I.J. Image, Graphics and Signal Processing, **2024, 1, 38-60**
Published Online on February 8, 2024 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijigsp.2024.01.04

Copy-Move Forgery Detection and Localization Framework for Images Using Stationary Wavelet Transform and Hybrid Dilated Adaptive VGG16 with Optimization Strategy

Prabhu Bevinamarad*
Department of Computer Science and Engineering, BLDEA's V.P. Dr. P.G. Halakatti College of Engineering and Technology (Affiliated to Visvesvaraya Technological University, Belagavi-590018, Karnataka), Vijayapura, Karnataka 586103, India
E-mail: prabhubev@gmail.com
ORCID iD: https://orcid.org/0000-0001-6266-7714
*Corresponding Author

Prakash Unki
Department of Information Science and Engineering, BLDEA's V.P. Dr. P.G. Halakatti College of Engineering and Technology (Affiliated to Visvesvaraya Technological University, Belagavi-590018, Karnataka), Vijayapura, Karnataka 586103, India
Email: prakashhunki@gmail.com
ORCID iD: https://orcid.org/0000-0003-3142-014X

Padmaraj Nidagundi
Computer Science and Information Technology, Riga Technical University, Riga, Latvia
Email: padmaraj.nidagundi_1@rtu.lv
ORCID iD: https://orcid.org/0000-0003-1521-3204

Received: 24 July, 2023; Revised: 26 August, 2023; Accepted: 10 September, 2023; Published: 08 February, 2024

Abstract: Due to the availability of low-cost electronic devices and advanced image editing tools, changing the semantic meaning of a particular image has become straightforward by employing various image manipulation techniques like image copy-move, image splicing and removal operations. The tampered images with this sophisticated software are rich in visualization, making the modifications invisible to the naked eye. Detecting these image alterations is laborious, time-consuming, and often yields inappropriate results. The current techniques use conventional square, slide regular, and artifacts procedures to identify image deviations to combat image forgery practices. Still, these techniques exhibit problems related to generalization, training and testing, and model complexity. So, in this paper, a novel image forgery detection and localization framework is implemented using stationary wavelet transform (SWT), and a Hybrid Dilated Adaptive VGG16 model with optimization is introduced to classify forgery images and localize the forgery regions present in an image. Initially, the proposed framework processes the input image with SWT to decompose an image into different subband and further divide it into patches. After that, the hybrid dilated adaptive VGG16 Network (HDA-VGG16Net) is built to extract the deep image features from the patches. Later, the Hybridized Tuna Swarm with Bald Eagle Search Optimization (HTS-BESO) technique is applied to optimize the VGG16 parameters. Finally, feature matching is formed using multi-similarity searching to recognize whether the input image is forged or original by locating forgery regions. The evaluation results are compared with existing forgery detection approaches to ensure the efficiency of the developed model by considering multiple performance measures.

Index Terms: Copy-move forgery; Stationary wavelet transform; Dilated CNN; Tuna swarm optimization; Bald eagle search optimization

Abbreviations

DWT	Discrete Wavelet Transform
DCT	Discrete Cosine Transform
SWT	Stationary Wavelet Transform

SVD	Singular Value Decomposition
FMT	Fourier-Mellin Transform
BRISK	Binary Robust Invariant Scalable Keypoints
SURF	Speeded-Up Robust Feature
SIFT	Scale Invariant Features Transform
CNN	Convolution Neural Network
LIPO	Local Intensity Order Pattern
GAN	Generative Adversarial Network
LSTM	Long-Short Term Memory
PRNU	Photo-Response Non-Uniformity
BDP	Boundary to Pixel Direction
A-EHO	Autoregressive Elephant Herding Optimization based Generative Adversarial Network

1. Introduction

Introducing the latest image technologies in digital processing and hardware capabilities caused many individuals to apply the changes in the original images and tamper without leaving visual clues[1]. When produced as authentic documents or evidence, the tampered images may cause various problems in the medical domain and courtrooms believing that the photos tampered with using low-cost, sophisticated image editing tools will leave some visual clues surrounding the tampered regions, several authentication techniques are employed to determine the authenticity of the images. These authentication techniques are divided into two categories: active and passive[2]. In the former method, the signature code is provided in picture capturing and used to test modified images. In this case, the multimedia information has a digital mark on it[3]. As a result, modifications to multimedia content are easily detected. However, only some image-capturing gadgets have a digital signature code. Therefore this approach is only sometimes appropriate. Latter, also called forensics, do not require additional data related to the image. Thus, their objective is to determine an image’s authenticity by studying it and looking for signs of any special processing it may have undergone. The image forgeries[4] can be done in different ways, such as copy-move image forgery, in which copying and pasting one or more regions of an image into the same image at various locations are performed to significantly alter the semantic content of the target image and concealing the information or duplicating objects and people. In contrast to copy-move forgeries, this one uses portions of other photos that have been taken out to create the pasted regions and objects. To hide some content or provide a fictitious context, splicing forgeries might be performed. A portion or "hole" in the image is filled with believable content in this type of attack. Usually, in-painting is used to repair damaged areas in pictures. However, potential attackers may use it maliciously to remove a visible watermark or conceal information from an image. Furthermore these tampering techniques use post processed by applying different operations such as contrast enhancement, blurring, and compression to make it hard for the detection systems.

Researchers have made several efforts in recent years to combat forgery practices to detect image forgeries by incorporating conventional forgery detection techniques and deep learning. The section-II will discuss the most current and prominent approaches that have evolved during recent years

The proposed deep learning-based forgery detection mechanism comprises the following key contributions

1. The suggested approach employs state-of-the-art deep networks and parameter optimization mechanisms to prevent image forgery and establish a cost-effective tool for forgery detection.
2. Using SWT for image decomposition helps to identify the most similar and discriminative image characteristics. Also, optimizing parameters like start level, wavelet type, and norm using HTS-BESO algorithm optimization technique enhance decomposition performance and improve the performance of the Adaptive Forgery Identification and Localization Framework.
3. The Hybrid dilated VGG16 network model employs convolution kernels with varying dilation rates across its layers. This design guarantees full coverage of a square area, eliminating gaps or missing information through convolution operations. Consequently, the model efficiently captures the most pertinent features from image patches, enhancing accuracy during training and testing while reducing time consumption.
4. We conducted a comparative analysis to assess the effectiveness of our proposed method, which utilizes the HTS-BESO-HDA-VGG16 Net-based forgery detection scheme. We evaluated its performance against various existing algorithms and recently developed techniques.

Here is a brief explanation of the paper’s reminder section. Section 2 covers the associated research for the forgery detection techniques. Section 3 details the proposed adaptive forgery detection and localization framework. Section 4 discusses the experimental setup, evaluation parameters, results and discussions of the proposed forgery detection method. Section 5 concludes the proposed HTS-BESO-HDA-VGG16Net-based forgery detection technique.

2. Literature Survey

Over the years, researchers have proposed various techniques for determining whether an input image has been forged, emphasizing forged areas. According to the literature review, the first block-based method for identifying forgery images was put forth in 2003[5]. They later developed various strategies using different feature extraction and matching methodologies. They most recently applied deep learning models to enhance detection outcomes and lessen the computational burden of detecting fake photos. Therefore this section summarizes the most recent methods based on conventional feature extraction and matching techniques and the deep learning models developed in recent years.

2.1 Conventional forgery detection techniques

These methods are primarily classified as block-oriented and key-point-oriented. In the former, an entire image is divided into irregular or regular overlapping shaped blocks and processed to extract significant features from each image block, followed by some block matching algorithm to isolate the forged blocks. In the latter, correlated pixels are discovered by removing the image's key points of significance.

In the paper, [6] proposed a technique based on Gaussian-Hermite Moments (GHM), which divides an input image into constant-sized overlapping blocks, and then the Gaussian-Hermite moments are extracted from each block. Later, the lexicographical sorting and matching process is used to find similar blocks. The forgery detection technique proposed in the paper[7] combines SURF and BRISK descriptors. A similar work in the paper[8] combines block-based and keypoint-based techniques using FMT and SIFT for forgery detection. The paper[9] introduces a method where features are extracted and compressed using a combination of DCT and SVD techniques. These features are then used for training support vector machine (SVM), enabling the identification of forgery regions within an image by applying the K-means machine learning technique. The work referenced in [10] identifies both types of forgeries, such as splicing and copy-move simultaneously, by extracting features using DCT, DWT, and Ensemble classifier to classify the input images as forged or authentic.

Further, it utilizes a key point-based method to detect the forgery regions. The paper in [11] presents a comprehensive approach for investigating JPEG compressed test images that are suspected of being altered through splicing or copy-move forgery and localizing the tampered region in forged images. The paper [12] proposes a technique that includes double matching and region-localizing processes for filtering isolated keypoint pairs, finding approximate suspicious parts, and localizing copy-move areas. The paper [13] introduces a frequency-domain image manipulation method that utilizes DWT and inverse DWT to identify the region within the host image that will be manipulated.

These techniques effectively locate the tampered areas even though the tampered regions are post-processed. However, the algorithm did not do any further robustness tests, and the size of the feature vectors increased the computational complexity. So, deep learning frameworks are evolved to overcome the difficulties identified in the conventional forgery detection techniques to achieve higher classification and detection accuracy rates

2.2 Deep learning models

The prior research has looked at residual patterns, wavelet transform, statistical properties, image pixel information, and other image attributes to verify the authenticity of an image. Due to the growing popularity of deep neural network frameworks recently, some efforts have been made to incorporate CNN to design deep learning models to improve forgery image detection. In paper,[14]the present paper investigates copy-move forgery detection with fusion comprising of a deep convolutional and an adversarial model. The proposed algorithm enables constantly update its learning via training data to differentiate areas from forged ones. The paper[15] presents a novel approach for detecting and locating non-aligned JPEG forgery. The method utilizes a deep neural network to perform semantic pixel-wise segmentation of JPEG blocks. The paper, [16]A high-confidence tampering localization structure is proposed by utilizing Long Short-Term Memory (LSTM) cells and an encoder-decoder network to distinguish tampered regions from un-tampered ones based on resembling features. In 2020, Lin and Li[17] investigated a segmentation-based forgery detection technique. This method focuses on the local uniformity of visually hidden clues, which helps overcome the limitations of existing segmentation methods that rely solely on visually perceptible content. Additionally, they proposed a forgery localization method based on PRNU.In the paper[18], CNN is used to train hierarchical features represented from an input image for identifying altered and original images. In the paper[19], proposed two approaches, a model using a custom architecture and a model using transfer learning to distinguish between altered and original images. The paper[20], proposes a method based on color illumination, deep CNN, and semantic segmentation is employed to detect and localize image forgeries. The paper[21] introduces a deep learning approach with a dual-branch CNN to detect passive copy-move forgery by extracting multi-scale features using various kernel sizes. An automated deep learning-based fusion model is implemented in[22] by combining GANs and densely connected network (DenseNets) models to create a layer for encoding the input vectors with the initial layer of the extreme learning machines(ELM) classifier along with artificial fish swarm technique to adjust parameters to distinguish between the input and target areas in a fake image. In 2022, Ganeshan*et al.*[23]Offered a GAN based on Autoregressive Elephant Herding Optimization (A-EHO-based GAN) for copy move forgery detection. The paper [24]presents a lightweight model constructed using mask R-CNN with MobileNet V1 for detecting and identifying copy move and image splicing forgeries present in an image along with corresponding percentages. In 2022,Koul*et al.*[24], proposed convolutional neural network with 3-Layered CNN algorithm for feature extraction and classification. Table 1 summarizes the methodology, features and future challenges of the current forgery detection techniques.

