*LAUTECH Journal of Engineering and Technology 19 (4) 2025: 112-123*
10.36108/laujet/5202.91.0411

# A Comparative Analysis of Zebra Optimization Algorithm and Chaotic Sinusoidal Zebra Optimization Algorithm for Video Forgery Detection System

**[1]Oke A. O., [2]Adio M. O., [3]Oladosu J. B. and [4]Awodoye O. O.**

*[1,3,4]Department of Computer Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria*
*[2]Department of Computer Engineering, Ajayi Crowther University, Oyo, Nigeria*

**Article Info**

***Article history:***

**Received: July 21,** 2025
**Revised: Aug 14,** 2025
**Accepted: Aug 15,** 2025

***Keywords:***

Video forgery detection, Zebra Optimization Algorithm, Chaotic Sinusoidal ZOA, CNN, hyperparameter tuning, metaheuristics, digital forensics

***Corresponding Author:***
*mo.adio@acu.edu.ng*
*+2348035690315*

**ABSTRACT**

***Video forgery, including deletion, duplication, and insertion, threatens multimedia integrity, yet CNN-based Video forgery detection models often suffer from suboptimal hyperparameter tuning. This study compares the Zebra Optimization Algorithm (ZOA) and a Chaotic Sinusoidal variant (CSZOA) for optimizing CNN performance in forgery detection. ZOA was chosen for its balanced search capability, while Chaotic Sinusoidal mapping was integrated to improve population diversity, avoid local optima, and accelerate convergence. The framework embedded the optimizer in CNN transfer learning layers to fine-tune parameters such as learning rates, Number of filters, filter sizes, and batch size configurations. A dataset of 270 forged videos acquired from kaggle.com underwent preprocessing through frame extraction, resolution normalization and histogram equalization. Results show CSZOA-CNN outperforms ZOA-CNN and CNN, achieving 99.51% accuracy, 0.32% False Positive Rate, and 39.86 s detection time. These findings highlight the benefit of Chaotic Sinusoidal dynamics in enhancing CNN training efficiency and robustness for real-world video forgery detection.***

## INTRODUCTION

The proliferation of digital media and the rapid growth of video-sharing platforms have increased the risk of content manipulation through advanced forgery techniques such as frame deletion, duplication, and insertion. Such alterations can be used to spread misinformation, fabricate evidence, or undermine trust in digital communications. Consequently, reliable video forgery detection has become a crucial component in multimedia forensics, legal investigations, and security-sensitive applications.

Deep learning methods, particularly Convolutional Neural Networks (CNNs), have shown promising results in spatiotemporal feature extraction for forgery detection. However, their performance largely depends on the optimal configuration of hyperparameters such as learning rate, convolutional filter size, number of filters and network depth (Olayiwola *et al*, 2023; Oguntoye *et al.,* 2025). Conventional gradient-based tuning methods may lead to premature convergence or suboptimal performance, especially in complex search spaces encountered in video analysis tasks.

Metaheuristic algorithms have emerged as powerful alternatives for global optimization due to their ability to balance exploration and exploitation (Ogundepo *et al.,* 2022; Oguntoye *et al,* 2023). The Zebra Optimization Algorithm (ZOA), inspired by the social behavior and defense mechanisms of zebras, offers strong search adaptability in high-dimensional optimization problems. Nevertheless, like many swarm-based algorithms, ZOA may suffer from limited diversity in later iterations,

reducing its ability to avoid local optima. To address this, chaotic sinusoidal mapping was integrated into ZOA, producing the Chaotic Sinusoidal Zebra Optimization Algorithm (CSZOA). This modification enhances population diversity, improves search dynamics, and accelerates convergence toward optimal solutions.

## RELATED WORK

Video forgery detection has been extensively explored through deep learning and forensic feature analysis, with approaches differing in network architecture, feature extraction strategy, and robustness to manipulations. Early methods such as Kingra *et al*. (2017) and Liu and Huang (2017) relied on handcrafted spatio-temporal descriptors for inter-frame forgery detection, but these struggled against complex editing and compression. The emergence of deep learning shifted the focus toward convolutional neural networks (CNNs) and advanced architectures. Afchar *et al.* (2018) introduced MesoNet, a compact CNN optimized for facial forgery detection, achieving computational efficiency but limited generalization to non-facial tampering. Gan *et al*. (2019) leveraged a VGG-11-based CNN for object forgery detection in videos, enhancing spatial feature learning yet remaining vulnerable to temporal inconsistencies. Other approaches incorporated domain-specific cues, such as Li *et al*. (2018), who detected eye-blinking anomalies, and Ciftci *et al*. (2020), who utilized photoplethysmographic biological signals, both achieving high precision but only within constrained scenarios. Recurrent architectures, as in Güera and Delp (2018), improved temporal modeling for DeepFake detection, although training costs were significant. More recent works explored hybrid and attention mechanisms (Lu *et al*. 2021) with a 3D-attentional inception CNN, achieving robust detection but at the expense of higher model complexity. Ahmed and Sonuç (2023) integrated rationale-augmented CNNs to enhance interpretability while maintaining high detection accuracy. Some studies targeted residual signal analysis, such as El Rai *et al.* (2020) using noise-based CNNs, or capsule-based learning, as in Nguyen *et al*. (2019), both offering resilience to compression but facing scalability challenges. Forgery localization techniques like Jia *et al*. (2018) adopted a coarse-to-fine strategy for copy–move detection, excelling in precision but suffering from long processing times. Recently, Ugale and Midhunchakkaravarthy (2025) proposed a two-layer hybridized deep CNN classifier for efficient video forgery detection, combining multi-level feature fusion with deep learning to improve accuracy and reduce processing overhead, though their method still faced challenges in handling highly compressed and adversarially manipulated videos. Despite these advancements, current methods often encounter trade-offs between accuracy, speed, and robustness when addressing diverse forgery types (Nguyen and Tran, 2022; Oraibi and Radhi, 2022). While recent studies have improved feature representation and temporal coherence modeling, none have applied metaheuristic optimization, specifically the Zebra Optimization Algorithm (ZOA) or its chaotic variants, to tune CNN parameters for video forgery detection. This gap motivates the proposed Chaotic Sinusoidal Zebra Optimization Algorithm (CSZOA), which integrates ZOA's exploration–exploitation balance with Chaotic Sinusoidal dynamics to improve CNN training efficiency, detection robustness, and adaptability to multiple forgery types.

