International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Chaos Game Optimization with Machine Learning Enabled Social Distance Detection and Classification Model

**Dhanya G. Nair[1], Dr. K. P. Sanal Kumar[2*], Dr. S. Anu H. Nair[3]**

**Submitted:** 02/11/2023 **Revised:** 19/12/2023 **Accepted:** 01/01/2024

**Abstract:** Social distancing (SD) links with an action that overcomes the spread of virus, by decreasing the physical contact of human beings such as gatherings at public locations such as parks, universities, schools, airports, malls, hospitals, offices, factories, etc. to avoid crowds and maintain a sufficient distance among people. SD is very vital mainly for people who are in high danger of severe infection from COVID-19. SD detection naturally denotes the usage of technical tools and techniques to observe and apply SD rules. In earlier years, machine learning (ML), deep learning (DL) and computer vision have displayed great outcomes for many daily life issues. In this view, this study presents a Chaos Game Optimization with ML Enabled Social Distance Detection and Classification (CGOML-SDDC) technique. The foremost aim of the CGOML-SDDC technique is to identify social distance between low-risk and high-risk groups. In the developed CGOML-SDDC model, the primary phase of data pre-processing has been performed in two ways: median filtering (MF) based noise elimination and Adaptive Histogram Equalization (AHE) based contrast enhancement. In addition, the CGOML-SDDC technique performs a segmentation process to detect the presence of pedestrians using a Simple Frame differencing-based background subtraction approach. Moreover, the distance among the detected pedestrians can be determined by employing Euclidean Distance. Furthermore, the support vector machine (SVM) model can be employed for the identification of SD. Finally, the CGO algorithm can be employed to adjust the parameters related to the SVM in such a way that the SD detection results can be enhanced. The performance analysis of the CGOML-SDDC methodology is tested by employing an SD dataset. An experimental values highlighted that the CGOML-SDDC technique reaches effective detection results over existing models.

***Keywords:*** *Social Distancing; Median Filtering; Machine Learning; Euclidean Distance; Image Segmentation*

## 1. Introduction

Social distancing (SD) generally links with measures that overcome virus spread by reducing physical connections between people such as crowds at public places like parks, universities, shopping malls, schools, airports, workstations, etc. [1]. This one prevents crowd gatherings as well as maintains a sufficient distance among people. SD is very vital, particularly for people who are suffering from illness of COVID-19 [2]. By reducing the danger of virus spreading from a diseased human to a healthy one, the virus' transmission and illness seriousness will be considerably decreased [3]. If SD is executed at an early stage, it can able to execute an essential role in overwhelming the virus extent and avoiding the pandemic illness's peak. It is highly detected that SD can able to decline the number of diseased patients as well as the load for healthcare groups [4]. In addition, it drops the rates of death by promising that the quantity of diseased patients does not beat public healthcare ability.

Numerous digital tools have been discovered and proposed for enclosing the increase of COVID-19 [5]. This section explains current SD monitoring as well as alerting systems. The current solutions are classified into dual types such as standalone social monitoring systems and wearable SD systems [6]. The former needs to attach a tag to a consumer in order to evaluate the space to nearby persons. In comparison, the latter is dependent on mobile gadgets which are used to observe the SDs among persons in areas depending on image analysis techniques [7]. Primary, the wearable-based system is measured depending on methods such as wireless sensor nodes, smartphone uses (IOS or Android) and smart tags (GPS, RFID, or Bluetooth).

*[1]Research Scholar, Department of Computer and Information Science, Annamalai University, Chidambaram, India.*
*[2]Assistant Professor, P.G Department of Computer Science, R. V. Government Arts College, Chengalpattu, India.*
*[3]Assistant Professor, Department of CSE, Annamalai University, Chidambaram, India (Deputed to WPT Chennai).*
*Email:[1]dgn123dhanya@gmail.com,*
*[2]sanalprabha@yahoo.co.in. [3]anu_jul@yahoo.co.in*

These approaches are essential to a consumer for performing distance sizes and delivering warning notices when any user is situated in a crowded location [8]. Then, a standalone social monitoring system depends on fixed or gadgets of mobile dispersed in an area to extent SD among people and contain digital cameras and robot methods [9]. In previous decades, machine learning (ML), deep learning (DL) and computer vision have proven promising results in daily life problems. The latest development in DL permits the task of object detection more effectual and employs these techniques to scope SD amongst people across the moving surroundings [10]. To define the distancing among people, distance-based as well as clustering models have been used.

This study presents a Chaos Game Optimization with ML Enabled Social Distance Detection and Classification (CGOML-SDDC) technique. In the developed CGOML-SDDC model, the primary phase of data pre-processing is performed in two ways: median filtering (MF) based noise elimination and Adaptive Histogram Equalization (AHE) based contrast enhancement. In addition, the CGOML-SDDC technique performs a segmentation process to detect the presence of pedestrians using a Simple Frame differencing-based background subtraction approach. Furthermore, the support vector machine (SVM) technique can be used for SD recognition. Finally, the CGO algorithm can be employed to adjust the parameters related to the SVM in such a way that the SD detection results can be enhanced. The performance analysis of the CGOML-SDDC model is tested employing an SD dataset.

