DeepCyberDetect: Hybrid AI for Counterfeit Currency Detection with GAN-CNN-RNN using African Buffalo Optimization

Dr. Franciskus Antonius[1], Jarubula Ramu[2], Dr. P. Sasikala[3], Dr. J. C. Sekhar[4], Dr. S. Suma Christal Mary[5]
Lecturer at School of Business and Information Technology STMIK LIKMI, Bandung, Indonesia[1]
Associate Professor and Head, Department of CSE, NRI Institute of Technology, Guntur[2]
Assistant professor, Computer Science, Government Science College (Nrupathunga University), Bangalore, Karnataka[3]
Professor IN CSE, NRI Institute of Technology, Guntur[4]
Professor, Department of Information Technology, Panimalar Engineering College, Poonamalle, Chennai[5]

***Abstract*—Modern technology has made a big contribution to the distribution of counterfeit money and the valuation of it. This paper recommends a deep learning-based methodology for currency recognition in order to extract attributes and identify money values; machine learning's binary classification task of fake currency detection. One can train a model that can distinguish between real and fake banknotes if one has enough information about actual and fake notes. The vast majority of older systems relied on hardware and techniques for image processing. Using such strategies renders identifying fake currency more challenging and inefficient. The proposed system has suggested deploying a deep convolution neural network to figure out fake currency in order to solve the aforementioned issue. By analyzing the images of the currency, our technique finds counterfeit notes. The transfer-learned convolutional neural network is trained using data sets that represent 2000 different currency notes in order to learn the unique characteristics map of the currencies. After becoming familiar with the feature map, the network is capable of real-time phoney cash detection. It is surprising how well deep learning models perform in photo classification tasks. The Deep CNN model that has been created in the proposed approach helps in the detection of the fake note without really manually extracting the properties of photographs. The model trains from the data set produced during training, letting us to identify fake currency. In multiple instances, techniques for deep learning have been shown to be more effective. Thus, deep learning is used to boost currency recognition accuracy. Among the techniques used are the African Buffalo Optimization Approach (ABO), recurrent neural networks (RNN), convolutional neural networks, generative adversarial networks (GAN) for identifying bogus notes, and classical neural networks.**

***Keywords*—Fake currency; convolutional neural network; generative adversarial networks; recurrent neural network; African Buffalo Optimization**

I. Introduction

With the rapid advancement of modern technology, the proliferation of counterfeit money poses a significant challenge for financial systems worldwide. Detecting and distinguishing fake currency from genuine notes is crucial for maintaining economic stability and trust in monetary transactions. This paper presents a deep learning-based methodology for currency recognition, aiming to extract attributes and accurately identify the value of money.

For the monetary exchange of some types of goods, several nations employ various types of currencies. One problem with cash that many countries face is the existence of counterfeit currency in the system. One of the countries with numerous problems and large losses as a result of the fake money is India. The entire economy of the nation experiences losses, as does the value of its currency. Technology advancements have made it possible for currencies to be duplicated to the point that it is difficult to distinguish between them. Modern editing tools and cutting-edge printers are used to create counterfeit money. False currency can easily be bundled with legitimate currency, as is the case in the majority of the world. In order to increase public awareness of the security issues with bank notes, the State Bank of Pakistan employs a variety of different marketing strategies, both directly to its consumers and in partnership with other institutions. Through a television advertisement and an application for smartphones that checks for counterfeit currency, the bank hopes to educate the public. But most individuals can't tell the distinction between a phoney and real currency note, especially those who are illiterate. Some people may be able to spot forged currency because they are not understanding the security elements of currency in public. Additionally, it might be challenging to identify these characteristics when the money notes are tattered, dirty, and ripped [1].

Money that has been produced without the State's or government's legal consent is known as counterfeit money. Its main purpose is to imitate real money and trick the person it is intended for. The production or use of counterfeit money is seen as a kind of fraud or forgery and is therefore illegal. Early on after gaining its independence, Bangladesh took on the monetary policy legacy left by Pakistan's Central Bank. The initial state of affairs was unstable, but the government soon established Bangladesh Bank in 1972 to stabilize the situation. Following that, Bangladesh has experienced a sharp rise in the availability of counterfeit currency. The suggested methods produce topic-based shards with minimal size variance and high densities of pertinent documents. They are scalable, efficient, and self-sufficient. Information gathering for the initial stage of the model is aided by the suggested approach. The CNN, a multilayer network of neurons, is the most widely used deep learning technique in the next stage. Since it has increased the accuracy of many machine learning tasks and is effective at handling picture classification and identification issues [2], [3]. The deep CNN model used in the proposed approach is trained on a diverse dataset comprising 2000 different currency notes, enabling it to learn unique characteristics and feature maps of various currencies. Once familiar with the feature map, the model is capable of real-time detection of counterfeit currency, providing a highly efficient and automated solution [4].

The development of automated methods and systems for recognizing currencies has accelerated daily. Effective currency identification systems and cost-effective solutions are crucial in a variety of settings, including the banking system, train ticket windows, retail malls, and currency exchange services, among others. For financial transactions, banknotes are utilized. The prevalence of bogus currency on the international market has dramatically increased. In many industries, fake money is a problem. The country's financial market is negatively impacted by the development and design of these bogus notes, which come in all denominations [5]. As a result of advances in current technology and gadget development, counterfeit banknote production has increased due to the introduction of scanners and copy machines. It is very challenging for the human eye to tell phoney notes from genuine ones since they are expertly designed to look so similar. The security of banknotes must be considered, and security components must be incorporated, to reduce counterfeit money. As a result, a system that determines whether a note is real or false needs to be put in place in banks and at ATMs for withdrawal and deposit. However, people are still able to create these fictitious currencies [6].

Deep learning models have demonstrated exceptional performance in photo classification tasks, and the deep CNN model developed in this study eliminates the need for manual feature extraction. By training on the provided dataset, the model gains the ability to accurately identify fake currency notes. Around 180 distinct currencies are used in transaction worldwide. Each note has distinct safeguards and an array of sizes. Because each note has distinct security features and a different size, it is simple to tell whether currency is from which country and how much it costs. Currency notes fraud has now become a noticeable problem in all nations, and the quantity of fake notes is increasing daily. The world faces a difficult issue in identifying fake currency due to the similar look of genuine and counterfeit currency. As a result, data augmentation methods including image improvement, color analysis, and others are used in currency detection systems. Deep learning methods, which often involve multilayer neural networks, have proven successful in a variety of fields. When huge data is accessible, they have done especially well. Accurate money recognition has a lot of potential to improve with deep learning. The transfer learned CNN is trained using two thousand currency note data sets and is given the feature map of various currencies. The network has been trained to identify counterfeit banknotes in real time after learning the feature map [7].

