International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.6646
***Article Received**: 27 February 2023 **Revised**: 08 April 2023 **Accepted**: 23 April 2023*

PSO Optimized CNN-SVM Architecture for Covid - 19 Classification

P.V. Naresh[1], R. Visalakshi[2], B.Satyanarayana[3]
[1]Research Scholar, Annamalai University, Department of Computer Science and Engineering, Tamilnadu, India
Email id : naresh.groups@gmail.com
[2]Assistant Professor. Annamalai University, Department of Information Technology, Tamilnadu, India
Email id : visalakshi_au@yahoo.in
[3]Principal, CMR Institute of Technology, Telangana, India
Email id : bsat777@gmail.com
*Corresponding Author ORCID ID : 0000-0003-1067-1038

Abstract— This paper presents a hybrid model that utilizes PSO particle swarm optimization, Convolution Neural Networks (CNN) and (SVM) Support Vector Machine architecture for recognition of Covid19.The planned model extracts optimized structures with particle swarm optimization then passes to Convolution Neural Network for automatic feature extraction, while the SVM serves as a Multi classifier. The dataset comprises Covid 19, Pneumonia and Normal Chest X-Ray pictures used to hone and evaluate the suggested algorithm. The most distinct traits are automatically extracted by the algorithm from these photographs. Experimental results show that the suggested framework is effective, with an average recognition accuracy of 97.42%.The most successful SVM Kernel was RBF.

Keywords- CNN, SVM, PSO, deep learning, Covid, Chest X-Ray)

I. INTRODUCTION

The primary purpose of this investigation is to optimize the features of images using particle swarm optimization and passing these features to CNN for automatic extraction of features and the SVM classifier is regarded known to be among machine learning's most reliable classifiers. This study employs the PSO CNN algorithm in combination with the SVM classifier to classify the disease. Most conventional approaches use different methods for feature extraction and classification, making them unsuitable for real-time applications due to high computational time. The CNN algorithm is dissimilar from traditional machine learning algorithms because it does not require a separate feature extraction operation.

At current, deep learning techniques like Convolution Neural Networks (CNNs) are fast and reliable for image recognition and classification[1-2].Yet, some photographs with a little more noise in them may cause a neural network to classify them wrongly. This is due to CNN's extensive usage of enhanced training data, which enables the use of changed input images. We tried to address this problem by implement our model using PSO – CNN-SVM model. The remainder of the essay is explained as follows. Section 2 details the suggested approach. Experimental design and findings are covered in Section 3, performance comparison is covered in Section 4, and the conclusion is covered in Section 5.

II. PROPOSED METHODOLOGY

The proposed planning of PSO-CNN-SVM Model is shown in **Fig 3;** it consists of particle swarm optimization technique which optimizes the features with base classifier as random forest. Then the optimized features are passed and given input to the convolution neural network (CNN) for automatic feature extraction. In the approach 3x3 the most recognizable features are extracted from the raw input photos using a kernel or filter. An nxn input neuron from the input layer is convolved with a mxm filter in the convolution layer, producing an output with the dimensions (n-m+1) x (n-m+1). Each layer's output serves as the following layer's input. Effective sub-regions are calculated from the image using CNN's receptive field feature. The extracted features are then approved for classification to SVM.

Support Vector Machine (SVM) is highly efficient in minimizing generalization error on unseen data; however, it may not perform well on noisy data. Moreover, SVM finds it challenging to learn complex features because of its shallow construction. In this paper, a hybrid CNN-SVM model is presented to address these constraints. The better classification accuracy may be due to the SVM classifier's mapping of the input features to a higher dimensional space. Here, SVM is employed as a Multi class classification and replaces the softmax layer of CNN

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.6646
Article Received: 27 February 2023 Revised: 08 April 2023 Accepted: 23 April 2023

II.I DATASET COLLECTION

The dataset used in this work is an open source compiled and taken from various sources like kaggle and github. The images are divided into Covid, Normal, Pneumonia and the splits of the dataset are given in the table 1. The dataset is further categorized into training and testing with 80% for training and 20% for testing purpose.

