SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Fake Image Detection on Social Networks using Hybrid Deep Learning Methodologies

Shweta Kumari[1], Ajeet Kumar Gupta[2*], Pooja[3], Vivekanand Jha[4]

[1]Department of Computer Science & Engineering, Darbhanga College of Engineering, Darbhanga- 846005, India.
E-mail: shwetaverma673@gmail.com
[2]Department of Computer Science & Engineering, Darbhanga College of Engineering, Darbhanga- 846005, India.
E-mail: ajeetkgupta12@gmail.com
[3]Department of Computer Science & Engineering, Darbhanga College of Engineering, Darbhanga- 846005, India.
E-mail: poojamunmun@gmail.com
[4]Department of Computer Science & Engineering, Darbhanga College of Engineering, Darbhanga- 846005, India.
E-mail: vivekcse42@gmail.com

Abstract
The availability of a large amount of accessible content on social media, along with sophisticated tools and processing infrastructure at a low cost, has made it extremely easy for users to produce sophisticated forgeries that have the potential to propagate hoaxes and misinformation. The ease with which anyone can use these tools to produce propaganda could lead to fear and anarchy because of the rapid growth of these technologies. Therefore, it has now become essential to have a mechanism to identify genuine and fake images. In this regard, this research aims to present a new hybrid classification model based on the Restricted Boltzmann Machine (RBM) and Recurrent Neural Network (RNN), which utilizes Deep Learning (DL) techniques to present an automated solution for detecting fake images on social media platforms. Firstly, the input images are pre-processed through image resizing and synthesizing its Error Level Analysis (ELA). Then, ResNet-152, Alex-Net, and Squeeze-Net are utilized to extract the features of the image. Moreover, a hybrid metaheuristic algorithm called Salp Inspired Reptile Search Optimization (SIRSO) is presented to optimize the hyperparameters of the suggested model. The suggested model achieves an Accuracy of 98.99%, and the model is implemented in Python programming language. From the results, it is clear that the suggested model is performing well when compared to SOTA methods.

Keywords: Social Networks, Recurrent Neural Network, Restricted Boltzmann Machine, Salp Swarm Optimization, Reptile Search Optimization.

1. Introduction

In recent years, the proliferation of fake images on social networks has become a significant concern, driven by the ease with which images can be manipulated and disseminated. Social networks, with their vast user base and rapid information-sharing capabilities, are particularly vulnerable to the spread of fake images, which can fuel misinformation, influence public opinion, and even incite violence [1][2]. As the quality of these manipulated images continues to improve, traditional methods of detection have proven inadequate. DL, with its ability to analyse large datasets and identify subtle patterns, has emerged as a powerful tool in the fight against fake image dissemination on social platforms. Leveraging Artificial Intelligence (AI) and advanced Neural Networks (NNs), DL models are being developed to automatically detect and flag fake images, helping to maintain the integrity of information shared on social networks [3][4].

Significant achievements have been made in the application of DL [5] to fake image detection on social networks. DL models, particularly CNNs and GANs, have been employed to detect inconsistencies and artefacts in images that are indicative of manipulation. These models have been trained on large datasets of real and fake images, allowing them to learn distinguishing features that are not easily visible to the human eye. One of the key successes in this area is the development of deepfake detection algorithms that can accurately identify synthetic media, even when the fake content is generated by sophisticated GANs [6]. Additionally, integrating these detection systems into social networks has enabled the automatic screening of uploaded images, significantly reducing the spread of fake content before it reaches a wide audience.

Despite these advancements, several challenges persist in the detection of fake images on social networks using DL [7]. The primary challenge lies in the dynamic and evolving nature of image manipulation techniques [8]. As

GANs and other image synthesis methods improve, they generate fake images with increasingly realistic features, making detection more difficult [9]. Furthermore, the sheer volume of images uploaded to social networks daily poses a scalability challenge for detection algorithms, which must operate efficiently in real time to be effective [10]. Another significant challenge is the diversity of content on social networks, where fake images can vary widely in terms of manipulation type, quality, and context [11]. This diversity complicates the training of DL models, as they must generalize well across a wide range of scenarios. Additionally, concerns about privacy and the ethical use of DL models for monitoring content add another layer of complexity to deploying these technologies on social networks.

Fake image detection on social networks using DL is an area of growing importance, with ongoing research and development aimed at enhancing the accuracy and scalability of detection systems [12]. While notable progress has been made, especially in the detection of deepfakes and other sophisticated image manipulations, the challenges posed by rapidly advancing synthesis techniques and the vast, diverse nature of social media content necessitate continuous innovation [13]. Future efforts will likely focus on developing more robust models capable of real-time detection, improving the generalization of models across different types of manipulations, and addressing ethical concerns related to privacy and content moderation [14]. As social networks continue to be a primary source of information for billions of users, the role of DL in safeguarding the authenticity of shared content will be crucial in maintaining trust and preventing the spread of misinformation [15]. For this reason, this paper aims to develop a novel fake image detection model with enhanced accuracy. The main contribution of the research includes

- Pre-processing is done by resizing the images and synthesizing their ELA.
- ResNet-152, Alex-Net, and Squeeze-Net are used to extract the features.
- A hybrid classification system based on RBM and RNN is proposed
- A novel hybrid SIRSO is introduced to hyper the RBM and RNN model.

2. Related Works

Some of the research done by the researchers based on Fake Image Detection on Social Networks Using various approaches are provided in this section.

Authors [16] explained the use of deepfake technology, which combines two people's faces to create a single picture or video. Concerns regarding the difficulty of identifying deepfakes are developing as the method is used more and more. In their study, eight distinct CNN models are trained to identify deep-fake images. With an accuracy of 99%, the VGGFace model outperformed the ResNet50 model, which came in second with 97%. CNNs are useful in detecting deep-fake images, as evidenced by the high accuracy rates achieved by other methods.

Authors [17] highlighted the rising problem of counterfeit money, particularly large-denomination notes. The fast development of technology has harmed society's economic and financial sectors. It draws attention to the dearth of computational resources available for identifying fake banknotes from Cuba and suggests an approach that blends DL with existing classifiers that rely on texture, colour, and form data. The evaluation shows that this is effective in identifying counterfeit money even in the absence of many real and fake samples.

Study [18] highlighted the difficulty of identifying false-picture generative adversarial networks (GANs) for contrastive loss-based DL detection. A two-streamed model is created, an identical fake network is trained using paired learning, and layer classification is used for detection. GANs are used to create false-real picture pairings. The suggested technique surpasses modern phoney picture detectors, according to their experimental findings.