The ABO technique, used in feature selection, aims to create an algorithm that is easy to use, reliable, efficient, and effective, yet has amazing skill in the exploitation and exploration of the search space. ABO makes sure that each buffalo's location is regularly updated in comparison to both its best prior location and the present location of the best buffalo in the herd in an effort to combat the problem of early convergence or stagnation. The entire herd may be reset, for instance, if the location of the dominant (best) buffalo is not altered after several occurrences. Finding the best buffalo guarantees that the search area has been adequately explored, and the ABO can achieve adequate exploitation by drawing on both the experience of other buffalos and the best buffalo. There are some limitations to the CNN-developed money-detecting system. The ABO simulates the three distinctive behaviors that enable the African buffalos to find pastures. Their tremendous memory capacity comes first. This makes it possible for the buffalos to trace their movements as they travel thousands of kilometers over the African terrain. Additionally, buffalos are quite helpful. The second characteristic of the buffalo is their cooperation and communicative abilities, which they exhibit in both good and bad circumstances. They are practically the only animal breed that will put their own life in peril to save the life of a member of their own species. The security features can only be recognized using the front image of the currency. The dataset is the most important element, and it must be accurate and of the greatest standard. The image quality has a significant impact on how well the classification works out. Additionally, each image in the created dataset needs to have its background clearly visible to avoid confusing the model with objects and noise. collecting data is therefore essential [8]. Overall, this paper aims to present a comprehensive and effective deep learning-based system for counterfeit currency detection and classification. By harnessing the power of deep learning and incorporating innovative optimization techniques, the proposed approach exhibits promising potential for improving currency recognition accuracy and combatting the issue of counterfeit money distribution.

The key contribution of proposed approach is,

- Generative Adversarial Networks (GAN), which can create new samples that are comparable to the original target data after being trained to learn its distribution, are used to create the data.
- Convolutional Neural Networks (CNN) are used to extract the features since they automatically produce the features from time series data and frequency representation images.
- African Buffalo Optimization (ABO) is used to choose the characteristics, which reduces computation complexity and boosts efficiency.
- Recurrent Neural Network (RNN) is used in the phase of classification which finally decides the whether the currency is fake or not.

II. Related Works

The identification of counterfeit money using deep learning was suggested by Shilpa et al, [9]. Two phases make up the system that is being suggested here. Classifying currency notes according to denomination is the first step. Checking the note's authenticity is the second step. Here, information has been taken from Kaggle, a well-known source of datasets. With the aim of spotting counterfeit money on devices like smart phones, tablets, and PCs, the recommended system is designed using a deep learning methodology, and a Neural CNN model is developed. On a set of data that was independently produced, the developed model was trained and tested. CNN receives images that have been collected by a smart camera. There are two primary phases to this web-based strategy. The first stage is to group the notes into different categories based on their worth, and the second is to determine whether or not they are real or fake. However, this strategy did not result in more favorable and successful outcomes. Saxena et al. suggested utilizing image processing to find counterfeit money. The image processing technique, which comprises changing a picture's fundamental properties to improve its visual information for human interpretation, is applied using MATLAB. The feature of the image processing programme increases the possibilities of the MATLAB numeric computing environment. The algorithm used in this method is image scaling, which resizes the input image to 100 dpi in order to provide a better KNN for classification. By doing this, the false positives are eliminated without obviously hurting how well our software works. The image is then converted from the RGB to the grayscale domain via grayscale computation. Here, a cell phone with a scanner or camera is used to use a MATLAB technique in order to identify fraudulent currency. However, it is crucial for researchers to change Matlab's features in order to achieve its high performance measures [10].

Machine learning algorithms were proposed by Shahani et al. to assess banknote authentication. Back propagation neural networks (BPN) and support vector machines (SVM) are examples of supervised learning techniques that are accustomed to distinguish actual banknotes from counterfeit ones. The dataset used to train the models was provided via the UCI machine learning repository. A ratio of 80:20 is utilized to divide the used dataset into its two parts. The smaller group is used to test whether the models can correctly predict whether a letter is authentic, while the bigger fraction is used to train the models. As a result, BPN can distinguish between genuine and counterfeit notes with accuracy. SVM also has trouble identifying real notes from false ones. By classifying the notes as Genuine, Low-Quality forgery, High-Quality forgery, and Inappropriate ROI, this study can be expanded [11]. Deep learning was suggested by Pachon et al. to identify fraudulent banknotes. This method compares the two design approaches using illustrations of various building types. The transfer learning technique identifies the optimal freezing sites in the sequential, residual, and inception CNN architectures. A sequential CNN-based original model similar to AlexNet is also suggested. Modern solutions have been presented for various CNN architectural kinds. On a dataset of Colombian banknotes, the TL and the custom models were trained, and then they were assessed. The results showed that ResNet18 had the highest accuracy. The key limitations of a customized model are its protracted training procedure and need for an appropriately diverse training sample [12].

Convolutional traces were recommended by L. Guarnera et al. [13] for identifying counterfeit currency. The suggested technique has a set of unique local characteristics designed specially to clarify the underlying neuron formation procedure using an Expectation Maximization (EM) algorithm. For spontaneous validation to confirm non-fakes, experimental tests using naive classifiers on five independent architectures have been used. For noisy photos, the computation is labor-intensive. Deep learning was suggested by Cruz et al. to identify counterfeit Indian banknotes. To aid in the training and testing of the deep network DCNN model, a fake data set was created utilizing the various image sets. VGG-16 In the context of the Single Shot Multi Box Detector (SSD) model, CNN is utilized as a feature extractor. For the purpose of object detection, SSD is a frame model. SSD using mappings of features from various layers, create bounding boxes. It has been trained with Tensorflow. A convolutional neural network will be trained using the suggested algorithm on the provided false and original currency data set, and it will then be able to determine if a particular currency image is fake or original. However, since the base VGG-16 network takes up 80% of the time, the computation takes a lengthy time [14].

Laavanya et al, proposed the detection of fake currency using deep learning. This method focuses on identifying bogus currency that is pervasive in the Indian market. The most common deep neural network technique, referred to as transfer learning utilizing Alex net, is used to detect counterfeit money. Alex net includes fully-connected layers, dropout, ReLU activations, max pooling, and convolutions. The data store of images of money notes is created with the intention of training the network. For each note, 100 images are created with the help of augmentation. Techniques for augmentation, such as rotating and resizing, are used to increase the amount of data bases. A camera is used to evaluate the procedure in real time. A "Real Note" or "Fake Note" result is generated by comparing the input currency note's properties to those that the network has already learned. This comparison occurs as soon as the picture has been acquired. Since the monetary distinctive attributes are gradually learned, the detection accuracy is at its highest. The acquired image may also contain noise, which must be taken into account as part of the pre-processing step in the currency detection process. By taking into account the surface patterns of the coin as characteristics, the detection accuracy of phoney currency can also be increased [6].

Deep learning was suggested by Gebremeskel et al. to identify Ethiopian banknotes. The image capture, image size regularization, grayscale conversion, and histogram equalization steps of the detection model sequence combine to significantly reduce the parameter number counts in the convolutional region of the DL structure. The Fully connected (FC) layer, which is basically in charge of evaluating the incoming image, executes the detection process. The deep CNN suggestion process organizes the input images into numerous groups based on the characteristics discovered at prior levels. Additionally, the input photos are introduced after the input classification process has converted the input pictures into the distribution of probabilities over classes and after the neural network has been trained via back propagation. The model delivers the following parameters throughout all relevant epochs: recall, precision, F1 score, confusion matrix, training accuracy and loss, validation accuracy and loss, recall, and accuracy and loss plot graphs. There are only 10 epochs in the pre-trained CNN model, thereby lowering its effectiveness in feature extraction [15].