Table 1. Summarization of methodology, features and challenges of previous forgery detection systems

Author [citation]	Methodology	Features	Challenges
M. Bilal et al. [7]	DWT, SURF and BRISK	• Identify single and multiple tampered regions that are processed with post processing operations.	• Complex post-processing attacks such as large scaling, smoothening, and brightness change.
K. B. Meena and V. Tyagi[8]	FMT and SIFT	• Capable of detecting forged areas scaled by a factor of 50% to 200% and subjected to JPEG compression.	• Unable to detect copy-move forgery regions present in videos.
G. S. Priyanka and K. Singh[9]	DCT and SVD	• Detect copy-move forgery and provides better results against post-processed images.	• Difficult in the detection of small-sized forged regions. • Appropriate clustering by the K-means and selecting optimal threshold values pose significant challenges.
S. P. Jaiprakash et al.[10]	DCT and DWT	• Can detect both spliced and copy-move forgeries simultaneously, exhibits high accuracy across various image formats and demonstrates generalization capability.	• Hard to detect images with varying resolutions, blurring attacks, and multiple forgery regions.
Dua, J. Singh, and H. Parthasarathy[11]	DCT	• Precisely identify duplicated regions within uncompressed, compressed, and doubly compressed images of various formats. • Locate single and multiple manipulated areas with precision, even when similar objects and regions are present.	• Experiences slightly higher rates of false detections, mainly when there is a change in illumination and blurring • Extracting phase congruency features from multiple orientations in the covariance matrix poses a challenge.
Q. Lyu et al.[12]	LIOP keypoints	• Achieves high recall with acceptable precision, and robustness.	• Precision decreases due to the extended triangles and increased number of keypoint pairs resulting from the double-matching process.
T. Qazi, et al.[13]	DWT and Inverse DWT	• Detect copy-move forgery regions using DWT and IDWT features	• The method fails to detect when the patch is not taken from the host image and high JPEG compression.
Y. Abdalla et al.[14]	Fusion of CNN and GAN.	• Detect image with different sized forgeries than the ones used during training.	• Experiences more training time
Bappy*et al.*[16]	CNN-LSTM	• Classify the different types tampering such as object removal, copy-move and splicing and Handles the large dimensional dataset	• Missing useful information during the splicing of large dimensional datasets and affected by noise.
X. Lin and C. T. Li [17]	PRNU	• It highly exploits the local homogeneity from indiscernible clues and able to detect and localize the object insert and object removal forgeries	• Need to investigate potential of image segmentation in other forensic detectors and the combination of different techniques in future work.
M. A. Elaskily et al[18]	CNN	• Able to classify original and forgery images accurately within no time.	• There is a need to address the detection and localization of other digital image forgeries.
Y. Rodriguez-Ortega[19]	Custom model and VGG16 with transfer learning	• Addressed the issue of generalization by training the architecture with only one dataset and evaluated on a variety of datasets instead of training on an extensive dataset.	• There is a need for further exploration in expanding the training dataset, understanding the influence of hyper-parameters on classifier performance, and investigating hybrid techniques that integrate deep learning-based feature extraction with domain transformation.
N. Jindal[20]	super‐BPD segmentation and deep CNN	• Able to accurately detect multiple, rotated, and scaling forgeries with large-scale scaling forgeries and small manipulated image areas. • Incorporates local information features from the shallow network outputs and employs the Atrous spatial pyramid pooling (ASPP) layer to create a feature pyramid to improve the overall performance.	• Detection of small altered regions needs to be improved which exhibits sharp edges and background shadow. • The SD-Net's segmentation module and dual-branch structure have complicated the method.
N. Goel et al.[21]	dual branch convolutional neural network	• Classify the image is original or forged.	• Forgery region detection is need to be addressed in the model. It is important to focus on improving the model's generalization capabilities, particularly in terms of handling varying image sizes and different types of forgeries.
N. Krishnaraj et al.[22]	GANs and DenseNets	• Classify the input image is forged or original • Achieved higher training and validation accuracy	• Unfortunately, training a GAN becomes challenging when the generator and discriminator are highly proficient, as GANs typically require significant training time.

Ganeshan*et al.*[23]	A-EHO based GAN	- It accurately captures the copied image portion in the host image. - Employing the separation operator with clan updating with regressive terms successfully achieves global convergence.	- Currently, the model does not address the detection of forgery regions, and achieving generalization of the model remains a challenge.
Koul*et al.*[24]	CNN	- It provides improved forgery detection accuracy than the other methods and alleviate the derelictions.	- It loses some information during merging. - It is affected by white Gaussian noise.

2.3 Forgery detection challenges in existing systems

With widespread digital devices and freely available open-source and commercial image editing tools, image modification poses a significant challenge in visual media. Due to this, an authenticity of an image becomes questionable. Moreover, it can lead to misleading testimonies within a court of law and poses a challenge because human eyes cannot discern alterations made to an original image due to its tiny nature. Also, the localization process is crucial for locating the places that have been altered. As a result, several conventional and deep learning methods are evolved to authenticate and detect forgery regions but exhibit several limitations. The preceding techniques carry a more significant burden of computational complexity and yield imprecise outcomes when manipulated images are subjected to a range of post-processing maneuvers, including substantial and subtle scaling, smoothening, and brightness adjustments.

Furthermore, this approach needs to be revised in identifying instances where the altered patch originates from a different source image, and it struggles in cases of high-level JPEG compression, rendering it unsuitable for real-time applications. In contrast, deep learning models focus solely on discriminating between forged and authentic images. Nonetheless, these models necessitate extensive training data and heightened computational capabilities. Despite these requirements, achieving effective model generalization remains an ongoing challenge.

3. Proposed Adaptive Forgery Detection and Localization Framework

Although many conventional and deep learning approaches mentioned above have demonstrated promising results in classifying images as forged or authentic and locating forgery regions, still, they have some drawbacks as mentioned in section 2.3. Furthermore, these current methods are susceptible to fuzzy attacks and necessitate usage in a distinct space, making distinguishing between innocent and malicious retouching difficult. Hence, they need to be revised, including imprecise results, low resilience, and a high rate of false alarms. Therefore, we propose a model called HTS-BESO-HDA-VGG16Net-based forgery detection that offers a solution to improve the training and testing performance and generalize the model for detecting forgeries to overcome some of these constraints. The suggested model can be used by various web applications, government agencies and social media platforms as the back end to verify the legitimacy of each image data provided during transactions. Fig. 1 depicts the visual representation of suggested forgery detection approach.

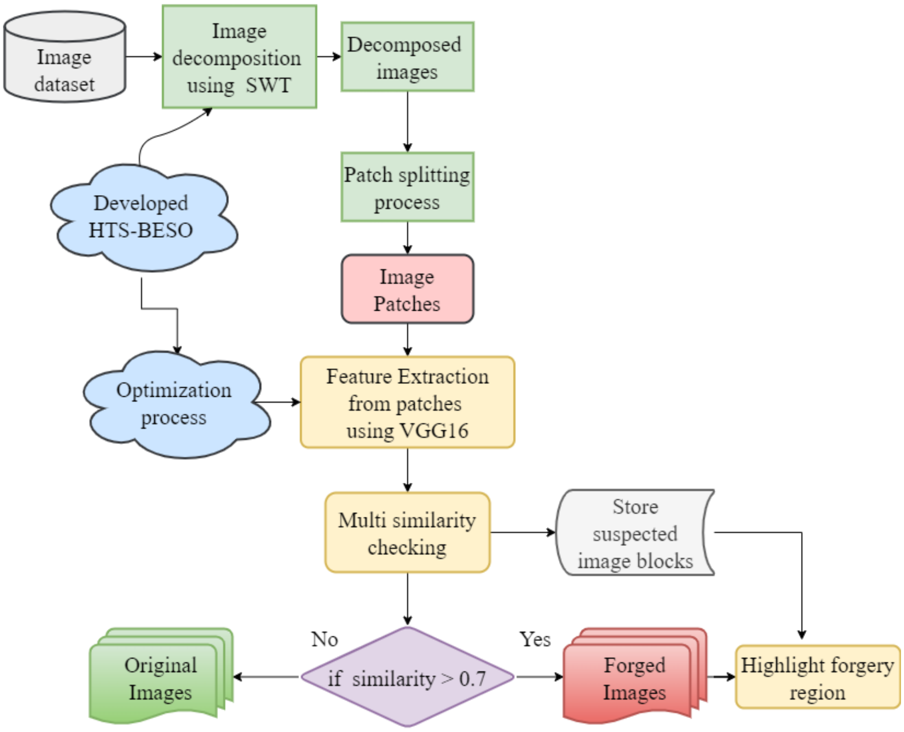


Fig. 1. Visual representation of the proposed forgery detection technique.

In the first step, we gathered the required images from two datasets. Then SWT is applied to decompose each image by optimizing the parameters like start level, wavelet type and norm with the help of the suggested HTS-BESO algorithm. Before the images are decomposed using SWT, all images are preprocessed to reduce the size to 400×400 image.The decomposed images split into several patches. Later the significant image features are extracted from the patch using hybrid dilated adaptive VGG16. The parameters from the VGG16, like hidden neuron count and epochs, are optimized to maximize the accuracy and precision values. Finally, keeping one patch as a constant and a matching process is performed to determine the similarity value for the remaining patches. If the similarity value exceeds 0.7, we consider the image forged and mark the forgery regions in the image; otherwise, we believe it is authentic. Finally, the final results are contrasted with the existing algorithms to evaluate the effectiveness of the offered HTS-BESO-HDA-VGG16Net-based forgery detection scheme. Comprehensive explanations of each step are explained in the following sections, encompassing all essential particulars.

3.1 Image Decomposition using Adaptive SWT

The input images collected from the datasets are indicate $\ln_a^{col}, where\, a = 1, 2, 3, ..., A$ as the total number of images. In the first step, each input image is decomposed by applying adaptive SWT[25]to divide an input image into a few frequency bands. The choice of adaptive Stationary Wavelet Transform (SWT) for image decomposition is founded on its ability to capture both high-frequency and low-frequency details within an image effectively. The adaptive SWT adjusts the filter lengths depending on the scale of analysis, unlike conventional SWT, which employs preset filter banks for decomposition. This versatility enables SWT to capture local image characteristics, improving the representation of structures at different scales and ensuring that essential features are not lost throughout the decomposition process. In the proposed approach, the SWT parameters such as norm, start level, decomposition level, and wavelet type are optimized using HTS-BESO, enabling the retention of significant image details for extracting discriminative features and achieving high forgery detection accuracy and precision. By incorporating the adaptive SWT-based decomposition, we greatly enhance the detection reliability of the proposed model. The adaptive SWT image decomposition process and corresponding mathematical expressions are depicted and illustrated in Fig. 2 and equation (1-6), respectively.

$$d_{k,l} = \sum_{m \in X} y(m) \omega'_{k,l}(m) \tag{1}$$

Here, discrete wavelet is represented as $\omega'_{k,l}(m)$ and its value is determined using equation (2).

$$\omega'_{k,l}(m) = 2^{-(k/2)} \omega_{0,0}(2^{-k}(m-l)), \tag{3}$$

$$\begin{gathered} db_{1.l}(m) = \sum i1(m-\upsilon)y(\upsilon) \\ de_{1.l}(m) = \sum h1(m-\upsilon)y(\upsilon) \end{gathered} \tag{4}$$

The above equations are generalized and written in equation (5).

$$\begin{gathered} db_{k,l}(m) = \left[\uparrow 2^{k-1}[i_1] * db_{k-1.l}\right] = \sum i^k(m-\upsilon) db_{k-1,l}(\upsilon) \\ de_{k,l}(m) = \left[\uparrow 2^{k-1}[h_1] * de_{k-1.l}\right](l) = \sum h^k(m-\upsilon) be_{k-1,l}(\upsilon) \end{gathered} \tag{5}$$

Here, $db_{k,l}$ and $de_{k,l}$ are the approximate and detailed coefficients, respectively. These coefficients are generated with the help of signal sequence and are represented as $y(m)$, i_1 and j_1 is the adaptive size of the high and low pass filter respectively, over sampling of high pass and low pass filter at coefficient values of $i^{k-1}(m)$ and $h^{k-1}(m)$ is indicated as $\uparrow 2^{k-1}[i_1] = i^k(m)$ and $\uparrow 2^{k-1}[h_1] = h^k(m)$ the expression is used to find the values and it is written in equation (6).

$$\begin{gathered} \begin{cases} h^j(2m) = h^{j-1}(m) \\ h^j(2m+1) = 0 \end{cases} \\ \begin{cases} i^j(2m) = i^{j-1}(m) \\ i^j(2m+1) = 0 \end{cases} \end{gathered} \tag{6}$$

The above equation gives very accurate output signals by utilizing adaptive SWT method. The output obtained from this process is denoted as D_b^{SWT} .

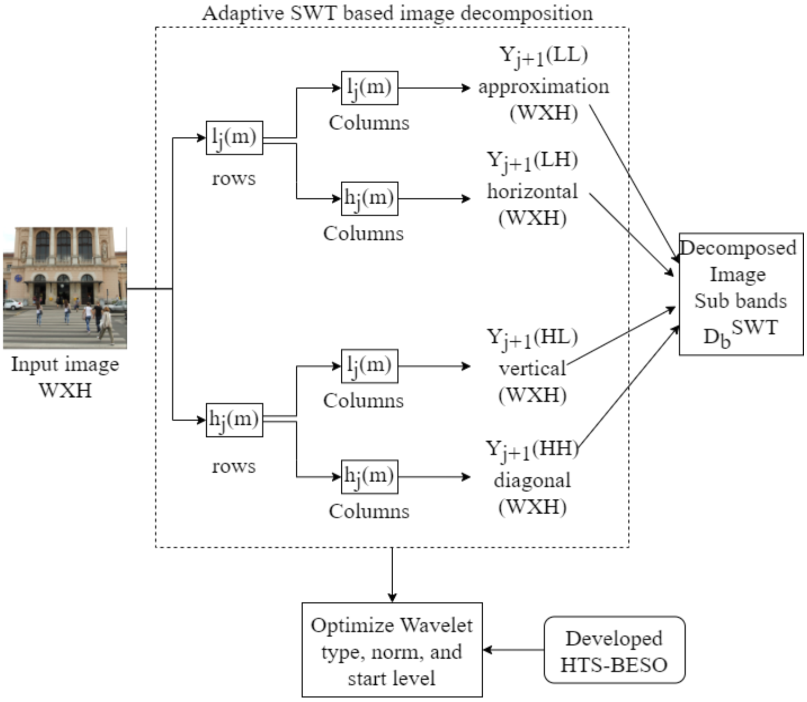


Fig. 2. Illustration of adaptive SWT-based image decomposition

3.2 Optimization Technique Adopted in Image Decomposition and VGG16

We have implemented the HTS-BESO algorithm in a proposed forgery detection scheme. This algorithm aims to optimize the parameters of SWT and VGG16 to enhance the detection performance. The proposed algorithm demonstrates an advantage over alternative swarm-based metaheuristic algorithms by effectively balancing exploitation and exploration to conduct a thorough search and identify the best answer. It also ensures the capacity to converge quickly to optimal or almost optimal solutions. The information-sharing and adaption mechanisms of the algorithm allow it to focus on promising areas of the search space efficiently. In contrast, the BESO algorithm reduces costs and provides high comfort.

