Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 8501738, 13 pages
https://doi.org/10.1155/2022/8501738

Research Article

Design of Automated Deep Learning-Based Fusion Model for Copy-Move Image Forgery Detection

N. Krishnaraj,[1] **B. Sivakumar**,[2] **Ramya Kuppusamy**,[3] **Yuvaraja Teekaraman**,[4] **and Amruth Ramesh Thelkar**[5]

[1]*Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur 603203, Chennai, India*
[2]*Department of Computing Technologies, School of Computing, SRM Institute of Science and Technology, Kattankulathur 603203, Chennai, India*
[3]*Department of Electrical and Electronics Engineering, Sri Sairam College of Engineering, Bangalore 562106, Karnataka, India*
[4]*Department of Electronic and Electrical Engineering, The University of Sheffield, Sheffield S1 3JD, UK*
[5]*Faculty of Electrical & Computer Engineering, Jimma Institute of Technology, Jimma University, Jimma, Ethiopia*

Correspondence should be addressed to Yuvaraja Teekaraman; yuvarajastr@ieee.org and Amruth Ramesh Thelkar; amruth.rt@gmail.com

Received 28 October 2021; Revised 3 January 2022; Accepted 12 January 2022; Published 31 January 2022

Academic Editor: Deepika Koundal

Due to the exponential growth of high-quality fake photos on social media and the Internet, it is critical to develop robust forgery detection tools. Traditional picture- and video-editing techniques include copying areas of the image, referred to as the copy-move approach. The standard image processing methods physically search for patterns relevant to the duplicated material, restricting the usage in enormous data categorization. On the contrary, while deep learning (DL) models have exhibited improved performance, they have significant generalization concerns because of their high reliance on training datasets and the requirement for good hyperparameter selection. With this in mind, this article provides an automated deep learning-based fusion model for detecting and localizing copy-move forgeries (DLFM-CMDFC). The proposed DLFM-CMDFC technique combines models of generative adversarial networks (GANs) and densely connected networks (DenseNets). The two outputs are combined in the DLFM-CMDFC technique to create a layer for encoding the input vectors with the initial layer of an extreme learning machine (ELM) classifier. Additionally, the ELM model's weight and bias values are optimally adjusted using the artificial fish swarm algorithm (AFSA). The networks' outputs are supplied into the merger unit as input. Finally, a faked image is used to identify the difference between the input and target areas. Two benchmark datasets are used to validate the proposed model's performance. The experimental results established the proposed model's superiority over recently developed approaches.

1. Introduction

Recently, the extension of Internet services and the strengthening and proliferation of social networks such as Reddit, Facebook, and Instagram had had an important effect on the number of content prevailing in digital media. As per the International Telecommunication Union (ITU), by the end of 2019, 53.6% of the world's population utilizes the Internet, which implies around 4.1 billion peoples have access to these technologies, as well as with distinct mechanisms accessed on the Internet [1]. Even though in many situations has only been manipulated or content shared is original for entertainment purposes only, in another case the manipulation might be intended for falsehood purposes, using forensic and political consequences, for example, utilizing the false contents as digital proof in criminal investigations. Video/Image manipulation represents few actions that are accomplished on the digital content via software editing tools (e.g., GIMP, PIXLR, Adobe Photoshop) or artificial intelligence. Especially, the

copy-move techniques copy a portion of the image and paste it onto similar images [2]. Since editing tools advance, the quality of false images rises and it seems to be original images. Furthermore, postprocessing manipulations, such as brightness equalization/changes and JPEG compression, might decrease the traces left by manipulation and make it very complex to identify [3]. The copy-move forgery detection (CMFD) consists of deep learning- and hand-crafted-based approaches. The previous one is largely separated into hybrid, block, and key point-based methods and next employs convention framework from fine-tuned/scratch algorithms.

Block-based methods utilized distinct kinds of feature extraction, for example, Tetrolet transforms/Fourier transform, and DCT (discrete cosine transform). The major concern is the performance reduction while the copied objects are resized/rotated since the recognition of forging can be performed by a matching procedure [4]. Conversely, key point-based methods such as SURF (Speed-Up Robust Features) and SIFT (scale-invariant feature transform) are very stronger to lighting and rotation differences; however, they have many problems to conquer, for example, natural duplicate objects spotted as false duplicate objects and reliance on original key point in an image, and detect forgeries in the area of uniform intensity [5]. A hybrid method provides constant results by means of F1-score, precision (P), and recall (R) for an individual dataset.

1.1. Motivation. There is a current development of deviating traditionally handcrafted feature extraction for employing convolutional neural network (CNN)-based extractor. But, in few conventional CNN-based forensic detectors is usually not real world for several details, for example, by means of strength in feature extraction and solution of tampering position. Thus, there are various attempts to develop a preprocessing layer for enhancing the strength of feature extraction [6] and combine several detector-based likelihood maps and individual CNN-based consistency maps for improving the solution of tampering location. But still, they endure numerous limits in the abovementioned methods. Initially, current pixel-wise tampering detector adapts an autonomous patch-based approach instead of utilizing the related data amongst patches [7]. Moreover, the lack of statistical features on flat regions (blue ocean, clear sky, and so on) leads to uncertainty approximation and degradation of recognition accuracy. In this situation, the texture of an image content has become a decisive factor to enhance recognition performance. In addition, with the quick growth of image-editing software, the remainder left by the manipulation process has behavior like its pristine versions (viz., tampering trace is difficult to identify) [8]. Then, decreasing the possibility of recognition mismatch and enhancing the solution of localization (managed by the small units of finding) still remain an open challenge.

1.2. Scope of the Research Work. This article presents an automated DL-based fusion model for copy-move forgery detection and localization (DLFM-CMDFC). The proposed DLFM-CMDFC technique comprises the fusion of generative adversarial network (GAN) and densely connected network (DenseNet) models.

2. Related Works

Yao et al. [9] develop efficient detectors, which can complete image fake localization and detection. Particularly, based on the developed continuous high-pass filter, they initially determine an effective CNN framework automatically for and adaptively extracting features and propose an RFM model for improving tamper recognition performance and localization solution. Abdalla et al. [10] examine copy-move counterfeit findings with a fusion processing method including an adversarial method and deep convolution method. Four databases were employed. The result indicates a considerably higher recognition accuracy (~95%) shown by the discriminator counterfeit detector and DL-CNN models. Accordingly, an end-to-end trained DNN method for counterfeit finding seems to be an optimum approach.

Diallo et al. [11] introduce an architecture enhancing strength for image counterfeit recognition. The vital stage of this architecture is to consider the image quality matching to the selected application. Consequently, it is based on a camera recognition method-based CNN model. Lossy compressions like JPEG are taken into account as general kind of inadvertent/intentional concealment of image counterfeit, which results in manipulation. Consequently, the trainable CNN is fed into a combination of distinct amounts of uncompressed and compressed images. Rodriguez-Ortega et al. [12] present 2 methods, which utilize the DL method, an approach with a convention framework, and a method with the TL model. In all the cases, the effect of depth of the network can be examined by means of F1-score, precision (P), and recall (R). In addition, the challenge of generalization can be resolved from 8 distinct open-access databases.

In the study by Doegar et al. [13], CNN-based pretrained AlexNet method deep feature was employed, which is effective and efficient than that of current advanced methods on open-source standard database MICC-F220. Marra et al. [14] introduce a CNN-based image counterfeit recognition architecture that makes decisions according to the full resolution data collected from the entire image. Because of gradient checkpointing, the architecture can be trained end to end using constrained memory resources and weak (image-level) supervision, which enables the joint optimization of each parameter.