Authors [19] explained the issue of phoney profiles on social media websites, which is dangerous for businesses and social networks. With an average accuracy of 83%, it shows the shortcomings of current techniques in correctly recognizing phoney profiles. The suggested remedy is an AI project built on Spark ML that seeks to more accurately identify fraudulent profiles. Their study makes use of Spark ML modules, which include a random forest classifier. It also uses ROC plots, learning curves, and confusion matrices to visually describe its findings. The results of their study suggested that the method detects phoney profiles with 93% accuracy and a 7% rate of false positives.

Authors [20] highlighted the growing use of modified multimedia material in ransomware-related crimes and other forms of cybercrime, including false information and fake news. The difficulty of distinguishing real from fraudulent images and videos is emphasized, and an ML technique that uses SVM to identify manipulated multimedia files is shown. The technique uses a DFT to extract characteristics from digital images and video frames. It is built in Python and integrated into the Autopsy digital forgery tool. The method demonstrated a high degree of accuracy in differentiating between authentic and fraudulent information when tested on a sizable dataset. The SVM method is appropriate for use in digital forensics tools since comparisons with neural network algorithms revealed competitive results.

Authors [21] covered the growing problem of counterfeit Indian rupee notes and suggested an autonomous approach that uses computational image analysis to identify counterfeit notes. To determine if a currency note is legitimate, the algorithm counts the total number of interruptions in the thread line. Furthermore, to effectively detect counterfeit notes, the system computes the amount of entropy of the notes. The manufacture of phoney banknotes has increased dramatically as a result of advancements in colour printing technology; hence, having reliable detection techniques is essential. To verify the validity of Indian currency notes, the suggested method offered benefits like simplicity and high-performance speed in extracting note attributes using MATLAB software.

Authors [22] explained the critical need for efficient techniques to identify deepfake images, which are produced by utilizing cutting-edge technologies like GANs. The technology known as Deepfake makes it possible to create phoney images and movies that are incredibly lifelike, which is a danger to cybersecurity. Their study suggested an intelligence forensic technique for deepfake identification that makes use of guided filtering and picture saliency to highlight and bring to light minute textural variations between images of real and fake faces. The technique solves the problems brought about by the growing realism of deepfake images while achieving cutting-edge detection accuracy. In addition, their study offered a range of relevant studies, experimental findings, and potential future paths for fake detection studies.

Authors [23] explained the occurrence of landslides regularly in the Karakoram area, especially along the Karakoram Highway, and the necessity of finding ways to lessen their effects. The suggestion is to evaluate landslide susceptibility and create susceptibility maps for training ANNs using metaheuristic and Bayesian methodologies. Employing eight feature selection techniques, their study also investigated the significance of several geographical factors in landslide prediction along the Karakoram Highway. The results are intended to offer guidance for constructing structures and developing efficient landslide avoidance techniques.

Authors [24] used a lightweight model for fake news detection using image captioning technique. Further, study [25] and [26] takes both the text and image modality for fake news detection and apply self and cross modal attention approach.

3. Methodology

A fake image detection process using DL techniques is proposed in this research. The process begins with pre-processing involving two key steps such as resizing and ELA. Resizing adjusts the image dimensions to a standard size suitable for further processing, while ELA examines the compression artefacts in an image to detect potential manipulations. These pre-processing steps help prepare the image for subsequent analysis by enhancing the quality and revealing any signs of tampering. Following pre-processing, the image moves to the feature extraction stage, where three different DL models including ResNet-152, AlexNet, and SqueezeNet are employed to extract features from the image. These features are then fed into the classification module, which classifies the image as either "REAL" or "FAKE." In between, the hyperparameters of the proposed model are tuned via the suggested SIRSO algorithm. The use of multiple feature extraction networks ensures a comprehensive analysis of the image, while the model ensures accurate classification of the image's authenticity. Fig 1 shows the suggested model.

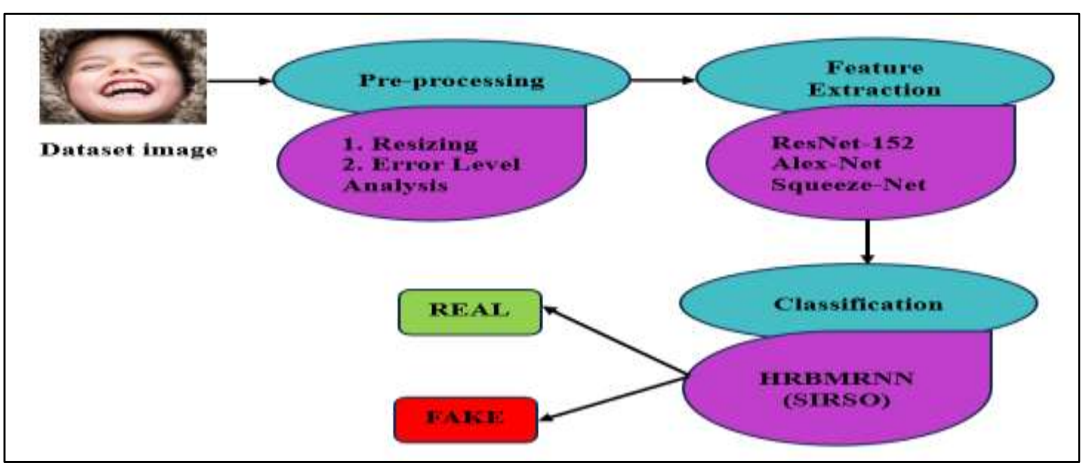


Fig 1: Overall flow of proposed model

3.1. Data pre-processing

Pre-processing in fake image detection enhances image quality and reveals subtle inconsistencies by standardizing dimensions and performing techniques like ELA, which highlights potential signs of manipulation. This step is crucial for accurate feature extraction and subsequent classification.

Resizing

Image resizing entails changing an image's actual dimensions to produce an appropriate image. The characteristics are rescaled as a result of normalization in order to guarantee that the mean μ and standard deviation σ are, respectively, 0 and 1.

$$A' = \frac{A-\mu}{\sigma} \quad (1)$$

$$\mu = \frac{1}{n}\sum_{i=1}^{n} A_i \quad (2)$$

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(A-\mu)^2} \quad (3)$$

Where A, A' are the input and standardized output.