Gopanae et al. proposed the use of Support Vector Machine (SVM) to identify fake bank notes. The suggested strategy is focused on exploiting open access to identify counterfeit money. A banknote's legality can be determined using the suggested system by examining specific security features like watermarks, latent images, security threads, etc. Algorithms for machine learning are used to detect fake currency. As part of the technique, certain security components are extracted and encoded. A support vector machine (SVM) is employed to extract security features, categorize features, and find features from the input image. The proposed system uses methods such image acquisition, region of interest extraction from the image, feature extraction, and machine learning algorithms to determine whether an input image of a banknote is authentic or fraudulent. However, a comprehensive verification of the proposed approach requires extensive evaluation in a wide range of real-world scenarios. Furthermore, deep learning techniques can be used on a substantial amount of training data to provide predictions that are more accurate [16].

III. Problem Statement

India and many other nations have suffered greatly and the issue has become grave. Fake notes are a catastrophe in almost every country. Systematic methods are utilized to distinguish between real and fake bank notes using bank note authentication. Nowadays, a wide variety of applications are accessible, including automated teller machines, self-serve checkout lanes, money exchange companies, hotels, banks, and retail establishments, among others. Forged bank notes are becoming more prevalent over time. In spite of the ease with which counterfeit bank notes may now be produced because of advances in scanning and printing technology, it is now incredibly challenging to distinguish between real and fake currency. However, current systems sound implausible to people like shopkeepers because of their price, requirement for a large power supply, and size. As a result, these people are more likely to encounter forgers. It is required to categorize and identify between photos of real and fake bank notes throughout training [17]. As a result, numerous conventional methods, such as KNN and Decision Trees, were employed; however, advanced machine learning algorithms, such as ELM, XGBOOST, or MLP, were not used. Applying either an image or a video as an input feed, the issue can be resolved by applying deep learning algorithms to recognize the currency.

IV. Proposed Counterfeit Currency Detection Using GAN-CNN-RNN with ABO Approach

A. Data Generation using GAN

The dataset should include a diverse collection of real banknotes and corresponding counterfeit samples. The GAN will be trained on this dataset to learn the underlying distribution of genuine banknote features and generate synthetic counterfeit currency images that closely resemble real counterfeit notes, ultimately enhancing the performance of the counterfeit currency detection algorithm [18].

The GAN, which has acquired prominence for their outstanding synthetic data production capabilities, serve as the foundation for the data generation. Two networks are trained simultaneously by a GAN. By examining the distribution of underlying data, the first network, known as the Generator network, frequently produces intimation images. . In order to determine if an input sample is real or fake (i.e., whether it came from the generator), the second network, the discriminator, is used. Our suggested solution, which is based on GANs, models the underlying data distribution to differentiate between seven target categories with a very low inter-class variation.

The generator, K(z;Wg), is a network of neurons with parameters such as Wg and an earlier distribution of pz(z) on the input random vector z. A convolutional neural network's discriminator E(x;Wd) decides whether the input vector x comes from the generator's dispersion p_g . A (n + 1) dimensional vector illustrating this probability is the result. In order increase the likelihood that it will correctly recognize its input, x, the discriminator, E, tries to distinguish between images created artificially by specimen p_g and those acquired from the authentic data samples during training. Additionally, by attempting to minimize log (1 D(G(z)), the generator, K, is trained to deceive the discriminator. By streamlining the procedure, you may do both.

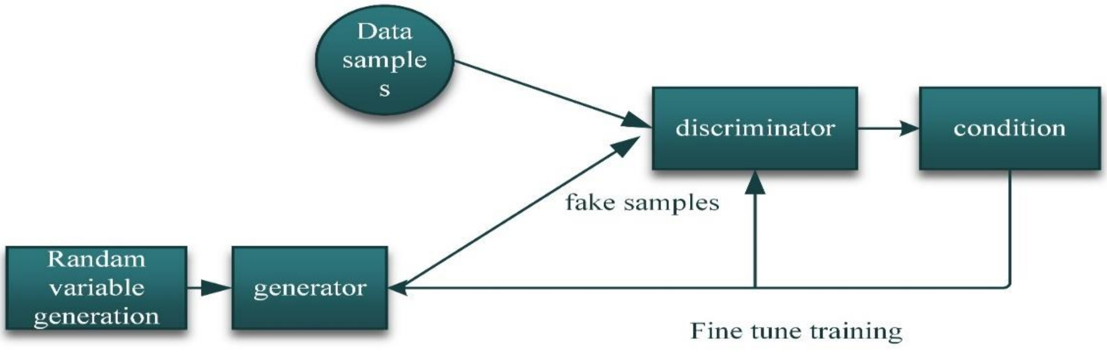


Fig. 1. GAN network.

In Fig.1, the generator, H(y;W$_h$) is a network of neurons with the values W$_h$ and a prior distribution of p$_z$(y) on the input noise vector y. An (n + 1) dimensional vector is the output of the convolutional neural network that serves as the discriminator, I(x;W$_i$).

$$min_H max_I V(H,I) = E_{x\sim qdata(x)}[\log(x)] + E_{y\sim q_y(y)}[\log(1 - I(H(y)))] \qquad (1)$$

In order to increase the possibility that the correct label will be assigned to its input y, the discriminator, I, attempts to distinguish between images acquired from the genuine data samples during training and those intentionally made by sampling ph. The generator H is also taught to deceive the discriminator by seeking to reduce $[\log(1 - I(H(y)))]$.

Our major objective is to be able to distinguish between the numerous classes that the photos comprise, even though the generator and discriminator are merely designed to discern actual images apart from false ones. Our key objective is to be able to discern between the many classes that the photographs represent. To determine if a photograph is authentic or fake, the (n + 1) th unit in the last layer of I applies the aforementioned loss function. This layer's remaining n units are trained using the following common cross-entropy loss function:

$$\boldsymbol{L_{supervised}} = -\sum_{I(x)} m(I(x)) log(n(I(x))) \quad (2)$$

Here, the discriminator projected class probability is represented by the symbol m(I(x)), which also serves as the input x's accurate class label. The discriminator network can be trained simultaneously using two loss functions—one for classifying inside actual images and another for distinguishing between fake and real images—thanks to the introduction of a second label for the false class [19].

B. Feature Extraction using CNN

The feature extraction component's goal is to extract the elements of the paper banknote sample's descriptive features. In this instance, a CNN is employed to extract features. Some of the unique features that CNN frequently employs to identify banknotes include color, size, form, and tactile aspects. To extract the CNN feature, a supervised variation of the CNN model is used. Input, hidden, and output layers are present in the CNN. The layer of convolution and the subsampling processes layer are both referred to as the feature extractor layer in the CNN model's hidden layer. The convolutional layers pick up on a variety of local traits that make up the image of a birr note, such as the identifying mark, security thread, tin and wide golden strip, and numerous other security features. As the image moves through each layer, the filters may spot more complex characteristics. Each neuron's input in the convolution layer is coupled to the local receptive field of the layer above it in order to extract the local feature [20]. Rectifier Unit, the most frequently used activation function for the outputs of the CNN neurons, is referred to as ReLu [21].

The ReLU can be written as in eqn. (3),

$$\sigma = \max(\sigma, v) \qquad (3)$$

Here v is the input.

