ISSN: 2096-3246
Volume 57, Issue 02, October, 2025

Multi-Face Deep Fake Recognition based on Fractional Secretary Bird Skill Optimization-enabled hybrid Deep Learning approach using Federated Learning

Stephy Joy D[1] and Dr.R.Thirumalai Selvi[2]

[1] Research Scholar, PG & Research Department of Computer Science, Government Arts College, Nandanam, Chennai.
[2] Associate Professor, PG & Research Department of Computer Science, Government Arts College, Nandanam, Chennai.
steffybala@gmail.com[1], sarasselvi@gmail.com[2]

Abstract

Over the past few years, the quick development in Artificial Intelligence (AI) has devised novel techniques to manipulate multimedia. The misuse of a face swap approach named deepfake has created various cybercrimes like the spreading of fake news, identity theft, and financial crime. One promising countermeasure in opposition to deepfakes is termed as deepfake detection. Still, the detection of deepfake is complex due to the larger dataset. To resolve such issues, this paper develops the Fractional Secretary Bird Skill Optimization Algorithm-enabled Pyramid Deep Belief Network (FSBSOA-PyramidFDBNet)-based multi-face deepfake detection using Federated Learning (FL). The nodes and servers are the major parts of FL. In the training model, the video frames are subjected to face detection. The facial action units are detected with the utilization of Action Unit Network (AUNet). The feature extraction extracts the required features and the deep fake detection is done using the proposed FSBSOA-PyramidFDBNet. The updated weight from the local nodes is aggregated at the server. In addition, the FSBSOA-PyramidFDBNet-based multi face deepfake detection attained the optimal accuracy, loss function, Mean Square Error (MSE), True Negative Rate (TNR), and True Positive Rate (TPR) of 93.91%, 0.064, 0.179, 94.16%, and 92.35%.

Keywords: Deep Belief Network, PyramidNet, Skill Optimization Algorithm, Secretary Bird Optimization Algorithm, Deepfake recognition.

1. Introduction

The current developments in computer-enabled editing generates the synthesizing and varying media in a simple way. With the ongoing rise in DL and computer hardware, image synthesis has been an enormous advance. Face alteration videos name Deepfake have increased on social media, and video-sharing sites [7]. The probability of misinformation has risen severely, especially with the phenomenon termed as Deepfake. Owing to the improvement of social media and smartphones, the deepfake approach poses numerous hazards to digital material and transfers deceptive information, which is not identified by human eyes [8]. Deepfakes include several utilizations in the domains of education, entertainment, and politics, but they create considerable ethical concerns and security. In the Deepfake system, a DL model generates the fake videos, changes the old video, and even changes someone's voice. It is considered as a potentially destructive weapon in malevolent applications aimed for propagating fake news and spreading dangerous or incorrect information. The face-swapping videos create more risk to societal stability, personal privacy, and national security [1]. Despite the fact that a deepfake image may mislead human vision, videos have visible elements to distinguish deepfakes and genuine ones [2]. Deepfake is a kind of artificial media, in which the fake contents are generated using the existing content. The fake content includes audio signals, face-swapping techniques, and video visuals. Particularly, Deepfakes are created by artificial intelligence (AI)-enabled global adversarial networks. Hence, the detection of deepfake media is a complex task with a considerable demand in digital forensics [5].

Nowadays, the creation of deepfakes is immoral and a serious crime. The Deepfakes is a part of various cybercrimes such as theft, cyber extortion, spreading of fake news, imposter fraud, financial fraud, inciting violence, blackmailing, cyberbullying, democratic elections, and so on. Owing to the large count of current

Deepfake videos and their growing authenticity, the manual distinguishing between actual and fake videos is highly challenging. Deepfakes include modified videos or images that show the person from different angles. As a result, automatic Deepfake detection technologies are needed to identify deepfakes to protect individuals from these kinds of cyber threats [9]. Numerous undesirable incidences of deepfakes have arisen on social media platforms, which increases the concerns about the prevalent misuse of public [13]. Deepfake includes additional concerns in political management systems, circumvention of biometric protection, and the creation of fake pornographic. Hence, the deepfake detection methods are employed as effective tools for differentiating the fake from genuine media [2]. They hope to keep the organizations and individuals from the risk of fake information. Deepfake detection is a form of a binary categorization, which contains the leveraging of Machine Learning (ML), forensic analysis, and advances in computer vision for detecting the key indicators of image manipulation. Moreover, these methods distinguish the manipulated and authenticated facial images [3]. Deepfake detection entails the estimation of anomalies and artifacts, that are not noticeable to the naked eye, but the AI-based detectors can identify them. These detection models typically solve the temporal and spatial irregularities [12].

Deepfake is employed for describing the numerous forms of face-altering methods, which utilize cutting-edge approaches such as DL, and computer vision techniques. In addition, Face modification is classified into four classes like full attribute manipulation, face synthesis, identity swap, and expression swap. One of the commonly employed deepfake videos is identity swap, which is also named as face swap. In face swap, the faces of the people are modified to those of the targeted person [11]. There are several significant strategies are used to recognize the fraudulent images. In DL-based detection, image frames are gathered from a video clip and applied to the network for detecting the authenticity. Nevertheless, these models frequently utilize lower generalization power, which resulting less performance while facing some kind of deepfake or manipulation approaches. Models like Convolutional Neural Networks (CNNs), VGG16, SqueezeNet, DenseNet, GoogleNet, and ResNet are employed for deepfake identification. Still, the detection accuracy of the models is not superior. As a result, the FL models are introduced for detecting the deepfake. [6]. The utilization of FL addresses the difficulties of DL and other AI-based models. In the FL architecture, every local model could utilize the private data to the training of local model, and the parameters are upgraded to the global server to attain the global training. After completing more aggregation, the aggregated update process continued till the training of loss convergences. The FL generates an effectual training method for the local data preservations with better generalization capacity [10].

In this work, the FSBSOA_PyramidFDBNet-based deepfake recognition and detection using FL is performed. Using FL, the local updation on every node is done using the local data. Afterward, the updated data are fed to the server, and the global models download the data at every node. The training is done by the downloaded model and the iteration is performed on several interval. In training, the video is fed to frame extraction, and then face detection is accomplished by YOLO v3-Tiny that detects the face. Following that, AU is detected by the AUNet. The relevant feature is extracted for generating the feature vector. Finally, the deep fake recognition is done using the PyramidFDBNet, and its hyperparameters are trained by the proposed FSBSOA. At last, the weighted average technique modifies the updating and aggregates the weights.

The contribution of this article is illustrated as follows:

- **Proposed FSBSOA_PyramidFDBNet-multi-face deepfake recognition using FL:** The FL-based multi-face deepfake is recognized using the proposed FSBSOA_PyramidFDBNet, where the hyperparameter of PyramidFDBNet is trained by the FSBSOA. The merging of Skill Optimization Algorithm (SOA), Secretary Bird Optimization Algorithm (SBOA), and Fractional Calculus creates the FSBSOA.

The paper is arranged in five sections: The motivations, and review deepfake detection techniques are given in section 2, and the proposed FSBSOA_PyramidFDBNet-based deepfake recognition is described in section 3. Moreover, the experimental outcome of FSBSOA_PyramidFDBNet-based deepfake recognition using FL is illustrated in section 4, and section 5 describes the conclusion.

2. Motivation

Deepfakes are generated by the DL approach, which replaces or manipulates the original face of the person from the video or image. Deepfake recognition is crucial, since the deepfakes threaten secure information, manipulate its originality, and spread disinformation. This motivates the researchers to devise the hybrid optimization-based DL model to detect and recognize deepfakes.

2.1 Literature review

Alnaim, N.M., *et al.* [1] devised an Inception-ResNet-v2-based detection of deepfake. It detected the module detected fake videos from a wide range of databases with various forms of fakeness. Still, it was complex to detect the fakeness of the person wearing face masks. Khalifa, A.H., *et al.* [2] developed a dual-scale large receptive field network (DSLRFN) for detecting the deepfake. In the model, the unified Gabor created the circular and linear Gabor filters. The DSLRFN included self-attention, dual-scale convolutional, classifier, and advanced embedding sections. The model diminished the dimensionality and enhanced the reliability. However, the sequenced detector was not used to detect the deepfake images owing to the need for temporal data. Awotunde, J.B., *et al.* [3] developed the deepfake detection model using CNN. In this approach, a noticeable degradation score was obtained on the lower step of video compression. Still, this model failed to identify the deepfake from the text and audio input. Coccomini, D.A., *et al.* [4] devised an EfficientNetV2-M-based deepfake detection. This model considered two different setups, in which the first one limited the preprocess and manipulation of video frames. The second setup considered a wider dataset with several manipulation techniques. These techniques were employed to generalize the detection of deepfake in a cross-dataset. However, it failed to attain the availability of resources during the detection.