Error Level Analysis (ELA)

ELA is a technique that uses the manipulated images' lossy compression method to detect fake images. An image's initial quality level has a distinct quality, and the ELA technique aids in recognizing the alterations made to the original image. For both the original and the recompressed images the error rate is calculated. Each color channel's error levels are calculated using,

$$ELA(a_1, a_2) = |Y(a_1, a_2) - Y_r(a_1, a_2)| \quad (4)$$

Where Y, Y_r are original and recompressed images.

The total error levels are error levels averaged across all colour channels computed by,

$$ELA(a_1, a_2) = \frac{1}{3}\sum_{j=1}^{3}|Y(a_1, a_2, j) - Y_r(a_1, a_2, j)| \quad (5)$$

Where $j = 1,2,3$ represents the R, G, and B channels.

3.2. Feature Extraction

Feature extraction in fake image detection identifies and isolates critical patterns, textures, and anomalies within an image, enabling DL models to effectively distinguish between real and manipulated content by focusing on the most relevant details. In this research, ResNet-152, Alex-Net and Squeeze-Net are utilized to extract the relevant features.

ResNet-152

For the extraction of spatial data, Residual Networks (ResNets) a kind of NN that incorporates skip connections and residual learning have been suggested. Deeper layers record more abstract and global information, while lower layers record local aspects and minute details. Fig 2 shows the ResNet-152 Architecture.

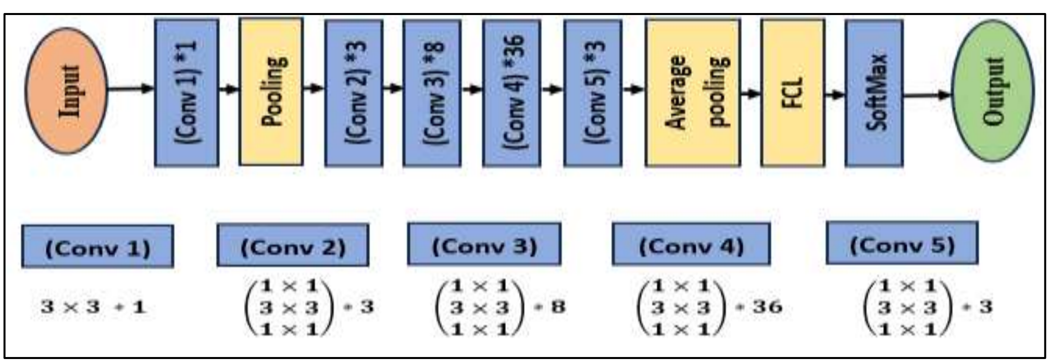


Fig 2: Overview of ResNet-152 model

The core component of ResNet-152 is the residual block as given in Eq. (6), in which X indicates the input feature map to the residual block, $f(X, \{W_i\})$ specifies the residual mapping composed of convolutional layers, batch normalization, and ReLU activation functions, W_i means to weights of the convolutional layers within the residual block, and Y addresses output feature map, which is the sum of the input Xand the residual mapping $f(X, \{W_i\})$.

$$Y_{resnet} = f(X, \{W_i\}) + X \quad (6)$$

AlexNet

Features like shapes, edges, and textures are extracted using AlexNet. Fig 3 shows the architecture of AlexNet.

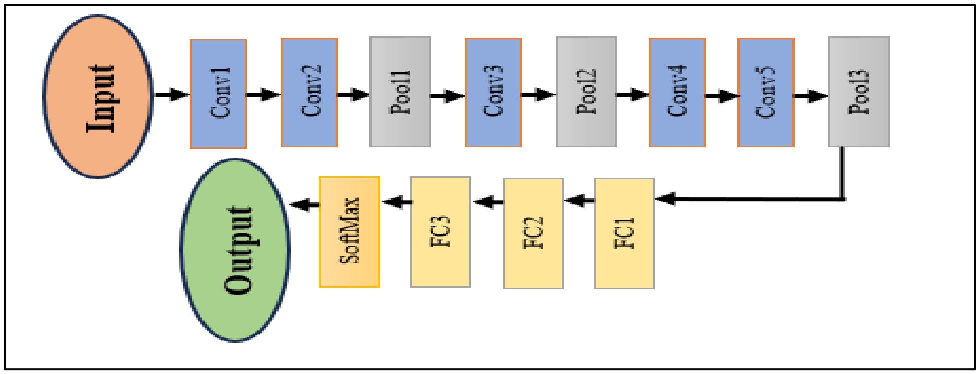


Fig 3: Overview of Alex Net model

The AlexNet model consists of five convolutional layers, three maximum pooling, and three FC layers. Image with a resolution of 227 × 227 is given as input to AlexNet. Nonlinearity is then removed using the ReLU activation function. Rich feature extraction is subsequently accomplished by down-sampling the images using pooling layers. After that, 50% of the features are eliminated when the feature vector is flattened using the first FC layer. The following Fully Connected (FC) layer sets up 4,096 convolution kernels in addition to the linear feature value of 4,096 sizes obtained by the dropout procedure. The outputs are produced by the output layer, or the final FC layer. In AlexNet, the primary feature extraction occurs in the convolutional layers. The operation of a convolutional layer is represented in eq. (7), in which X_l states input feature map to l^{th} layer, W_l means to convolutional filter weights, b_l signifies bias, $*$ represents convolution operation, $f(\cdot)$ addresses the ReLU activation function, and $Y_{alexnet}$ points out output feature map after applying the convolution, bias, and activation functions.

$$Y_{alexnet} = f(W_l * X_l + b_l) \tag{7}$$

SqueezeNet

SqueezeNet is used to extract high-level features. It has two convolution layers (CL), three max-pooling levels, eight fire layers, and one global average pooling layer. Fig 4 shows the architecture of SqueezeNet.

