



Research article

Face detection based on K-medoids clustering and associated with convolutional neural networks



Potharla Ramadevi, Raja Das*

Department of Mathematics, School of Advanced Sciences, Vellore Institute of Technology (VIT), Vellore, 632014, Tamil Nadu, India

ARTICLE INFO

Keywords:

K-medoid clustering
 Convolutional neural networks
 K-means clustering
 Fuzzy K-Means clustering
 SVM
 DenseNet201

ABSTRACT

Over the last several years, the COVID-19 epidemic has spread over the globe. People have become used to the novel standard, which involves working from home, chatting online, and keeping oneself clean, to stop the spread of COVID-19. Due to this, many public spaces make an effort to make sure that their visitors wear proper face masks and maintain a safe distance from one another. It is impossible for monitoring workers to ensure that everyone is wearing a face mask; automated solutions are a far better option for face mask identification and monitoring to assist control public conduct and reduce the COVID-19 epidemic. The motivation for developing this technology was the need to identify those individuals who uncover their faces. Most of the previously published research publications focused on various methodologies. This study built new methods namely K-medoids, K-means, and Fuzzy K-Means(FKM) to use image pre-processing to get the better quality of the face and reduce the noise data. In addition, this study investigates various machine learning models Convolutional neural networks (CNN) with pre-trained (DenseNet201, VGG-16, and VGG-19) models, and Support Vector Machine (SVM) for the detection of face masks. The experimental results of the proposed method K-medoids with pre-trained model DenseNet201 achieved the 97.7 % accuracy best results for face mask identification. Our research results indicate that the segmentation of images may improve the identification of accuracy. More importantly, the face mask identification tool is more beneficial when it can identify the face mask in a side-on approach.

Abbreviation

CNN
 DCNN
 SVM
 FKM
 WHO
 AI
 ML
 CCTV
 CBIR
 PAM
 CLARA

Convolutional Neural Networks
 Deep Convolutional Neural Networks
 Support Vector Machine
 Fuzzy K-Mean Clustering
 World Health Organization
 Artificial Intelligence
 Machine Learning
 Closed Circuit Television
 Content-Based Image Retrieval
 Partitioning Around Medoids
 Clustering Large Applications

(continued on next page)

* Corresponding author.

E-mail address: rajadasrkl@gmail.com (R. Das).

(continued)

ROC	Receiver Operating Characteristic
AUC	Area Under the Curve
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
TPR	True Positive Rate
FPR	False Positive Rate

1. Introduction

Face mask detection has several uses in real-world situations, including remote surveillance of individuals and real-time biometrics. Criminals often cover their faces around the mouth. Researchers have already found a solution to the challenge of detecting occluded faces in the form of a person's head and shoulders [1]. The need to constantly verify many people for using face masks makes this work more difficult. Additionally, the new coronavirus illness (COVID-2019) outbreak required the use of face masks and various additional limitations. The World Health Organization (WHO) declared COVID-19 a pandemic and suggested many preventative measures [2], including the use of face masks, because of their deadly effects, quickly spread, and lack of proper medication and medical care [3,4]. In certain nations, wearing a face mask is required for admission into public buildings. However, the manual inspection process is impossible since so many people use government buildings, and public spaces like airports railway stations, and retail centers. The study of face mask's automated identification and detection has recently attracted interest. Researchers have begun developing automated detection systems to assist with monitoring and surveillance applications during COVID-19 [5]. Identification and classification are two tasks involved in the detection of face masks since it is necessary to identify faces to determine whether people are wearing masks. The research group created Many face-detection approaches, making the first task a heavily researched issue in computer vision [6]. The identification of face masks on various datasets has been the subject of a lot of effort in the last year [7].

Before declaring using a mask is the most effective way to stop the coronavirus from spreading, getting the government and other authorities to enforce mask wear in public spaces can be difficult. At least artificial intelligence (AI) applications that use machine learning (ML) or deep learning algorithms (DL) may detect masks in real-time utilizing a built-in surveillance add network (network of CCTV cameras or any other) that may help in forcing the wearing of masks in open spaces. It is a simple method of maintaining social quiet, having people under control locally, and ensuring that everyone is wearing masks [8]. The technique of recognizing pixels that belong to the same region and assigning them a common name is the fundamental aspect of picture segmentation. Therefore, a wide range of industries, including object identification [9], moving object tracking, and medical image processing [10], have widely used image segmentation. For the first time, this paper uses image segmentation for face mask detection to get high accuracy.

The foundation of picture recognition and comprehension is image segmentation [11]. The method of classifying data items into several groups or classes is known as clustering [12]. Image segmentation commonly uses K-means, a traditional technique for grouping analysis difficulties. The original cluster centers have a significant impact on the K-means clustering process. distinct clustering outcomes might arise from distinct starting cluster centers. In the K-means algorithm, you need to specify an initial number of clusters (K) which are randomly chosen at the beginning. That means it is frequently challenging to identify the optimal K value with accuracy.

The K-medoid clustering approach represents the center, or medoid, of a cluster as an actual location within the cluster. The heart of a cluster is home-based to the medoid, or cluster center [13]. The location of the medoid is also the spot where the total distances to all other objects are the lowest. Because of its resilience to noise and outliers, a medoid can serve as an accurate cluster center representation. In this technique, start with K objects (points) and iteratively work our way up to the best cluster objects (points). Next, our work examined every possible combination of points and determined the quality of clustering for each pair of points. The preset work is to replace the existing best object with a new object (point) if the located one with the best-improved distortion function value.

The primary reason for the industrial relevance of K-medoid clustering for face classification is its effective processing of large data sets. When compared to other clustering methods like FKM. K-medoid clustering is more resilient to outliers. This resilience is useful for accurately grouping facial characteristics in face classification, where the existence of outliers (for example, significant lighting conditions and occlusions) is commonplace. K-medoid clustering helps classify faces by giving each cluster a medoid that represents common facial features. It is appropriate for face classification jobs since it is robust and capable of handling huge datasets with ease. K-medoid clustering enables the inclusion of information such as pixel intensities and face landmarks by providing flexibility in the definition of the distance metric. Its characteristics enable it to adjust to various jobs and situations. K-medoid clustering yields clusters that are useful for Content-Based Image Retrieval (CBIR) and image retrieval, which makes it crucial for industrial applications such as picture databases and security systems.

There is limited research and data on face mask detection [14]. Most entries regarding mask usage are concise and simply indicate whether masks are present. In the field of computer vision, researchers have advanced various cutting-edge techniques, especially in deep learning, for object detection, recognition, tracking, and scene understanding. These advanced methods offer effective solutions for face mask detection by identifying visual entities or objects within images that belong to specific categories. Consequently, deep learning-based models for detecting and identifying face masks have become a crucial computer vision task, supporting healthcare systems and the global community [15]. Techniques such as CNNs can be employed to create intelligent surveillance systems that

Table 1

Summary of literature review.

Author	Face dataset	Method	Accuracy
Li, Yang, and Wu 2020 [37]	WIDER FACE Face dataset	Multi-Task Cascaded CNN with Inception-V2	96.35 %
Biju and Edison 2020 [38]	CEW database	Convolutional neural networks	80.9 %
Mellouk and Handouzi 2020 [39]	EMOTIC dataset	Convolutional neural Networks	86.4 %
Saravanan et al., 2022 [40]	Face image dataset	CNN with VGG 16	96.5 %
Agarwal, Itondia, and Mishra 2023 [41]	RMFD dataset	Extreme Learning Machine (ELM)	97 %
Balasubramanian et al., 2023 [42]	Face mask dataset	Depth-wise Separable Convolutional Neural Network (DWSC-NN)	94.16 %
Dash, Sethy, and Behera 2023 [43]	Plant village dataset	CNN with DenseNet201	94 %
Rahman et al., 2023 [44]	FMPD real dataset	CNN with MobileNet	81.95 %

detect mask-wearing individuals and evaluate social distancing in public spaces [16].

The convolutional properties of the DCNN (Deep Convolutional Neural Network) using VGG19 with a spatial attention method. The method is designed for accurately classifying traffic incidents with an average accuracy of 93.72 % [17]. DenseNet201 [18] is an existing iteration based on the dense-network concept, this design uses a unique method that connects each layer feed-forward to every other layer. In addition, the DenseNet201 model integrates a blend of pooling layers and a compact structure. As a consequence, these design decisions help to decrease the number of parameters and the overall complexity of the model, leading to enhanced efficiency.

Face recognition in computer vision is determining a person from their face image. The well-known CNN model VGG16 (Simonyan and Zisserman, 2015) [19] has been trained on various image datasets to improve its feature extraction skills and perform well in facial recognition applications. The face-detection approach is based on CNN pre-trained models and SVM. Due to its clear classification effect on nonlinear data, utilizing SVM [20] as our final classifier to recognize faces after feature extraction is complete. SVM offers several unique advantages regarding high-dimensional, nonlinear, and small-sample pattern recognition. SVM is also applicable to other machine learning issues including the curse of dimensionality and function overfitting. SVM is used in our system to do further feature extraction and ultimate classification using the face characteristics that are taken from CNN. In this method, we may potentially extract a greater number of characteristics than just CNN.

The novelty of this research paper is to create CNN with pre-trained models DenseNet201, VGG-16, and VGG-19 with SVM for classification that can identify who is wearing masks or not wearing masks. Before that, we used image segmentation to remove the noise data from the images. The proposed method is shown in Fig. 3. The model identifies people in the public setting who are not wearing masks by using the DenseNet201 model from the security cameras. Overall workflow as shown in Fig. 2.

In this contribution, datasets have been assessed by data-driven, unsupervised face detection methods from the literature review Table 1. Totally 2000 images were used from the dataset, which is utilized for clustering to lower the data noise and then K-medoids clustering to group the samples. The item that is most central within a group and has the least average distance from other entities is known as the medoid. The procedure identified as K-medoids aims to reduce the total differences between every data point and its nearest medoid. To lessen the K-medoids clustering algorithm's computing load, a hybrid strategy is suggested. The proposed model DenseNet201 can accurately identify face masks. For dependable face mask identification, our unique end-to-end DenseNet201 model automatically extracts the most discriminating characteristics. To assess the effectiveness of the face mask identification technique to create an extensive dataset. To demonstrate the effectiveness of our model, extensive tests were conducted to compare its performance with three pre-trained models using SVM for face mask classification.

The following are the paper's main contributions.

- Developed a face dataset from different poses, and directions, with mask without mask images using a multiple free online database of face images.
- For face image segmentation, a new region-based clustering method known as K-medoid clustering has been proposed.
- To increase the identification accuracy of existing recognition techniques for face mask detection, convolutional neural network pre-trained models(VGG16, VGG19, and DenseNet201 were used.
- A comparative study of different methods for face mask dataset pre-processing and applied the pre-trained models to improve the accuracy.
- The proposed model DenseNet201 approach is evaluated using F1-score, precision, sensitivity, specificity, and accuracy.
- The development of an automated face mask detection system that enables real-time detection with little resource use. The rest of the sections of this paper are structured as follows: section 2 represents the related works. In section 3, the presented methodology. The method is explained in detail. Section 4 presents the results while section 5 presents the conclusion of the presented method.

2. Related work

There continues to be assurance for machine vision systems in the areas of computer vision, face, and skin recognition, etc. Applications for image processing have demonstrated that analyzing images is a successful technique for research. The identification of face masks is a crucial problem that requires the development of sophisticated automatic detection methods. The advancements in machine learning and deep learning have made it feasible to detect faces using technology rather than human observation.

Our objective is to create a more straightforward and effective K-medoid clustering technique. For the same goal, various other

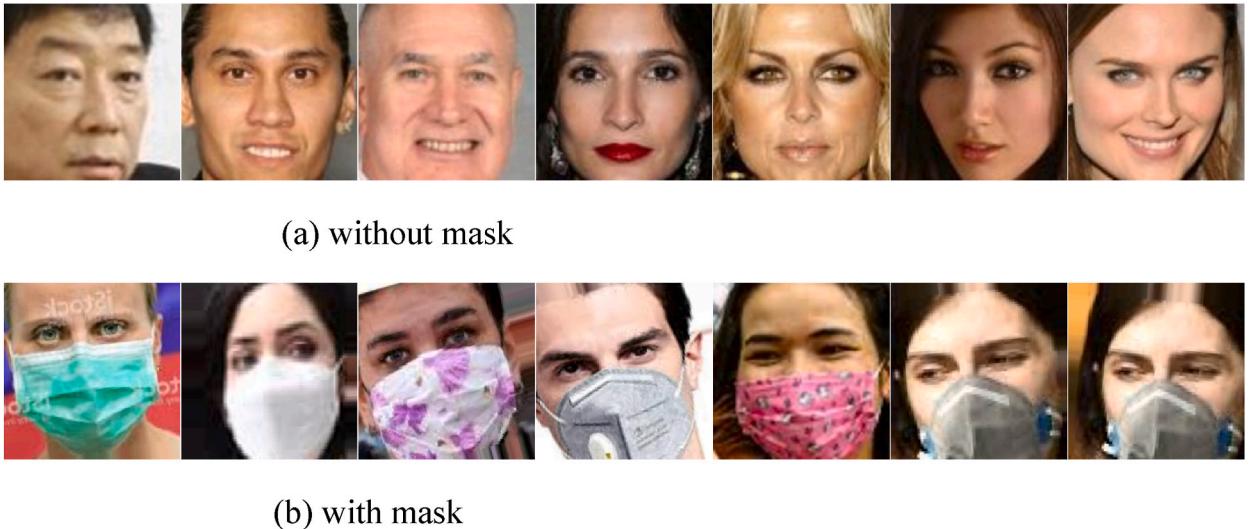


Fig. 1. (a–b) Represents the sample images of our dataset.

ways are accessible in the published works. Rather than applying PAM to all objects in a dataset, kambe, and Pe et al. [21] created CLARA, which applies PAM to derived objects only. CLARA's effectiveness deteriorated with the addition of more clusters, according to Verma et al. [22] suggested a method that maximizes the silhouette rather than minimizing the sum of distances to the nearest medoid. The suggested technique aims to balance network energy usage while extending its lifespan. To combine local search with the K-medoids algorithm and global search by the particle swarm optimization method [23], the hybrid K-medoids and QKSCO algorithm.