Typically, CNN retrieves high-level characteristics from the model's final output layer. It is also crucial to remember that grayscale and RGB images are typically examined by CNN models utilizing two-dimensional (2-D) convolution filters. The input picture is only N1 by N1 pixels, and the first 2-D convolution layer has n1 2-D convolution kernels with a size of p1 by p1. Because of this, the initial 2-D convolution layer will produce n1 feature maps that are (N1- p1 +1) by (N1- p1 +1) in size, depending on the size of the 2-D convolution kernels. The dot product of the weight matrix and the location of the local area (x, y) is also used to construct each feature map; and you can use the formula to determine the value of a neuron's V_{ij}^{ab} at position (a, b) on the jth feature map in the ith layer.

$$V_{ij}^{ab} = \sigma\Big(c_{ij} + \sum_n x \sum_{p=0}^{p_i-1} x \sum_{q=0}^{q_i-1} w_{ijn}^{p} V_{(i-1)n}^{(a+p)(b+q)}\Big) (4)$$

Where, σ(·) indicates the ith layer's activation function, c$_{ij}$ is an additive bias of the ith layer's jth feature map,

Ki and Pi symbolizes the height and width of the 2-D convolution kernel, respectively, while m indexes the association between the feature map in the (i$_1$)th layer and the present (jth) feature map, and $w_{ijn}^{p}$is a weight for input $V_{(i-1)n}^{(a+p)(b+q)}$ with an offset of (q, p) in 2-D convolution kernel. The 2-D pooling approach is used to minimize the feature maps' resolution.

In Fig. 2, the features of 2-D CNN architecture is composed of three layers: an input layer, a 2-D CNN block (stacked with many Conv layers and POOL layers), and a fully connected neural network (FNN) block. Feature maps' resolution is decreased using the pooling procedure, which begins after the convolution stage. The features can then become location-invariant. In the first POOL layer, where n3 = n2/k2, there are m2 x n3x 1 nodes and m2 kernels of size k2 x 1. In addition, there is a link between the neurons of the POOL layer and the n 1 patch of the Conv layer. Applying equation 5 below completes the pooling step of the max-pooling technique.

$$d_j = max_{nx1\left(d_i^{kx1}\, u(K,1)\right)} \qquad (5)$$

The largest value in the neighborhood, and u(k, 1) specifies a window function matching to the small n 1 patch of the Conv layer. Richer and more robust features can be extracted using a features extraction method that can quickly and efficiently coordinate the training of subnetworks [22].

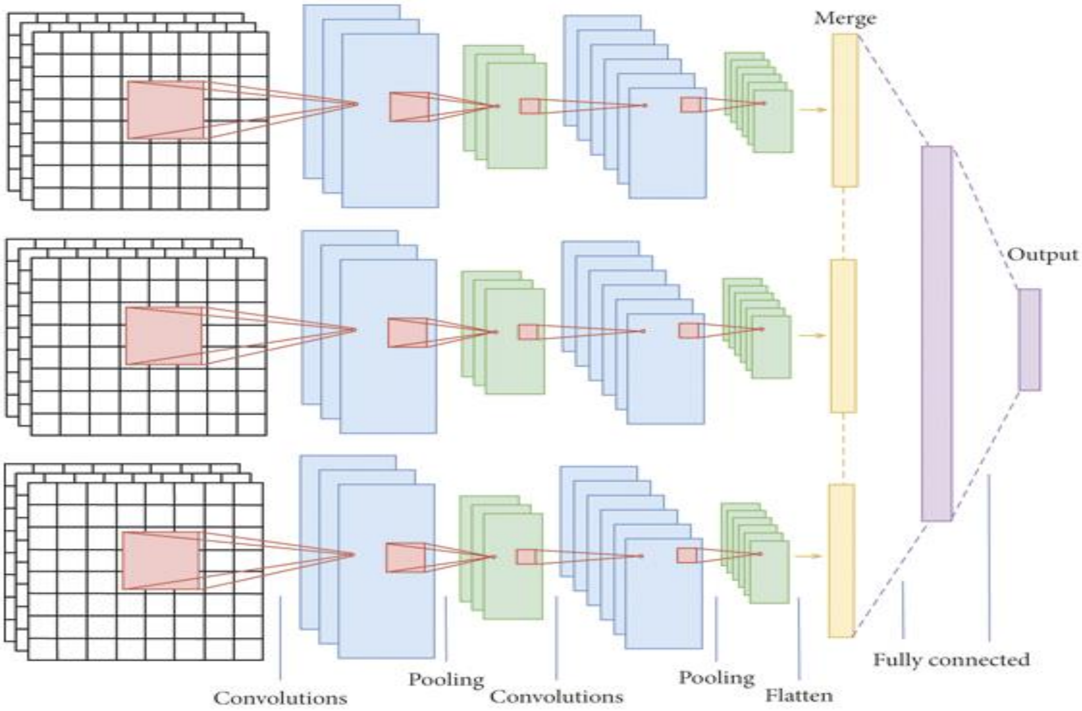


Fig. 2. 2-D CNN architecture.

C. Feature Selection using African Buffalo Optimization (ABO)

The African Buffalo Optimization (ABO) mimics the alert ('maaa') and alarm ('waaa') behaviors of African buffalos when they are out hunting. When searching for food and avoiding predators, African buffaloes primarily use these two noises to coordinate their movements. The waaa noise is used to rouse the buffalos to move on and investigate the search area, whereas the maaa noise instructs the buffalos to remain and make use of their immediate surroundings because it is secure and has enough of pastures. The buffaloes are aided in their food source seek by these noises. The African Buffalo Optimization (ABO) mimics the alert ('maaa') and alarm ('waaa') behaviors of African buffalos when they are out hunting. When searching for food and avoiding predators, African buffaloes primarily use these two noises to coordinate their movements. The waaa noise is used to rouse the buffalos to move on and investigate the search area, whereas the maaa noise instructs the buffalos to remain and make use of their immediate surroundings because it is secure and has enough of pastures. The buffaloes are aided in their food source seek by these noises.

1) The working of the ABO algorithm: The method starts by purposely assigning each buffalo a random location inside the N-dimensional space in order to initialize the population of animals. The algorithm then determines the bn_{max} (the herd's ideal position) and bm_{max} (each individual buffalo's location) with respect to the ideal solution. The position vector for that specific buffalo is saved if the person's current fitness level exceeds their maximal fitness level (bm_{max}). The fitness is saved if it exceeds the herd's maximum, or bn_{max}, which is the entire fitness of the herd. The position of the current buffalo is updated, and the population's future buffalo are taken into account. The ABO algorithm is shown below, same to algorithm 1.

2) Controlling the movement of the buffalos: According to the communal intellect of the herd after engaging in a collective decision-making process, this democratic equation (10) in figure allows for the choice to either continue in order to further utilize the environment or to go on to explore other location. Depending on the outcome of the democratic eqn. (6), the maaa eqn. (5) advises the buffalos to move on and explore different areas after carefully examining the two conflicting pressures (bn_{max} and bm_{max}). The option, which identifies the discrete time interval that the buffalo must migrate through, has a default value of 1.0. When these two equations are applied, the position of the buffaloes is altered. To realize the full potential of the algorithm, it is required to investigate the ability of the African Buffalo Optimization to search a range of solution spaces, including separable and non-separable, limited and unconstrained, mono-modal and multi-modal. One extreme is present in a mono modal function, which is defined as having only one minimum or maximum within the range specified for x. In a manner similar to this, it is said to be multimodal if a function contains more than one peak, either on the minimum or minimum sides.