Consequently, the HTS-BESO algorithm addresses the limitations of existing algorithms by significantly improving accuracy and maximizing precision in detecting forgeries. To implement the suggested HTS-BESO algorithm, we incorporate an updated uniform random number based on the fitness value. This allows us to obtain the best solution by adjusting the random parameters. The key requirement is to update the candidates' positions, as specified in equation (7).

$$if\ c < \frac{Bf}{Wf} \tag{7}$$

B_f and W_f stand for the best and worst fitness values, respectively, and c is a uniform random number. The conventional method's c weight varies between [0-1]. The suggested strategy provides a highly optimized solution in the problem space under the abovementioned criterion. The HTS-BESO algorithm also enhances the proposed system's rate of convergence.

- ***Tuna Swarm(TSO)***

TSO[26] algorithm is designed based on individual movement and interaction principles within a swarm, enabling it to tackle complex optimization problems effectively. The marine predatory fish known as tuna is officially known as Thunnini. Tuna comes in a variety of species, and they come in a wide range of sizes. Tuna are top marine predators that eat various surface and midwater fish. The fishtail form, which tunas use to swim, is a unique and effective swimming style that keeps the body tight while allowing the long, thin tail to swing quickly. Tunas are continuous swimmers. Despite swimming rapidly, the lone tuna takes longer to react than the nimble little fish. So the tuna will hunt in groups, using the "group travel" strategy. To locate and catch their prey, they employ intellect. Numerous valuable and clever foraging techniques have evolved in these species.

The first technique is Spiral foraging. To lure their prey into shallow water, where they may be more readily attacked, tuna swim in a spiral shape when feeding. Parabolic foraging is the second tactic. Each tuna form a parabolic shape as it swims behind the one before it to encircle its prey. The two approaches mentioned above are effective for tuna foraging. Following is a description of the TSO's mathematical model. Like most swarm-based metaheuristics, TSO initiates the optimization process by uniformly generating beginning populations at random in the search space and it is given in equation (8).

$$Y_j^I = rand(ua - lb), j = 1, 2, 3, ..., MP \tag{8}$$

The rand is used to represent a uniformly distributed random vector and has a range of [0-1]. The upper and lower boundaries of MP are given as ua and lb, respectively. MP is the number of Tuna populations. The initial individual in the j^{th} direction is denoted by $Y_j^I J^{th}$,

Sardines, herring, and other small schooling fish establish a thick configuration when they come into contact with predators, which makes it harder for the predator to lock on a victim. The tuna school forms a tight spiral formation. It begins chasing the prey at this point, even though most of the fish in the school has a poor sense of direction. When a small number swims steadily in a particular sequence, the neighboring fish gradually alter their focus until they eventually form a large group with the same aim and begin to hunt. Tuna schools communicate with one another in addition to spiraling after their prey. Since each tuna follows the one before, information can be shared between nearby tuna. Based on the above principles, the mathematical formula for the spiral foraging strategy is as follows: equation (9).

$$Y_j{}^{u+1} = \begin{cases} \beta_1 \cdot \left(Y_{premi}^u + \chi \cdot \left| Y_{premi}^u - Y_i^u \right| \right) + \beta_2 \ . Y_j^u , j=1, \\ \beta_1 \cdot \left(Y_{premi}^u + \chi \cdot \left| Y_{premi}^u - Y_i^u \right| \right) + \beta_2 \ . Y_{j-1}^u \ , \ j=2,3,...,MP \end{cases} \tag{9}$$

$$\beta_1 = b + (1-b). \frac{u}{u_{\max}},$$

$$\beta_2 = (1-b) - (1-b). \frac{u}{u_{\max}},$$

$$\chi = e^{cm} .\cos(2\pi c),$$

$$m = e^{3\cos((u\max + 1/u) - 1)\pi)},$$

Where $Y_j{}^{u+1}$ j^{th} individual of the $u+1$ iteration, the weight coefficients β_1 and β_2 govern how inclined people are to travel in the direction of the ideal person and the preceding person, respectively. Parameter b determines how closely the tuna follow the ideal person and the preceding individual in the first phase. The current iteration is denoted by u, while the $u_{\max}$ represents the maximum iteration, the current optimal individual is indicated as Y_{premi}^u, the parameter c denotes uniform random number and its ranges in between $[0,1]$.

All tuna can utilize the search space surrounding the food when they forage spirally around it. However, unthinkingly following the ideal individual to feed is not advantageous for group foraging when that person cannot find food. As a result, we consider creating a random coordinate in the search space as a starting point for the spiral search to enable everyone to look for a larger area and offer TSO the power to explore the entire world. The details of the specific mathematical model are as follows in equation (10).

$$Y_j^{u+1} = \begin{cases} \beta_1 \cdot \left(Y_{rand}^u + \chi \cdot \left| Y_{rand}^u - Y_i^u \right| \right) \\ + \beta_2 \, . \, Y_j^u \, , j=1, \\ \beta_1 \cdot \left(Y_{rand}^u + \chi \cdot \left| Y_{rand}^u - Y_i^u \right| \right) \\ + \beta_2 \, . \, Y_{j-1}^u \, , \; j=2,3,...,MP \end{cases} \tag{10}$$

As a result, the random generated reference point in the search space is denoted by Y_{rand}^u .

Metaheuristic algorithms usually start by engaging in sizeable global exploration before progressively shifting to focused local exploitation. As a result, TSO switches the spiral foraging reference points from random individuals to optimal individuals with each subsequent iteration. The final mathematical model summarizes the spiral foraging strategy illustrated in equation (11) as follows.

$$Y_j^{u+1} = \begin{cases} \beta_1 \cdot \left(Y_{rand}^u + \chi \cdot \left| Y_{rand}^u - Y_i^u \right| \right) & \\ + \beta_2 \, . \, Y_j^u \, , j=1, & \\ \beta_1 \cdot \left(Y_{rand}^u + \chi \cdot \left| Y_{rand}^u - Y_i^u \right| \right) & \\ + \beta_2 \, . \, Y_{j-1}^u \, , \; j=2,3,...,MP & if \;\; rand \; < \dfrac{u}{u\max} \\ \beta_1 \cdot \left(Y_{premi}^u + \chi \cdot \left| Y_{premi}^u - Y_i^u \right| \right) & if \;\; rand \; \geq \dfrac{u}{u\max} \\ + \beta_2 \, . \, Y_j^u \, , j=1 & \\ \beta_1 \cdot \left(Y_{premi}^u + \chi \cdot \left| Y_{premi}^u - Y_i^u \right| \right) & \\ + \beta_2 \, . \, Y_{j-1}^u \, , \; j=2,3,...,MP & \end{cases} \tag{11}$$

Tunas also engage in cooperative parabolic feeding in addition to spiral formation feeding. Tuna use food as a point of reference to construct an illustrative structure, using their surroundings as a search area and hunting for food by searching around themselves. To perform these two approaches simultaneously, we assume a selection probability of 50%. And we describe the specific mathematical model as in equation (12) as follows.

$$Y_j^{u+1} = \begin{cases} Y_{premi}^u + rand.(Y_{premi}^u - Y_j^u) & \\ + UG \, . \, q^2 \, .(Y_{premi}^u - Y_j^u), & if \;\; rand < 0.5, \\ UG.q^2.Y_j^u & if \; rand \geq 0.5, \end{cases} \tag{12}$$

$$q = \left(1 - \frac{u}{u_{\max}} \right)^{(u/u_{\max})}$$

Here, UG is a random number and the value is between $[-1,1]$.

Tuna uses two foraging techniques to locate their prey cooperatively. The initial population is generated at random in the search space for the TSO optimization procedure. Each person determines whether to regenerate their location in the search space based on probability ‘x’ or selects one of the two foraging strategies to use randomly in each iteration. The value of the parameter ‘x’ is determined using parameter setting simulation experiments for parameter setting. All TSO members are continually updated and calculated throughout the optimization procedure until the end condition is satisfied. The ideal person is then given back along with the associated fitness value.

- ***Bald Eagle Search Optimization (BESO)***

The BESO[27] algorithm, inspired by the hunting behaviors of bald eagles, captures the sequential nature of each hunting phase. This algorithm can be divided into three stages: search space selection, exploration of the selected search space, and decisive swooping.

Select stage: During the selection stage, bald eagles exhibit a strategic behavior of choosing the optimal hunting location within a defined search region based on the abundance of available food. This behavior is mathematically represented by equation (13).

$$SW_{new}(j)=SW_{best}+\delta.rand.(SW_{mean}-SW(j)) \tag{13}$$

Where δ is a control parameter for regulating positional changes and ranges between $[1.5, 2]$ and rand is a random number with a range of values between 0 and 1. The term SW_{best} refers to the area of the search that bald eagles have chosen based on the best location they discovered during their previous quest. The $SW_{new}(j)$ signifies new position and Eagles randomly search all locations near the pre-selected search zone. While doing so, SW_{mean} shows that these eagles have consumed all of the information from the earlier points.

Search stage: Bald eagles maneuver in a spiral pattern within the chosen search zone during the search stage to speed up their quest for prey. Equation 14 mathematically expresses the ideal swoop position.

$$SW_{np}(j) = SW(j) + o(j).(SW(j) - SW(j+1)) + n(j).(SW(j) - SW_{m}) \tag{14}$$

Here, directional coordinates for j^{th} position is signified by o , m parameters.

$$\begin{gathered} n(j) = \frac{n\,rand(j)}{\max(|\,n\,rand\,|)};nrand(j)=rand(j).\sin(\zeta(j)) \\ o(j) = \frac{o\,rand(j)}{\max(|\,o\,rand\,|)};o\,rand(j)=rand(j).\cos(\zeta(j)) \\ \zeta(j)=b.\pi.rand;rand(j)=\zeta(j).S.rand \end{gathered} \tag{15}$$

S is a parameter that accepts a value between 0.5 and 2 to represent the number of search cycles, and b is a parameter that takes a value between 5 and 10 to represent the corner between the point searches in the central point.

Swooping stage: Bald eagles swing to their intended prey in the swooping stage from the best location in the search area. Every point moves in the direction of the ideal time. This behavior is represented quantitatively in equation (16).

$$SW_{np}(j)=rand.SW_{premi}+n1(j).(SW(j)-d1.\ SW_{m})+o1(j)\ .(SW(j)-d2.SW_{premi}) \tag{16}$$

In this context, the random number is denoted as d1 and d2, respectively, while the directional coordinates are indicated as n1 and o1. Finally, the values of the directional coordinates are calculated using equation (17).

$$\begin{gathered} n1(j) = \frac{n\,rand(j)}{\max(|\,n\,rand\,|)};nrand(j)=rand(j).\sin i(\zeta(j)) \\ o1(j) = \frac{o\,rand(j)}{\max(|\,o\,rand\,|)};o\,rand(j)=rand(j).\cos i(\zeta(j)) \\ \zeta(j)=b.\pi.rand;rand(j)=\zeta(j) \end{gathered} \tag{17}$$

Here, directional coordinates are denoted by m and o , respectively, and b is signifies a control parameter. Pseudocode 1 provides a visual representation of the algorithmic steps employed in the proposed HTS-BESO method.

Pseudocode 1: HTS-BESO algorithmic steps

```
Set the initial population size as $N_p$
Set maximum iteration value $U_{max}$
Compute the value of $B_f$ and $T_f$ then
Calculate the value of uniform random number $C$
for t=1 to $U_{max}$
    for j=1 to $N_p$
    $ifc < B_f/W_f$
```

```
                Discovery the value of $Y_{rand}^{u}$
                Update the position using the TSO method mentioned in equation (10).
            Else
                Find the value of best position
                    Update the position using BESO by equation (13)
            end if
        end for
end for
```

3.3 *Block tilig process*

Here, the decomposed image D_b^{SWT} is input to the block tiling process to divide an image into blocks. The image block is a small rectangular box composed of several pixels to help the image processing algorithms easily carry the feature extraction process to identify the similarity values in detecting the forged images with high accuracy and reduce the computational complexity. Initially, we mark the seed points to divide an image into different blocks. After that, we find the movement of this point using the slight gradient. We must increase the number of seed points to separate the image into small patches. The decomposed images are split into nine non-overlapping patches of size 128 x 128denoted as P_C^{sp} .The developed forgery detection model employs a non-overlapping block tiling process to reduce the redundancy and higher computational complexity during feature extraction and help to capture more global features and textures in an image. Fine details can be captured from individual patches during the feature extraction process from hybrid dilated VGG16 later. The smaller block sizes can capture finer details but might increase storage requirements and introduce block artifacts.

