	0.977	0.7237	0.8112
	SVM	**0.9113**	0.9703	0.9953	0.9362	0.9529
	DA	0.9072	0.9712	0.9949	0.9320	0.9512
RESNET101	KNN	0.8090	0.9700	0.9910	0.8927	0.9297
	EL	0.7764	0.9546	0.9813	0.7821	0.8598
	SVM	0.9076	0.9700	0.9947	0.9288	0.9490
	DA	0.8918	0.9703	0.9925	0.9012	0.9344
GOOGLENET	KNN	0.7964	0.9084	0.9880	0.8589	0.8830
	EL	0.7871	0.9021	0.9791	0.7525	0.8205
	SVM	0.8555	0.9207	0.9959	0.9480	0.9341
	DA	0.8682	0.9436	0.9931	0.9109	0.9270

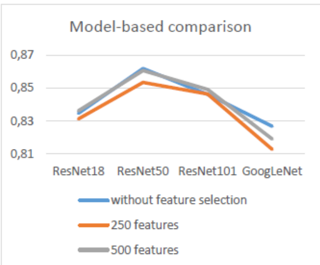
Table 5. Results for hybrid models using MRMR feature selection.

MODEL	FEAT	CLASS.	ACC	SENS	SPEC	PREC	F1
ResNet18	250	KNN	0.8362	0.9627	0.9879	0.8491	0.9023
		EL	0.7551	0.92871	0.9687	0.6366	0.7554
		SVM	0.8745	0.9612	0.9920	0.8959	0.9274
		DA	0.8591	0.9535	0.9917	0.8937	0.9227
	500	KNN	0.8359	0.9616	0.9881	0.8512	0.9030
		EL	0.7565	0.9416	0.9686	0.6344	0.7581
		SVM	0.8798	0.9596	0.9930	0.9086	0.9334
		DA	0.8725	0.9581	0.9920	0.899	0.9276

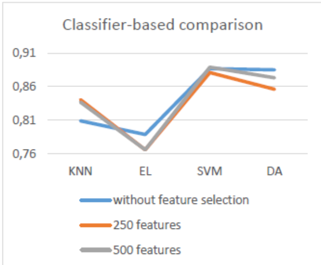
Table 5 (continued)

MODEL	FEAT	CLASS.	ACC	SENS	SPEC	PREC	F1
ResNet50	250	KNN	0.8557	0.9760	0.9895	0.8661	0.9178
		EL	0.7896	0.9205	0.9780	0.7386	0.8196
		SVM	0.8979	0.9668	0.9946	0.9288	0.9474
		DA	0.8707	0.9726	0.9897	0.8661	0.9162
	500	KNN	0.8561	0.9778	0.9912	0.8884	0.9310
		EL	0.7921	0.9327	0.9782	0.7365	0.8230
		SVM	**0.9067**	0.9680	0.9949	0.9309	0.9491
		DA	0.8879	0.9745	0.9919	0.8937	0.9324
ResNet101	250	KNN	0.8480	0.9720	0.9910	0.8863	0.9272
		EL	0.7784	0.9530	0.9808	0.7758	0.8553
		SVM	0.8922	0.9608	0.9952	0.9373	0.9489
		DA	0.8671	0.9674	0.9908	0.8820	0.9227
	500	KNN	0.8388	0.9775	0.9903	0.8789	0.9256
		EL	0.7759	0.9488	0.9802	0.7683	0.8491
		SVM	0.9000	0.9670	0.9950	0.9330	0.9497
		DA	0.8821	0.9679	0.9921	0.8969	0.9311
GoogLeNet	250	KNN	0.8231	0.9606	0.9862	0.8300	0.8905
		EL	0.7411	0.9138	0.9649	0.5972	0.7224
		SVM	0.8592	0.9596	0.9909	0.8842	0.9204
		DA	0.8276	0.8936	0.9891	0.8661	0.8797
	500	KNN	0.8167	0.9503	0.9847	0.8130	0.8763
		EL	0.7402	0.8929	0.9631	0.5760	0.7003
		SVM	0.8694	0.9644	0.9917	0.8927	0.9272
		DA	0.8499	0.9302	0.9915	0.8927	0.9111

Figure 5 offers a comparison of model and classifier performances both before and after feature selection.



