

OPTIMIZING HYPERPARAMETERS OF A DEEP LEARNING ALGORITHM FOR A REAL TIME FACE RECOGNITION SYSTEM

Mayowa O. OYEDIRAN¹, Olufemi S. OJO², Adeyinka M. AMOLE³, Olubunmi J. JOODA⁴, Olufikayo A. ADEDAPO⁵

Department of Computer Engineering, Ajayi Crowther University, Oyo, Oyo State, Nigeria¹

Department of Computer Science, Ajayi Crowther University, Oyo, Oyo State, Nigeria^{2,3}

Department of Computer Engineering, Redeemer's University, Ede, Osun State, Nigeria⁴

Department of Mathematical and Computer Sciences, Fountain University, Osogbo⁵



ABSTRACT— Convolutional Neural Networks (CNNs) have demonstrated remarkable success in various image recognition and classification tasks. This model cannot perform well in challenging conditions, such as, low light, varying camera angles and occlusions or crowded scenes. Therefore, this research Developed Spider wasp Optimized Convolutional neural network base on real time face recognition system (SWO-CNN). The outcomes of this research thus lend credence to the assertion that the SWO method, when combined with CNN technology, enhanced the detection of faces while also accelerating improved. This research has contributed to knowledge with an empirical proof that the application of SWO based CNN technique achieved an improved performance with low processing time in the detection of faces in videos processing. With the developed technique, face detection in surveillance systems could be greatly improved.

KEYWORDS: Convolution Neural Network, Spider Wasp Optimizer, Metaheuristics optimization algorithms, Hyper-parameters, False Positive Rate.

1. INTRODUCTION

Convolutional neural networks, or CNNs, have produced outstanding results in a range of image identification and classification applications [1], [2]. CNNs are capable of automatically learning relevant features from input images, making them suitable for detecting complex patterns in face recognition system [3]. It has been established that CNNs are utilized for many forms of picture classification as evidenced by recent state-of-the-art research on deep learning applications. There is not a single network that works well for all kinds of problems thus a CNN with a suitable architecture needs to be designed for every real-world situation [4]. The values of the CNN's hyperparameters determine its architecture. It would take a long time to manually search for the right combination of hyperparameters [5]. The optimization of hyperparameters is considered as an NP-hard problem due to the comparatively huge search space, and metaheuristics have been revealed to be highly effective in solving these kinds of problems [6].

Metaheuristic optimization algorithms are approximation techniques that, while they may not always produce the optimal outcome, will locate a close alternative [7]. Two search types—exploration and exploitation—and randomization are the distinguishing features of metaheuristics. A global search space exploration is the function of the exploration process, whilst the algorithm searches locally for a solution surrounding the best or other solution in its proximity during the exploitation phase [8]. The two major types of metaheuristics are swarm intelligence and evolutionary algorithms. Notable among them are

Particle Swarm Optimization and Spider Wasp Optimizer [9].

Real-time face detection is crucial in numerous applications, including security, surveillance, and human-computer interaction [10], [11]. Existing real-time face recognition systems often struggle to deliver accurate results under varying lighting conditions, facial expressions, and occlusions [12], [13]. This study developed a resource-efficient alternative for face detection by optimizing the CNN using SWO, which could be essential for deploying face detection systems on resource-constrained devices.

2. Related Works

2.1 Spider Wasp Optimizer

A new optimization technique called Spider Wasp Optimizer (SWO) is based on the hunting and nesting patterns of some wasp species, as well as their required brood parasitism, which involves placing a single egg in each spider's abdomen [14]. Initially, female spider wasps scavenge their surroundings for appropriate spiders, immobilizing and pulling them to ready-made nests; this activity serves as the model for our suggested algorithm, SWO. They identify the appropriate prey and nests, drag them inside, lay an egg on the spider's abdomen, and then shut the nest [15]. Within the search space, a number of female wasps are assigned at random by the algorithm (SWO). After that, each one will continually explore the search space for a spider that matches the sex of its progeny, which is decided by the haplodiploid sex-determination mechanism present in all hymenopterans based on their hunting behaviors, also referred to as hunting and following behaviors [16]. Following the identification of suitable spiders, the female wasps will feed the spiders inside their web hub and then do six searches of the ground to find any spiders that have dropped from the web. Next, the female wasps will attempt to immobilize the victim so that it can be carried to the nest that has been built by attacking it [17].