## METHODOLOGY

A dataset of 270 forged videos, encompassing deletion, duplication, and insertion manipulations, was obtained from Kaggle.com. Preprocessing involved frame extraction at a fixed rate, resizing to

224 × 224 pixels, histogram equalization, and label encoding. A transfer learning-based CNN served as the detection backbone, with hyperparameters such as learning rate, number of filters, filter size, and batch size optimized using the Zebra Optimization Algorithm (ZOA) and its Chaotic Sinusoidal variant (CSZOA). Figure 1 is the flow diagram for video forgery detection using CNN with hyperparameter optimization through Zebra Optimization Algorithm (ZOA) and Chaotic Sinusoidal ZOA (CSZOA). The process includes dataset acquisition, preprocessing, CNN model setup, optimization, training, and performance evaluation.

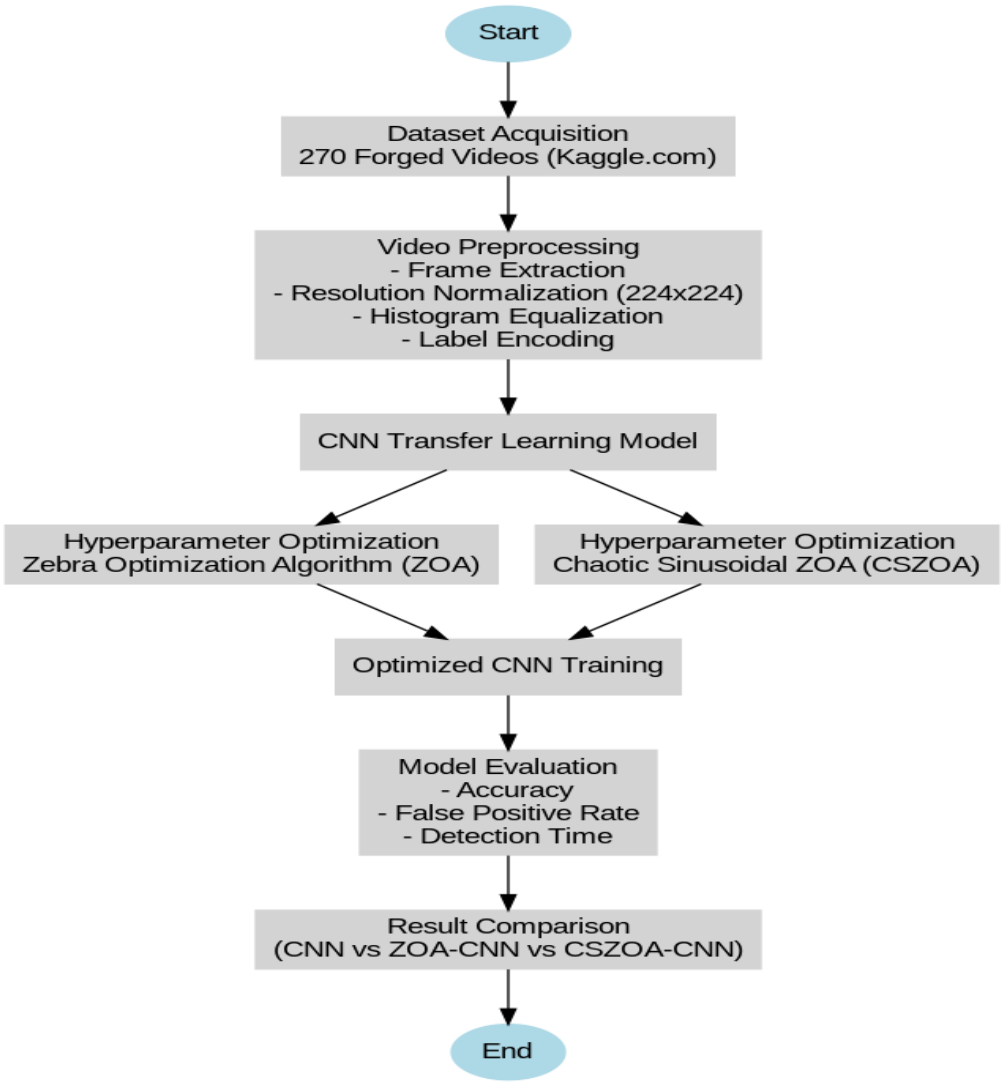


**Figure 1**: Flow diagram for Video Forgery Detection

**Dataset and Preprocessing**

The acquired video datasets were split into their corresponding frames using the Viola-Jones Algorithm in MATLAB. The raw video signals from the acquired video data were first pre-processed before being used as input for feature extraction. The pre-processing stage started by reading a video file frame by frame, converting each frame to grayscale, and applying object detection using the pre-trained Viola-Jones object classifier in the Algorithm. Another pre-processing technique used is contrast enhancement. The Red, Green, and Blue (RGB) images are converted into gray images using color conversion in Equation (1) (Saravanan, 2010).

$$\text{Gray}(x) = \alpha R + \beta G + \gamma B \quad (1)$$

Where α, β and γ are the coefficient factors of Red, Green and Blue of the images.

The dataset consists of 270 videos divided into five groups, each containing original and forged videos with deletion, duplication, and insertion forgeries. Frames were extracted and analyzed using a 10-fold cross-validation approach.