Dixit and Bag [15] presented a technique where SWT and spatial-limited edge-preserving watershed segmentation are employed on input images in the preprocessing phase. Descriptor computation and key point extraction were implemented. Outlier removal can be executed by the RANSAC approach. Furthermore, counterfeit areas are positioned by relation map generation. In Bi et al. [16], a counterfeit localization generator GM has been presented on the basis of a multidecoder single task method. Through adversarial training 2 generators, the presented alpha-learnable WCT blocks in GT suppress manually the

tampering artifact in the counterfeit images. In the meantime, the localization and detection capacities of GM would be enhanced by learning the phony images restored by GT.

Ghai et al. [17] aim at designing a DL-based image counterfeit recognition architecture. The presented model focuses on detecting images counterfeit with splicing and copy-move methods. The image conversion method supports the detection of related features to the network for training efficiently. Next, the pretrained personalized CNN is utilized for training the public standard databases. In Rao et al. [18], a new image counterfeit localization and detection system has been presented on the basis of the DCNN model that integrates a multisemantic CRF-based attention method. The presented model depends on the main findings that the boundary transition artifact arising from the blending operation is global in several image counterfeit manipulations, that is, established in this model using a method with CRF-based attention method through making attention mapping to characterize the possibility of being counterfeit for all the pixels in an image.

3. The Proposed Model

In this study, an efficient DLFM-CMDFC technique is presented for automated copy-move forgery detection and localization model. The proposed DLFM-CMDFC technique encompasses the fusion of GAN and DenseNet models. In DLFM-CMDFC technique, the two outcomes are combined into a layer to define the input vectors with the initial layer of the ELM classifier. Moreover, the optimal parameter tuning of the ELM model takes place by the use of AFSA. The outcomes of the networks are fed as input to the merger unit. Lastly, the difference between the input and targets areas is identified in a forged image.

3.1. GAN-Based Forgery Image Generation. Advancements of technology are assisting GAN to generate forged images, which fool even the more advanced detector [58]. It must be noted that the main objective of generative adversarial network is to create images that could not be differentiated from the primary source image. As demonstrated, generator G_A was applied for transforming input images A from domain D_A to output domain D_B. Then, generator G_B can be utilized for mapping image B back to domain D_A (the original domain). Thereby, another set of cycle consistency losses are included in the standard adversarial losses borne by the discriminator, therefore, attaining $A = G(G(A))$ and assisting the 2 images to be coupled. Highly advanced editing tools are needed for changing an image context. This tool should be capable of altering images when preserving the original source perspectives, shadowing, etc. Those without forgery detection training will not able to differentiate the actual image from an image forged utilizing this methodology that implies that it is the best candidate to develop support material for false news reports.

GAN task is given in the following: (1) build a discriminator network; (2) load a dataset; (3) generate a sample image; (4) build a generator network; (5) closing thoughts; (7) training difficulties. The GAN network branch is shown in Figure 1 [19].

In the presented GAN network, it is considered 2 major phases: (1) in the initial phase, the generator fashions an image from haphazard noise input, and (2) then, the image, as well as various images based on a similar database, is proposed for the discriminator. (3) After the discriminator is proposed by the real and forged images, it provides likelihoods through numbers in the range of zero and one, extensive. Now, zero denotes a forged image and one represents a higher probability for validity. It should be noted that the discriminator must be pretrained previous to the generator since it generates clear gradients. Retaining the constant values enables the network to possess a good understanding of the gradients, that is, the foundation of its learning. But GAN has been proposed as a kind of game performed among opposite networks, and retaining their balance could be problematic. Inopportunely, learning is hard for GAN when the generator/discriminator is highly proficient since GAN usually needs extensive training time. Thus, for example, a GAN can take a long time for an individual GPU, whereas for an individual CPU, a GAN might need few more days.

3.2. DenseNet Model. In this study, the DenseNet-121 framework is utilized as the foundation. In addition, the transfer learning method has been employed in the DenseNet architecture for enhancing the system performance [20]. DenseNets in contrast to common belief require fewer parameters when compared to traditional CNN models since they do not want to learn unnecessary feature maps. The basic idea of the DenseNet architecture is the feature reuse that leads to tremendously compact version. Consequently, it requires fewer parameters when compared to another CNN model because no feature map is repeated. Once CNN goes further, it faces challenges. DenseNet makes this connectivity much easier by simply interconnecting all the layers straightforwardly with every layer. DenseNets utilize the network's capability by reutilizing features. All the layers in DenseNet obtain further input over every prior layer and transmit its feature map to the succeeding layers.

All the layers receive good understanding from the above layers, namely, the idea of concatenation that is utilized. For maximizing computational recycling among the classifiers, incorporating several classifiers to a model and DCNN and interconnect with dense connectivity for effective image classification [21]. A study has proved that a convolution network with smaller connections among layers and those nearer to the output could be very much deeper, and it would be more precise for training. DenseNet attains important developments over the advanced technology when consuming minimum memory and processing to improve its efficiency. The DL library PyTorch and torchvision are utilized, that is, a pretrained data learning method that contains a maximal control across overfitting and also improves the optimization of results from the very first. It consists of 1 classification layer (16), 2 DenseBlocks (1×1 and 3×3 convs), 3 transition layers (6, 12, and 24), and 5 convolution and pooling layers.

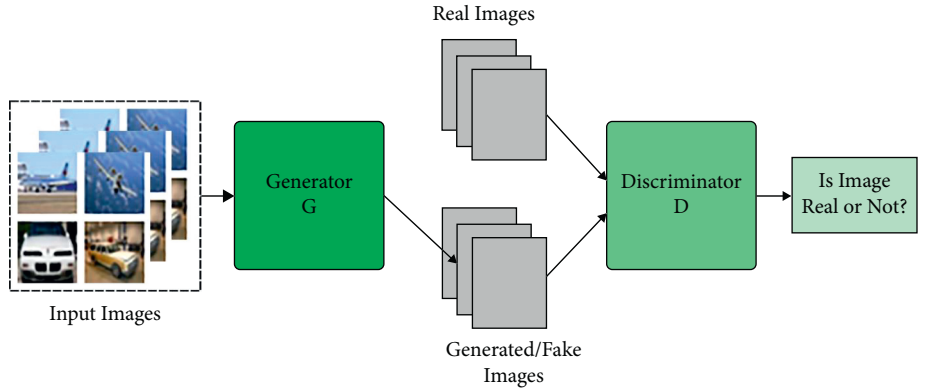

Figure 1: Framework of GAN.

3.3. Optimal ELM Model Using AFSA. ELM is essentially an SLFN algorithm. The variance among ELM and SLFN exists within the weight of the output layer, and hidden layer neurons are upgraded. In SLFN, the weight of input and output layers is initiated arbitrarily, and the weight of the layers is upgraded using the BP model. In ELM, the weight of the hidden layer is allocated arbitrarily but not upgraded, and the weight of the output layer is upgraded at the time of training. Since in ELM, the weight of single layer is upgraded against both layers of SLFN, it would make ELM quicker when compared to SLFN.

