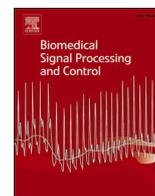




Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.



Detection and classification of COVID-19 disease from X-ray images using convolutional neural networks and histogram of oriented gradients

Aleka Melese Ayalew^a, Ayodeji Olalekan Salau^{b,d,*}, Bekalu Tadele Abeje^c, Belay Enyew^a

^a Department of Information Technology, University of Gondar, Ethiopia

^b Department of Electrical/Electronics and Computer Engineering, Afe Babalola University, Ado-Ekiti, Nigeria

^c Department of Information Technology, Haramaya University, Ethiopia

^d Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, India

ARTICLE INFO

Keywords:

COVID-19
Deep learning
Machine learning
CNN
HOG
SVM
YOLOv3

ABSTRACT

COVID-19 is now regarded as the most lethal disease caused by the novel coronavirus disease of humans. The COVID-19 pandemic has spread to every country on the planet and has wreaked havoc on these countries by increasing the number of human deaths, and in addition, caused intense hunger, and lowered economic productivity. Due to a lack of sufficient radiologist, a restricted amount of COVID-19 test kits is available in hospitals, and this is also accompanied by a shortage of equipment due to the daily increase in cases, as a result of increase in the number of persons infected with COVID-19. Even for experienced radiologists, examining chest X-rays is a difficult task. Many people have died as a result of inaccurate COVID-19 diagnosis and treatment, as well as ineffective detection measures. This paper, therefore presents a unique detection and classification approach (DCCNet) for quick diagnosis of COVID-19 using chest X-ray images of patients. To achieve quick diagnosis, a convolutional neural network (CNN) and histogram of oriented gradients (HOG) method is proposed in this paper to help medical experts diagnose COVID-19 disease. The diagnostic performance of the hybrid CNN model and HOG-based method was then evaluated using chest X-ray images collected from University of Gondar and online databases. The experiment was performed using Keras (with TensorFlow as a backend) and Python. After the DCCNet model was evaluated, a 99.9% training accuracy and 98.3% test accuracy was achieved, while a 100% training accuracy and 98.5% test accuracy was achieved using HOG. After the evaluation, the hybrid model achieved 99.97% and 99.67% training and testing accuracy for detection and classification of COVID-19 which was better by 1.37% compared to when features were extracted using CNN and 1.17% when HOG was used. The DCCNet achieved a result that outperformed state-of-the-art models by 6.7%.

1. Introduction

The current coronavirus outbreak was detected by a Chinese physician in Wuhan, the capital city of Hubei province in mainland China, in late December 2019 [1]. This new virus is very contagious and has rapidly spread over the world. Because the outbreak has spread to 216 nations, the World Health Organization (WHO) declared it a public health emergency of international concern on January 30, 2020. On February 11, 2020, this new coronavirus associated acute respiratory disease was officially named as Corona Virus disease-19 (COVID-19) by the WHO.

Radiologists perform COVID-19 diagnosis manually. However, manual diagnosis of COVID-19 requires experienced radiologists and is a

time-consuming process, tedious, error-prone, and exhaustive since the radiologists are expected to diagnosis a large number of COVID-19 patients. Developing countries of the world face a shortage of equipment, test kits, and radiologists for diagnosing COVID-19. In addition to the diagnostic process being time-consuming, the scarcity of skilled radiologists to provide accurate diagnoses, especially in light of the rapidly increasing COVID-19 cases, is still a major concern.

To determine if a person is infected with COVID-19, a sample is taken from the nose (nasopharyngeal swab) or throat (throat swab) by a healthcare professional. The samples are then transferred to a laboratory for testing. The manual outcome of the test, however, could be a false-negative depending on the timing and quality of the test sample. This method of testing, accurately detected COVID-19 infections 71% of the

* Corresponding author.

E-mail address: ayodejisalau98@gmail.com (A.O. Salau).

<https://doi.org/10.1016/j.bspc.2022.103530>

Received 12 October 2021; Received in revised form 4 January 2022; Accepted 21 January 2022

Available online 26 January 2022

1746-8094/© 2022 Elsevier Ltd. All rights reserved.

time [2]. Another essential diagnostic approach for COVID-19 detection is by using chest X-ray. Although, if the X-ray scans are not clear, COVID-19 is frequently misdiagnosed as another disease. As a result of the inaccurate diagnosis, patients are most times given wrong prescriptions, which ends up complicating their health status. This urgent health concern has led to a pressing need to create more accurate and automated diagnostic methods that use chest X-ray images to identify COVID-19.

Machine learning methods, most especially deep learning has been identified as a more accurate method as compared to other methods [3–5]. Authors in [3,4] diagnosed COVID-19 disease using deep learning methods from X-ray and CT scan images. Despite the fact that the proposed models in these research works was tested with a small number of X-ray images, especially the positive case images of COVID-19 patients, there was a class imbalance problem in earlier investigations. Furthermore, they used CNN exclusively as a deep learning model.

Similarly, authors in [6], focused on the classification of the COVID-19 epidemic by using only CNN and got less accurate result. Using only CNN is most times not sufficient for detecting COVID-19 accurately [7,8]. Therefore, numerous methods of improving the results obtained from previous CNN methods and their limitations are being explored.

This paper presents an effective classifier to classify cases of COVID-19 into two classes; either negative (normal) or positive (infected) by using chest X-ray images. In this regard, we proposed a deep learning approach for automated COVID-19 disease classification aimed at maintaining a high accuracy, and reducing false-negative cases. In general, threshold image segmentation was used to segment chest X-ray images of COVID-19 infected persons. Also we used anisotropic diffusion filtering (ADF) to remove noise and also compared its performance with other filters. A hybrid combination of CNN and HOG was used for feature extraction, YOLOv3 was used for object detection (whether the input image was chest X-ray or not), and SVM was used for classification.

This remaining parts of this paper are organized as follows. Section 2 presents a review of related works. Section 3 describes the methodology used in the development of the proposed COVID-19 disease classifier, while section 4 presents the results and discussion. Finally the paper is concluded in section 5.

2. Related works

In [9], the authors proposed deep a learning method for a chest X-ray image dataset. They used domain extension transfer learning (DETL) with a pre-trained deep CNN. The proposed system was specially implemented for detecting COVID-19 cases from chest X-ray images. The authors used Gradient Class Activation Map (Grad-CAM) for COVID-19 detection and to identify the areas where the model focused more during classification. An overall accuracy of $90.13\% \pm 0.14$ was obtained. Although, the authors used a small amount of data, the proposed method was still able to detect COVID-19.

Authors in [10] developed a system for the early prevention and detection of COVID-19 disease using image analysis. The authors used a deep learning model on a chest X-ray image dataset to determine the effects of COVID-19 on people who have pneumonia or lung illness. In the study, the authors proposed five pre-trained CNN based models (ResNet50, ResNet101, ResNet152, InceptionV3, and Inception-ResNetV2) for the detection of coronavirus pneumonia infected patients. The pre-trained ResNet50 model achieved the highest classification performance (96.1% accuracy for Dataset-1, 99.5% accuracy for Dataset-2, and 99.7% accuracy for Dataset-3) out of the four models utilized.

COVID-19 prediction from chest X-ray images using deep transfer learning was presented in [11]. The researchers employed a collection of X-ray images from a variety of sources to detect COVID-19 disease using transfer learning methods such as ResNet18, ResNet50, SqueezeNet, and DenseNet-121. They collected 250 X-ray images of COVID-19 infected patients who had tested positive for the virus. They tested the

models and found that they had a sensitivity rate of 98% and a specificity rate of roughly 90%. For feature extraction, they used CNN and pre-trained models, and got less accurate results. This however shows that using CNN alone is not sufficient for accurately detecting and recognizing COVID-19. The main reason for the poor results is that they used fewer images to train both the CNN and the pre-trained model, whereas generally, more images are required to accurately recognize COVID-19. By detecting patterns from past patient data and chest X-ray scans to diagnose COVID-19, the bulk of published research have predicted that the most likely outcome that a patient has contacted COVID-19 is based on their symptoms, travel history, and the delay in reporting the case. However, the previous works have some limitations such as: class imbalance in the manipulation of data, applying images to CNN without applying image preprocessing like equalization of the intensity, removing noise from the X-ray images, verification of the input image, and segmenting the region of interest. To overcome these drawbacks, this study applied a number of image preprocessing techniques, YOLOv3 for input object detection (used to check whether the given input image is a COVID-19 patient's image or another object image), and for segmenting the region of interest. Furthermore, HOG and CNN features were combined and classification was performed with SVM to improve the COVID-19 detection accuracy.