## 2. Related works

In [11], the foremost aim of this paper is to construct an ML technique depending on TensorFlow API and YOLO Objection Detection defines red and green rectangles near the face. If an individual is identified in the camera the person wears a mask or not and sends an email alarm to the specialists who are on duty notifying them regarding an individual's damage to face cover plan and returns a green or red limiting box consequently if SD has been preserved among two people and simultaneously alert others by a beep sound. In [12], a hybrid Computer Vision and YOLOv4-based Deep Neural Networks (DNNs) method has been presented. The developed DNN technique in integration with an altered inverse perspective mapping (IPM) model and SORT tracking procedure which mains to a strong human recognition and SD observing. This technique has been trained against dual datasets of Google Open Image and MS COCO.

In [13], an SD Detector System and COVID-19 Face Mask have been developed, which is a one-phase detector and contains an artificial neural network (ANN) in order to fuse higher-level semantic data by many feature maps, and then ML method concentrates on identifying face Mask and SDs at the same time. Additionally, the system will practice current IP and CCTV cameras united with computer vision. In [14], a technique that depends on wireless localization and space among wireless consumers employing ML as well as correlation distance methods expansion has been presented. The method is composed of indoor RSSI data of WIFI stations utilizing smartphones, executing feature selection and data pre-processing RSSI, detection models are used to define place, and lastly employ correlation amongst extents to evaluate distance by employing a developed method.

Javed et al. [15] project a high-scale dataset and a channel to execute real-time face mask recognition and SD size in an outside place. Originally, current modern single and multi-stage object recognition systems are perfected on developed datasets for estimating their performance in accuracy as well as suggestion time. Besides, the YOLO-v3 framework is enhanced by fine-tuning its feature extraction as well as area proposal generation layers. In [16], an active surveillance system has been developed. The involvement is double. Primary, a vision-based real-time method is presented that can identify SD destructions as well as send non-intrusive audio-visual signals by utilizing DL techniques. Next, the model defines a new serious social density value as well as displays that the accidental SD violation incidence can be detained close to zero if an ordinary mass is preserved below this value.

## 3. The Proposed Model

In this research, an automated SD model named CGOML-SDDC model has been proposed for the recognition and classification of SD. The CGOML-SDDC technique comprises three major procedures namely pre-processing, segmentation and classification. At the pre-processing stage, the quality of the input images can be enhanced by the use of MF based noise removal and AHE based contrast improvement. Besides, the segmentation process takes place where the Simple Frame Differencing based

background subtracts and Euclidean distance-based pedestrian distance calculation are involved. Finally, the CGO with the SVM model can be exploited for accurate detection and classification of the SD. Fig. 1 depicts the entire flow of the CGOML-SDDC algorithm.

### 3.1. Image Pre-processing

In the initial stage, the MF based pre-processing and AHE based contrast enhancement process is performed to boost the quality of the input images.

#### 3.1.1. MF based Pre-processing

To eliminate the sound in input images, MF based pre-processing is utilized. MF is a non-linear filter which totals the medium of pixel set that drops in the filter mask [17]. Every pixel is found as well as exchanged by an arithmetical median of its $NxM$ neighborhood. Then, the median value is intended from neighborhod pixel, it is stronger to outliers and does not generate a novel accurate pixel value that conserves edge blurring and loss of image part. It preserves the sharp frequency facts. It is suitable to remove salt and pepper sound from an image. The result of MF in eliminating sound is enlarged as the window size rises. The formulation of MF is as follows:

$$\hat{f}(x,y) = \tag{1}$$

Where $S_{xy}$ is the coordinates of subimage window of size $NxM$.

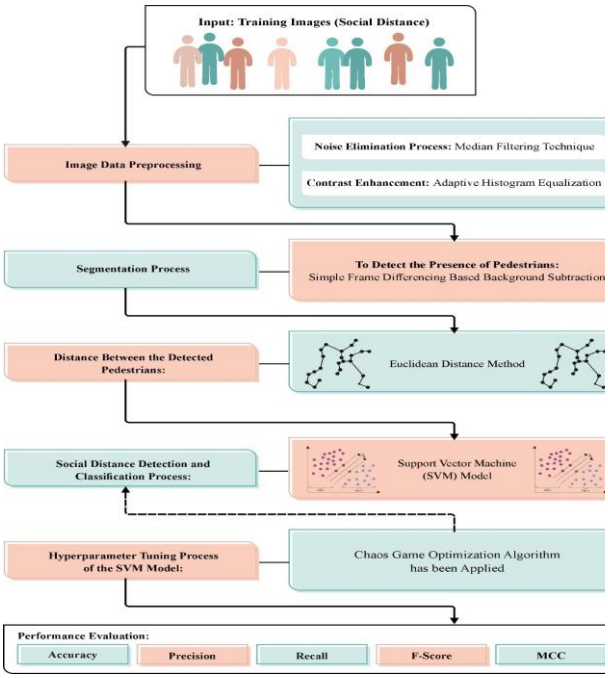


**Fig. 1.** Overall flow of CGOML-SDDC model

#### 3.1.2. AHE-based Contrast Enhancement

To enhance the contrast level in the images, the AHE approach is applied. The CLAHE technique sub-divides the image into many tiles [18]. We evaluate the tile size $s$ by employing the total entropy of an input frame. For a greyscale image with 256 grey levels, an entropy $E$ is definite below.

$$E = -\sum_{k=0}^{255} p_k log_2(p_k) \tag{2}$$

where $p_k$ is the probability related to gray level $k$. We employ an exponential decay function of the type

$$s = (M - E_{max})e^{-\lambda E} + E_{max} \tag{3}$$

To calculate the tile extent $s$ depends on entropy value $E$ for an assumed image of size $M$. The extreme entropy $E_{max}$ for gray-level images is 8. For an input image, $\lambda = 0.7$ and size $M = 256$ pixels displays an instance of tile size difference with entropy. When an entropy is at maximum, the function offers a tile size of $8 \times 8$ pixels. For a very tiny entropy value of 1 equivalent to a closely even intensity distribution, we get a tile size that is half the size of an input image.