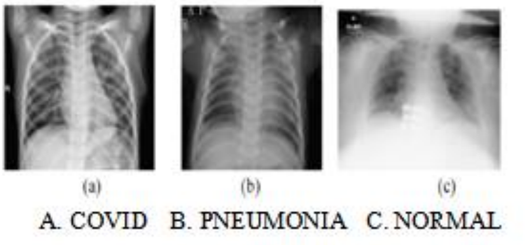

Fig.1 Sample of the labeled X-Rays

Table 1
Dataset

	Training	Testing	Total
Covid-19	796	212	1008
Pneumonia	846	210	1056
Normal	845	200	1045
Total	2487	622	3109

II.II PSO-OPTIMIZATION IN THE PROPOSED ARCHITECTURE

Dr. Kennedy first presented the PSO optimization technique in 1995[3]. A swarm of particles is randomly scattered over the search space at the beginning of the PSO algorithm. Each particle stands for a possible answer to the optimization issue. Each particle's position indicates a potential solution, and its velocity describes its travel in the search space in terms of both its direction and its speed. At respectively iteration, the particles move towards the best position they or their neighbors have visited so far, called the personal best (Pbest) and global best (Gbest), respectively. The movement of each particle is determined by the velocity update equation:

$$v_i(t+1) = w * v_i(t) + c_1 * r_1 * \left(Pbest_i - x_i(t)\right) + c_2 * r_2 * \left(Gbest - x_i(t)\right) \quad (1)$$

where $v_i(t)$ is the velocity of particle i at time t, $x_i(t)$ is the position of particle i at time t, $Pbest_i$ is the personal best position of particle i, $Gbest$ is the global best position among all particles, w is the inertia weight that controls the impact of the previous velocity, c_1 and c_2 are the acceleration coefficients that control the influence of $Pbest$ and $Gbest$, respectively, and r_1 and r_2 are random numbers uniformly distributed between 0 and 1the velocity update equation is used to compute the new position of each particle

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

After updating the positions of all particles, the fitness function is evaluated for respectively particle, and the personal and global best positions are efficient if necessary. Unless a stopping requirement is satisfied, such as a maximum number of iterations or a suitable fitness value, the procedure is repeated.

PSO algorithm reads the images from the specified directory, resizes them to 32x32 pixels, converts them to numpy arrays, and appends them to the X array. It also assigns labels to the images based on their directories. Then, it shuffles the X array and normalizes its values.PSO algorithm, including the number of particles, dimensions, and options. It creates an instance of the PSO optimizer object and applies it to the cost function, f, which is defined as f_per_particle. The PSO algorithm is run for 2 iterations to optimize the features. Finally, the it selects the optimized features with a value of 1 from the PSO and saves them as X_selected_features. It also saves the labels as Y. The code displays some information about the dataset, including the total number of images, the total number of features before and after applying PSO, and the classes in the dataset in Fig 2

```
Total images found in dataset: 3109
Total features found in image before applying PSO: 1024
Total features found in image after applying PSO: 576
Classes found in dataset: ['Covid19', 'Normal', 'Penumonia']
```

Fig.2. PSO Optimized features

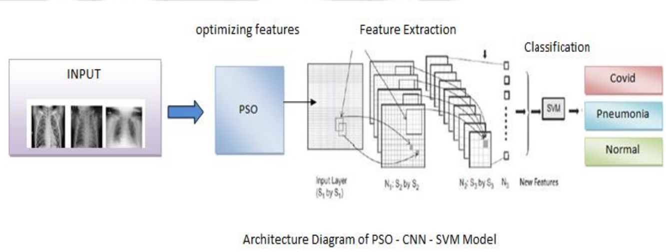

Fig 3. Architecture of the PSO-CNN- SVM Model

Algorithm 1. PSO Algorithm

1 Initialization
1.1 For each particle I in a swarm population size p:
1.1.1 Initialize X_i randomly
1.1.2 Initialize V_i randomly
1.1.3 Evaluate the fitness $f(X_i)$
1.1.4 Initialize $pbest_i$ with a copy of X_i
1.2 Initialize gbest with a copy of X_i with the best fitness
2 Repeat until a stopping criterion is satisfied:
2.1 For each particle in i:
2.1.1 Update V^t_i and X^t_i accordingly to Eq (1) ,(2)