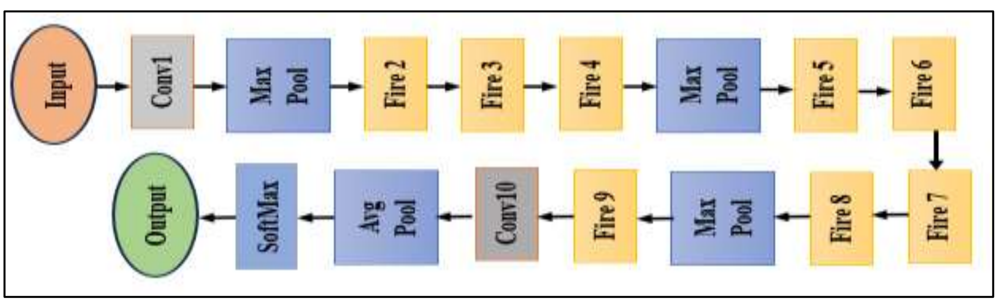


Fig 4: Overview of SqueezeNet model

Fire modules (fire2–9) make up SqueezeNet after the first convolution layer. Filters are included in every Fire module. SqueezeNet uses a two-step process called max-pooling to further specialize the input images following convolution. The convolution layer joins the restricted regions and weights in the input volumes using 3 × 3 kernels. Every convolution layer separately activates each component as the actual component. Activation is carried out by each convolution layer. In order to achieve the squeeze phase (S-p) and expanded phase (E-p) between the convolution layers, it makes use of fire layers. In S-p, 1 × 1 filters are utilized, but in E-p, 3 × 3 filters are used in addition to the preceding filter. The squeeze operation computes the weighted sum of the feature maps for each individual tensor, effectively capturing the most relevant information. In this process, global average pooling summarizes the class-specific feature maps into a single representative value, while max pooling performs down-sampling by selecting the maximum values along the spatial dimensions, thereby preserving the most prominent features. The initial convolution operation is explained in Eq. (8), in which $Conv_7$ defines convolution operation with 7×7 convolution, and X indicates the input feature map.

$$initial = Conv_7(X)\ with\ stride\ 2 \quad (8)$$

The fire module is stated in Eq. (9), and Eq. (10) gives the global average pooling (GAP).

$$fire2 = expand\big(Squeeze(initial)\big) \quad (9)$$

$$Y_{squeezenet} = GAP(last\_fire\_module\_output) \quad (10)$$

3.3. Classification

It is the final stage which detects the false images using the extracted features. A hybrid model is introduced, and the parameters are fine-tuned by a hybrid metaheuristic algorithm named, SIRSO which results in better classification accuracy. Next, utilize the RBM's hidden layer activations as features for further processing. Sequential data is handled by RNN architecture. The RNN receives the output of the hidden layer of the RBM.

Restricted Boltzmann Machine (RBM)

RBM learns input probability distribution and supervised and unsupervised learning approaches are used for reduce the dimensionality, extraction of feature, classification, and feature learning. Neurons in a bipartite graph are required for RBM organization. Unlike other NNs, RBMs are distinguished by their training approach. Gibbs Sampling and Contrastive Divergence are the main training phases. The usage of Gibbs Sampling is part of the first training phase. By computing the input vector q and the values from the hidden layer g, the prediction is obtained.

$$Z\left(q_j = \frac{1}{g}\right) = \frac{1}{1+e^{\left[-\left(b_j + w_j g_j\right)\right]}} \quad (6)$$

During the Contrastive Divergence step, the weight matrix is updated. The vectors q_0 and q_k are used to compute the activation probabilities of g_0 and g_k, where g_0 and g_k represent the hidden layer activations at the initial and k-th steps, respectively. The updated weight matrix is given as follows:

$$\Delta w = (q_0 \otimes Z.(\ g_0/q_0) - q_k \otimes Z(\ h_k/q_k) - q_k) \quad (7)$$

w_o is old and w_n is new weights and the updated values of weight matrix is updated as follow,

$$w_n = w_0 + \Delta w \quad (8)$$

Recurrent Neural Network (RNN)

RNNs are a type of neural network in which the output from a previous step is fed back as input to the current step. Unlike traditional neural networks, where inputs and outputs are treated independently, RNNs are designed to handle sequential data by retaining information from earlier steps. A key component of an RNN is its hidden state, which stores sequence-related information and helps the model make more context-aware predictions. The current state can be calculated as,

$$g_z = f(g_{z-1}, x_z) \quad (9)$$

Where g_z is the current state, g_{z-1} is the previous state x_z is the input state

Afte applying the activation function it can be written as,

$$g_z = tan\,h\,(wg_{z-1}, +ux_z) \quad (10)$$

Where $tan\,h$ is used as activation function and the outcome is given as,

$$y_z = v.\,g_z \quad (11)$$

Where v is the weight for the connection of the hidden and output layer.

A novel SIRSO optimization algorithm is introduced by which the parameters of the proposed model is tuned to improve the performance of the model. Fig. 5 shows the architecture of proposed model.

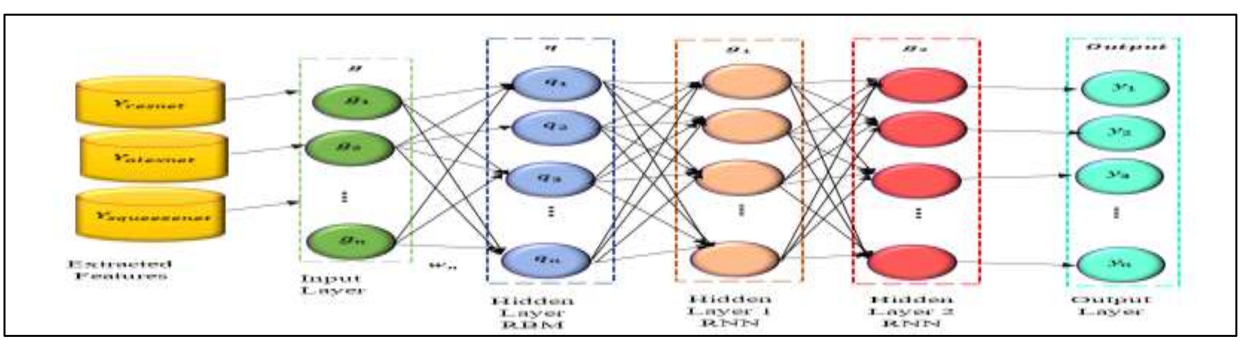


Fig 5: Architecture of proposed model

Salp Inspired Reptile Search Optimization (SIRSO)

The proposed SIRSO combines the exploratory capabilities of RSO with the exploitation strengths of SSO to balance exploration and exploitation in solving optimization problems. The initial population of crocodiles in the RSA is created at random and is stated as,

$$A = \begin{bmatrix} a_1 \\ a_1 \\ \vdots \\ a_b \end{bmatrix}_{b\times c} = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,c} \\ a_{2,1} & a_{2,2} & \dots & a_{1,c} \\ \vdots & \vdots & \vdots & \vdots \\ a_{b,1} & a_{b,2} & \dots & a_{b,c} \end{bmatrix}_{b\times c} \quad (12)$$

Where A represents a randomly generated initial solution, b, c are candidate solutions and dimensions, $a_{b,c}$ is the position of b^{th} solution in the c^{th} dimension.

$$p_{x,y} = rd * \left(U_y - L_y\right) + L_y \ ; x\epsilon\{1,..,b\}\ y\epsilon\{1,..,c\} \quad (13)$$

Where rd is a random number [0,1], U_y, L_y are max and min values.