(a)



(b)

Figure 5. Comparisons using average accuracy rates (a) Model-based (b) Classifier-based.

In this graph, blue lines show the results when feature selection is not applied, red lines show the results obtained for 250 features, and green lines show the results obtained for 500 features. When Figure 5(a) is examined, the average accuracy rates obtained when classification is performed without applying feature selection are very close to the average accuracy rates obtained when 500 features are extracted using mRMR and classification is performed. Classification performances for 250 features are always slightly lower. Among the four models, the CNN model from which the features with the highest classification performance are extracted is ResNet50. GoogLeNet is the CNN model that produces the lowest performing and most vulnerable features to changes. Looking at Figure 5(b), it can be seen that SVM is the classifier with the highest performance. Especially when SVM is used, the results obtained for all three cases (for without feature selection, for 250 features, and for 500 features) are close to each other, while the accuracy rates obtained from other classifiers differs.

The classifier with the second highest success is DA. However, it is seen that DA is not as robust as SVM and its success decreases when feature selection is applied. Classifiers other than KNN perform better when feature selection is not applied. If we talk about the number of features, it can be said that while the performance is lower for 250 features, higher performance is achieved for 500 features. This is valid for both graphs in Figure 5.

4.2. Experimental results for hybrid models with feature fusion

The performances obtained by classifying fused features without feature selection are given in Table 6. Because 1000 features are extracted from ResNet architectures and 1024 features are extracted from GoogLeNet, the sum of two feature vectors of different sizes cannot be made. Therefore, only the performance metrics obtained for concatenating the fetaures are included in this table.

Classification success of the features extracted from ResNet50&ResNet101 with SVM is 0.9509. This rate is the highest accuracy rate in the table. It is striking that the performances of SVM and DA are close to each other. On the other hand, it is obvious that EL is always the worst performing classifier in all cases.

The success rates achieved when employing feature selection in hybrid models with feature fusion are presented in Table 7. Notably, when classifying the ResNet50&ResNet101 fusion using VM, a higher result has been obtained (0.9575) surpassing all results obtained after applying feature selection. This achievement closely approaches the success (0.9509) achieved without implementing feature selection. When concatenating is applied to features as a fusion model, an average accuracy of 0.8811 is achieved, while an average accuracy rate of 0.8722 is achieved when summing is applied as a fusion model.

Table 8 includes average accuracy rates of the fusion models. It is observed that the highest result is achieved when ResNet50&ResNet101 is used (0.9112). While the average success rates achieved after feature selection increase in some models such as ResNet18&ResNet101, ResNet50&ResNet101, and ResNet101&GoogLeNet, the average success rates decrease in the models such as ResNet18&ResNet50, and ResNet50&GoogLeNet. In the ResNet101&GoogLeNet model, the success rates obtained in each case are approximately the same.

When a classifier-based comparison is made, it can be said that SVM is the best classifier while EL is the classifier that gives the lowest successful results in this study (Table 9). It seems that DA is as successful as SVM, but its success begins to decrease as the number of features decreases, indicating that it is not as robust as SVM.

Table 6. Results for hybrid models with feature fusion without using feature selection.

FUSION	Classifier	ACC	SENS	SPEC	PREC	F1
RESNET18 & RESNET50	KNN	0.8570	0.9789	0.9918	0.8959	0.9356
	EL	0.7823	0.9501	0.9732	0.6796	0.7924
	SVM	0.9099	0.9790	0.9953	0.9368	0.9574
	DA	0.9051	0.9724	0.9939	0.9182	0.9446
RESNET18 & RESNET101	KNN	0.8525	0.9773	0.9919	0.8981	0.9361
	EL	0.7755	0.9421	0.9777	0.7383	0.8278
	SVM	0.9097	0.9736	0.9951	0.9338	0.9533
	DA	0.9073	0.9716	0.9938	0.9160	0.943
RESNET18 & GOOGLENET	KNN	0.8602	0.9732	0.9915	0.8914	0.9305
	EL	0.7750	0.9489	0.9727	0.6766	0.7899
	SVM	0.9000	0.9741	0.9941	0.9219	0.9473
	DA	0.8959	0.9741	0.9942	0.9234	0.9481
RESNET50 & RESNET101	KNN	0.8381	0.9877	0.9952	0.9405	0.9635
	EL	0.8325	0.9676	0.9835	0.7938	0.8722
	SVM	**0.9509**	0.9892	0.9980	0.9713	0.9802
	DA	0.9494	0.9868	0.9966	0.9522	0.9692
RESNET50 & GOOGLENET	KNN	0.8681	0.9877	0.9953	0.9405	0.9635
	EL	0.8183	0.9745	0.9753	0.6908	0.8085
	SVM	0.9448	0.9914	0.9983	0.9766	0.9839
	DA	0.9507	0.9934	0.9977	0.9671	0.9801
RESNET101 & GOOGLENET	KNN	0.8345	0.9821	0.9947	0.9352	0.9581
	EL	0.8254	0.9664	0.9834	0.7938	0.8716
	SVM	0.9482	0.9935	0.9978	0.9692	0.9812
	DA	0.9480	0.9956	0.9975	0.9649	0.9444