### 3.2. Image Segmentation

Once input images are pre-processed, the next phase is to perform segmentation to detect the occurrence of pedestrians and estimate the space among them by employing Euclidean distance.

#### 3.2.1. Simple Frame Differencing based Background Subtraction Method

The application of simple frame differencing for background removal verifies to be a straightforward yet effective method in pedestrian recognition states. By equating consecutive frames of a video feed, variations in pixel values are distinguished, emphasizing the moving units against a static context. This foreground mask is then developed via morphological actions, resulting in a clearer picture of possible pedestrian areas. The use of contour detection, attached with filtering depending on norms such as contour area, enables the identification of pedestrians. Bounding boxes drawn around these areas then aid as a visual sign of identified individuals. This simple frame differencing method lays the base for pedestrian recognition applications, providing a main tool for differentiating dynamic objects in video streams.

### 3.2.2. Distance Calculation Using Euclidean Distance

To calculate the space among detected pedestrians, the Euclidean distance (ED) based distance calculation is applied. ED is widely applied in pedestrian detection for calculating the spatial distance between various points, viz., a point in person or between a reference point and the centroid of a detected pedestrian [19]. The ED between two points (x1,y1) and (x2,y2) in 2D space is represented as follows:

$$Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (4)$$

Keypoint Matching: Keypoints including head, shoulders, hips, and so on, are identified on the person. ED matches the corresponding keypoints between successive frames, which helps to track the pedestrian movement.

Bounding Box Analysis: the ED between the centroid of the bounding box is used to evaluate spatial distance travelled by the pedestrian once the bounding box of the pedestrian is detected in successive frames.

Region of Interest (ROI) Filtering: once the area of interest or reference point exists, ED determines the distance between a reference point and the detected pedestrian. This is helpful in scenarios where the action or alert is triggered after the pedestrian leaves or enters a certain area.

Velocity Calculation: By applying the ED between the pedestrian positions in consecutive frames and knowing the time variance between frames, the velocity of a pedestrian is defined.

### 3.3. SVM based Classification

For accurate detection of SD, the SVM model is used. As a result of binary classification problems, SVM is established as a linear classification algorithm to create dual feature space with extreme interval [20]. By establishing a divided hyperplane, the interval between the training dataset is maximized using optimization theory.

The training data is considered as $(t_i, f_i)$, whereas $= 1,2 \cdots, N$. The input dataset and learning objective in detection problems are represented as $T = \{T_1, \ldots, T_N\}$ and $f = \{f_1, \ldots, f_N\}$, correspondingly. The input data creates the feature space based on $t_i \in R^n$. The updated objective is bi-category objectives: $f \in \{-1,1\}$. An updated optimization objective is separated into twofold classes and then the hyperplane is considered as a decision boundary in the feature space where an input dataset is positioned. The space from the geometric position of any data point to the hyperplane is equal to or larger than 1. The decision border can be described as follows:

$$(t \cdot \omega + b = 0). \quad (5)$$

The separation line accurately categorized the data sample as two types and fulfilled the separation interval.

$$F_i[(t_i \cdot \omega) + b] \geq 1, i = 1,2, \cdots, N. \quad (6)$$

Now, $2/\|\omega\|^2$ is the classification interval margin. The optimum hyperplane is created by the transformation into the least issue with constraint. Firstly, consider hyperplane has a parameter viz., the geometric interval among data set and plane. To control the geometric interval, the constraint function is utilized. Once the geometric interval is small, then the hyperplane is superior. By finding the minimal value of $\omega$ and $b$ parameters, the issue of finding the optimum plane becomes the main issue of addressing its restraint optimization problems for producing the minimal value of $\|\omega\|$variable. After training the data sample, the last classification hyperplane is defined by the sample data point at the surface limit, viz., and the training sample point on $H_1$ and $H_2$ denotes the support vectors.

The drawback of hard-margin SVM in resolving non-separable linear difficulties is that it can effortlessly generate classifier faults. Depending on margin maximization in the proposal of a new optimization technique, we introduced a loss function. To reduce the classification errors or misclassification and find the maximal interval, an objective function needs to be altered by adding $C$ penalty factor for balancing the value of $\xi$, which controls the optimization tendency and by introducing a slack parameter, $\xi$, for minimizing the constraints:

$$\frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^{N} \xi_i \quad (7)$$

$$s.t. f_i(t_i \cdot \omega + b) \geq 1 - \xi_i. \quad (8)$$

A non-linear SVM is attained by mapping an original input dataset into higher-dimension feature space by applying a non-linear function in linear SVM. The non-linear SVM has an optimization problem. According to Karush-KuhnTucker (KKT) theory, Eqs.