**Formulation of the Enhanced Zebra Optimization Algorithm**

The standard Zebra Optimization Algorithm (ZOA) exhibits limited localization efficiency, high sensitivity to initial conditions, weak local search capability, and parameter dependence, often resulting in premature convergence (Rana *et al*., 2022). These limitations arise from the position initialization and objective function evaluation mechanisms, defined in the original update rules Equation (2–5).

In the first phase, ZOA updates the i-th zebra's position as:

$$x_{i,j}^{newP1} = x_{i,j} + r.(PZ_j - I.x_{i,j}) \quad (2)$$

$$X_i = \begin{cases} X_i^{newP1}, & F_i^{newP1} < F_I \\ X_I & else \end{cases} \quad (3)$$

where PZ is the pioneer zebra, r∈[0,1], I∈{1,2} and PZj is the j-th dimension of PZ, $\boldsymbol{X_i^{newP1}}$ is the new status of the ith zebra based on the first phase, $\boldsymbol{x_{i,j}^{newP1}}$ is its jth dimension value, $\boldsymbol{F_i^{newP1}}$ is its objective function value. The second phase, Equation (6-9), handles exploitation and exploration for predator-defense strategies.

**Chaotic Sinusoidal Zebra Optimization Algorithm (CSZOA)**

To overcome ZOA's shortcomings, a Chaotic Sinusoidal ZOA (CSZOA) was developed by integrating a sinusoidal chaotic map into initialization, exploration, and exploitation stages. The chaotic map enhances population diversity, injects controlled randomness, and adaptively balances global and local search.

**First phase with chaotic update:**

$$x_{i,j}^{newP1} = x_{i,j} + r.(PZ_j - I.x_{i,j})$$

$$cx_{i,jinitial} = \frac{mod(abs(x_{i,j}^{newP1},\ x_{i,j})}{x_{i,j}^{newP1}}$$

$$cx_{i,jfinal} = a * cx_{i,jinitial}^2\ sin(\pi * cx_{i,jinitial})$$

$$CX_{i,j}^{new,P1} = cx_{i,jfinal} + r.(PZ_j - I.cx_{i,jfinal}) \quad \textbf{(4)}$$

$$CX_i = \begin{cases} CX_i^{newP1}, & F_i^{newP1} < F_I \\ CX_I & else \end{cases} \quad \textbf{(5)}$$

**Second phase with chaotic update:**

$$\boldsymbol{x_i^{newP2} = x_{i,j} + r.(AZ_j - I.x_{i,j})} \quad \textbf{(6)}$$

$$\boldsymbol{cx_{i,jinitial} = \frac{mod(abs(x_i^{new,P2},\ x_{i,j})}{x_i^{newP2}}} \quad \textbf{(7)}$$

$$\boldsymbol{cx_{i,jfinal} = a * cx_{i,jinitial}^2\ sin(\pi * cx_{i,jinitial})} \quad \textbf{(8)}$$

$$\boldsymbol{CX_i^{newP2}} = \begin{cases} \boldsymbol{S_1}:\ cx_{i,jfinal} + R.(2r-1).\left(1 - \frac{t}{T}\right).cx_{i,jfinal}, & P_s \leq 0.5 \\ \boldsymbol{S_2}:\ cx_{i,jfinal} + r.(AZ_j - I.cx_{i,jfinal}) & else \end{cases} \quad \textbf{(9)}$$

Here, AZ is the attacked zebra, t is the iteration index, T is the maximum iteration count, R=0.01, and $P_s \in [0,1]$ selects the update strategy. The chaotic mapping mechanism dynamically adapts search parameters according to optimization progress, preventing premature convergence and improving convergence accuracy. In CNN hyperparameter tuning for video forgery detection, CSZOA consistently yielded optimal fitness values and superior generalization performance. Algorithm 1 summarizes the procedure.

**CNN Architecture**

A basic CNN architecture was used, consisting of three convolutional layers followed by fully connected layers.

**Inputs:** CNN hyperparameters (learning rate, number of filters, filter size, batch size)
**Outputs:** Optimal CNN hyperparameters

1. Initialize parameters: number of iterations T, zebra population size N.
2. Use a sinusoidal chaotic map to initialize zebra positions and evaluate the objective function.
3. **For** t=1:T
   a. Update Pioneer Zebra (PZ) with chaotic perturbation for exploration.
   b. **For** i=1:N
   **Phase 1 – Foraging (Global Search):**
   - Update position:
   $x_{i,j}^{newP1} = x_{i,j} + r(PZj - Ix_{i,j})$
   $cx_{i,jinitial} = mod(|x_{i,j}^{newP1}, x_{i,j}|)/x_{i,j}^{newP1}$
   $cx_{i,j}^{final} = a(cx_{i,j\ initial})^2 \sin(\pi cx_{i,jinitial})$
   $\boldsymbol{CX_{i,j}^{newP1}} = cx_{i,jfinal} + r(PZj - Icx_{i,jfinal})$
   - Accept update if $F_i^{newP1} < Fi$
   **Phase 2 – Predator Defense:**
   - Generate $P_s \in [0,1]$.
   - **If** $P_s \leq 0.5$ (**Strategy 1 – Exploitation, vs. lion**):
   Update using:
   $x_i^{newP2} = x_{i,j} + r(AZj - Ix_{i,j})$
   Apply chaotic map as in Phase 1, with exploitation term $R(2r-1)(1-t/T)$.
   - **Else** (**Strategy 2 – Exploration, vs. other predators**):
   Update as above, with exploration emphasis via chaotic adjustment.
   - Accept update if $F_i^{newP2} < F_i$
   c. End for i.
   d. Save the best candidate solution.
4. End for t.
5. Return optimal CNN hyperparameters.