Assume the trained database as (x_j, t_j) in which $x_j = [x_{j1}, x_{j2}, \ldots, x_{jN}]^T$ represents the input vector and t_j denotes the output vector. The output of j^{th} hidden layer neuron is represented as $g(w_i, b_i, x_j)$, in which w_i indicates the weight vector connected the input neuron to i^{th} hidden layer neuron, b_i signifiers the bias of i^{th} hidden neurons, and g denotes the activation function. All the hidden layer neurons of ELM are interconnected to all the output layer neurons with related weight, and they represent the weight interconnecting the i^{th} hidden layer neuron with output neuron as β_i. This framework is denoted arithmetically by

$$\sum_{i=1}^{L} \beta_i g\left(w_i, b_i, x_j\right) = t_j, \tag{1}$$

where L represents the number of hidden neurons, and j indicates the output or input sample of overall N trained samples. The aforementioned formula is expressed by

$$H\beta = T. \tag{2}$$

In the above formula, consider m output node as

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L\times m} \quad \text{and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N\times m}, \tag{3}$$

where H denotes the output matrix of the hidden layer, which is given as

$$H = \begin{bmatrix} g(w_1, b_1, x_1) & \cdots & g(w_{\text{L}}, b_{\text{L}}, x_1) \\ \vdots & \ddots & \vdots \\ g(w_1, b_1, x_N) & \cdots & g(w_{\text{L}}, b_{\text{L}}, x_N) \end{bmatrix}. \tag{4}$$

The minimum norm least square of (2) is

$$\widehat{\beta} = H^{+}T, \tag{5}$$

where H^{+} is the Moore–Penrose generalized inverse of matrix H. H^{+} is evaluated by singular value decomposition (SVD), QR approach, orthogonal projection model [22], and orthogonalization method.

It must standardize the scheme (to avoid overfitting), and the optimization issues turn into

$$\min\left(\|\beta\|^2 + C\sum_{i=1}^{N} \left\|\xi_i^T\right\|^2 \right), \tag{6}$$

where $\xi_i = t_i^T - h(x_i)\beta$ denotes the trained error of i^{th} instance and C denotes the appropriate penalty factor. It might convert these problems to its dual form and create the Lagrangian function as

$$F = \|\beta\|^2 + C\sum_{i=1}^{N} \|\xi\|^2 - \sum_{i=1}^{N}\sum_{j=1}^{L} \alpha_{ij}\left(h(x_i)\beta_j - t_{ij} + \xi_{ij}\right). \tag{7}$$

Take the partial derivative of the aforementioned formula and apply KKT condition. When $L < N$, the size of matrix $H^T H$ is lesser when compared to matrix HH^T,

$$\beta = H^{+}T = \left(\frac{I}{C} + H^T H\right)^{-1} H^T T. \tag{8}$$

Hence, the last output of ELM is

$$f(x) = h(x)\beta = h(x)\left(\frac{I}{C} + H^T H\right)^{-1} H^T T. \tag{9}$$

Once $L > N$, the size of matrix HH^T is lesser when compared to the matrix $H^T H$, the solution of the equation becomes

$$\beta = H^{+}T = H^{T}\left(\frac{I}{C} + HH^{T}\right)^{-1} T. \quad (10)$$

Thus, the last output of ELM is

$$f(x) = h(x)\beta = h(x)H^{T}\left(\frac{I}{C} + HH^{T}\right)^{-1} T. \quad (11)$$

For the binary classification problems, the decision function of ELM can be expressed by

$$f(x) = \text{sign}(h(x)\beta). \quad (12)$$

For multiclass instance, the class label of instance is expressed by

$$\text{label}(x) = \arg\max_{1 \le i \le m}\{f_i(x)\}. \quad (13)$$

Then

$$f(x) = r_1(x), f_2(x), f_3(x), \ldots, f_n(x)]^{T}. \quad (14)$$

ELM was employed for the classification and prediction tasks in various fields. To optimally adjust the learning rate of the ELM model, the AFSA is used, which is a kind of swarm intelligence method depending on the behavior of the animal. It was developed by Li et al. in 2002 [23]. Its fundamental is the inspiration of collision, foraging, and clustering behavior of fish and the collective support in a fish swarm for realizing a global optimum points. The highest distance pass through in the artificial fish method can be determined by *Step*, the apparent distance pass through by the artificial fish can be determined by *Visual*, the retry amount represent the Try_ *Number* also the factors of crowd amount represent η. The location of a single artificial fish is defined by the resulting vectors $X = (X_1, X_2, \ldots, X_n)$, and the distance among artificial fish i and j denotes $d_{ij} = |X_i - X_j|$. The behavior function for the artificial fish can be determined by random, prey, swarm, and follow.

Assume that the fish observe their food using their eyes and the present location is X_i, as well as an arbitrarily elected location is X_j within their perceptive range:

$$X_j = X_i + \text{Visual} \times \text{rand}(0 \sim 1), \quad (15)$$

where *rand* (0-1) represents an arbitrary value between zero and one. When $Y_i > Y_j$, the fish move in this direction. Or else, the method arbitrarily selects a novel location X_j for judging whether it fulfills the moving criteria. When it performs,

$$X_i^{t+1} = X_i^t + \frac{X_j - X_i^t}{\left\|X_j - X_i^t\right\|} \times \text{Step} \times \text{rand}(0 \sim 1). \quad (16)$$

When it does not Try_ *Number* times, an arbitrary movement can be generated by

$$X_i^{t+1} = X_i^t + \text{Visual} \times \text{rand}(0 \sim 1). \quad (17)$$

In order to prevent overcrowding, an artificial present location X_i is fixed. Next, the amount of fish in its n_f company and X_c center in the region (i.e., $d_{ij} < Visual$) are defined. When $Y_c/n_f < \eta \times Y_i$, the position of companion represents the optimal number of food and lower crowding. Subsequently, the fish moves to its companion region center position:

$$X_i^{t+1} = X_i^t + \frac{X_c - X_i^t}{\left\|X_c - X_i^t\right\|} \times \text{Step} \times \text{rand}(0 \sim 1). \quad (18)$$

Or else it starts to perform the behavior of prey.

The present location of artificial fish swarm can be determined by X_i. The swarm defines its main company Y_j as X_j in the region (i.e., $d_{ij} < \text{Visual}$). When $Y_j/n_f < \eta \times Y_i$, the position of companies represents the optimal number of food and lesser crowd [24]. Next, the swarm moves to X_j:

$$X_i^{t+1} = X_i^t + \frac{X_j - X_i^t}{\left\|X_j - X_i^t\right\|} \times \text{Step} \times \text{rand}(0 \sim 1). \quad (19)$$

It enables artificial fish to attain company and food through a large regional area. A location is arbitrarily chosen, as well as artificial fish moves to it. Figure 2 illustrates the flowchart of AFSA.

With the searching space of D dimensional, highly probable distance amid 2 artificial fishes is utilized for vigorously limiting the *Visual* & *Step* of an artificial fish. It is determined by Max$D$:

$$\text{Max}D = \sqrt{(x_{\max} - x_{\min})^2 \times D}, \quad (20)$$

where $x_{\min}$ and $x_{\max}$ represent the lower and upper bounds of the optimization range, respectively, and D indicates the dimension of the search space.

4. Experimental Validation

This section investigates the result analysis of the proposed model on MNIST and COCO datasets. Figure 3 shows a few sample image, tampered image, and localization image.