In addition, a function is considered to be dissociable if it can be written as the sum of 'p' functions with just one variable. It is harder to optimize non-separable functions. This is due to the fact that accurate search directions need the presence of two or more components in the search space (or solution vector). The issue is complicated when a function has many modes. The last challenge that optimization algorithms must overcome is the situation of many dimensions. This is because as issue dimensions increase, so do the number of local optima. These multimodal benchmark functions test an algorithm's resistance to local minima or maxima. An algorithm will stack in a local minimum if its exploration capacity is inadequate. In this section, we'll examine a few of the flat-surfaced functions. Because these functions don't provide enough data to enhance the search, they pose certain challenges for algorithms [23].

The parameters used are N,α,lq1,lq2

A large group of n buffaloes, where each buffalo stands for a potential solution.

- The buffaloes' index k, where k is in the range of [1,...,N].
- The alpha value, excluding the zero, with a domain in [1,1][1,1].
- The learning factors lq1 and lq2 with a [0,1][0,1]real number domain.
- The exploitation man oeuvre is represented by the mk variable.
- The bnmax$_k$ variable has the highest fitness for the herd.
- The bmmax$_k$ variable represents each buffalo k's optimal finding location.
- The w$_k$ variable, which represents the exploration move, is the last.

$$m_{k+1} = m_k + lq1(bnmax - w_k) + lq2(bmmax - w_k) \qquad (6)$$

On Line 6, Eqn. (5) indicates the location updated by the buffalo k [24].

$$w_{k+1} = \frac{w_k + w_m}{\pm\alpha} \qquad (7)$$

The location update formula is Since Eq. (7) is a straightforward random number generator, the search inside the search space is likely to be aimless, leading to a relatively inefficient solution or, in certain cases, premature convergence. These issues, nevertheless, might be resolved by utilizing chaotic and levy distribution features.

Algorithm 1: *ABO algorithm.*

```
Input: N, α, lq1, lq2
Output: Original or fake currency

Set the parameters initialized
Make random answers for N buffaloes.
While the criterion of the terms has not ended do
    for all buffaloes do
        update the buffaloes $m_k$
        update the location $w_k$
        if the problem is minimization then
            if fitness $m_k$ < fitness $bnmax_k$ , then
                update the $bnmax_k$
            end
        else
            if fitness $m_k$ > fitness $bnmax_k$ then
                update the $bnmax_k$
            end
        end
    end
        update bmmax from bgmax
    if the term criterion was met then
    output
    end
end
```

ABO uses just a few parameters, principally the learning parameters lq1 and lq2, to guarantee quick convergence. Depending on the algorithm's concentration at a given iteration, these parameters allow the animals to move towards greater exploitation or exploration.

D. Classification using RNN

The classification phase employs RNN. An RNN connects its nodes in a way that closely resembles how neurons in the human brain are connected to one another. After processing the signal, it just received, the artificial neuron sends it to the other neurons or nodes that are connected to it. Weights frequently exist in neurons and synapses to change how learning occurs. The strength of the signal can be changed by varying the weight as it passes from the input layers to the output layers. An RNN has additional layers in addition to the input and output layers. Three hidden layers are the absolute minimum for an RNN. Input, output, and hidden units, all of which use weighted calculations, make up the fundamental elements of RNNs [25].

In Fig. 3 a sequence of input vector is given as A = (a$_1$, a$_2$,.. a$_T$).

The sequence of hidden vector as B = (b$_1$, b$_2$, . . . ,b$_T$) and

Sequence of output vector as C = (c$_1$, c$_2$, . . . , c$_T$) with t = 1 to T as follows:

$$b_t = \sigma(V_{ab}a_t + V_{bb}b_{t-1} + d_b) \qquad (8)$$

$$c_t = V_{bc}b_t + d_c \qquad (9)$$

d is a bias element, function is an exponential activation function, and V is a weight matrix. The output of the hidden layer at each t-time steps is denoted by the symbol b$_t$, while the output of the hidden layer before it is denoted by b$_{t-1}$ in Eqn. (8).

All of the units in the hidden are replaced by LSTM, the RNN's memory unit. The regulatory gates open, activating the memory cells. These gates regulate the flow of information coming in and leaving out. Between an input gate and an output gate is positioned a forget-gate. If the originally saved details are no longer needed, forget gates are able to recover the linear unit's state. These gates are composed of uncomplicated sigmoid threshold units. Memory block layers for these activation functions fluctuate between 0 and 1. At least one memory cell has to be encountered by each memory block.

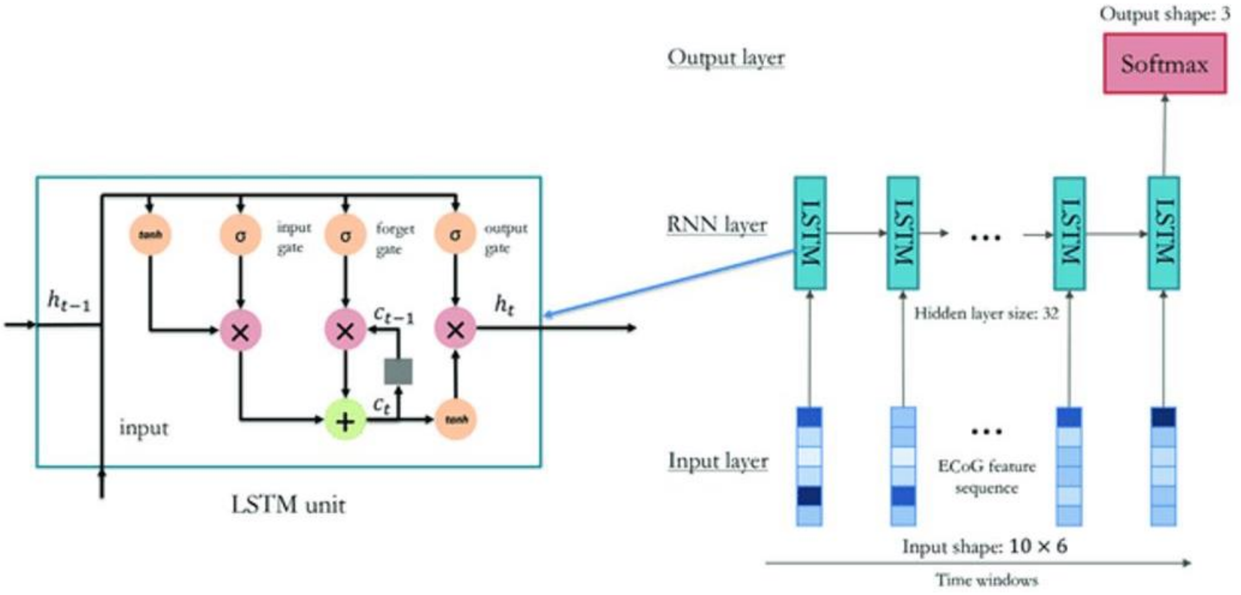


Fig. 3. A simple RNN.