**Algorithm 1: Chaotic Sinusoidal Zebra Optimization Algorithm (CSZOA) for CNN Hyperparameter Selection**

The input to the network comprised preprocessed frames from the dataset. The optimization of CNN hyperparameters using the Zebra Optimization Algorithm (ZOA) and Chaotic Sinusoidal Zebra Optimization Algorithm (CSZOA) played a vital role in enhancing the performance of the video forgery detection model. These algorithms were used to fine-tune critical hyperparameters such as learning rate, number of filters, filter size, and batch size, which directly influence the network's learning efficiency and accuracy.

**CSZOA-CNN at Convolution Layer as Feature Extraction**

The optimal convolution layer is the most significant portion of the CSZOA-CNN design since it extracts specific information from images using a variety of optimal convolution kernel sizes. After conducting the convolution procedure multiple times, a group of feature maps is created from the input images. This is expressed as Equation 10 if $F_i$ is considered to be the ith layer in the CNN architecture.

$$F_i = \varphi\ (F_{i-1} Wcszoa_i + b_i) \quad \textbf{(10)}$$

where $F_i$ and $F_{i-1}$ are the feature maps of the current network layer and the previous layer, respectively. W and bi correspond to the weight and offset vector,

respectively, of the ith layer and φ ($F_{i-1}Wcszoa_i+b_i$) denotes the Rectified Linear Unit (ReLU) function.

**Optimal Pooling layer**

The minimum pooling layers generated by CSZOA are utilized to reduce the spatial dimensions, which in turn are expected to reduce the computational complexity and also the overfitting issues. The output feature on the jth location of the corresponding field in the *lth* pooling layer will be estimated with Equation 11.

$$X_j^l = \text{down}\left(X_j^{l-1}, S_{cszoa}\right) \quad \textbf{(11)}$$

where $down(X_j^{l-1}, S_{cszoa})$ corresponds to the down-sampling function, $X_j^{l-1}$ denotes the preceding layer feature vector and *s* stands for pooling size.

**CSZOA-CNN at Fully Connected Layers as Classification**

One or more Fully Connected Layers (FCL) after the optimal convolution and pooling layers in the CSZOA-CNN architecture are used to extract features to classify the forgery videos. The softmax function is used to conduct class identification with prior layer feature extraction. The softmax function is expressed in Equation 12.

$$(z) = (3)\ \sqrt[*]{Z}\ (fore = 1, 2, 3) \quad \textbf{(12)}$$

The newly built network can be fine-tuned in order to reduce the loss function (E), which is represented in Equation 13 by considering the datasets X and respective labels Y.

$$E(w) = -\frac{1}{n}\sum_{xi=1}^{n}\ \sum_{k=1}^{k}[ytk\log P(xi = k) + (1 - ytk)\log(1 - P(xi = k))] \quad \textbf{(13)}$$

n is the total number of samples, K is the number of classes, ytk is the ground truth label for sample i for class k, and P(xi=k) is the predicted probability that sample iii belongs to class k. In this research, the CSZOA algorithm is employed to estimate the optimum value of W by optimizing the loss function on the target datasets, which is expressed in Equation 14, where $\alpha_{cszoa}$ denotes the optimal learning rate.

$$W_k = W_{k-1} - \alpha_{cszoa}\left(de\ \frac{(W)}{d}\ W\right) \quad \textbf{(14)}$$

**ZOA for CNN Optimization**

ZOA mimics zebra herd behavior for optimization. It was used to tune the CNN's learning rate, filter count, filter size, and batch size. ZOA explored 30 iterations, evaluating different combinations of hyperparameters, and achieved its best result at iteration 27 with a learning rate of 0.00597, 64 filters, a filter size of 3, and a batch size of 128. This combination resulted in an objective function value of 0.0146, indicating a high optimization performance.

**CSZOA for CNN Optimization**

CSZOA enhances ZOA using chaotic sinusoidal functions to improve exploration and prevent local optima entrapment. CSZOA also underwent 30 iterations, and its best result was observed at iteration 27, but with a lower objective function value and faster convergence, similar to ZOA. However, the optimal configuration from CSZOA was slightly different: a learning rate of 0.00705, 64 filters, a filter size of 3, and a smaller batch size of 32. This configuration led to a lower objective function value of 0.0129, demonstrating an improved capacity to minimize the loss function more effectively than ZOA.

**Performance Evaluation**

The last phase of building the model is to make some predictions and evaluate the performance of the model and this is done using these parameters: False Positive Rate (FPR), Detection Accuracy (DA), and Detection Time (DT) as performance metrics in comparison with CNN, CNN-ZOA and CSZOA-CNN.

**False Positive Rate (FPR)**

The **FPR**, or **"Fall-Out"**, is the proportion of negative cases incorrectly identified as positive cases in the data (i.e., the probability that false alerts will be raised). It is defined in Equation (15) as the total number of negative cases incorrectly identified as positive cases divided by the total number of negative cases (i.e., normal data).

$$\text{False Positive Rate} = \frac{\text{FP}}{\text{FP+TN}} \quad \textbf{(15)}$$

**Detection Accuracy**

This measures the total number of correct classifications divided by the total number of cases, as it is indicated in Equation (16).

$$\text{Detection Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad \textbf{(16)}$$

Where:

**True positive TP**: If a frame is verified present in a dataset and the frame recognition system also affirms the presence of the frame, the result of the frame recognition system is true positive.

**True Negative TN**: If a frame is verified absent in a dataset and the frame recognition system also affirms the absence of the frame, the result of the recognition system is a true negative.

**False Positive FP**: If a frame recognition system confirms the presence of a frame in a dataset that does not really exist, the test result is a false positive.

**False Negative FN:** If a frame recognition system suggests the absence of a frame in a dataset that does exist, the test result is a false negative.