The mathematical model of those behaviors, is presented as follow:

The following equation can be used to encode each female spider-wasp in the D-dimension vector, which in the proposed approach shows a solution in the current generation:

$$\overrightarrow{SW} = [x_1, x_2, \dots, x_D] \quad 1$$

Set of N vectors could be generated randomly between upper parameter bound \vec{H} and the lower initial parameter bound \vec{L} , as:

$$SW_{POP} = \begin{bmatrix} SW_{1,1} & SW_{1,2} & \dots & SW_{1,D} \\ SW_{2,1} & SW_{2,2} & \dots & SW_{2,D} \\ \vdots & \vdots & \vdots & \vdots \\ SW_{N,1} & SW_{N,2} & \dots & SW_{N,D} \end{bmatrix} \quad 2$$

where SW_{POP} is the spider wasps initial population.

$$\overrightarrow{SW}_i^t = \vec{L} + \vec{r} \times (\vec{H} - \vec{L}) \quad 3$$

where t denotes the generation index; i denotes the population index ($i = 1, 2, \dots, N$); and \vec{r} is a vector of D-dimension. Next, a novel metaheuristic approach was employed to address the optimization problems by

mathematically simulating the behaviors of spider wasps [15].

2.2 Related Literature

[18] used a face recognition model to compare and detect faces, which has been researched to stop thieves from robbing other people's homes and to identify the criminal's image. In addition, the alarm feature is crucial nowadays to get a response from the model and is based on increasing the number of image capture frames to set the accuracy and loss, which are dependent on the learning rate and iteration count.

[19] identified and tracked the face of an in-court witness. In the work, human faces are extracted using the Viola-Jones method, and the image is subsequently cropped using a specific transformation. Convolutional Neural Networks (CNN) are used for the classification of witness and non-witness images. Trained features were used to track the witness face using the Kanade-Lucas-Tomasi (KLT) algorithm. In order to optimize each method's speed and minimize the space needed for CNN implementation and detection accuracy, the two approaches were integrated into one model in this model. Following the test, the suggested model's results demonstrated that, when used in real-time and under suitable conditions, accuracy was 99.5%.

[20] developed a real-time framework for the detection and recognition of human faces in closed-circuit television (CCTV) images using machine learning and deep learning techniques. Two feature extraction approaches were employed in the study: convolutional neural networks (CNN) and principal component analysis (PCA). K-nearest neighbor (KNN), decision tree, random forest, and CNN algorithms were employed in the study, and their performances were compared. By using these methods on a dataset of over 40K real-time photos captured at various conditions (such as light intensity, rotation, and scaling for simulation and performance evaluation), recognition is accomplished. In the end, the study identified faces with over 90% accuracy and a minimum computation time.

Most existing work could not focus on the development of techniques for efficient domain adaptation, enabling face detection models trained on one dataset to be applied to different real-world scenarios or environments with minimal retraining. Most research could not investigate how to optimize face detection CNN models for deployment on edge devices, where energy efficiency and resource constraints are critical factors.

3. Methodology

In developing a real-time face detection system using Spider Wasp Optimization based Convolutional Neural Network algorithm (SWO-CNN), the following stages were involved.

- i. Acquisition of MP4 datasets were obtained as a primary data and from Youtube.Com.
- ii. Face Detection from the acquired video frames using Viola-Jones Algorithm
- iii. Pre-processing of the face detected by resizing the images, cropping the images, conversion to grayscale and adjusting their brightness and contrast.
- iv. Use SWO to select CNN hyperparameters such as weights, number of layers, filter size and number of filters.
- v. Feature Extraction, Training and Recognition of face images were achieved by using Spider Wasp Optimized based Convolutional Neural Network (SWO-CNN).
- vi. Evaluation of the proposed SWO-CNN with existing CNN technique for real-time face detection was done using false positive rate, sensitivity, specificity, precision, accuracy, and recognition time.