(7) and (8) become a dyadic expression by introducing the Lagrange function:

$$\sum_{i}^{N} \alpha_i - \frac{1}{2}\sum_{i}^{N} \sum_{j}^{N} \alpha_i \alpha_j f_i f_j K(t_i, t_j) \quad (9)$$

$$s.t. \sum_{i}^{N} \alpha_i f_i = 0, \alpha_i \in [0, C]. \quad (10)$$

In Eqs. (9) & (10), $\alpha$ represents the KKT multiplier that the multiplier of Lagrange used to execute an inequality constraint. Gaussian radial basis kernel is widely applied, hence it was selected as a function of the kernel. According to Eq. (11), the $\sigma$ unique parameter should be set up:

$$K(t_i, f_i) = exp\left(-\frac{\|t_i - f_i\|^2}{2\sigma^2}\right). \quad (11)$$

Thus, the objective function is formulated as:

$$H(t) = \sum_{i}^{N} \alpha_i f_i K(t_i, f_i) + b. \quad (12)$$

### 3.4. CGO based Parameter Tuning

Finally, the CGO algorithm adjusts the parameters related to the SVM model. CGO algorithm is a newly developed metaheuristic optimization approach that integrates the mathematical model of fractals and the game theory [21]. The fractal creation is inspired by the polygon shape that begins with an affine function and a random initial point. Next, the fractal shape is generated by the repetitive series of points and the function selected is applied repeatedly to a new point. The CGO technique objective to produce a Sierpinski Triangle based on the fractal aspects of the chaos concept. The initial population begins with many candidate solutions ($X$) which comprise on decision variable ($x_{i,j}$) represents a suitable point within the search range:

$$x_i^j(0) = x_{i,min}^j + rand.(x_{i,max}^j - x_{i,min}^j), \{i = 1,2,\dots,n\ j = 1,2,\dots,d \quad (13)$$

Now the number of suitable solution candidates represents $n$, and dimension of the solution refers to $d$. The initial position of the solution indicates $x_i^j(0)$. $x_{i,max}^j$, $x_{i,min}^j$ are the maximal and the minimal values in $j^{th}$ decision parameter of $j^{th}$ solution and $rand$ shows the random integer within [0,1]. Based on three vertices, a temporary Sierpinski triangle is generated within the search range for the suitable solution for the initial search:

- The location of Global Best($GB$).
- The location of the Mean Group($MG_i$).
- The location of $i^{th}$ solution candidates($X_i$).

The location updating process can be performed. The first one simulates the movement of solution $X_i$ towards $GB$ and $MG_i$ modelled according to the following equation:

$$Seed_i^1 = X_i + \alpha_i \times (\beta_i \times GB - \gamma_i \times MG_i), i = 1,2,\dots,n \quad (14)$$

In Eq. (2), $\alpha_i$, $\beta_i$ and $\gamma_i$ are random integers between zero and one.

The $GB$ movement towards $X_i$ and $MG_i$ is arithmetically expressed as below:

$$Seed_i^2 = GB + \alpha_i \times (\beta_i \times X_i - \gamma_i \times MG_i), i = 1,2,\dots,n \quad (15)$$

The tertiary approach illustrates the movement of $MG_i$ towards $X_i$ and$GB$:

$$Seed_i^3 = MG_i + \alpha_i \times (\beta_i \times X_i - \gamma_i \times GB), i = 1,2,\dots,n \quad (16)$$

The updating location process can be modelled by the initial three approaches that describe an exploitation stage of the CGO optimization technique. In the meantime, the fourth approach depicts the mutation for an exploration stage as follows:

$$Seed_i^4 = X_i(x_i^k = x_i^k + R), k = [1,2,\dots,d] \quad (17)$$

In Eq. (17), $R$ is a uniform distribution random integer within [0,1] and $k$ shows the stochastic value ranges [1,$d$]. The four dissimilar formulations are taken into account for $\alpha_i$ to tune the diversification and intensification rates of the CGO.

$$\alpha_i = \{Rand\ 2 \times Rand\ (\delta \times Rand) + 1\ '(\varepsilon \times Rand) + (\sim \varepsilon) \quad (18)$$

In Eq. (18), $Rand$ is a standard distributed random value[0,1], as for $\delta$ and $\varepsilon$ parameters, they signify random integer within [0,1]. Fig. 2 demonstrates the steps involved in the CGO algorithm.

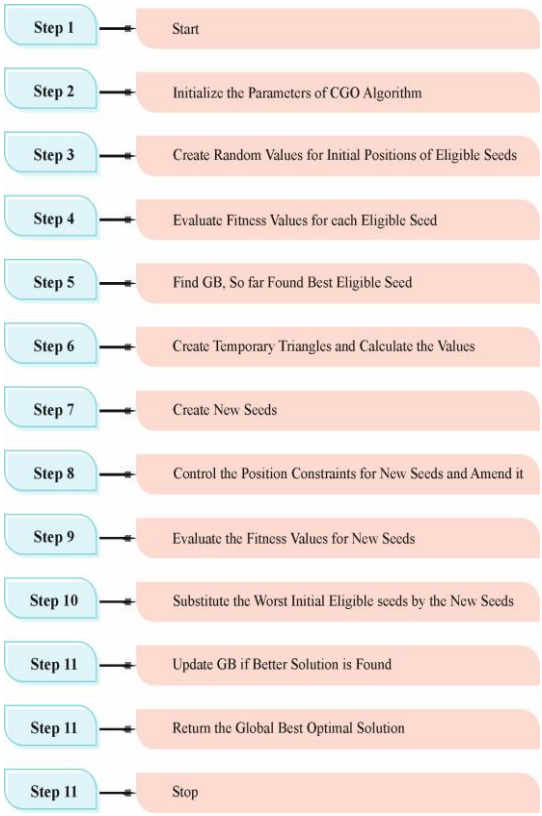


**Fig. 2.** Steps involved in CGO

The CGO model originates a fitness function (FF) to reach superior classification performance. It defines an optimistic integer to signify the better performance of candidate solutions. In this research, the classification error rate minimization has been measured as FF which is provided in Eq. (19).