The sealed samples after attaining a state of secular equilibrium were each placed on the detector one after the other for analysis and were counted for 18000 s. An empty container was also counted for the same period of time so as to determine the background gamma-ray distribution count. The activity concentration A (Bq/kg) of each identified radionuclide in the sample was estimated using Equation 1:

$$A = \frac{C_{net}}{P_\gamma \times \varepsilon \times m \times t} \quad (1)$$

where $C_{net}$ is the net peak count (count/seconds) for each radionuclide present in the sample after subtracting the background count from the gross count, Pγ is the absolute gamma ray emission probability of the identified radionuclide, ε is the obtained full energy peak efficiency for each identified radionuclide, m is the mass (kg) of the sample and t is the counting time (s). In addition, the Minimum Detectable Activity (MDA) for each radionuclide was also calculated using Equation 2:

$$MDA = \frac{2.71+4.66\,(\sigma)}{\text{P}\gamma \times \text{E} \times \text{m} \times \text{t}} \quad (2)$$

where σ is the standard deviation of the background collected during time t over the energy range of interest.

**Statistical Analysis**

The mean activity concentrations of each radionuclide for each category of sample size were subjected to One-Way Analysis of Variance (ANOVA-1) and Scheffe Pair-Wise Comparisons tests to investigate any significant differences in their mean value.

**RESULTS AND DISCUSSION**

As summarized in Table 1, CSZOA-CNN consistently outperforms the baseline CNN and ZOA-CNN models across all evaluated metrics, including false positive rate, detection accuracy, and computational efficiency. The integration of the Chaotic Sinusoidal function into the Zebra Optimization Algorithm enables more effective hyperparameter tuning, resulting in superior generalization across different forgery types and video groups. These performance gains highlight CSZOA-CNN's capability to deliver high detection reliability while maintaining low processing time, making it particularly suitable for real-time and resource-constrained video forensic applications.

**False Positive Rate**

The experimental results in Figure 2 present the False Positive Rates for five video groups evaluated using three approaches: CNN, ZOA-CNN, and CSZOA-CNN. Each group was tested against three forgery types: deletion, duplication, and insertion.

**Table 1: Combined Evaluation Results based on CNN, ZOA-CNN and CSZOA**

| Metrics | Group of Videos | Group1 | | | Group2 | | | Group3 | | | Group4 | | | Group5 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Forged Type** | **Del** | **Dup** | **Ins** | **Del** | **Dup** | **Ins** | **Del** | **Dup** | **Ins** | **Del** | **Dup** | **Ins** | **Del** | **Dup** | **Ins** |
| **FPR (%)** | **CNN** | 1.23 | 1.35 | 1.49 | 1.28 | 1.64 | 1.92 | 1.37 | 1.66 | 2.10 | 1.56 | 1.79 | 2.29 | 1.64 | 1.97 | 1.91 |
| | **ZOA-CNN** | 0.75 | 0.71 | 0.90 | 0.79 | 0.87 | 1.16 | 0.84 | 0.87 | 1.27 | 0.95 | 0.94 | 1.38 | 1.00 | 1.04 | 1.15 |
| | **CSZOA-CNN** | 0.32 | 0.43 | 0.45 | 0.33 | 0.52 | 0.58 | 0.35 | 0.53 | 0.63 | 0.40 | 0.57 | 0.69 | 0.43 | 0.63 | 0.58 |
| **SPEC (%)** | **CNN** | 98.77 | 98.65 | 98.51 | 98.72 | 98.36 | 98.08 | 98.63 | 98.34 | 97.90 | 98.44 | 98.21 | 97.71 | 98.36 | 98.03 | 98.09 |
| | **ZOA-CNN** | 99.25 | 99.29 | 99.10 | 99.21 | 99.13 | 98.84 | 99.16 | 99.13 | 98.73 | 99.05 | 99.06 | 98.62 | 99.00 | 98.96 | 98.85 |
| | **CSZOA-CNN** | 99.68 | 99.57 | 99.55 | 99.67 | 99.48 | 99.42 | 99.65 | 99.47 | 99.37 | 99.60 | 99.43 | 99.31 | 99.57 | 99.37 | 99.42 |
| **SEN (%)** | **CNN** | 96.74 | 96.71 | 97.88 | 94.57 | 96.52 | 97.80 | 94.57 | 95.97 | 97.78 | 94.57 | 95.30 | 97.52 | 95.34 | 96.17 | 96.84 |
| | **ZOA-CNN** | 97.96 | 98.21 | 98.69 | 96.60 | 98.11 | 98.65 | 96.60 | 97.81 | 98.63 | 96.60 | 97.45 | 98.47 | 97.09 | 97.92 | 98.05 |
| | **CSZOA-CNN** | 99.07 | 98.88 | 99.31 | 98.46 | 98.81 | 99.28 | 98.46 | 98.63 | 99.28 | 98.46 | 98.40 | 99.19 | 98.68 | 98.70 | 98.97 |
| **PREC (%)** | **CNN** | 96.85 | 96.81 | 97.95 | 94.74 | 96.63 | 97.88 | 94.74 | 96.09 | 97.85 | 94.74 | 95.44 | 97.60 | 95.50 | 96.29 | 96.94 |
| | **ZOA-CNN** | 98.07 | 98.32 | 98.76 | 96.78 | 98.22 | 98.72 | 96.78 | 97.94 | 98.71 | 96.78 | 97.59 | 98.56 | 97.24 | 98.04 | 98.16 |
| | **CSZOA-CNN** | 99.18 | 98.98 | 99.38 | 98.64 | 98.92 | 99.36 | 98.64 | 98.75 | 99.35 | 98.64 | 98.55 | 99.28 | 98.83 | 98.82 | 99.08 |
| **ACC (%)** | **CNN** | 98.20 | 98.07 | 98.24 | 97.90 | 97.75 | 97.95 | 97.79 | 97.63 | 97.84 | 97.56 | 97.39 | 97.61 | 97.55 | 97.38 | 97.61 |
| | **ZOA-CNN** | 98.89 | 98.97 | 98.93 | 98.70 | 98.80 | 98.75 | 98.63 | 98.73 | 98.68 | 98.49 | 98.60 | 98.55 | 98.49 | 98.60 | 98.54 |
| | **CSZOA-CNN** | 99.51 | 99.36 | 99.45 | 99.43 | 99.26 | 99.36 | 99.40 | 99.22 | 99.32 | 99.34 | 99.14 | 99.25 | 99.33 | 99.14 | 99.25 |
| **F1-SCORE (%)** | **CNN** | 96.79 | 96.76 | 97.91 | 94.66 | 96.57 | 97.84 | 94.66 | 96.03 | 97.82 | 94.66 | 95.37 | 97.56 | 95.42 | 96.23 | 96.89 |
| | **ZOA-CNN** | 98.02 | 98.26 | 98.73 | 96.69 | 98.17 | 98.68 | 96.69 | 97.88 | 98.67 | 96.69 | 97.52 | 98.51 | 97.17 | 97.98 | 98.10 |
| | **CSZOA-CNN** | 99.13 | 98.93 | 99.35 | 98.55 | 98.87 | 99.32 | 98.55 | 98.69 | 99.31 | 98.55 | 98.47 | 99.23 | 98.76 | 98.76 | 99.02 |
| **Time (Sec)** | **CNN** | 92.20 | 89.72 | 91.81 | 85.71 | 85.09 | 86.88 | 83.84 | 82.31 | 84.26 | 80.24 | 78.81 | 81.09 | 81.59 | 81.25 | 83.11 |
| | **ZOA-CNN** | 70.04 | 69.16 | 68.95 | 65.41 | 64.86 | 64.64 | 62.95 | 63.31 | 62.56 | 60.66 | 59.67 | 60.04 | 62.07 | 60.74 | 60.83 |
| | **CSZOA-CNN** | 50.27 | 49.81 | 49.69 | 46.32 | 46.34 | 47.08 | 43.99 | 43.46 | 43.77 | 41.01 | 41.74 | 40.97 | 40.97 | 40.09 | 39.86 |