3.4 *Hybrid Dilated Adaptive VGG16 Network for Forgery Detection and Localization*

In many traditional image classification algorithms, the conventional CNN plays a vital role. However, in these traditional CNN models, increasing the network depth by stacking layers is often employed to achieve higher accuracy and handle more complex scenarios. Unfortunately, this approach results in excessive computational resource consumption and can lead to the vanishing gradient problem, where gradients become extremely small as they propagate through the network layers. Consequently, the performance of the network may saturate or even deteriorate significantly. To address these challenges, a dilated CNN model is introduced as an alternative to the convolution layers of traditional CNN models. In this approach, dilated convolution layers, which include holes within the convolution kernels, are utilized. This strategy helps reduce the computational resource requirements during feature extraction while expanding the receptive field without increasing the number of parameters[28]. However, simply stacking dilated convolution kernels can expedite the training process and improve training accuracy to some extent. Nevertheless, it does not effectively enhance testing accuracy because dilated convolution kernels can result in the omission of specific pixels, leading to the potential neglect of continuity information within the image. Additionally, when the size rate remains constant, it becomes challenging to consider both large and small-scale information simultaneously while extracting the image's feature map.

Therefore, we have employed hybrid dilated convolution(HDC) layers[29] to process images and ensure that the vital information won't be lost to the greatest extent when extracting the feature map. In this process, the image patches P_C^{sp} obtained from block tiling process are given as an input to HDC-VGG16Net to extract relevant image features and incorporate the HTS-BESO for parameter optimization to efficiently detection the forgery images. The dilated convolution kernels in the HDC model have different dilation rates in the various layers. Instead of using the same dilation rate for all layers after down sampling, we use a different dilation rate for each layer. The dilation rates in the dilated convolution kernels are set as 1, 2, and 5 in one set and 1, 2, and 5 in another set, respectively, to cover every pixel point in the image and preserve critical information during a series of convolution operations—the Fig. 3 Illustrates dilated convolution kernel stacking effect that contributes to the calculation of the center pixel marked as a red through three convolution layers with kernel size 3 x 3. (a) All convolution layers have a dilation rate r = 2. (b) Subsequent convolution layers have dilation rates of r = 1, 2, 3, respectively. It can be seen that (a) always leaves some holes between pixels, and these holes, with different dilation rates, every single step in the HDC model can fill the gaps. Hence the top layer can access information from a broader range of pixels in the same region as the original configuration. This process is repeated through all layers, thus making the receptive field unchanged at the top layer. We constructed the HDC model with six dilated convolution-pooling modules, two fully connected layers, and a softmax function. To prevent overfitting, we optimized the model using a two-layer dropout function.

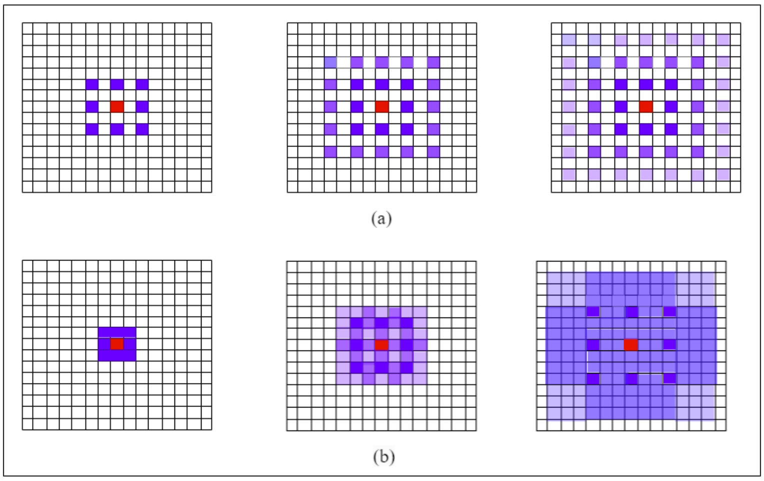


Fig. 3. Dilated convolution kernel stacking effect[30]

Hence the designed HDC model effectively retrieves the most relevant features from the image patches, improves accuracy in training and testing, and reduces time consumption. As a result, it maximizes accuracy and precision values, thereby increasing the system's overall effectiveness. We represent the objective function of the proposed deep learning-based forgery detection scheme as equation (18).

$$FN^1 = \underset{\left\{SWT^T, SWT^L, SWT^{SL}, SWT^N, VGG16^{Hn}, VGG16^e\right\}}{\arg\min} \left(\frac{1}{D^A + PPV}\right) \tag{18}$$

Thus, accuracy and precision is represented as D^A, and PPV respectively, $SWT^T, SWT^L, SWT^{SL}, SWT^N$ are the optimized parameters of SWT and they are wavelet type and it is ranges between $[0,4]$, level and its value is ranges between $[5,15]$, start level and the value is ranges between $[1,4]$, norm and its value is ranges between $[0,1]$ and $VGG16^{Hn}, VGG16^e$ are the optimized parameters of VGG16 and they are hidden neuron count its value is ranges between $[5,255]$ and epochs and its value is range in between $[5,50]$ respectively. The value of accuracy and precision also called as positive predictive value (PPV) is determined by using equation (23) and equation (24).The block diagram of hybrid dilated adaptive VGG16-based forgery detection in Fig. 4.

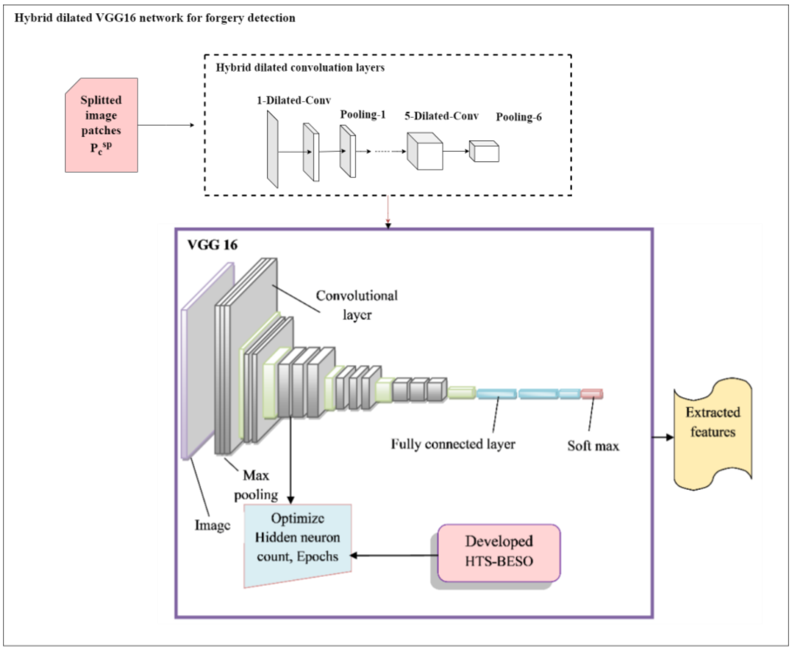

Hybrid dilated VGG16 network for forgery detection
Splitted image patches P_c^{sp}
Hybrid dilated convoluation layers
1-Dilated-Conv
Pooling-1
5-Dilated-Conv
Pooling-6
VGG 16
Convolutional layer
Fully connected layer
Soft max
Extracted features
Max pooling
Image
Optimize Hidden neuron count, Epochs
Developed HTS-BESO

Fig. 4. Block diagram of Hybrid dilated VGG16-based forgery detection process

3.5 Multi-Similarity-based Image Localization

We use a multi-similarity analysis technique to identify similar regions in images. When copy-move image tampering occurs, the copied parts do not need to be precisely positioned where they should be, resulting in altered blocks that do not intersect with the duplicated region. After splitting the image patches, we adopt a multi-similarity analysis by keeping one patch constant and comparing the others to find similar image patches. If the similarity value exceeds 0.7, we consider the image patch altered; otherwise, it belongs to the original image. So, this process allows us to separate the forgery and original image patches precisely. Finally, to visualize the resulting image with the discovered forgery region, we apply binary mapping techniques and additional morphological operations like opening and closing to remove isolated image patches. Fig. 5 presents a Multi-similarity-based image localization process for highlighting forgery regions.

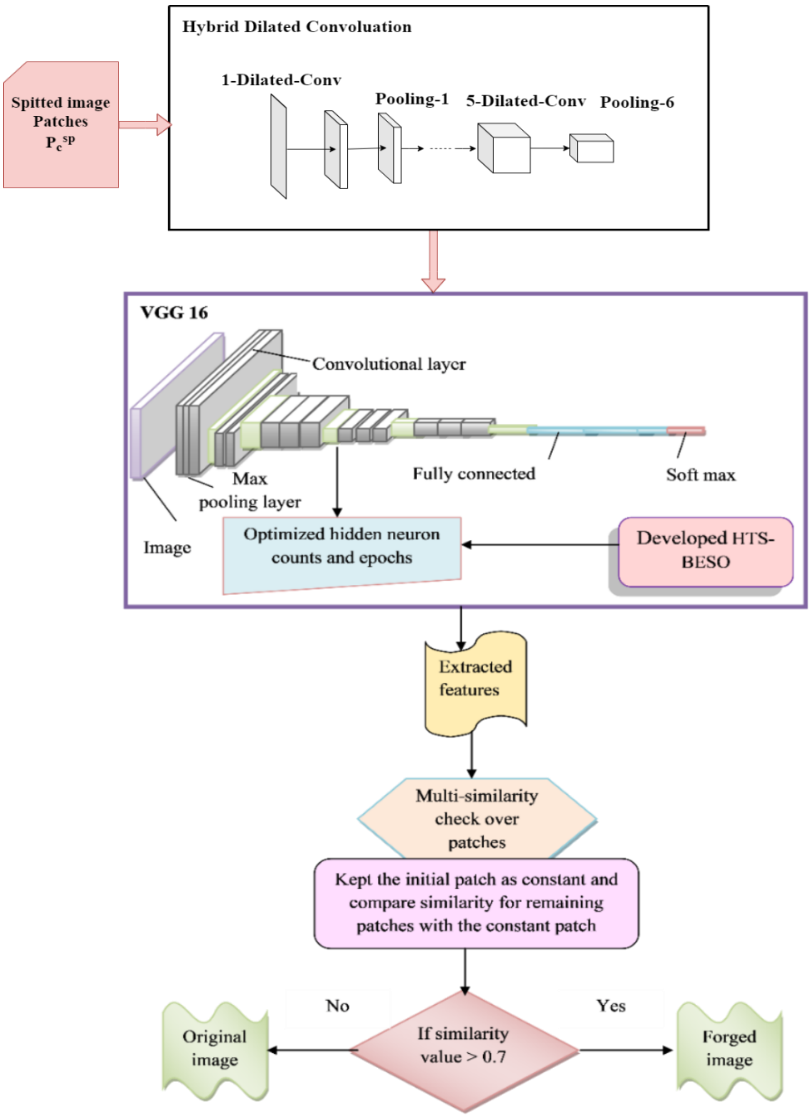

Hybrid Dilated Convolution
Spitted image Patches P_c^{sp}
1-Dilated-Conv
Pooling-1
5-Dilated-Conv
Pooling-6
VGG 16
Convolutional layer
Max pooling layer
Fully connected
Soft max
Image
Optimized hidden neuron counts and epochs
Developed HTS-BESO
Extracted features
Multi-similarity check over patches
Kept the initial patch as constant and compare similarity for remaining patches with the constant patch
No
Yes
If similarity value > 0.7
Original image
Forged image

Fig. 5. Multi-similarity-based image localization process

4. Experimental Setup

The proposed forgery detection model HTS-BESO-HDA-VGG16Net, was implemented using Python 3.8 with Keras and TensorFlow as the backend toolkit for training and testing. The Keras default function is used to initialize the parameters of all layers. We use the Adam optimizer with a learning rate 0.01 and the binary cross-entropy loss function. The epoch and mini-batch sizes are set to 90 and 30, respectively. The proposed framework is tested on an Intel Core i7 64-bit processor with 16 GB RAM and an 8GB GPU. To assess the efficiency of the proposed method, we have set the population and maximum iteration value as 10 and 50, respectively. To evaluate the proposed approach, we have used two publicly available benchmark datasets employing various evaluation parameters and the results were compared with those of existing algorithms. The comparative study was performed by considering the Jaya(JA)[31], Deer Hunting Optimization (DHOA)[32], TSO[26], and BESO[27] algorithms and different deep learning techniques. The subsequent subsection provides detailed information about the dataset, evaluation metrics and analysis of the results.