The output $u^b k(t)$ of a memory cell is computed as

$$u^b k(t) = u^{out} k(t) b(s_{b_i}(t) \tag{10}$$

Where, $u^b k(t)$is the activation of output gate, s_{b_i} is the output gate's internal state, and b is the output hidden layer.

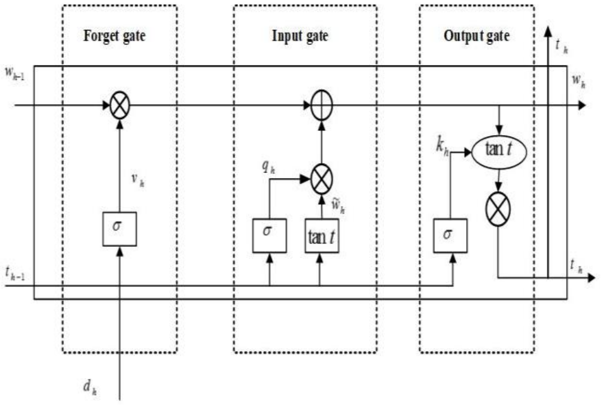


Fig. 4. A basic LSTM network diagram.

In cases when natural language processing is problematic, the amount of text that is present in each data instance is not given as a fixed value. In order to comprehend every word of text, the sequence dimensions are scaled down to a certain degree. If the sequence size is lower than the desired value, the sequence is filled until the value is reached. The surplus is eliminated if the length of the sequence exceeds the necessary value. In Fig. 4, the Convolutional LSTMs are capable of simultaneously encoding long-term relationships and extracting characteristics from the time-frequency domain. The convolution procedure is carried out using linear activation in the first stage. The second phase, also known as the rectifying step, is when the activation function used in the first phase is identified and converted into a linear function. By applying a pooling function to reduce the dimensionality of the time series data, the network's training period is shortened. The collected feature map must also be down sampled without the depth being altered. For the maximum pooling operation of the pooling layer, which is launched on the maximum pooling window, it is customary to choose the maximum values of the convolution layer. This differs from the convolutional layer parameter. Fig. 5 shows the comprehensive flow diagram for counterfeit currency detection and classification. Alternative pooling methods, such minimal and average pooling, are, however, often used.

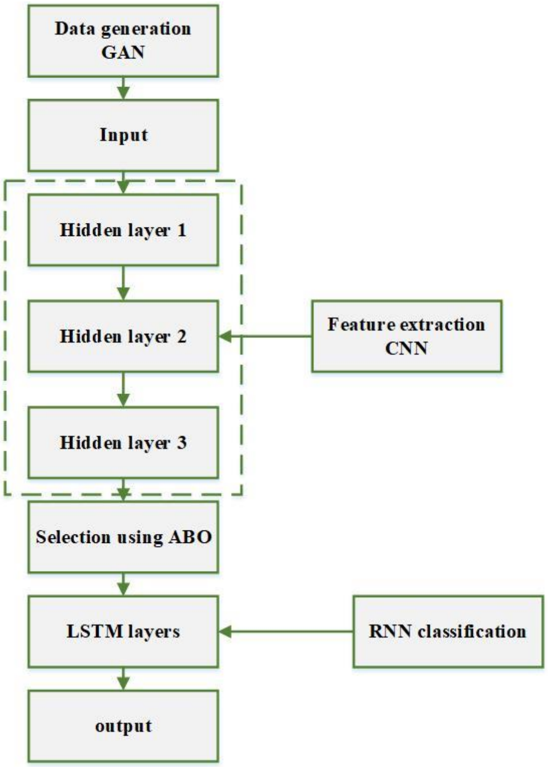
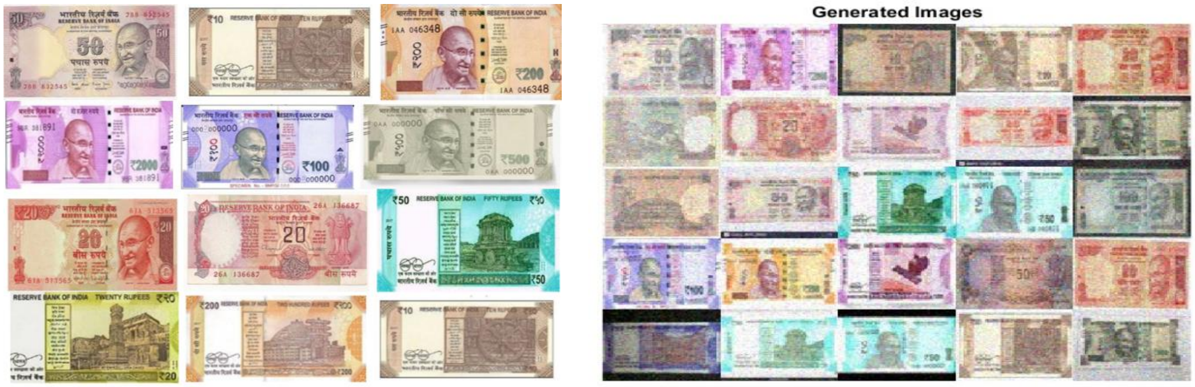


Fig. 5. Comprehensive flow diagram for counterfeit currency detection and classification.

V. Result and Discussion

The process is implemented in MATLAB software in windows 10 platform. The generating the data from GAN is given as input for feature extraction by CNN, feature selection is carried out by the ABO and classification is by RNN. The experiment assessed the models using four evaluation metrics: accuracy, F1-score, precision, and recall. Fig. 6 shows the a) The original images of currency notes used for training the GAN-CNN-RNN model and b) Generated Fake Images - The synthetic counterfeit currency images generated by the GAN-CNN-RNN model during the training process.

GAN generates the images from the dataset. The elapsed time, iteration time and epoch time are plotted in the graph as shown in Fig 7. An ROC curve demonstrates how better the classification model performs at each level of categorization, seen in Fig. 8. On this curve, two parameters are plotted: 100% True Positive and False Positive Rate. A confusion matrix serves as a visual representation and summary of a classification algorithm's results. In Fig. 9, the proposed algorithm's confusion matrix is provided. Fig. 10 shows a concise representation of the training process for the counterfeit currency detection and classification model, utilizing GAN-CNN-RNN with African Buffalo Optimization.



(a)

(b)

Fig. 6. (a) The original images of currency notes used for training the GAN-CNN-RNN model and (b) Generated fake images - The synthetic counterfeit currency images generated by the GAN-CNN-RNN model during the training process.