Table 1 and Figure 4 provide the performance analysis of the proposed model on the applied MNIST dataset under varying runs. The results demonstrated that the proposed model has gained effective outcomes under distinct runs. For instance, under run-1, the proposed model has attained effective outcome with the prec_n of 96.38%, rec_l of 93.71%, acc_y of 94.29%, and F_{score} of 95.98%. Also, under run-3, the presented manner has reached effective outcome with the prec_n of 93.54%, rec_l of 97.30%, acc_y of 94.88%, and F_{score} of 97.19%. Besides, under run-5, the presented technique has obtained effective outcome with the prec_n of 96.80%, acc_y of 97.43%, acc_y of 96.87%, and F_{score} of 94.69%.

Figure 5 demonstrates the ROC analysis of the DLFM-CMDFC technique on the test MNIST dataset. The figure has shown that the DLFM-CMDFC technique has resulted in an effective outcome with a maximum ROC of 98.5180.

Figure 6 portrays the accuracy analysis of the DLFM-CMDFC technique on the test MNIST dataset. The results demonstrated that the DLFM-CMDFC technique has accomplished improved performance with increased training and validation accuracy. It is noticed that the DLFM-

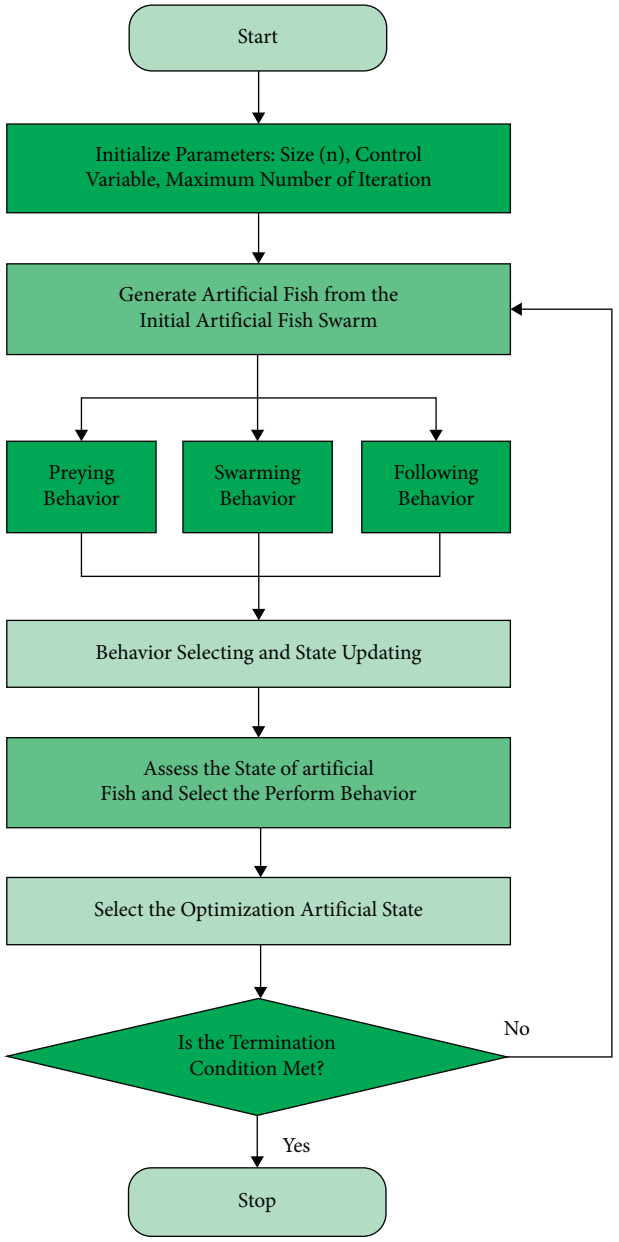

Figure 2: Flowchart of AFSA.

CMDFC technique has gained improved validation accuracy over the training accuracy. Similarly, Figure 7 depicts the loss analysis of the DLFM-CMDFC technique on the test MNIST dataset. The results established that the DLFM-CMDFC technique has resulted in a proficient outcome with reduced training and validation loss. It is observed that the DLFM-CMDFC technique has offered reduced validation loss over the training loss.

Table 2 and Figure 8 offer the performance analysis of the presented technique on the applied CIFAR-10 dataset under varying runs. The outcomes exhibited that the presented approach has reached effectual outcomes under different runs. For instance, under run-1, the presented manner has attained effective outcome with the $prec_n$ of 96.52%, rec_l of 96.15%, acc_y of 96.36%, and F_{score} of 96.66%. Followed by, under run-3, the proposed model has attained effective outcome with the $prec_n$ of 97.95%, rec_l of 96.68%, acc_y of 97%, and F_{score} of 96.57%. In addition, under run-5, the projected system has achieved effective outcome with the $prec_n$ of 97.46%, rec_l of 96.50%, acc_y of 97.35%, and F_{score} of 94.52%.

Figure 9 depicts the ROC analysis of the DLFM-CMDFC technique on the test CIFAR-10 dataset. The figure outperformed that the DLFM-CMDFC scheme has resulted in an effective outcome with the maximal ROC of 98.7262.

Figure 10 demonstrates the accuracy analysis of the DLFM-CMDFC technique on the test CIFAR-10 dataset. The outcomes showcased that the DLFM-CMDFC technique has accomplished improved efficiency with increased training and validation accuracy. It can be noticed that the DLFM-CMDFC manner has gained increased validation accuracy over the training accuracy.

Figure 11 represents the loss analysis of the DLFM-CMDFC manner on the test CIFAR-10 dataset. The outcomes recognized that the DLFM-CMDFC approach has resulted in a proficient outcome with the decreased training and validation loss. It can be stated that the DLFM-CMDFC technique has obtainable minimum validation loss over the training loss.

The $prec_n$ analysis of the DLFM-CMDFC technique with existing ones on the test dataset is given in Table 3.

Figure 12 illustrates the $prec_n$ analysis of the DLFM-CMDFC technique with existing ones. The figure has shown that the IFD-AOS-FPM and CMFD-BMIF techniques have obtained reduced $prec_n$ of 53.90% and 54.40%. At the same time, the CMFD and BB-KB-ICMFD techniques have resulted in moderate $prec_n$ of 57.34% and 56.62%, respectively. Moreover, the CMFD-GAN-CNN technique has accomplished near optimal $prec_n$ of 69.64%. However, the DLFM-CMDFC technique has resulted in superior performance with the $prec_n$ of 97.27%.

Figure 13 illustrates the rec_{ll} analysis of the DLFM-CMDFC approach with current ones. The figure exhibited that the CMFD and CMFD-BMIF algorithms have obtained reduced rec_{ll} of 49.39% and 80.20%, respectively. Concurrently, the CMFD-GAN-CNN and BB-KB-ICMFD techniques have resulted in a moderate rec_{ll} of 80.42% and 80.40%, respectively. In addition, the IFD-AOS-FPM system has accomplished near optimal rec_{ll} of 83.27%. But, the DLFM-CMDFC technique has resulted in a maximal performance with the rec_{ll} of 96.46%.

Figure 14 depicts the F_{score} analysis of the DLFM-CMDFC system with present ones. The figure portrayed that the IFD-AOS-FPM and CMFD techniques have obtained reduced F_{score} of 54.39% and 49.26, respectively. Simultaneously, the CMFD-BMIF and BB-KB-ICMFD techniques have resulted in a moderate F_{score} of 59.43% and 60.55%, respectively. Also, the CMFD-GAN-CNN algorithm has accomplished near optimal F_{score} of 88.35%. Eventually, the DLFM-CMDFC manner has resulted in increased efficiency with the F_{score} of 96.06%.