3. Methodology

3.1. System architecture

The major steps of the proposed method system architecture is shown in Fig. 1. The steps include image preprocessing, object detection, image segmentation, feature extraction, and COVID-19 disease classification. In the image processing part we used image size rescale and resize, normalized the images to a standard size, and employed filtering techniques for noise removal. Furthermore, YOLOv3 was used to detect if the input image is a chest X-ray image or not. Secondly, image segmentation was performed to identify the region of interest and distinguish the foreground image from the background image. Thirdly, feature learning was accomplished by extracting essential information from the input image using CNN and HOG. The CNN and HOG techniques were used to extract feature vectors from the chest X-ray images. The features obtained from these two techniques were combined and fed into the model as input in order to train the classification model. The combination of CNN and HOG feature extraction yields a large number of features for diagnosing if COVID-19 is present or not. In comparison to the original CNN, the combine features provide improved classification accuracy and require less number of learning iterations. From the preprocessing to the training, validation, and testing phases in CNN, sigmoid is used as a function learning technique. Finally, we used SVM to classify the image into one of two predetermined classes (Normal or COVID-19).

3.2. Data collection

A dataset was created comprising of 300 original images collected from the University of Gondar hospital in Ethiopia and 2200 images collected from the online repository used by authors in [12]. An image augmentation technique was applied to increase our dataset as the data was not adequate enough for the feature extraction stage. Image augmentation is typically applied to a dataset to expand the size of the training dataset by creating modified versions of the images in the dataset. The original images were transformed by shifts, flips, zooms, cropping the images, and rotating the images [13]. For this experiment, we have used a total of 6000 augmented images (3000 images of Normal and 3000 images of COVID-19 infected patients). The dataset was divided into training, validation, and test set.

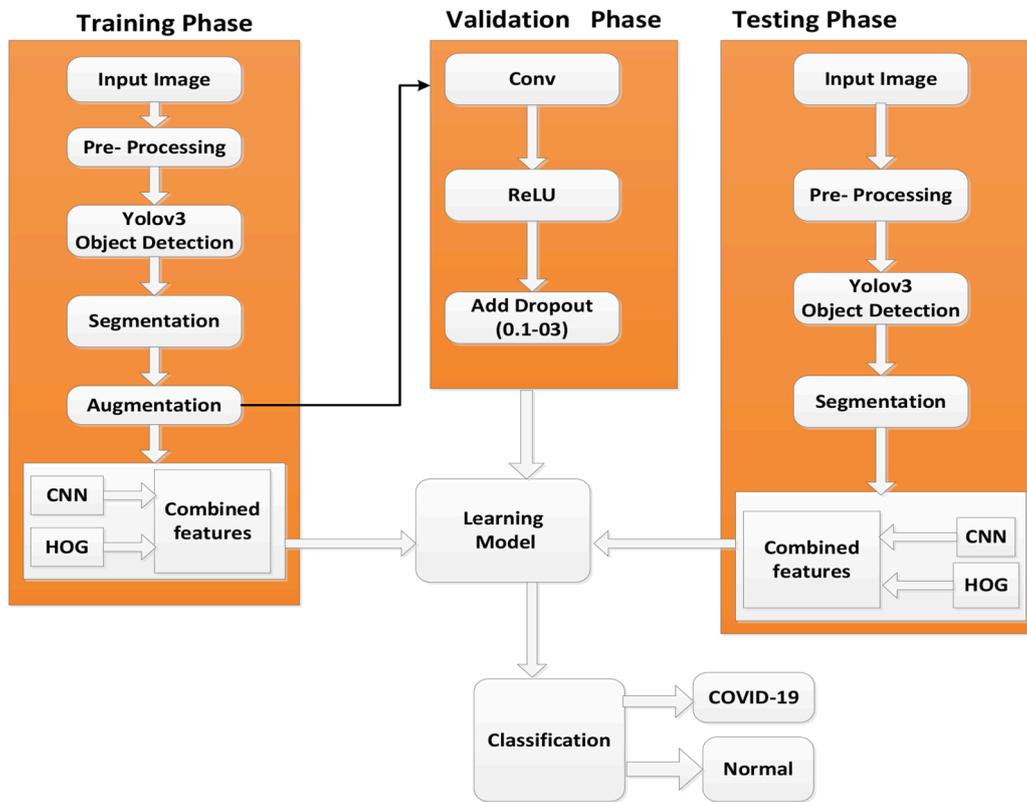


Fig. 1. Proposed system architecture.

3.3. Preprocessing

The first step performed in the detection and classification of COVID-19 is preprocessing of the chest X-ray images to improve their quality. This includes elimination of noise or unnecessary information from the chest X-ray images without obliterating essential information. In this phase, we resized the images of the dataset into 224×224 pixels to reduce the processing time and computational cost. Also, the images were converted into NumPy array which Keras can work with easily. OpenCV was used for image resizing and conversion into the NumPy array.

3.3.1. Histogram equalization

Histogram equalization was performed on the acquired images with a sample shown in Fig. 2. This method is one of the pixel brightness transformation techniques. It is one of the sophisticated methods for modifying the dynamic range and contrast of an image by altering the image such that its intensity histogram gives the desired shape.

Histogram equalization is a computer image processing technique for increasing image contrast. It accomplishes this task by effectively spreading out the most frequent intensity values, i.e. stretching out the intensity range of the image [14].

3.3.2. Anisotropic diffusion filter

Anisotropic diffusion filter (ADF) has been successfully employed in the field of image processing to remove noise while preserving the main edges of existing objects [15]. It is a technique aimed at reducing image noise without removing substantial parts of the image content such as edges, lines or other crucial elements. ADF is a popular medical image de-noising technique which has the ability to preserve micro texture information [16]. This method was employed to decrease diffusivity, while convolving the regions near the edges to minimize the blurring effect. This is described using Eq. (1).

$$\frac{\partial}{\partial t} u(\bar{x}, t) = \text{div}(c(\bar{x}, t) \nabla u(\bar{x}, t)) \quad (1)$$

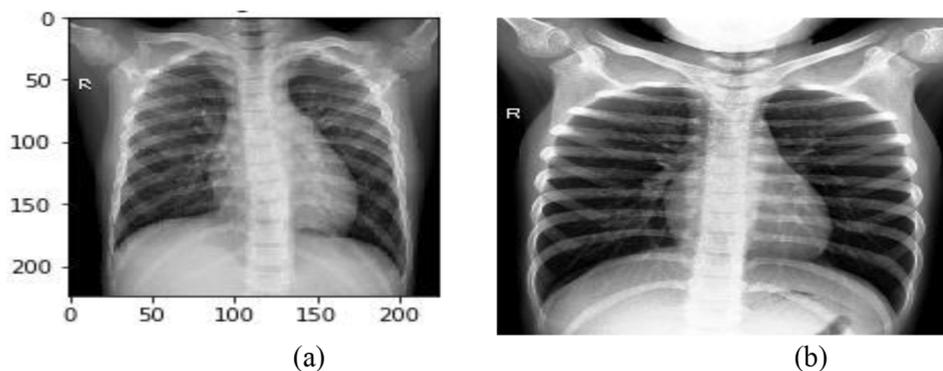


Fig. 2. (a) Original image, and (b) Histogram equalized image.

where, (x,t) is the divergence operator. It is a diffusion threshold function which denotes the gradient operator, while $\nabla u(x,t)$ is the image intensity. During the testing phase, images affected by Poisson noise were used to evaluate different filter performances. We tested several filtering methods such as arithmetic mean filter, contra harmonic mean filter, Gaussian filter, median filter, block-matching and 3D filtering (BM3D), bilateral filter, and anisotropic diffusion filter (ADF) to determine the most appropriate. Based on the Shannon entropy measure and signal to noise ratio (SNR), both the contra harmonic mean filter and ADF gave promising results, but the ADF outperformed other filters. Contra harmonic mean and ADF were found to be the best suited denoising methods for X-ray image noises among numerous filtering methods that were evaluated. Due to this reason, both the Contra harmonic mean filter and the ADF satisfy the Shannon entropy measure [17].