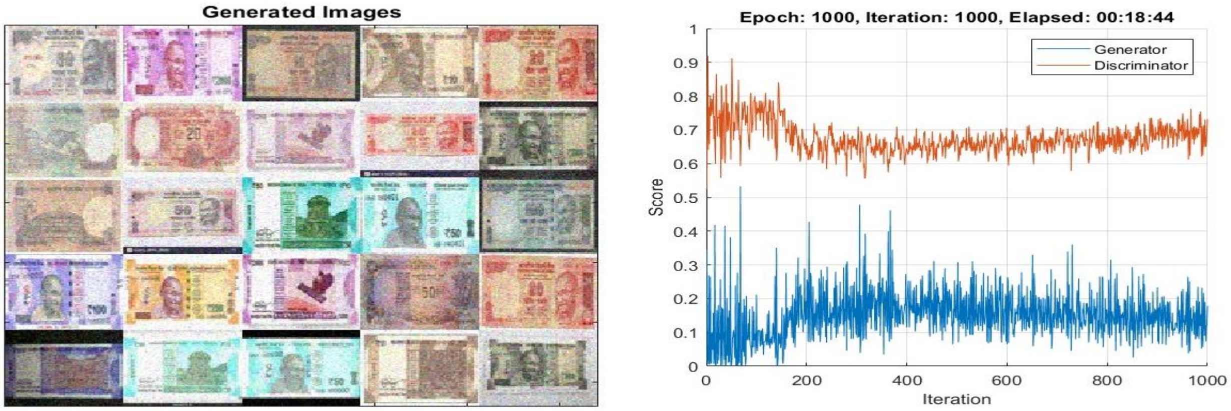


Fig. 7. Generated images using GAN: Synthetic currency samples created by the GAN during training.

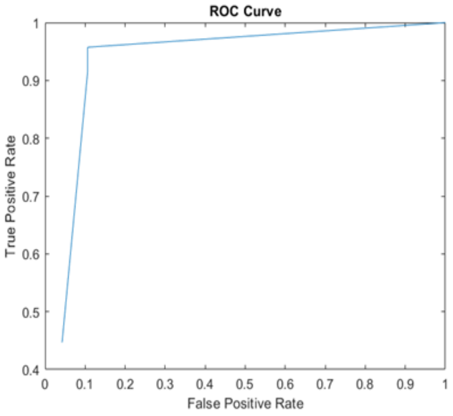


Fig. 8. ROC curve.

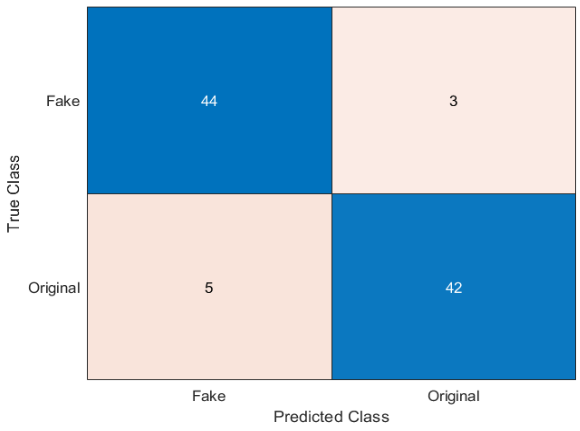
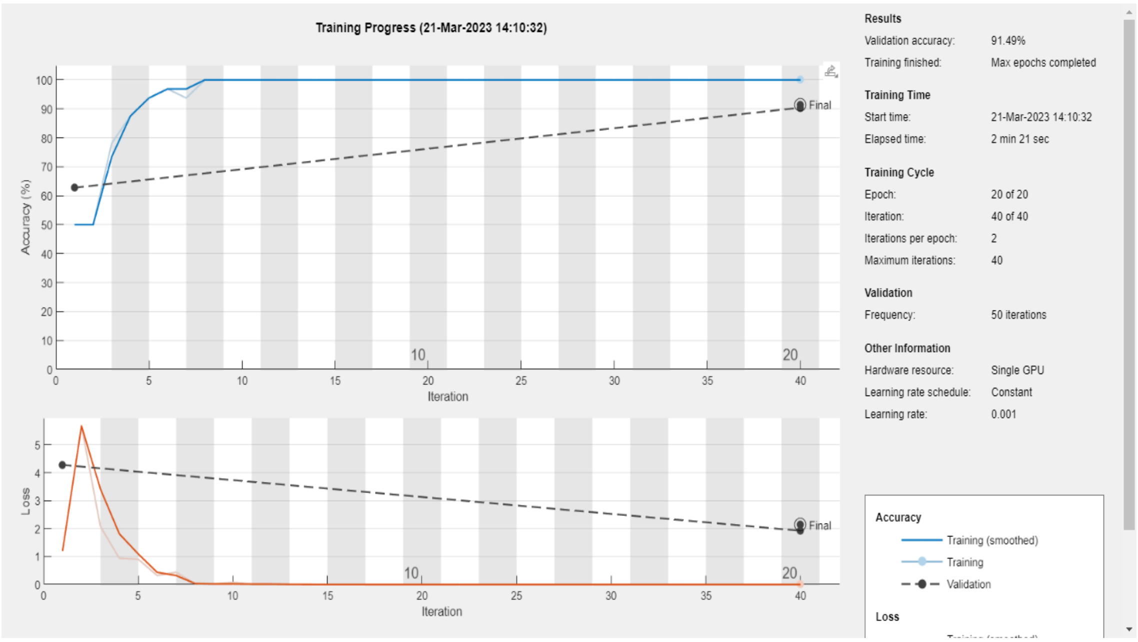


Fig. 9. Confusion matrix.



Results	
Validation accuracy:	91.49%
Training finished:	Max epochs completed
Training Time	
Start time:	21-Mar-2023 14:10:32
Elapsed time:	2 min 21 sec
Training Cycle	
Epoch:	20 of 20
Iteration:	40 of 40
Iterations per epoch:	2
Maximum iterations:	40
Validation	
Frequency:	50 iterations
Other Information	
Hardware resource:	Single GPU
Learning rate schedule:	Constant
Learning rate:	0.001

Fig. 10. A concise representation of the training process for the counterfeit currency detection and classification model, utilizing GAN-CNN-RNN with African Buffalo Optimization.

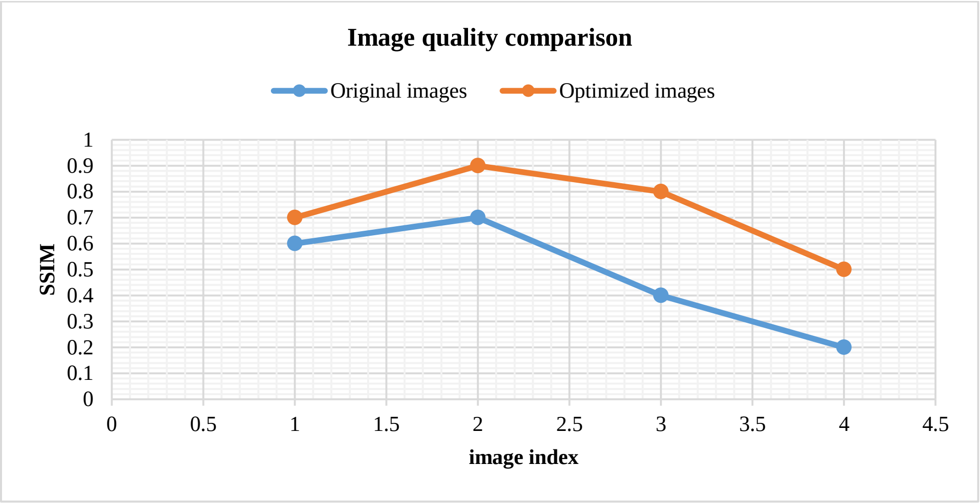


Fig. 11. Comparison using ABO optimization.