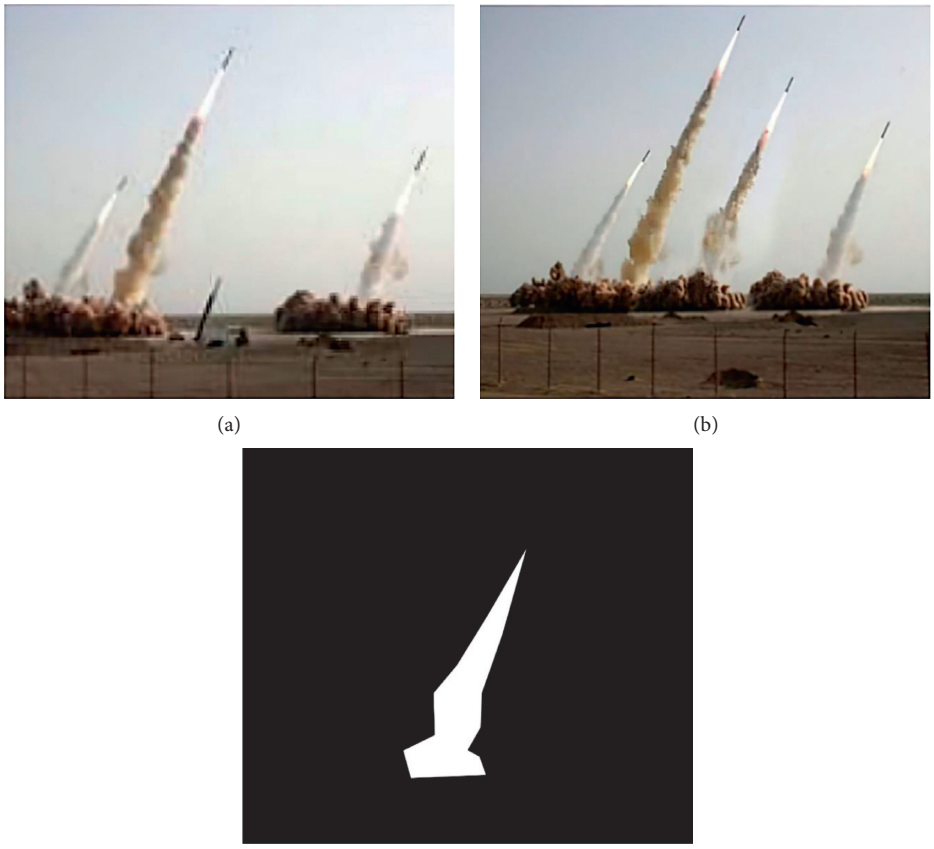

Figure 3: Sample image forgery detection results. (a) Original image. (b) Tampered image. (c) Localization image.

Table 1: Result analysis of DLFM-CMDFC model on MNIST dataset.

No. of runs	Precision	Recall	Accuracy	F-score
Run-1	96.38	93.71	94.29	95.98
Run-2	93.83	95.51	94.61	93.93
Run-3	93.54	97.30	94.88	97.19
Run-4	96.54	95.52	96.45	97.32
Run-5	96.80	97.43	96.87	94.69
Average	**95.42**	**95.89**	**95.42**	**95.82**

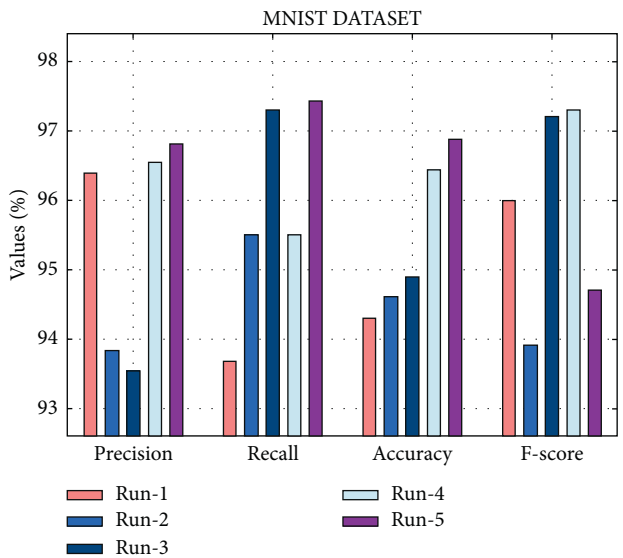

Figure 4: Result analysis of DLFM-CMDFC model on MNIST dataset.

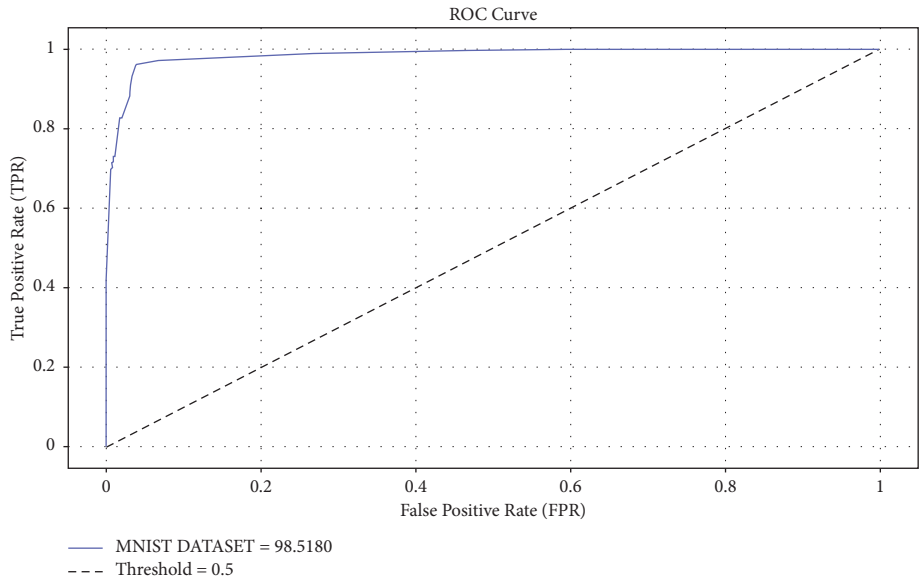

Figure 5: ROC analysis of the DLFM-CMDFC model on the MNIST dataset.

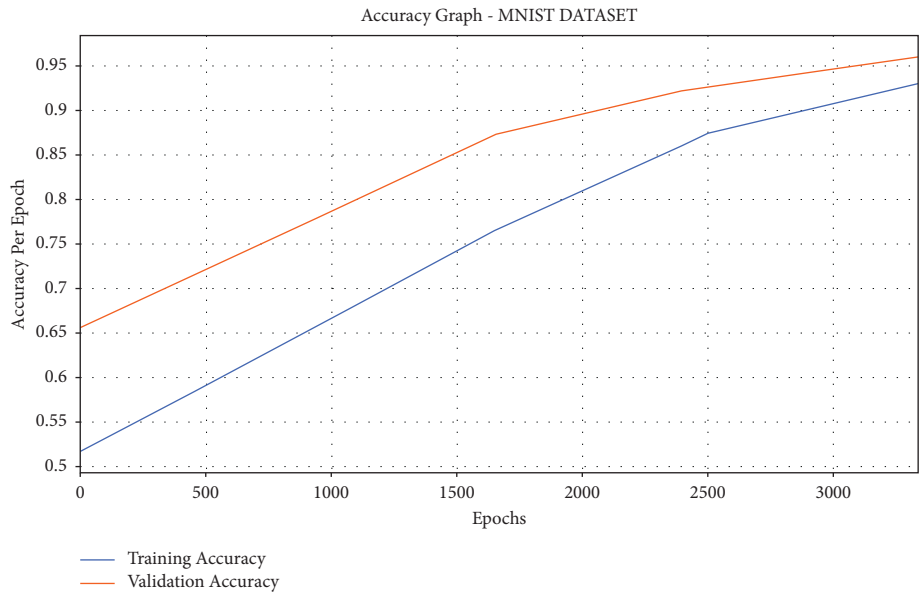

Figure 6: Accuracy analysis of the DLFM-CMDFC model on the MNIST dataset.