4.1 Dataset description and performance evaluation metrics employed

The presented approach for detecting forgeries utilizes two diverse benchmark datasets, CMFD[33]and CoMoFoD[34], to perform an experiment to detect and localize forged content. These datasets consists of large image samples with real-world scenarios and the diversity of forgeries encountered (plane image forgery with post-processing). The CMFD, comprised of medium-sized BMP format images measuring 1000x700 or 700x1000 pixels, is divided into different subsets (D0, D1, D2, and D3). D0 consists of 50 images that are exact translations of the original images and corresponding binary masks indicating the source and destination areas of tampering. D1 contains tampered images created by copy-pasting objects after rotation, while D2 comprises tampered images obtained through scaling. Finally, D3 includes 50 original, untampered images. Hence, it encompasses 1,210 images of originals, tampered versions, and tampered images subjected to post-processing attacks. On the other hand, the CoMoFoD dataset comprises 200 primary images with a resolution of 512X512 pixels. These photographs are categorized based on four types of geometric alterations: translation, rotation, scaling, and distortion. Each category consists of 40 pictures. Within each category, six subcategories correspond to six different post-processing techniques that can be applied to an image: JPEG compression, noise addition, image blurring, brightness alteration, color reduction, and contrast changes. Considering all geometric alterations and post-processing methods, the CoMoFoD dataset encompasses 10,400 image samples. The comprehensive description of these datasets are tabulated in the table as follows.

Table 2. Comprehensive description of these datasets

Dataset	**Post processing methods**	**Parameters and corresponding values**
CMFD	Rotation	(Range, step)= [((-25 °, 25 °), 5 °),((0 °, 360 °),30 °), ((-5 °, 5 °), 1 °))]
	Scaling	(range, step)=[((0.25, 2), 0.25), ((0.75, 1.25), 0.05)]
CoMoFD	Post processing methods	Parameters and corresponding values
	Jpeg Compression	Factors=[20,30,40,50,60,70,80,90,100]
	Noise adding	μ=0, σ2=[0.009, 0.005,0.0005]
	Image blurring	Average filter=[3x3,5x5,7x7]
	Brightness change	(lower bound, upper bound)=[(0.01,0.95),(0.01,0.9),(0.01,0.8)]
	Contrast adjustment	(lower bound, upper bound)=[(0.01,0.95),(0.01,0.9),(0.01,0.8)]

The model is validated on an image level by considering both forgery images and post-processed with JPG compression, image blurring, rotation and scaling from both datasets to practical applicability and generalization to ensure whether or not the model can accurately classify forged and original images. Once the fake image has been accurately identified, we use the block similarity matching algorithm to construct a bounding box around the affected region to highlight it. The following assessment metrics stated in the equations (19-28) are formulated to assess the proposed model's effectiveness and compare it to other state-of-the-art methodologies.

Accuracy (D^A)

The accuracy rate represents the ratio of accurately detected forged images among all the images in the dataset. It is computed using,

$$Accuracy(D^A) = \frac{T^p+T^n}{T^p+T^n+F^p+F^n} \tag{19}$$

True positive rate (TPR)

This is defined as the ratio of correctly detected forged images. The TPR is calculated by,

$$TPR = \frac{T^p}{T^p+F^n} \tag{20}$$

True Negative rate (TNR)

It indicates the negative event rates i.e. it defines the rate of the identified forged images. The TNR is calculated by,

$$TNR = \frac{T^n}{T^n + F^n} \tag{21}$$

Positive predictive value (PPV)

It defines the ratio of correctly forged images and total predicted forged images. The PPV rate indicates that the detected forged images are true. It is calculated by

$$\Pr ecision(PPV) = \frac{T^p}{T^p + F^p} \tag{22}$$

False positive rate (FPR)

It indicates the original images which are not detected as not forged. The FPR rate is calculated by,

$$FPR = \frac{F^p}{F^p + T^n} \tag{23}$$

False negative rate (FNR)

It indicates the missing detection rate of the forged images. i.e., the detection system failed to detect the forged images is shown as FNR. Getting lower values on this metric is much better for improving the performance. The FNR is calculated by,

$$FNR = \frac{F^n}{F^n + T^p} \tag{24}$$

Negative Predictive Value (NPV)

Negative Predictive Value (NPV) is the proportion of -1 instances correctly classified by the ML classifier.

$$NPV = \frac{1}{1 + \frac{(1-PPV)}{PPV} X \frac{TPR}{(1-TPR)} X \frac{TNR}{1-TNR}} \tag{25}$$

False discovery rate (FDR)

Defined as the number of images that are forged, but are identified as original, divided by the total number of forged images. It is calculated by,

$$FPR = \frac{F^p}{F^p + T^p} \tag{26}$$

F1-score

It integrates the PPV and TPR rates and gives the single value and it is given by,

$$F1 - Score = 2X\left(\frac{PPV*TPR}{PPV+TPR}\right) \tag{27}$$

Matthews Correlation Coefficient (MCC)

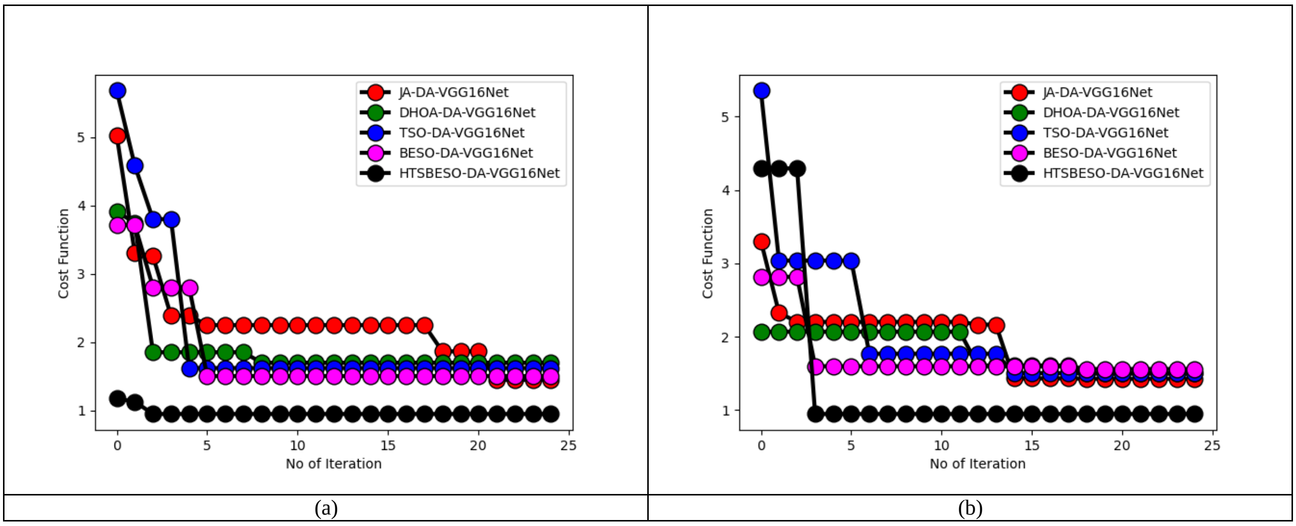
It includes all the parameters and it is defined as the geometric mean of the regression coefficients of the problem and it’s dual. It can be also formulated as follows,

$$MCC = \frac{1 - \frac{F^p - F^n}{T^p - T^n}}{\sqrt{(PPV*TPR*TNR*NPV)}} \tag{28}$$

Where the parameters $T^p, T^n, F^p, and F^n$ signifies true positive, true negative, false positive and false negative.

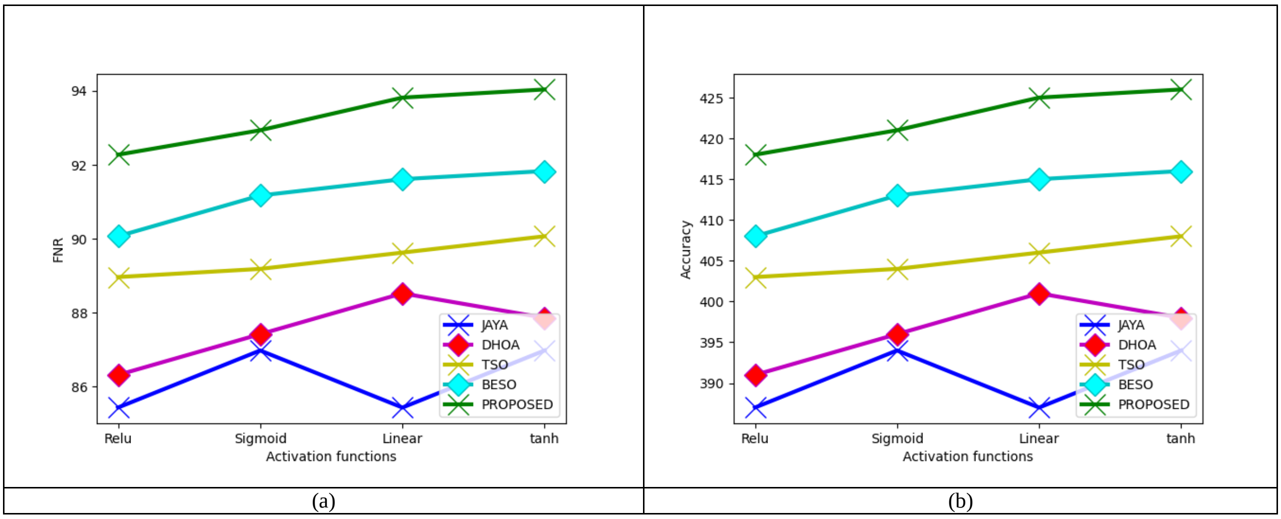
4.2 Performance Evaluation and results discussions

Convergence analysis of proposed system is performed to optimize the training procedure and ensure model achieves the best possible performance on unseen data. According to the analysis's findings, the proposed deep network-based forgery detection system delivers more accuracy and PPV because its cost function converges at the 15th iteration 40% more than that of JA-DA-VGG16Net, 50% more than TSO-DA-VGG16Net, and 60% more than BESO-DA-VGG16Net. The analysis's findings indicate that the proposed method's convergence rate is higher when compared to the current techniques. Fig. 6 depicts the result of convergence analysis for both CMFD and CoMoFoD datasets comparison with the existing methods. The performance analysis in terms of accuracy, FNR, PPV and MCC for both datasets are depicted in Fig. 7 and Fig. 8 and Fig. 9 and Fig. 10, respectively, along with the comparison with existing techniques. The comparison reveals that none of the current algorithms achieve satisfactory results in detecting forgeries. The x-axis represents the linear stage, and the performance analysis considers the Linear, Sigmoid, Tanh, and Relu activation functions, enabling clear visualization of the suggested model's performance. Remarkably, the proposed HTS-BESO-HDA-VGG16Net-based forgery detection scheme outperforms JA-DA-VGG16Net, DHOA-DA-VGG16Net, TSO-DA-VGG16Net, and BESO-DA-VGG16Net by 9.60%, 6.36%, 5.40%, and 2.87%, respectively, at the linear stage. Consequently, the output demonstrates that the proposed method achieves a high false negative rate (FNR) for the CMFD dataset. On the other hand, the proposed forgery detection system surpasses that of JA-DA-VGG16Net by 35.2%, TSO-DA-VGG16Net by 28.5%, and BESO-DA-VGG16Net by 14.34%. The analysis indicates an increased convergence rate of the proposed method compared to the existing algorithms, as evidenced by the output.

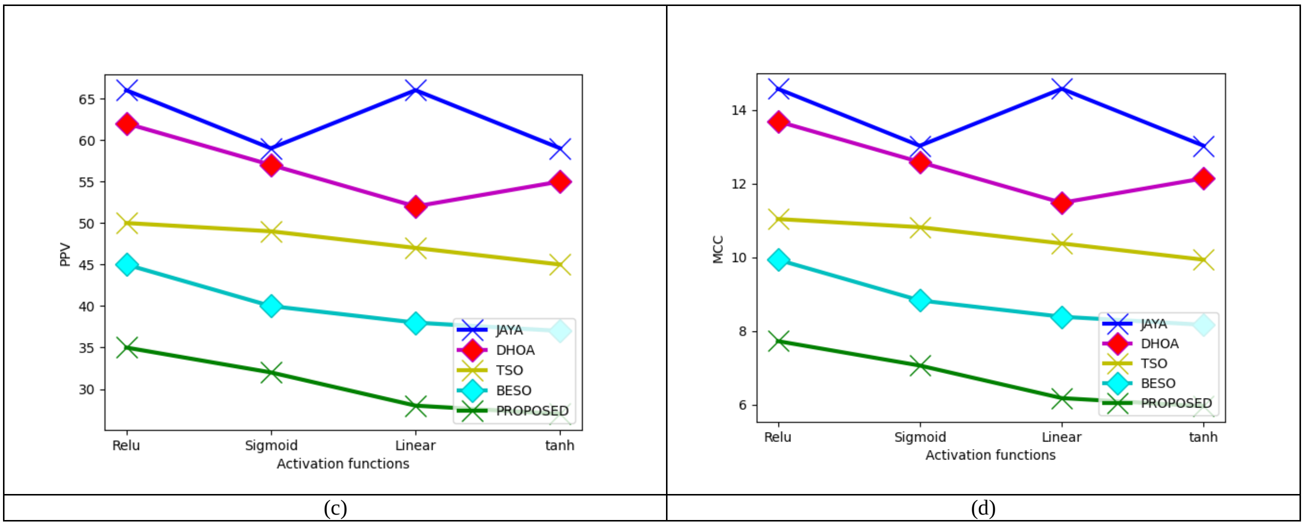


(a) (b)