Table 7. Results for hybrid models with feature fusion using MRMR feature selection.

FUSION	FEAT.	CLASS.	ACC	SENS	SPEC	PREC	F1	FUSION	FEAT.	CLASS.	ACC	SENS	SPEC	PREC	F1
ResNet18 & ResNet50	250+250	KNN	0.8669	0.9765	0.9910	0.8842	0.9281	ResNet18 + ResNet50	250	KNN	0.8537	0.9728	0.9901	0.8746	0.9211
		EL	0.7862	0.9099	0.9773	0.7301	0.8101			EL	0.7750	0.9569	0.9715	0.6610	0.7819
		SVM	0.8987	0.9624	0.9943	0.9245	0.9431			SVM	0.8911	0.9612	0.9940	0.9214	0.9409
		DA	0.8905	0.9713	0.9924	0.8990	0.9338			DA	0.8728	0.9659	0.9925	0.9033	0.9336
	500+500	KNN	0.8663	0.9779	0.9916	0.8916	0.9327		500	KNN	0.8499	0.9739	0.9898	0.8714	0.9198
		EL	0.7917	0.9171	0.9784	0.7407	0.8195			EL	0.7797	0.9525	0.9716	0.6599	0.7797
		SVM	0.9023	0.9614	0.9945	0.9267	0.9437			SVM	0.8988	0.9657	0.9945	0.9267	0.9458
		DA	0.9029	0.9717	0.9934	0.9118	0.9408			DA	0.8913	0.9704	0.9928	0.9054	0.9368
ResNet18 & ResNet101	250+250	KNN	0.8942	0.9867	0.9957	0.9437	0.9647	ResNet18 + ResNet101	250	KNN	0.8845	0.9865	0.9947	0.9309	0.9579
		EL	0.8254	0.9694	0.9845	0.8087	0.8818			EL	0.8161	0.9550	0.9865	0.8353	0.8912
		SVM	0.9421	0.9871	0.9982	0.9745	0.9807			SVM	0.9353	0.9838	0.9975	0.9660	0.9748
		DA	0.9432	0.9835	0.9963	0.9479	0.9654			DA	0.9303	0.9812	0.9960	0.9447	0.9626
	500+500	KNN	0.8832	0.9922	0.9958	0.9458	0.9684		500	KNN	0.8703	0.9863	0.9939	0.9214	0.9527
		EL	0.8221	0.9703	0.9836	0.7981	0.8758			EL	0.8072	0.9382	0.9853	0.8225	0.8766
		SVM	0.9438	0.9870	0.9978	0.9692	0.9780			SVM	0.9405	0.9892	0.9978	0.9692	0.9791
		DA	0.9515	0.9847	0.9970	0.9575	0.9709			DA	0.9421	0.9835	0.9966	0.9522	0.9676

Table 7 (continued)