Qadir, A., *et al.* [5] developed the ResNet-Swish-BiLSTM-based Deepfake detection. This model was fine-tuned and retrained with new data, and it detected new kinds of deepfakes. The model created latency owing to the complex computation, and the real-time implementation was inefficient. Soudy, A.H., *et al.* [6] devised the CNN with a vision transformer for detecting the deepfake. It was effective in extracting the local features like textures, and edges from the image for the identification of the finest manipulations. Still, the model required more resources due to the combination of vision transformers in the training and inference stages. El-Gayar, *et al.* [7] developed deepfake detection using a Graph Neural network (GNN). This approach was employed for handling the complex, and large graphs to analyze the whole video sequences. Still, the training of GNNs needed labeled graph data that was readily available. Saravana Ram, R., *et al.* [8] devised the DBN with a pairwise learning approach for deepfake identification. In the model, the pairwise learning resolved the variation between the successive frames like the unnatural transitions or movements. Nevertheless, it required a large count of pairs to identify the deepfake, which increased the requirement of resources, and computational complexity.

2.2 Major challenges

The complexities faced in the existing works related to deepfake detection are listed below:

- The Inception-ResNet-v2 in [1] extracted hierarchical aspects to detect the artifacts in the deepfake images. Still, the processing duration was high to process the complex and large-scale detection task.

- In [2], the DSLRFN employed a dual-scale receptive model for capturing the global, and local features for the extraction of extract comprehensive features, and improved the capability of distinguishing the fake, and genuine images. Still, this model was suffered due to overfitting problems.

- The CNN [3]-based deep fake detection learned the discriminative features from raw pixels and increased the detection performance in an integrated system. Nevertheless, it needed a wider range of labeled data for generalizing the unseen manipulations of deepfake.

- In [4], the EfficientNetV2-M offered better generalization ability for all the contexts and it offered a minimum false-positive rate. Still, it failed to consider the attention function for allowing the generalization of deepfakes with limited data.

- The identification of deepfake has a widespread attention in the past years. Even though various deepfake detection methods were developed; but it was complex owing to the dissimilarity among various deepfake models and the adjustment of videos throughout the development.

3. Proposed Fractional Secretary Bird Skill Optimization-enabled PyramidFDBNet for Multi-Face Deep Fake Recognition using Federated Learning

Due to the remarkable growth in technology, highly realistic Artificial Intelligence (AI) generated videos termed deepfakes, which control facial attributes and generate modified expressions with remarkable realism. This kind of synthetically created threatens the privacy of the individual and the integrity of the social, and legal systems. Hence, Deepfake recognition is critical to mitigate the communal threats created by manipulated videos. As a result, this work proposes an effective model named FSBSOA_PyramidFDBNet for multi-face deep fake recognition and detection. The entities of the FL are servers, and nodes. In each node, local updation is done by the local data, and the updations are given to the server. Then, the global model downloads the updated local data on every node. The iteration is continued using the downloaded data. The training model includes the following steps: At first, the input video collected from the database [14] is subjected to frame extraction. The extracted frame is passed to the face detection phase, where the face detection is carried out by using YOLO v3-Tiny [15]. After that, the detected face is forwarded to Facial AUs detection, which is done by employing AUNet [16] [28]. Thereafter, feature extraction is employed to extract essential features like Histogram of Oriented Gradients (HOG) [17], statistical features [18], and CNN features [19]. The features are concatenated and the resultant features vector is given to deep fake recognition, where the PyramidFDB Net is employed. Moreover, the PyramidFDB is the combination of the Pyramid Network [20] and Deep Belief Network (DBN) [21], where the proposed FSBSOA is utilized to train the parameters of PyramidFDB. In the server, the modification of local updation and aggregation is done based on the weighted average technique. Figure 1 shows the raphial illustration of FSBSOA_PyramidFDBNet for multi-face deep fake recognition and detection.

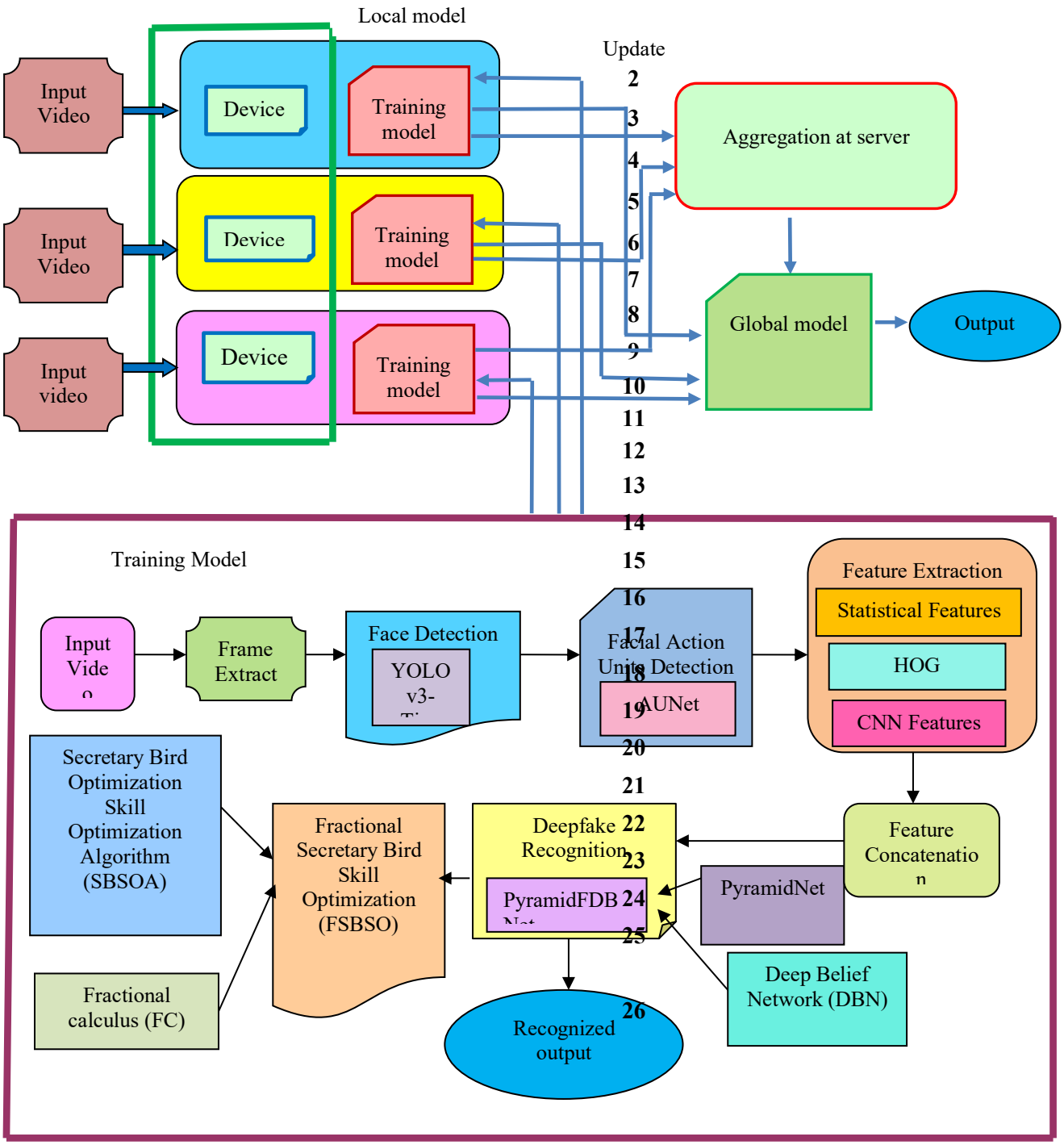


Figure 1.

Block diagram of FSBSOA_PyramidFDBNet-enabled multi-face deep fake recognition

3.1 Training model sent to device

The data training is done in the local node, and transmits the local updates towards the server. The aggregated local data within the global model is used to download the global data till achieving precise constraints.

3.1.1 Training and validation for each local node

The validation of local model is carried out via the local data, and the training is done on every node.

3.1.2 Training model

The deepfake recognition is done in the training model, where the FSBSOA_PyramidFDBNet detects the deepfake. The following processes are done to obtain the multi-face deepfake detection.

3.1.2.1Acquisition of the video sample

The input video sample is collected from the OpenForensics dataset [14]. In the dataset, z counts of videos are presented, which is given as,

$$D = \left\{ D_1, D_2, ..., D_q, ..., D_z \right\} \tag{1}$$

Here, the q^{th} video is represented as D_q. The dataset includes the video from various real-world scenarios with different backgrounds, face orientations, and varying lighting conditions. Moreover, the image shows the facial features with forgery artifacts.

3.1.2.2 Extraction of video frames

Frame extraction is needed for the gathering of particular frames from a video clip. It evaluates every frame separately and finds discrepancies to indicate the presence of deepfake. The analysis of video frames identifies the changes in facial emotions and actions. The extraction of frames is needed to recognize and deteriorate the deepfake content. In the extraction process, the frames are divided as per the count of frames, and the video duration. The following mathematical expression shows the frame extraction,

$$f_q = F_{peS} \times Vid_\tau \tag{2}$$

where, the video duration is denoted by Vid_τ and F_{peS} shows the frames per second. The count of frames extracted from the video D_q is given as,

$$D_q = \left\{ f_1, f_2, \cdots f_q \cdots f_y \right\} \tag{3}$$

where, the q^{th} video in the dataset is specified as D_q, which includes y count of frames. The q^{th} frame in the video D_q is portrayed as f_q.