The two aspects of RSA's operation are exploration and exploitation. In order to provide a more thorough search of the solution space, crocodiles start to circle about the area. This can be expressed mathematically as,

$$p_{x,y}(z+1) = \begin{cases} B_y(z) \times \delta_{x,y}(z) \times 0.1 - R_{x,y}(z) \times rd \ ; z \le \frac{Z}{4} \\ B_y(z) \times p_{d1,y} \times E(z) \times rd;\ \frac{Z}{4} < z \le \frac{Z}{2} \end{cases} \quad (14)$$

Where $B_y(z)$ is the best position of optimal solution in y^{th} dimension at time z, Z is maximum iterations, rd is a random number [0,1], $\delta_{x,y}(z)$ the hunting operator for the y^{th} dimension of the x^{th} candidate solution, $R_{x,y}(z)$ is a scaling function used to reduce the search area, $d1$ is a random number [1, b] and $E(z)$ is an evolutionary factor, with a randomly decreasing value [2,-2].

$$\delta_{x,y}(z) = B_y(z) \times f_{x,y} \quad (15)$$

$$R_{x,y}(z) = \frac{B_y(z) - p_{d2,y}}{B_y(z) + \theta} \quad (16)$$

$$E(z) = 2 \times rd \times \left(1 - \frac{z}{Z}\right) \quad (17)$$

Where θ prevents the cases when the denominator is zero, rd is an integer [-1,1], $f_{x,y}$ the percentage difference between the best and present solution

$$f_{x,y} = 0.1 + \frac{p_{x,y} - \frac{1}{c}\sum_{y=1}^{c} p_{x,y}}{B_y(z) \times (U_y - L_y) + \theta} \quad (18)$$

As a result, after multiple tries, exploitation search may find nearly ideal solutions. The following is the follower population during the exploratory phase

$$p_{x,y}(z+1) = \begin{cases} B_y(z) \times f_{x,y} \times rd \ ; \quad \frac{Z}{2} < z \le \frac{3Z}{4} \\ B_y(z) - \delta_{x,y}(z) \times \theta - R_{x,y}(z) \times rd;\ \frac{3Z}{4} < z \le Z \end{cases} \quad (19)$$

The Salp Swarm Algorithm (SSA) is inspired by the swarming behaviour of salps, marine organisms that move in the form of a chain-like structure. This chain formation plays a key role in guiding the search process, where individuals follow a leader, enabling efficient exploration and exploitation of the solution space. Every salp chain has a leader salp and the others follow. Salp positions, which include food source M, are allocated to the particular search space. In order to inform the leadership about the food source,

$$\mathrm{Y}_{\mathrm{a}}^{\mathrm{i}} = \begin{cases} \mathrm{M_a} + \mathrm{k_1}[(\mathrm{U_a} - \mathrm{L_a})\mathrm{k_2} + \mathrm{L_a}];\ \mathrm{k_3} \ge 0 \\ \mathrm{M_a} - \mathrm{k_1}[(\mathrm{U_a} - \mathrm{L_a})\mathrm{k_2} + \mathrm{L_a}];\ \mathrm{k_3} \le 0 \end{cases} \quad (20)$$

$\mathrm{Y}_{\mathrm{a}}^{\mathrm{i}}$ is the leader's position, $\mathrm{M_a}$ indicates the source of food in a^{th} dimension, $\mathrm{U_a}, \mathrm{L_a}$ are upper and lower bounds. $\mathrm{k_1}, \mathrm{k_2}, \mathrm{k_3}$ are randomly chosen. $\mathrm{k_1}$ balances between exploration and exploitation, $\mathrm{k_2}, \mathrm{k_3}$ are random values in [0,1].

$$\mathrm{k_1} = 2\mathrm{e} - \left(\frac{4\mathrm{t}}{\mathrm{T}}\right)^2 \quad (21)$$

Where t, T are present and maximum iterations.

The exploitation phase of RSO gains improved exploration-exploitation balance, adaptability to dynamic landscapes, efficient exploration of promising regions, preservation of solution diversity, overall robustness and generalization, and improved optimization performance by incorporating the controlling factor from SSO. The modified equation is given as,

$$\mathrm{p_{x,y}}(\mathrm{z}+1) = \begin{cases} \mathrm{B_y}(\mathrm{z}) \times \mathrm{f_{x,y}} \times \mathrm{rd}\ ; \quad \frac{\mathrm{Z}}{2} < \mathrm{z} \le \frac{3\mathrm{Z}}{4} \\ \mathrm{k_1} \times \mathrm{B_y}(\mathrm{z}) - \delta_{\mathrm{x,y}}(\mathrm{z}) \times \theta - \mathrm{R_{x,y}}(\mathrm{z}) \times \mathrm{rd};\ \frac{3\mathrm{Z}}{4} < \mathrm{z} \le \mathrm{Z} \end{cases} \quad (22)$$

Thus, the false images from the original images are detected by using the above-explained process. Algorithm 1 gives the pseudocode of the proposed SIRSO.

Algorithm 1: Pseudocode of Proposed SIRSO

```
Begin
    Initialize population p, dimension d, maximum iteration M_t, and current iteration t
    Initial fitness estimation
    For i = 1: M_t
        Update reptiles as per Eq. (14)
        Update Leader Salp Position using SSO in Eq. (20)
        Update Follower Salp Positions using SSO in Eq. (19)
        Combine and Evaluate Hybrid Population as per Eq. (22)
        Update the best solution if a better one is found
    End for
    Return best hyperparameters
End
```

4. Result and Discussion

Evaluation metrics are employed to assess the performance of the models on the given dataset. The obtained results are systematically compared between the proposed approach and existing deep learning models, including CNN, RNN, AlexNet, and ResNet. All models are implemented and evaluated using the Python platform, ensuring a consistent and reliable comparison framework.

4.1. Dataset Description

The Real and Fake Face Detection dataset is a collection of information utilized for the research which is accessible via https://www.kaggle.com/datasets/ciplab/real-and-fake-face-detection. The dataset contains real and fake images of human faces. The dataset comprises 2041 images total, of which 1081 are real and 960 are counterfeit [27].