3.4. Object detection using YOLOv3 detector

Object detection entails identifying and classifying existing objects in a single image, as well as labeling them with rectangular bounding boxes to indicate their appearance. Before segmenting the chest X-ray images, we applied YOLOv3 for cross checking to determine if the input image is a chest X-ray image of a human lung or not. YOLO uses CNN to perform feature extraction, classification, and localization of an object. YOLOv3 improves the detection performance by training the entire dataset of images. This unified model has various advantages over the traditional object detection approaches. First and foremost, YOLOv3 is slightly faster and also does not require a complicated pipeline because we defined detection as a regression problem. During the training and testing stages, YOLOv3 identifies the complete image, so it implicitly encodes contextual information of the classes as well as their appearance. This is shown in Fig. 3.

3.5. Segmentation

Segmentation is the process of dividing an image into separate sections, each part representing homogeneous regions with similar features such as intensity, color, and texture. Thresholding is one of the common segmentation techniques. It operates on the basis of grey-level images. Thresholding was used to convert the grayscale images into black and white images out of a gray scale by setting exactly those pixels above a given threshold value to white. In this paper, thresholding-based segmentation was chosen because it is computationally fast and easy to implement [18]. However, since chest X-ray images are gray in color when converted into binary images, parts of the chest X-ray image were matched with the background. This is in contrast to effective segmentation which results in complete separation of the images (background and foreground) without information loss. In this paper, among the



Fig. 3. YOLO object detection to detect input chest X-ray image.

several thresholding techniques, we mainly applied Global thresholding. Local thresholding methods apply distinct threshold values to different regions of the image, whereas global thresholding methods apply a single threshold to the entire image. Global thresholding methods are extremely rapid and produce excellent results. Global thresholding using an appropriate threshold, T : by replacing each pixel in the image with a black pixel if the image intensity, $src(x, y)$ is less than some fixed constant, T (that is, $src(x, y) < T$), if the image intensity is greater than that constant. This is given by Eq. (2). The original and segmented image is shown in Fig. 4.

$$g(x, y) \begin{cases} 1 & \text{if } f(x, y) > T \\ 0 & \text{if } f(x, y) \leq T \end{cases} \quad (2)$$

3.6. Data augmentation

Data augmentation is a method that can be used to artificially increase the size of a training dataset. This helps train the model to enhance and augment its quality and aptness by producing updated versions of images in the dataset. Image data augmentation was used to increase the size of the training dataset in order to enhance the CNN models performance. The augmented data is fed into the CNN model to prevent overfitting of the data and to optimize the performance of the model. Data augmentation was performed after the images were pre-processed. To balance the insufficiencies of our dataset, we implemented a random transformation (such as zooming, rotation, shearing, and resizing) of our data [19]. The random rotation degree range was regulated by the rotation range parameter. In our case, we allowed a rotation of ± 25 degrees of the input image. The horizontal and vertical shifts were regulated by the width and vertical shift range. 25% of the input images was shifted horizontally and vertically. The shear range controls the angle in an anti-clockwise direction in radians in which our input image is allowed to be sheared. The zoom range value determines whether the input image is to be zoomed-in or zoomed out based on a regular distribution of values [zoom range -1 , zoom range $+1$].

3.7. Feature extraction

3.7.1. Feature Extraction Using CNN

Technically, the CNN model was used to extract features, train and validate each input image of the dataset which is passed through a series of layers which include the pooling layer and fully connected layers. Thereafter, the SVM classifier was used to classify the images into COVID-19 positive and COVID-19 negative. The deep learning approach called YOLOv3 was used to detect whether the given chest X-ray image has COVID-19 or not. To achieve this, CNN has been implemented with multiple convolutional layers followed by maxpooling layers. Subsequently, the output of the last convolutional and maxpooling layer is flattened and fed into a dense layer with 256 neurons. After being fed into the dense layer, then the Sigmoid Function is used for activation. The convolutional network layers provided by the Keras API of the TensorFlow library in Python were used to implement the sequential model. We noticed that adding more layers does not help our dataset's performance, but instead causes overfitting, higher memory consumption, and computation time. The following terms were considered for the CNN model.

- (i) **Pooling Layer:** After the convolutional layer, images are then fed into the maxpooling layer. The maxpooling layer has a window dimension of (2×2) filters. The pooling layer further helps to down-sample the input image. In other words, it helps to reduce the dimensions of the input image, thereby reducing the number of parameters of the image with the aim of reducing the computational complexity of the CNN model. The technique of sub-sampling used in the model is max-pooling and average-pooling. Max-pooling is a sample-based discretization process

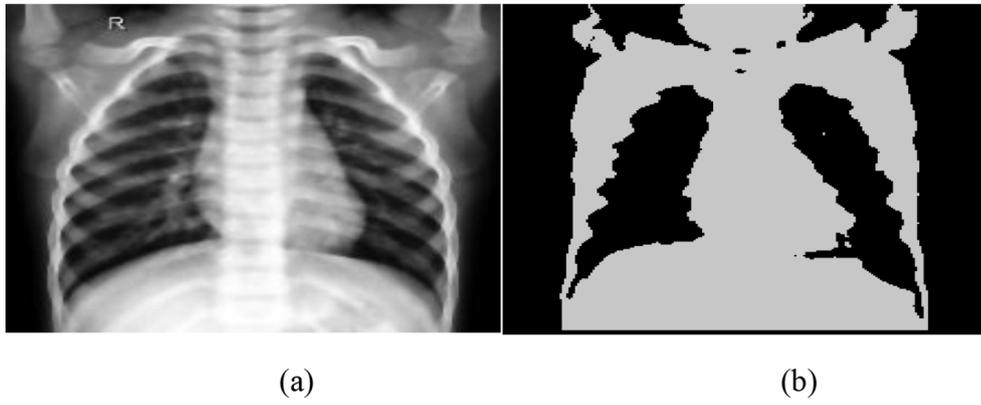


Fig. 4. (a) Original and (b) Segmented image.

which is the most common type of pooling layer. The pooling layer of dimension 2×2 works for each feature map and scales its dimensionality using the 'MAX' function. Two hyper parameters are necessary for the pooling layer, namely: filter (F) and stride (S). If the volume of the input image is $W1 \times H1 \times D1$, then an output of size $W2 \times H2 \times D2$ is produced by the pooling layer. The equations for W2, H2, and D2 in the pooling layer are given as follows:

$$W2 = ((W1 - F)/S) + 1 \quad (3)$$

$$H2 = ((H1 - F)/S) + 1 \quad (4)$$

$$D2 = K$$

where F denotes the filter size, S denotes the stride size, K denotes the number of filters applied. Note that, for all Conv2D layers, we used one Maxpool 2D layer.

