

COVID-19 Detection on Chest x-ray Images by Combining Histogram-oriented Gradient and Convolutional Neural Network Features

Reda Elbarougy^{1,*}, Esmail Aboghrara², G. M. Behery¹, Y. M. Younes³, and Noha M. El-Badry³

¹Department of Information Technology, Faculty of Computer and Artificial Intelligence, Damietta University, New Damietta, Egypt

²Department of Computer Science, Faculty of Sciences, Sebha University, Sebha, Libya

³Department of Mathematics, Faculty of Sciences, Damietta University, New Damietta, Egypt

Received: 1 Feb. 2023, Revised: 22 Mar 2023, Accepted: 11 Mar. 2023.

Published online: 1 May 2023.

Abstract: The COVID-19 coronavirus epidemic has spread rapidly worldwide after a person became infected with a severe health problem. The World Health Organization has declared the coronavirus a global threat (WHO). Early detection of COVID 19, particularly in cases with no apparent symptoms, may reduce the patient's mortality rate. COVID 19 detection using machine learning techniques will aid healthcare systems around the world in recovering patients more rapidly. This disease is diagnosed using x-ray images of the chest; therefore, this study proposed a machine vision method for detecting COVID-19 in x-ray images of the chest. The histogram-oriented gradient (HOG) and convolutional neural network (CNN) features extracted from x-ray images were fused and classified using support vector machine (SVM) and softmax. The proposed feature fusion technique (99.36 percent) outperformed individual feature extraction methods such as HOG (87.34 percent) and CNN (93.64 percent).

Keywords: Coronavirus, Chest x-ray, Deep Learning, COVID-19, Convolutional Neural Network (CNN), Histogram Oriented Gradient (HOG), Radiology images.

1 Introduction

COVID-19 is a fatal coronavirus infection caused by a newly discovered coronavirus. In December 2019, the SARS-COV-2 virus infected humans for the first time, and then it spread primarily through droplets formed by infected people when they sneeze, cough, or speak [1–2]. Since the droplets are too heavy to travel long distances unless they come into close contact, they can only spread from person to person [3]. COVID-19 infects the lungs and causes tissue damage in those who become infected. Some people may not experience any symptoms in the early stages, whereas the majority of people who experienced the primary symptoms are fever and cough. Muscle aches, a sore throat, and a headache are all possible secondary symptoms. COVID-19 is caused by a virus that affects people of all ages. However, evidence to date indicates that two groups of people are more likely to develop severe COVID-19: the elderly (people over the age of 70) and those with severe chronic diseases, including cardiovascular disease, cancer, diabetes, high blood pressure, respiratory disease, and chronic liver disease [4]. The COVID-19 disease is currently spreading due to a

lack of rapid detection methods. In 2020, this disease claimed the lives of a significant number of people all over the world. The respiratory tract and lungs are particularly vulnerable to virus transmission, resulting in inflammation. The process is responsible for obstructing oxygen intake. In order to reduce the number of deaths caused by this virus, health professionals and doctors around the world must be able to detect it quickly and accurately.

Medical imaging, blood tests, and viral tests are the most used diagnostic medical procedures [5].

First, blood tests are performed to detect the presence of coronavirus (COVID-19) antibodies. This test's sensitivity for detecting COVID-19 in the blood is low, ranging from 20% to 30% [6]. A type of antibody detection test is the rapid diagnostic test (RDT), which takes 30 minutes or less. However, RDT test kits are in short supply, and their efficacy is dependent on sample quality and disease onset time. Furthermore, because the test does not differentiate COVID-19 from other viral infections, it can result in false positives; thus, it is not recommended for COVID-19 diagnosis [7]. The second method uses viral tests to detect COVID-19

*Corresponding author e-mail: elbarogy2000@yahoo.com

antigens in respiratory tract samples. The Reverse Transcription Polymerase Chain Reaction is one of the most important and widely used tests (RT-PCR). Large-scale studies show that disease recognition rates range between 60 and 70 percent [8], implying that a negative RT-PCR result is possible initially. Several RT-PCR tests are thus performed over a 14-day observation period to ensure that the test result is accurate for the diagnosis. In other words, a negative RT-PCR result of a suspected COVID-19 case is only considered true negative if no positive RT-PCR results are obtained after 14 days of testing [9]. A shortage of RT-PCR test kits in several countries [10] can be frustrating for patients and costly for healthcare authorities.

Radiological imaging of the chest, such as x-rays and computed tomography (CT), is a third important method for diagnosing covid-19 [11]. Since COVID-19 is primarily a respiratory disease, chest radiology scans are an important diagnostic and early management tool. CT scanning is a reliable method of detecting COVID-19 pneumonia [12]. These advantages of CT have proven it more accurate and faster than RT-PCR for detection; consequently, clinicians are being encouraged to make diagnoses on basis of clinical as well as chest (CT) findings [13,14]. In countries such as Turkey, china and japan. CT is used for COVID-19 identification, and the findings are evaluated over time. Within the first 0–2 days, the majority of patients have a normal CT scan [15]. The most significant lung damage is detected ten days after the onset of symptoms in research on lung CT of patients who survived COVID-19 pneumonia [16]. Therefore, CT is examined over a lengthy period of time. CT have some limitations, such as imaging taking longer, and in many countries, CT scanners are not widely available. Furthermore, the cost of CT imaging is prohibitively high, and children and pregnant women may face health risks as a result of the high radiation levels [17]. X-ray imaging, on the contrary, because of its increased availability, has played a significant role in many medical and epidemiological cases. [18,19]. Chest X-rays are promising for emergency cases and treatment due to their operational speed, low cost, and ease of use for radiologists, and the patient is exposed to low doses of ionizing radiation, which is available in many countries. Chest X-ray has been used as a first-line diagnostic tool [20,21]. According to radiologists, a variety of abnormalities were discovered in COVID-19 radiology scans. These abnormalities are as follows: (1) right inifrahilar airspace opacities in the patient [22]. (2) unidentified nodular lesion with inverted halo sign and thickened edge in the left lower lobe, adjacent to the adjacent pleura at 2.3 cm resolution [23]. (3) coherence and widening of the blood vessels in the lesion [24]. (4) There is an air bronchogram sign and interlobular septal thickening with or without vascular expansion. [20]. (5) As shown in Fig. 1, there is a single nodular opacity in the lower left lung.

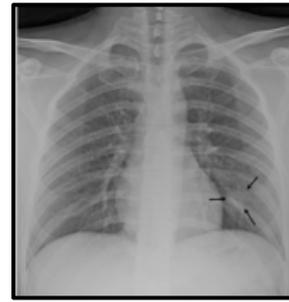


Fig. 1: Chest x-ray images patient [23].

X-ray, despite its significant advantages, but it needs a radiologist. Radiologists play a vital role due to their extensive experience in this field. However, due to the limited number of radiologists, it is challenging to provide expert doctors for each hospital. Therefore, accurate and rapid Artificial Intelligence (AI) models may help overcome this issue and provide patients with timely assistance. Techniques of artificial intelligence in radiology can help obtain an accurate diagnosis [25]. Furthermore, AI methods can help remove drawbacks such as test costs, a lack of RT-PCR test kits, and test results waiting time.

Machine Learning (ML) is a data-driven modeling technique. This machine learning technique derives the "model" from data, where data refers to documents, audio, images, and other data types. As shown in Fig. 2, the "model" is the result of machine learning[26].

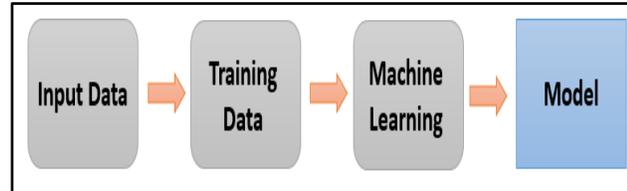


Fig. 2: Machine learning process.

The steps of ML are as follows: (1) gathering data and preparing that data, (2) choosing a model, (3) training, (4) evaluation (5) hyperparameter tuning (6) prediction. In order to train in machine learning, we have to extract features manually, as shown in the following figure.