$$fitness(x_i) = ClassifierErrorRate(x_i)$$

$$= \frac{number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \qquad (19)$$

## 4. Results and Discussion

The SD detection performance of the CGOML-SDDC model can be tested using the SD dataset [22] comprising 1000 samples with 2 classes as represented in Table 1. Fig. 3 illustrates the sample images.

**Table 1** Details of dataset

| **Class** | **No. of Instances** |
|---|---|
| High_Risk | 500 |
| Low_Risk | 500 |
| **Total Instances** | **1000** |

**Fig. 3.** Sample Images

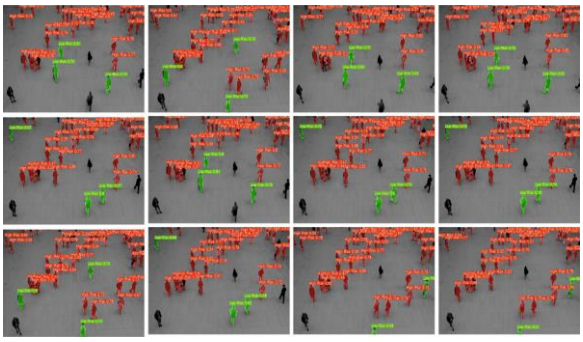

**Fig. 4.** Sample visualization of SD detected images

Fig. 4 shows the sample classification output obtained by the CGOML-SDDC model on the input images. The results indicate that the CGOML-SDDC model properly recognized the high risk and low risk classes.

Fig. 5 displays the classifier outcomes of the CGOML-SDDC method below the test dataset. Figs. 5a-5b represents the confusion matrix acquired by the CGOML-SDDC technique at 70:30 of the training phase (TRPH)/testing phase (TSPH). The figure exhibits that the CGOML-SDDC method can be precisely identified and categorized with high_risk and low_risk classes. Then, Fig. 5c reveals the PR analysis of the CGOML-SDDC methodology. The figure stated that the CGOML-SDDC technique gets higher PR effectiveness in two classes. Besides, Fig. 5d displays the ROC analysis of the CGOML-SDDC system. The figure revealed that the CGOML-SDDC algorithm provides efficient results with increased ROC values with diverse class labels.

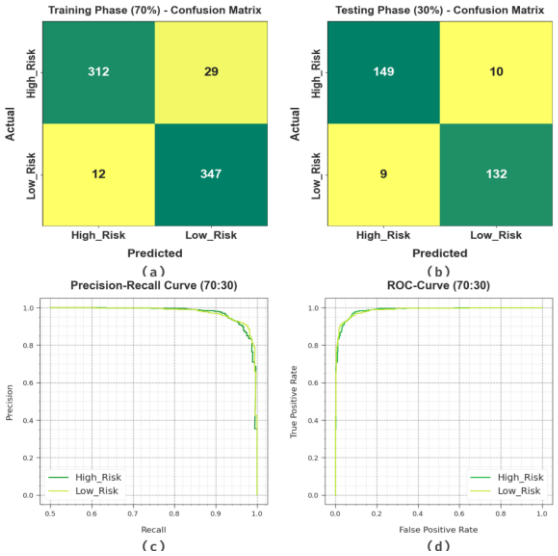


**Fig. 5.** (a-b) Confusion matrices of 70:30 of TRPH/TSPH (c) PR-curve and (d) ROC-curve

The SD detection results of the CGOML-SDDC method on 70:30 of TRPH/TSPH are given in Table 2 and Fig. 6. The accomplished result shows that the CGOML-SDDC technique categorizes the high_risk and low_risk classes. With 70% of TRPH, the CGOML-SDDC technique provides effectual performance with maximum $accu_y$ of 94.08%, $prec_n$ of 94.29%, $reca_l$ of 94.08%, $F_{score}$ of 94.13, and MCC of 88.37%. Additionally, with 30% of TSPH, the CGOML-SDDC technique gives efficient performance with maximal $accu_y$ of 93.66%, $prec_n$ of 93.63%, $reca_l$ of 93.66%, $F_{score}$ of 93.65, and MCC of 87.29% respectively.