- (ii) **Activation:** All the convolutional layers were activated by the Rectified Linear Unit (ReLU) function, which is conventionally used for classification. The ReLU function was used because it avoids and corrects the vanishing gradient problem. Neural network models that employ ReLU are easier to train and perform better than models that employ other activation functions such as sigmoid or hyperbolic tangent activation functions [20].
- (iii) **Pool Size Selection:** In the convolution module, we have used a 3x3 filter size in that layer. We selected the filter size systematically based on the characteristic features of each chest X-ray image recognized.
- (iv) **Flatten Layer and Fully Connected Layer:** After using the pooling layer, we applied a flatten layer to flatten the entire network in our model. The feature map that was pooled is straightened out into a column before being fed into the neural network. This allows the neural network to process the generated feature maps quickly. After the input image is fed to the convolutional, pooling, and flattening layers, it is then fed into the fully connected layer. The flatten layer is used to convert two-dimensional data to one dimensional data. Flatten layer was used before classification.
- (v) **Optimizer:** We have used Adaptive Moment Estimation (Adam) for training our models. In all of the models, Adam was used. Adam is a computer program that does optimizations. The Adam optimizer was used to adjust the network's weights iteratively based on the training data. The Adam optimizer is beneficial for networks training on huge datasets or parameters. It is simple to build, computationally efficient, and takes a tiny amount of memory, which is why it was chosen for this work [21].

- (vi) **Reducing Overfitting:** Neural networks trained on relatively small datasets most times overfit their training data. Dropout is mainly used to minimize overfitting by randomly disconnecting inputs from the previous layer to the next layer in the network architecture. Dropout is implemented per layer in a neural network. We have applied dropout layers with P (dropping probability) value of 0.2, 0.25, and 0.3 immediately before the fully connected layers at each convolution, which is followed by the SVM classifier.

3.7.2. Feature extraction using HOG

We used HOG feature extraction in addition to CNN. The reason for employing HOG in combination with CNN is that HOG hardware implementations are commonly known to be more energy-efficient than CNN features [22]. In most cases, neural networks are more computationally expensive and produce more image redundancy than standard techniques. State-of-the-art deep learning algorithms that accomplish successful CNN training can take a lengthy time to train from the scratch. HOG algorithms, on the other hand, take less time to train, ranging from a few minutes to a few hours [23]. HOG is one of the well-recognized feature extractors due to its superior performance and relatively simple computation. HOG-based algorithms are still favorable in many applications due to their balanced tradeoff between accuracy and complexity. HOG features have been extracted for each pre-processed image of dimension (64×128) in this work. For object detection, HOG descriptors are widely employed in computer vision and image processing. HOG is used to extract characteristics about an object's look and shape from a distribution of local gradients. It is also capable of defining a distinct texture and shape [24,25].

3.7.3. Combined feature extraction

After CNN and HOG feature extraction, the model shows that the CNN feature extractors provide a feature vector of output of 13,824 from the CNN model, while the HOG feature extractors provide a feature vector of 3780. Then the features extracted using HOG and CNN are combined to give 17,604 features. Feature combination helps to fully learn image features to give a description of their rich internal information. We combine the features of the output of the HOG feature extraction with the output of the CNN feature extraction using hstack function [26]. After combining the features of the output, we save the output in CSV format. Afterwards, we split it into training, validation, and testing set by assigning 0 for COVID-19 affected persons and 1 for Normal as a target class, and then apply SVM and random forest classifier for classification. The combined integrated features results give a better performance of COVID-19 recognition.

3.7.4. Classification

Support Vector Machines are typically thought of as a classification

method, but they can be used to solve both classification and regression problems. They can handle both continuous and categorical variables easily. SVM constructs a hyperplane in a multidimensional space to separate different classes. SVM iteratively produces the best hyperplane, which is then used to minimize errors in classification. The features obtained after the features were concatenated are used to train the Linear SVM, Sigmoid kernel function, polynomial kernel function, RBF, and random forest classifiers using binary classification. We classified each image in the test dataset into a predefined class (COVID-19 or Normal) using the knowledge of the learning model. Thereafter, a comparison between, random forest, Linear SVM, Sigmoid kernel function, polynomial kernel function, and RBF was performed. Linear kernel function was finally chosen because it achieved better classification results as a final kernel function than others.

3.8. Evaluation techniques

Cross-validation was not used in this study. Instead of the cross-validation method, the holdout validation technique was used since the dataset has enough samples for training and testing. The performance of the CNN, HOG, and hybrid model over the testing dataset was determined after the completion of the training of the model. The performance was measured using the most widely used metrics such as accuracy, precision, sensitivity or recall, and F1 score. Accuracy is a measure of the validation (training) accuracy or classification accuracy of the model. With the help of a confusion matrix, the number of true positives, true negatives, false positives, and false negatives can be calculated, which further helped in checking the efficacy of the proposed model. The Eqs. (6–9) present the formulas for calculating the mentioned metrics.

Accuracy: This is the amount of correct predictions made compared to the total number of predictions made. This can be calculated using Eq. (6).

$$\frac{TP + TN}{TP + FP + FN + TN} \quad (6)$$

Precision: It is used to calculate how many cases are predicted as positive by the model were actually supposed to be predicted as positive. It is defined as the ratio of predicted true positives to the total number of predicted positives, and is given as

$$\frac{TP}{TP + FP} \quad (7)$$

Recall: It is used to calculate how well the model has classified the positive examples. It is defined as the ratio of predicted true positives to the actual positives, and is given as

$$\frac{TP}{TP + FN} \quad (8)$$

The F1 score can be interpreted as a weighted harmonic mean of the precision and recall and is defined as

$$F1 \text{ score} = \frac{2 * (\text{Recall} * \text{Precision})}{\text{Recall} + \text{Precision}} \quad (9)$$

To use the Eqs. ((6)–(9)), we classify the chest X-rays into 4 major categories:

- False Negative (FN)
- False Positive (FP)
- True Negative (TN)
- True Positive (TP)

The true positive metrics denotes that the model correctly predicts a case of COVID-19. True Negative signifies a case where it correctly predicts when a person is not affected by COVID-19. False Negative signifies that the model misinterprets a person having COVID-19 as being benign. False Positive signifies that a person, not affected by COVID-19 is classified as having it.

4. Model evaluation and discussion

In this section, we discuss the 5 different experiments performed to evaluate the model.

4.1. Dccnet model evaluation using preprocessed images at training and validation phase

In experiment 1, the proposed DCCNet model achieves 99% training accuracy and 96.8% testing accuracy as shown in Fig. 5. This classification accuracy was achieved when the model is trained without segmentation, data augmentation, and dropout at the beginning. The simulation takes 37 min to train (45 s per each epoch (50)). Fig. 5 shows that the validation accuracy is less than the training accuracy, which indicates that overfitting occurred. In addition to the fact that the validation loss is significantly higher than the training loss, it shows our model is overfitted. As seen in Fig. 5, the training and testing accuracy increases while training loss decreases almost linearly, but validation loss varies until it reaches epoch 28, then it increases linearly. This can also be seen in Fig. 6 with the training loss and accuracy curve. Adding dropout and increasing the dataset by using image augmentation were used in the training and validation phases to overcome these issues (mitigate overfitting), as shown in Fig. 7.

4.2. Feature extraction using HOG before image segmentation

In experiment 2, we used a 8x8 cell size to extract the HOG descriptors from the chest X-ray images which were then feed into the random forest and SVM classifiers. We observed that the training accuracy is 100%, and it increases significantly compared to that of experiment 1 and the testing accuracy is 97.84%, which is almost the same with experiment 1. The result shows that the performance of the testing accuracy is less than that of the training accuracy due to overfitting. In this experiment (experiment 2), 509 testing images were employed and divided into two; COVID-19 class (240 images) and normal class (269 images). Fig. 7 shows the confusion metrics for the generalization outcomes. Only 6 images out of the 240 COVID-19 images are miss detected.