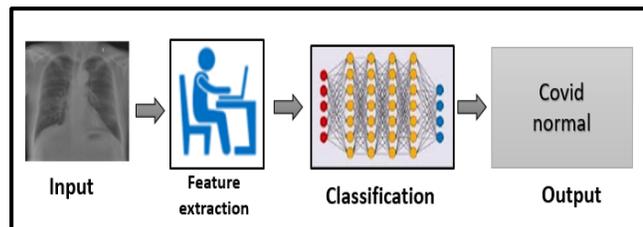


Fig. 3: Feature extraction by machine learning.

Muhammad Imad et al. extracted the discriminant characteristics using the Histogram of Oriented Gradients (HOG). Five different machine learning methods are utilized

to effectively differentiate between COVID-19 and normal chest x-ray images in the classification stage, including K-Nearest Neighbors, naive bayes algorithm, random forest, decision tree, and support vector machine (SVM). The results reveal that among the four classifiers, SVM has the greatest accuracy of 96 % K-Nearest Neighbors and random forest had 92 % accuracy, naive bayes had 90 % accuracy, and decision tree had 82 % accuracy) [27].

Deep learning, which is also a branch of AI, enables the development of end-to-end models that can achieve promised results using input data without the need for manual feature extraction [28,29], as demonstrated in Fig. 4. Deep learning techniques have been successfully used to detect arrhythmias [30,31], pneumonia detection from chest X-ray images [32], classification of skin cancer [33,34], breast cancer detection [35,36], brain disease classification [37] and lung segmentation [38,39].

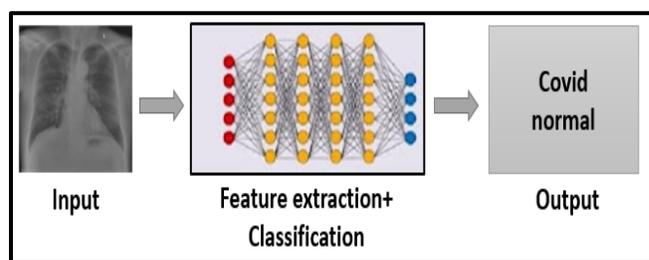


Fig. 4: Feature extraction by deep learning.

Many x-ray images have been widely used to detect COVID-19 using deep learning. Kumar and Kumari obtained an accuracy of 93.0 percent using AlexNet as a feature extractor to feed Support Vector Machine (SVM) classifier to x-ray images. They also used VGG16 and VGG19 to extract COVID-19 features and feed them to the SVM, yielding final accuracies of 92.7 and 92.9 percent, respectively [40].

Al-Qudah et al. ShuffleNet model was used to extract features automatically of x-ray images, which were then fed into four distinct classifiers: Softmax, SVM, Random Forest, and KNN, [41]. Sethy and Behera used an x-ray image to classify features obtained from various convolutional neural network (CNN) models with a support vector machine (SVM) classifier, achieving a final accuracy of 95.33 [42].

In order to improve the detection of COVID-19 from chest x-ray images, we propose combining the manual features extracted and deep learning extracted for a chest x-ray. This paper proposed a feature vector fusion obtained by combining HOG and CNN techniques. As the last input for the training and test datasets. Individual feature extraction techniques produced less satisfactory results than the fusion approach, demonstrating that the proposed method was effective and more accurate than single feature extraction approaches in classifying COVID-19 cases. Other machine-learning methods were also utilized in the final classification, such as Softmax and SVM. The fused feature vectors were fed to find the best classification of multiple

classes during the classification. Machine learning and deep learning techniques combined with radiography can help detect this disease accurately. The proposed method can help address the issue due to a scarcity of specialized physicians in outlying villages and increase the speed and accuracy of diagnosis.

2 Materials and Methods

The proposed system used X-ray grayscale images as input to identify COVID-19. The system considered two feature extractors: Histogram oriented gradient (HOG) and convolutional neural network (HOG). A feature vector was extracted from the x-ray COVID-19 dataset using the CNN technique. The HOG method was then applied to the same images to extract another feature vector. These two features were combined and fed into the classification model as input. Figure 5 depicts the fundamental steps of the proposed system architecture.

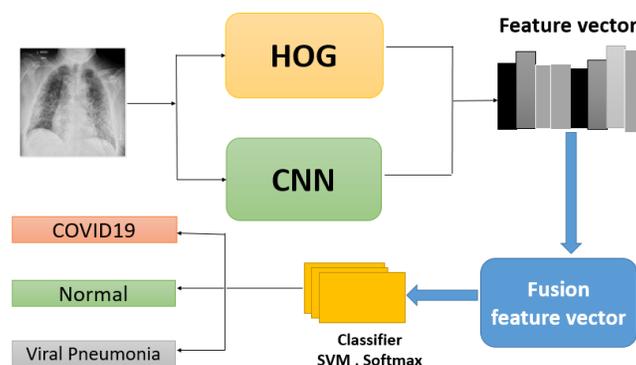


Fig. 5: The proposed fusion method for covid-19 detection based on fusion of CNN and HOG features.

A single technique's extraction of features was insufficient to identify COVID-19 correctly. In contrast, the fusion approach, which extracts features using two different techniques, could provide a large number of features for accurate identification. In this context, fusion was viewed as a concatenation of the two individual vectors. Finally, the SVM and Softmax classifiers determined whether x-ray images were COVID 19 or not.

The dataset and suggested feature extraction and classification approach are explained in further depth in the next part.

2.1 Description of the Dataset

X-rays of the chest The patients' data set was obtained and saved in a familiar location. As a reference for evaluating the smart system's performance, images were classified as positive, negative, or viral pneumonia. The data set was used in this study to evaluate the system's performance. As shown in Table 1, our experimental evaluation used a data set of 3,624 images. Then, they were divided into 970 positive images, 1,311 normal images, and 1,345 images with viral pneumonia.

Table 1: The class of the images that were utilized in the dataset that was chosen.

Class	COVID	Normal	Viral pneumonia	Total images
Number of images	970	1311	1345	3626

The database contains the size of all images 227 x 227 pixels, and the images obtained were in grayscale format. Figure 6 shows a sample of positive, negative, and viral pneumonia images.

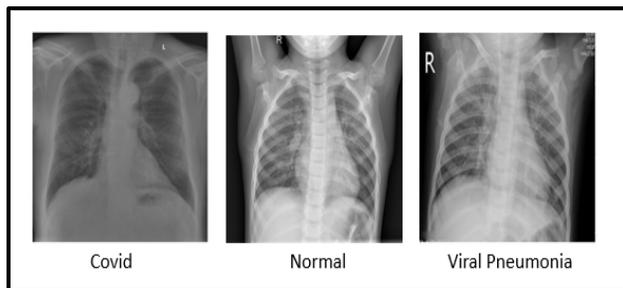


Fig. 6: Images of chest x-rays from COVID-19 infected patients, healthy patients, and viral pneumonia patients.

2.2 Feature Extractor using (CNN).

The extraction of features using CNN is a significant area of study in computer science [43]. CNN offers better feature extraction when the neural network for feature extraction is deeper (with more layers); therefore, we can obtain a more accurate classification process. CNN comprises two neural networks: one that extracts the input image's features and another classifies the feature image. A typical CNN feature extraction architecture is depicted in Fig.7.

The feature extraction neural network architecture comprises an input layer, an output layer, and a set of hidden layers. Hidden layers consist of a convolutional layer(s), then a pooling layer (s). The convolution layer defines the most crucial kernel relevant to the application.

The pooling layer reduces dimensionality by obtaining average or maximum image corrections. The output of the

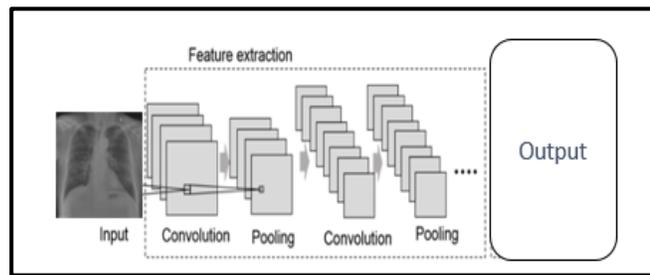


Fig.7: Typical architecture of CNN.