3.1 Video Acquisition and Pre-processing

The uncompressed video files were obtained online via YouTube in AVI format, which are amongst the

most popular videos, have the same resolution, and are high definition. The acquired video was split into its corresponding frames using Viola-Jones Algorithm in MATLAB. Before being used as input for feature extraction, the raw video signal from the acquired video data was first pre-processed. The acquired data's composite video signal was digitized into a time series of raw 120 x 160 RGB images using a video capture board. Each RGB color image was then converted into a YUV representation, and a difference (D) image was created by calculating the absolute value of the difference between consecutive frames. The four YUVD images were then subsampled successively at each time step to produce representations at lower and higher resolutions.

3.2 Image pre-processing

The following: Image brightness, contrast alteration, image filtering, scaling, cropping and conversion into grayscale, cropping the image, normalizing of face vectors by computing the average face vector. It deducts average face from each face vector.

3.3 The CNN Hyper-parameter Optimization

There are various drawbacks to pre-trained CNN models. The drawbacks are as follows: the batch size, as well as the unit numbers in each dense layer and dropout layer, are among the hyper-parameters that need to be adjusted. The majority of these hyper-parameters in any pre-trained CNN cannot be changed. To optimize the batch size and dropout layer rate in this study, the SWO technique was used in the CNN architecture models classifier part.

The number of convolutional layers, the number of convolutional filters, the size of the filters used in each convolutional layer, and the batch size are the dynamic parameters that SWO optimizes. The male and female spider wasps are produced in this procedure by initializing the SWO in accordance with the execution parameter provided in Algorithm 1. Considering the position of each spider wasp has a parameter that needs to be optimized, every possible solution comprises an entire CNN training set.

Algorithm 1: Spider Wasp Optimized based CNN

Input: CNN parameters such as the weights, number of layers and filters, $N, Nmin, CR, TR, tmax$

Step1: Initialize N female wasps, \overrightarrow{SW}_i ($i = 1, 2 \dots \dots, N$), using

$$\overrightarrow{SW}_i^t = \vec{L} + \vec{r} \times (\vec{H} - \vec{L})$$

Step2: Evaluate each \overrightarrow{SW}_i and finding the one with the best fitness in \overrightarrow{SW}^*

Step3: $t = 1$; //the current function evaluation

Step4: while ($t < t_{max}$)

Step5: r_6 : generating a random number between 0 and 1

Step6: If ($r_6 < TR$) %% Hunting and Nesting behaviors

Step7: for $i = 1:N$

Step8: **Applying**

Step9: **Compute** $f(\overrightarrow{SW}_i)$

Step 10 $t = t + 1$

Step11: End for

Step12: Else %% Mating Behavior

Step13: for $i = 1:N$

Applying

$$SW_i^{t+1} = \text{Crossover}(SW_i^t, SW_m^t, CR)$$

SW_m^t and SW_i^t are two vectors that represent the male and female spider wasps, respectively.

Step14: $t = t + 1$
 Step15: End for
 Step16: End if
 Step17: Applying *Memory Saving*

Updating N

$$N = N_{min} + (N - N_{min}) \times k$$

where N_{min} is the minimum number of the population

Step18: End while

Step19: Output optimum CNN parameters was selected solution as: \overline{SW}^*

Upon evaluation of each generation of spider wasps generated by the SWO, the training process comes to an end. It is an iterative cycle. The size of the spider wasps, the number of SWO iterations, the number of male and female spider wasps in each iteration, and the database size all determine computational cost, which is higher. In other words, 100 iterations of the CNN training procedure would be carried out if the SWO were to be carried out using 10 male and female spider wasps. The steps to optimize the CNN by the SWO algorithm are explained as follows.