**Table 2** SD detection outcome of CGOML-SDDC model under 70:30 of TRPH/TSPH

| **Classes** | $Accu_y$ | $Prec_n$ | $Reca_l$ | $F_{score}$ | **MCC** |
|---|---|---|---|---|---|
| **TRPH (70%)** | | | | | |
| High_Risk | 91.50 | 96.30 | 91.50 | 93.83 | 88.37 |
| Low_Risk | 96.66 | 92.29 | 96.66 | 94.42 | 88.37 |
| **Average** | **94.08** | **94.29** | **94.08** | **94.13** | **88.37** |
| **TSPH (30%)** | | | | | |
| High_Risk | 93.71 | 94.30 | 93.71 | 94.01 | 87.29 |
| Low_Risk | 93.62 | 92.96 | 93.62 | 93.29 | 87.29 |
| **Average** | **93.66** | **93.63** | **93.66** | **93.65** | **87.29** |

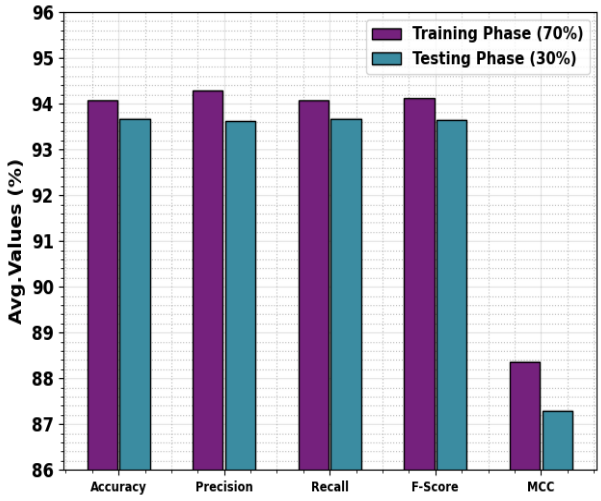


**Fig. 6.** Average outcomes of CGOML-SDDC algorithm on 70:30 of TRPH/TSPH

Fig. 7 represents the classifier analysis of the CGOML-SDDC system on the test dataset. Figs. 7a-7b displays the confusion matrices created by the CGOML-SDDC technique under 80:20 of TRPH/TSPH. The figure signified that the CGOML-SDDC system can be appropriately identified and categorized as high_risk and low_risk classes. Meanwhile, Fig. 7c exhibits the PR analysis of the CGOML-SDDC algorithm. The figure stated that the CGOML-SDDC model achieves improved PR performance with each class. Also, Fig. 7d exemplifies the ROC curve of the CGOML-SDDC methodology. The figure exemplified that the CGOML-SDDC technique leads to successful outcomes with superior ROC values with different class labels.

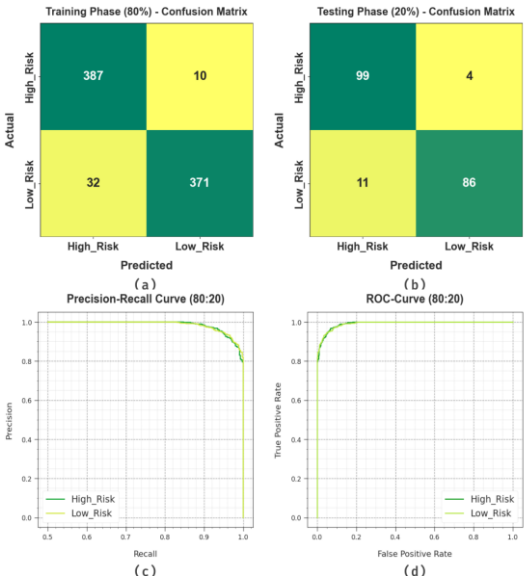


**Fig. 7.** (a-b) Confusion matrices on 80:20 of TRPH/TSPH (c) PR-curve and (d) ROC-curve

The SD detection analysis of the CGOML-SDDC system on 80:20 of TRPH/TSPH can be described in Table 3 and Fig. 8. The attained outcomes display the CGOML-SDDC system categorizes as high_risk and low_risk classes. According to 80% of TRPH, the CGOML-SDDC algorithm offers proficient performance with great $accu_y$ of 94.77%, $prec_n$ of 94.87%, $reca_l$ of 94.77%, $F_{score}$ of 94.75, and MCC of 89.64%. Besides, with 20% of TSPH, the CGOML-SDDC methodology gives effective performance with maximum $accu_y$ of 92.39%, $prec_n$ of 92.78%, $reca_l$ of 92.39%, $F_{score}$ of 92.47, and MCC of 85.17% correspondingly.

**Table 3** SD detection analysis of CGOML-SDDC model on 80:20 of TRPH/TSPH

| **Classes** | $Accu_y$ | $Prec_n$ | $Reca_l$ | $F_{score}$ | **MCC** |
|---|---|---|---|---|---|
| **TRPH (80%)** | | | | | |
| High_Risk | 97.48 | 92.36 | 97.48 | 94.85 | 89.64 |
| Low_Risk | 92.06 | 97.38 | 92.06 | 94.64 | 89.64 |
| **Average** | **94.77** | **94.87** | **94.77** | **94.75** | **89.64** |
| **TSPH (20%)** | | | | | |
| High_Risk | 96.12 | 90.00 | 96.12 | 92.96 | 85.17 |
| Low_Risk | 88.66 | 95.56 | 88.66 | 91.98 | 85.17 |
| **Average** | **92.39** | **92.78** | **92.39** | **92.47** | **85.17** |