2.1.2 Evaluate the fitness $f(X^t_i)$
2.1.3 $Pbest_i X^t_i \longleftarrow$ if $f(pbest_i) < f(X^t_i)$
2.1.4 $gbest_i X^t_i \longleftarrow$ if $f(gbest_i) < f(X^t_i)$

II.III CONVOLUTION NEURAL NETWORK IN THE PROPOSED ARCHITECTURE

An input layer, convolution and pooling layers, and a fully connected classification layer make up the CNN architecture. The most crucial component of the CNN architecture is the convolution layer. This layer is made up of kernels or filters that cover the entire input. Each unit in this layer receives input from the layer above it., the last layer is dense with no of classes. The developed CNN for feature extraction has total 141,667 parameters.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 22, 22, 32)        320
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 11, 11, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 9, 9, 32)          9248
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 4, 4, 32)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 512)               0
_________________________________________________________________
dense_1 (Dense)              (None, 256)               131328
_________________________________________________________________
dense_2 (Dense)              (None, 3)                 771
=================================================================
Total params: 141,667
Trainable params: 141,667
Non-trainable params: 0
```

Fig: 4 Model Summary of PSO CNN

II.IV SUPPORT VECTOR MACHINE IN THE PROPOSED ARCHITECTURE

After feature extraction and pre-processing, an SVM classifier is used to categorise the images. Using feature vectors that are stored as matrices, the SVM classifier's training procedure is carried out. Using the learned SVM classifier, testing of the images is then done. In contrast, the extracted features are given to the SVM module in the PSO CNN-SVM model for the dataset's training and testing. Both the training and testing data are used, and the results are recorded, to assess the accuracy of the SVM classifier and the PSO-CNN-SVM model. The hybrid model uses an RBF kernel function, and the SVM parameters, including degree and gamma of the kernel function and the shape of the decision function, are carefully determined because they have a big impact on SVM classification. [4]

III. EXPERIMENTAL SETUP AND RESULTS

For experiments, Google Colaboratory is used. For up to 8 hours, Colab offers 12GB of RAM with an NVIDIA Tesla K80 GPU. We have used the pyswarms ,swarmpackagepy, sklearn, matplotlib, libraries to implement the proposed model. The proposed model uses the Adam optimizer with loss as definite cross entropy metrics = correctness. The extracted features from CNN are passed to SVM classifier for classification; we have achieved promising results with an overall correctness of 97.42

III.I PERFORMANCE EVALUATION

The performance of PSO-CNN-SVM Model has been assessed by means of confusion matrix, classification report and correctness vs. loss graphs.

III.II CONFUSION MATRIX

Confusion Matrix generally has a 2 × 2 matrix to every cell representing the model detection rate as True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