3.1.2.3 Face detection by YOLO v3-Tiny

Face detection locates and identifies the human faces from the frames. This procedure evaluates the features like nose, mouth, and eyes. Here, the extracted frame f_q is fed to the YOLO v3-Tiny [15] to detect the face. The YOLO v3-Tiny includes the YOLOv3 with a minimal-intensity convolutional layer. In the model, a pooling layer with few convolution layers is presented. The frame is fed to the YOLO v3-Tiny, which divides a video clip. Following that, bounding boxes with lower scores are eliminated. The max-pooling, and convolutional layers are employed as the feed-forward model to extract the face features. The YOLOv3-Tiny includes fewer parameters, and layers compared to YOLOv3; hence, it offers quicker performance with less memory utilization. Furthermore, the face-detected outcome attained from the YOLOv3-Tiny is denoted by L_q.

3.1.2.4 AU detection using AU-Net

AU detection is also termed as Action Unit detection, which detects the movements of exact facial muscles. Here, the AU-Nets [16 [28] is utilized for the detection of AU. The face-detected outcome $\left(L_q\right)$ is fed as the input of AU-Nets, which effectively find the facial actions. Moreover, AU-Net identifies the important parts in the face like the nose, mouth, and eyes. AUNet divides the face into several regions (eyes, cheeks, and lips) that are significant for the detection of facial expressions. In AUNet, the attention function increases

segmentation accuracy, and it enables the essential facial features to find the changes in facial expressions. Due to the attention process, the AUNet effectively detects and segments facial actions. The AU detected outcome attained from the AU detected outcome M_q .

3.1.2.5 Feature extraction from AU detection

Feature extraction is essential to minimize the dimensionality of real information via the reduction of inappropriate information and emphasizing the essential qualities of the image. Here, the HoG, statistical, and CNN feature extraction are done from the AU-detected outcome M_q.

a) HOG feature

For extracting the HoG [17] feature, the histogram is created by dividing the image into small cells. Here, the HoG feature is extracted from the AU-detected outcome M_q, in which every cell is assigned a bin histogram. The division process provides an accurate description of the image, and the image gradients are sensible to lighting. The obtained HOG feature is specified by x_q.

b) Statistical features

After getting the HoG feature x_q, the statistical features [18] like mean, variance, standard deviation, skewness, and entropy are extracted. The mean feature shows the whole count of features to the count of samples. The mean feature is denoted by a term n_1. The deviation of the feature from the average value portrays the variance, which is denoted by n_2. Moreover, the square root of variance is indicated as standard deviation, and it is specified as n_3. Entropy describes the quantity of unpredictability or randomness in the feature, which is represented by the term n_4. The Skewness estimates the asymmetrical distribution of values within a dataset, and it is expressed as n_4. The overall feature vector is expressed as,

$$N_{q1} = \{n_1, n_2, \cdots, n_5\} \tag{4}$$

c) CNN Features

The CNN [19] contains the convolutional, pooling, and fully connected layers. The convolution layer is considered as the prime layer in CNN. Here, the AU-detected outcome M_q is applied to CNN feature extraction. The pooling layer obtains a fixed function for enhancing feature invariance. Moreover, the CNN feature vector N_{q2} is attained through the convolution layer. Figure 2 depicts the CN features.

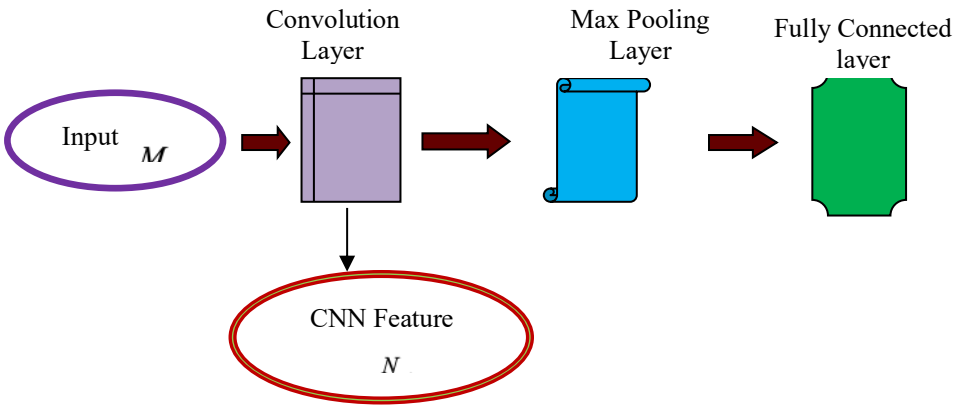


Figure 2. CNN Features

After getting the the statistical, and CNN feature vectors, the concatenated output vector is given as,

$$N_q = N_{q1} \,||\, N_{q2} \tag{5}$$

where, N_{q1} and N_{q2} shows the feature vector comes from the statistical and CNN features.

3.1.2.6 multi-Face Deepfake recognition using FSBSOA_PyramidFDBNet

Deepfakes are considered as fake news about a particular person, which is made using DL techniques. These fake contents negatively impact the privacy and disinformation of society. Hence, it is essential to detect deepfakes to control the spreading of fake information. As a result, the FSBSOA_PyramidFDBNet is developed for detecting the multi-face deepfake. The integration of PyramidNet [20] with DBN [21] creates the PyramidFDBNet. Initially, the frame extracted from the video clip f_q is applied as the input of PyramidNet, wherein the outcome H_1 is obtained. Following that, the output H_1 and the feature vector N_q are fed to the Pyramid Deep Belief layer, where the regression validates the similarity among the extracted, and target features to provide the fused output H_2. Finally, the output H_2 from the PyramidFDBNet layer and the outcome from AU detection $\left(M_q\right)$ considered by a single term MH_q is given to the DBN model, which creates the deepfake recognized outcome H_3. The structure of PyramidFDBNet is exhibited in figure 3.

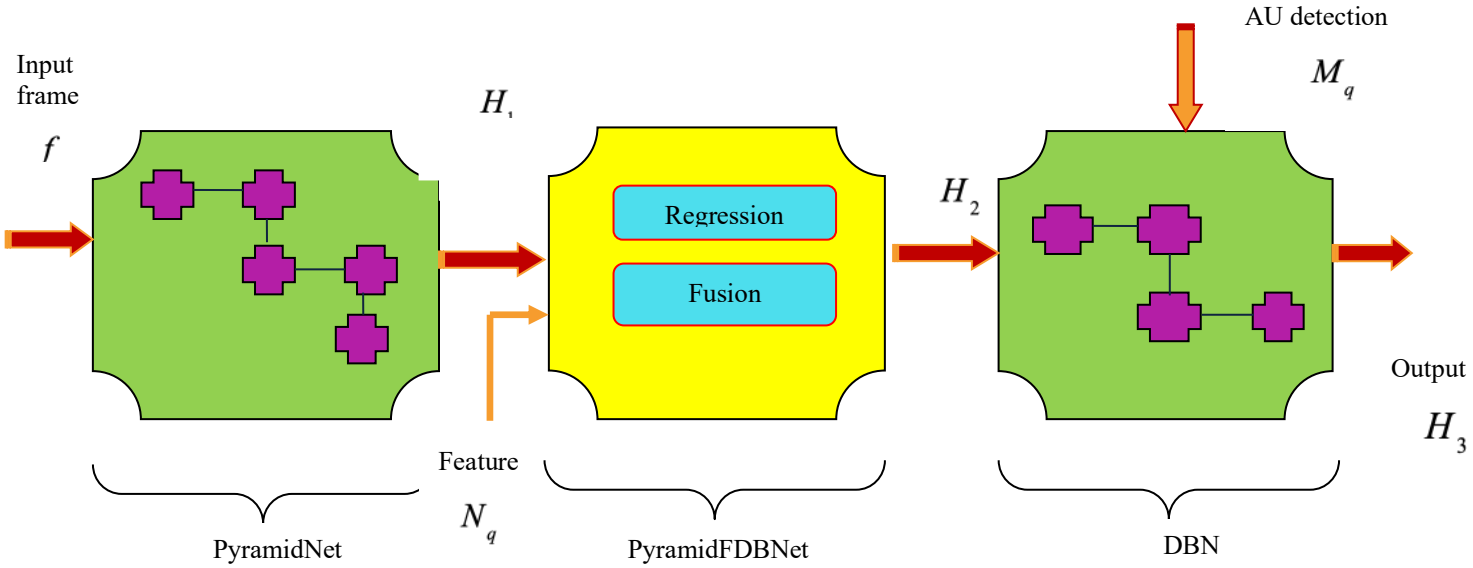


Figure 3. Network design of PyramidFDBNet

a) PyramidNet

The configuration of PyramidNet [20] is similar to the CNN, which collects the features with several resolutions. Here, the frame f_q extracted from the video clip is fed as the input of PyramidNet. The pyramidal model increases the information flow, and controls the size of a residual unit by the downsampling technique. The residual links present in the PyramidNet control the flow of gradients, and lessen the problem of vanishing gradients. The configuration of PyramidNet is adjusted easily via the tuning of the initial count of feature maps, and growth rate. In the PyramidNet model, the blocks are designed via the convolution filter, Batch Normalization (BN), and ReLUs layers. The individual mapping of the PyramidNet is not utilized in the shortcut because the feature map's size. Therefore, a zero-padded shortcut relation is employed in the residual links. Moreover, the pre-activation ResNets are utilized for the resolving of backward gradients. In the PyramidNet, the following outcome is attained.