Fig. 6. Convergence analysis of proposed technique and existing methods for (a) CMFD and (b) CoMoFoD datasets

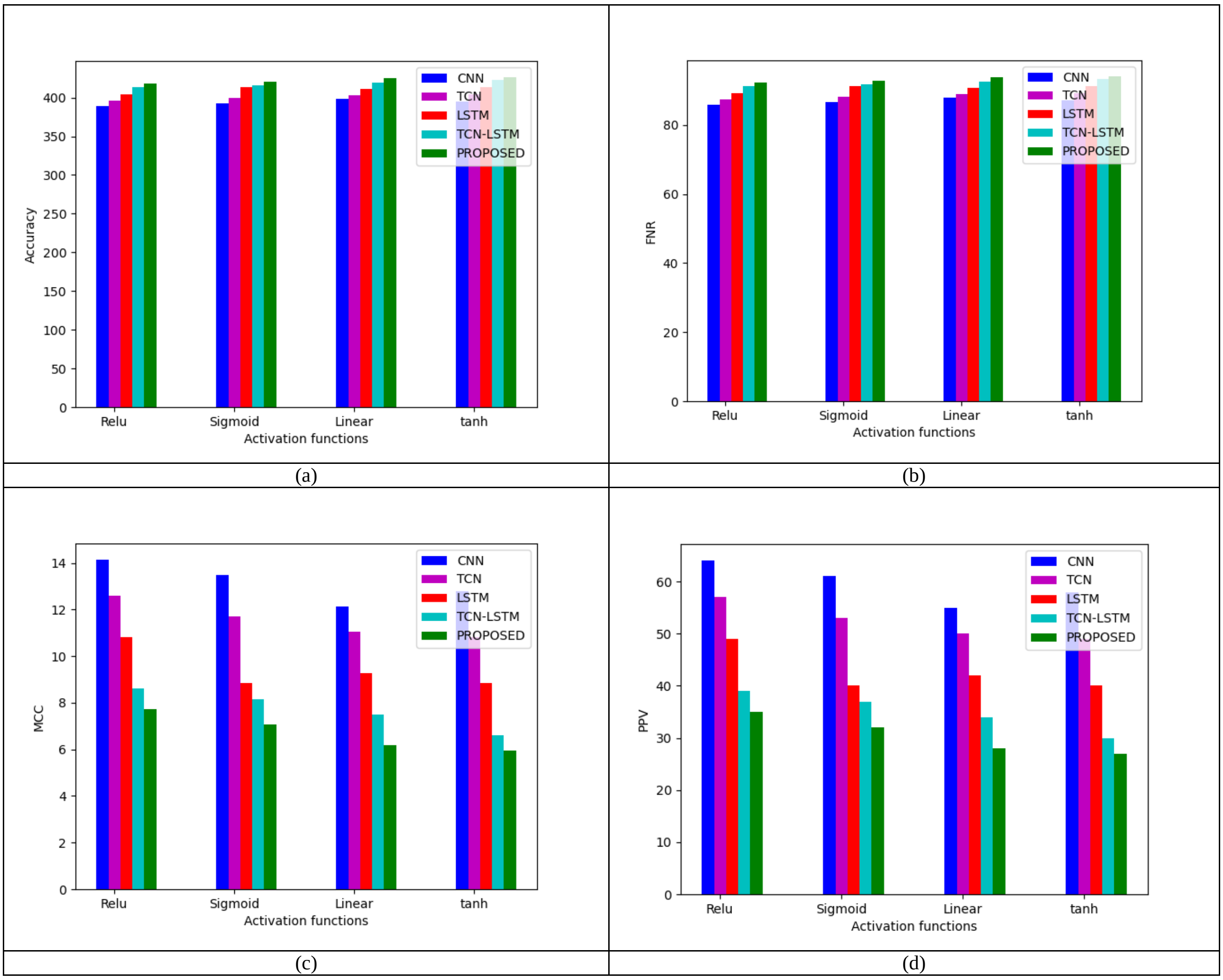


(a) (b)



(c) (d)

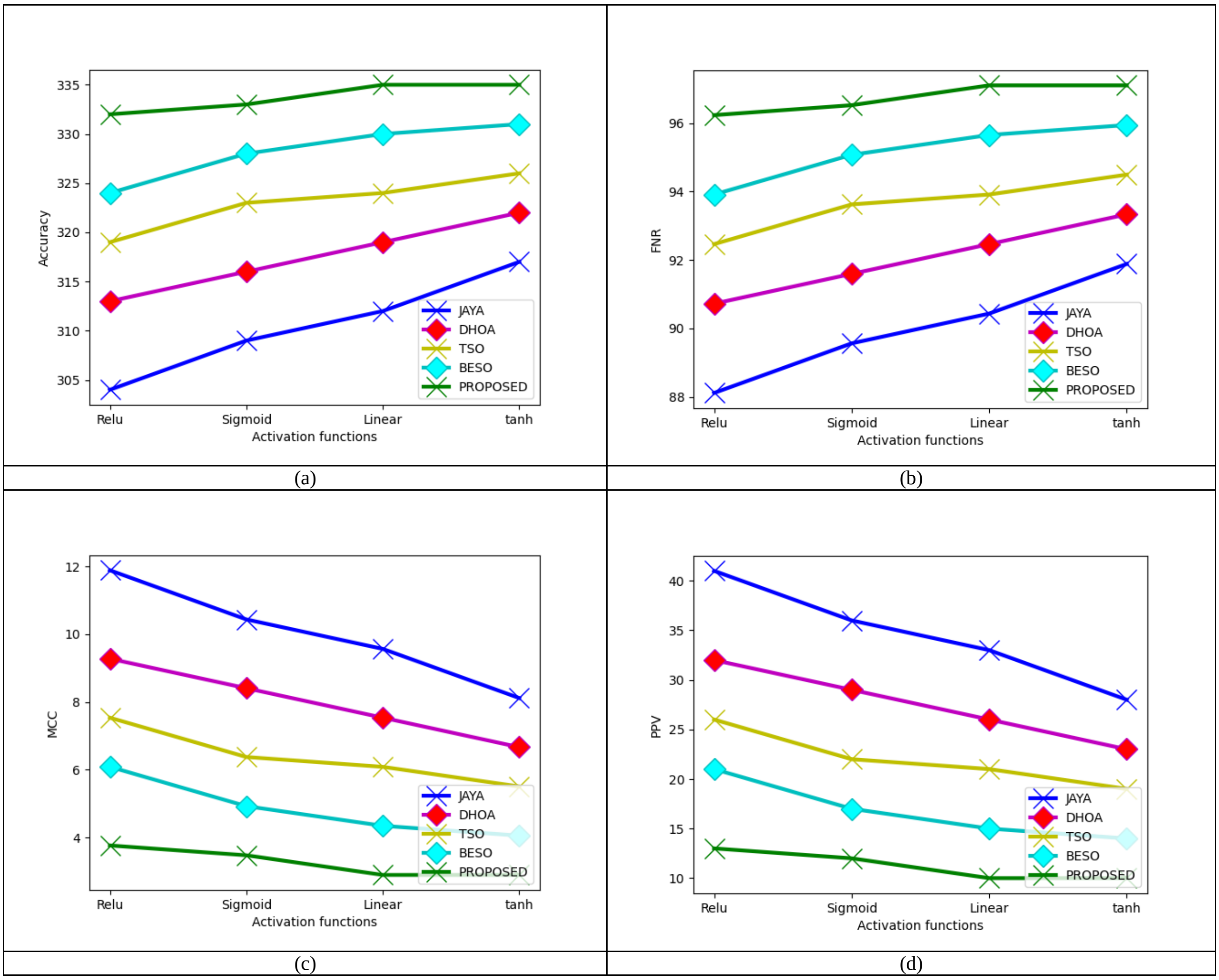
Fig. 7. Performance estimation of the proposed technique in terms of (a) FNR, (b) Accuracy, (c) PPV, (d) MCC with existing algorithms for CMFD dataset



(a) (b)

(c) (d)

Fig. 8. Performance estimation of the proposed technique in terms of (a) FNR, (b) Accuracy, (c) PPV, (d) MCC with existing techniques for CMFD dataset



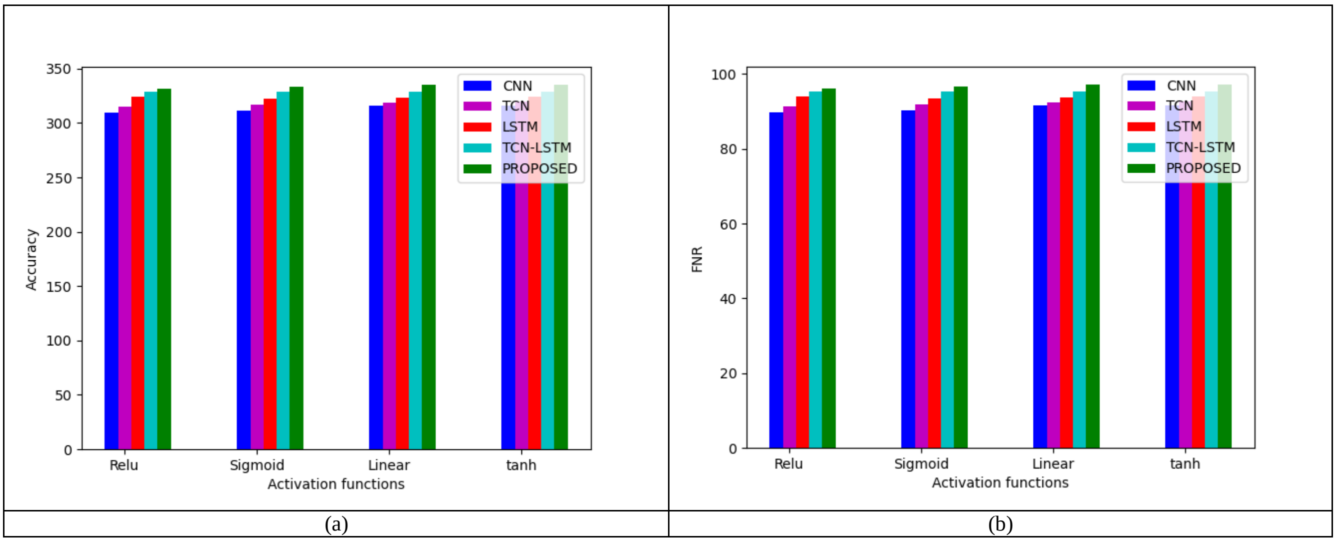
(a)

(b)

(c)

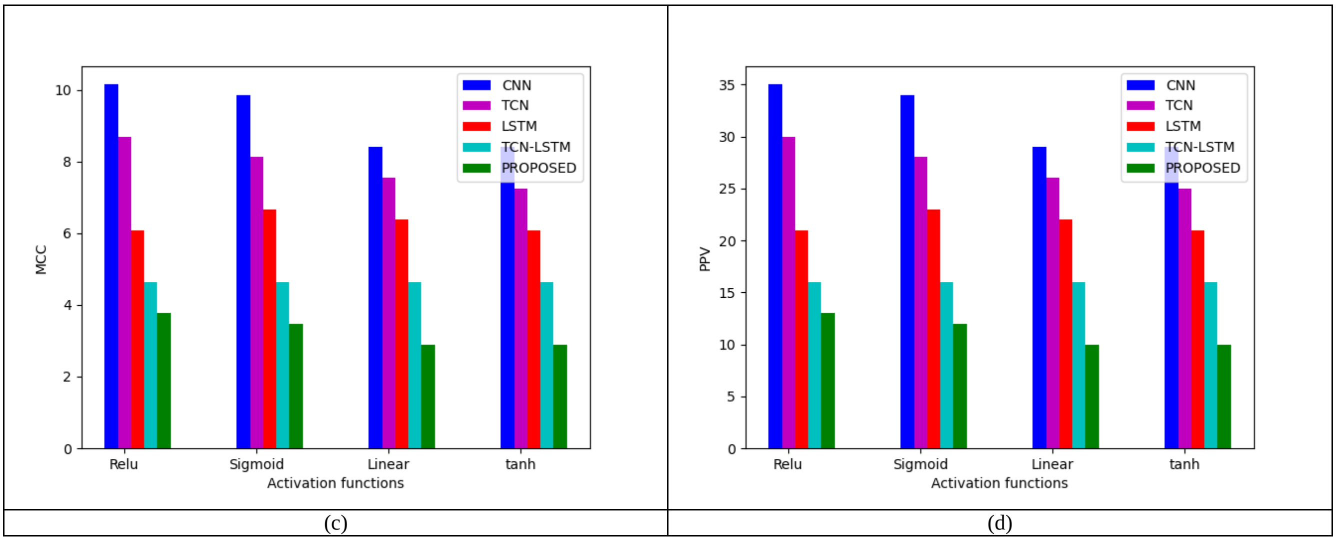
(d)

Fig. 9. Performance estimation of the proposed technique in terms of (a) FNR, (b) Accuracy, (c) PPV, (d) MCC with existing algorithms for CoMoFoD dataset



(a)

(b)



(c) (d)

Fig. 10. Performance estimation of the proposed technique in terms of (a) FNR, (b) Accuracy, (c) PPV, (d) MCC with existing techniques for CoMoFoD dataset

4.3 Overall analysis of the suggested model

The overall analysis of the proposed model is tabulated in Table 3 and Table 4. The accuracy and PPV values are used to find out the fitness value. Fitness is the main parameter, and it is used to find the value of a random number. The proposed HTS-BESO-HDA-VGG16Net-based forgery detection scheme obtained a TPR value of 8.08% more than JA, 6% more than DHOA, 4.24% more than TSO, and 1.84% more than BESO. Therefore, the analysis results show that the proposed HTS-BESO-HDA-VGG16Net-based forgery detection scheme attains better than the presented algorithms. Fig.12 and Fig. 13 depict the suggested forgery detection scheme results, showing forgery detection and the highlighted regions. Fig. 12 includes the plane forgery image (rows 2 and 5) and the forgery image post-processed with rotation and scaling (rows 1, 3 and 4). Fig. 13 includes the plane forgery image (rows 1 and 2) and the forgery image post-processed with JPG compression and image blurring (rows 3 and 4). To comprehend the suggested model's effectiveness, we compared the proposed model and a pre-existing metaheuristic algorithm for image forgery detection. The results in Table 5 and Fig. 11 indicate that the presented model outperforms the current metaheuristic optimization methods.