- i. FACE datasets were used to train the CNN. The FACE datasets (accessible or deniable faces) must be chosen in this stage in order for the CNN to process and recognize images.
- ii. Generate the population of spider wasps needed by the SWO algorithm. The number of male and female spider wasps, as well as the number of iterations, are set as SWO parameters. The spider wasps' design is the focus of this step.
- iii. Initialize the CNN architecture: When the CNN is initialized and combined with the additional parameter given, it is prepared to train the input FACE datasets. The parameters obtained by the SWO are the number of convolution layers, the filter size, the number of convolution filters, and the batch size.
- iv. CNN training and validation: The CNN first reads and processes the input FACE databases comprising the images for testing, validation, and training in order to determine the recognition rate. The objective function that returns to the SWO includes these values.
- v. Evaluate the objective function: In order to determine the optimal value, the SWO algorithm examines the objective function.
- vi. Hunting and Nesting SWO parameters. At each iteration, each male and female spider wasp hunting and nesting behaviors depending on its own best-known position (\overline{SW}^*) in the search-space and the best-known position in the whole population (\overline{SW}^*).
- vii. The process is repeated, evaluating all the spider wasps until the stop criteria are found (in this case, it is the number of iterations).
- viii. Finally, the optimal CNN parameters were selected. In this process, the spider wasp represented by \overline{SW}^* is the optimal one for the CNN model.

Figure 1 shows the face Recognition using Spider Wasp Optimizer based Convolutional Neural Network (SWO-CNN).