The dimension of the feature space is a common basis for splitting FKM approaches into two categories. The first speaks of executing fuzzy clustering in the initial high-dimensional space. Another method, known as extended FKM clustering, teaches the fuzzy membership connection in the low-dimensional space. All characteristics provide the same weights to the membership relationship learning for the first class of FKM algorithms. Numerous fuzzy theories and techniques have been developed recently to enhance the functionality of the original FKM [24]. To build a data entropy-based split network, for instance, maximum entropy fuzzy clustering techniques are given. These algorithms have a distinct physiological significance. With a weaker restriction on the membership, Pal et al. [25] suggested a possible FKM method that combines FKM and probabilistic K-Means. A few researchers additionally take into account the geometric details of a single pixel and the impact of the neighborhood to produce excellent picture segmentation results [26].

Abou Chaaya et al. [27] suggested a CNN model with two-stage neural networks for detecting face masks in crowded areas. Additionally, the researcher proposed a CFMD-Net which is a two-stage detector to overcome the crowded environment situation. The results showed that the average categorization accuracy was 96.5 %. Deep learning models, as opposed to shallow models, have recently become more popular in the development of object identification algorithms [28]. Because of this development, it is now believed that deep learning models are more capable of completing complicated tasks than shallow models. The creation of a model or system that works in real-time and can tell if people are wearing masks or not when used in public places would be a nice example of technology. Employed real-time deep learning to identify and differentiate emotions, using VGG-16 to classify seven faces [28]. This tactic works effectively during the current COVID-19 lockdown phase, which aims to halt the disease's future spread. Principal analysis of components was also used to distinguish between persons who had their faces covered from those whose have them uncovered [29].

In terms of datasets and detection methods, this section gives a broad overview of the literature on face mask detection. Several computer vision problems, including face identification, face tracking, face retrieval, and face occlusion detection have been the focused. Researchers used PCA [30], SVM [31], and Markov models [32] to carry out different tasks related to face occlusion detection, including head identification, facial feature detection, mask detection, skin color-based detection, and face parts detection. Low-resolution images of the monitoring instruments were the biggest problem for this research.

Shijie et al. [33] introduced a transfer learning method for the identification and categorization of pests and illnesses affecting tomato plants, using VGG16. Additionally, the experimented VGG16 feature extractor with SVM as a classifier with an average accuracy of 89 %. The VGG16 method to transfer learning outperformed the VGG16+ SVM strategy. Hemming et al. [34] have suggested a method that uses a deep CNN model to identify tomato whiteflies and the bugs that prey on them. The yellow sticky trap method was used to manually count the insects, and the results were compared. The results showed that the average categorization accuracy was 87.40 %. This research work finds the research gap in the literature review improves the high accuracy level using the K-medoid clustering for image segmentation.

Sheikh and Zafar et al. [35] suggest a transfer learning method MobileNetV2, ResNet50, and EfficientNet-B2, against such attacks, a face mask detection algorithm that is resilient to adversarial attacks with an accuracy of. For a similar scenario, the proposed transfer learning method with only a single fine-tuning MobileNetv2 achieved a better accuracy of 92.79 %. In both above works, researchers

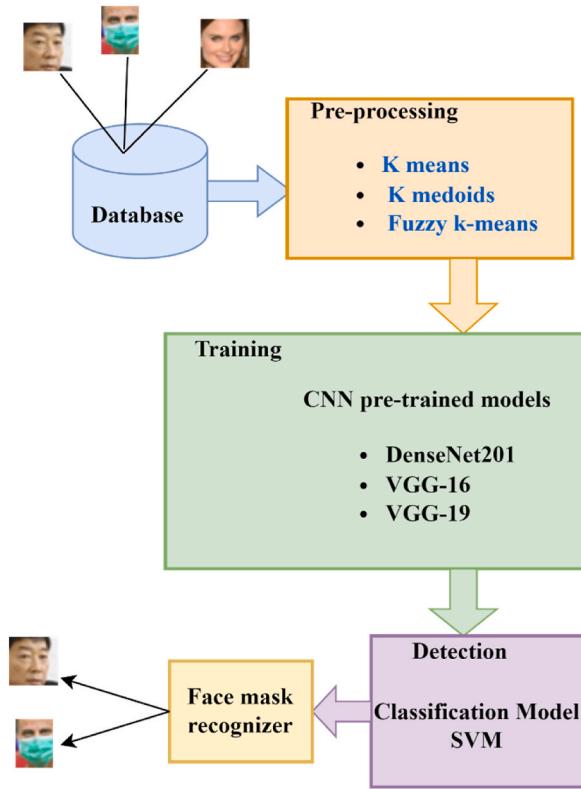


Fig. 2. Block Diagram of the overall work system.

have experimented with the fast gradient sign method (FGSM) and projected gradient descent (PGD) for face mask detection [36].

3. Methodology

3.1. Data description

Fig. 1 incorporates both types of images owing to the absence of a standardized dataset for the detection of face masks and recognition of masked faces. The 2000 different types of images from the dataset utilized in this research, which includes data both with mask and without mask in different poses light condition images as shown in **Fig. 1**, were taken from the Kaggle repository [45]. Different datasets are needed for each of these activities. Facemask detection makes up the bulk of our datasets. Only 2000 images from the dataset were taken into consideration. The face mask detection images collection consists of photographs of faces with and without masks as shown in **Fig. 1**. Because square-shaped images (usually $224 \times 224 \times 3$ pixels) are often used as inputs by traditional AI deep models, additionally total pixels to the boundary of several images. Finally, each picture was cropped and resized to $224 \times 224 \times 3$ pixels using bilinear interpolation.

3.2. Image segmentation for using the clustering methods

3.2.1. K-means clustering

This method is particularly useful in image processing, where it can segment images into distinct regions based on pixel similarities, enhancing the ability to detect and analyze specific features. By reducing the complexity of the data and highlighting important structures, K-means clustering aids in improving the accuracy and efficiency of machine learning models. The K-means cluster analysis is the partitioning technique used most often.

This is how the algorithm works.

- (1) Choose K centroids (K randomly selected rows).
- (2) Pick K items at random from the data collection to serve as the first cluster means or centers.
- (3) Based on the Euclidean distance between the item and the centroid, assign each data point to its nearest centroid.
- (4) The new mean values of each of the K cluster's data points are calculated to update each cluster's centroid.

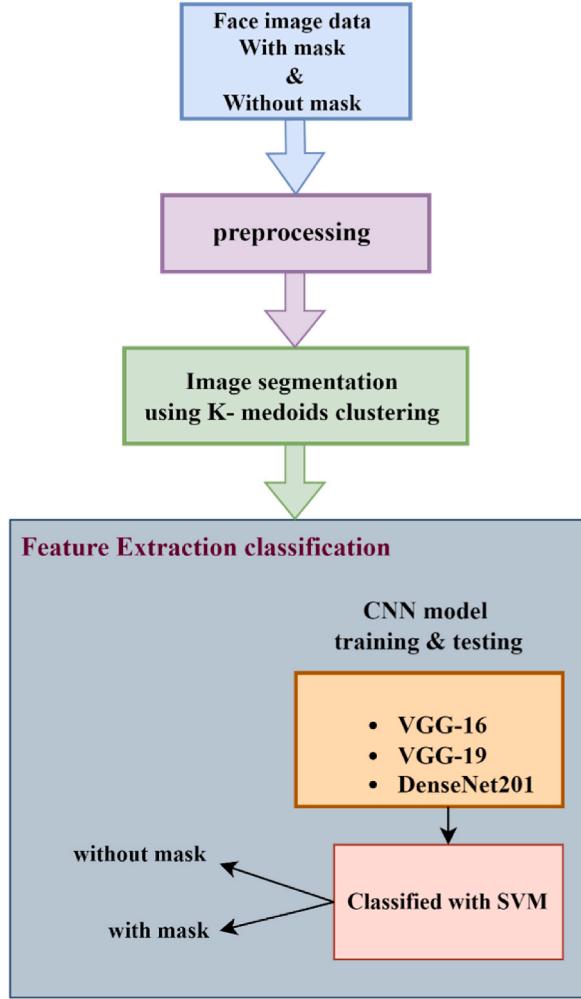


Fig. 3. Demonstrate Block diagram of the proposed method.

A vector of length p that contains the K-means of all the variables for the observations in the cluster is the centroid of a k^{th} cluster; p is the number of variables.

Repeat steps 3 and 4 until the cluster assignments stop shifting or you reach the optimum number of repeats.

$X = [x_1, \dots, x_n] \in R^{d \times n}$ where d is the number of features and n is the number of data points. Finding the data split that minimizes the following eq. (1) objective function is the goal of the K-means clustering technique.

$$J(X, V) = \sum_{j=1}^k \sum_{x_i \in X_j} \|x_i - v_j\|_2^2, \quad (1)$$

Where K is the number of clusters, v_j is the cluster center of the j^{th} cluster, and contains data points that belong to the j^{th} cluster.

It $H = \{h_{ij}\} \in R^{n \times k}$ is the cluster indicator matrix and its element is defined as $h_{ij} = 1$ if it belongs to the j^{th} cluster, and otherwise. Then, the objective function of K-means can also be written as following eq. 2

$$\min_{H,V} \sum_{i=1}^n \sum_{j=1}^k h_{ij} \|x_i - v_j\|_2^2. \quad (2)$$

Iteratively minimizing the sum of distances between each element and its cluster centroid for K clusters is the foundation of the K-means algorithm. As soon as the total cannot be dropped anymore, the samples are moved across clusters.

3.2.2. Proposed method K-medoids

The K-medoid clustering algorithm employs a medoid or actual cluster point to serve as the representative center of the cluster. The center of a cluster is typically identified by a method or a cluster center, as noted in sources [46].

Furthermore, the medoid is situated at the minimum total distance from all other entities (points). The utilization of a medoid as a cluster center is deemed effective in accurately representing the cluster center, primarily due to its robustness against noise and outliers. The PAM algorithm is a widely used K-medoid clustering technique. The algorithm initiates by selecting K objects or points and subsequently iteratively approaches the optimal cluster objects or points. Subsequently, the clustering efficacy is evaluated by scrutinizing all possible point pairings. If an object (point) is identified with the most optimal distortion function value, it shall supersede the present best object (point). The recently produced optimal entities, referred to as points, comprise the upgraded and refined medoids. The algorithm aims to minimize the disparity between objects and their corresponding reference points. The process assists in reducing the total amount of differences between a given entity and its nearest neighboring cluster. To clarify, the K-medoids algorithm endeavors to minimize the objective function known as the absolute error (E). The aforementioned function is presented in the form of an eq. (3).

$$\text{Absolute error}(E) = \sum_{j=0}^k \sum_{p \in C_j}^n |p - ob_j| \quad (3)$$

In the context of cluster analysis, Eq. (3) the sum of absolute error (E) is denoted as a function of a data point (p) that represents an object within the cluster. Additionally, the representative object of the cluster (C_j) is also denoted. The above-mentioned procedure is repeated until the object within the cluster that is (C_j) situated at the centremost position, also known as the medoid, is attained. The present study involves the development of the K-medoids algorithm to cluster n objects into k clusters.

The grayscale image is segmented using a K-Medoids clustering-based technique to separate the facial region from the background. The results show that the K-Medoids are more resilient to noise and outliers when compared to K-means and FKM [47]. The K-Medoids technique selects one real data point from that dataset to be used as each cluster center, in contrast to the K-means approach, which utilizes the average of connected data points in the provided dataset to represent the cluster centers. Fig. 3 displays the block schematic for the segmentation procedure. Using the K-Medoids clustering approach, the K value indicates the number of clusters that need to develop. The K-Medoids clustering technique is used to segment the face region after initializing K = 4 and receiving the grayscale face picture with a mask and without a mask. Therefore, the face image is reduced from the surrounding region on the face mask surface image. Fig. 11 displays the result of grouping facial images using K-medoids.