4.2. Evaluation parameters

The performance metrics presented below are used to evaluate both the proposed and existing models. Table 1 provides a comparative analysis of the proposed model with established deep learning models such as CNN, RNN, AlexNet, and ResNet using a 70:30 train–test split. The results clearly show that the proposed model consistently achieves higher values across all evaluation metrics, demonstrating its superior effectiveness in classification tasks. Similarly, Table 2 presents the comparison using an 80:20 train–test split. Once again, the proposed model outperforms CNN, RNN, AlexNet, and ResNet across all considered metrics. In addition to achieving the highest accuracy and related measures, it records the lowest False Positive Rate (FPR) and a very low False Negative Rate (FNR). This indicates that the model is highly efficient in correctly identifying true positives while minimizing misclassifications, thereby confirming its robustness and overall superior performance.

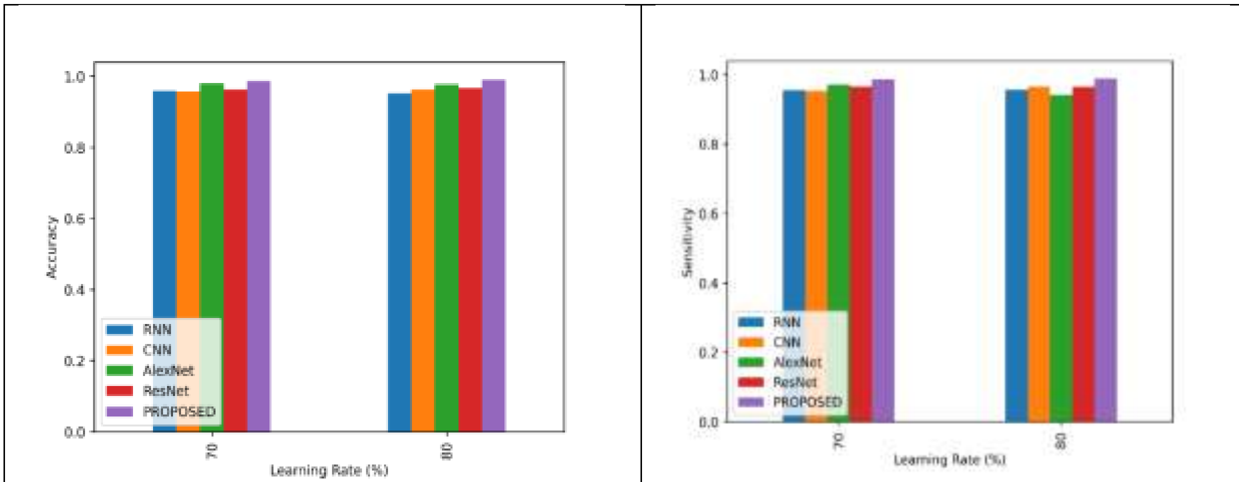
Table 1: Numerical results of models (Training-70% and Testing- 30%)

Model	Accuracy	Precision	F-Score	Specificity	Sensitivity	MCC	FPR	FNR
CNN	95.834	95.786	95.248	95.259	95.326	95.551	4.563	5.891
RNN	95.946	95.712	95.729	95.721	95.528	95.698	4.172	4.983
AlexNet	97.982	97.093	97.195	97.402	97.072	97.491	5.781	6.562
ResNet	96.429	96.364	96.377	96.333	96.625	96.659	4.348	5.172
Proposed	98.682	98.529	98.426	98.629	98.612	98.599	3.809	0.748

Table 2: Numerical results of models (Training-80% and Testing- 20%)

Model	Accuracy	Precision	F-Score	Specificity	Sensitivity	MCC	FPR	FNR
CNN	96.429	96.364	96.377	96.333	96.625	96.659	4.348	5.172

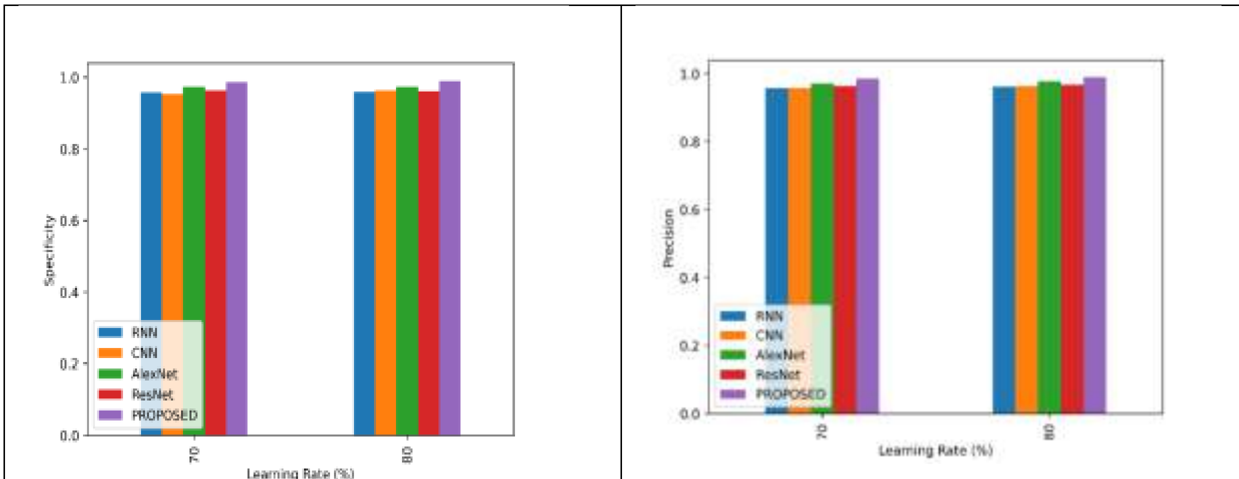
RNN	95.312	96.078	95.973	95.822	95.652	95.879	6.78	5.983
AlexNet	97.795	97.755	97.146	97.318	94.121	97.424	4.983	5.89
ResNet	96.899	96.745	96.599	96.124	96.525	96.145	4.781	4.12
Proposed	98.991	98.912	98.893	98.928	98.929	98.862	2.786	0.701