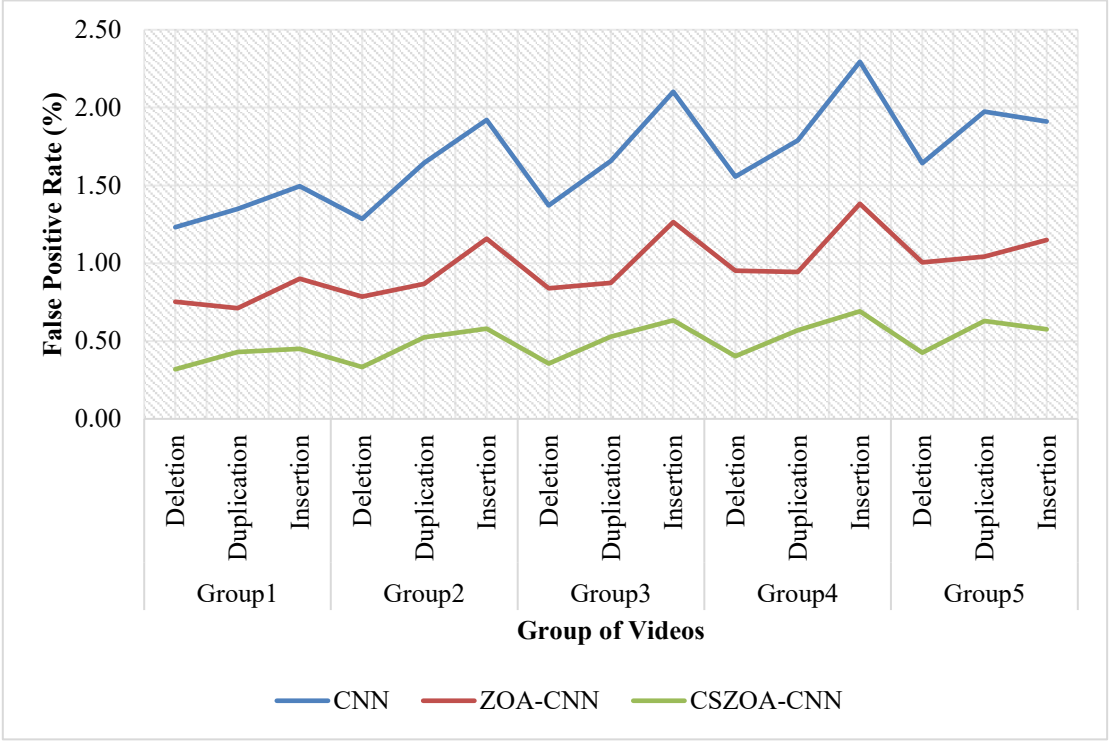


**Figure 2:** Graph of False Positive Rate against Group of Videos based on CNN, ZOA-CNN and CSZOA-CNN

The baseline CNN model recorded the highest false positive rates (1.23%–2.29%), indicating a greater tendency to misclassify authentic videos as forged. Incorporating the Zebra Optimization Algorithm (ZOA) reduced these rates to 0.71%–1.38%, confirming its effectiveness in enhancing CNN hyperparameter selection. This observation is consistent with Kaur and Singh (2021), who noted that metaheuristic algorithms like ZOA can significantly boost CNN performance by identifying optimal hyperparameter configurations. The CSZOA-CNN model achieved the lowest false positive rates (0.32%–0.69%), demonstrating the superior discrimination capability provided by integrating the Chaotic Sinusoidal map with ZOA.

**Overall Accuracy**

Figure 3 summarizes the overall accuracy of the three models across all video groups and forgery types. CNN records the lowest accuracy (97.38%–98.24%) with a downward trend from Group 1 to Group 5. ZOA-CNN performs consistently better (98.49%–98.97%), while CSZOA-CNN achieves the highest and most stable results (99.14%–99.51%). These outcomes validate that incorporating chaotic maps into metaheuristic optimization enhances CNN hyperparameter tuning, with the Chaotic Sinusoidal function enabling more robust generalization across varied forgery scenarios.