3.2.3. Fuzzy K-Means clustering

This method is used for the pre-processing of the images it will work the segment the face mask images. FKM assigns each data point to a single cluster with full membership (0 or 1) and allows partial membership. Each data point belongs to all clusters with varying degrees of membership represented by values between 0 and 1. This method is one of the original algorithms projected to manage overlap clusters [24]. FKM is predicated on assigning each data point to numerous cluster samples based on its degree of participation. Technically, a data set is represented $X = [x_1, \dots, x_n] \in R^{d \times n}$ where n is the number of data points and d is the dimension. $x_i \in R^d$ is the point of data. Assume that these data points correspond to c clusters. The objective of FKM is to divide n data points into C clusters in the absence of label information [48] is defined as following Eq. (4).

$$\min_{U,M} \sum_{i=1}^n \sum_{k=1}^c u_{ik}^h \left\| x_i - m_k \right\|_2^2, \quad (4)$$

$$\text{s.t } \sum_{k=1}^c u_{ik} = 1, 0 \leq u_{ik} \leq 1,$$

The fuzzy exponent h, which is often modified to a real value greater than 1, is used for modifying the fuzzy degree. u_{ik} is the component of the matrix $U \in R^{n \times c}$ and represents the amount of the i^{th} data points membership in the cluster. m_k is the prototype cluster for the k^{th} cluster, and $M = [m_1, \dots, m_c]$. By modifying the parts of U and M according to the following Eq. (5) and Eq. (6) instructions may get the best resolution to problem (1):

$$u_{ik} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - m_k\|_2}{\|x_i - m\|_2} \right)^{\frac{2}{h-1}}}, \quad (5)$$

And

$$m_k = \frac{\sum_{i=1}^n u_{ik}^h x_i}{\sum_{i=1}^n u_{ik}^h}. \quad (6)$$

When h = 1 the problem (1) is equal to

$$\min_{U,M} \sum_{i=1}^n \sum_{k=1}^c u_{ik} \left\| x_i - m_k \right\|_2^2,$$

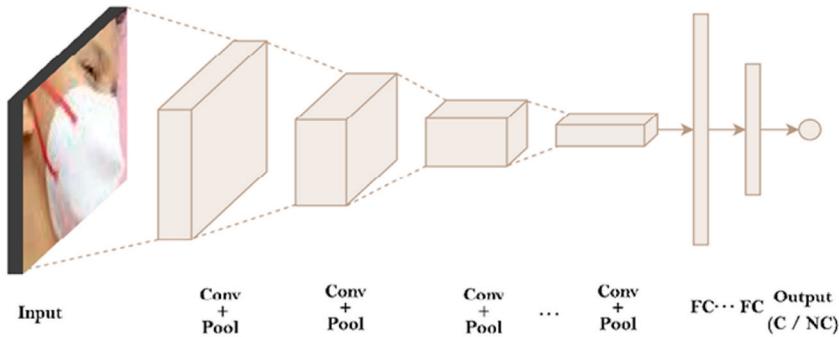


Fig. 4. Architecture of convolutional neural networks.

Table 2

Face mask detection of CNN pre-trained models Hyperparameters.

Parameters	VGG 16	VGG19	DenseNet201
Learning Rate	0.01	0.01	0.001
Batch size	32	32	32
Epochs	30	30	30
Input layer size	224 × 224 × 3	224 × 224 × 3	224 × 224 × 3

$$\text{s.t } \sum_{k=1}^c u_{ik} = 1, u_{ij} \in \{0, 1\},$$

It is precisely the intended purpose of k-means. If the first cluster examples are provided, U is calculated as.

3.3. Convolutional neural networks approaches

The image-take technique creates a facial picture collection that includes every surface problem. Preprocessing involved K-medoid, K-means, and FKM clustering to eliminate noise from the obtained picture. Following the application of the filter, facial surface pictures are separated from the background area using clustering methods. The CNN received the dataset's separation image through segmentation for training. Utilized the three distinct CNN models VGG 16, VGG19, and DenseNet201 with SVM for face classification.

CNNs are an important tool in machine learning. The whole CNN algorithm structure consists of an input layer, convolutional layers, pooling layers, fully connected layers, a classification layer, and at last the final output layer [49]. Fig. 4 illustrates the CNN's internal structure. The essential element of CNN is the convolutional layer. It performs a process known as a "convolution operation" which uses a filtering technique to build an activation on an input. It is possible to extract features from a picture, such as edges, textures, and objects, by employing convolutional layers. Updates to the filter weights during training led to the creation of the feature maps. Max-pooling and average-pooling are the two types of pooling layers, and both are used to reduce the top layer's dimension. The combination of the convolution and pooling layers qualifies it as a feature extractor. The completely linked layers are used during the classification step [50]. ReLU (Rectified Linear Unit) is a popular activation function in CNNs because of its straightforward implementation and high performance. Represented as $\text{ReLU}(x) = \max(0, x)$ Where x is the input to the neuron. This function outputs x if x is positive; otherwise, it outputs 0. It provides non-linearity, computational efficiency, and sparse activation by passing through positive inputs unchanged while outputting zero for negative inputs. This property helps to alleviate the vanishing gradient problem, enabling quicker and more consistent convergence during the training process.

The SoftMax function is used to convert features into class probabilities. This layer includes the same number of units as classes. The SoftMax function is given by Eq. (7).

$$\text{Soft max}(a_i) = \frac{e^{a_i}}{\sum_{j=1}^m e^{a_j}} \quad (7)$$

Where $\text{SoftMax}(a_i)$ and a_i represent the probability and feature of the class, respectively. The denominator is used to regularize the probability distribution across m classes.

The learning rate (α) is an essential hyperparameter in CNNs that determines the step size for updating the network's weights during training. It specifies the magnitude of adjustments made to the model in response to the calculated error with each weight update. Table 2 shows the CNN hyperparameters used with different pre-trained models throughout the training process.

Face verification has made major advancements as a result of CNN's recent success. our suggested CNN-based models VGG-16, VGG-19, and DenseNet201, in this area. This research concentrated on the network architecture of the proposed CNN since the structure of CNN has been described in several studies [51].

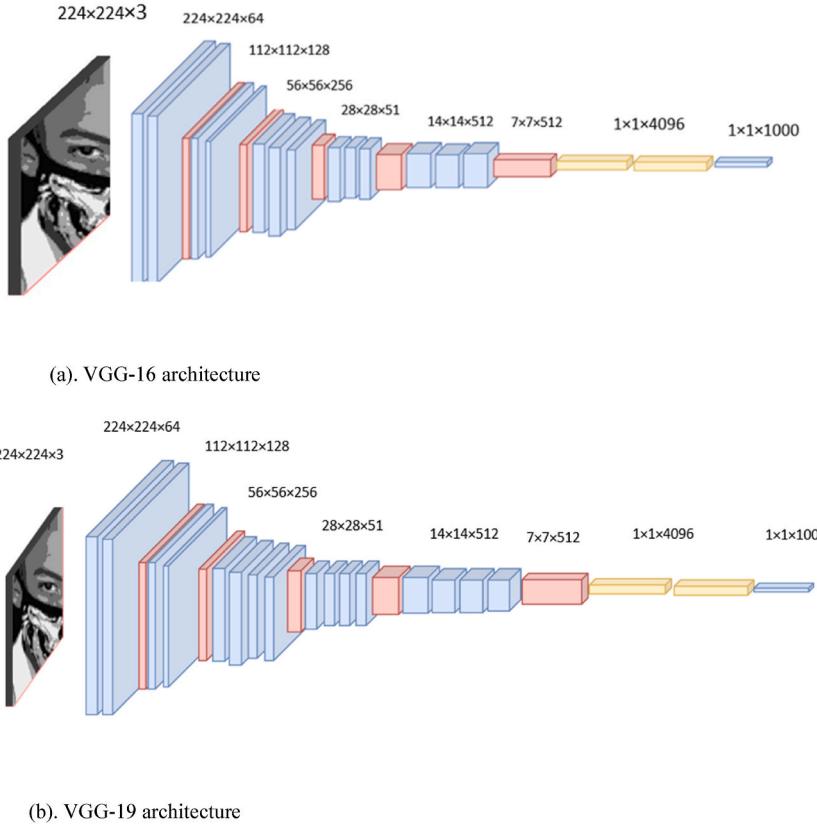


Fig. 5. (a–b) Represents the architecture of CNN pre-trained models.

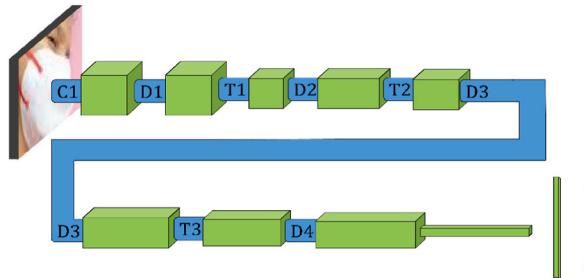


Fig. 6. Architecture of DenseNet201.

The suggested simulation uses VGG-16, VGG-19, and DenseNet201, three distinct pre-trained DCNN models. The images have been downsized to a standard size of 224×224 pixels to match the default input size of the VGG-16, VGG-19, and DenseNet201, CNN architectures as shown in Figs. 5 and 6. This is a result of the various image sizes in the extended face image collection of data. Throughout the training process, utilized the image objects to help reduce over-fitting, a common issue when using pre-trained CNN models with limited data.

3.3.1. VGG-16 and VGG-19

The first layer of the VGG-16 and VGG-19 network in the proposed models, as illustrated in Fig. 5 has been supplied with the 224×224 pixel images for feature extraction [52]. VGG-19 and VGG-16 are two different designs of the VGG network. The number of layers in each design is represented by the numbers 16 and 19. For example, VGG-16 has 16 convolutional layers, 3 fully connected layers, 5 max-pooling layers, and 1 SoftMax layer. 13 convolutional layers, two fully linked layers, and one SoftMax classifier make up this structure. The image processing begins with multiple convolutional layers. A 224×224 -pixel image goes through the first set of two convolutions with a receptive area of 3×3 after applying ReLU activation functions. Each level has 64 filters. The stride value is always set to 1, and padding remains at one pixel to maintain full spatial resolution. Thus, the output activation dimensions match

	Positive	Negative	
Positive	True Positive (TP)	False Positive (FP)	Predicted class
Negative	False Negative (FN)	True Negative (TN)	
True class			

Fig. 7. Confusion matrix.

those of the source images. Next, the activations are sent to a pooling layer with a stride value of 2 pixels since the window size is set to 2×2 . This reduces the magnitude of activations by half. At this point in the first stack, the output size becomes $112 \times 112 \times 128$.

The output from this stage then goes through additional convolutional layers before being passed to another pooling layer. This process continues until reaching the final convolutions and pooling layer steps. Each time, the output size decreases while maintaining depth.

3.3.2. Proposed model DenseNet201

Another among the DenseNet set of architectures created for image classification is DenseNet201. This was the ImageNet challenge's 2015 top-performing model. DenseNet layers have immediate accessibility to both the color gradients of the Loss function and the initial input image. DenseNet is a great option for image classification jobs due to the greatly decreased computing cost [53]. The network loaded into pre-trained weightiness from the ImageNet database.

The architecture of DenseNet201 is illustrated in Fig. 6, with the primary element being the dense block. Similar to the residual structure in ResNet [54], a dense block is divided into two parts: the backbone and the residual edge. The backbone consists of a 1×1 convolution followed by a 3×3 convolution, both with a stride of 1×1 , while the residual edge remains unprocessed. These two parts are then merged using a concatenation layer to integrate the features. DenseNet201 is built by stacking multiple dense blocks along with convolutional and pooling layers.

3.4. Model evaluation and verification

The confusion matrix, sometimes referred to as the error matrix, evaluates the precision of a classifier's categorization. When analyzing the output of binary regression models like logistic regression and SVM, the confusion matrix is often employed shown in Fig. 7. The accurate rate of 0-value prediction, the correct rate of 1-value prediction, and the total prediction rate in the model outputs may all be expressed quantitatively using this approach [55].

Precision: Precision is defined as the ratio of accurately anticipated positive occurrences to all expected positive outcomes.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Sensitivity: Sensitivity is usually calculated as follows and is the only precise positive metric that is proportionate to the total number of instances:

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

Recall and "True positive rate" are two more terms for sensitivity (TPR)

Specificity: The concept of "specificity" refers to the quantity of accurately identified and calculated true negatives and the technique below may be used to calculate it. $\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$

$$\text{Specificity} = \frac{TN}{TN + FP}$$

Accuracy: The accuracy is the sum of all instances that were successfully identified across all cases. Accuracy is measurable by

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

F1-Score: The F1-score is the harmonic mean of recall and accuracy. The F1 score of 1, which represents perfect recall and accuracy, is the highest achievable. $F1 - Score = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$.

3.4.1. ROC curve

A thorough indication of response sensitivity and particular factors is the Receiver Operating Characteristic (ROC) curve [55]. The specificity of the ROC curve's X-axis, which is used to quantify the risk of landslides, represents the likelihood that non-disaster points would be incorrectly predicted. The sensitivity, which represents the probability of the crisis point being predicted correctly, is shown on the Y-axis. The size of the region bounded by the curve and the abscissa is a measure of the model's predictive power. The accuracy of the model increases with the distance of the curve from the top left corner. The range of area under the curve (AUC) values is $[0, 1]$, and the term "area under the curve" is used. AUC values that are nearer 1 show that the model is more accurate.

3.5. Classification model

3.5.1. Support vector machine (SVM)

In this section, we opted for Support Vector Machines (SVM) due to their outstanding capability in handling problems that are not linearly separable. SVM can identify an optimal separating hyperplane that maximizes the margin between training samples. The objective of SVM is to minimize empirical risk and confidence interval, thereby achieving robust statistical rules for samples and enhancing the generalization capability of machine learning models. For problems that are not linearly separable, SVM projects input data from a low-dimensional space into a high-dimensional feature space, facilitating easier separation.