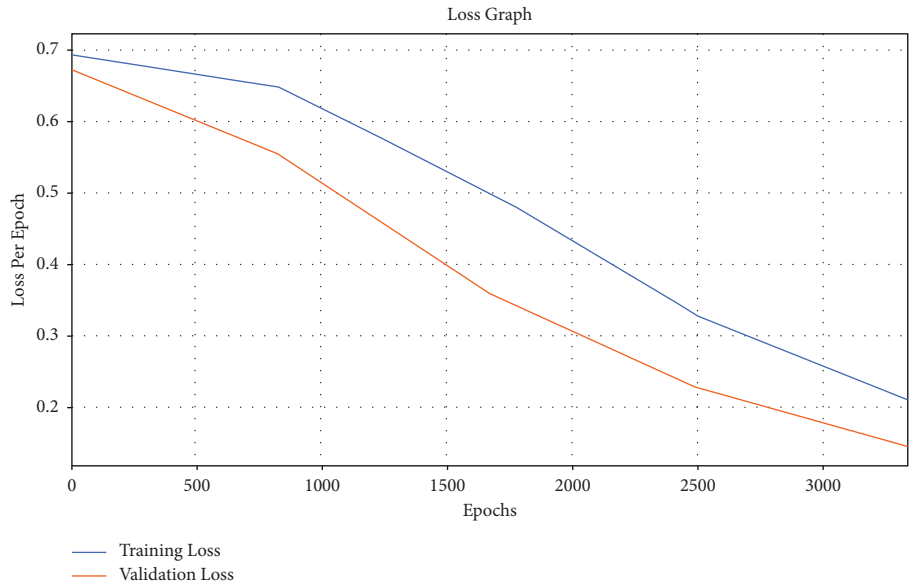

Figure 7: Loss analysis of the DLFM-CMDFC model on the MNIST dataset.

Table 2: Result analysis of the DLFM-CMDFC model on the CIFAR-10.

No. of runs	Precision	Recall	Accuracy	F-score
Run-1	96.52	96.15	96.36	96.66
Run-2	95.75	97.45	96.90	94.77
Run-3	97.98	96.68	97.00	96.57
Run-4	97.51	95.93	97.02	93.50
Run-5	97.46	96.50	97.35	94.52
Run-6	97.78	96.70	97.20	97.23
Run-7	97.71	96.03	97.22	96.86
Run-8	96.98	96.68	96.00	96.82
Run-9	97.31	95.73	96.82	96.51
Run-10	97.66	96.70	97.55	97.17
Average	**97.27**	**96.46**	**96.94**	**96.06**

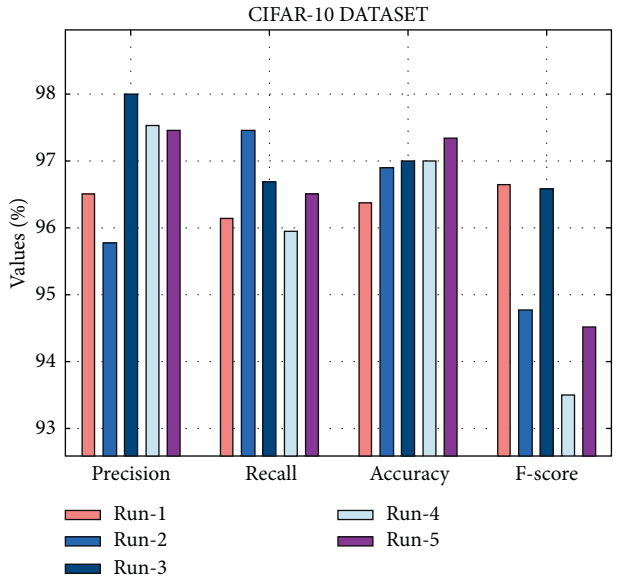

Figure 8: Result analysis of the CIFAR-10 model on the DLFM-CMDFC dataset.

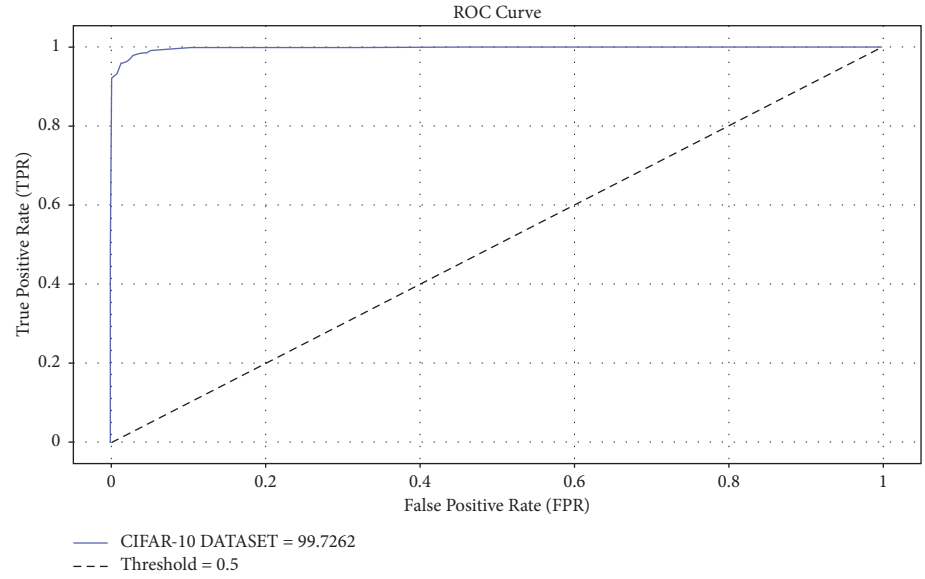

Figure 9: ROC analysis of the DLFM-CMDFC model on the CIFAR-10 dataset.