4.3. Image segmentation using DCCNet model at the training and validation phases

In experiment 3, the model was trained using the augmented images, while segmentation was performed with ReLU as an activation function during the validation phase. The thresholding segmentation technique was utilized to separate the ROI from the background and remove unwanted parts of the X-ray image. This was aimed at making the learning

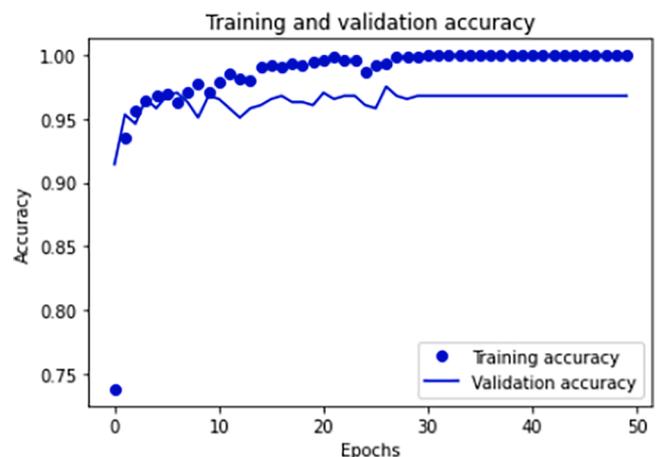


Fig. 5. Training and validation accuracy.

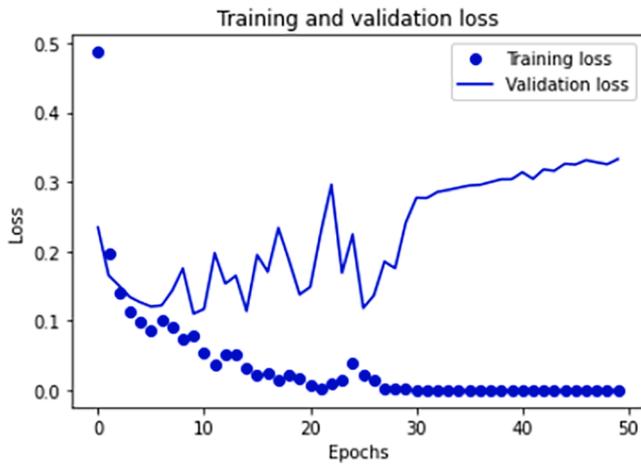
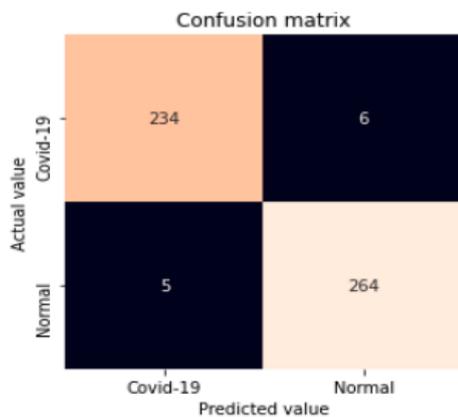


Fig. 6. Training and validation loss.

of features from the input image easier and to make the analysis of X-ray images faster. Figs. 8, 9, and 10 show the model’s training and validation accuracy as well as loss progress following image segmentation. The model takes approximately 1 h and 38 min (98 min) to train, since it takes 117 s on average per epoch. This findings demonstrates that image segmentation is important in improving the models performance. We were able to attain 99.9% training accuracy, this was because the model fits with the segmented dataset; both the CNN classifier and random forest achieved 98.3%, linear SVM achieved 98.1%, RBF achieved 98.5%, polynomial kernel function achieved 97.7%, and kernel with sigmoid achieved 96.8% testing accuracy.

As seen in Fig. 9 and Fig. 10, the number of epochs increases as the training and testing accuracy improves, while simultaneously, the training and validation loss decreases. The lower the loss, the better the models recognition results. When we compared experiment 1, we have got 1.5%, experiment 2 gave a better performance of 0.5%. The results show that the testing accuracy is less than the training accuracy, which indicates that overfitting occurred. In Fig. 8, the results show that the precision, micro average, recall, and F1-score for experiment 3 achieves a test accuracy of 98.3% and loss of 0.064%. This result is better, compared to that of experiment 1 and 2. Based on this, we have generated the testing, precision, recall and F1-score of the COVID-19 disease class and normal class. After applying thresholding-based image segmentation and augmentation, the model’s performance improved due to data augmentation and dropout as seen in Figs. 9 and 10. No overfitting was experienced as the training curves were closely tracking the validation curves. The training accuracy is relatively



(a)

greater than the validation accuracy throughout the curve. Furthermore, across the curve, the training loss is smaller than the validation loss.

4.4. Feature extraction using HOG after image segmentation

In experiment 4, we extracted features using HOG. After applying image augmentation, we used a 8x8 cell size to extract the HOG descriptors from the chest X-ray images and thereafter these features were then feed into the SVM classifier. We observed that the testing accuracy

Test loss: 0.06407415121793747
 Test accuracy: 0.9833333492279053

	precision	recall	f1-score	support
Covid (Class 0)	0.99	0.98	0.98	600
Normal (Class 1)	0.98	0.99	0.98	600
micro avg	0.98	0.98	0.98	1200
macro avg	0.98	0.98	0.98	1200
weighted avg	0.98	0.98	0.98	1200

Fig. 8. Training and Validation progress after image segmentation and augmentation.

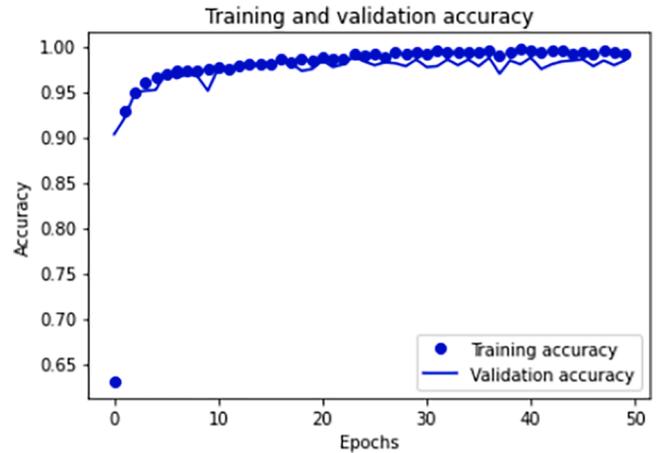


Fig. 9. Training and Validation accuracy after segmentation and image augmentation.

Number of data
 Covid-19: 1201
 Normal: 1340
 Validation Accuracy of SVM Algorithm: 99.51%
 Test Accuracy of SVM Classifier: 97.84%
 Train Accuracy of SVM Classifier: 100.00%

	precision	recall	f1-score	support
Covid-19	0.98	0.97	0.98	240
Normal	0.98	0.98	0.98	269
micro avg	0.98	0.98	0.98	509
macro avg	0.98	0.98	0.98	509
weighted avg	0.98	0.98	0.98	509

(b)

Fig. 7. (a) Confusion Matrix and (b) Classification accuracy of HOG before applying image augmentation.

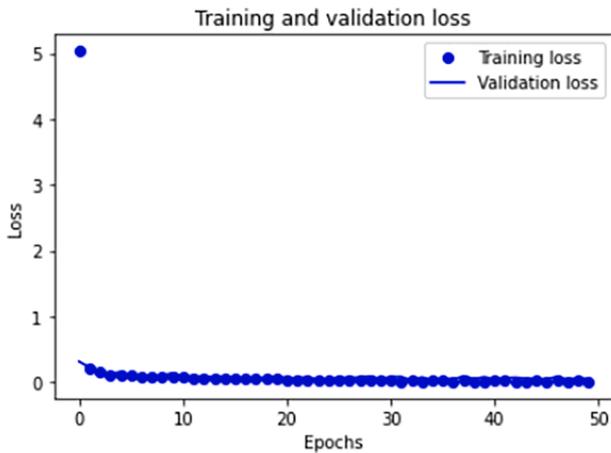


Fig. 10. Training and Validation loss after segmentation and image augmentation.

increases by 1.7% compared to that of experiment 1, and by 0.7% compared to that of experiment 2, and 0.2% compared to that of experiment 3. The training accuracy achieved was 100% and testing accuracy is 98.5% which is almost similar to the results achieved in experiment 3. The results show that the training accuracy is greater than the testing accuracy, which indicates that overfitting occurred. 1200 testing images were employed and divided into two: COVID-19 class (613 images) and normal class (587 images). Fig. 11 shows the confusion metrics for the obtained outcomes. Only 13 COVID-19 images are miss detected.