The element of face detection picture deformation is simulated using Deep neural networks like VGG-16, VGG-19, and ReseNet201, and then identified using SVM. CNN typically makes use of the SoftMax classifier. Assuming that the weight of the one before the last layer to the SoftMax layer and its activation value are the same, the SoftMax layer's input may be interpreted as following Eq. (8).

$$a_i = \sum_k h_k w_k \quad (8)$$

Consider the following for this N-class classification method: let N be the number of nodes in the SoftMax layer and every node is registered as p_i , where $i = 1, 2, \dots, N$, and p_i is a discrete probability distribution such that $\sum_{i=1}^N p_i = 1$. Among them

$$p_i = \frac{\exp(a_i)}{\sum_{j=1}^N a_j} \quad (9)$$

The solution to equation (9) is used to compute the cross-entropy loss function of SoftMax. CNNs can gather visual data short of delivering the best categorization results. SVM with a fixed kernel function is unable to learn the complex image properties. The "soft interval" method may be used to increase the interval to gain any determining planes. Thus, the classification problem may be best resolved in the learning feature space. SVM is commonly used in data analysis, pattern recognition, regression analysis, and other fields including supervised machine learning. Standard SVM predicts which of the two categories each input belongs to, making it a non-probabilistic binary linear classifier. The following is the fundamental tenet of SVM [56].

Setup the training set data samples as follows: $\{(x_i, y_i) | x_i \in R^d, y_i \in \{-1, 1\}, i = 1, 2, \dots, N\}$, y_i for the category name d for the data's dimension and N for the training datasets. There is a generalized optimum categorization hyperplane for linearly separable data sets:

$$w \cdot x + b = 0 \quad (10)$$

Eq. (10) elements work in concert to provide the class interval with the best result, with $\frac{1}{2}\|w\|^2$ the lowest and $\frac{2}{\|w\|}$ the highest. Among these, b is an offset, dot is an inner product operator, and w is an n-dimensional vector. As a result, the following categories may be added to the optimizing problem categorization as shown in Eq. (11).

$$\min \frac{1}{2}\|w\|^2 \text{ such that } y_i(w^T x_i + b) \geq 1, i = 1, 2, \dots, n \quad (11)$$

The learning method used by CNN minimizes errors in the training samples by using empirical risk reduction. Once the back-propagation approach has identified the first classification hyperplane, the training procedure concludes, regardless of whether it is a local or global maximum. The most useful method for classifying SVM on a global scale is using the structure-based risk-reduction principle. SVM provides superior generalization capabilities compared to multi-layered neural networks. Replacing CNNs SoftMax layer with SVM will enhance the classification performance.

4. Results and discussions

We used the MATLAB program to implement the above-mentioned approaches. All tests utilize a Lenovo machine with an Intel(R) Core (TM) i5-1021U CPU running at 2.11 GHz. It meets the following requirements: 512 GB Solid-state Drive and 8.00 GB Random Access Memory (RAM) used for the training process. The dataset was used for training 90 % and testing 10 % during the training process. Execution time has taken around 3–4 h for 30 epochs, furthermore, we can achieve dynamic performance for the proposed model with high-end processors. The primary goal of evaluate the K-means, K-medods, and FKM clustering methods on the dataset for segmentation before giving it to the CNN architecture. This study evaluated to classify Face mask classification using deep CNN, which is more accurate than current state-of-the-art methods. The next subsections go through the operation of the categorization model. The output results for our suggested approaches utilizing pre-trained models. For this research, three different pre-trained architectures

Table 3

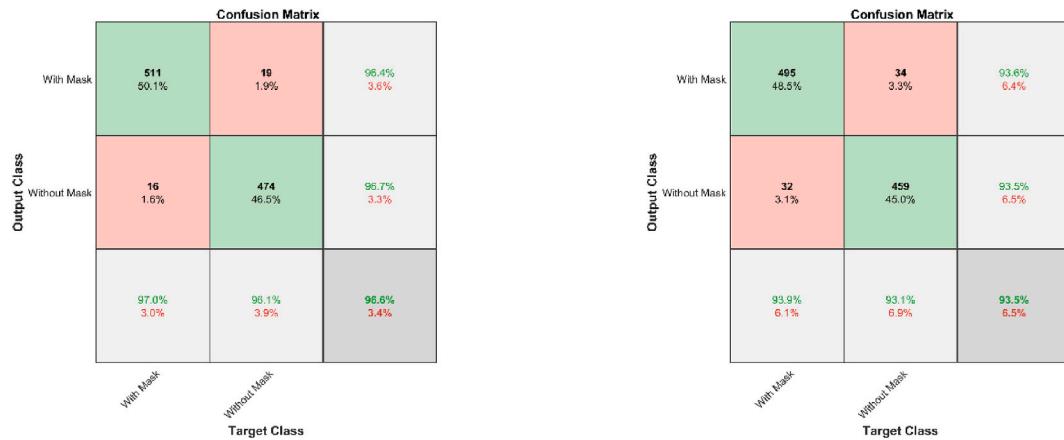
Performance of the k means clustering with DenseNet201.

Name	Classes		Macro AVG	Micro AVG
True positive	511	474	492.5	492.5
False positive	19	16	17.5	17.5
False Negative	16	19	17.5	17.5
True Negative	474	511	492.5	492.5
Precision	0.96415	0.96735	0.96575	0.96569
Sensitivity	0.96964	0.96146	0.96555	0.96569
Specificity	0.96146	0.96964	0.96555	0.96569
Accuracy	0.96569	0.96569	0.96569	0.96569
F-measure	0.96689	0.96439	0.96564	0.96569
Loss	0.0451			
AUC	0.9948			

Table 4

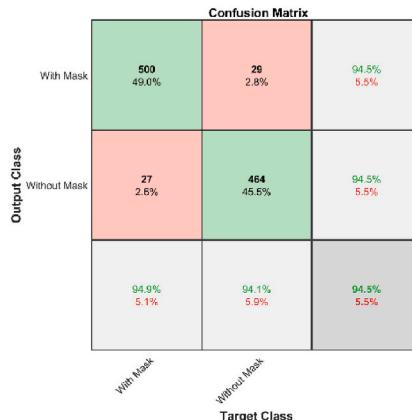
Comparative of deep learning models.

	Model name	Precision%	Recall%	F1-Score%	Accuracy%
K-mean clustering	DenseNet201	96.97 %	97.1 %	97 %	96.7 %
	VGG16	93.5 %	93.9 %	93.6 %	93.6 %
	VGG19	94.5 %	94.8 %	94.6 %	94.5 %



(a) DenseNet201

(b) VGG-16



(c) VGG-19

Fig. 8. (a–c) Represents the Depicts of the confusion matrices of with mask and without mask images.

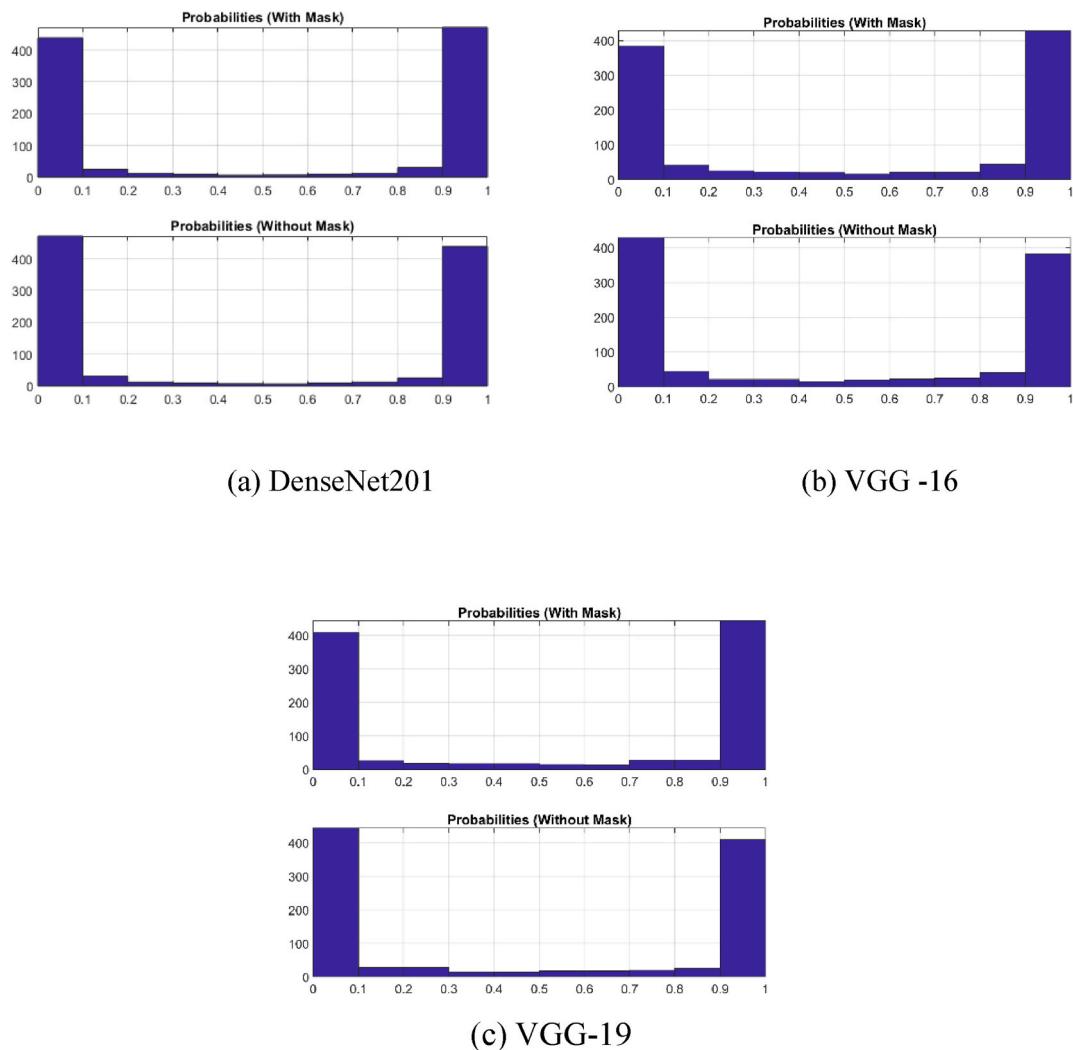


Fig. 9. (a–c) Illustrates the graphics that depict the expected probability scores of both mask- and non-mask images.

(VGG-16, VGG-19, and DenseNet201) are used, with SVM for the classification of face mask detection.

In the second experimental configuration, a completely linked layer with 10 neurons replaces the previous fully connected layer, which had 1000 neurons. The face image dataset, which contains two classes (with mask and without mask), was utilized to train all three pre-trained models that are being analyzed here. As a result, the final layer includes 1000 neurons.

4.1. Results of K-means clustering with pre-trained models

Mask detection techniques may assist with pre-processing with K-means clustering. It could be possible to identify possible face regions for further mask detection processing by segmenting the image into areas depending on intensity. This will be helpful for images with distracting backdrops or different lighting conditions. This paper provides our first findings for the image pre-processing applied to the face image dataset before giving it to CNN. The K-means clustering used for the face mask images will reduce the noise data. This method of clustering selects the usual of connected data points. The K value represents the cluster of centroids. Where $K = 4$ to initialize the face image dataset is used.

The CNN pre-trained models applied the face mask detection after the pre-processing. Although the performance of the resulting classification appears to be good, several restrictions prohibit the findings from being used more widely. To evaluate the models first selected for the binary classification task, then examined images of faces with mask and without mask. The findings were summarised in [Table 4](#) which also included the F1-score, recall, precision, accuracy, specificity, and AUC. The highest accuracy is achieved by DenseNet201 (96.6 %) as shown in [Table 3](#). In this part, comparison [Table 4](#) existed for the classification performance of our three cutting-edge deep CNN models. This experiment compares the performance of the three pre-trained models for face mask identification.

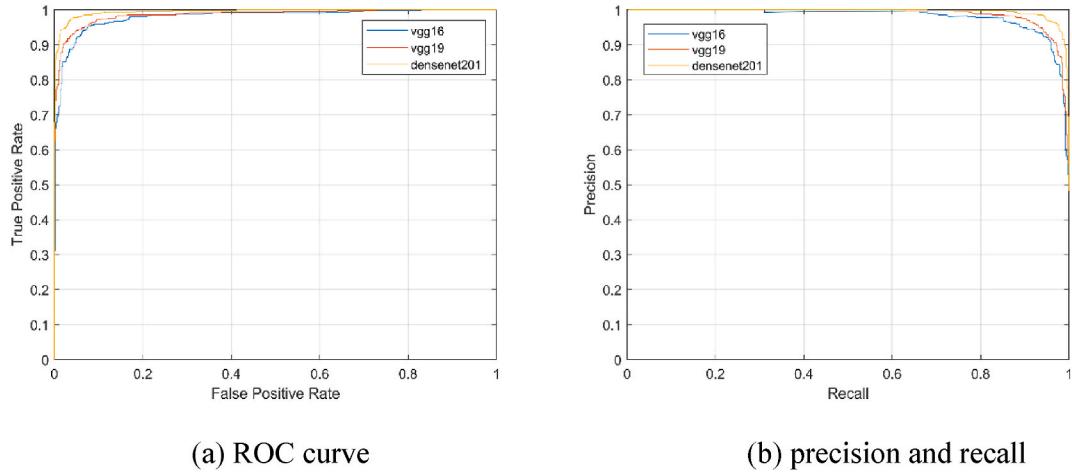


Fig. 10. (a–b) Represents the ROC Curve and Precision-recall curve.