CNN of feature extraction is a tensor converted into a vector. This vector is a deep learning feature and is considered input for the classifier [44]. Before classifying, CNN was trained using extracted features. CNN uses a variety of pre-trained models, including Google Net, XCEPTION, VGGNet, ResNet, AlexNet, SENet, DenseNet, MobileNet, ShuffleNet, and DarkNet-53[45]. To extract features from the training and test data sets. All of these models produce comparable results. AlexNet, which was proposed in this work, clearly outperformed the other CNN models in terms of accuracy and specificity.

AlexNet network that has been pre-trained was fine-tuned in this research to serve as a feature extractor for this study's experimental dataset. AlexNet has been trained on over a million images and can categorize them into 1000 different object categories, such as keyboards and pencils. As a result, the model was able to learn detailed feature representations for a wide range of images. AlexNet finished first in the ILSVRC Challenge 2012, with an error rate of 17% in the top five. The authors used a dropout regularization technique and data augmentation in AlexNet to address the overfitting problem.

AlexNet was the first convolutional neural network to use graphics processing units (GPUs) to improve performance. As shown in Fig.8, AlexNet comprises five convolutional layers and three max-pooling layers. Convolutional filters are used in conjunction with rectified linear units (ReLU). Pooling layers are used to maximize pooling. The first layer's input size in AlexNet is 227*227. The size of the input image must be resized to 227*227.



Fig. 8: Image feature extraction using an AlexNet pre-trained (CNN) model.

Next, we will look at how the image has been processed as it passes through the convolution and pooling layers. In our dataset, the x-ray image's original size is 227x227 pixels. This section describes how the convolution layer creates new images known as feature maps. The feature map emphasizes the original image's distinguishing features. Compared to other neural network layers, the convolution layer operates in a very different manner.

It should be noted that the actual ConvNet's filters are determined through the training process rather than by manual decision. Begin by applying the first filter. The convolution operation begins in the submatrix's upper-left corner, which is the same size as the feature map used in equation (1). The images in the following section will elaborate on this on a 2x2 filter and a 4x4 image, as well as how the filter moves on the image.

$$c_{ij} = a_{i_1}b_{1j} + a_{i_2}b_{2j} + \dots + a_{i_n} + b_{nj} = \sum_{k=1}^n a_{i_k}b_{kj}, \quad i = 1, \dots, m \text{ and } j = 1, \dots, p. \quad (1)$$

The convolution operation is the sum of the products of the elements in the two matrices in the same positions. The result of 2 is calculated as follows in Fig. 9:

$$(1*1) + (3*0) + (4*0) + (1*1) = 2$$

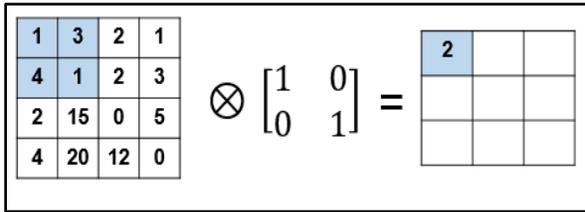


Fig. 9: The convolution process begins in the upper-left corner.

Similarly, another convolution operation is performed for the second submatrix, as demonstrated in Fig 10.

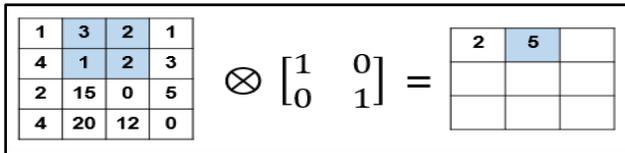


Fig. 10: The second convolution operation.

The next row begins on the left when the top row has been completed, as shown in Fig 11.

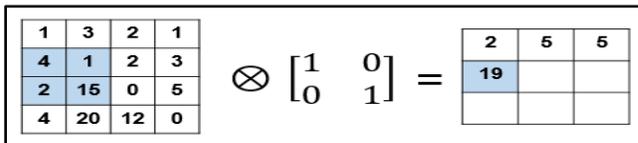


Fig. 11: The convolution operation is restarted from the left.

As demonstrated in Figure 12, we repeat the process until we get the feature map for the given filter.

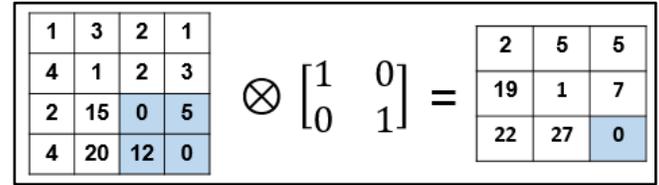


Fig. 12: The given filter's feature map has been completed.

The number of pixels in which the filter matrix slides over the input matrix is referred to as the stride. If the stride is 1 at a time, the filter is pushed one pixel, as shown in Fig. 13 (a). When the stride is set to 2, the filter hops two pixels at a time as it moves around them, as shown in Fig. 13. (b). Making a larger move stride will result in a smaller feature map.

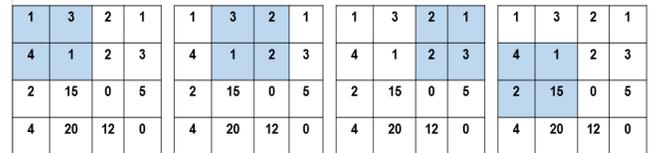


Fig. (a): Stride=1



Fig. (b): Stride=2

Fig. 13: (a,b) How to slide the filter matrix over the input matrix with selecting strides [46].

There is another operation in the convolution layer it is called padding. The input matrix padding with zeros around the boundary, we use this feature to apply the filter to all elements of the entered matrix. It also has the advantage of controlling the size of the feature map, as shown in Fig. 14.

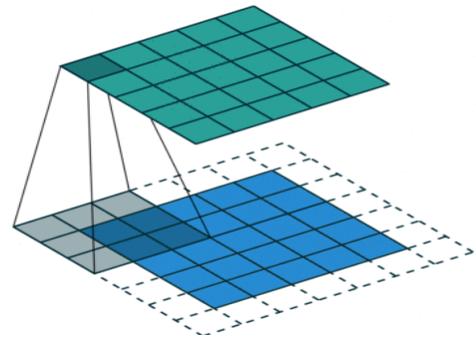


Fig. 14: The matrix padding [45].

We apply these steps to the proposed study in the same way. This study contains filters 96 filters, with a size of 11x11 per

one filter in the first layer, and the size and number of filters change from one layer into another. When the image is processed with the 11x11 convolution filter on a 227x227 image, it becomes a 55x55 feature map, as shown in equation 2. Because we have 96 convolution filters, the layer produces 96 feature maps, each of which is 55x55 in size. Figure 15 depicts the convolution layer's image processing results.

$$\text{Size of a feature map} = \text{input size} - (\text{Filter size} + (2 * \text{Padding}) / \text{Stride} + 1) \tag{2}$$

$$(227 - 11 + (2 * 0) / 4) + 1 * (227 - 11 + (2 * 0) / 4) + 1 = 55 * 55, \text{ 96}$$

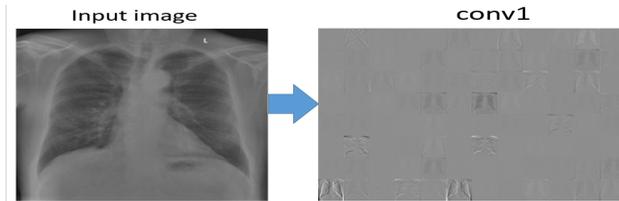


Fig. 15: The first convolution layer of the image's output.

ReLU is the second function, which performs a threshold operation on each element, setting any input value less than zero to zero as shown in equation 3. The ReLU function does not affect the size of its input.

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3}$$

The image in Figure 16 is the result of the ReLU function processing the feature map produced by the convolution layer. This value must be an integer to ensure that the entire image is fully covered. If these options do not completely cover the image, the software ignores the remaining portion of the image along the right and bottom edges in the convolution by default. The outputs of these neurons are subjected to some form of nonlinearity, such as ReLU specified by a ReLU layer.



Fig. 16: Image depicting what the ReLU function did with the convolution layer's feature map.

Pooling is a process that reduces the size of the rectified feature maps produced by ReLU. Two functions used in the aggregation process are average pooling; the maximum is the most used type. The max-pooling function uses the proposed AlexNet model where we combine the input image pixels, pooling using a 3x3 window. So that it is passed on the rectified feature map by amount two cells (stride=2) and the maximum value in each region is chosen as shown in following Fig.