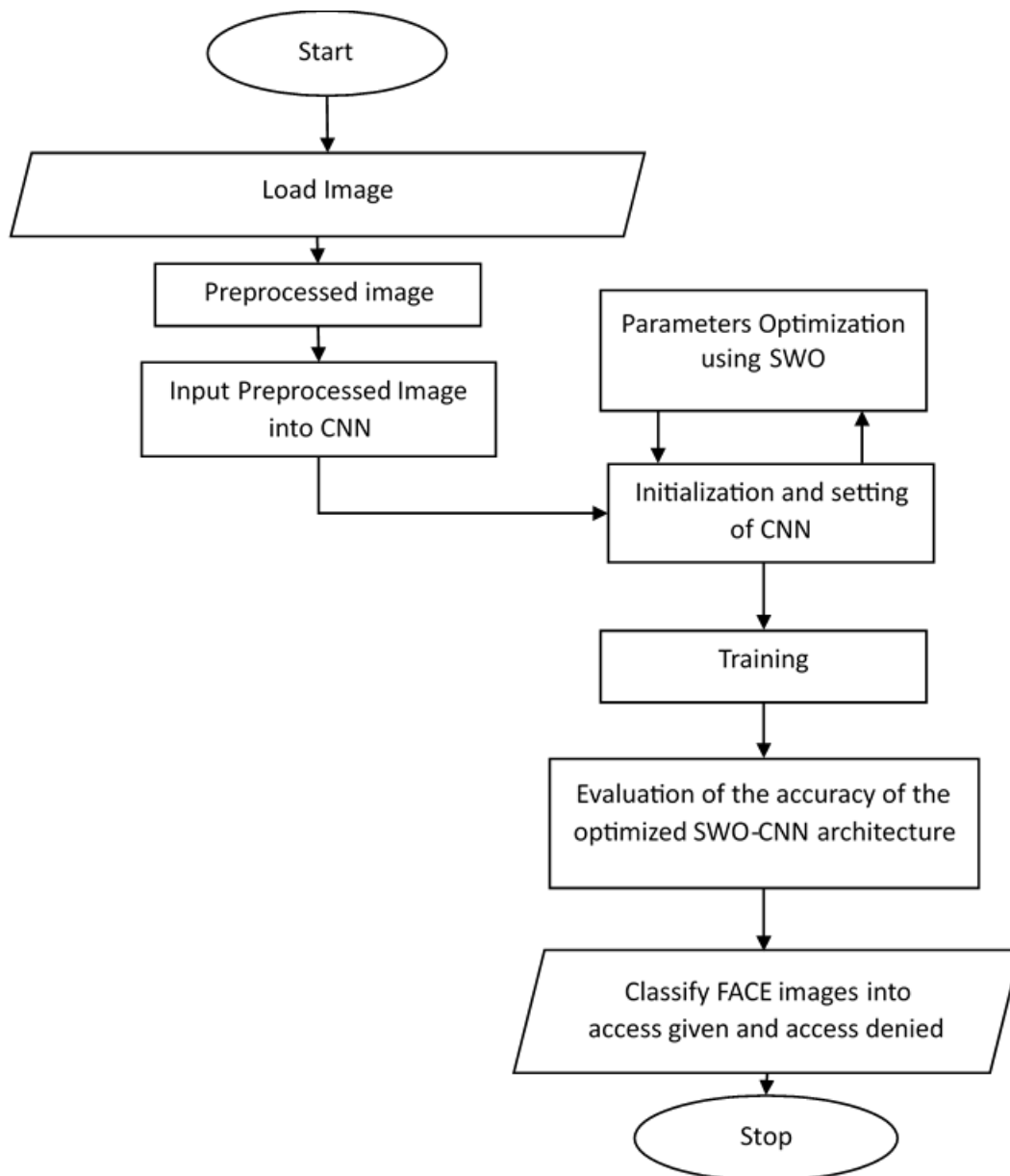


Figure 1: Face Recognition using Spider Wasp Optimizer based Convolutional Neural Network (SWO-CNN)

3.4 Results

The study was implemented on the computer with Intel(R) Core (TM) i7-480M CPU 2.7 GHz and 8.00G memory. Only four videos were used to evaluate the techniques in this study. The software is programmed using MATLAB R2016a. An extensive evaluation on the acquired videos with some people face based on the SWO-CNN and CNN method. Figures 2 show the graphical user interface of the simulation results.

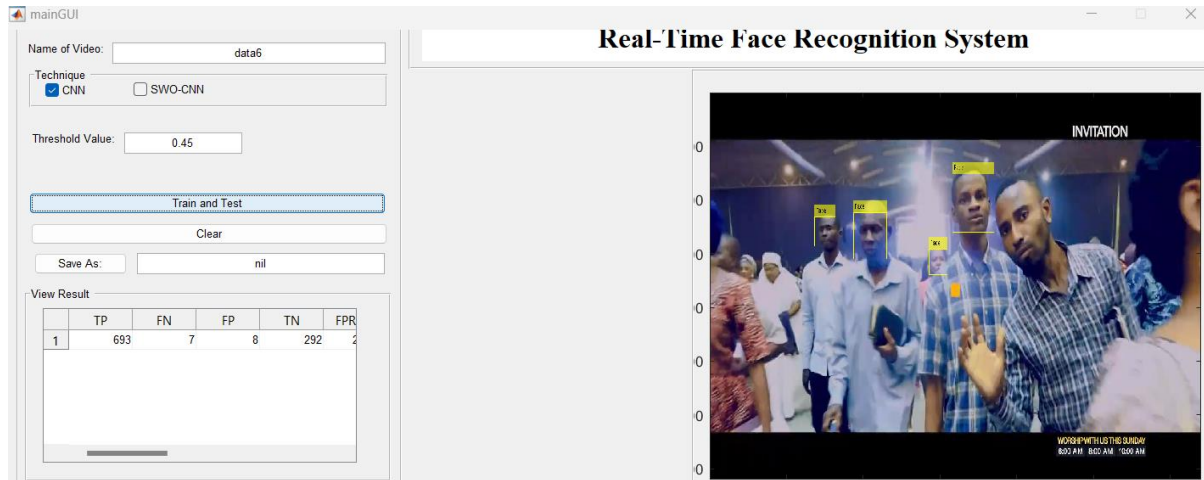


Figure 2: Graphical User Interface by CNN, SWO-CNN techniques during Training and Detection with locally acquired data.