(a). without mask



(b). with mask

Fig. 11. (a–b) Represents the K-medoids clustering images.

Table 4 presents the DenseNet201 model surpassed the other three models by obtaining 96.7 % accuracy for face mask identification, according to these data. With a second-best accuracy of 94.5 % the VGG-19 model performed well. As opposed to VGG-16, had the lowest accuracy of all models at 93.6 %. DenseNet201 on the other hand, fared better than other models in terms of true positive rate, obtaining a recall of 97.1 %. Also interesting is the fact that our suggested Densenet201 technique achieved the best 96.7 % accuracy, 97.1 % recall, 96.9 % precision, and 97 % F1 score.

Table 4 compares the accuracy, precision, recall, and F1-Score performance of a DenseNet201 the most accurate model. Consequently, the DenseNet201 model outperforms all other challenging methods based on different backbones. **Fig. 8** represents the confusion matrices for binary classification(with mask and without mask) performance of the DenseNet201, VGG16, and VGG19 models. **Fig. 8** shows the confusion matrix of the pre-trained models when applied to the face dataset. It can be observed that the proposed model DenseNet201(96.6 %) achieved high accuracy. The lowest accuracy is achieved by VGG -16(93.5 %).

Our work focused on our three recognized pre-trained CNN models; these algorithms should provide a probability to each picture that shows in what way prospective it is to be classified as a face mask. By comparing these probabilities to a beginning, it is possible to create a binary label indicating the with and without face mask. **Fig. 9** displays the with-mask and without a mask using this probability distribution, all with-mask samples should have a projected probability close to 1. In terms of prediction probability, DenseNet201 beat

Table 5

Performance of the K-medoid clustering with DenseNet201.

Name	Classes		MacroAVG	MicroAVG
True positive	514	483	498.5	498.5
False positive	10	13	11.5	11.5
False-negative	13	10	11.5	11.5
True negative	483	514	498.5	498.5
precision	0.98092	0.97379	0.97735	0.97745
Sensitivity	0.97533	0.97972	0.97752	0.97745
Specificity	0.97972	0.97533	0.97752	0.97745
Accuracy	0.97745	0.97745	0.97745	0.97745
F- Measure	0.97812	0.97674	0.97743	0.97745
Loss	0.0284			
AUC	0.9980			

Table 6

Comparative of deep learning models.

	Model name	precision%	Recall%	F1-Score %	Accuracy%
K-medoids clustering	DenseNet201	98 %	97.9 %	97.9 %	97.7 %
	VGG-16	95.4 %	95.1 %	95.2 %	94.8 %
	VGG-19	97.4 %	97.7 %	97.5 %	96.2 %

all other models.

To provide a summary of the results of each of these techniques, this work has included ROC curves. It performed an investigative investigation to evaluate the performance of various methods in terms of a probability model's histogram, ROC and AUC curves, precision-recall curve, accuracy, sensitivity, specificity, and F1 score. The suggested method performs better for classifying with-mask and without-mask detection than the current approaches. Fig. 10 presents the precision-recall and ROC curves for the test set, showcasing the performance of each of the three pre-trained models. DenseNet201 achieves the highest precision(96.97 %) and recall (97.1 %). The pre-trained models ROC and AUC values are plotted against the test set's true positive rate (TPR) and false positive rate (FPR).

The FPR vs. TPR graphic results in the ROC curve. This demonstrates that the three model's ROC curves exhibit comparable behavior. DenseNet201 (AUC = 0.9948) exhibits the greatest performance.

4.2. Results of the proposed method of K-medoid clustering

K-Medoid is a point in the cluster from which dissimilarities with all the other points in the clusters are minimal. Instead of centroids as reference points in K-means algorithms, the K-medoids algorithm takes a medoid as a reference point. The details information on K-medoids is described in Section 3.

K-medoids is an unsupervised clustering approach where “methods” data points function as the cluster's center. A method is the cluster point with the smallest sum of distance (also known as dissimilarity) to all other cluster objects. The distance may be calculated using the Euclidian distance, the Distance measure, or any other appropriate distance function. Hence, this method separates the data into K clusters by selecting K-medoids from the sample data. Using this method approach, the K value indicates the number of clusters that need to develop. For face area segmentation, the grayscale face image is passed to the K-medoids clustering method once K = 4 has been initialized. An example of a K-medoids clustered image is shown in Fig. 11.

4.2.1. Results of K-medoid clustering with pre-trained models

This section, explains about to verify the K-medoids method. Extensive pre-processing is conducted on the face mask (with and without mask) dataset. To improve classification accuracy, this study develops a collection model that combines CNN and SVM. The tests to classify with and without a mask verify the approach's influence on classification. The proposed model performs better. Evaluations have been done for the performance of the DenseNet201 model to that of various current models for this purpose. i.e.; Vgg16, and VGG-19. To assess the efficacy of the pre-trained models were selected for classification, and then evaluated face images with and without masks. The results are summarised in Table 5 which also includes the F1-score, precision, recall, specificity, accuracy, and AUC. With an accuracy of 97.7 %, DenseNet201 gives the best results. In this part, as shown in Table 6, later the comparison of the classification performance of our three cutting-edge deep CNN models was given.

This experiment compares the DenseNet201 model's face mask detection effectiveness against that of other deep learning models currently in use. The results are shown in Table 6. These results influenced us that the DenseNet201, VGG-16, and VGG-19 models were used for face mask identification. The DenseNet201 model achieved the best accuracy of 97.7 %. The least accurate model was VGG-16 which had a 96.5 % accuracy. By reaching a recall of 97.9 %, DenseNet201, on the other hand, performed well better in terms of true positive rate than other models. It should be highlighted that the four-comparator model's accuracy, which was more than 94 %, was VGG16, and VGG19. Another significant achievement was the greatest 97.7 %, precision, recall, F1 score, and accuracy for our

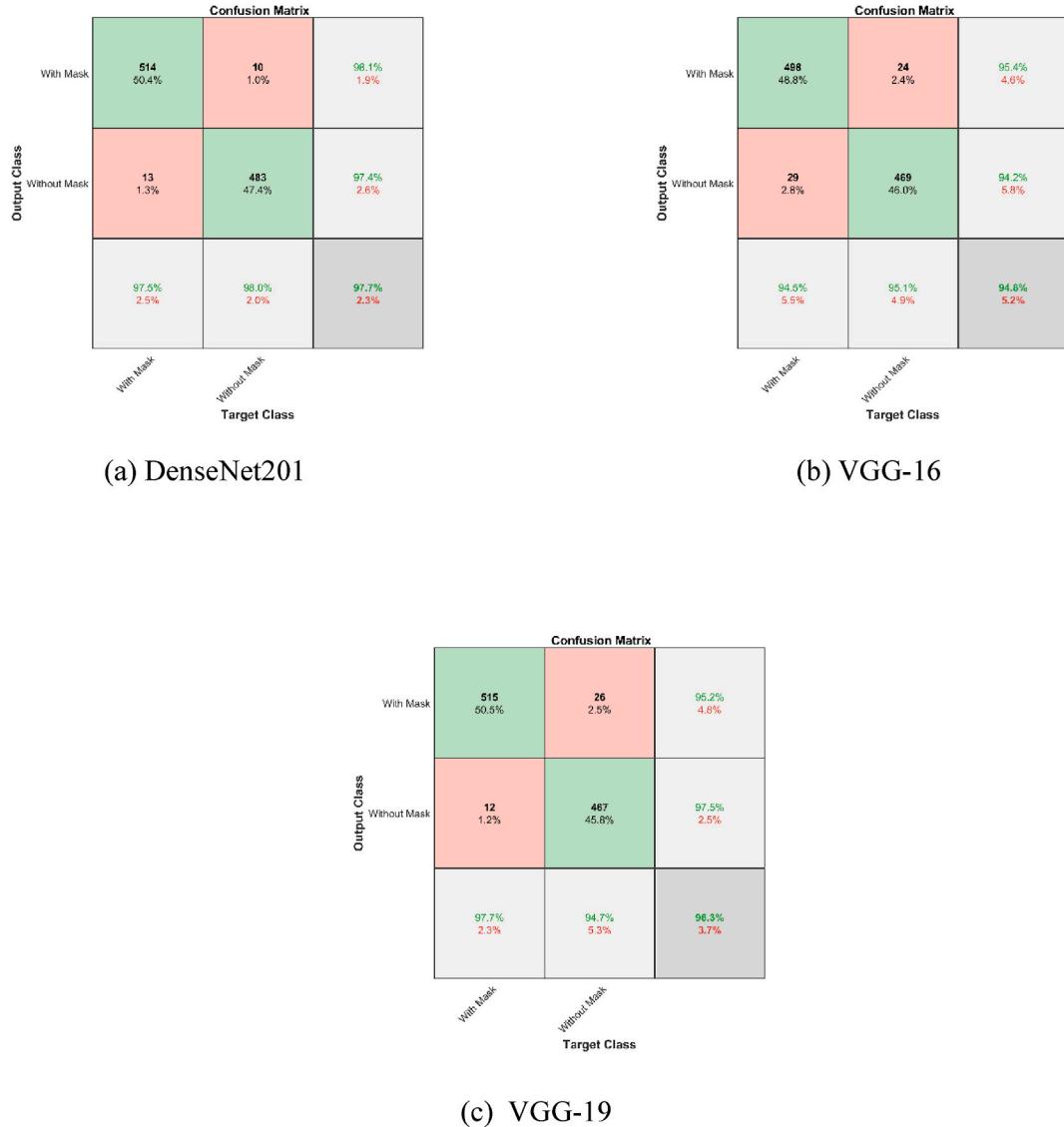


Fig. 12. (a–c) Represents the confusion matrices of with-mask and without-mask images.

recommended model.

Fig. 12 shows the confusion matrices for binary classification for the DenseNet201, VGG16, and VGG19 models. Fig. 7 shows the confusion matrix performance based on the two different classes. It classifies the face mask detection based on the TP and TN for each class, where TP means the model has detected the class correctly, while TN means the model has detected the class incorrectly. Based on the confusion matrix DenseNet201(97.7 %) achieved the best accuracy compared to the other models as shown in Fig. 12. The second highest accuracy was VGG-19(96.2 %) and the lowest accuracy was VGG16 (94.8 %).

This work focused on our three recognized pre-trained CNN models, these algorithms should assign a probability to every picture that demonstrates how likely it is to be classified as a face mask. Compared to these probabilities with a threshold, a binary label representing whether or not the picture is of a face mask may be generated. Fig. 13 shows the probability. All with-mask samples should have a predicted probability near 1, with this probability distribution, it is easy to classify a with mask and without a mask. In terms of prediction probability, DenseNet201 was superior to all other models.

ROC curves have been included to summarise the findings of each of these methods. Also, this work carried out an exploratory analysis of the performance of several techniques in terms of sensitivity, precision, specificity, F1-score accuracy, ROC and AUC curve, precision-recall, and a probability model's histogram. For categorizing the with-mask and without-mask detection, the suggested techniques outperform the present methods. Fig. 14 shows, for the three CNN models, the precision-recall curves for the test set. Fig. 14 displays the ROC curves with mask and without mask detection, along with the test set's TPR and FPR on the vertical and horizontal axes, respectively. The three pre-trained model's performance using the ROC curve is shown in Fig. 14. The proposed model shows