(a) Accuracy **(b)** Sensitivity

Fig 6: Performance Assessment of model with respect to Accuracy and Sensitivity

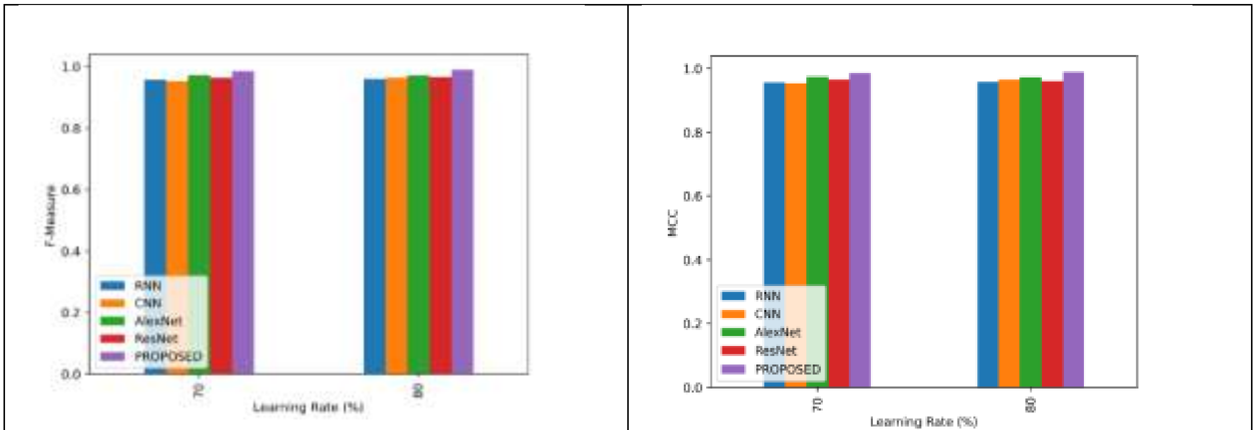
Figure 6(a) illustrates the comparative accuracy of the proposed model against existing approaches. It can be observed that the proposed model achieves a superior accuracy of 98.99%, outperforming all other models presented in the graph. This highlights its effectiveness and robustness in handling the given task. The detailed numerical outcomes supporting these observations are provided in Tables 1 and 2. Furthermore, Figure 6(b) shows the sensitivity analysis of both the proposed and baseline models. The proposed model attains a high sensitivity of 98.92%, indicating its strong ability to correctly identify positive instances. Overall, both the graphical and tabular results clearly demonstrate that the proposed model consistently performs better than existing methods, particularly in terms of accuracy and sensitivity.



(a) Specificity (b) Precision

Fig 7: Comparison of Specificity and Precision of the models

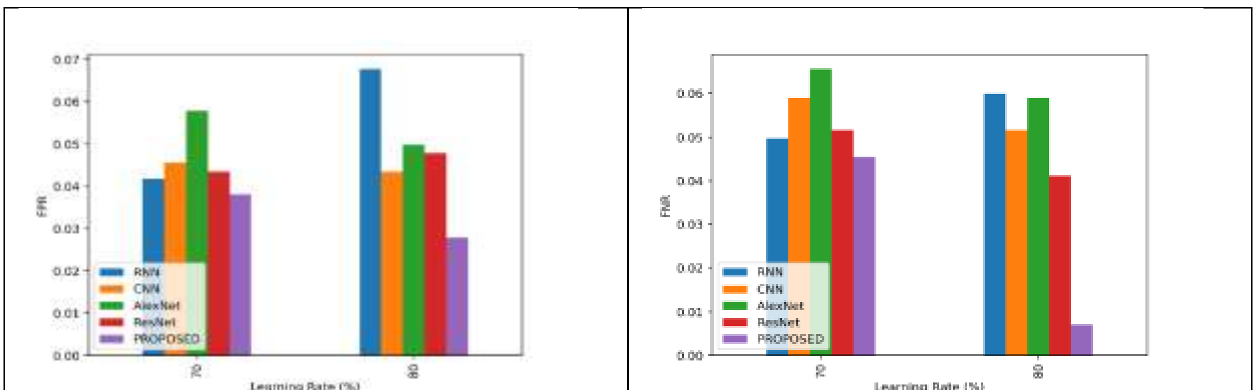
Figure 7(a) presents the specificity comparison between the existing and proposed models. From the graph, it is clearly observed that the proposed model achieves a higher specificity of 98.92%, indicating its improved capability in correctly identifying negative instances compared to the existing models. Similarly, Figure 7(b) illustrates the precision analysis of the models. The proposed model attains a precision of 98.91%, demonstrating its effectiveness in minimizing false positive predictions. Overall, the graphical results confirm that the proposed model outperforms the existing approaches in terms of both specificity and precision, further validating its reliability and performance.



(a) F1 Score	(b) MCC

Fig 8: Evaluation of model in terms of F1-measure and MCC

Figure 8(a) shows the evaluation of the models based on the F-measure. From the graph, it is evident that the proposed model outperforms the existing methods, achieving a high F-measure of 98.89%. This indicates a strong balance between precision and recall, reflecting the model's overall effectiveness. Figure 8(b) presents the MCC analysis of both the proposed and existing models. The proposed model attains an MCC of 98.86%, which demonstrates its superior predictive performance and reliability. Overall, these results confirm that the proposed model consistently performs better across multiple evaluation metrics.



(a) FPR	(b) FNR

Fig 9: FPR and FNR analysis of the recommended and current models

Figures 9(a) and 9(b) illustrate the comparison of the proposed and existing models in terms of False Positive Rate (FPR) and False Negative Rate (FNR). The results clearly show that the proposed model achieves a lower FPR of 2.7% and a significantly reduced FNR of 0.7%. These lower error rates indicate that the model makes fewer incorrect predictions, thereby improving its overall reliability and performance compared to existing methods.

5. Conclusion

In conclusion, eliminating forgeries and false information is made more difficult by the abundance of fake content on social media. This study presented a novel method for automatically detecting fraudulent images by proposed model, which makes use of DL techniques. The images are resized and error-level analysis is carried out. Then the features are extracted using ResNet-152, Alex-Net, and Squeeze-Net models. A hybrid classification system based on Recurrent Neural Network and Restricted Boltzmann Machine is proposed and the parameters of the model are tuned by a novel Salp Inspired Reptile Search Optimization Algorithm. The suggested model attained an Accuracy of 98.99%, Sensitivity of 98.92%, Specificity of 98.92%, Precision of 98.91%, F-measure of 98.89%, MCC of 98.86%, FPR of 2.7% and FNR of 0.7%. These findings show how the suggested model outperforms current techniques and provides an accurate mechanism for detecting real and fake images on social media networks.

Declarations

Author Contributions: All authors contributed equally.

Funding: Not applicable.

Data Availability: The datasets used in this study are available from the corresponding author upon reasonable request.

Ethical Approval: This research does not involve any studies with human participants or animals conducted by the authors.