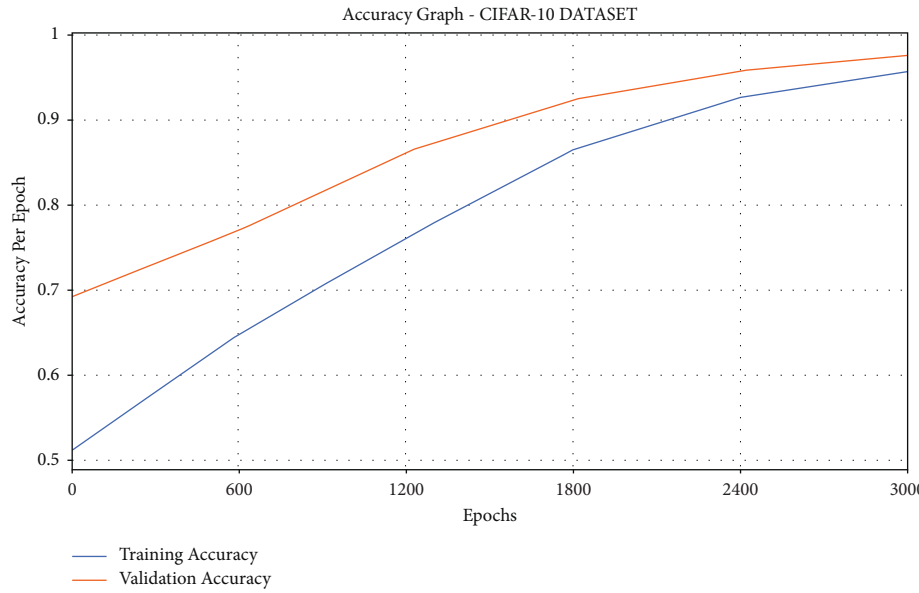

Figure 10: Accuracy analysis of the DLFM-CMDFC model on the CIFAR-10 dataset.

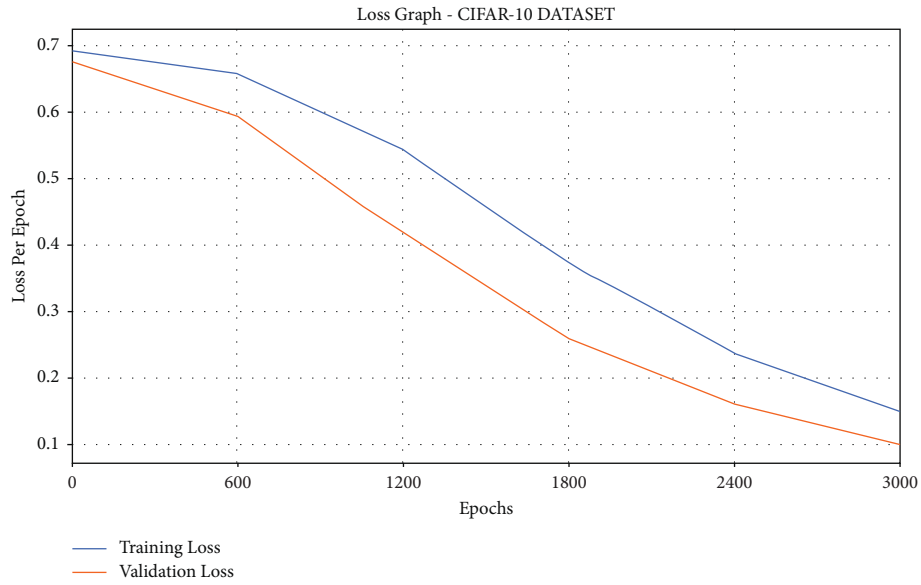

Figure 11: Accuracy analysis of THE DLFM-CMDFC model on THE CIFAR-10 dataset.

Table 3: Comparative analysis of DLFM-CMDFC mode with existing techniques.

Methods	Precision	Recall	F-score
CMFD	57.34	49.39	49.26
IFD-AOS-FPM	53.90	83.27	54.39
CMFD-BMIF	54.40	80.20	59.43
BB-KB-ICMFD	56.62	80.40	60.55
CMFD-GAN-CNN	69.63	80.42	88.35
DLFM-CMDFC	97.27	96.46	96.06

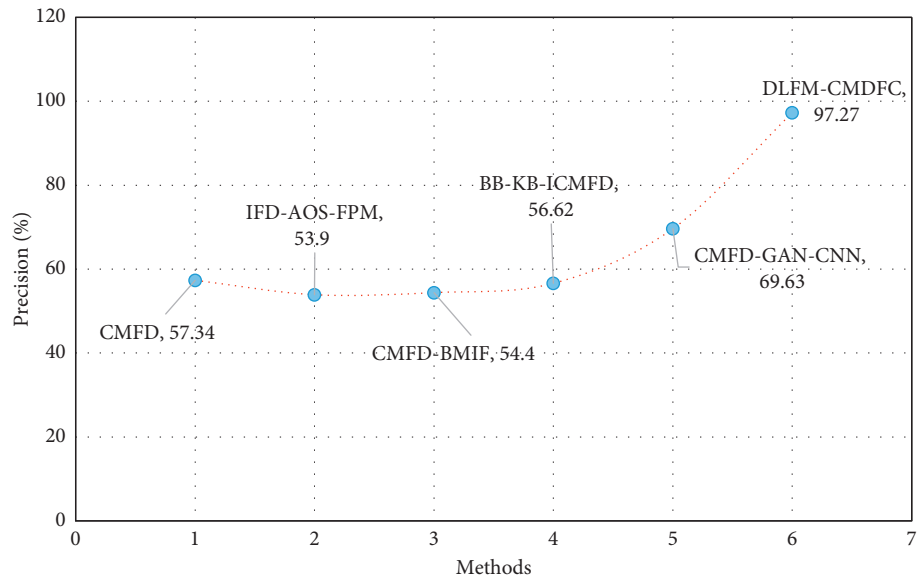

Figure 12: Precision analysis of DLFM-CMDFC technique with existing manners.

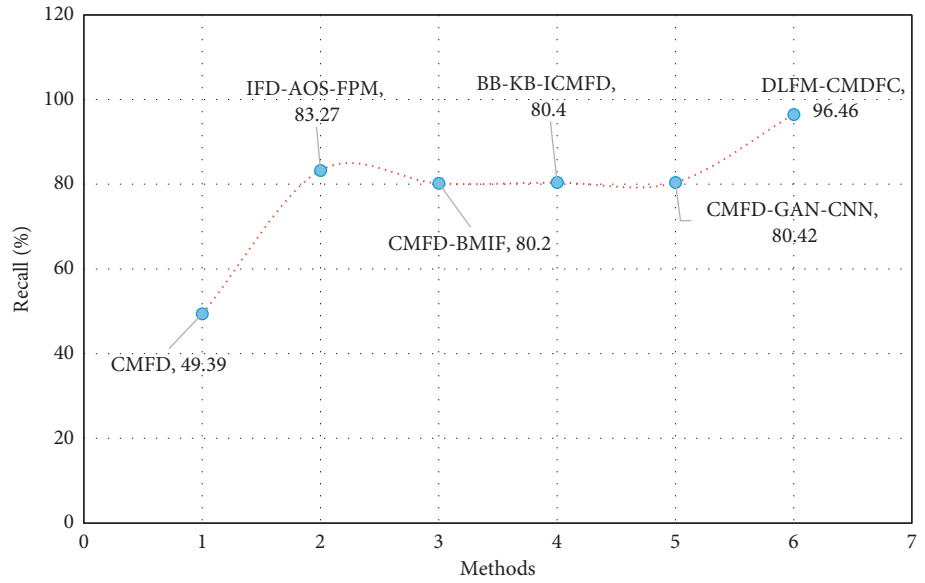

Figure 13: Recall analysis of DLFM-CMDFC technique with existing manners.