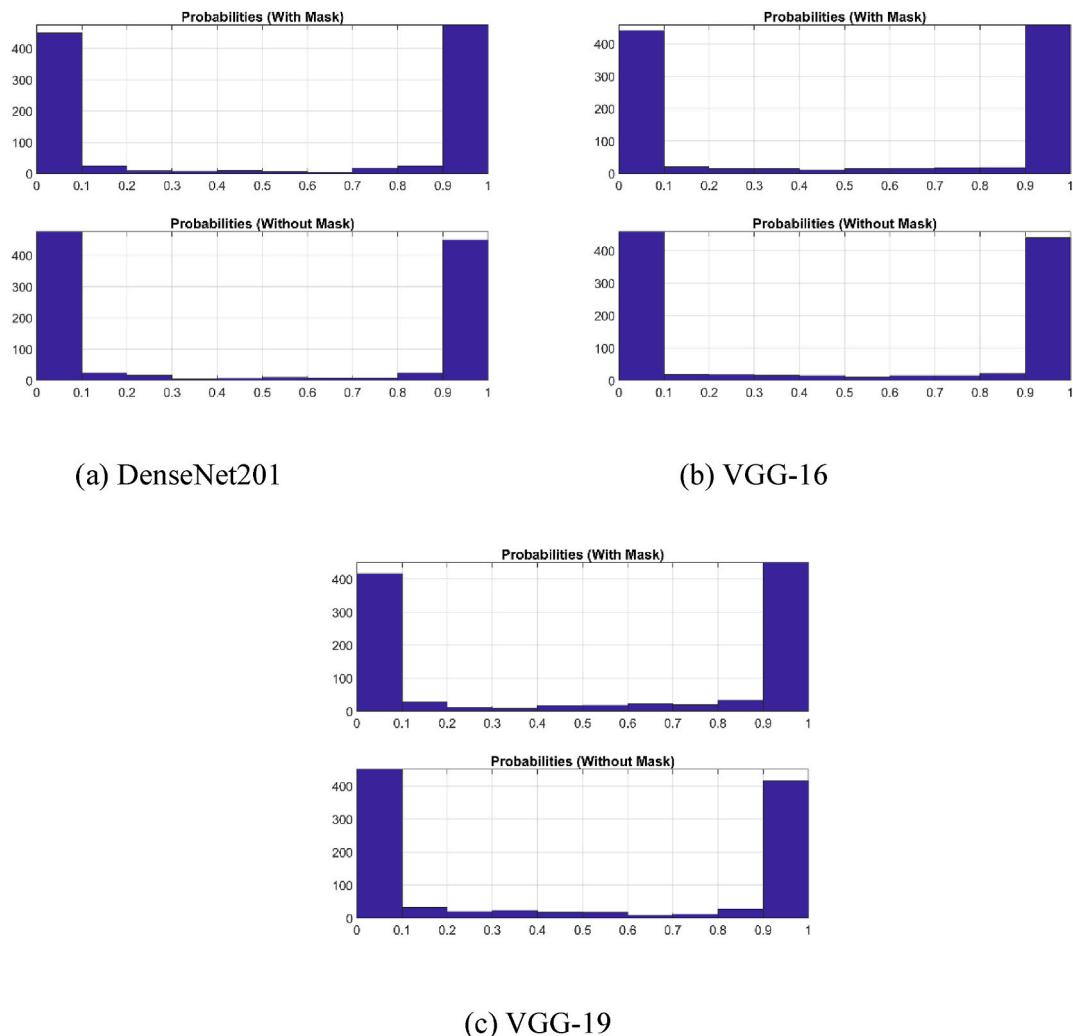


Fig. 13. (a–c) Illustrates the predicted probability scores of images with masks and without masks.

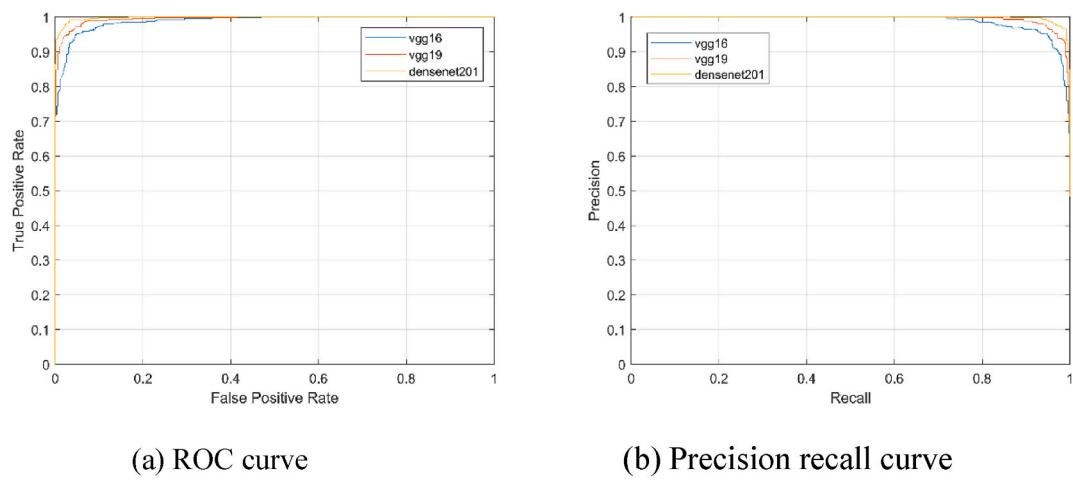


Fig. 14. (a–b) Represents the ROC Curve and Precision-recall curve.

Table 7

Performance of the fuzzy k-means clustering with DenseNet201.

Name	Classes		MacroAVG	MicroAVG
True positive	510	483	496.5	496.5
False positive	10	17	13.5	13.5
False-negative	17	10	13.5	13.5
True negative	483	510	496.5	496.5
precision	0.98077	0.966	0.97338	0.97353
Sensitivity	0.96774	0.97972	0.97373	0.97353
Specificity	0.97972	0.96774	0.97373	0.97353
Accuracy	0.97353	0.97353	0.97353	0.97353
F -measure	0.97421	0.97281	0.97351	0.97353
Loss	0.0304			
AUC	0.9980			

Table 8

Comparative of deep learning models.

	Model Name	Precision %	Recall %	F1-Score%	Accuracy %
Fuzzy k-Means	DenseNet201	98 %	96.7 %	97.3 %	97.3 %
	VGG-19	96.2 %	96.2 %	96.2 %	96 %
	VGG-16	95.6 %	96 %	95.8 %	95.3 %

high performance with an AUC value of 0.9980. the model also achieves a precision of 98 % and a recall of 97.9 %.

4.3. Results of fuzzy K-means clustering with pre-trained models

Fuzzy K-means clustering is an effective method for facial mask dataset analysis. It addresses the inherent ambiguity in mask data, where classification can be challenging due to elements like partial occlusions and light. This method can be used for the pre-processing of the face mask dataset. This method can remove the noise data as well as it will improve the accuracy level. The algorithm assigns partial memberships (between 0 and 1) to each data point (face image) across a predefined number of clusters (K). Using the FKM clustering approach, the K value indicates the number of clusters that need to be developed. The grayscale image is passed to the FKM clustering method for face area segmentation. It has used k = 4 clusters that have been initialized.

This work creates a combined model using CNN pre-trained models and SVM to increase classification accuracy. The tests to classify with and without a mask verify the approach's effects on classification. Evaluations have been done for the performance of the DenseNet201 model to that of various current models for this purpose i.e.; Vgg16, and VGG-19. To assess the efficacy of the pre-trained models were selected for classification, and then evaluated face images with and without masks. The findings are summarised in [Table 7](#) with the F1-score, precision, recall, specificity, accuracy, and AUC. The most accurate algorithm is DenseNet201, which achieves 97.3 % accuracy.

This experiment compares the DenseNet201 model's face mask detection effectiveness against that of other deep learning models currently in use.

[Table 8](#) represents the results of data discovered that the DenseNet201 model beat the other three models for face mask recognition, obtaining 97.3 % accuracy. VGG-19 obtained the second-highest accuracy of 96 % Among all models, VGG-16 had the lowest accuracy of 95.3 %. DenseNet201, on the other hand, beat other models in terms of true positive rate, with a recall of 96.7 %. [Fig. 15](#) represents the confusion matrices for binary classification for the DenseNet201, VGG-16, and VGG-19 models. Each confusion matrix represents the accuracy of each model. [Fig. 15](#) shows the Densenet201(97.3 %) achieved the highest accuracy.

This research mainly focused on their popular, already-trained CNN models. These algorithms should assign a probability to every picture that demonstrates how likely it is to be classified as a face mask. Comparing these probabilities to a threshold can result in a label with a binary value that indicates the probability that the image depicts a face mask. [Fig. 16](#) shows the probability. All with-mask samples should have a predicted probability near 1, with this probability distribution, it is easy to classify a with mask and without a mask. In terms of prediction probability, DenseNet201 was superior to all other models.

ROC curves are included to summarise the findings of each of these models. The exploratory analysis of a probability model's performance using several approaches, looking at its precision, sensitivity, specificity, F1-score, accuracy, ROC and AUC curve, precision-recall curve, and histogram has been done. For categorizing the with-mask and without-mask detection, the suggested techniques outperform the present methods. [Fig. 17](#) displays the test set precision-recall curves for each of the three CNN models. The TPR and FPR of the test set are shown on the vertical and horizontal axes, respectively, in [Fig. 17](#), which also displays ROC and AUC measures.

By graphing the FPR vs. the TPR, the ROC curve is created. It can be seen from this that the three model's ROC curves exhibit similar behavior. DenseNet201 (AUC = 0.9980) shows the greatest performance.

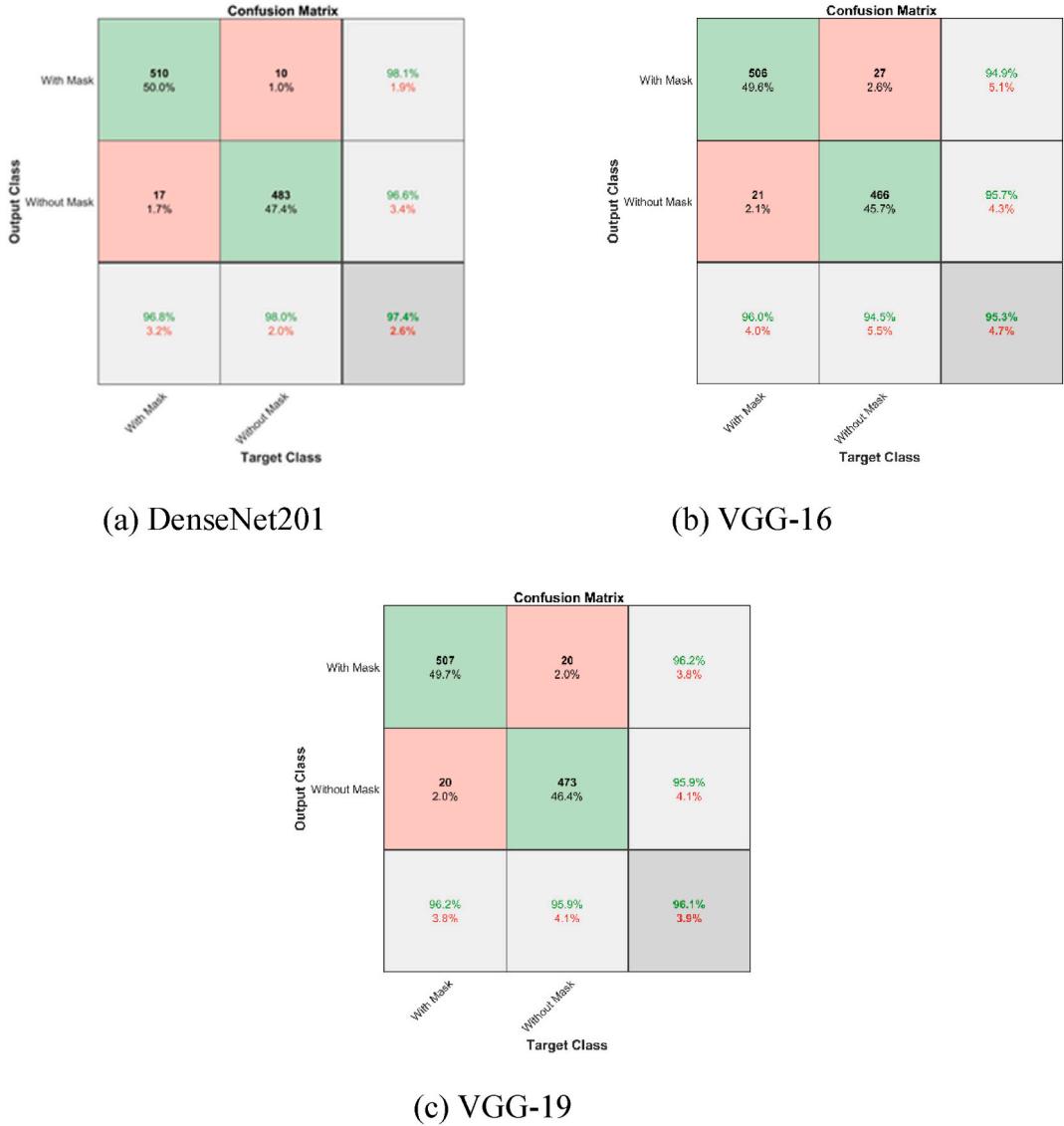


Fig. 15. (a–c) Represents the confusion matrices of with-mask and without-mask images.

4.4. Comparative scheme

Compared to the current methods such as K-means clustering, K-medoids, and FKM clustering for image pre-processing with our techniques. It was observed that as the noise density increases, the performance of clustering algorithms decreases. Furthermore, the effects of salt and pepper noise and Gaussian noise on clustering outcomes are comparable.

Table 9 shows the comparison between different pre-processing approaches on CNN pre-trained models. Our proposed method with the pre-trained model DenseNet201 achieves an accuracy of 97.7 %, precision of 98 %, recall of 97.9 %, and F1-Score of 97.9 %, the best results for face identification. Although VGG 16 and VGG19 have comparable accuracy, both models are difficult because of their huge number of parameters. The K-medoids cluster pre-processing of the face dataset with CNN pre-trained model DenseNet201 achieved the highest accuracy of 97.7 %, precision of 98 %, recall of 97.9 %, and F1-Score of 97.9 %. CNN of different models developed to simulate various data clustering combinations used for training and testing datasets. The emphasized values represent the greatest degree of performance.