18.21	15.52	17.67	18.07	13.01	14.45	15.98	18.21	15.52	17.67	18.07	13.01	14.45	15.98	18.21	15.52	17.67	18.07	13.01	14.45	15.98
16.67	12.52	13.71	9.35	13.15	19.83	11.28	16.67	12.52	13.71	9.35	13.15	19.83	11.28	16.67	12.52	13.71	9.35	13.15	19.83	11.28
13.88	11.56	10.91	8.85	2.04	13.81	14.7	13.88	11.56	10.91	8.85	2.04	13.81	14.7	13.88	11.56	10.91	8.85	2.04	13.81	14.7
10.45	13.36	9.64	23.01	0.98	5.59	6.65	10.45	13.36	9.64	23.01	0.98	5.59	6.65	10.45	13.36	9.64	23.01	0.98	5.59	6.65
9.55	15.68	5.22	9.21	11.05	11.66	7.05	9.55	15.68	5.22	9.21	11.05	11.66	7.05	9.55	15.68	5.22	9.21	11.05	11.66	7.05
12.23	13.84	5.03	1.11	2.93	8.00	7.14	12.23	13.84	5.03	1.11	2.93	8.00	7.14	12.23	13.84	5.03	1.11	2.93	8.00	7.14
18.21	15.52	17.67	18.07	13.01	14.45	15.98	18.21	15.52	17.67	18.07	13.01	14.45	15.98	18.21	15.52	17.67	18.07	13.01	14.45	15.98
16.67	12.52	13.71	9.35	13.15	19.83	11.28	16.67	12.52	13.71	9.35	13.15	19.83	11.28	16.67	12.52	13.71	9.35	13.15	19.83	11.28
13.88	11.56	10.91	8.85	2.04	13.81	14.7	13.88	11.56	10.91	8.85	2.04	13.81	14.7	13.88	11.56	10.91	8.85	2.04	13.81	14.7
10.45	13.36	9.64	23.01	0.98	5.59	6.65	10.45	13.36	9.64	23.01	0.98	5.59	6.65	10.45	13.36	9.64	23.01	0.98	5.59	6.65
9.55	15.68	5.22	9.21	11.05	11.66	7.05	9.55	15.68	5.22	9.21	11.05	11.66	7.05	9.55	15.68	5.22	9.21	11.05	11.66	7.05
12.23	13.84	5.03	1.11	2.93	8.00	7.14	12.23	13.84	5.03	1.11	2.93	8.00	7.14	12.23	13.84	5.03	1.11	2.93	8.00	7.14

Fig. 17: The maximum pooling procedure.

The pooling layer shrinks each feature map to 27 * 27 pixels after the input image passes through it. Figure 18 depicts the procedure.

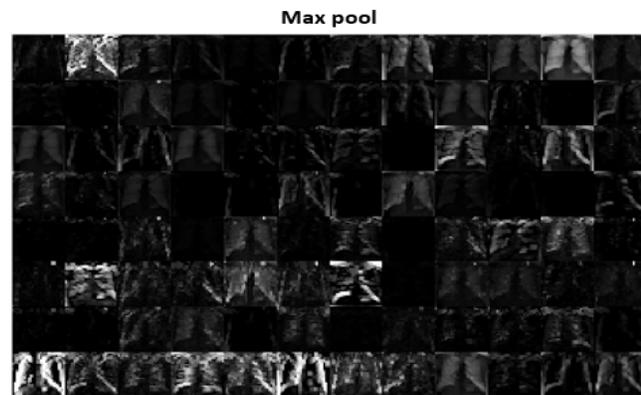
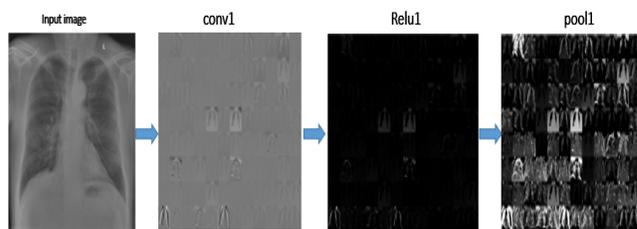


Fig. 18: After the maximum pooling process.

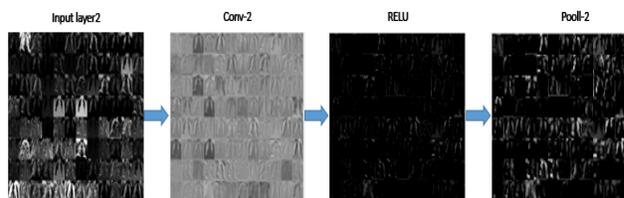
In the second layer of the neural network, the input size for this layer was 27x27. This layer consists of the 256 trained convolution filters of size 5x5. The resulting feature map from the second layer is size 27x27 according to Equation 1, considering the change in stride and padding. This image is completely dark after the ReLU operation. Each feature map is shrunk by the pooling layer of 55x55 to 13 * 13 pixels, as depicted in Fig. 19 (b).

The third layer and fourth of the neural network; these two layers are the same stride and padding and filter size and different in the number of filters where the number of filters

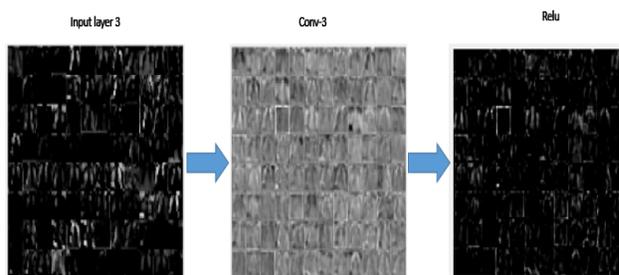
in the third layer was 256 and in the fourth layer 384, as shown Fig. 19 (c), (d).



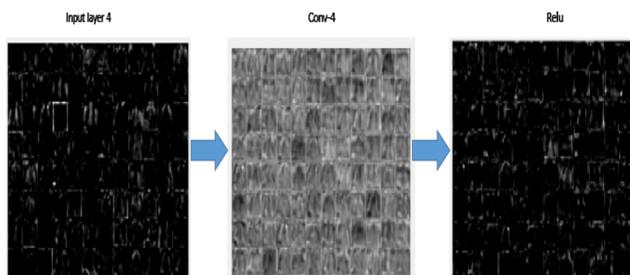
a. First convolution layer.



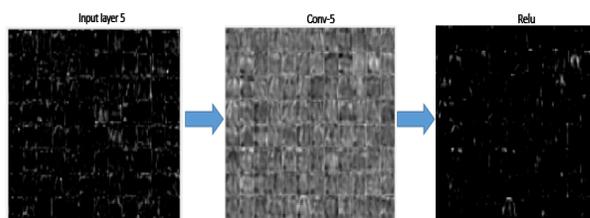
b. second convolution layers.



c. third convolution layers.



d. fourth convolution layers.



e. Five convolution layers

Fig. 19: The five stages of feature extraction for all layers.

The feature extraction neural network's final output. These images are converted into one-dimensional vectors before being saved in the classification neural network. The greater the number of small images containing key network features, the better the recognition performance. The number of features in the last layer was 9216, and we note a significant decrease in features after the original image was size 51529, as shown in Fig. 20.

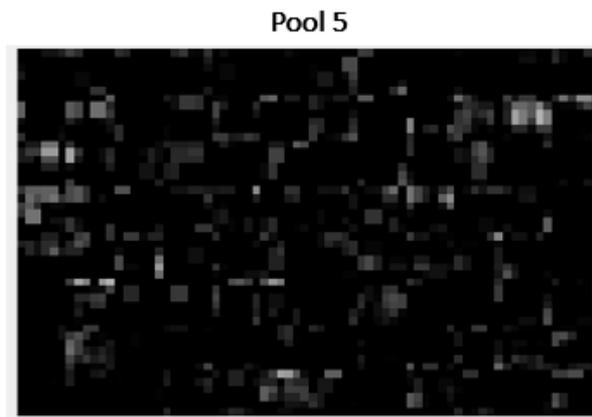


Fig. 20: The images after the max pooling5.