$$H_1 = P_{(q,d)} f_q + P_{q-1}^d \tag{6}$$

The outcome from the PyramidNet is denoted as M_1, and the frame extracted from the video is indicated by f_q . Moreover, the residual function is represented by $P_{(q,d)}$.

b) PYDB layer

In PyramidFDBNet layer, the output H_1 and the feature vector N_q is fed as the input, where the regression evaluates the correlation between the target and the fusing of feature vector. The output in the g^{th} iteration is given as,

$$B_g = \sum_{\lambda=1}^{u} N_g \times S_\lambda \tag{7}$$

Here, the count of the feature vector is specified as u, and the feature vector is portrayed as N_g. The weight is indicated by S_λ, and the outcome of the PyramidFDBNet layer is given as,

$$H_2 = uB_g + \frac{u}{2} H_1 \tag{8}$$

$$H_2 = u \sum_{\lambda=1}^{u} N_g \times S_\lambda + \frac{u}{2} H_1 \tag{9}$$

$$H_2 = u \sum_{\lambda=1}^{u} N_g \times S_\lambda + \frac{u}{2} \left[P_{(q,d)} f_q + P_{q-1}^d \right] \tag{10}$$

The outcome of PyramidFDBNet ia specified by H_2.

c) DBN

DBN [21] contains the latent variables termed as "hidden units,", which are employed for unsupervised learning. Moreover, DBNs are employed as a generative graphical framework for learning the hierarchical structure of the data, which are essential for pre-training the neural networks. The output H_2 from the

PyramidFDBNet layer and the outcome from AU detection $\left(M_q\right)$ is denoted by a term $MH_q \in \left\{H_2, M_q\right\}$is given to the DBN. The DBN is utilized for back-propagation, identifying the smallest periphery region, and fine-tuning parameters. The DBN is created by the layer-based training model of RBM. In the RBM, the two-layer network detects the Markov random value with hidden and visible units. The outcome of DBN is given as,

$$H_3 = -\sum_{\mu=1}^{f} \hbar_\mu \xi_\mu - \sum_{\varsigma=1}^{N} i_\varsigma j_\varsigma - \sum_{\mu=1}^{F}\sum_{\varsigma=1}^{G} \varpi_\mu \xi_\mu j_\varsigma \tag{11}$$

$$H_3 = -\hbar^{\mathrm{T}} \xi - i^{\mathrm{T}} H_2 - \xi^{\mathrm{T}} W H_2 \tag{12}$$

$$H_3 = -\hbar^{\mathrm{T}} \xi - i^{\mathrm{T}} \left(u \sum_{\lambda=1}^{u} N_g \times S_\lambda + \frac{u}{2}\left[P_{(q,d)} f_q + P_{q-1}^{d} \right] \right) - \xi^{\mathrm{T}} W \left(u \sum_{\lambda=1}^{u} N_g \times S_\lambda + \frac{u}{2}\left[P_{(q,d)} f_q + P_{q-1}^{d} \right] \right) \tag{13}$$

where, $\Phi = \left\{\hbar_\mu, i_\varsigma, \varpi_{\mu\varsigma}\right\}$, and $\varpi_{\mu\varsigma}$ shows the weight for the hidden unit ς, and visible unit μ. The terms $\hbar_\mu$, and i_ς specifies bias, and the outcome from the DBN is denoted by H_3. Tthe outcome from previous layer (PyramidFDBNet) is portrayed as H_2.

3.1.3 Training of PyramidFDBNet using the proposed FSBSOA

The hyperparameter of PyramidFDBNet is trained by the FSBSOA. The merging of SBO [22], SOA [23], and FC [24] develops FSBSOA. SBO is based on the secretary bird's tactics to survive in the surroundings. Secretary birds protect themselves from predators, and continuously seeking for prey. In the exploration stage, the simulation of secretary birds pursuing snakes, while the exploitation function specifies the fight against the predators. In this stage, secretary birds survey their lives and determine the optimal route to reach the safe location. The SBO is employed to solve real-world optimization problems. Moreover, the SBOA controls a massive number of variables and constraints to handle complicated real-world problems. To achieve faster convergence, the SOA is added to the SBOA. Human efforts for gathering and increasing skills are the basis of SOA. The SOA includes the exploration and exploitation. In the initial phase, skill is acquired from experts, while the second phase includes the development of skill by practice as well as the effort of individuals. Furthermore, the SOA offered optimal convergence speed. For resolving the complex mathematical function in the optimization, the FC is added to SBSOA. The FC includes the effectual mathematical methods to solve the differential equations and attain easier computation. The step included in the FSBSOA is described below:

Step 1: Initialization

The SBOA method is based on a metaheuristic system, in which the secretary bird portrays the population member. The location of the member finds the decision variables. The initialization of SBOA is formulated as by $K_{m,v} = \ell_{bou} + b \times \left(\vartheta_{bou} - \ell_{bou}\right), m = 1,...,G, and\ v = 1,2,...,\Re$, where, the location of the m^{th} secretary bird is specified as K_m. The upper and lower bounds are indicated as ϑ_{bou} and ℓ_{bou}. Moreover, the random number b lies in [0, 1]. The candidate solution for the secretary bird is given as,

$$K=\begin{bmatrix} k_{1,1} & \cdots & k_{1,v} & \cdots & k_{1,\Re} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ k_{m,1} & \cdots & k_{m,v} & \cdots & k_{m,\Re} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ k_{G,1} & \cdots & k_{G,v} & \cdots & k_{G,\Re} \end{bmatrix}_{G\times\Re} \tag{14}$$

where, the group of secretary bird is denoted as K, and $K_{m,v}$ shows the m^{th} secretary bird in the v^{th} dimensional space. The group of members are symbolized by the term G, and the problem with numerous dimensions are indicated by $\Re$.

$$J=\begin{bmatrix} J_1 \\ \vdots \\ J_m \\ \vdots \\ J_G \end{bmatrix}_{G\times 1} = \begin{bmatrix} J(K_1) \\ \vdots \\ J(K_m) \\ \vdots \\ J(K_G) \end{bmatrix}_{G\times 1} \tag{15}$$

The objective vector is denoted by J, in which the objective value for m^{th} secretary bird is indicated by J_m .

Step 2: Evaluation of Fitness

The optimal solution is achieved by the value of fitness parameters, which is estimated by the error function as follows,

$$Fitness=\frac{1}{z}\sum_{q=1}^{z}\left(H_q^*-H_3\right)^2 \tag{16}$$

Here, the he expected outcome is specified by H_q^*, and the output of PyramidFDBNet is given as H_3.

Step3: Updated equation of FSBSOA

To attain faster convergence, the SOA [23] is integrated with the exploration phase (escaping tactics) of SBO. Hence, the updated position for SBSOA is expressed as,

$$k_{m,v}(r+1)=\frac{b*C_{m,v}(I_2O-1)+I_2*k_{Ran}(1-bA)}{(I_2O-bA)} \tag{17}$$

where, the random numbers b lies within [0, 1], and A lies in between of {1,2}. The term $C_{m,v}$ indicates the selected experts to train the m^{th}candidate in the v^{th}dimensional solution space. Moreover, I_2 implies the randomly created array of size $1\times\Re$and k_{Ran} shows the random candidate solution in the present iteration. Furthermore, O portrays the randomly selected integer 1 or 2. Thus, the upgraded equation of SBSOA is obtained from Eq. (17).