ResNet50 & ResNet101	250+250	KNN	0.8959	0.9876	0.9949	0.9320	0.9590	ResNet50 + ResNet101	250	KNN	0.8932	0.9887	0.9948	0.9320	0.9595
		EL	0.8429	0.9677	0.9864	0.8289	0.8930			EL	0.8302	0.9616	0.9859	0.8247	0.8879
		SVM	0.9518	0.9839	0.9983	0.9756	0.9797			SVM	0.9454	0.9850	0.9982	0.9756	0.9802
		DA	0.9476	0.9846	0.9966	0.9522	0.9681			DA	0.9328	0.9845	0.9961	0.9469	0.9653
	500+500	KNN	0.8837	0.9899	0.9950	0.9352	0.9617		500	KNN	0.8892	0.9922	0.9955	0.9405	0.9656
		EL	0.8468	0.9715	0.9869	0.8342	0.8977			EL	0.8318	0.9594	0.9862	0.8289	0.8894
		SVM	**0.9575**	0.9881	0.9982	0.9745	0.9813			SVM	0.9528	0.9892	0.9983	0.9766	0.9829
		DA	0.9567	0.9890	0.9971	0.9596	0.9741			DA	0.9444	0.9879	0.9967	0.9543	0.9708
ResNet18 & GoogLeNet	250+250	KNN	0.8584	0.9736	0.9892	0.8618	0.9143	ResNet18 + GoogLeNet	250	KNN	0.8543	0.9771	0.9891	0.8608	0.9153
		EL	0.7701	0.9349	0.9709	0.6567	0.7715			EL	0.7630	0.9092	0.9701	0.6493	0.7576
		SVM	0.8877	0.9676	0.9939	0.9192	0.9428			SVM	0.8819	0.9612	0.9940	0.9214	0.9409
		DA	0.8746	0.9624	0.9921	0.8969	0.9285			DA	0.8594	0.9499	0.9912	0.8874	0.9176
	500+500	KNN	0.8553	0.9726	0.9896	0.8682	0.9175		500	KNN	0.8495	0.9723	0.9888	0.8576	0.9113
		EL	0.7720	0.9440	0.9701	0.6451	0.7664			EL	0.7670	0.9222	0.9706	0.6546	0.7657
		SVM	0.8939	0.9676	0.9940	0.9203	0.9434			SVM	0.8881	0.9707	0.9935	0.9138	0.9414
		DA	0.8869	0.9673	0.9933	0.9118	0.9387			DA	0.8764	0.9680	0.9924	0.9012	0.9334
ResNet50 & GoogLeNet	250+250	KNN	0.8642	0.9749	0.9897	0.8672	0.9179	ResNet50 + GoogLeNet	250	KNN	0.8643	0.9810	0.9903	0.8757	0.9253
		EL	0.7911	0.9118	0.9771	0.7248	0.8076			EL	0.7807	0.9489	0.9772	0.7301	0.8252
		SVM	0.9007	0.9710	0.9945	0.9267	0.9483			SVM	0.8937	0.9688	0.9942	0.9235	0.9456
		DA	0.8867	0.9780	0.9922	0.8980	0.9363			DA	0.8646	0.9698	0.9913	0.8884	0.9273
	500+500	KNN	0.8651	0.9810	0.9904	0.8767	0.9259		500	KNN	0.8617	0.9751	0.9903	0.8757	0.9227
		EL	0.7926	0.9084	0.9781	0.7375	0.8141			EL	0.7873	0.9472	0.9785	0.7439	0.8333
		SVM	0.9081	0.9765	0.9946	0.9277	0.9515			SVM	0.9052	0.9744	0.9947	0.9288	0.9510
		DA	0.9021	0.9785	0.9940	0.9192	0.9479			DA	0.8867	0.9726	0.9929	0.9065	0.9384
ResNet101 & GoogLeNet	250+250	KNN	0.8875	0.9875	0.9941	0.9224	0.9538	ResNet101 + GoogLeNet	250	KNN	0.8873	0.9875	0.9943	0.9256	0.9556
		EL	0.8291	0.9652	0.9835	0.7949	0.8718			EL	0.8134	0.9704	0.9838	0.8023	0.8784
		SVM	0.9426	0.9828	0.9978	0.9692	0.9759			SVM	0.9340	0.9774	0.9976	0.9671	0.9722
		DA	0.9338	0.9811	0.9955	0.9373	0.9587			DA	0.9199	0.9798	0.9947	0.9277	0.9531
	500+500	KNN	0.8719	0.9899	0.9950	0.9352	0.9617		500	KNN	0.8755	0.9887	0.9943	0.9267	0.9567
		EL	0.8281	0.9641	0.9839	0.8002	0.8746			EL	0.8167	0.9646	0.9845	0.8098	0.8804
		SVM	0.9470	0.9903	0.9982	0.9756	0.9829			SVM	0.9417	0.9860	0.9980	0.9724	0.9791
		DA	0.9471	0.9846	0.9964	0.9501	0.9670			DA	0.9351	0.9823	0.9961	0.9458	0.9637

Table 8. Average accuracy rates of fusion models.