2.3 Feature Extractor using (HOG).

In this study, we will use the same input image used in CNN Input image, selected grayscale. The input image must be a real value that is not scattered. If the images are cropped too tightly, they may lose the shape information that the HOG function can encode. The main idea behind HOG is that shape and local appearance within an image can be described by the intensity gradients distribution or edge display HOG method from the image. Figure21 describes the HOG method for extracting features from the image [47].

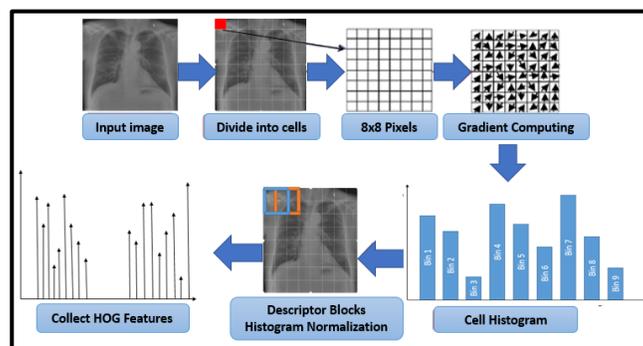


Fig. 21: Steps of HOG feature extraction process.

HOG approach is used to characterize the gradient orientation in small areas of an image, and the data from all regions are then combined into a single vector. The HOG feature extraction technique is split down into many steps.

Step1: The input image is divided into smaller related sub-images known as cells. Each cell has a size of 8×8 pixels.

Step 2: This is used to calculate the gradient's magnitude and orientation for every single pixel in the cell. In both the horizontal and vertical directions, the most efficient method is to employ a 1-D centered point discrete derivative D_x and D_y . Using convolution operation, the horizontal (I_x) and vertical (I_y) derivatives are obtained, see Eqs. (4 and 5).

$$I_x = I * D_x \text{ where } D_x = [-1,0,1] \tag{4}$$

$$I_y = I * D_y \text{ where } D_y = [-1,0,1] \tag{5}$$

Eq. 4. means that $I_x = I(x + 1, y) - I(x - 1, y)$, while Eq. 5 a. means that $I_y = I(x, y + 1) - I(x, y - 1)$. For each pixel $I(x, y)$, The gradient's magnitude is expressed by the given Eq. 6.

$$|G(x, y)| = \sqrt{I_x^2 + I_y^2} \tag{6}$$

The gradient's direction and orientation are given by Eq.7.

$$\theta(x, y) = \tan^{-1} \frac{I_y}{I_x} \tag{7}$$

Step 3: the 3rd step is to compute each cell's histogram. The region of interest (ROI) is divided into small areas known as cells. The cell has a resolution of 8×8 pixels. Each pixel in the cell contributes a weighted value to a histogram channel based on orientation. The gradient magnitude can be used to represent the pixel contribution. In [3] the gradient magnitude value is used as a vote weight. When the gradient is unsigned, the channels in the histogram are evenly spaced from 0° to 180° ; when the gradient is signed, the channels are evenly spaced from 0° to 360° . As shown in Table 2, the orientation bins in this study are evenly spaced from 0° to 180° and are provided by nine of 20° . One cell has a 9-dimensional feature vector. Each bin contributes by adding the magnitude of the gradient in the bin corresponding to the gradient direction.

Table 2: The orientation bins.

Bin number	Direction
1	$0^\circ-20^\circ$
2	$20^\circ-40^\circ$
3	$40^\circ-60^\circ$
4	$60^\circ-80^\circ$
5	$80^\circ-100^\circ$
6	$100^\circ-120^\circ$
7	$120^\circ-140^\circ$
8	$140^\circ-160^\circ$
9	$160^\circ-180^\circ$

Finally, each cell C_i has 9 features as follows: $C_i = [h_1^i, h_2^i, \dots, h_9^i]$ where i is the cell number.

Step 4: This step involves calculating descriptors for each block. Because of local differences in illumination and contrast, gradient strengths vary greatly. By combining the cells into larger, spatially related blocks, it is possible to avoid changes in light and contrast. The block has a size of

2×2 cells. These blocks frequently overlap, implying that each cell contributes significantly to the final feature. According to several studies, every four cells form one block. The descriptors for each block include 4×9 features, i.e., the 36-component vector given by eq. 8.

$$V = [a_1, a_2, \dots, a_{36}] \tag{8}$$

The gradient local normalization stage comes next. After calculating the HOG features for each cell, the arrays were normalized and combined into a single 36-component array. The vector V is normalized by dividing its values by the constant k given by eq. 9.

$$k = \sqrt{(a_1)^2 + (a_2)^2 + \dots + (a_{36})^2} \tag{9}$$

The is a normalized vector is obtained as in Eq. 10.

$$|V| = \left(\frac{a_1}{k}, \frac{a_2}{k}, \dots, \frac{a_{36}}{k} \right) \tag{10}$$

Step 5: This is the final step in the image's HOG feature generation process. For 16×16 of the image's blocks, we created features. Subsequently, all of this is put together to get the features for the final image, and a 227×227 pixel image contains 27×27 overlapping blocks. We combine the normalized vectors from all blocks into a single 1-D vector with 26244 features representing the final HOG feature.

We apply these steps to the proposed dataset in the same way. In step 1, the input image is divided into small connected sub-images called cells, each cell for size 8×8 , as demonstrated in Fig. 22.

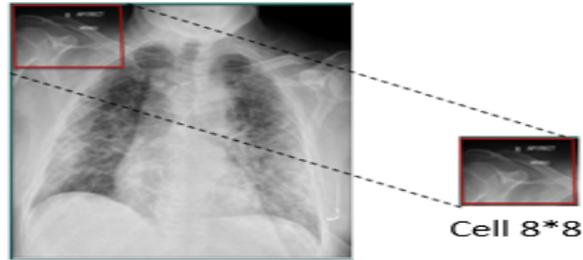


Fig. 22: image segmentation into cells of size 8×8 .

In step 2, magnitude and orientation computation are the second step in the feature extraction process in the image, as depicted in Table 3 and Table 4.

Table 3: Calculate the magnitude for the selected cell.

8	11	10	10	17	15	16	30
8	12	15	13	13	26	19	12
10	15	17	20	13	23	20	22
12	16	16	14	40	22	11	35
12	30	23	37	45	10	12	42

26	37	35	31	12	15	15	26
25	11	21	18	26	21	16	14
13	10	15	14	14	13	19	17

Table 4: Calculate the orientation for the selected cell.

36	80.3	85.23	61.6	68.9	87.7	51.7	58.3
9.46	29.7	75.9	63.4	170.1	53.1	15.9	22.8
148.9	29.7	11.3	144.0	83.6	29.7	82.8	41.9
7.12	14.9	171.5	35.3	75.9	24.1	31.6	61.1
25	62.3	169.7	37.6	-46.04	110.9	7.12	36.8
19.3	64.6	154.4	39.9	149.8	174.9	19.9	61.8
49.7	60.2	71.5	39.5	29.3	74.7	29	60.6
68.19	179.6	79.2	86.8	176.7	76.6	76	36.3

In step 3, the range of orientation is from 0° to 180°. This range is divided into 9 Bin of 20 degrees, and the Bin number for each orientation is determined gBin by equation (11).

$$gBin(gDir(x,y)) = \begin{cases} k; & \text{if } (k - 1) \times 20 \leq gDir(x,y) < k \times 20 \\ 9; & \text{if } gDir(x,y) = 180 \end{cases}; \quad (11)$$

where $k = 1,2,3 \dots 9$

For example, using equation (11), the angles between 0-20 take the number 1, and from 20-40 take the number 2, as displayed in Table 5.

Table 5: The orientation bin for the selected cell.

2	5	5	4	4	5	3	3
1	2	4	4	9	3	1	2
7	2	1	7	5	2	5	3
1	1	9	2	4	2	2	4
2	4	9	2	3	6	1	2
1	4	8	2	8	9	1	4
3	4	4	2	2	4	2	4

4	9	4	5	9	4	4	2
---	---	---	---	---	---	---	---

Based on Table 4, the orientation is of pixel A (1,1) is 36. According to equation (11), the bin number is 2 because $20 \leq 36 < 40$. Then, $k=2$, as depicted in Table 5.

The same procedure is followed for each pixel in the image. We will check the orientation of each pixel and store the frequency of the orientation values in the form of a 9×1 matrix, as shown in Table 6.