To solve the complex computation, the FC [24] is added to the updated expression of SBSOA. Subtract $k_{m,v}(r)$ on both sides of Eq. (17)

$$k_{m,v}(r+1)-k_{m,v}(r)=\frac{b*C_{m,v}(I_2O-1)+I_2*k_{Ran}(1-bA)}{(I_2O-bA)}-k_{m,v}(r) \tag{18}$$

Applying FC [24], then the equation becomes,

$$Q^t\left[k_{m,v}(r+1)\right]=\frac{b*C_{m,v}(I_2O-1)+I_2*k_{Ran}(1-bA)}{(I_2O-bA)}-k_{m,v}(r) \tag{19}$$

$$\begin{aligned}&k_{m,v}(r+1)-t.k_{m,v}(r)-\frac{1}{2}t.k_{m,v}(r-1)-\frac{1}{6}(1-t).k_{m,v}(r-2)\\&-\frac{1}{24}t(1-t)(2-t).k_{m,v}(r-3)=\frac{b*C_{m,v}(I_2O-1)+I_2*k_{Ran}(1-bA)}{(I_2O-bA)}-k_{m,v}(r)\end{aligned} \tag{20}$$

$$\begin{aligned}&k_{m,v}(r+1)=t.k_{m,v}(r)+\frac{1}{2}t.k_{m,v}(r-1)+\frac{1}{6}(1-t).k_{m,v}(r-2)\\&+\frac{1}{24}t(1-t)(2-t).k_{m,v}(r-3)+\frac{b*C_{m,v}(I_2O-1)+I_2*k_{Ran}(1-bA)}{(I_2O-bA)}-k_{m,v}(r)\end{aligned} \tag{21}$$

$$\begin{aligned}&k_{m,v}(r+1)=k_{m,v}(r)[t-1]+\frac{1}{2}t.k_{m,v}(r-1)+\frac{1}{6}(1-t).k_{m,v}(r-2)\\&+\frac{1}{24}t(1-t)(2-t).k_{m,v}(r-3)+\frac{b*C_{m,v}(I_2O-1)+I_2*k_{Ran}(1-bA)}{(I_2O-bA)}\end{aligned} \tag{22}$$

$$k_{m,v}(r+1)=\frac{\left[\begin{array}{l}k_{m,v}(r)[t-1]+\frac{1}{2}t.k_{m,v}(r-1)+\frac{1}{6}(1-t).k_{m,v}(r-2)\\+\frac{1}{24}t(1-t)(2-t).k_{m,v}(r-3)\end{array}\right](I_2O-bA)+b*C_{m,v}(I_2O-1)+I_2*k_{Ran}(1-bA)}{(I_2O-bA)} \tag{23}$$

Hence, the updated solution for FSBSOA is obtained from Eq.(23), where, t implies the derivative order.

Step 4: Re-Evaluation of fitness

The fitness function is re-computed until getting the superior solution.

Step 5: Termination

The FSBSOA reaches the termination until the maximum count iterations are performed for detecting the

optimal result. Algorithm 1 shows the Pseudocode for FSBSOA.

Algorithm 1. Pseudocode for FSBSOA

SL. No	Pseudocode for FSBSOA
1	**Input:** Population members, current iteration (r), and maximum count of iteration (R)
2	**Output**: The optimal solution $k_{m,v}(r+1)$
3	Initiate the population
4	For $r=1:R$ do
5	Upgrade the secretary bird K
6	For $m=1:G$ do
7	Exploration
8	If $r<\frac{1}{3}R$
9	Compute the upgraded position of a secretary bird in the prey-searching period
10	Upgrade the m^{th} secretary bird
11	Else If $\frac{1}{3}R<r<\frac{2}{3}R$
12	Compute the upgraded position of a secretary bird in the prey-consuming period
13	Upgrade the m^{th} secretary bird
14	Else
15	Compute the upgraded position of a secretary bird in the prey attacking period
16	Upgrade the m^{th} secretary bird
17	End if
18	Exploitation
19	Compute the upgraded expression for FSBSOA using Eq.(23)
20	End if
21	Upgrade the m^{th} secretary bird
22	End for $m=1:G$
23	Store the optimal candidate solution
24	End $r=1:R$
25	Attain the superior solution
26	Return

The proposed FSBSOA_ PyramidFDBNet offered the effectual accuracy for multi-face deepfake recognition due to the optimal training of PyramidFDBNet using FSBSOA.

3.2 Aggregation on the Server

The aggregation of weights from various local models is done in the server. The Conditional Autoregressive

Value at Risk by Regression (CAViar) [25] carries out aggregation and the weights are averaged. Here, the local models X_1 and X_2 are considered, in which the weights from the local models 1, and 2. are denoted by ψ_1 and ψ_2. The schematic representation of data aggregation is depicted in figure 4.

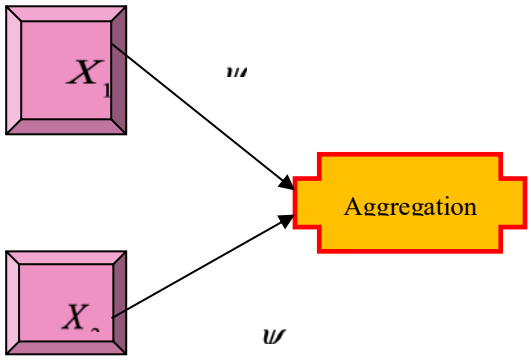


Figure 4. Schematic view of data aggregation

Applying CAViaR [25], the obtained weights are formulated as,

$$\psi_{global} = \sigma_0 + \sigma_1\psi_1(r-1) + \sigma_2\psi_2(r-2) + \sigma_1\, fun\left(\psi_1(r-1)\right) + \sigma_2\, fun\left(\psi_2(r-2)\right) \quad (24)$$

The local nodes X_1 and X_2 contain the weight on the iteration $(r-1)$, and $(r-2)$ are denoted by $\sigma_1\psi_1(r-1)$ and $\sigma_2\psi_2(r-2)$. The vector of the unknown parameter is indicated by σ. Furthermore, the fitness of local nodes X_1 and X_2 at the iterations $(r-1)$, and $(r-2)$ are specified as $\sigma_1\, fun\left(\psi_1(r-1)\right)$ and $\sigma_2\, fun\left(\psi_2(r-2)\right)$.

3.3 Apply global training model on every local nodes

The average weights are upgraded to the server, and iteration is repeated over and over until the optimal result is achieved.

4. Result and discussion

The experimental outcome of FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL is described here. In addition, the dataset considered, implementation tool, the assessment are described.

4.1 Experimental setup

The implementation of FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL is done using the PYTHON tool.

4.2 Dataset description

The OpenForensicsd dataset [14] is utilized in this paper. The dataset is generated for the detection and segmentation of face forgery in numerous real-world scenarios. This dataset includes numerous images, each of which includes various faces, some of which have been changed by the forgery techniques. The major goal for developing this dataset is to offer testing algorithms, which precisely find and segment the fake faces.

4.3 Experimental Outcomes

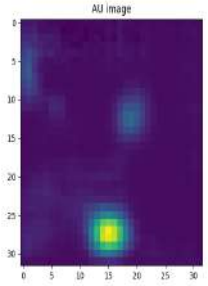
The experimental outcomes for FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL is shown in figure 5. Here, the input, face detected outcome for frames 1, and 2 are illustrated in figures 5 a), 5 b), 5 c), and 5 d). Moreover, the AU detected, and the detected outcome images from frames 1, and 2 are portrayed in figures 5 e), 5 f), 5 g), and 5 h).

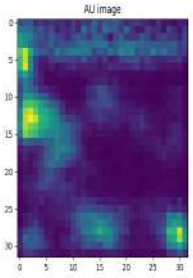
(a)

(b)

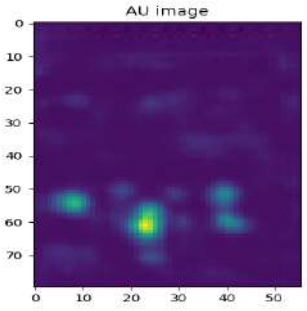
(c)

(d)



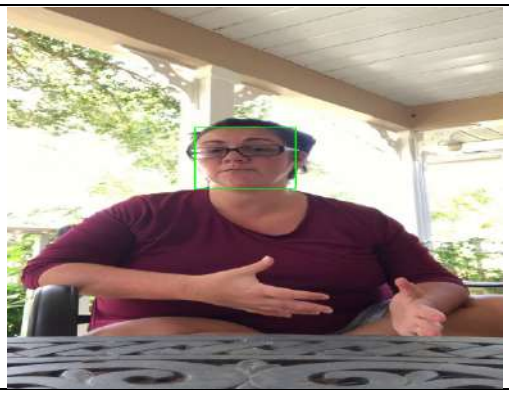


(e)



(f)

(g)



(h)

Figure 5. Experimental outcomes, a) input frame-1, b) input frame-2, c) face detected images -1, d) face detected image -2, e) AU detected image-1, f) AU detected image -2, g) detected output images -1 with deepfake and h) detected output image-2 without deepfake

4.4 Performance metrics

The following metrics are used to compute the performance of FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL.

4.4.1 Accuracy

Accuracy [26] describes the number of precise predictions to the entire predictions, which is given as,

$$Accuracy = \frac{U_{PO} + V_{PO}}{U_{PO} + U_{NE} + V_{PO} + V_{NE}} \tag{25}$$

where, U_{PO} and V_{PO} refers to true positives and true negatives, U_{NE} and V_{NE} represents false positive and false negatives.

4.4.2 Loss function

The loss function is defined as the error function. Moreover, the lower loss function offered the optimal solution.

4.4.3 MSE

MSE [27] is defined as the average of square of deviation between the expected and the real outcome. The MSE is formulated as,

$$Fitness = \frac{1}{z}\sum_{q=1}^{z}\left(H_q^* - H_3\right)^2 \tag{26}$$

where, the expected outcome is denoted by H_q^*, and the outcome of PyramidFDBNet is given as H_3.