4.5. Combined features used for training and validation phase after segmentation

In experiment 5, we combined both the CNN and HOG features after applying image augmentation. The combination of CNN and HOG-based features employed with the SVM classifier achieved 99.97% training accuracy and 99.67% testing accuracy. The experimental results indicate that the results achieved with the CNN and HOG with SVM classifier technique was better than the results achieved in the other four experiments for the recognition of COVID-19 disease. Hence, the training accuracy, validation accuracy, and testing accuracy are almost equal; therefore there is no overfitting and underfitting problem with this experiments results. The results of various separate feature extraction methods were less adequate when compared to the combined method. This shows that the proposed method accurately categorized COVID-19 cases compared to single feature extraction methods. In order to achieve

generalization, 1200 testing images were used and divided into two: COVID-19 class (613 images) and normal class (587 images). Fig. 12 shows the confusion metrics for the obtained outcomes. Only 4 images out of 613 COVID-19 images are miss detected. The COVID-19 chest X-ray images had the lowest false detection rate, with a testing accuracy of 99.67%. This indicates that the image classification relies on the model even with a completely new set of data. The wrong detection rate experienced in other four experiments was reduced in this experiment (experiment 5).

The proposed technique, which combined HOG and CNN feature fusion, outperformed linear SVM classifier, RBF, poly, random forest, and CNN classifiers. For the chest X-ray dataset, linear SVM performs well for the binary class. When the classification accuracies were obtained with features retrieved using CNN or HOG separately, the results were substantially lower than when combined. Therefore, the combined feature method provided a higher testing accuracy compared to the separate feature extraction. The combined model didn't overfit, and achieved no significant difference between the results of the testing accuracy and training accuracy as shown in Fig. 12. Table 3 summarizes the performance indices for the different experiments. The combined feature extractor after image segmentation outperforms other models with respect to the different performance indices.

4.6. Discussion

4.6.1. Comparison with AlexNet implementation result

AlexNet uses deep layers with 650 k neurons and 60 million parameters to categorize over 1000 different classes with five convolutional layers with three pooling layers, and two fully connected layers. The AlexNet input image must have a dimension of $227 \times 227 \times 3$ pixels, and the first convolutional layer transforms the input image into 96 kernels sized at $11 \times 11 \times 3$ pixels with a stride of 4 pixels, which serves as the input to the second layer [15].

Deep neural network models include AlexNet, ResNet, VGG16, VGG19, GoogleNet, etc. From among these models, we chose AlexNet to test our model. The AlexNet model took around 9 h and 23 min to train, with 675 s for each epoch (50), with a training accuracy of 96.9% and testing accuracy of 92.5%. This is lower than our DCCNet model, which has a training and testing accuracy of 99.9% and 98.3%, respectively. It also takes a lot more time to compute than the DCCNet model. As shown in Fig. 13, the AlexNet model training accuracy is higher than its testing accuracy throughout the curve, which shows there is substantial overfitting. The graphical representation of the model accuracy and model loss for AlexNet are shown in Fig. 13 and Fig. 14. These figures show the variation in the training and validation accuracies and loss decrease from epoch to epoch. In Fig. 14, we clearly see that the result of the

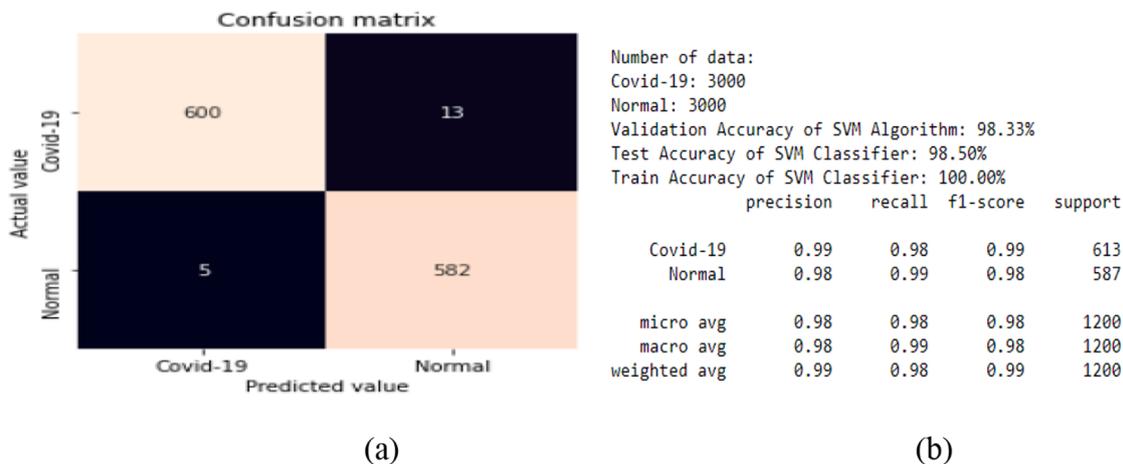
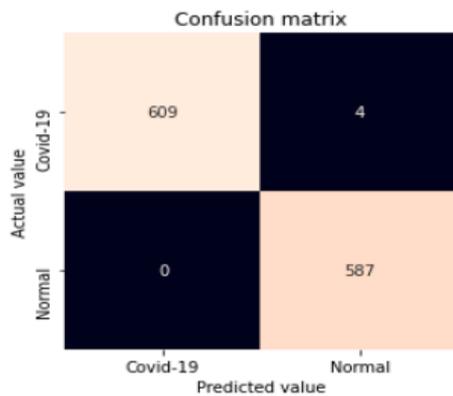


Fig. 11. (a) Confusion Matrix and (b) Classification accuracy of HOG while applying image augmentation.



(a)

Number of data
 Covid-19: 3000
 Normal: 3000
 Validation Accuracy of SVM Algorithm: 99.58%
 Test Accuracy of SVM Classifier: 99.67%
 Train Accuracy of SVM Classifier: 99.97%

	precision	recall	f1-score	support
Covid-19	1.00	0.99	1.00	613
Normal	0.99	1.00	1.00	587
micro avg	1.00	1.00	1.00	1200
macro avg	1.00	1.00	1.00	1200
weighted avg	1.00	1.00	1.00	1200

(b)

Fig. 12. (a) Confusion Matrix and (b) Classification accuracy of combined feature extraction.

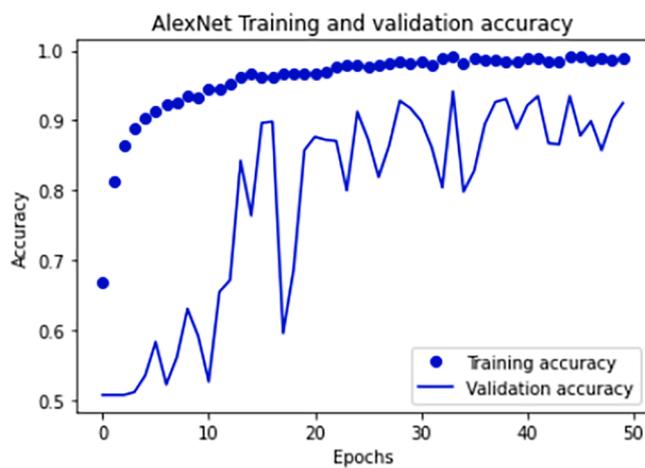


Fig. 13. Training and validation accuracy of AlexNet.

training loss curve closely tracks the validation loss curve. The training loss and validation loss decreases down to 0.8. As a result, the proposed DCCNet and hybrid approach model outperforms the AlexNet model. The training accuracy was quite substantial and increased linearly reaching 96.8%. However, the validation accuracy oscillates between 0.7 and 0.85 and does not become stable. Overfitting was observed in the AlexNet model in terms of the training accuracy being greater than the testing accuracy.