FUSION	without feature selection	250+250	500+500	MODEL	250	500
ResNet18&ResNet50	0.8636	0.8606	0.8658	ResNet18+ResNet50	0.8482	0.8549
ResNet18&ResNet101	0.8613	0.9012	0.9002	ResNet18+ResNet101	0.8916	0.8900
ResNet50&ResNet101	0.8578	0.9096	**0.9112**	ResNet50+ResNet101	0.9004	0.9046
ResNet18&GoogLeNet	0.8927	0.8477	0.8520	ResNet18+GoogLeNet	0.8397	0.8453
ResNet50&GoogLeNet	0.8955	0.8607	0.8670	ResNet50+GoogLeNet	0.8508	0.8602
ResNet101&GoogLeNet	0.8890	0.8983	0.8985	ResNet101+GoogLeNet	0.8887	0.8923

Table 9. Classifier based comparison.

CLASSIFIER	without feature selection	250+250	500+500	250	500
KNN	0.8517	0.8779	0.8709	0.8729	0.8660
EL	0.8015	0.8075	0.8089	0.7964	0.7983
SVM	**0.9273**	0.9206	0.9254	0.9136	0.9212
DA	0.9261	0.9127	0.9245	0.8966	0.9127

5. Discussion

A multi-level fusion approach for radiography images has been presented in this paper. For this purpose, the successes have been compared as a result of fusing different numbers of features taken from different CNN models. In this study, where multi-feature fusion with CNN is examined from various aspects, the fusion model that shows the highest success is ResNet50&ResNet101 with 95.75% (for 500+500 features). When compared to the studies in the literature, it is seen that this success is much higher than the studies of Hafeez et al. [18] and Khan et al. [29]. A comparison with other studies in the literature is illustrated in Table 10.

6. Conclusion

While CNN-based classification shows high performance, it is known to have a disadvantage in time. So, the reason why hybrid models are preferred instead of CNN-based classification is to take advantage of time. Hybrid studies reveal that CNN has high performance not only in classification but also in feature extraction. In this study, the performances of hybrid models are compared with the performances of the systems that implement hybrid models with feature level fusion. This is valid both when feature selection is applied and when it is not applied. The purpose of using feature fusion is to investigate whether performance can be further improved.

In this study, where the feature extraction performances of different versions of residual networks (ResNet18, ResNet50, ResNet101) and GoogLeNet are tested, striking results are achieved. While the success in hybrid models reaches a maximum of 0.9113 (ResNet50&SVM), the success increases up to 0.9575 (ResNet50&ResNet101&SVM) when feature level fusion is used. The success of ResNet50 surpasses other CNN models. It is observed that the success of ResNet50 exceeds other CNN models in hybrid models, and the results obtained when feature level fusion is used are compatible with the graph given in Figure 5. It can be seen that the second most successful CNN model is ResNet101 and that ResNet50 gives the most successful results when fused with ResNet101.

When evaluated on a classifier basis, the highest success is achieved when classification is performed with SVM, whether feature selection is applied or not. The second most successful classifier is found to be DA. When DA is used for classification, the results obtained are very close to the results of SVM. The fact that DA gives results in a much shorter time compared to SVM may be a justified reason why DA is preferable.

In addition, a different number of features have been selected using MRMR, which is a feature selection method, and the effect of the number of features on the performance has been examined. According to the experimental studies, although choosing 250 or 500 features from the models does not affect the achievements much, generally higher results are obtained if 500 features are selected for both hybrid systems and hybrid systems with feature fusion (except for some cases of KNN).

Two different fusion models have been used in the feature level fusion: concatenating the features, and summing the features. The average accuracy rate obtained when concatenating the features is used is slightly higher than the average accuracy value obtained when summing the features is used. On the other hand, the number of features obtained in the concatenating process is double that of the features obtained in the summing process. Therefore, when concatenating is preferred instead of summing, the classification time increases slightly.