4.4.4TPR

TPR [26] shows the detected positive output among the entire positive. The mathematical expression for TPR is given by,

$$TPR = \frac{U_{PO}}{U_{PO} + V_{NE}} \tag{27}$$

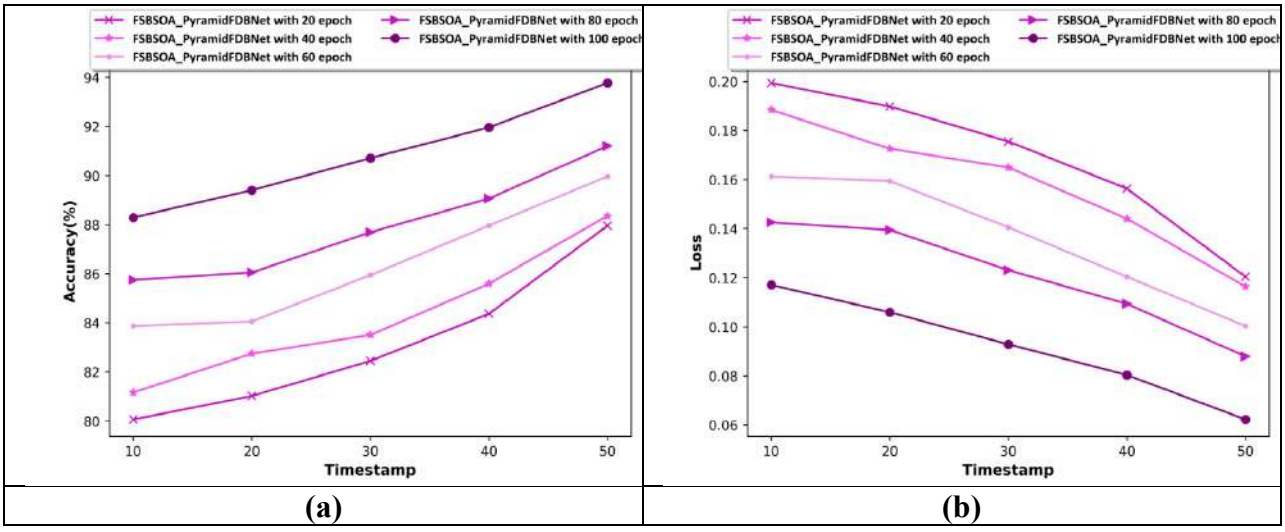
4.4.5 TNR

TNR [26] defines the detected negative output among the overall negatives. The TNR is mathematically expressed as,

$$TNR = \frac{U_{NE}}{U_{NE} + V_{PO}} \tag{28}$$

4.5 Performance analysis

The performance assessment of FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL is deliberated in figure 6. The effectiveness of FSBSOA_PyramidFDBNet is validated via the variation of time stamp with respect to the metrics. Figure 6 a) shows the analysis regarding accuracy. For the time stamp of 50, the FSBSOA_PyramidFDBNet attained the accuracy of 87.95%, 88.36%, 89.97%, 91.20%, and 93.77% under the epoch of 20, 40, 50, 80, and 100. The estimation concerning the loss function is given in figure 6 b). While considering the time stamp of 50, the loss function of FSBSOA_PyramidFDBNet for epoch 20 is 0.120, 40 is 0.116, 60 is 0.100, 80 is 0.087, and 100 is 0.062. Figure 6 c) shows the analysis concerning MSE. With the time stamp of 50, the FSBSOA_PyramidFDBNet achieved the MSE regarding the epochs 20, 40, 60, 80, and 100 are 0.307, 0.270, 0.236, 0.215, and 0.179. The assessment in connection with TNR is shown in figure 6 d). The FSBSOA_PyramidFDBNet got the TNR with respect to the epochs 20, 40, 60, 80 are 86.26%, 87.86%, 89.95%, 92.25%, and 94.67% under the time stamp 0f 50. Furthermore, the estimation related to TPR is portrayed in figure 6 e). While considering the time stamp=50, the TPR of FSBSOA_PyramidFDBNet are 84.26%, 85.33%, 87.90%, 89.33%, and 92.20% under the for the epochs of 20, 40, 60, 80, and 100.



(a) **(b)**

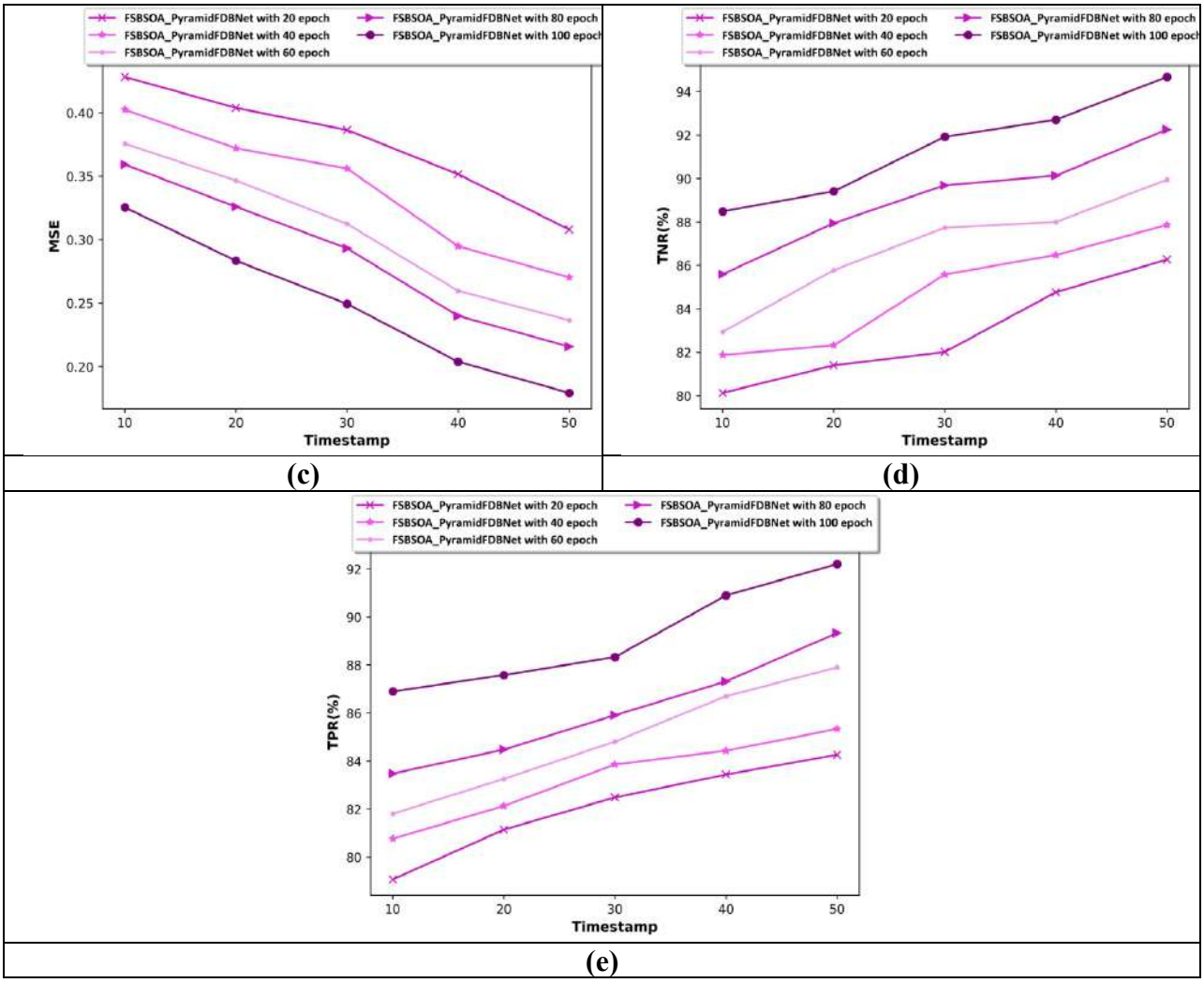


Figure 6. Performance assessment, a) Accuracy, b) Loss function, c) MSE, d) TNR, c) TPR