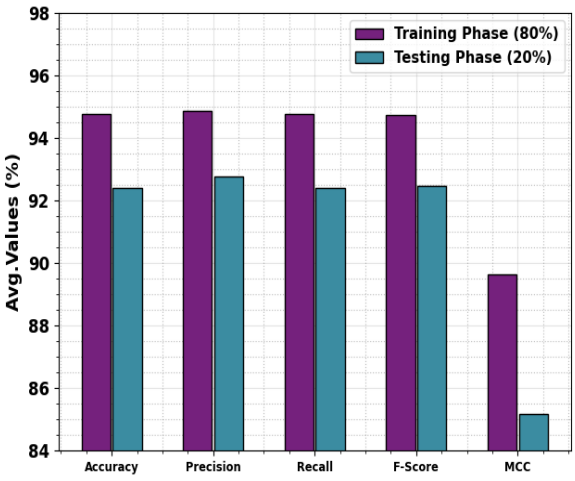


**Fig. 8.** Average outcomes of CGOML-SDDC method on 80:20 of TRPH/TSPH

A detailed comparison analysis of the CGOML-SDDC technique with other approaches is given in Table 4 and Fig. 9 [23]. The achieved values indicate the ineffectual performance of the CNN and ResNet-50 models. Although the MobileNetv3, Inception v3, RF, LR, and Stacked ResNet-50 models obtain considerable performance, the CGOML-SDDC technique shows promising performance with $accu_y$ of 94.77%, $prec_n$ of 94.87%, $reca_l$ of 94.77%, and $F1_{score}$ of 94.75%.

**Table 4** Comparison analysis of the CGOML-SDDC model with other algorithms

| **Model** | $Accu_y$ | $Prec_n$ | $Reca_l$ | $F_{score}$ |
|---|---|---|---|---|
| CNN Model | 75.00 | 51.00 | 65.00 | 52.00 |
| MobileNet V3 | 83.00 | 56.00 | 93.00 | 59.00 |
| Inception V3 | 83.00 | 66.00 | 93.00 | 65.00 |
| ResNet-50 Model | 77.00 | 54.00 | 67.00 | 59.00 |
| Random Forest Model | 82.30 | 64.40 | 90.27 | 71.23 |
| Logistic Regression Model | 82.98 | 69.12 | 90.54 | 74.57 |
| Stacked ResNet-50 | 87.00 | 71.00 | 92.00 | 79.00 |
| CGOML-SDDC | 94.77 | 94.87 | 94.77 | 94.75 |

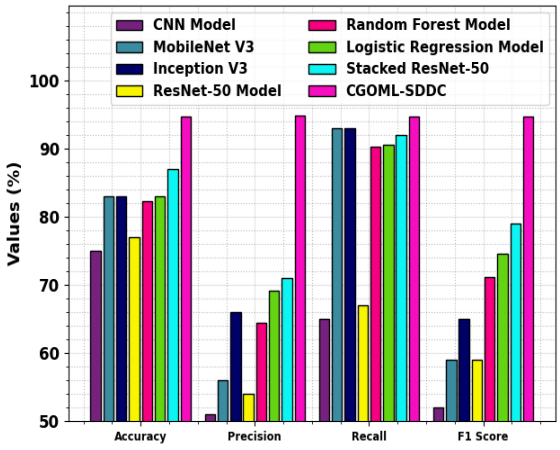


**Fig. 9.** Comparison analysis of CGOML-SDDC system with other models

In Table 5 and Fig. 10, the computation time (CT) results of the CGOML-SDDC technique with existing approaches are provided. The results imply that the CNN, RF, LR, and stacked ResNet-50 models have shown worse performance. Along with that, the MobileNetv3, Inception v3, and ResNet-50 models have exhibited closer results. However, the CGOML-

SDDC technique gains maximum performance with minimal CT of 2.41s. Thus, the CGOML-SDDC technique can be applied for an accurate SD detection process.

**Table 5** CT analysis of the CGOML-SDDC system with other methods

| Model | Computational Time (sec) |
|---|---|
| CNN Model | 5.52 |
| MobileNet V3 | 3.86 |
| Inception V3 | 3.52 |
| ResNet-50 Model | 3.30 |
| Random Forest Model | 4.77 |
| Logistic Regression Model | 4.29 |
| Stacked ResNet-50 | 4.13 |
| CGOML-SDDC | 2.41 |

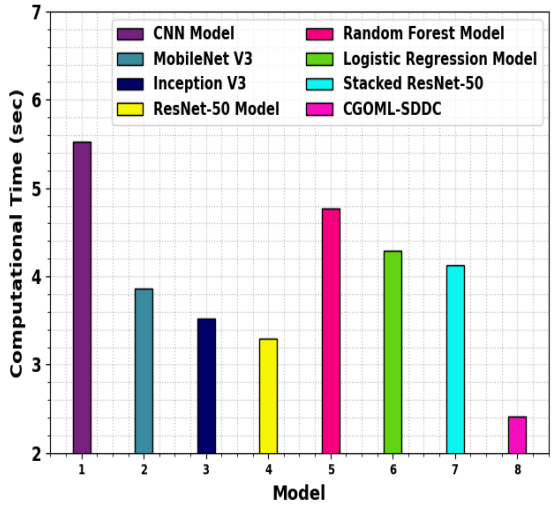


**Fig. 10.** CT analysis of the CGOML-SDDC algorithm with other models

## 5. Conclusion

In this study, an automated SD model named CGOML-SDDC methodology has been proposed for the classification and detection of SD. The CGOML-SDDC technique comprises three main methods namely pre-processing, segmentation and classification. At the time of pre-processing stage, the quality of the input images can be enhanced by the use of MF based noise removal and AHE based contrast improvement. Besides, the segmentation process takes place where the Simple Frame Differencing based background subtract and Euclidean distance-based pedestrian distance calculation are involved. Finally, the CGO with the SVM model can be exploited for accurate classification and recognition of the SD. The performance analysis of the CGOML-SDDC methodology is experimented with utilizing an SD dataset. An experimental value highlighted that the CGOML-SDDC technique reaches effective detection results over existing models.