Table 1: Result of performance evaluation of CNN and SWO-CNN on Video 1

Technique Used	CNN	SWO-CNN
Video Type	Mp4	Mp4
Total face	73	80
Object Detected	112	120
Correct Object (TP)	66	79
Misclassified Correct Object (FN)	7	1
Non-face (TN)	35	39
Misclassified Static Object (FP)	4	1
Accuracy (%)	90.18	98.33
Precision (%)	94.29	98.75
FPR (%)	10.26	2.50
Sensitivity (%)	90.41	98.75
Specificity (%)	89.74	97.50
Time	129.70	113.89

Table 2: Result of performance evaluation of CNN and SWO-CNN on Video 2

Technique Used	CNN	SWO-CNN
Video Type	Mp4	Mp4
Total face	111	127
Object Detected	169	186
Correct Object (TP)	106	124
Misclassified Correct Object (FN)	5	3
Non-face (TN)	51	58
Misclassified Static Object (FP)	7	1
Accuracy (%)	92.90	97.85
Precision (%)	93.81	99.20
FPR (%)	12.07	1.69
Sensitivity (%)	95.50	97.64

Specificity (%)	87.93	98.31
Time	159.22	137.66

Table 3: Result of performance evaluation of CNN and SWO-CNN on Video 3

Technique Used	CNN	SWO-CNN
Video Type	Mp4	Mp4
Total face	95	102
Object Detected	143	151
Correct Object (TP)	90	100
Misclassified Correct Object (FN)	5	2
Non-face (TN)	44	48
Misclassified Static Object (FP)	4	1
Accuracy (%)	93.71	98.01
Precision (%)	95.74	99.01
FPR (%)	8.33	2.04
Sensitivity (%)	94.74	98.04
Specificity (%)	91.67	97.96
Time	129.70	115.26

Table 4: Result of performance evaluation of CNN and SWO-CNN on Video 4

Technique Used	CNN	SWO-CNN
Video Type	Mp4	Mp4
Total face	113	120
Object Detected	172	180
Correct Object (TP)	106	119
Misclassified Correct Object (FN)	7	1
Non-face (TN)	55	59
Misclassified Static Object (FP)	4	1
Accuracy (%)	93.60	98.89
Precision (%)	96.36	99.17
FPR (%)	6.78	1.67
Sensitivity (%)	93.81	99.17
Specificity (%)	93.22	98.33
Time	264.30	232.07

Table 5: Result of performance evaluation of CNN and SWO-CNN on Video 5

Technique Used	CNN	SWO-CNN
Video Type	Mp4	Mp4
Total face	171	187
Object Detected	249	266
Correct Object (TP)	166	184
Misclassified Correct Object (FN)	5	3
Non-face (TN)	71	78
Misclassified Static Object (FP)	7	1

Accuracy (%)	95.18	98.50
Precision (%)	95.95	99.46
FPR (%)	8.97	1.27
Sensitivity (%)	97.08	98.40
Specificity (%)	91.03	98.73
Time	288.31	249.28

Table 6: Result of performance evaluation of CNN and SWO-CNN on Video 6

Technique Used	CNN	SWO-CNN
Video Type	Mp4	Mp4
Total face	160	167
Object Detected	226	234
Correct Object (TP)	155	165
Misclassified Correct Object (FN)	5	2
Non-face (TN)	62	66
Misclassified Static Object (FP)	4	1
Accuracy (%)	96.02	98.72
Precision (%)	97.48	99.40
FPR (%)	6.06	1.49
Sensitivity (%)	96.88	98.80
Specificity (%)	93.94	98.51
Time	261.24	232.16

Table 7: Result of performance evaluation of CNN and SWO-CNN on Video 7

Technique Used	CNN	SWO-CNN
Video Type	Mp4	Mp4
Total face	180	187
Object Detected	255	263
Correct Object (TP)	175	185
Misclassified Correct Object (FN)	5	2
Non-face (TN)	71	75
Misclassified Static Object (FP)	4	1
Accuracy (%)	96.47	98.86
Precision (%)	97.77	99.46
FPR (%)	5.33	1.32
Sensitivity (%)	97.22	98.93
Specificity (%)	94.67	98.68
Time	242.88	215.85

In comparison to CNN and SWO-CNN according to the results of the research, SWO-CNN techniques were more effective at detecting faces. A demonstration of the effectiveness of the methods for detecting faces is provided by the graphical representation of the simulation produced by the technique. The processing times for each method, are shown in Figure 2. When SWO are used in conjunction with CNN, faces are quickly detected, resulting in a faster processing time than with CNN alone.