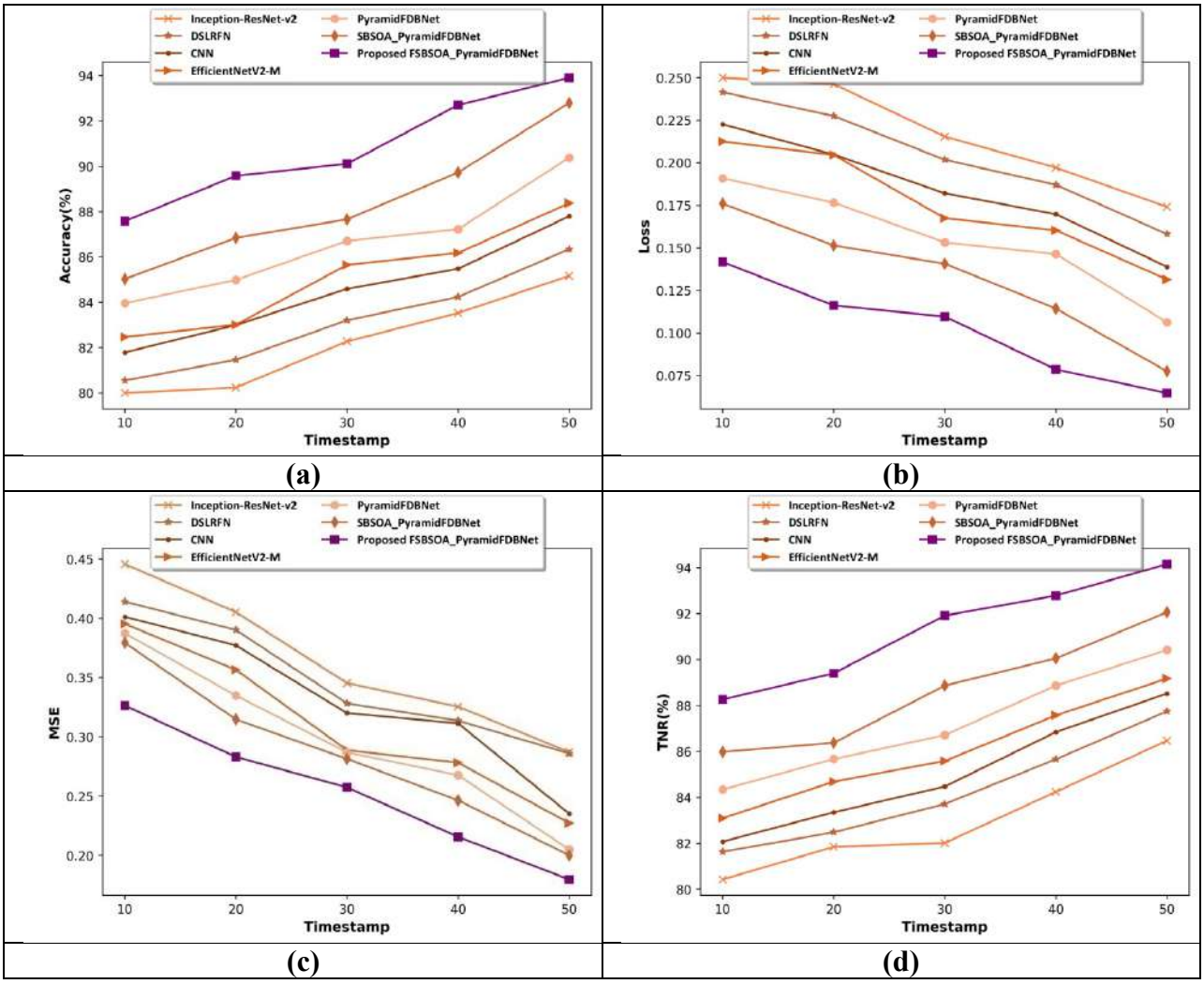
4.6 Comparative analysis with existing and proposed techniques

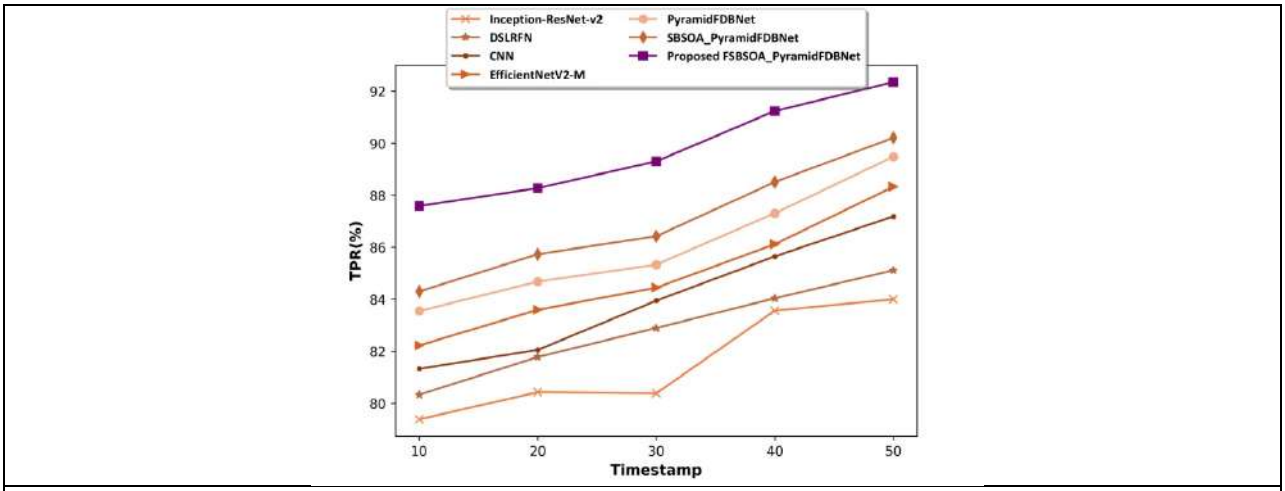
The techniques like Inception ResNet-v2 [1], DSLRFN [2], CNN [3], EfficientNetV2-M [4], PyramidFDBNet, SBSOA_PyramidFDBNet are considered as the existing methods to validate the efficiency of FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL.

4.7 Comparative assessment

Figure 7 shows the comparative analysis of FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL. Here also the assessment is attained by varying the time stamp from 10 to 50 with respect to the evaluation measures. The estimation of FSBSOA_PyramidFDBNet-based deepfake recognition in terms of accuracy is depicted in figure 7 a). For the time stamp=30, the FSBSOA_PyramidFDBNet achieved an accuracy of 90.11%, whereas the existing techniques attained an accuracy of 82.27%, 83.20%, 84.59%, 85.65%, 86.70%, and 87.66%. The assessment concerning loss function is illustrated in figure 7 b). The loss function of 0.109 is attained by the FSBSOA_PyramidFDBNet, whereas the existing approaches attained the loss function of 0.215, 0.201, 0.182, 0.167, 0.153, and 0.140 under the time stamp of 30. Figure 7 c) displays the evaluation

with respect to MSE. For the time stamp of 30, the FSBSOA_PyramidFDBNet attained the MSE of 0.257, in which the MSE of 0.345, 0.328, 0.320, 0.288, 0.287, and 0.281 are obtained by the existent techniques. The assessment related to TNR is depicted in figure 7 d). With the time stamp of 30, the TNR of the existing methods, and the FSBSOA_PyramidFDBNet are 82.01%, 83.71%, 84.47%, 85.58%, 86.71%, 88.87%, and 91.92%. Moreover, the assessment regarding TPR is shown in figure 7 e). While considering the time stamp of 30, the TPR of 80.38%, 82.89%, 83.94%, 84.44%, 85.33%, 86.42%, and 89.30% are attained by the existing and the FSBSOA_PyramidFDBNet.





(e)

Figure 7. Comparative assessment, a) Accuracy, b) Loss function, c) MSE, d) TNR, c) TPR

4.8 Comparative discussion

The comparative assessment of FSBSOA_PyramidFDBNet-based multi-face deepfake recognition using FL with respect to the existing techniques are described in table 1. The performance of the proposed FSBSOA_PyramidFDBNet and the existing models are given with respect to the time stamps 0f 10, and 50. The FSBSOA_PyramidFDBNet achieved the ideal values using the time stamp of 50. For the time stamp 50, the proposed FSBSOA_PyramidFDBNet achieved a better accuracy of 93.91%, whereas the existing techniques attained the accuracy of 85.16%, 86.34%, 87.79%, 88.37%, 90.38%, 92.79%, and 93.77%. The minimum loss function of 0.064 is attained by the FSBSOA_PyramidFDBNet, in which the existing approaches attained the loss function of 0.174, 0.158, 0.138, 0.131, 0.106, and 0.077. The minimum MSE of FSBSOA_PyramidFDBNet is 0.179, whereas the existent techniques achieved the MSE of 0.287, 0.286, 0.235, 0.227, 0.205, and 0.200. The superior TNR of FSBSOA_PyramidFDBNet is 94.16%, whereas the TNR of existing approaches are 86.47%, 87.75%, 88.52%, 89.17%, 90.42%, and 92.06%. The highest TPR of 92.35% is attained by FSBSOA_PyramidFDBNet, and the TPR of the existing models are 84.00%, 85.11%, 87.18%, 88.33%, 89.48%, and 90.21%. Moreover, the FSBSOA_PyramidFDBNet obtained the better accuracy, loss function, MSE, TNR, and TPR of 87.58%,0.141, 0.326, 88.26%, and 87.59% for the time stamp of 10. Here, the FSBSOA_PyramidFDBNet achieved better values due to the precise training and computational process.