**Detection Time**

Figure 4 compares the detection times of the three models. CNN is the slowest (70.00–92.22 s) despite its lower accuracy. ZOA-CNN improves efficiency (60.06–70.04 s), while CSZOA-CNN achieves the fastest processing (39.94–50.22 s) alongside the highest accuracy. The Chaotic Sinusoidal function guides ZOA toward hyperparameter configurations that maximize detection performance while reducing computational complexity, making CSZOA-CNN well-suited for real-time video forgery detection.

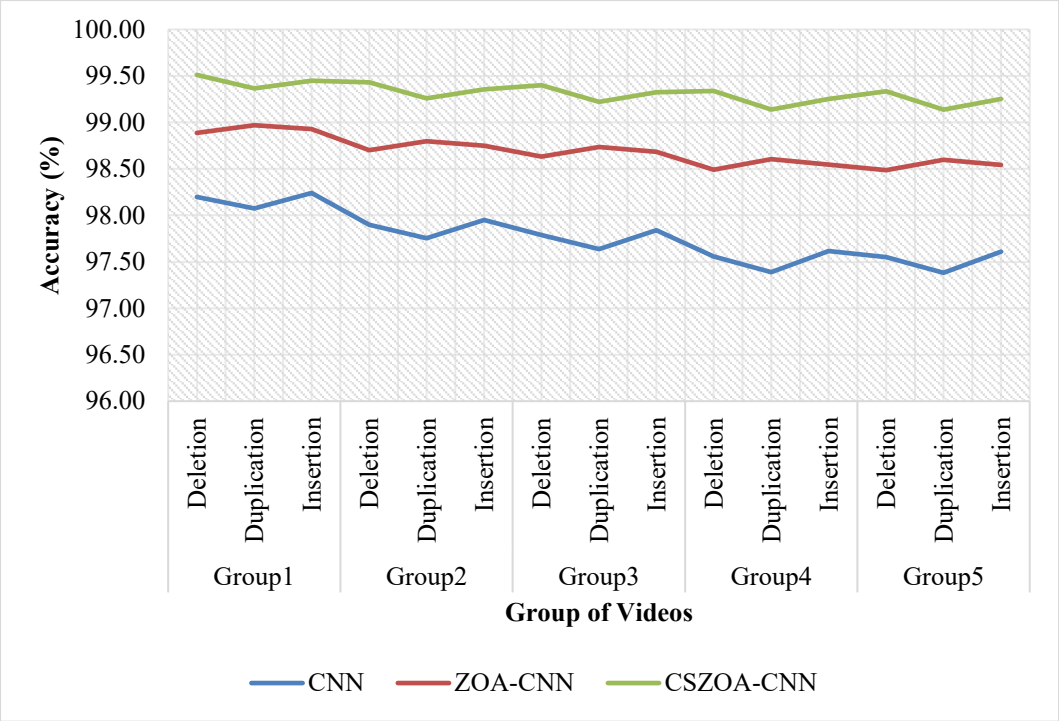


**Figure 2:** Mean activity concentrations of each radionuclide according to sample types

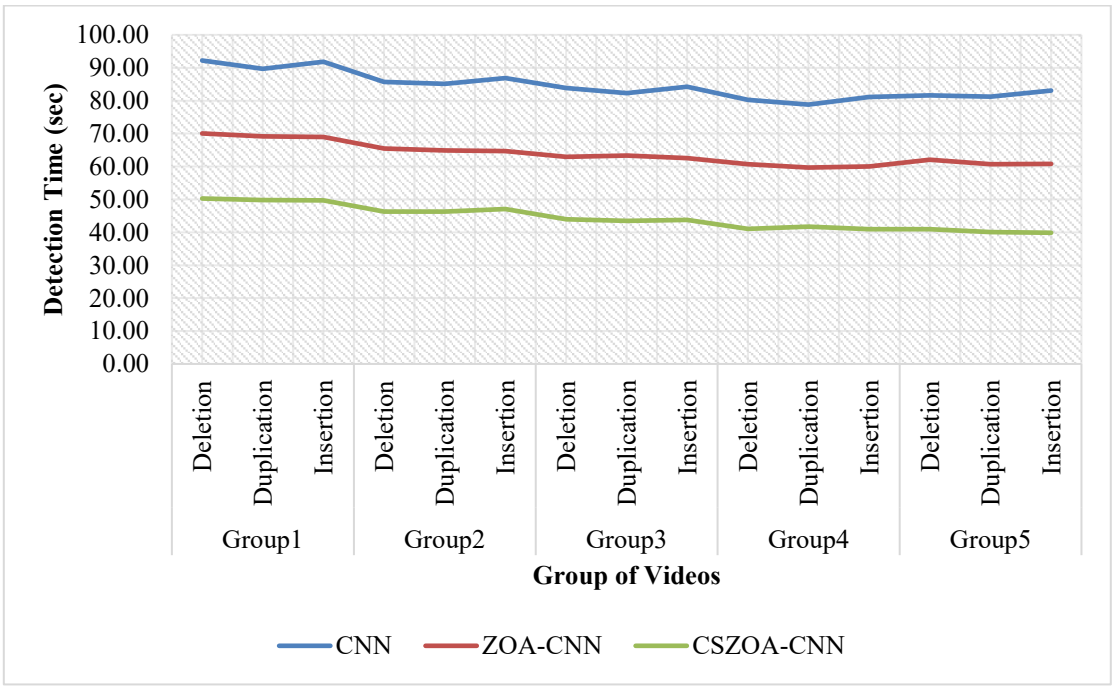


**Figure 4:** Graph of Detection Time against Group of Videos based on CNN, ZOA-CNN and CSZOA-CNN

## CONCLUSION

The CSZOA-CNN model demonstrates strong robustness and efficiency in video forgery detection. By integrating chaotic dynamics into the Zebra Optimization Algorithm, it significantly improves CNN training, achieving higher detection accuracy, reduced false positive rates, and lower computational cost. Future research could investigate alternative chaotic maps, extend deployment to real-time detection environments,

and enhance model interpretability through attention mechanisms or advanced visualization techniques.