1. **True Positive (TP):** If the individual is essentially Covid positive and forecast also Covid then it is Called True Positive
2. **True Negative (TN):** If the subject is normal and the outcome is as expected, this is known as a true negative.
3. **False Positive (FP):** When a normal person is identified as having covid, it is referred to as a false positive and it reflects an inaccurate detection.
4. **False Negative (FN):** False Negative is a term used to describe an inaccurate detection when a Covid positive person predicts Normal.

Accuracy: (3)

$$\frac{\text{True Positive} + \text{True Negative}}{True\ Positive + False\ Positive + True\ Negative + False\ Negative}$$

Precision: (4)

$$\frac{\text{True Positive}}{True\ Positive + False\ Positive}$$

Recall: (5)

$$\frac{\text{True Positive}}{True\ Positive + False\ Negative}$$

Fl Score: (6)

$$2x\,\frac{\text{PRECISION} * \text{RECALL}}{Precision + Recall}$$

ISSN: 2321-8169 Volume: 11 Issue: 5s
DOI: https://doi.org/10.17762/ijritcc.v11i5s.6646
***Article Received**: 27 February 2023 **Revised**: 08 April 2023 **Accepted**: 23 April 2023*

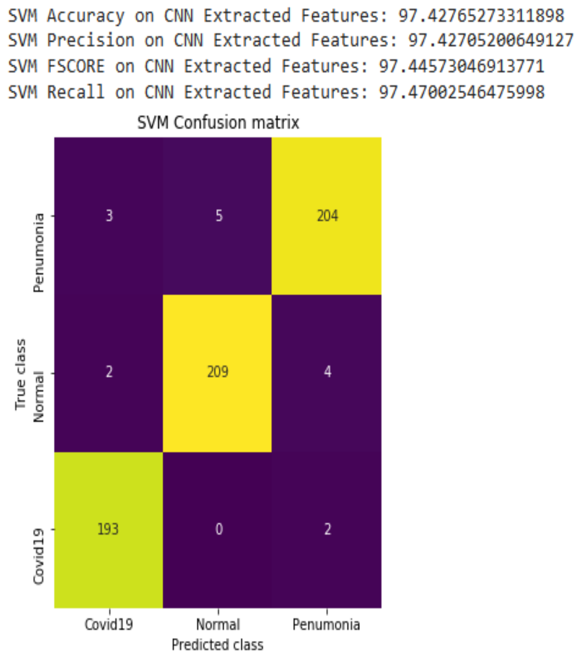

Fig 5. Confusion matrix with Classification Report

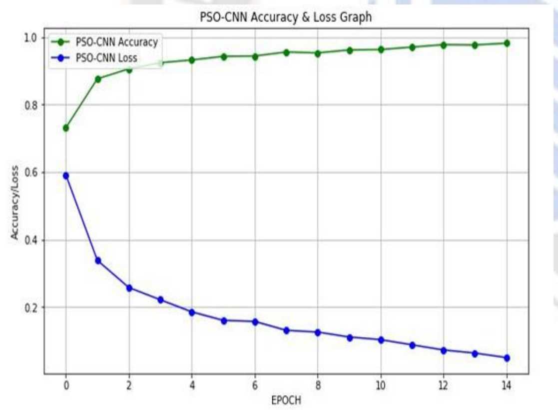

Fig 6. Accuracy vs Epoch Graph

IV. PERFORMANCE EVALUATION

This section compares the accuracy with some best models developed previously in multi classification with X-Rays. Though we found many articles on Covid-19. We have chosen only ten articles on 3 class classification with X-Rays. However from the below table we see our model performing better than previously developed models. We also found the scope of improving our model further using tuning the hyper parameters and increasing the dataset

Table 2: Comparison of Deep Learning Models

AUTHOR	MODEL	DATASET	CLASS	ACC
Arman Haghanifar et al [5]	COVID-CXNe	X-RAY	3	87.88
M.k. Pandit et al [6]	VGG-16	X-RAY	3	92.50%
Sarki R et al [7]	CNN	X-RAY	3	93.75%
Emtiaz Hussain et al[8]	CoroDet	X-RAY	3	94.20%
Ahmed s Elkorany[9]	Covidetection- Net	X-RAY	3	94.44%
Mahmoud Ragab [10]	CAPSnET	X-RAY	3	95.00%
Tianbowu et al [11]	ULNet	X-RAY	3	95.35%
Anubhav Sharma et al [12]	COVDC-NET	X-RAY	3	96.48 %
Abhijit Bhattacharya et al[13]	VGG-19	X-RAY	3	96.60%
Gaurav Srivastava et al[14]	CoviXNet	X-RAY	3	96.61%
Our proposed Model	**PSO-CNN-SVM**	**X-RAY**	**3**	**97.42 %**

V. CONCLUSION

The paper presents the hybrid model developed with combination of particle swarm optimization, CNN and SVM (PSO-CNN-SVM) model to optimize the features. The model produced good results of 97.42%. This method can be further improved and achieve better accuracy with larger datasets.

REFERENCES

[1] V. Venugopal, J. Joseph, M. Vipin Das, M. Kumar Nath, An EfficientNet-basedmodified sigmoid transform for enhancing dermatological macro-images ofmelanoma and nevi skin lesions, Comput. Methods Programs Biomed. 222 (2022)106935doi https://doi.org/10.1016/j.cmpb.2022.106935

[2] E. Benmalek, J. Elmhamdi, A. Jilbab, Comparing CT scan and chest X-ray imagingfor COVID-19 diagnosis, Biomed. Eng. Adv.1 (2021) 100003, https://doi.org/10.1016/j.bea.2021.100003