Consent to Participate: Not applicable, as no human participants are involved in this study.

Consent to Publish: All authors have provided their consent for the publication of this manuscript.

Competing Interests: The authors declare that they have no competing interests.

References

[1] Gupta, A. K., & Kumar, P. (2018). A comparative analysis of machine learning techniques for fake news detection. International Journal of Research and Analytical Reviews (IJRAR), E-ISSN, 2348-1269.

[2] Gupta, A. K., Kumar, P., Kumar, S. (2019). Enhancing fake news detection through user credibility analysis. International Journal of Research and Analytical Reviews (IJRAR).

[3] Gupta, A. K. (2023, August). Fake News Detection based on User Credibility: A Comprehensive Analysis. In Proceedings of 2nd International Conference on Research in Multidisciplinary Studies-2023.

[4] Gupta A.K.; (February 2025); Artificial Intelligence (AI) in Healthcare: A Comprehensive Analysis; International Journal of Universal Science and Engineering; Vol 11, Issue 1, Special Issue, 17-23

[5] Yadav, A. K., Kumar, S., Kumar, D., Kumar, L., Kumar, K., Maurya, S. K., ... & Yadav, D. (2023). Fake news detection using hybrid deep learning method. SN Computer Science, 4(6), 845.

[6] Rafique, R., Gantassi, R., Amin, R., Frnda, J., Mustapha, A. and Alshehri, A.H., 2023. Deep fake detection and classification using error-level analysis and deep learning. Scientific Reports, 13(1), p.7422.

[7] Yavuzkilic, S., Sengur, A., Akhtar, Z. and Siddique, K., 2021. Spotting deepfakes and face manipulations by fusing features from multi-stream cnns models. Symmetry, 13(8), p.1352.

[8] Su, J., Bai, Y., Wang, X., Lu, D., Zhao, B., Yang, H., Mas, E. and Koshimura, S., 2020. Technical solution discussion for key challenges of operational convolutional neural network-based building-damage assessment from satellite imagery: Perspective from benchmark xBD dataset. Remote Sensing, 12(22), p.3808.

[9] AlShariah, N.M., Khader, A. and Saudagar, J., 2019. Detecting fake images on social media using machine learning. International Journal of Advanced Computer Science and Applications, 10(12), pp.170-176.

[10] Karnouskos, S., 2020. Artificial intelligence in digital media: The era of deepfakes. IEEE Transactions on Technology and Society, 1(3), pp.138-147.

[11] Dame Adjin-Tettey, T., 2022. Combating fake news, disinformation, and misinformation: Experimental evidence for media literacy education. Cogent arts & humanities, 9(1), p.2037229.

[12] Bowles, J., Croke, K., Larreguy, H., Marshall, J. and Liu, S., 2023. Sustaining Exposure to Fact-checks: Misinformation Discernment, Media Consumption, and its Political Implications. Media Consumption, and its Political Implications (September 25, 2023).

[13] Marra, F., Gragnaniello, D., Cozzolino, D. and Verdoliva, L., 2018, April. Detection of gan-generated fake images over social networks. In 2018 IEEE conference on multimedia information processing and retrieval (MIPR) (pp. 384-389). IEEE.

[14] Kyza, E.A., Varda, C., Panos, D., Karageorgiou, M., Komendantova-Amann, N., Coppolino Perfumi, S., Shah, S.I.H. and Hosseini, A.S., 2020. Combating misinformation online: re-imagining social media for policy-making. Internet Policy Review, 9(4), pp.1-24.

[15] Prasetiyo, W.H., Sumardjoko, B., Muhibbin, A., Naidu, N.B.M. and Achmad, M.İ., 2023. Promoting digital citizenship among student-teachers: The role of project-based learning in improving appropriate online behaviors. Participatory Educational Research, 10(1), pp.389-407.

[16] Shad, H.S., Rizvee, M.M., Roza, N.T., Hoq, S.M., Monirujjaman Khan, M., Singh, A., Zaguia, A. and Bourouis, S., 2021. Comparative analysis of deepfake image detection method using convolutional neural network. Computational Intelligence and Neuroscience, 2021.

[17] Sánchez-Rivero, R., Febles-Espinosa, Y., Silva-Mata, F.J. and Morales-González, A., 2023, October. Authenticity Assessment of Cuban Banknotes by Combining Deep Learning and Image Processing Techniques. In International Workshop on Artificial Intelligence and Pattern Recognition (pp. 433-444). Cham: Springer Nature Switzerland.

[18] Hsu, C.C., Zhuang, Y.X. and Lee, C.Y., 2020. Deep fake image detection based on pairwise learning. Applied Sciences, 10(1), p.370.

[19] Awan, M.J., Khan, M.A., Ansari, Z.K., Yasin, A. and Shehzad, H.M.F., 2022. Fake profile recognition using big data analytics in social media platforms. International Journal of Computer Applications in Technology, 68(3), pp.215-222.

[20] Ferreira, S., Antunes, M. and Correia, M.E., 2021. Exposing manipulated photos and videos in digital forensics analysis. Journal of Imaging, 7(7), p.102.

[21] Venu, D.N., 2023. An Automatic recognition system of fake Indian currency notes detection using Image processing analysis. European Chemical Bulletin, 12(9), pp.280-307.

[22] Yang, J., Xiao, S., Li, A., Lan, G. and Wang, H., 2021. Detecting fake images by identifying potential texture difference. Future Generation Computer Systems, 125, pp.127-135.

[23] Abbas, F., Zhang, F., Abbas, F., Ismail, M., Iqbal, J., Hussain, D., Khan, G., Alrefaei, A.F. and Albeshr, M.F., 2023. Landslide susceptibility mapping: analysis of different feature selection techniques with artificial neural network tuned by bayesian and metaheuristic algorithms. Remote Sensing, 15(17), p.4330.

[24] Gupta, A.K. and Singh, M.P., 2025. Multimodal Fake News Detection through Semantically Enriched Image Captions. SN Computer Science, 6(8), p.1020.

[25] Gupta, A.K. and Singh, M.P., 2026. Self and cross-modal attention based features fusion for fake news detection. Multimedia Tools and Applications, 85(1), p.10.

[26] Gupta, A.K. and Singh, M.P., 2026. Multimodal Fake News Detection using Semantic-Visual Alignment and Attention Fusion. SN Computer Science,7(1), p.305.

[27] Dataset https://www.kaggle.com/datasets/uditsharma72/real-vs-fake-faces?select=fake