## REFERENCES


Afchar, D., Nozick, V., Yamagishi, J., and Echizen, I. (2018). MesoNet: A compact facial video forgery detection network. *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, 1–7. https://doi.org/10.1109/WIFS.2018.8630761

Ahmed, H., and Sonuç, E. (2023). Rationale-augmented convolutional neural networks for interpretable deepfake detection. *Journal of Visual Communication and Image Representation, 90*, 103747. https://doi.org/10.1016/j.jvcir.2023.103747

Ciftci, U. A., Demir, I., and Yin, L. (2020). FakeCatcher: Detection of synthetic portrait videos using biological signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 42*(11), 2823–2837. https://doi.org/10.1109/TPAMI.2019.2926746

El Rai, A., Taha, T., Taha, M., and Elmarakbi, A. (2020). Video forgery detection using noise-based convolutional neural network. *Journal of Visual Communication and Image Representation, 71*, 102775. https://doi.org/10.1016/j.jvcir.2020.102775

Gan, W., Zhang, H., Guo, Y., and Liu, W. (2019). Video object forgery detection based on a convolutional neural network. *Multimedia Tools and Applications, 78*(14), 19225– 19243. https://doi.org/10.1007/s11042-019-7434/8

Güera, D., and Delp, E. J. (2018). Deepfake video detection using recurrent neural networks. *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–6. https://doi.org/10.1109/AVSS.2018.8639163

Jia, Z., Feng, D., and Zhang, X. (2018). Coarse-to-fine copy–move forgery detection for video forensics. *Signal Processing: Image Communication, 68*, 220–233. https://doi.org/10.1016/j.image.2018.07.002

Kaur, G., and Singh, S. (2021). Metaheuristic optimization techniques for a convolutional neural network hyperparameter tuning. Expert Systems with Applications, 168, 114-129.

Kingra, S., and Kaur, G. (2017). Inter-frame forgery detection in digital video using the correlation coefficient and optical flow method. *Multimedia Tools and Applications, 76*(24), 25723–25745.https://doi.org/10.1007/s11042-017-4638-3

Li, Y., Chang, M. C., and Lyu, S. (2018). In Ictu Oculi: Exposing AI-generated fake face videos by detecting the eye blinking. *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, 1–7. https://doi.org/10.1109/WIFS.2018.8630787

Liu, Y., and Huang, T. S. (2017). Detecting video frame duplication with spatial-temporal consistency analysis. *Proceedings of the 2017 ACM on Multimedia Conference*, 376–384. https://doi.org/10.1145/3123266.3123344

Lu, Y., Li, Z., Sun, X., and Zhao, X. (2021). 3D-attentional inception convolutional neural network for video forgery detection. *Neurocomputing, 423*, 1–14.https://doi.org/10.1016/j.neucom.2020.10.059

Nguyen, H. H., Yamagishi, J., and Echizen, I. (2019). Capsule-forensics: Using capsule networks to detect forged images and videos. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2307–2311.https://doi.org/10.1109/ICASSP.2019.8682602

Nguyen, T. T., and Tran, T. (2022). A comprehensive survey on video forgery detection techniques. *Multimedia Tools and Applications, 81*(10), 13673–13707. https://doi.org/10.1007/s11042-022-11989-9

Ogundepo, O. Y., Omeiza, I. O. A., and Oguntoye, J. P. (2022). Optimized textural features for mass classification in digital mammography using a weighted average gravitational search algorithm. *International Journal of Electrical and Computer Engineering (IJECE)*, *12*(5), 5001-5013.

Oguntoye, J. P., Ajagbe, S. A., Adedeji, O. T., Awodoye, O. O., Adetunji, A. B., Omidiora, E. O., and Adigun, M. O. (2025). An Improved

Chicken Swarm Optimization Technique Based on Cultural Algorithm Operators for Biometric Access Control. *Computers, Materials & Continua*, *84*(3): 1-19.

Oguntoye, J. P., Awodoye, O. O., Oladunjoye, J. A., Faluyi, B. I., Ajagbe, S. A., and Omidiora, E. O. (2023). Predicting COVID-19 from chest X-ray images using an optimized convolutional neural network. *LAUTECH Journal of Engineering and Technology*, *17*(2), 28-39.

Olayiwola, D. S., Olayiwola, A. A., Oguntoye, J. P., Awodoye, O. O., Ganiyu, R. A., and Omidiora, E. O. (2023). Development of a Fingerprint Verification and Identification System Using a Gravitational Search Algorithm-Optimized Deep Convolutional Neural Network. *Adeleke University Journal of Engineering and Technology*, *6*(2), 296-307.

Oraibi, A. K., and Radhi, A. M. (2022). An overview of video forgery detection approaches and challenges. *Journal of King Saud University - Computer and Information Sciences,34*(10), 8984–8995. https://doi.org/10.1016/j.jksuci.2022.06.005

Rana, A., Khurana, V., Shrivastava, A., Gangodkar, D., Arora, D., and Dixit, A. K. (2022). A ZEBRA Optimization Algorithm Search for Improving Localization in Wireless Sensor Networks. In *2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS)* (817-824). IEEE.

Saravanan, C. (2010). *Color image to grayscale image conversion*. In *Proceedings of the 2010 Second International Conference on Computer Engineering and Applications*. 2, 196–199. IEEE. https://doi.org/10.1109/ICCEA.2010.192.

Ugale, M., and Midhunchakkaravarthy, J. (2025). An efficient video forgery detection using a two-layer hybridized deep CNN classifier. *EAI Endorsed Transactions on Scalable Information Systems, 12*(1), 1–25. https://doi.org/10.4108/eetsis.4108