Table 3. Comparison of results obtained for proposed HTS-BESO-HDA-VGG16Net-based forgery detection scheme with existing algorithms for target datasets

Algorithm Comparison										
	CMFD dataset					**CoMoFoD dataset**				
Assessment metrics	JA[31]	DHOA[32]	TSO[26]	BESO[27]	**HTS-BESO**	JA[31]	DHOA[32]	TSO[26]	BESO[27]	**HTS-BESO**
Accuracy	87.28261	88.58696	90.54348	92.5	**95.0117**	91.58621	93.24138	94.62069	96.27586	**97.8753**
TPR	86.9936	88.69936	90.1919	92.32409	**94.0301**	91.57303	93.25843	94.66292	96.34831	**98.1219**
TNR	87.58315	88.47007	90.90909	92.68293	**93.0133**	91.59892	93.22493	94.57995	96.20596	**98.2881**
PPV	87.93103	88.88889	91.16379	92.91845	**94.3204**	91.31653	92.9972	94.39776	96.07843	**97.1771**
FPR	12.41685	11.52993	9.090909	7.317073	**6.97661**	8.401084	6.775068	5.420054	3.794038	**2.82001**
FNR	13.0064	11.30064	9.808102	7.675906	**4.87015**	8.426966	6.741573	5.337079	3.651685	**2.45309**
NPV	86.62281	88.27434	89.91228	92.07048	**94.7054**	91.84783	93.47826	94.83696	96.46739	**98.2213**
FDR	12.06897	11.11111	8.836207	7.081545	**4.65913**	8.683473	7.002801	5.602241	3.921569	**2.90212**
F1-Score	87.45981	88.79402	90.67524	92.62032	**95.1211**	91.4446	93.12763	94.53015	96.21318	**97.8841**
MCC	0.745653	0.771663	0.810885	0.84998	**0.89019**	0.831682	0.864794	0.892388	0.9255	**0.95557**

Table 4. Classification results comparison of proposed HTS-BESO-HDA-VGG16Net-based forgery detection scheme with existing techniques for target datasets

Classifier Comparison										
	CMFD dataset					**CoMoFoD dataset**				
Assessment metrics	CNN[35]	LSTM[36]	TCN[37]	TCN-LSTM[36]	**HTS-BESO**	CNN[35]	LSTM[36]	TCN[37]	TCN-LSTM[36]	**HTS-BESO**
Accuracy	87.93478	89.13043	91.19565	93.36957	**95.0114**	91.72414	92.96552	94.2069	95.31034	**97.6773**
TPR	88.0597	88.91258	91.04478	93.60341	**95.0275**	91.57303	92.69663	94.10112	95.22472	**97.5791**
TNR	87.80488	89.35698	91.35255	93.12639	**94.0113**	91.86992	93.22493	94.30894	95.39295	**97.2797**
PPV	88.24786	89.67742	91.6309	93.40426	**94.1307**	91.57303	92.95775	94.10112	95.22472	**97.1889**
FPR	12.19512	10.64302	8.64745	6.873614	**5.98219**	8.130081	6.775068	5.691057	4.607046	**2.72027**
FNR	11.9403	11.08742	8.955224	6.396588	**4.77019**	8.426966	7.303371	5.898876	4.775281	**2.52809**
NPV	87.61062	88.57143	90.7489	93.33333	**94.1071**	91.86992	92.97297	94.30894	95.39295	**98.5543**

FDR	11.75214	10.32258	8.369099	6.595745	**4.86923**	8.426966	7.042254	5.898876	4.775281	**2.90132**
F1-Score	88.15368	89.29336	91.3369	93.50373	**94.1202**	91.57303	92.827	94.10112	95.22472	**98.3252**
MCC	0.758615	0.782592	0.823886	0.867337	**0.8703**	0.83443	0.859261	0.884101	0.906177	**0.95327**

Table 5. Comparison of results with existing metaheuristic algorithms for image forgery detection

Existing techniques	**Accuracy**	**TPR**	**TNR**
Cristin R et al. [38]	95.1	95.1	93.8
S. Uma and P. D. Sathya[39]	96.0	94.2	100.0
C. B and P. V. Bhaskar Reddy [40]	91.9	92.6	92.8
Proposed	**97.6**	**97.5**	**97.2**

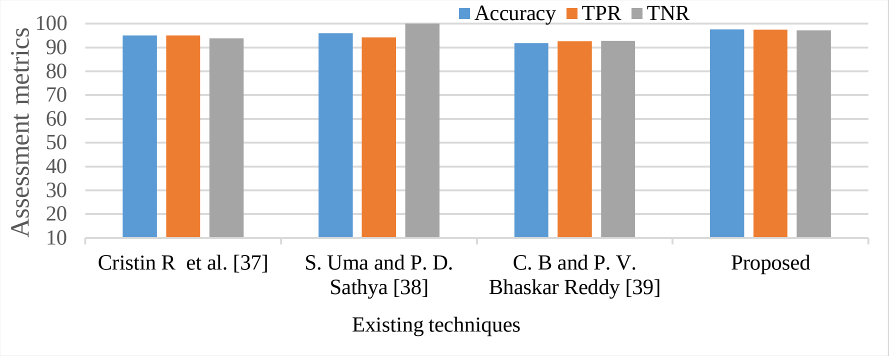


Fig. 11. Comparison of results with existing metaheuristic algorithm for image forgery detection

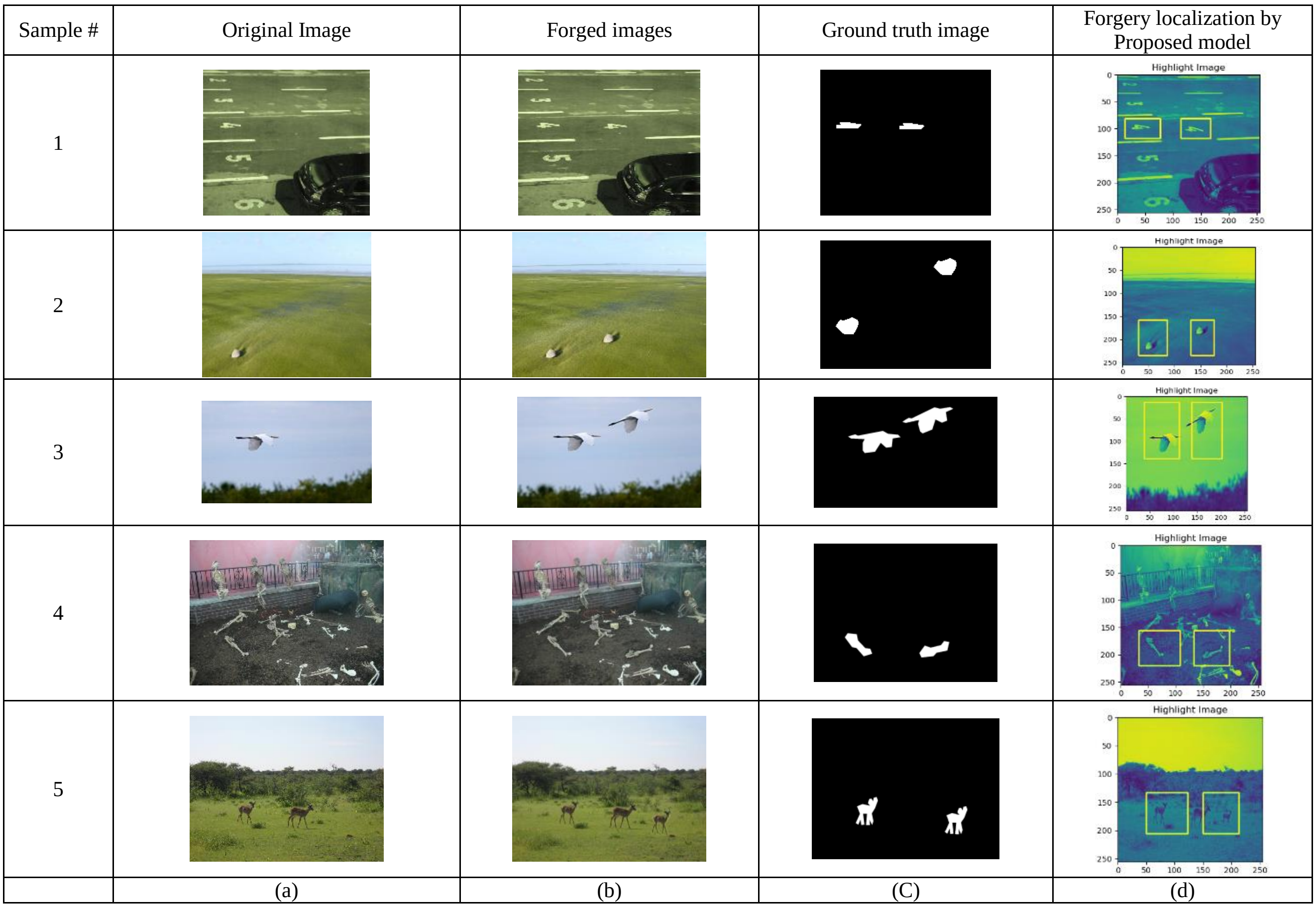


Fig. 12. Visualization of forgery detection results obtained using proposed technique for CMFD dataset

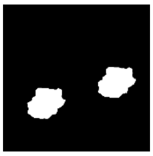
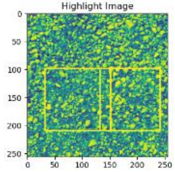
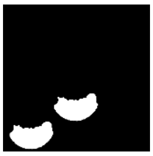
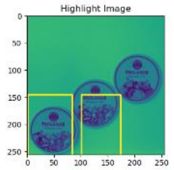
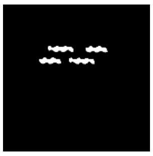
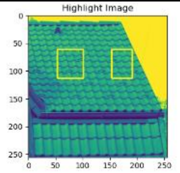
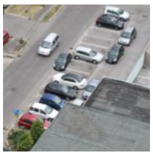
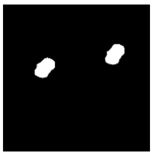
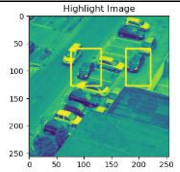
Sample #	Original Image	Forged images	Ground truth image	Forgery localization by Proposed model
1				
2				
3				
4				
	(a)	(b)	(C)	(d)

Fig. 13. Visualization of forgery detection results obtained using proposed technique for CoMoFoD dataset

5. Conclusion

This article has presented a deep learning-based forgery detection and localization model. The proposed technique encompasses the fusion of HTS-BESO-HDA-VGG16Net modules. Initially, the input images are selected from two datasets and decomposed with the help of the SWT technique. The SWT and VGG16 parameters are optimized to reduce the model complexity. The proposed model uses dilated convolution kernels with different dilation rates in the various layers VGG16 to ensure complete coverage of a square area without any holes or missing information during convolution operations. As a result, it maximizes accuracy and precision values, thereby increasing the system's overall effectiveness. The proposed model is evaluated using two standard datasets: CMFD and CoMOFoD. As per the results discussed in Section 5, the proposed methodology outperforms with a TPR value of 8.08% JA, 6% than DHOA, 4.24% than TSO, and 1.84% than BESO. Therefore the analysis shows that the HTS-BESO-based forgery detection scheme achieves maximized accuracy and precision value compared to the existing algorithms. The proposed technique also addressed the post-processing operations (scaling and rotation). However, the forgery attacks with a blend of multiple post-processing operations, such as JPEG compression, Noise, Texture, and a combination of these, can hide forgery clues. In this instance, the forgery detection significantly increases its complexity and difficulty. In the future, we extend this work to overcome all these complexities.

References

[1] "Photo tampering throughout history," 2004. http://pth.izitru.com/2004_02_00.html (accessed Sep. 25, 2020).