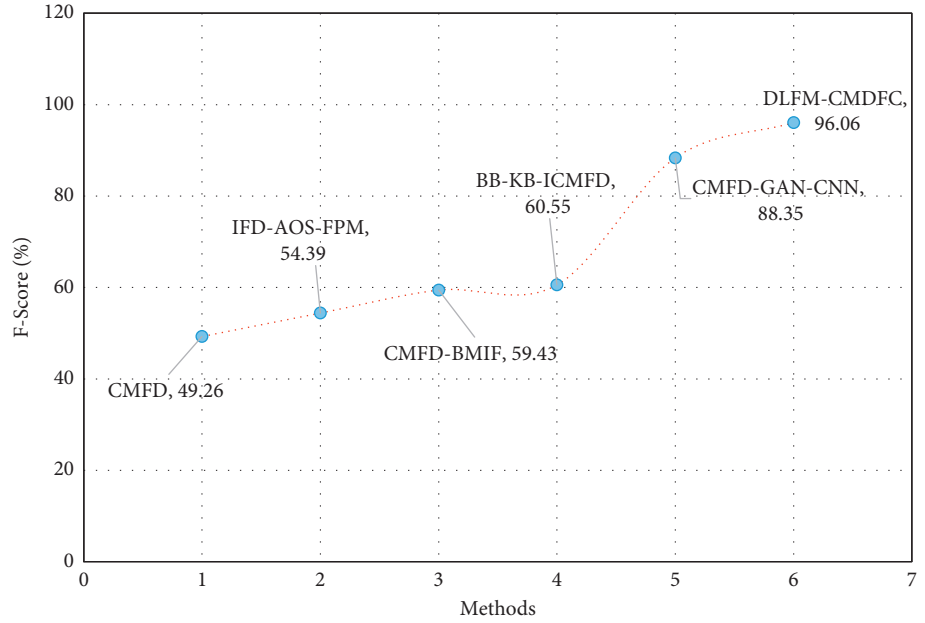

Figure 14: F-Score analysis of DLFM-CMDFC model with existing manners.

5. Conclusion

This article has presented an automated copy-move forgery detection and localization model, named DLFM-CMDFC. The proposed DLFM-CMDFC technique encompasses the fusion of GAN and DenseNet models. In DLFM-CMDFC technique, the two outcomes are combined into a layer to define the input vectors with the initial layer of the ELM classifier. Moreover, the optimal parameter tuning of the ELM technique takes place by the use of AFSA. The outcomes of the networks are fed as input to the merger unit. Lastly, the difference between the input and targets areas is identified in a forged image. The performance validation of the proposed manner takes place using two benchmark datasets. The proposed research work outperforms with 97.27% of precision, 96.46% of recall, and 96.06% of F-score. The experimental outcomes pointed out the supremacy of the proposed technique on the recently developed approaches. As a part of future scope, the detection performance can be improved by the use of generative adversarial network (GAN) model.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] R. Thakur and R. Rohilla, "Recent advances in digital image manipulation detection techniques: a brief review," *Forensic science Bar International Series*, vol. 312, Article ID 110311, 2020.

[2] 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, Article ID 115778, 2020.

[3] G. Gani and F. Qadir, "A robust copy-move forgery detection technique based on discrete cosine transform and cellular automata," *Journal of Information Security and Applications*, vol. 54, Article ID 102510, 2020.

[4] A. Badr, A. Youssif, and M. Wafi, "A robust copy-move forgery detection in digital image forensics using SURF," in *Proceedings of the 2020 eighth international symposium on digital forensics and security (ISDFS)*, pp. 1–6, Beirut, Lebanon, June 2020.

[5] S. Tinnathi and G. Sudhavani, "An efficient copy move forgery detection using adaptive watershed segmentation withAGSO and hybrid feature extraction," *Journal of Visual Communication and Image Representation*, vol. 74, Article ID 102966, 2020.

[6] H. Li, W. Luo, X. Qiu, and J. Huang, "Image forgery localization via integrating tampering possibility maps," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 5, pp. 1240–1252, 2017.

[7] D. Güera, F. Zhu, S. K. Yarlagadda, S. Tubaro, P. Bestagini, and E. J. Delp, "Reliability map estimation for CNN-based camera model attribution," in *Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 964–973, Lake Tahoe, NV, USA, March 2018.

[8] L. Bondi, S. Lameri, D. Güera, P. Bestagini, E. J. Delp, and S. Tubaro, "Tampering detection and localization through clustering of camera-based CNN features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1855–1864, Honolulu, HI, USA, July 2017.

[9] H. Yao, M. Xu, T. Qiao, Y. Wu, and N. Zheng, "Image forgery detection and localization via a reliability fusion map," *Sensors*, vol. 20, no. 22, p. 6668, 2020.

[10] Y. Abdalla, M. T. Iqbal, and M. Shehata, "Copy-move forgery detection and localization using a generative adversarial network and convolutional neural-network," *Information*, vol. 10, no. 9, p. 286, 2019.

[11] B. Diallo, T. Urruty, P. Bourdon, and C. Fernandez-Maloigne, "Robust forgery detection for compressed images using CNN supervision," *Forensic Science International: Reports*, vol. 2, Article ID 100112, 2020.

[12] Y. Rodriguez-Ortega, D. M. Ballesteros, and D. Renza, "Copy-move forgery detection (CMFD) using deep learning for image and video forensics," *Journal of Imaging*, vol. 7, no. 3, 2021.

[13] A. Doegar, M. Dutta, and K. Gaurav, "Cnn based image forgery detection using pre-trained alexnet model," *International Journal of Computational Intelligence & IoT*, vol. 2, no. 1, 2019.

[14] F. Marra, D. Gragnaniello, L. Verdoliva, and G. Poggi, "A full-image full-resolution end-to-end-trainable CNN framework for image forgery detection," *IEEE Access*, vol. 8, Article ID 133488, 2020.

[15] A. Dixit and S. Bag, "Adaptive clustering-based approach for forgery detection in images containing similar appearing but authentic objects," *Applied Soft Computing*, vol. 113, Article ID 107893, 2021.

[16] X. Bi, Z. Zhang, and B. Xiao, "Reality transform adversarial generators for image splicing forgery detection and localization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14294–14303, Nashville, TN, USA, March 2021.

[17] A. Ghai, P. Kumar, and S. Gupta, "A deep-learning-based image forgery detection framework for controlling the spread of misinformation," *Information Technology & People*, vol. 23, 2021.

[18] Y. Rao, J. Ni, and H. Xie, "Multi-semantic CRF-based attention model for image forgery detection and localization," *Signal Processing*, vol. 183, Article ID 108051, 2021.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 2, 2014.

[20] S. R. Abdani and M. A. Zulkifley, "Densenet with spatial pyramid pooling for industrial oil palm plantation detection," in *Proceedings of the 2019 International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE)*, pp. 134–138, IEEE, Bali, Indonesia, December 2019.

[21] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, Honolulu, HI, USA, July 2017.

[22] A. S. Hashmi and T. Ahmad, "GP-ELM-RNN: Garson-pruned extreme learning machine based replicator neural network for anomaly detection," *Journal of King Saud University-Computer and Information Sciences*, 2019, In press.

[23] X. L. Li, Z. J. Shao, and J. X. Qian, "Geothermal-solar combined organic rankine cycle power generation technology research," *System Engineering Theory and Practice*, vol. 35, no. 11, pp. 32–38, 2002.

[24] J. Li and P. Dong, "Global maximum power point tracking for solar power systems using the hybrid artificial fish swarm algorithm," *Global Energy Interconnection*, vol. 2, no. 4, pp. 351–360, 2019.