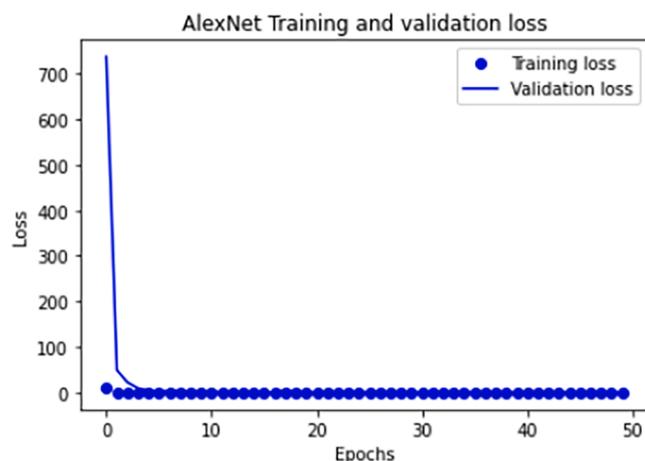


Fig. 14. Training and validation loss of AlexNet.

Firstly, the model was trained using the original data without applying any image augmentation technique to remove noise by using the anisotropic diffusion filter, with Adam optimizer and ReLU as activation function. The end-to-end training approach was adopted to classify the acquired images. Secondly, we applied HOG feature extraction without applying any image augmentation technique and crosschecked the results with three different experiments. In the third experiment, we used image augmentation and segmentation which gives a better accuracy than the original dataset result. As mentioned before, image augmentation methods were used to equalize the number of samples over the classes. Here, the number of the normal class with the augmentation technique was balanced by the number of the COVID-19 classes. Experiment 3 shows that the augmentation method contributed to improving the classification success and reduced over-fitting. Fourthly, we applied HOG feature extraction on the image augmentation and segmentation data to achieve a better result. In the fifth experiment, we combined CNN and HOG features together to solve the overfitting problem, and achieved a more accurate result. CNN + HOG approach resulted in an optimal solution for classifying COVID-19 and normal chest X-ray images.

From the results of our simulation, the computation time for both HOG experiments was significantly smaller than that of the DCCNet model as shown in Table 1. This is because HOG feature extraction is not hierarchical and does not use much hyperparameters as CNN. The hybrid model achieves a higher classification accuracy than the AlexNet. However, AlexNet uses a larger number of parameters, with a risk of overfitting. Table 1 shows the results of the training and validation performance of the model, while in Table 2, the testing performance of the various models is presented. A summary of all the results is presented in Table 3, while in Table 4, we present a comparison of the proposed method with existing methods in literature. The results show that the proposed hybrid method outperforms other existing methods for the

Table 1
 Training and validation performance of the models.

Model	Training accuracy (%)	Training time (minutes)	Validation accuracy (%)
DCCNet (before image segmentation)	99	37	96.8
HOG (before image segmentation)	100	10	99.5
DCCNet (after applying segmentation)	99.9	98	98.4
HOG (after image segmentation)	100	6	98.3
Combined of CNN and HOG	99.97	7	99.5
AlexNet	96.9	563	92.5

Table 2
Performance evaluation of the various models and classifiers.

Model	Classifier	Testing accuracy (%)
DCCNet (before image segmentation)	Linear SVM	97.0
	Sigmoid in CNN	96.8
	RF	97.2
HOG (before image segmentation)	Linear SVM	97.8
	RF	97.2
DCCNet (after segmentation)	Linear SVM	98.1
	Sigmoid in CNN	98.3
	RF	98.3
HOG (after segmentation)	Linear SVM	98.5
	RF	97.5
Combined of CNN and HOG	Linear SVM	99.67
	RF	98.6
AlexNet	Linear SVM	92.9
	Sigmoid in CNN	92.5
	RF	92.0

detection and classification of COVID-19.

5. Conclusion

A number of studies have been conducted on COVID-19 disease classification. In this paper, we presented a deep learning method for fast detection and classification of COVID-19 disease. Convolutional Neural Network (CNN) and Histogram of Oriented Gradients (HOG) were utilized for feature extraction, and Support Vector Machine (SVM) was employed for classification. The primary goal of this work is to improve medical proficiency in areas where the number of radiotherapists are limited. Our study facilitates the early identification of COVID-19 to prevent adverse consequences, such as death. We were able to achieve 99.9% training accuracy for the DCCNet model, 98.3% for Sigmoid CNN classifier, 98.3% for random forest, and 98.1% testing accuracy for linear SVM. In addition, 100% training accuracy was achieved for HOG, while 98.5% was achieved for linear SVM, and 97.5% testing accuracy for random forest. The most efficient results were obtained by combining all features provided by CNN and HOG. The hybrid method provided the best performance with a training accuracy of 99.97%, and 99.67% testing accuracy on the acquired dataset for COVID-19 detection and recognition which is higher than the recognition capability of other state-of-the-art methods. The CNN architecture (AlexNet) achieved a training accuracy of 96.9%, while the testing accuracy of linear SVM is 92.9%, sigmoid CNN classifier achieved 92.5%, while 92% was achieved for random forest.

Table 3
Model performances using Precision, Recall, and F1-Score.

Model	Criterion	COVID-19	Normal
DCCNet Model Using Preprocessed Images	Precision	0.98	0.96
	Recall	0.95	0.99
	F1-Score	0.97	0.97
Feature Extraction Using HOG before Image Segmentation	Precision	0.98	0.98
	Recall	0.97	0.98
	F1-Score	0.98	0.98
DCCNet Model Using Image Segmentation	Precision	0.99	0.98
	Recall	0.98	0.99
	F1-Score	0.98	0.98
Feature Extraction Using HOG after Image Segmentation	Precision	0.99	0.98
	Recall	0.98	0.99
	F1-Score	0.99	0.98
Combined Feature after Segmentation	Precision	1.00	0.99
	Recall	0.99	1.00
	F1-Score	1.00	1.00
AlexNet	Precision	0.97	0.95
	Recall	0.95	0.98
	F1-Score	0.96	0.96

Table 4
Comparison of the proposed hybrid method with existing methods for the detection of COVID-19.

Authors	Purpose	Features extraction method	Classifier	Accuracy (%)
[27]	Detection/Classification	Not mentioned (NM)	Deep learning	86.7
[28]	Detection	Not mentioned (NM)	Deep learning	89.5
[29]	Detection/Classification	Grey Level Co-occurrence Matrix (GLCM), Local Directional Pattern (LDP), Grey Level Run Length Matrix (GLRLM), Grey Level Size Zone Matrix (GLSZM) and DWT	SVM	over 90
[30]	Classification	Not mentioned	Linear Discriminant (LD), Linear SVM, Quadratic SVM, Fine KNN, Subspace Discriminant (SD), and Subspace KNN SVM	98.1
[31]	Detection/Classification	DWT	SVM	98.2
[32]	Detection	DenseNet121	Bagging tree classifier	99
Implemented work	Detection/Classification	CNN and HOG	SVM	99.67

5.1. Recommendation for future work

From the work presented, there are still some research gaps which can be filled in future works. To increase the performance of the proposed model, the proposed method can be expanded upon. As a result, when carrying out future works, the following are some noteworthy recommendations:

- The proposed model can be improved further to offer stage wise diagnosis of COVID-19.
- The hybrid model can be used for other medical applications and diagnosis such as bone suppression in the chest cavity, diagnosing pneumonia, lung cancer, and other respiratory disorders.
- In the dataset, there are some small boxes indicating the presence of COVID-19 in the chest X-ray using YOLO. For future works, these bounding boxes can be used to train the CNN and not only to identify the chest X-ray images but also for identifying whether the given input image is either COVID-19 or other diseases such as pneumonia, lung cancer, or lung lesions.