It is a known fact that most of the completion time of CNN-based models is spent in the classification phase. Despite all its positive aspects, it is clear that extracting features from two models instead of one in hybrid models with multi-feature fusion causes extra loss of time. It is also obvious that hybrid models with multi-feature fusion give results in a longer time compared to hybrid models.

In the future study, it is aimed to improve the existing CNN models in terms of time and performance and compare them with the multi-feature fusion examined in this study.

Table 10. A comparison with the other studies in the literature.

Study	Model	Dataset	Accuracy
MobileNetV2-Based Model [13]	Based on MobileNetV2	1,576 normal, 3,616 COVID-19, 4,265 pneumonia images	97.61%
LW-CORONet [17]	Custom CNN	Dataset-1 with 2,250 images (750 pneumonia, 750 normal, and 750 COVID-19), and dataset-2 with 15,999 images (5,575 pneumonia, 8,066 normal, and 2,358 COVID-19)	98.67% for dataset-1 and 95.67% for dataset-2
COVID-AleXception [20]	Combination of Xception and AlexNet	15,153 X-ray images (1,345 pneumonia, 3,616 COVID-19, 10,192 normal)	98.68%
Custom CNN [21]	Custom CNN	Not specified.	89.855%
Lightweight CNN for COVID-19 [22]	Seven pre-trained CNNs (InceptionV3, Xception, ResNet50V2, etc.)	600 COVID-19, 600 normal, 600 pneumonia images	98.33% (EfficientNetV2: 97.73%)
Custom CNN Diagnosis [23]	Custom CNN	10,293 X-ray images (4,200 pneumonia, 2,875 COVID-19, 3,218 normal)	98.50%
DL Diagnosis (VGG19 and ResNet50) [24]	VGG19 and ResNet50	11,263 pneumonia, 11,956 COVID-19, 10,701 normal images	VGG19: 96.60%, ResNet50: 95.80%
CoroNet [29]	Based on Xception	330 bacterial pneumonia, 327 viral pneumonia, 284 COVID-19, 310 normal X-ray images	89.60%
Comparative DL Analysis [30]	VGG16, ResNet50, DenseNet121, VGG19	1,592 X-ray images (802 normal, 790 COVID-19)	VGG16: 99.33%, ResNet50: 97.00%, DenseNet121, VGG19: 96.66%
Our Study	**Multi-feature fusion**	**21,168 X-ray images (3,619 covid, 6,012 lung_opacity, 10,192 normal, 1,345 viral pneumonia)**	**95.75%**

Ethics committee approval and conflict of interest statement

This article does not require ethics committee approval. This article has no conflicts of interest with any individual or institution.

Author Contribution Statement

In the study carried out, Author 1 contributed to the formation of the idea, design and obtaining the results, and Author 2 contributed to the literature review, evaluation of the obtained results and examination of the results.

References

[1] Alimohamadi, Y., Sepandi, M., Taghdir, M., Hosamirudsari, H., 2020. Determine the most common clinical symptoms in COVID-19 patients: a systematic review and meta-analysis. Journal of Preventive Medicine and Hygiene, Vol.61(3).

[2] Bayram, F., Eleyan, A., 2022. COVID-19 detection on chest radiographs using feature fusion based deep learning. Signal, Image and Video Processing, Vol.16(6), pp.1455–1462.

[3] Çelik, G., Talu, M., 2021. Generating the image viewed from EEG signals. Pamukkale University Journal of Engineering Sciences, Vol.27(2).

[4] Altan, G., 2022. Breast cancer diagnosis using deep belief networks on ROI images. Pamukkale University Journal of Engineering Sciences, Vol.28(2), pp.286–291.

[5] Yuzkat, M., İlhan, H., Aydın, N., 2023. Detection of human sperm cells using deep learning-based object detection methods. Pamukkale University Journal of Engineering Sciences, Vol.30(4).

[6] Cevik, F., Kilimci, Z.H., 2020. The evaluation of Parkinson's disease with sentiment analysis using deep learning methods and word embedding models. Pamukkale University Journal of Engineering Sciences, Vol.27(2), pp.151–161.