[3] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95- international Conference on Neural Networks, vol. 4, IEEE, 1995, pp. 1942–1948.

[4] dugganikeerthana, vipinvenugopal, Malaya kumarnath, madhusudhanmishra, "Hybrid Convolutional neural networks with SVM classifier for classification of skin cancer, Biomedical Engineering Advances 5 (2023) 100069doi : https://doi.org/10.1016/j.bea.2022.100069

[5] Arman Haghanifar, MahdiyarMolahasaniMajdabadi · Younhee Choi, S. Deivalakshmi · Seokbum Ko, COVID-CXNet: Detecting COVID-19 in frontal chest X-ray images using deep learning, Multimedia Tools and Applications (2022) 81:30615–30645, https://doi.org/10.1007/s11042-022-12156-z

[6] M.K. Pandit, S.A. Banday, R. Naaz, M.A. Chisti, "Automatic detection of Covid-19 from chest radiography using deep learning. Radiography 27 (2021), 483-489. Doi: https://doi.org/10.1016/j.radi.2020.10.018

[7] Sarki R, Ahmed K, Wang H, Zhang Y, Wang K (2022) Automated detection of COVID-19through convolutional neural network using chestx-ray images. PLoS ONE 17(1): e0262052. https://doi.org/10.1371/journal.pone.0262052

[8] EmtiazHussain , Mahmudul Hasan , Md Anisur Rahman , Ickjai Lee , Tasmi Tamanna ,Mohammad Zavid Parvez," CoroDet: A deep learning based classification for COVID-19 detectionusing chest X-ray images", Chaos, Solitons and Fractals 142 (2021) 110495 doi :https://doi.org/10.1016/j.chaos.2020.110495

[9] Ahmed S. Elkorany, Zeinab F Elsharkawy,COVIDetection-Net: A tailored COVID-19 detection from chestradiography images using deep learning",Optik - International Journal for Light and Electron Optics 231 (2021)166405, doi :https://doi.org/10.1016/j.ijleo.2021.166405

[10] Mahmoud Ragab, SamahAlshehri,Nabil A. Alhakamy ,Romany F. Mansour,and Deepika Koundal," Multiclass Classification of Chest X-Ray Images for thePrediction of COVID-19 Using Capsule Network," Computational Intelligence and Neuroscience, Volume 2022, Article ID 6185013, 8 pages doi : https://doi.org/10.1155/2022/6185013

[11] Tianbu Wu, Chen Tang, Min Xu, Nian Hong, Zhenkun Lei, ULNet for the detection of coronavirus (COVID-19) from chest X-ray images," Computers in Biology and Medicine 137 (2021) 104834" doi : https://doi.org/10.1016/j.compbiomed.2021.104834

[12] Anubhav Sharma, Karamjeet Singh, Deepika Koundal,"A novel fusion based convolutional nerural network approach for classification of Covid 19 from chest X -ray images",Biomedical Signal Processing and Control 77(2022) 103778 doi ;https://doi.org/10.1016/j.bspc.2022.103778

[13] Abhijit Bhattacharya, DivyanshuBhaik, Sunil Kumar, Prayas Thakur, Rahul Sharma, Ram Bilas Pachori, "A deep learning based approach for automatic detection of COVID-19 cases using chest X-ray images" https://doi.org/10.1016/j.bspc.2021.103182

[14] Gaurav Srivastava, Aninditaa Chauhan, Mahesh Jangid, Sandeep Chaurasia,"CovixNet: A Novel and efficient deep learning model for detection of Covid-19 using chest X-Ray images."Biomedical Signal Processing and Control 78 (2022) 103848 doi :https://doi.org/10.1016/j.bspc.2022.103848