## References


[1] Pooranam, N., Priya, S.P.N., Sruthi, S., Dhanya, S.K.: A safety measuring tool to maintain social distancing on COVID-19 using deep learning approach. ICCCEBS J. Phys. Conf. Series 1916(012122), 1–8 (2021)

[2] Ahmed, I., Ahmad, M., Jeon, G.: Social distance monitoring framework using deep learning architecture to control infection transmission of COVID-19 pandemic. Sustain. Cities Soc. 69(102777), 1–11 (2021)

[3] Tuli, S., Tuli, S., Tuli, R., Gill, S.S.: Predicting the growth and trend of covid-19 pandemic using machine learning and cloud computing. Internet Things 11, 100222 (2020)

[4] Sathyamoorthy, A. J., Patel, U., Savle, Y. A., Paul, M., & Manocha, D. (2020). COVIDrobot: Monitoring social distancing constraints in crowded scenarios. arXiv:2008.06585.

[5] Ahmed, I., Ahmad, M., Rodrigues, J.J.P.C., et al.: A deep learning-based social distance monitoring framework for COVID-19. Sustain. Cities Soc. 65(102571), 1–12 (2021)

[6] Punn, N. S., Sonbhadra, S. K., & Agarwal, S. (2020). Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques. arXiv:2005.01385.

[7] Payedimarri, A.B., Concina, D., Portinale, L., Canonico, M., et al.: Prediction models for public health containment measures on COVID-19 using artifcial intelligence and machine learning: a systematic review. Int. J. Environ. Res. Public Health 18(4499), 1–11 (2021)

[8] Pouw, C. A., Toschi, F., van Schadewijk, F., & Corbetta, A. (2020). Monitoring physical distancing for crowd management: Real-time trajectory and group analysis. arXiv: 2007.06962.

[9] Nguyen, C. T., Saputra, Y. M., Van Huynh, N., Nguyen, N.-T., Khoa, T. V., Tuan, B. M., et al. (2020). Enabling and emerging technologies for social distancing: a comprehensive survey

and open problems. arXiv:2005.02816.

[10] Rahim, A., Maqbool, A., Rana, T.: Monitoring social distancing under various low light conditions with deep learning and a single motionless time of fight camera. PLoS ONE 16(2), e0247440 (2021)

[11] Choudhury, M., Paul, D., Acharya, A., Banerjee, N. and Datta, D., 2023. Face Mask and Social Distancing Detection in Real Time. In *Perspectives on Social Welfare Applications' Optimization and Enhanced Computer Applications* (pp. 157-180). IGI Global.

[12] Rezaei, M. and Azarmi, M., 2020. Deepsocial: Social distancing monitoring and infection risk assessment in covid-19 pandemic. *Applied Sciences*, *10*(21), p.7514.

[13] Limbasiya, B. and Raut, C., 2021. COVID-19 face mask and social distancing detector using machine learning. *International Research Journal of Engineering and Technology (IRJET)*, *8*(05), pp.2056-2061.

[14] Brahmi, R., Boujnah, N., Abdennour, G.B. and Ejbali, R., 2022, May. Social Distancing elaboration for indoor environment using machine learning techniques. In *2022 International Wireless Communications and Mobile Computing (IWCMC)* (pp. 1022-1027). IEEE.

[15] Javed, I., Butt, M.A., Khalid, S., Shehryar, T., Amin, R., Syed, A.M. and Sadiq, M., 2023. Face mask detection and social distance monitoring system for COVID-19 pandemic. *Multimedia Tools and Applications*, *82*(9), pp.14135-14152.

[16] Yang, D., Yurtsever, E., Renganathan, V., Redmill, K.A. and Özgüner, Ü., 2021. A vision-based social distancing and critical density detection system for COVID-19. *Sensors*, *21*(13), p.4608.

[17] Sukassini, M. and Velmurugan, T., 2016. Noise removal using morphology and median filter methods in mammogram images. In *The 3rd International Conference on Small and Medium Business* (pp. 413-419).

[18] Singh, P., Mukundan, R. and De Ryke, R., 2020. Feature enhancement in medical ultrasound videos using contrast-limited adaptive histogram equalization. *Journal of digital imaging*, *33*, pp.273-285.

[19] Vadivu, G.E. and Muthusamy, T., 2023. Synthesis of deep learning technique for social distance monitoring in pandemic areas. *Multimedia Tools and Applications*, pp.1-16.

[20] Li, S., Chen, H., Chen, Y., Xiong, Y. and Song, Z., 2023. Hybrid Method with Parallel-Factor Theory, a Support Vector Machine, and Particle Filter Optimization for Intelligent Machinery Failure Identification. *Machines*, *11*(8), p.837.

[21] Ouertani, M.W., Manita, G. and Korbaa, O., 2022. Automatic Data Clustering Using Hybrid Chaos Game Optimization with Particle Swarm Optimization Algorithm. *Procedia Computer Science*, *207*, pp.2677-2687.

[22] https://github.com/ChargedMonk/Social-Distancing-using-YOLOv5

[23] Walia, I.S., Kumar, D., Sharma, K., Hemanth, J.D. and Popescu, D.E., 2021. An integrated approach for monitoring social distancing and face mask detection using stacked Resnet-50 and YOLOv5. *Electronics*, *10*(23), p.2996.