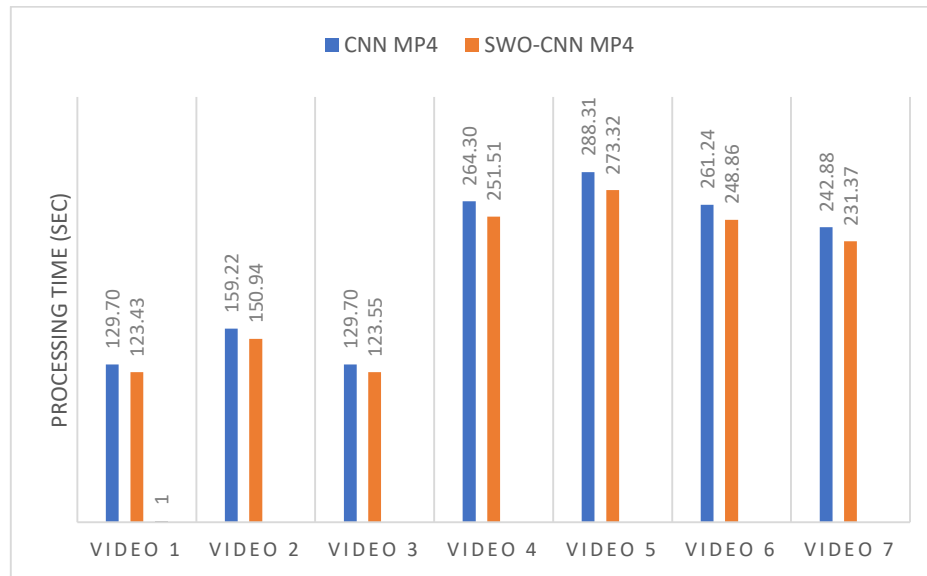


Figure 2: Graph of Processing Time against each technique with Mp4

According to the findings of this research, the SWO-CNN technique outperforms the CNN techniques in terms of accuracy, precision, FPR, specificity and processing time as asserted by [1], [21].

4. Conclusion And Recommendation

The evaluation results demonstrated that in terms of face detection accuracy, the SWO-CNN outperforms the CNN techniques. Precise detection was made feasible by the developed method's reasonable processing time. The outcomes of this research thus lend credence to the assertion that the SWO method, when combined with CNN technology, enhanced the detection of faces while also accelerating processing. With the developed technique, face detection in surveillance systems could be greatly improved.

5. References

- [1] Ojo, O. S., Oyediran, M. O., Bamgbade, B. J., Adeniyi, A. E., Ebong, G. N., & Ajagbe, S. A. (2023). Development of an improved convolutional neural network for an automated face based university attendance system. *ParadigmPlus*, 4(1), 18-28.
- [2] Olusanya O. O., Oyediran M. O., Elegbede A. W., Adeola A. O., Ojo O. S. (2023). An Offline Handwriting Age Range Prediction System Using an Optimized Deep Learning Technique. *International Journal of Emerging Technology and Advanced Engineering*. 13 (04), pp. 111-121
- [3] Oloyede, M. O., Hancke, G. P., & Myburgh, H. C. (2020). A review on face recognition systems: recent approaches and challenges. *Multimedia Tools and Applications*, 79(37), 27891-27922.
- [4] Bacanin, N., Bezdán, T., Venkatachalam, K., & Al-Turjman, F. (2021). Optimized Convolutional Neural Network By Firefly Algorithm For Magnetic Resonance Image Classification Of Glioma Brain Tumor Grade. *Journal of Real-Time Image Processing*, 18(4), 1085-1098.
- [5] Bibaeva, V. (2018, September). Using metaheuristics for hyper-parameter optimization of convolutional neural networks. In *2018 IEEE 28th international workshop on machine learning for signal processing (MLSP)* (pp. 1-6). IEEE.