Table 6: Frequency of the orientation values.

BIN	gDir
BIN1	8
BIN2	16
BIN3	6
BIN4	17
BIN5	6
BIN6	1
BIN7	2
BIN8	2
BIN9	6

Finally, the histogram for the given cell is as demonstrated in Fig. 23.

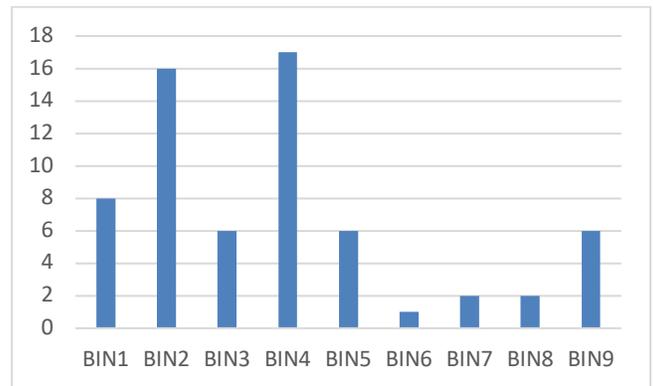


Fig. 23. Histogram for the selected cell.

In step 4, by taking 2×2 cells and normalizing the gradients, we can reduce the lighting variation, i.e., each block contains four cells, as demonstrated in Fig 24. These blocks typically overlap, meaning that each cell contributes more than once to the final feature.

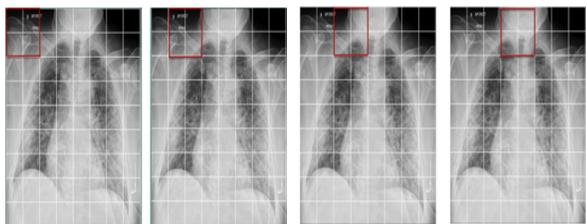


Fig. 24: image segmentation into a block of size 2×2.

Four cells are combined to make a single block. Based on Table 7, four 9×1 matrices or a single 36×1 matrix could be obtained.

Table 7: A single block of four cells.

Cell1	8	16	6	17	6	1	2	2	6
Cell2	12	8	5	3	9	14	1	8	4
Cell3	2	3	9	20	3	14	10	2	1
Cell4	5	4	11	2	7	14	8	9	4

For the final step of calculating HOG features for the image, we created features for one block only of the image. The normalized feature for each block in the segmented image was calculated, and finally, all these features were combined to obtain the features of the full image. An image of 227×227 pixels contains 27×27 blocks, i.e., the total number of blocks is 729 with various overlays. Finally, we construct the normalized vectors acquired from all blocks in a single vector of size 729×36 features representing the final HOG features for one image. Therefore, we obtain 26,244 features for each image. A graphical representation for an example of the obtained features of the full image is demonstrated in Fig. 25.

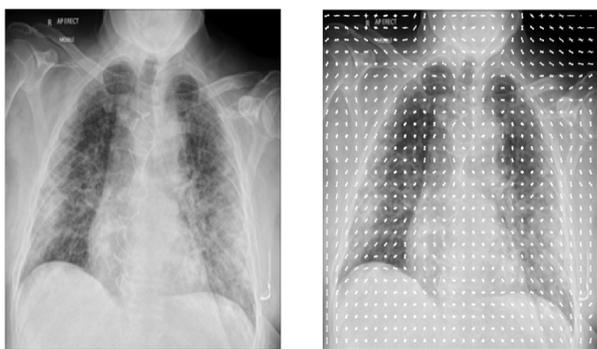


Fig. 25:(Left) An input image. (Right) HOG features of the input image.

2.4 Multi-feature Fusion

In this section, we explain how to apply feature fusion. We describe how to apply feature fusion to combine the CNN and HOG features. Feature fusion has been used in a variety

of applications of machine learning and computer vision [48]. Feature fusion, in particular, has the ability to combine multiple feature vectors. In this study, two feature extractors produce one feature vector each of 4096 and 26244. The final fusion features were fed to the classifiers in order to identify COVID-19 images, and the feature vectors obtained by HOG and CNN were fused to validate this system, as shown in Fig. 26.

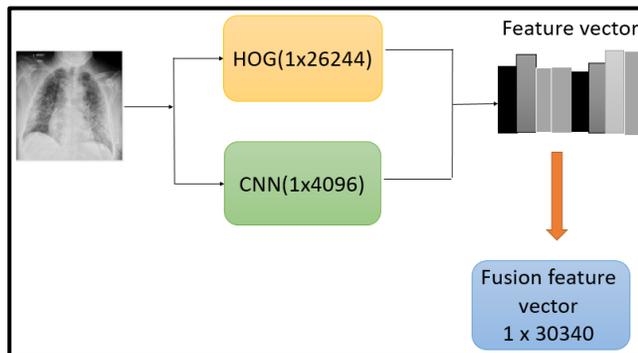


Fig. 26: The proposed fusion steps for HOG and CNN extracted features.

In this study, SVM and Softmax were used as a classifier. In the next section, we will explain this classifier in detail.

3 Classifications

Classification is the final stage of the disease recognition process in which the disease is recognized based on the feature set extracted from the fusion. The classifier's performance depends on the quality of the extractor feature being tuned. The classification process converts the quantitative input data into qualitative information (output categories). There are two stages to the classification process training and testing. The proposed training stage is to constrict the training model based on the extracted feature, and the testing stage is carried out to get the classification accuracy. [2,8,88]. SVM and Softmax classifier is used in this study.

3.1 Support Vector Machine(SVM)

This classifier uses the SVM algorithm. SVM is one of the supervised learning algorithms used in classification and regression problems. The SVM can be effective and achieve excellent accuracy in most user data. There are two types SVM: Binary Classification, where it classifies only two groups, and Multi-class Classification, where it classifies more than one group. The main idea of the SVM algorithm is to find the best separating hyperplane to separate classes. In this study, SVM classifier was used for classification. This classifier is based on statistical learning theory. Given training data:

$$\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} \in R^n \tag{12}$$

It means that a pair of data represent our data (x_n, y_n) , where x_n represents the feature vector which has n dimensions, and

y_n represents the label of x_n . When $y_i = 1$, then this means that the data belongs to group 1, and if $y_i = -1$, then it belongs to group -1.

$$\begin{cases} k \cdot a_i + b \geq +1 - \xi_i \text{ if } c_i = +1 \\ k \cdot a_i + b \leq -1 + \xi_i \text{ if } c_i = -1 \end{cases}$$

Where k is the weight vector, b is the value of the tendency and ξ_i is the positive artificial variable. The classification of the SVM depends on the artificial variable as follows: if $\xi_i = 0$ then the sample a_i is correctly classified. If the ξ_i is in range; $0 < \xi_i < 1$, then a_i is also correctly classified; however, its position is among extreme planes. And when $\xi_i > 1$, it is wrongly classified. If the two-class problem cannot be separated linearly. Then, the kernel function is used for classification in a higher dimension.

3.1 Softmax Deep Neural Network Classifier

ANNs are mathematical algorithms used to recognize patterns and images [49]. The Softmax layer is essential for the neural network to determine the classification process, a significant part in calculating the Loss function. The Softmax function often follows the fully-connected layer, called the normalized exponential. The Softmax layer only has three layers, as shown in Figure.27: input, softmax, and output, so it does not take long to identify the disease.

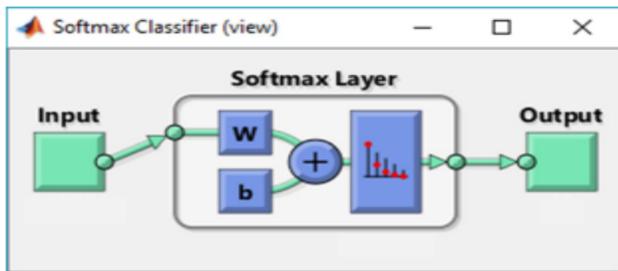


Fig. 27. Softmax classifier

The Softmax layer is an activation function that converts the inputs to a probability distribution over classes, with the output consisting of positive numbers of a total one. These probabilities are used for a classification process. The Softmax function $f(x)$ is defined as follows:

$$\text{Softmax: } \{x \in R^C\} \rightarrow \left\{ p \in R^C \mid p_i > 0, \sum_{i=1}^C p_i = 1 \right\}$$

$$p_j = \frac{e^{x_j}}{\sum_{c=1}^C e^{x_c}} \quad \text{for } j = 1, \dots, C \quad (13)$$

where x_i is an input element in the Softmax layer, and x is the input vector. p_i represents the relative output of x_i , p is the output vector, and C is the total number of classes.