[2] M. Zanardelli, F. Guerrini, R. Leonardi, and N. Adami, "Image forgery detection: a survey of recent deep-learning approaches," *Multimed. Tools Appl.*, pp. 17521–17566, 2022, doi: 10.1007/s11042-022-13797-w.

[3] M. K. Arnold, M. Schmucker, and S. D. Wolthusen, "Techniques and applications of digital watermarking and content protection," *Ann. Phys. (N. Y).*, vol. 54, no. 2, 2003, [Online]. Available: http://www.mendeley.com/research/no-title-avail/%5Cnhttp://books.google.com/books?hl=en&amp;lr=&amp;id=iVEOYEukfmUC&amp;oi=fnd&amp;pg=PR13&amp;dq=Techniques+and+Applications+of+Digital+Watermarking+and+Content+Protection&amp;ots=Pz_DSY_0e-&amp;sig=2gsD5uG.

[4] A. Piva, "An Overview on Image Forensics," *ISRN Signal Process.*, vol. 2013, pp. 1–22, 2013, doi: 10.1155/2013/496701.

[5] A. C. Popescu and H. Farid, "Exposing Digital Forgeries by Detecting Duplicated Image Regions," *Tech. Report, TR2004-515, Dep. Comput. Sci. Dartmouth Coll. Hanover, New Hampsh.*, no. 2000, pp. 1–11, 2004, [Online]. Available: http://os2.zemris.fer.hr/ostalo/2010_marceta/Diplomski_files/102.pdf.

[6] K. B. Meena and V. Tyagi, "A copy-move image forgery detection technique based on Gaussian-Hermite moments," *Multimed. Tools Appl.*, vol. 78, no. 23, pp. 33505–33526, Dec. 2019, doi: 10.1007/s11042-019-08082-2.

[7] M. Bilal, H. A. Habib, Z. Mehmood, T. Saba, and M. Rashid, "Single and Multiple Copy–Move Forgery Detection and Localization in Digital Images Based on the Sparsely Encoded Distinctive Features and DBSCAN Clustering," *Arab. J. Sci. Eng.*, vol. 45, no. 4, pp. 2975–2992, 2020, doi: 10.1007/s13369-019-04238-2.

[8] K. B. Meena and V. Tyagi, "A copy-move image forgery detection technique based on tetrolet transform," *J. Inf. Secur. Appl.*, vol. 52, Jun. 2020, doi: 10.1016/j.jisa.2020.102481.

[9] G. S. Priyanka and K. Singh, "An improved block based copy-move forgery detection technique," *Multimed. Tools Appl.*, vol. 79, pp. 13011–13035, 2020.

[10] S. P. Jaiprakash, M. B. Desai, C. S. Prakash, V. H. Mistry, and K. L. Radadiya, "Low dimensional DCT and DWT feature based model for detection of image splicing and copy-move forgery," *Multimed. Tools Appl.*, vol. 79, no. 39–40, pp. 29977–30005, 2020, doi: 10.1007/s11042-020-09415-2.

[11] S. Dua, J. Singh, and H. Parthasarathy, "Detection and localization of forgery using statistics of DCT and Fourier components," *Signal Process. Image Commun.*, vol. 82, no. July 2019, p. 115778, 2020, doi: 10.1016/j.image.2020.115778.

[12] Q. Lyu, J. Luo, K. Liu, X. Yin, J. Liu, and W. Lu, "Copy Move Forgery Detection based on double matching," *J. Vis. Commun. Image Represent.*, vol. 76, no. September 2019, p. 103057, 2021, doi: 10.1016/j.jvcir.2021.103057.

[13] T. Qazi, M. Ali, K. Hayat, and B. Magnier, "Seamless Copy–Move Replication in Digital Images," *J. Imaging*, vol. 8, no. 3, pp. 1–15, 2022, doi: 10.3390/jimaging8030069.

[14] Y. Abdalla, M. Tariq Iqbal, and M. Shehata, "Copy-move forgery detection and localization using a generative adversarial network and convolutional neural-network," *Inf.*, vol. 10, no. 9, 2019, doi: 10.3390/info10090286.

[15] N. Alipour and A. Behrad, "Semantic segmentation of JPEG blocks using a deep CNN for non-aligned JPEG forgery detection and localization Content courtesy of Springer Nature , terms of use apply . Rights reserved . Content courtesy of Springer Nature , terms of use apply . Rights re," pp. 8249–8265, 2020.

[16] J. H. Bappy, C. Simons, L. Nataraj, B. S. Manjunath, and A. K. Roy-Chowdhury, "Hybrid LSTM and Encoder-Decoder Architecture for Detection of Image Forgeries," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3286–3300, 2019, doi: 10.1109/TIP.2019.2895466.

[17] X. Lin and C. T. Li, "PRNU-Based Content Forgery Localization Augmented with Image Segmentation," *IEEE Access*, vol. 8, no. 2, pp. 222645–222659, 2020, doi: 10.1109/ACCESS.2020.3042780.

[18] M. A. Elaskily *et al.*, "A novel deep learning framework for copy-moveforgery detection in images," *Multimed. Tools Appl.*, vol. 79, no. 27–28, pp. 19167–19192, 2020, doi: 10.1007/s11042-020-08751-7.

[19] Y. Rodriguez-Ortega, D. M. Ballesteros, and D. Renza, "Copy-move forgery detection (Cmfd) using deep learning for image and video forensics," *J. Imaging*, vol. 7, no. 3, 2021, doi: 10.3390/jimaging7030059.

[20] N. Jindal, "Copy move and splicing forgery detection using deep convolution neural network , and semantic segmentation Content courtesy of Springer Nature , terms of use apply . Rights reserved . Content courtesy of Springer Nature , terms of use apply . Rights reser," pp. 3571–3599, 2021.

[21] N. Goel, S. Kaur, and R. Bala, "Dual branch convolutional neural network for copy move forgery detection," *IET Image Process.*, vol. 15, no. 3, pp. 656–665, 2021, doi: 10.1049/ipr2.12051.

[22] N. Krishnaraj, B. Sivakumar, R. Kuppusamy, Y. Teekaraman, and A. R. Thelkar, "Design of Automated Deep Learning-Based Fusion Model for Copy-Move Image Forgery Detection," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/8501738.

[23] R. Ganeshan, S. Muppidi, D. R. Thirupurasundari, and B. S. Kumar, "Autoregressive-Elephant Herding Optimization based Generative Adversarial Network for copy-move forgery detection with Interval type-2 fuzzy clustering," *Signal Process. Image Commun.*, vol. 108, no. November 2021, p. 116756, 2022, doi: 10.1016/j.image.2022.116756.

[24] S. Koul and M. Kumar, "An efficient approach for copy-move image forgery detection using convolution neural network Content courtesy of Springer Nature , terms of use apply . Rights reserved . Content courtesy of Springer Nature , terms of use apply . Rights reserved .," pp. 11259–11277, 2022.

[25] G. P. Nason and B. W. Silverman, "The Stationary Wavelet Transform and some Statistical Applications," pp. 281–299, 1995, doi: 10.1007/978-1-4612-2544-7_17.

[26] L. Xie, T. Han, H. Zhou, Z. R. Zhang, B. Han, and A. Tang, "Tuna Swarm Optimization: A Novel Swarm-Based Metaheuristic Algorithm for Global Optimization," *Comput. Intell. Neurosci.*, vol. 2021, 2021, doi: 10.1155/2021/9210050.

[27] S. Ferahtia, H. Rezk, M. A. Abdelkareem, and A. G. Olabi, "Optimal techno-economic energy management strategy for building's microgrids based bald eagle search optimization algorithm," *Appl. Energy*, vol. 306, no. PB, p. 118069, 2022, doi: 10.1016/j.apenergy.2021.118069.

[28] Y. Zhou *et al.*, "Remote sensing scene classification based on rotation-invariant feature learning and joint decision making," *Eurasip J. Image Video Process.*, vol. 2019, no. 1, 2019, doi: 10.1186/s13640-018-0398-z.

[29] X. Lei, H. Pan, and X. Huang, "A dilated cnn model for image classification," *IEEE Access*, vol. 7, pp. 124087–124095, 2019, doi: 10.1109/ACCESS.2019.2927169.

[30] P. Wang *et al.*, "Understanding Convolution for Semantic Segmentation," *Proc. - 2018 IEEE Winter Conf. Appl. Comput. Vision, WACV 2018*, vol. 2018-Janua, no. March, pp. 1451–1460, 2018, doi: 10.1109/WACV.2018.00163.

[31] R. A. Zitar, M. A. Al-Betar, M. A. Awadallah, I. A. Doush, and K. Assaleh, "An Intensive and Comprehensive Overview of JAYA Algorithm, its Versions and Applications," *Arch. Comput. Methods Eng.*, vol. 29, no. 2, pp. 763–792, 2022, doi: 10.1007/s11831-021-09585-8.

[32] G. Brammya, S. Praveena, N. S. Ninu Preetha, R. Ramya, B. R. Rajakumar, and D. Binu, "Deer Hunting Optimization Algorithm: A New Nature-Inspired Meta-heuristic Paradigm," *Comput. J.*, vol. 133, no. 1, p. bxy133, 2019, doi: 10.1093/comjnl/bxy133.

[33] E. Ardizzone, A. Bruno, and G. Mazzola, "Copy-Move Forgery Detection by Matching Triangles of Keypoints," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 10, pp. 2084–2094, 2015, doi: 10.1109/TIFS.2015.2445742.

[34] D. Tralic, I. Zupancic, S. Grgic, and M. Grgic, "CoMoFoD - New database for copy-move forgery detection," *Proc. Elmar - Int. Symp. Electron. Mar.*, no. September, pp. 49–54, 2013.

[35] Q. Yang, D. Yu, Z. Zhang, Y. Yao, and L. Chen, “Spatiotemporal Trident Networks: Detection and Localization of Object Removal Tampering in Video Passive Forensics,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 10, pp. 4131–4144, 2021, doi: 10.1109/TCSVT.2020.3046240.

[36] A. R. Gu, J. H. Nam, and S. C. Lee, “FBI-Net: Frequency-Based Image Forgery Localization via Multitask Learning with Self-Attention,” *IEEE Access*, vol. 10, pp. 62751–62762, 2022, doi: 10.1109/ACCESS.2022.3182024.

[37] P. Hewage *et al.*, “Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station,” *Soft Comput.*, vol. 24, no. 21, pp. 16453–16482, 2020, doi: 10.1007/s00500-020-04954-0.

[38] Cristin R, G. M. N.R, R. L, and V. S, “Image Forgery Detection Using Back Propagation Neural Network Model and Particle Swarm Optimization Algorithm,” *Multimed. Res.*, vol. 3, pp. 21–32, 2020.

[39] S. Uma and P. D. Sathya, “Copy-move forgery detection of digital images using football game optimization,” *Aust. J. Forensic Sci.*, vol. 54, no. 2, pp. 258–279, 2022, doi: 10.1080/00450618.2020.1811376.

[40] C. B and P. V. Bhaskar Reddy, “An approach for copy-move image multiple forgery detection based on an optimized pre-trained deep learning model,” *Knowledge-Based Syst.*, vol. 269, p. 110508, 2023, doi: 10.1016/j.knosys.2023.110508.

Authors’ Profiles

Mr. Prabhu Bevinamarad completed his B.E. in Information Science and Engineering and M.Tech. Computer Science and Engineering from Visvesvaraya Technological University, Belgaum, Karnataka, India, in 2008 and 2013, respectively. Presently he is pursuing a P.hD. in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum, and Karnataka, India. His research interests include Deep learning, digital image and audio forensics.

Dr. Prakash H. Unki completed his B.E. in Electronics & Communication form Karnataka University, Dharawad and M.Tech. & Ph.D. in Computer Science and Information Sciences from Visvesvaraya Technological University, Belgaum, Karnataka, India. He is working as Professor in Department of Information Science and Engineering, B.L.D.E.A’s V. P. Dr. P. G. H. College of Engineering and Technology, Vijayapur. He is having teaching experience of 26 years in an engineering college. He published 21 papers in various international journals and conferences. Currently he is guiding two doctoral students. His research interests include Digital Image Processing, Pattern Recognition, Machine learning, Cloud Computing, Data Science and Big data. He is member of different professional societies.

Dr. Padmaraj Nidagundi completed a Bachelor’s degree in Information Science and Engineering from Visvesvaraya Technological University, a Master’s degree in Computer Engineering and a Ph.D. from Riga Technical University. Since 2010, he has been working at different software development companies. He is currently a docent at Riga Technical University. His most successful field of activity in previous years has been software development and testing, where he gained significant international work experience. In recent years, he has been a teaching staff member and participated in research projects. His research interests are software development, quantum computing, cybersecurity, and artificial intelligence.

How to cite this paper: Prabhu Bevinamarad, Prakash Unki, Padmaraj Nidagundi, "Copy-Move Forgery Detection and Localization Framework for Images Using Stationary Wavelet Transform and Hybrid Dilated Adaptive VGG16 with Optimization Strategy", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.16, No.1, pp. 38-60, 2024. DOI:10.5815/ijigsp.2024.01.04