- [6] Stoean, C., Zivkovic, M., Bozovic, A., Bacanin, N., Strulak-Wójcikiewicz, R., Antonijevic, M., & Stoean, R. (2023). Metaheuristic-based hyperparameter tuning for recurrent deep learning: application to the prediction of solar energy generation. *Axioms*, 12(3), 266.
- [7] Halim, A. H., Ismail, I., & Das, S. (2021). Performance assessment of the metaheuristic optimization algorithms: an exhaustive review. *Artificial Intelligence Review*, 54(3), 2323-2409.
- [8] Morales-Castañeda, B., Zaldivar, D., Cuevas, E., Fausto, F., & Rodríguez, A. (2020). A better balance in metaheuristic algorithms: Does it exist?. *Swarm and Evolutionary Computation*, 54, 100671.
- [9] Chakraborty, A., & Kar, A. K. (2017). Swarm intelligence: A review of algorithms. *Nature-inspired computing and optimization: Theory and applications*, 475-494.
- [10] Kumar, A., Kaur, A., & Kumar, M. (2019). Face detection techniques: a review. *Artificial Intelligence Review*, 52, 927-948.
- [11] Ahmad, A. H., Saon, S., Abd Kadir Mahamad, C. D., Wiwoho, S., Mudjanarko, S. M. S. N., & Hariadi, M. (2021). Real time face recognition of video surveillance system using haar cascade classifier. *Indonesian Journal of Electrical Engineering and Computer Science*, 21(3), 1389-1399.
- [12] Adjabi, I., Ouahabi, A., Benzaoui, A., & Taleb-Ahmed, A. (2020). Past, present, and future of face recognition: A review. *Electronics*, 9(8), 1188.
- [13] Jegham, I., Khalifa, A. B., Alouani, I., & Mahjoub, M. A. (2020). Vision-based human action recognition: An overview and real world challenges. *Forensic Science International: Digital Investigation*, 32, 200901.
- [14] Fei, M., Gols, R., & Harvey, J. A. (2023). The biology and ecology of parasitoid wasps of predatory arthropods. *Annual review of entomology*, 68, 109-128.
- [15] Abdel-Basset, M., Mohamed, R., Jameel, M., & Abouhawwash, M. (2023). Spider wasp optimizer: A novel meta-heuristic optimization algorithm. *Artificial Intelligence Review*, 56(10), 11675-11738.
- [16] Mostafa, R. R., Khedr, A. M., Al Aghbari, Z., Afyouni, I., Kamel, I., & Ahmed, N. (2024). An adaptive hybrid mutated differential evolution feature selection method for low and high-dimensional medical datasets. *Knowledge-Based Systems*, 283, 111218.
- [17] Saber, S., & Salem, S. (2023). High performance technique for estimating the unknown parameters of photovoltaic cells and modules based on improved spider wasp optimizer. *SMIJ*, 5(2), 14.
- [18] Basil, N., Raad, M., Wazzan, A. N., & Marhoon, H. M. (2022). Face recognition with real-time framing based on multi task convolutional neural network: a case study. *Int. J. Mech. Eng*, 7(2), 3170-3178.
- [19] Obaida, T. H., Jamil, A. S., & Hassan, N. F. (2022). Real-time face detection in digital video-based on Viola-Jones supported by convolutional neural networks. *International Journal of Electrical & Computer Engineering* (2088-8708), 12(3).

[20] Ullah, R., Hayat, H., Siddiqui, A. A., Siddiqui, U. A., Khan, J., Ullah, F., and Karami, G. M. (2022). A real-time framework for human face detection and recognition in cctv images. *Mathematical Problems in Engineering*, pp. 1-12.

[21] Fregoso, J., Gonzalez, C. I., and Martinez, G. E. 2021. Optimization of Convolutional Neural Networks Architectures Using PSO for Sign Language Recognition. *Axioms*, 10(3), 139.



This work is licensed under a Creative Commons Attribution Non-Commercial 4.0 International License.