The pre-processing of various models on the approached K-medoids clustering, K-means, FKM, and without segmentation has been updated in Table 8. This work projected that the model with pre-trained DenseNet201 shows an accuracy of 97.7 %, an F1 score of 97.9 %, a precision of 98 %, and a Recall of 97.9 %. The segmentation of the dataset has been used for three clustered methods: K-means, FKM, and K-medoids. K-medoids have done the best segmentation on the face dataset trained with a CNN-pre-trained model, and DenseNet201 achieved the best accuracy of 97 % and 98.3 %. In our proposed K-medoids clustering using the dataset samples, the

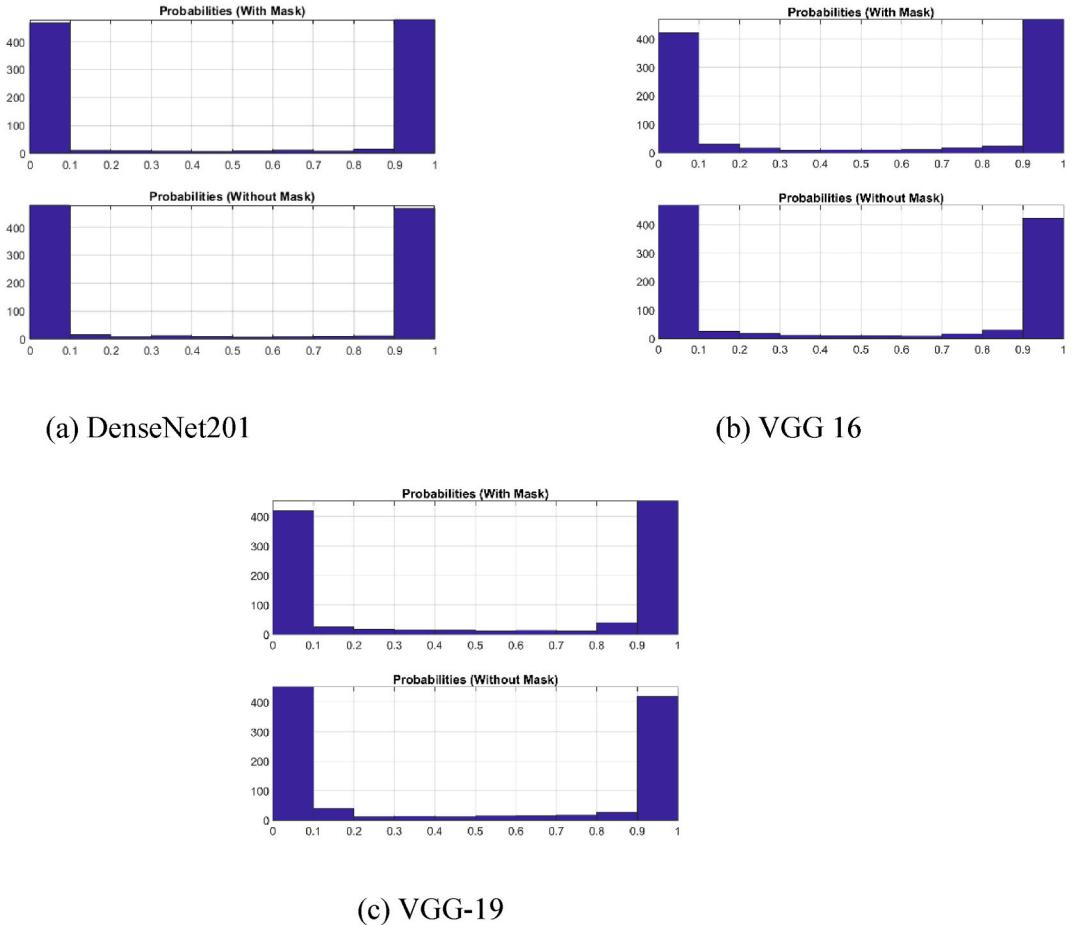


Fig. 16. (a–c) the predicted probability Scores of with-mask and without-mask images.

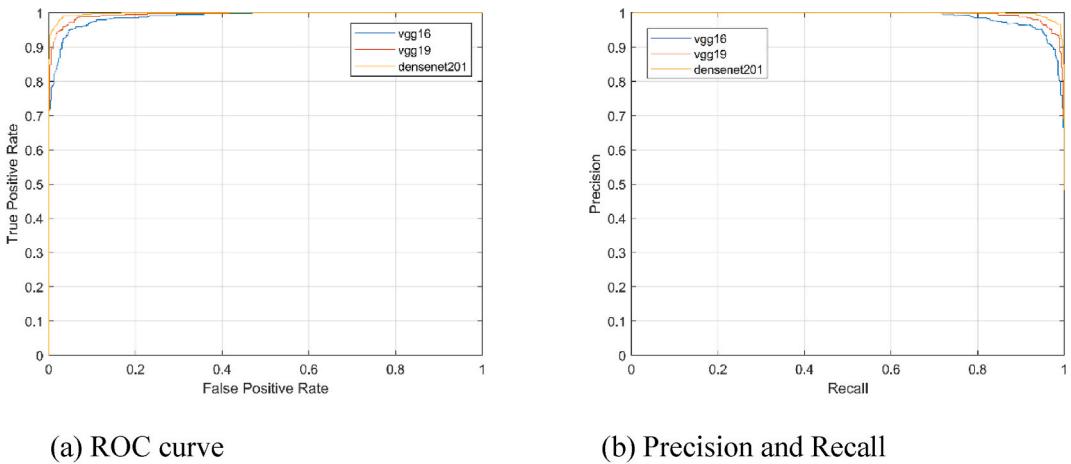


Fig. 17. (a–b) Represents the ROC Curve and Precision-recall curve.

approach of substituting out the sample with and without a mask has been done. At last, the categorization of dataset samples has confirmed the adaptability of our model. Our proposed method with DenseNet 201 achieves the best accuracy compared to the other models as shown in [Table 10](#).

In the case of the dataset, there are only two classes (with mask and without mask), this additionally proposed a K-medoids

Table 9

Comparative of the Semisupervised learning methods with pre-trained models.

	Model	Precision%	Recall%	F1-Score	Accuracy%
K-means Clustering	DenseNet201	96.97 %	97.1 %	97 %	96.7 %
	VGG16	93.5 %	93.9 %	93.6 %	93.6 %
	VGG19	94.5 %	94.8 %	94.6 %	94.5 %
K-medoids	DenseNet201	98 %	97.9 %	97.9 %	97.7 %
	VGG16	95.4 %	95.1 %	95.2 %	94.8 %
	VGG19	97.4 %	97.7 %	97.5 %	96.2 %
Fuzzy K-means clustering	DenseNet201	98 %	96.7 %	97.3 %	97.3 %
	VGG16	95.6 %	96 %	95.8 %	95.3 %
	VGG19	96.2 %	96.2 %	96.2 %	96 %
Without pre-processing	DenseNet201	79 %	78 %	78 %	80 %
	VGG16	75 %	74.3 %	74.2 %	75 %
	VGG 19	72.3 %	73 %	73 %	72.5 %

Table 10

Comparison with existing state-of-art methods.

Author	Method	Accuracy
Lakshmi and Das 2023 [57]	K means clustering with CNN pre-trained models (DenseNet201, VGG-19, VGG-16, ResNet101, ResNet50, and AlexNet) with SVM	92.25 %
Umer et al., 2023 [31]	Depth-wise separable convolutional neural networks	94.1 %
Dash, Sethy, and Behera 2023 [43]	Pre-trained Convolutional Neural Networks	96.5 %
Balasubramanian et al., 2023 [42]	Histogram normalization with RCNN	96 %
Proposed method	K medoid clustering with CNN pre-trained models VGG16, VGG 19, and DenseNet201with SVM	97.7 %

clustering model which gives the best result with the pre-trained model DenseNet201 among all models. Then, for the classification of these two different classes, our proposed method explains the K-medoids of our DenseNet201 model, which has an accuracy of 97.7 %, a precision of 98 %, and a recall of 97.9 %. When used on the test set of the pre-trained models, the proposed method's confusion matrix is shown in Fig. 12. The model has a very high degree of accuracy while making predictions without the use of a mask. A mask results in the highest accuracy DenseNet201 achieved 97.7 % and 97.9 % recall.

The proposed model can utilize different types of applications such as medical image segmentation in real-time, such as distinguishing between different tissue types in MRI or CT scans. This aids radiologists in quickly identifying regions of interest. This method will aim to identify face masks from live streaming in crowded areas as well and assess, if Densenet201 may assist for better results.

The limitations of the method are representing each cluster center, the K-medoids algorithm chooses a single actual data point from a given dataset. The K-medoid method thus offers a more accurate assessment of the cluster centers. When applying the K-medoids clustering approach, the K value denotes the number of clusters that must develop. The grayscale face picture is sent to the K-medoids clustering method designed for face surface segmentation once K = 4 has been initialized. As the cluster centers expand, faces acquire more complex geometric and textural characteristics that could deviate from the algorithm's assumption of simple spherical clusters due to some drawbacks. As a result, this disparity may result in less accurate categorization and less-than-ideal cluster placements.

5. Conclusion

In this research, deep CNN pre-trained models VGG16, VGG19, and DenseNet201 are utilized to categorize face detection with mask and without mask. Additionally, K-means, K-medoids, and FKM clustering are used for the pre-processing of the dataset. Our proposed model K-medoids for pr-processing with DenseNet201 achieves a high 97.7 % Accuracy. However, the accuracy of the proposed method worked, and among our three pre-trained models, the best result was DenseNet201 with 97.7 % after pre-processing of the dataset. Although the model used in this research performs comparably to those without pre-processing, K-means, and FKM approaches, the result shows that the K-medoids with the DenseNet201+SVM model are well-trained. Additionally, the effectiveness of the suggested method gave the best results for images after segmentation of the dataset DenseNet201 giving the best accuracy.

The K-medoid clustering method has shown superior performance compared to the traditional K-means and FKM clustering methods for the pre-processing of the dataset. The experimental findings demonstrate that the K-medoid clustering technique can be used as an image segmentation tool for image processing in face masks. Moreover, there is a need to enhance the segmentation efficiency of K-medoids to fulfill the requirements of real-time applications. The next research will prioritize and examine the segmentation efficiency of K-medoids.

This demonstrates that the proposed combination of Densenet201+ SVM architecture is the most effective among those considered and is also suitable for demonstrating real-time capabilities. The real-time limitations are very beneficial for an image and video processing application including these. One additional contribution of this research is to demonstrate the advantages of the proposed

Densenet201+SVM model compared to other comparable models utilized for face mask identification activities.

Data availability

The datasets generated and analyzed during this study are available at <https://www.kaggle.com/datasets/ashishjangra27/facemask-12k-images-dataset>.

CRediT authorship contribution statement

Potharla Ramadevi: Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Formal analysis, Data curation. **Raja Das:** Visualization, Supervision, Conceptualization.