Table 1. Comparative Discussion

Variations	Methods /Metrics	Inception-ResNet-v2	DSLRFN	CNN	EfficientNet V2-M	Pyramid FDBNet	SBSOA_ Pyramid FDBNet	Proposed FSBSOA _Pyramid FDBNet
Time Stamp=10	*Accuracy (%)*	79.99	80.54	81.79	82.46	83.96	85.02	87.58

	Loss Function	0.250	0.241	0.222	0.212	0.190	0.176	0.141
	MSE	0.445	0.414	0.401	0.395	0.387	0.379	0.326
	TNR (%)	80.42	81.63	82.07	83.09	84.34	85.98	88.26
	TPR (%)	79.38	80.33	81.32	82.22	83.55	84.30	87.59
Time Stamp=50	*Accuracy (%)*	85.16	86.34	87.79	88.37	90.38	92.79	**93.91**
	Loss Function	0.174	0.158	0.138	0.131	0.106	0.077	**0.064**
	MSE	0.287	0.286	0.235	0.227	0.205	0.200	**0.179**
	TNR (%)	86.47	87.75	88.52	89.17	90.42	92.06	**94.16**
	TPR (%)	84.00	85.11	87.18	88.33	89.48	90.21	**92.35**

5. Conclusion

The deepfake videos created via the DL techniques have attracted extensive attention. Although the technology has been employed in various useful applications like education, and healthcare, malicious users also exploit these technological advancements in some nefarious activities. These videos damage the reputation of societal stability, personal privacy, and national security. Hence, the effectual deepfake detection technique is developed to detect such issues. Still, the deepfake recognition using a larger dataset was complex. Therefore, the FSBSOA- PyramidFDBNet-based multi-face deepfake recognition using FL is developed in this research. The nodes and servers are the crucial entities of FL. In the training model, frames extraction, face detection, facial AU detection, and feature extraction are done. Furthermore, the proposed FSBSOA-PyramidFDBNet effectively recognized the deep fake. In the global model, the updated weight from the local nodes is aggregated. The performance of FSBSOA- PyramidFDBNet-based deepfake recognition is estimated using the accuracy, loss function, MSE, TNR, and TPR parameters, which attained the optimal values of 93.91%, 0.064, 0.179, 94.16%, and 92.35%. In the future, the detection accuracy will be implemented by considering other learning techniques like ensemble Learning, and Transfer Learning (TL).

References

[1] Alnaim, N.M., Almutairi, Z.M., Alsuwat, M.S., Alalawi, H.H., Alshobaili, A. and Alenezi, F.S., "DFFMD: A Deepfake Face Mask Dataset for Infectious Disease Era With Deepfake Detection Algorithms", IEEE Access, vol. 11, pp.16711-16722, 2023.

[2] Khalifa, A.H., Zaher, N.A., Abdallah, A.S. and Fakhr, M.W., "Convolutional neural network based on diverse Gabor filters for deepfake recognition", IEEE Access, vol. 10, pp.22678-22686, 2022.

[3] Awotunde, J.B., Jimoh, R.G., Imoize, A.L., Abdulrazaq, A.T., Li, C.T. and Lee, C.C., "An Enhanced Deep Learning-Based DeepFake Video Detection and Classification System", Electronics, vol. 12, no. 1, pp.87, 2022.

[4] Coccomini, D.A., Caldelli, R., Falchi, F. and Gennaro, C., "On the Generalization of Deep Learning Models in Video Deepfake Detection", Journal of Imaging, vol. 9, no. 5, pp.89, 2023.

[5] Qadir, A., Mahum, R., El-Meligy, M.A., Ragab, A.E., AlSalman, A. and Awais, M., "An efficient deepfake video detection using robust deep learning", Heliyon, vol.10, no.5, 2024.

[6] Soudy, A.H., Sayed, O., Tag-Elser, H., Ragab, R., Mohsen, S., Mostafa, T., Abohany, A.A. and Slim, S.O., "Deepfake detection using convolutional vision transformers and convolutional neural networks", Neural Computing and Applications, pp.1-17, 2024.

[7] El-Gayar, M.M., Abouhawwash, M., Askar, S.S. and Sweidan, S., "A novel approach for detecting deep fake videos using graph neural network", Journal of Big Data, vol.11, no.1, pp.22,2024.

[8] Saravana Ram, R., Vinoth Kumar, M., Al-shami, T.M., Masud, M., Aljuaid, H. and Abouhawwash, M., "Deep Fake Detection Using Computer Vision-Based Deep Neural Network with Pairwise Learning", Intelligent Automation & Soft Computing, vol.35, no.2, 2023.

[9] Wang, T., Cheng, H., Chow, K.P. and Nie, L., "Deep convolutional pooling transformer for deepfake detection", ACM transactions on multimedia computing, communications and applications, vol.19, no.6, pp.1-20, 2023.

[10] Liu, D., Dang, Z., Peng, C., Zheng, Y., Li, S., Wang, N. and Gao, X., "FedForgery: generalized face forgery detection with residual federated learning", IEEE Transactions on Information Forensics and Security, 2023.

[11] Heidari, A., Navimipour, N.J., Dag, H., Talebi, S. and Unal, M., "A novel blockchain-based deepfake detection method using federated and deep learning models", Cognitive Computation, pp.1-19, 2024.

[12] Amerini, I., Barni, M., Battiato, S., Bestagini, P., Boato, G., Bonaventura, T.S., Bruni, V., Caldelli, R., De Natale, F., De Nicola, R. and Guarnera, L., "Deepfake Media Forensics: State of the Art and Challenges Ahead", arXiv preprint arXiv:2408.00388, 2024.

[13] Patel, Y., Tanwar, S., Gupta, R., Bhattacharya, P., Davidson, I.E., Nyameko, R., Aluvala, S. and Vimal, V., "Deepfake generation and detection: Case study and challenges", IEEE Access, 2023.

[14] The OpenForensicsd dataset taken from "https://zenodo.org/records/5528418", accessed on August 2024.

[15] Adarsh, P., Rathi, P. and Kumar, M., "YOLO v3-Tiny: Object Detection and Recognition using one stage improved model", In proceedings of 2020 6th international conference on advanced computing and communication systems (ICACCS), pp. 687-694, IEEE, March 2020.

[16] Romero, A., León, J. and Arbeláez, P., "Multi-view dynamic facial action unit detection", Image and Vision Computing, vol. 122, pp.103723, 2022.

[17] Dalal, N. and Triggs, B., "Histograms of oriented gradients for human detection", In proceedings of 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol. 1, pp. 886-893, IEEE, June 2005.

[18] Khairandish, M.O., Sharma, M., Jain, V., Chatterjee, J.M. and Jhanjhi, N.Z., "A hybrid CNN-SVM threshold segmentation approach for tumor detection and classification of MRI brain images", Irbm, vol.43, no.4, pp.290-299, 2022.

[19] Matsukawa, T. and Suzuki, E., "Person re-identification using CNN features learned from combination of attributes", IEEE, In 2016 23rd international conference on pattern recognition (ICPR), pp. 2428-2433, December,2016.

[20] Han, D., Kim, J. and Kim, J., "Deep pyramidal residual networks", In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5927-5935, 2017.

[21] Chen, Y., Zhao, X. and Jia, X., "Spectral–spatial classification of hyperspectral data based on deep belief network", IEEE journal of selected topics in applied earth observations and remote sensing, vol. 8, no. 6, pp.2381-2392, 2015.

[22] Fu, Y., Liu, D., Chen, J. and He, L., "Secretary bird optimization algorithm: a new metaheuristic for solving global optimization problems", Artificial Intelligence Review, vol. 57, no. 5, pp.1-102, 2024.

[23] Givi, H. and Hubalovska, M., "Skill Optimization Algorithm: A New Human-Based Metaheuristic Technique", Computers, Materials & Continua, vol. 74, no. 1, 2023.

[24] Bhaladhare, P.R. and Jinwala, D.C., "A clustering approach for the l-diversity model in privacy preserving data mining using fractional calculus-bacterial foraging optimization algorithm', Advances in Computer Engineering, vol.2014, no.1, pp.396529, 2014.

[25] Engle, R.F. and Manganelli, S., "CAViaR: Conditional autoregressive value at risk by regression

quantiles", Journal of business & economic statistics, vol. 22, no. 4, pp.367-381, 2004.
[26] Cifci, M.A., Hussain, S. and Canatalay, P.J., "Hybrid Deep Learning Approach for Accurate Tumor Detection in Medical Imaging Data", Diagnostics, vol.13, no.6, pp.1025, 2023.
[27] Bahadure, N.B., Ray, A.K. and Thethi, H.P., "Image analysis for MRI based brain tumor detection and feature extraction using biologically inspired BWT and SVM", International journal of biomedical imaging, vol.2017, no.1, pp.9749108, 2017.
[28] Sun, H., Li, C., Liu, B., Liu, Z., Wang, M., Zheng, H., Feng, D.D. and Wang, S., "AUNet: attention-guided dense-upsampling networks for breast mass segmentation in whole mammograms", Physics in Medicine & Biology, vol. 65, no. 5, pp. 055005, 2020.