[7] Akalın, F., Yumuşak, N., 2024. Detection of gastrointestinal anomalies with a deep learning-based ensemble classifier approach. Pamukkale University Journal of Engineering Sciences, Vol.30(3), pp.366–373.

[8] Kong, L., Cheng, J., 2022. Classification and detection of COVID-19 X-Ray images based on DenseNet and VGG16 feature fusion. Biomedical Signal Processing and Control, Vol.77.

[9] Sitaula, C., Hossaini, M.B., 2021. Attention-based VGG-16 model for COVID-19 chest X-ray image classification. Applied Intelligence, Vol.51, pp.2850–2863.

[10] El-Kenawy, E.S.M., Ibrahim, A., Mirjalili, S., Eid, M.M., Hussein, S.E., 2020. Novel feature selection and voting classifier algorithms for COVID-19 classification in CT images. IEEE Access, Vol.8, pp.179317–179335.

[11] El-Kenawy, E.S.M., Mirjalili, S., Ibrahim, A., Alrahmawy, M., El-Said, M., Zaki, R.M., Eid, M.M., 2021. Advanced meta-heuristics, convolutional neural networks, and feature selectors for efficient COVID-19 X-ray chest image classification. IEEE Access, Vol.9, pp.36019–36037.

[12] Reshi, A.A., Rustam, F., Mehmood, A., Alhossan, A., Alrabiah, Z., Ahmad, A., Alsuwailem, H., Choi, G.S., 2021. An efficient CNN model for COVID-19 disease detection based on X-ray image classification. Complexity, pp.1–12.

[13] Barshooi, A.H., Amirkhani, A., 2022. A novel data augmentation based on Gabor filter and convolutional deep learning for improving the classification of COVID-19 chest X-Ray images. Biomedical Signal Processing and Control, Vol.72, Article 103326.

[14] Srinivas, K., Gagana Sri, R., Pravallika, K., Nishitha, K., Polamuri, S.R., 2024. COVID-19 prediction based on hybrid Inception V3 with VGG16 using chest X-ray images. Multimedia Tools and Applications, Vol.83(12), pp.36665–36682.

[15] El Houby, E.M., 2024. COVID-19 detection from chest X-ray images using transfer learning. Scientific Reports, Vol.14(1), Article 11639.

[16] Abdullah, M., Berhe Abrha, F., Kedir, B., Tagesse, T.T., 2024. A Hybrid Deep Learning CNN model for COVID-19 detection from chest X-rays. Heliyon, Vol.10(5).

[17] Shaban, W.M., Rabie, A.H., Saleh, A.I., Abo-Elsoud, M.A., 2020. A new COVID-19 Patients Detection Strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier. Knowledge-Based Systems, Vol.205.

[18] Loey, M., Manogaran, G., Taha, M.H.N., Khalifa, N.E.M., 2021. A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. Measurement, Vol.167.

[19] Yaşar Çıklaçandır, F.G., Utku, S., 2022. Ensemble learning based classification of infected and uninfected cells. 2nd International Conference on Artificial Intelligence and Data Science, pp.15–18.

[20] Yaşar Çıklaçandır, F.G., Utku, S., Özdemir, H., 2023. Determination of various fabric defects using different machine learning techniques. The Journal of The Textile Institute, pp.1–11.

[21] Kaya, Y., Gürsoy, E., 2023. A MobileNet-based CNN model with a novel fine-tuning mechanism for COVID-19 infection detection. Soft Computing, Vol.27, pp.5521–5535.

[22] Nayak, S.R., Nayak, D.R., Sinha, U., Arora, V., Pachori, R.B., 2022. An Efficient Deep Learning Method for Detection of COVID-19 Infection Using Chest X-ray Images. Diagnostics, Vol.13, Article 131.

[23] Sanida, T., Tabakis, I.M., Sanida, M.V., Sideris, A., Dasygenis, M., 2023. A Robust Hybrid Deep Convolutional Neural Network for COVID-19 Disease Identification from Chest X-ray Images. Information, Vol.14(6), Article 310.

[24] Sanida, T., Sideris, A., Tsiktsiris, D., Dasygenis, M., 2022. Lightweight neural network for COVID-19 detection from chest X-ray images implemented on an embedded system. Technologies, Vol.10, Article 37.