4 The Proposed System's General Framework

1. Read x-ray image dataset $I_1, I_2 \dots I_k$, each image has size (227×227) , where k represents the number of images in the dataset M .
2. Using the HOG technique, a feature vector was extracted from the X-ray M dataset, which was then fed into two distinct classifiers: Softmax and SVM.
3. the CNN method Alex Net model was used to automatically extract features from the same images, which were then fed into two distinct classifiers: Softmax and SVM.
4. These two features were combined to train the classification model (SVM and Softmax).

5 Results and Discussion

This study used a total of 3624 chest X-ray images for training, testing, and validation. In order to test the Intelligent COVID-19 Framework through its paces, as shown in Table 8. This study used 642 normal images, 474 positive images, and 659 viral pneumonia images for training purposes. In the testing stage, 393 normal images, 290 positive images, and the image of 404 viral types of pneumonia were used. These test images did not appear in the training dataset. The validation set includes 203, 276, 282 for covid-19, normal images, and viral pneumonia images. In addition, these validation images were driven from the training dataset.

Table 8: The number of images used in training, validation, and testing in the categories normal, COVID 19, and viral pneumonia.

	Covid	Normal	pneumonia	Total images
Train	477	642	659	1778
Test	290	393	404	1087
Validation	203	276	282	761
Total	970	1311	1345	3626

This paper proposed a feature vector fusion obtained by combining HOG and CNN techniques. The final input for the training and testing stages was this fusion vector.

Figure 28 depicts a comparison of various feature extraction approaches.

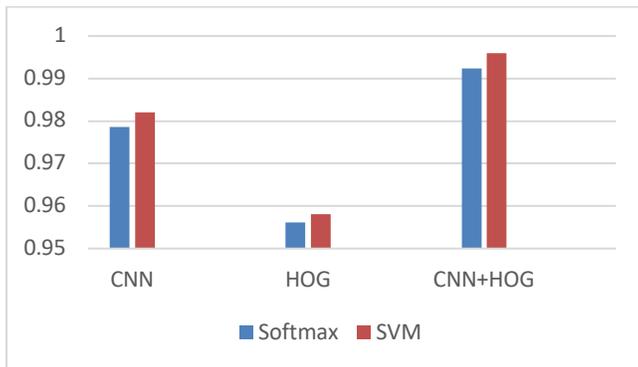


Fig. 28: Comparative results of individual and fusion features.

Table 9: Comparison of CNN, HOG, and proposed model using chest X-ray images

	Softmax	SVM
CNN	0.9786	0.9820
HOG	0.9562	0.9581
CNN+HOG	0.9923	0.9960

COVID-19 detection takes time when using relevant datasets of chest x-ray images. The researchers employed several preprocessing, feature extraction, and classification techniques. At this time, it is difficult to recommend a promising technique or combination of techniques for diagnosing COVID-19 from a chest x-ray image. The majority of cases reported more than 90 percent accuracy, which is statistically very high. The goal, however, would be to increase accuracy as close to 100 percent as possible because misdiagnosis is unacceptable even in a small number of cases. Compared to other techniques proposed in the literature, the proposed technique significantly improved classification accuracy in detecting COVID-19. However, Loey, et al. [50] reported a higher accuracy (100%) than this study, which could be attributed to the smaller number of images (69 COVID-19 and 79 normal images) in their dataset to test the system's performance.

The proposed method, which employs feature fusion extracted by HOG and CNN(AlexNet), SVM, and softmax classifiers, performed better in terms of accuracy. For the chest X-ray dataset, CNN indicates promising results. However, when the classification was done with features that were extracted individually. The accuracies achieved by HOG or CNN were significantly lower than the reported values. The worst of the three was HOG, with an accuracy of 95.62% in the Softmax classifier and 95.81% in the SVM classifier. Similarly, CNN showed higher accuracy of HOG 97.86% in Softmax and 98.20% in SVM.

Despite this, the system demonstrated its robustness in this study by correctly diagnosing COVID-19 cases, which could be attributed to the higher number of distinct features obtained from the fusions of HOG and CNN extracted features.

6 Conclusions

Due to the large number of deaths caused by the coronavirus pandemic, healthcare systems in every country worldwide were stretched to their breaking point. COVID-19 detection at an earlier stage in a more rapid, simple, and cost-effective manner has the potential to save lives and lessen the strain on healthcare professionals by applying image processing techniques to x-ray images. Artificial intelligence can aid in the identification of COVID-19. This research aimed to design and develop an intelligent system for COVID-19 identification with high accuracy and low complexity by combining features extracted from histogram-oriented gradient (HOG) features and convolutional neural networks (CNN). Appropriate feature fusion and classification are critical in detecting COVID-19 using chest X-ray images. The system was fed chest X-ray images to generate the output of the highlighted lung significant region, which was used to identify COVID-19. The proposed feature fusion system (99.60 percent) outperformed individual feature extraction techniques such as HOG and CNN in terms of classification accuracy. In three cases, SVM outperformed other classification techniques such as Softmax in terms of classification accuracy.

Reference

- [1] F. Wu, S. Zhao, B. Yu, Y. M. Chen, W. Wang, Z.G. Song, Y. Hu, Z.W. Tao, J.H. Tian and Y.Y. Pei, A new coronavirus associated with human respiratory disease in China. *Nature*, **579**, 265–269(2020).
- [2] Q. Li, X. Guan, P. Wu, X. Wang, L. Zhou, Y. Tong, R. Ren, K.S. Leung, E.H. Lau and J.Y. Wong, Early transmission dynamics in Wuhan, China, of novel coronavirus-infected pneumonia. *N. Engl. J. Med.*, **382**, 1199–1207(2020).
- [3] M.L. Holshue, C. DeBolt, S. Lindquist, K.H. Lofy, J. Wiesman, H. Bruce, C. Spitters, K. Ericson, S. Wilkerson and A. Tural, First Case of 2019 Novel Coronavirus in the United States, *N Engl J Med.* **382**, 929-936(2020)
- [4] T. Singhal, A review of coronavirus disease-2019 (COVID-19), *Indian J. Pediatr.*, **87**, 281–286(2020).
- [5] N. H. S. England and N. H. S. Improvement, "Guidance and standard operating procedure: COVID-19 virus testing in NHS laboratories." (2020).
- [6] S. Boseley, "WHO warns that few have developed antibodies to Covid-19," *The Guardian*, (2020).
- [7] F. C. Bull et al., World Health Organization 2020 guidelines on physical activity and sedentary behaviour, *Br. J. Sports Med.*, **54**(24), 1451–1462 (2020).
- [8] S. A. Bahammam, M. M. Sharif, A. A. Jammah and A. S. BaHammam, Prevalence of thyroid disease in patients with obstructive sleep apnea, *Respiratory medicine*,