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] Y. Suryawanshi, V. Meshram, V. Meshram, K. Patil, P. Chumchu, Face mask wearing image dataset: a comprehensive benchmark for image-based face mask detection models, *Data Br* 51 (Dec. 2023) 109755, <https://doi.org/10.1016/J.DIB.2023.109755>.
- [2] B. Manohar, R. Das, Comparison of hybrid artificial neural networks with GA, PSO, and RSA in predicting COVID-19 cases: a case study of India, *Multi-Disciplinary Appl. Fog Comput. Responsiveness Real-Time* (Aug. 2023) 207–244, <https://doi.org/10.4018/978-1-6684-4466-5.CHO11>.
- [3] N.H.L. Leung, et al., Respiratory virus shedding in exhaled breath and efficacy of FaceMasks, *Nat. Med.* 26 (5) (May 2020) 676, <https://doi.org/10.1038/S41591-020-0843-2>.
- [4] B. Manohar, R. Das, Artificial neural networks for prediction of COVID-19 in India by using backpropagation, *Expert Syst.* 40 (5) (Jun. 2023) e13105, <https://doi.org/10.1111/EXSY.13105>.
- [5] S. Feng, C. Shen, N. Xia, W. Song, M. Fan, B.J. Cowling, Rational use of face masks in the COVID-19 pandemic, *Lancet Respir. Med.* 8 (5) (May 2020) 434–436, [https://doi.org/10.1016/S2213-2600\(20\)30134-X](https://doi.org/10.1016/S2213-2600(20)30134-X).
- [6] S. Zafeirou, C. Zhang, Z. Zhang, Accepted manuscript A survey on face detection in the wild: past, present and future, *CVIU* (2015) 1–33, <https://doi.org/10.1016/j.cviu.2015.03.015>. A Survey on Face Detection in the wild: past, present and future CVIU A Survey on Face Detection in the wild: past, present and future.
- [7] A. Kumar, A. Kaur, M. Kumar, Face detection techniques: a review, *Artif. Intell. Rev.* 52 (2) (Aug. 2019) 927–948, <https://doi.org/10.1007/S10462-018-9650-2/METRICS>.
- [8] A.H.I. Alrammahi, M.J. Radif, Neural networks in business applications, *J. Phys. Conf. Ser.* 1294 (4) (Oct. 2019), <https://doi.org/10.1088/1742-6596/1294/4/042007>.
- [9] Y. Chen, L. Wu, G. Wang, H. He, G. Weng, H. Chen, An active contour model for image segmentation using morphology and nonlinear Poisson's equation, *Optik* 287 (Sep. 2023) 170997, <https://doi.org/10.1016/J.IJLEO.2023.170997>.
- [10] A. Pratondo, C.K. Chui, S.H. Ong, Integrating machine learning with region-based active contour models in medical image segmentation, *J. Vis. Commun. Image Represent.* 43 (Feb. 2017) 1–9, <https://doi.org/10.1016/J.JVCIR.2016.11.019>.
- [11] H. Li, H. He, Y. Wen, Dynamic particle swarm optimization and K-means clustering algorithm for image segmentation, *Optik* 126 (24) (2015) 4817–4822, <https://doi.org/10.1016/j.ijleo.2015.09.127>.
- [12] J. Ha, M. Kambe, J. Pe, *Data Mining: Concepts and Techniques*, Elsevier, 2011.
- [13] N. Sureja, B. Chawda, A. Vasant, An improved K-medoids clustering approach based on the crow search algorithm, *J. Comput. Math. Data Sci.* 3 (Jun. 2022) 100034, <https://doi.org/10.1016/J.JCMDS.2022.100034>.
- [14] S. Abbasi, H. Abdi, A. Ahmadi, A face mask detection approach based on YOLO applied for a new collected dataset, in: *Int. Comput. Soc. Iran Comput. Conf.*, Mar. 2021, <https://doi.org/10.1109/CSICC52343.2021.9420599>.
- [15] I. Oztel, G. Yolcu Oztel, D. Akgun, A hybrid LBP-DCNN based feature extraction method in YOLO: an application for masked face and social distance detection, *Multimed. Tools Appl.* 82 (1) (Jan. 2023) 1565–1583, <https://doi.org/10.1007/S11042-022-14073-7/FIGURES/10>.
- [16] S.A. Mostafa, et al., A YOLO-based deep learning model for Real-Time face mask detection via drone surveillance in public spaces, *Inf. Sci.* 676 (May) (2024) 120865, <https://doi.org/10.1016/j.ins.2024.120865>.
- [17] S. Bouhsissin, N. Sael, F. Benabbou, Enhanced VGG19 model for accident detection and classification from video, in: *Proc. - 2021 Int. Conf. Digit. Age Technol. Adv. Sustain. Dev. IC DATA 2021*, 2021, pp. 39–46, <https://doi.org/10.1109/ICDATA52997.2021.00017>.
- [18] A. Dash, P.K. Sethy, S.K. Behera, Maize disease identification based on optimized support vector machine using deep feature of DenseNet201, *J. Agric. Food Res.* 14 (October) (2023) 100824, <https://doi.org/10.1016/j.jafr.2023.100824>.
- [19] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, Sep. 2014 [Online]. Available: <https://arxiv.org/abs/1409.1556v6>. (Accessed 24 June 2024).
- [20] S. Guo, S. Chen, Y. Li, Face recognition based on convolutional neural network & support vector machine, in: *2016 IEEE Int. Conf. Inf. Autom. IEEE ICIA 2016*, 2017, pp. 1787–1792, <https://doi.org/10.1109/ICInfA.2016.7832107>, no. August.
- [21] J. Ha, M. Kambe, J. Pe, Data mining: concepts and techniques, *Data Min. Concepts Tech.* (Jan. 2011) 1–703, <https://doi.org/10.1016/C2009-0-61819-5>.
- [22] C.B. Lucasius, A.D. Dane, G. Kateman, On k-medoid clustering of large data sets with the aid of a genetic algorithm: background, feasibility and comparison, *Anal. Chim. Acta* 282 (3) (Oct. 1993) 647–669, [https://doi.org/10.1016/0003-2670\(93\)80130-D](https://doi.org/10.1016/0003-2670(93)80130-D).
- [23] T.F. Yang, X.P. Zhang, Spatial clustering algorithm with obstacles constraints by quantum particle swarm optimization and K-medoids, in: *2010 2nd Int. Conf. Comput. Intell. Nat. Comput. CINC 2010*, vol. 2, 2010, pp. 105–108, <https://doi.org/10.1109/CINC.2010.5643776>.
- [24] E.H. Ruspini, A new approach to clustering, *Inf. Control* 15 (1) (Jul. 1969) 22–32, [https://doi.org/10.1016/S0019-9958\(69\)90591-9](https://doi.org/10.1016/S0019-9958(69)90591-9).
- [25] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A possibilistic fuzzy c-means clustering algorithm, *IEEE Trans. Fuzzy Syst.* 13 (4) (Aug. 2005) 517–530, <https://doi.org/10.1109/TFUZZ.2004.840099>.
- [26] X. Zhao, F. Nie, R. Wang, X. Li, Improving projected fuzzy K-means clustering via robust learning, *Neurocomputing* 491 (2022) 34–43, <https://doi.org/10.1016/j.neucom.2022.03.043>.
- [27] J. Abou Chaaya, B. Zaraket, H. Harb, A. Mansour, Convolutional neural network-based architecture for detecting face mask in crowded areas, in: *IEEE Work. Stat. Signal Process. Proc.*, 2023-July, 2023, pp. 408–412, <https://doi.org/10.1109/SSP53291.2023.10207983>.
- [28] G. Wang, , et al. P. Jo, *Sci. Total Environ.* (2020) 140326, <https://doi.org/10.1016/j.jarray.2023.100294>.
- [29] M.S. Ejaz, M.R. Islam, M. Sifatullah, A. Sarker, Implementation of principal component analysis on masked and non-masked face recognition, in: *1st Int. Conf. Adv. Sci. Eng. Robot. Technol.* 2019, *ICASERT* 2019, May 2019, <https://doi.org/10.1109/ICASERT.2019.8934543>.

- [30] Z. Pan, P. Jin, J. Lei, Y. Zhang, X. Sun, S. Kwong, Fast reference frame selection based on content similarity for low complexity HEVC encoder, *J. Vis. Commun. Image Represent.* 40 (Oct. 2016) 516–524, <https://doi.org/10.1016/J.JVCIR.2016.07.018>.
- [31] M. Umer, et al., Face mask detection using deep convolutional neural network and multi-stage image processing, *Image Vis Comput.* 133 (May 2023) 104657, <https://doi.org/10.1016/J.IAMAVIS.2023.104657>.
- [32] H.T. Ho, R. Chellappa, Pose-invariant face recognition using Markov random fields, *IEEE Trans. Image Process.* 22 (4) (2013) 1573–1584, <https://doi.org/10.1109/TIP.2012.2233489>.
- [33] J. Shijie, J. Peiyi, H. Siping, S. Haibo, Automatic detection of tomato diseases and pests based on leaf images, in: Proc. - 2017 Chinese Autom. Congr. CAC 2017, 2017-January, Dec. 2017, pp. 3507–3510, <https://doi.org/10.1109/CAC.2017.8243388>.
- [34] A.T. Nieuwenhuizen, J. Hemming, H. Suh, *Detection and Classification of Insects on Stick-Traps in a Tomato Crop Using Faster R-CNN*, 2018.
- [35] B.U.H. sheikh, A. Zafar, Beyond accuracy and precision: a robust deep learning framework to enhance the resilience of face mask detection models against adversarial attacks, *Evol. Syst.* 15 (1) (2024) 1–24, <https://doi.org/10.1007/s12530-023-09522-z>.
- [36] B.U. haque Sheikh, A. Zafar, Untargeted white-box adversarial attack to break into deep learning based COVID-19 monitoring face mask detection system, *Multimed. Tools Appl.* 83 (8) (2024) 23873–23899, <https://doi.org/10.1007/s11042-023-15405-x>.
- [37] X. Li, Z. Yang, H. Wu, Face detection based on receptive field enhanced multi-task cascaded convolutional neural networks, *IEEE Access* 8 (2020) 174922–174930, <https://doi.org/10.1109/ACCESS.2020.3023782>.
- [38] A. Biju, A. Edison, Drowsy driver detection using two stage convolutional neural networks, in: 2020 IEEE Recent Adv. Intell. Comput. Syst. RAICS 2020, Dec. 2020, pp. 7–12, <https://doi.org/10.1109/RAICS51191.2020.9332476>.
- [39] W. Mellouk, W. Handouzi, Facial emotion recognition using deep learning: review and insights, *Procedia Comput. Sci.* 175 (Jan. 2020) 689–694, <https://doi.org/10.1016/J.PROCS.2020.07.101>.
- [40] T.M. Saravanan, K. Karthiha, R. Kavinkumar, S. Gokul, J.P. Mishra, A novel machine learning scheme for face mask detection using pretrained convolutional neural network, *Mater. Today Proc.* 58 (Jan. 2022) 150–156, <https://doi.org/10.1016/J.MATPR.2022.01.165>.
- [41] C. Agarwal, P. Itondia, A. Mishra, A novel DCNN-ELM hybrid framework for face mask detection, *Intell. Syst. with Appl.* 17 (Feb. 2023) 200175, <https://doi.org/10.1016/J.ISWA.2022.200175>.
- [42] S. Balasubramanian, R. Cyriac, S. Roshan, K. Maruthamuthu Paramasivam, B. Chellantha Jose, An effective stacked autoencoder based depth separable convolutional neural network model for face mask detection, *Array* 19 (Sep. 2023) 100294, <https://doi.org/10.1016/J.ARRAY.2023.100294>.
- [43] A. Dash, P.K. Sethy, S.K. Behera, Maize disease identification based on optimized support vector machine using deep feature of DenseNet201, *J. Agric. Food Res.* 14 (Dec. 2023) 100824, <https://doi.org/10.1016/J.JAAGR.2023.100824>.
- [44] M.H. Rahman, M.K.A. Jannat, M.S. Islam, G. Grossi, S. Bursic, M. Aktaruzzaman, Real-time face mask position recognition system based on MobileNet model, *Smart Heal.* 28 (Jun. 2023) 100382, <https://doi.org/10.1016/J.SMHL.2023.100382>.
- [45] Face Mask Detection ~12K Images Dataset.” <https://www.kaggle.com/datasets/ashishjangra27/face-mask-12k-images-dataset> (accessed July. 7, 2024).
- [46] M.J. Van der Laan, K.S. Pollard, J. Bryan, A new partitioning around medoids algorithm, *J. Stat. Comput. Simul.* 73 (8) (Aug. 2003) 575–584, <https://doi.org/10.1080/0094965031000136012>.
- [47] kalaivani Anbarasan, “K-Medoid Clustering Algorithm-A Review.” Accessed: February. 8, 2024. [Online]. Available: https://www.academia.edu/8446443/K_Medoid_Clustering_Algorithm_A_Review.
- [48] F. Nie, X. Zhao, R. Wang, X. Li, Z. Li, Fuzzy K-means clustering with discriminative embedding, *IEEE Trans. Knowl. Data Eng.* 34 (3) (2022) 1221–1230, <https://doi.org/10.1109/TKDE.2020.2995748>.
- [49] M. Togacar, B. Ergen, Z. Cömert, F. Özuyurt, A deep feature learning model for pneumonia detection applying a combination of mRMR feature selection and machine learning models, *IRBM* 41 (4) (Aug. 2020) 212–222, <https://doi.org/10.1016/J.IRBM.2019.10.006>.
- [50] D. Cireşan, U. Meier, J. Masci, J. Schmidhuber, Multi-column deep neural network for traffic sign classification, *Neural Network*. 32 (Aug. 2012) 333–338, <https://doi.org/10.1016/J.NEUNET.2012.02.023>.
- [51] Y. Zhang, J. Lian, M. Fan, Y. Zheng, Deep indicator for fine-grained classification of banana's ripening stages, *Eurasip J. Image Video Process* 2018 (1) (Dec. 2018) 1–10, <https://doi.org/10.1186/S13640-018-0284-8/TABLES/3>.
- [52] P. Ramadevi, R. Das, An extensive analysis of machine learning techniques with hyper-parameter tuning by Bayesian optimized SVM kernel for the detection of human lung disease, *IEEE Access* (2024), <https://doi.org/10.1109/ACCESS.2024.3422449>, 1–1.
- [53] M.A. Amou, K. Xia, S. Kamhi, M. Mouhafid, A novel MRI diagnosis method for brain tumor classification based on CNN and Bayesian optimization, *Healthc. (Basel, Switzerland)* 10 (3) (Mar. 2022), <https://doi.org/10.3390/HEALTHCARE10030494>.
- [54] M. Lakshmi, R. Das, B. Manohar, A new COVID-19 classification approach based on Bayesian optimization SVM kernel using chest X-ray datasets, *Evol. Syst.* (Apr. 2024) 1–20, <https://doi.org/10.1007/S12530-024-09575-8/METRICS>.
- [55] W. Xie, W. Nie, P. Safra, L.F. Robledo, P.Y. Descote, W. Jian, Landslide hazard assessment based on Bayesian optimization-support vector machine in Nanping City, China, *Nat. Hazards* 109 (1) (2021) 931–948, <https://doi.org/10.1007/s11069-021-04862-y>.
- [56] A.S. Manek, P.D. Shenoy, M.C. Mohan, K.R. Venugopal, Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier, *World Wide Web* 20 (2) (Mar. 2016) 135–154, <https://doi.org/10.1007/S11280-015-0381-X>.
- [57] M. Lakshmi, R. Das, Classification of monkeypox images using LIME-enabled investigation of deep convolutional neural network, *Diagnostics* 13 (9) (May 2023), <https://doi.org/10.3390/DIAGNOSTICS13091639>.