[25] Ayadi, M., Ksibi, A., Al-Rasheed, A., Soufiene, B.O., 2022. COVID-AleXception: A Deep Learning Model Based on a Deep Feature Concatenation Approach for the Detection of COVID-19 from Chest X-ray Images. Healthcare, Vol.10.

[26] Hafeez, U., Umer, M., Hameed, A., Mustafa, H., Sohaib, A., Nappi, M., Madni, H.A., 2022. A CNN based coronavirus disease prediction system for chest X-rays. Journal of Ambient Intelligence and Humanized Computing, pp.1–15.

[27] Huang, M.L., Liao, Y.C., 2022. A lightweight CNN-based network on COVID-19 detection using X-ray and CT images. Computers in Biology and Medicine, Vol.146.

[28] Ghose, P., Uddin, A., Acharjee, U.K., Sharmin, S., 2022. Deep viewing for the identification of COVID-19 infection status from chest X-ray image using CNN based architecture. Intelligent Systems with Applications, Vol.16.

[29] Ibrokhimov, B., Kang, J.Y., 2022. Deep Learning Model for COVID-19-Infected Pneumonia Diagnosis Using Chest Radiography Images. BioMedInformatics, Vol.2, pp.654–670.

[30] Ji, D., Zhang, Z., Zhao, Y., Zhao, Q., 2021. Research on classification of COVID-19 chest X-ray image modal feature fusion based on deep learning. Journal of Healthcare Engineering.

[31] Narin, A., Kaya, C., Pamuk, Z., 2021. Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. Pattern Analysis and Applications, Vol.24, pp.1207–1220.

[32] Hussain, E., Hasan, M., Rahman, M.A., Lee, I., Tamanna, T., Parvez, M.Z., 2021. CoroDet: A deep learning based classification for COVID-19 detection using chest X-ray images. Chaos, Solitons & Fractals, Vol.142.

[33] Ozturk, T., Talo, M., Yildirim, E.A., Baloglu, U.B., Yildirim, O., Acharya, U.R., 2020. Automated detection of COVID-19 cases using deep neural networks with X-ray images. Computers in Biology and Medicine, Vol.121.

[34] Khan, A.I., Shah, J.L., Bhat, M.M., 2020. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest X-ray images. Computer Methods and Programs in Biomedicine, Vol.196.

[35] Khan, I.U., Aslam, N., 2020. A deep-learning-based framework for automated diagnosis of COVID-19 using X-ray images. Information, Vol.11, Article 419.

[36] Rahman, T., 2021. COVID-19 radiography database. [Online] Available at: https://www.kaggle.com/tawsifurrahman/covid19-radiography-database.

[37] Kochgaven, C., Mishra, P., Shitole, S., 2021. Detecting presence of COVID-19 with ResNet-18 using PyTorch. International Conference on Communication Information and Computing Technology (ICCICT), pp.1–6.

[38] Yazan, E., Talu, M.F., 2023. Integration of attention mechanisms into segmentation architectures and their application on breast lymph node images. Pamukkale University Journal of Engineering Sciences, Vol.29(3), pp.248–255.

[39] Aksoy, B., Salman, O.K.M., 2022. Prediction of COVID-19 disease with ResNet-101 deep learning architecture using computerized tomography images. Turkish Journal of Nature and Science, Vol.11(2), pp.36–42.

[40] Al-Huseiny, M.S., Sajit, A.S., 2021. Transfer learning with GoogLeNet for detection of lung cancer. Indonesian Journal of Electrical Engineering and Computer Science, Vol.22(2), pp.1078–1086.

[41] Peng, H., Long, F., Ding, C., 2005. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.27(8), pp.1226–1238.

[42] Cai, Y., Huang, T., Hu, L., Shi, X., Xie, L., Li, Y., 2012. Prediction of lysine ubiquitination with mRMR feature selection and analysis. Amino Acids, Vol.42, pp.1387–1395.

[43] Cover, T., Hart, P., 1996. Nearest neighbor pattern classification. IEEE Transactions on Information Theory, Vol.13(1), pp.21–27.

[44] Breiman, L., 1996. Bagging predictors. Machine Learning, Vol.24, pp.123–140.

[45] Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine Learning, Vol.20, pp.273–297.

[46] Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Annals of Eugenics, Vol.7(2), pp.179–188.