- 105(11),1755-1760(2011).
- [9] X. Jiang, M. Coffee, A. Bari, J. Wang, X. Jiang, J. Huang and Y. Huang, Towards an artificial intelligence framework for data-driven prediction of coronavirus clinical severity. *Computers, Materials Continua*, **63**(1),537-551(2020).
- [10] H. Y. F. Wong, H. Y. S. Lam, A. H. Fong, S. T. Leung, T. W. Y. Chin, C. S. Y. Lo and M. Y. Ng, Frequency and distribution of chest radiographic findings in COVID-19 positive patients. *Radiology*, **296**(2),E72-E78(2020).
- [11] Z. Y. Zu, M. D. Jiang, P. P. Xu, W. Chen, Q. Q. Ni, G. M. Lu and L. J. Zhang, Coronavirus disease 2019 (COVID-19): a perspective from China. *Radiology*, **296**(2),E15-E25(2020).
- [12] C. Zhou, C. Gao, Y. Xie and M. Xu, COVID-19 with spontaneous pneumomediastinum. *The Lancet Infectious Diseases*, **20**(4),510(2020).
- [13] K. Muhammad, S. Khan, J. Del Ser and V. H. C. De Albuquerque, Deep learning for multigrade brain tumor classification in smart healthcare systems: A prospective survey. *IEEE Transactions on Neural Networks and Learning Systems*, **32**(2),507-522(2020).
- [14] J. Liu, Y. Pan, M. Li, Z. Chen, L. Tang, C. Lu and J. Wang, Applications of deep learning to MRI images: a survey, *Big Data Min Anal* **1**(1),1-18(2018).
- [15] F. Altaf, S. M. Islam, N. Akhtar and N. K. Janjua, Going deep in medical image analysis: concepts, methods, challenges, and future directions. *IEEE Access*, **7**,99540-99572(2019).
- [16] F. Xing, Y. Xie, H. Su, F. Liu and L. Yang, Deep learning in microscopy image analysis: A survey. *IEEE transactions on neural networks and learning systems*, **29**(10),4550-4568(2017).
- [17] H. E. Davies, C. G. Wathen, F. V. Gleeson, The risks of radiation exposure related to diagnostic imaging and how to minimise them. *Bmj*, **342**(2011).
- [18] T. Cherian, E.K. Mulholland, J.B. Carlin, H. Ostensen, R. Amin, M.D. Campo, D. Greenberg, R. Lagos, M. Lucero and S.A. Madhi, Standardized interpretation of paediatric chest radiographs for the diagnosis of pneumonia in epidemiological studies, *Bull. World Health Organ*, **83**, 353-359(2005).
- [19] T. Franquet, Imaging of pneumonia: Trends and algorithms. *Eur. Respir. J.*, **18**, 196-208(2001).
- [20] L. N. Mahdy, K. A. Ezzat, H. H. Elmousalami, H. A. Ella and A. E. Hassanien, Automatic x-ray covid-19 lung image classification system based on multi-level thresholding and support vector machine. *MedRxiv*.(2020)
- [21] N. J. Vickers, Animal communication: when i'm calling you, will you answer too?. *Current biology*, **27**(14),R713-R715(2017).
- [22] W. Kong and P. P. Agarwal, Chest imaging appearance of COVID-19 infection, *Radiology: Cardiothoracic Imaging*, **2**(1),e200028(2020).
- [23] S. H. Yoon, K. H. Lee, J. Y. Kim, Y. K. Lee, H. Ko, K. H. Kim and Y. H. Kim, Chest radiographic and CT findings of the 2019 novel coronavirus disease (COVID-19): analysis of nine patients treated in Korea, *Korean journal of radiology*, **21**(4),494-500(2020).
- [24] W. Zhao, Z. Zhong, X. Xie, Q. Yu, and J. Liu, Relation between chest CT findings and clinical conditions of coronavirus disease (COVID-19) pneumonia: a multicenter study. *AJR Am J Roentgenol*, **214**(5),1072-1077(2020).
- [25] F. Caobelli, Artificial intelligence in medical imaging: Game over for radiologists?. *European journal of radiology*, **126**(2020).
- [26] K. Phil, *Matlab deep learning with machine learning, neural networks and artificial intelligence*. Apress, New York,(2017).
- [27] N. Khan, F. Ullah, M. A. Hassan and A. Hussain, COVID-19 classification based on Chest X-Ray images using machine learning techniques, *Journal of Computer Science and Technology Studies*, **2**(2),01-11(2020)..
- [28] Y. LeCun, Y. Bengio and G. Hinton, Deep learning, *nature*, **521**(7553),436-444(2015).
- [29] A. Krizhevsky, I. Sutskever and G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, **25**(2012)..
- [30] O. Yildirim, P. Plawiak, R. S. Tan and U. R. Acharya, Arrhythmia detection using deep convolutional neural network with long duration ECG signals, *Computers in biology and medicine*, **102**, 411-420 (2018).
- [31] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, M. Adam, A. Gertych and R. San Tan, A deep convolutional neural network model to classify heartbeats, *Computers in biology and medicine*, **89**,389-396(2017).
- [32] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan and A. Y. Ng, Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning, *arXiv preprint arXiv:1711.05225* (2017)..
- [33] A. Estava, B. Kuprel, R. Novoa, J. Ko, S. M. Swetter, H. M. Blau and S. Thrun, Dermatologist level classification of skin cancer with deep neural networks J, *Nature*, **542**(7639),115- 118(2017).
- [34] N. C. Codella, Q. B. Nguyen, S. Pankanti, D. A. Gutman, B. Helba, A. C. Halpern and J. R. Smith, Deep learning ensembles for melanoma recognition in dermoscopy images, *IBM Journal of Research and Development*, **61**(4/5),5-1(2017).
- [35] Y. Celik, M. Talo, O. Yildirim, M. Karabatak and U. R. Acharya, Automated invasive ductal carcinoma detection based using deep transfer learning with whole-slide images, *Pattern Recognition Letters*, **133**, 232-239(2020).
- [36] A. Cruz-Roa, A. Basavanahally, F. Gonza'lez, H. Gilmore, M. Feldman, S. Ganesan and A. Madabhushi, . Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks. In Medical Imaging 2014, *Digital Pathology*, **9041**,904103,(2014).
- [37] M. Talo, O. Yildirim, U. B. Baloglu, G. Aydin and U. R. Acharya, Convolutional neural networks for multi-class brain disease detection using MRI images, *Computerized Medical Imaging and Graphics*, **78**,101673(2019).
- [38] G. Gaa'l, B. Maga and A. Luka'cs, Attention u-net based adversarial architectures for chest x-ray lung

- segmentation, *arXiv preprint arXiv*,**2003**,10304(2020).
- [39] J. C. Souza, J. O. B. Diniz, J. L. Ferreira, G. L. F. da Silva, A. C. Silva and A. C. de Paiva, An automatic method for lung segmentation and reconstruction in chest X-ray using deep neural networks, *Computer methods and programs in biomedicine*,**177**,285-296(2019).
- [40] Z. Wu and J. M. McGoogan, Charakterystyka i waz'ne wnioski z epidemii choroby koronawirusowej 2019 (COVID-19) w Chinach: streszczenie raportu 72 314 przypadko'w z Chin'skiego Centrum Kontroli i Zapobiegania Chorobom, *JAMA*,**323**(13),1239-1242(2020).
- [41] H. Alghamdi, G. Amoudi, S. Elhag, K. Saeedi and J. Nasser, Deep learning approaches for detecting COVID-19 from chest X-ray images: A survey. *IEEE Access* (2021).
- [42] P. K. Sethy and S. K. Behera, Detection of coronavirus disease (covid-19) based on deep features (2020).
- [43] A. A. Barbhuiya, R. K. Karsh and R. Jain, CNN based feature extraction and classification for sign language, *Multimedia Tools and Applications*,**80**(2),3051-3069(2021).
- [44] R. Yamashita, M. Nishio, R. K. G. Do and K. Togashi, Convolutional neural networks: an overview and application in radiology, *Insights into imaging*,**9**(4),611-629(2018).
- [45] Z. Zhou, H. C. Kuo, H. Peng and F. Long, DeepNeuron: an open deep learning toolbox for neuron tracing. *Brain informatics*,**5**(2),1-9(2018).
- [46] S. Albawi, T. A. Mohammed, and S. Al-Zawi, *Understanding of a convolutional neural network*, in 2017 International Conference on Engineering and Technology (ICET),Ieee, 1-6 (2017).
- [47] M. N. ElBedwehy, G. M. Behery and R. Elbarougy, Face recognition based on relative gradient magnitude strength, *Arabian Journal for Science and Engineering*,**45**(12),9925- 9937(2020).
- [48] R. Mostafiz, M. M. Rahman and M. S. Uddin, Gastrointestinal polyp classification through empirical mode decomposition and neural features, *SN Applied Sciences*,**2**(6),1-10(2020)..
- [49] D. A. Medler, A brief history of connectionism, *Neural computing surveys*,**1**,18-72 (1998).
- [50] M. Loey, F. Smarandache and N. E. M Khalifa, Within the lack of chest COVID-19 X-ray dataset: a novel detection model based on GAN and deep transfer learning. *Symmetry*,**12**(4),651(2020).