CRediT authorship contribution statement

Aleka Melese Ayalew: Conceptualization, Methodology, Software, Writing – original draft. **Ayodeji Olalekan Salau:** Data curation, Methodology, Software, Writing – review & editing. **Bekalu Tadele Abeje:** Visualization, Investigation. **Belay Enyew:** Visualization, Investigation, Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial

interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] Advice for the public on COVID-19 – World Health Organization. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public> (accessed Nov. 03, 2021).
- [2] Getting safe emergency care during the COVID-19 (coronavirus) pandemic - Mayo Clinic.html. <https://newsnetwork.mayoclinic.org/discussion/getting-safe-emergency-care-during-the-covid-19-pandemic/> (2021).
- [3] R. Sethi, M. Mehrotra, D. Sethi, Deep Learning based Diagnosis Recommendation for COVID-19 using Chest X-Rays Images, in: 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2020, pp. 1–4, <https://doi.org/10.1109/ICIRCA48905.2020.9183278>.
- [4] H. Jiang, S. Tang, W. Liu, Y. Zhang, “Deep learning for COVID-19 chest CT (computed tomography) image analysis: A lesson from lung cancer, Comput. Struct. Biotechnol. J. 19 (2021) 1391–1399, <https://doi.org/10.1016/j.csbj.2021.02.016>.
- [5] R. Sadik, L. Reza, A.A. Noman, S.A. Mamun, M.S. Kaiser, M.A. Rahman, COVID-19 pandemic: a comparative prediction using machine learning, Int. J. Auto AI. Mach. Learn 1 (1) (2020) 16.
- [6] N. Wang, H. Liu, C. Xu, Deep Learning for The Detection of COVID-19 Using Transfer Learning and Model Integration, in: 2020 IEEE 10th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 2020, pp. 281–284, <https://doi.org/10.1109/ICEIEC49280.2020.9152329>.
- [7] H. Jiang, S. Tang, W. Liu, Y. Zhang, Deep learning for COVID-19 chest CT (computed tomography) image analysis: A lesson from lung cancer, Comput. Struct. Biotechnol. J. 19 (2021) 1391–1399, <https://doi.org/10.1016/j.csbj.2021.02.016>.
- [8] N.-A.-A. Alam, M. Ahsan, M.A. Based, J. Haider, M. Kowalski, COVID-19 Detection from Chest X-ray Images Using Feature Fusion and Deep Learning, Sensors 21 (4) (2021) 1480, <https://doi.org/10.3390/s21041480>.
- [9] S. Basu, S. Mitra, N. Saha, Deep Learning for Screening COVID-19 using Chest X-Ray Images, in: 2020 IEEE Symposium Series on Computational Intelligence (SSCI), 2020, pp. 2521–2527, <https://doi.org/10.1109/SSCI47803.2020.9308571>.
- [10] A. Narin, C. Kaya, Z. Pamuk, Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks, Pattern Anal. Appl. 24 (3) (2021) 1207–1220, <https://doi.org/10.1007/s10044-021-00984-y>.
- [11] S. Minaee, R. Kafieh, M. Sonka, S. Yazdani, G. Jamalipour Soufi, Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning, Med. Image Anal. 65 (2020) 101794, <https://doi.org/10.1016/j.media.2020.101794>.
- [12] J.P. Cohen, P. Morrison, L. Dao, “Covid-19 image data collection,” arXiv: 2003.11597, 2020. URL: <https://github.com/ieee8023/covid-chestxray-dataset>.
- [13] A.O. Salau, S. Jain, Adaptive Diagnostic Machine Learning Technique for Classification of Cell Decisions for AKT Protein, Inf. Med. Unlocked 23 (1) (2021) 1–9, <https://doi.org/10.1016/j.imu.2021.100511>.
- [14] Histogram Equalization: Image Contrast Enhancement| What is Histogram Equalization? <https://www.mygreatlearning.com/blog/histogram-equalization-explained/> (accessed Mar. 15, 2021).
- [15] Anisotropic Diffusion Filtering Operation and Limitations - Magnetic Resonance Imaging Evaluation,” IFAC Proc. 47(3), 3887–3892, 2014. DOI: 10.3182/20140824-6-ZA-1003.02347.
- [16] J. Joseph, R. Periyasamy, A polynomial model for the adaptive computation of threshold of gradient modulus in 2D anisotropic diffusion filter, Optik 157 (2018) 841–853, <https://doi.org/10.1016/j.ijleo.2017.11.177>.
- [17] A. Swain, “Noise filtering in Digital Image Processing,” Image Vision, Jul. 16, 2020. <https://medium.com/image-vision/noise-filtering-in-digital-image-processing-d12b5266847c> (accessed Nov. 30, 2021).
- [18] S. N, V. S, Image segmentation by using thresholding techniques for medical images, Comput. Sci. Eng. Int. J. 6 (1) (2016) 1–13, <https://doi.org/10.5121/csej.2016.6101>.
- [19] How to Configure Image Data Augmentation in Keras. <https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/> (accessed Mar. 17, 2021).
- [20] J.N. Hasoon, A.H. Fadel, R.S. Hameed, S.A. Mostafa, B.A. Khalaf, M.A. Mohammed, J. Nedoma, COVID-19 anomaly detection and classification method based on supervised machine learning of chest X-ray images, Results Phys. 31 (2021) 105045, <https://doi.org/10.1016/j.rinp.2021.105045>.
- [21] N. Indumathi, M. Shanmuga Esvari, A.O. Salau, R. Ramalakshmi, R. Revathy, Prediction of COVID-19 Outbreak with Current Substantiation Using Machine Learning Algorithms, in: Intelligent Interactive Multimedia Systems for e-Healthcare Applications, Springer Singapore, Singapore, 2022, pp. 171–190, https://doi.org/10.1007/978-981-16-6542-4_10.
- [22] A. Suleiman, Y.H. Chen, J. Emer, and V. Sze, “Towards Closing the Energy Gap Between HOG and CNN Features for Embedded Vision,” ArXiv170305853 Cs, Mar. 2017, Accessed: May 17, 2021. [Online]. Available: <http://arxiv.org/abs/1703.05853>.
- [23] “4 Reasons Why Deep Learning and Neural Networks Aren’t Always the Right Choice,” Built In. <https://builtin.com/data-science/disadvantages-neural-networks> (accessed Jun. 07, 2021).
- [24] A.O. Salau, S. Jain, “Feature Extraction: A Survey of the Types, Techniques, and Applications,” in: 5th IEEE International Conference on Signal Processing and Communication (ICSC), Noida, India, 158-164, 2019. doi: 10.1109/ICSC45622.2019.8938371.
- [25] B. Li, K. Cheng, Z. Yu, Histogram of Oriented Gradient Based Gist Feature for Building Recognition, Comput. Intell. Neurosci. 2016 (2016) 1–9, <https://doi.org/10.1155/2016/6749325>.
- [26] X. Zhang, et al., A hardware-oriented histogram of oriented gradients algorithm and its VLSI implementation, Jpn. J. Appl. Phys. 56 (4) (2017), <https://doi.org/10.7567/JJAP.56.04CF01>.
- [27] X. Xu et al., “Deep learning system to screen coronavirus disease 2019 pneumonia,” 1-29, 2020.
- [28] S. Wang et al., “A deep learning algorithm using CT images to screen for corona virus disease (COVID- 19),” 2020. doi: 10.1101/2020.02.14.200230284.
- [29] M. Barstugan, U. Ozkaya, S. Ozturk, “Coronavirus (COVID-19) classification using CT images by machine learning methods,” 1-10, 2020.
- [30] S. Gurjit et al., “Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: COVID-19 case study,” 1-26, 2020. doi: 10.1101/2020.02.03.932350.
- [31] A.O. Salau, Detection of Coronavirus Disease Using a Novel Machine Learning Approach. 2021 International Conference on Decision Aid Sciences and Application (DASA), Sakheer Bahrain, 2021, pp. 587–590, <https://doi.org/10.1109/DASA53625.2021.9682267>.
- [32] S.H. Kassania, P.H. Kassanib, M.J. Wesolowski, K.A. Schneidera, R. Detersa, Automatic Detection of Coronavirus Disease (COVID-19) in X-ray and CT Images: A Machine Learning Based Approach, Biocybernet. Biomed. Eng. 41 (3) (2021) 867–879, <https://doi.org/10.1016/j.bbe.2021.05.013>.