In the above Fig. 11, by using ABO optimization, the Quality of the original image is always higher than the optimized image that shows the image quality comparison between image indexes Vs SSIM.

A. Performance Metrics

These parameters are specifically defined as:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (11)$$

$$Precision = \frac{TP}{TP+FP} \qquad (12)$$

$$Recall = \frac{TP}{TP+FN} \qquad (13)$$

$$F1score = \frac{2*Recall*Precision}{Recall+precision} \qquad (14)$$

$$Specificity = \frac{TN}{TN+FP} \qquad (15)$$

The amount of data that were correctly declared as positive out of all the actually positive data is referred to as the TP. TN indicates to the proportion of data that were incorrectly classified as negative out of all the data that were actually negative. The term "FN" indicates to the number of variables that the model misclassified as negative even though they were positive in the dataset. The false positive rate, or FP, is the number of variables that the model incorrectly identified as positive even though they were in fact negative in the dataset. . Recall is the proportion of positive data classified as such in the dataset as compared to the number of positive data identified as such by the model. In terms of the total quantity of data that were classified as positive, precision is the portion of data that the model correctly recognized as positive. Simply said, the harmonic average of recall and precision is the F1-score focusing on Accurate Predictions with Specificity.

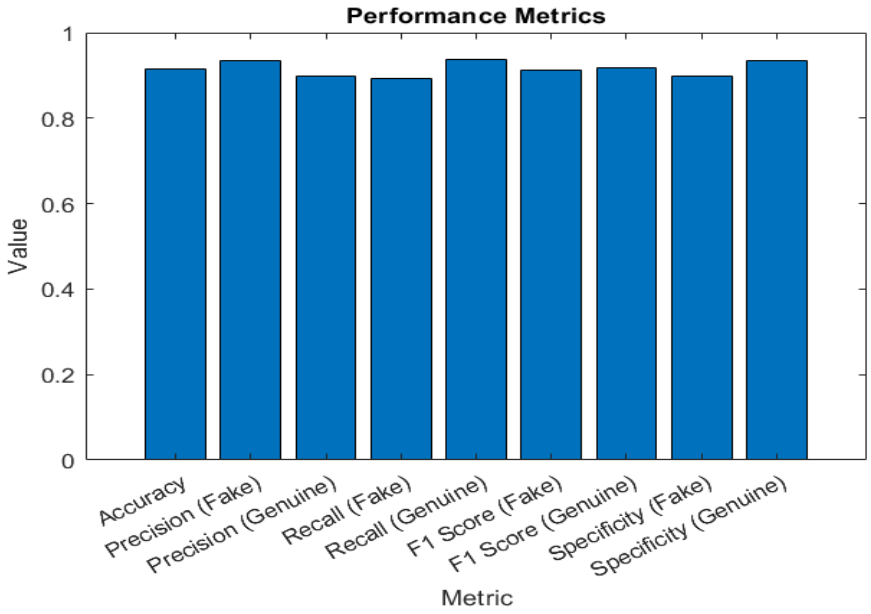


Fig. 12. Experimental result analysis.

In Fig. 12, the proposed system's performance metrics is tabulated. The overall accuracy for the process ranges between 0.8 and 1. The value of fake and genuine notes for precision, recall, F1 Score, Specificity is given above.

Comparison of the proposed model performance results with the existing methods is mentioned in Table I. For clear understanding comparative analysis of sensitivity and specificity are graphically represented in Fig. 13.

TABLE I. COMPARED RESULT IN TERMS OF SENSITIVITY AND SPECIFICITY

Method	Specificity	Sensitivity
KNN	94.75	89
SVM	94	91.42
SSD model-CNN	92	95
Proposed buffalo optimization approach	98.92	96.8

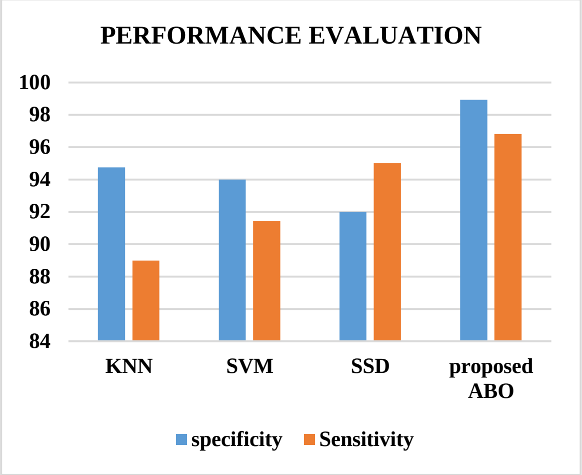


Fig. 13. Comparison of sensitivity and specificity for existing and proposed methods.

VI. Conclusion

This method suggests using of deep learning to detect counterfeit money. The main benefit of employing a deep learning approach is the ability to automatically extract task-specific features from the data; in this case, the choice to utilize deep learning is affected by the graph-structured nature of the data. This eliminates the need for "handcrafted" features. Our model exhibits extremely high accuracy and stable behavior in a variety of difficult situations involving massive amounts of real data, underscoring the enormous promise of hybrid deep learning techniques for fake cash identification. Future research is still needed on many fascinating phenomena and concepts. On varied banknote picture quality, the effectiveness of the GAN, RNN, ABO, and CNN feature extraction technique was assessed. The respective recognition accuracy was 99.2%. Based on recognition precision the CNN is the best feature extraction method in terms of the model's overall goal. As a result, the CNN model is the best option for classifying the banknote, and the RNN also favors the best data creation, ABO optimization for selection, and RNN for classification. The given model significantly advances the study of identifying fake cash. The end research findings are satisfactory and include identifying the currency range in the categorization label, currency denomination, and currency front and rear. It is also possible to state that the accuracy of currency recognition is extremely high. Following study, we discovered that our recognition is quick and precise when the currency is in a clear state over the full screen and the angles are parallel.

References

[1] Toqeer Ali and S. Jan, ‘DeepMoney: Counterfeit Money Detection Using Generative Adversarial Networks’. figshare, p. 5295688709 Bytes, 2019. doi: 10.6084/M9.FIGSHARE.9164510.V3.

[2] M. Jadhav, Y. K. Sharma, and G. M. Bhandari, ‘Currency Identification and Forged Banknote Detection using Deep Learning’, in 2019 International Conference on Innovative Trends and Advances in Engineering and Technology (ICITAET), SHEGAON, India: IEEE, Dec. 2019, pp. 178–183. doi: 10.1109/ICITAET47105.2019.9170225.

[3] N. Chaubey, S. Parikh, and K. Amin, Eds., Computing Science, Communication and Security: First International Conference, COMS2 2020, Gujarat, India, March 26–27, 2020, Revised Selected Papers, vol. 1235. in Communications in Computer and Information Science, vol. 1235. Singapore: Springer Singapore, 2020. doi: 10.1007/978-981-15-6648-6.

[4] I. Aldridge and M. Avellaneda, ‘Neural Networks in Finance: Design and Performance’, J. Financ. Data Sci., vol. 1, no. 4, pp. 39–62, Oct. 2019, doi: 10.3905/jfds.2019.1.4.039.

[5] Q. Zhang, W. Q. Yan, and M. Kankanhalli, ‘Overview of currency recognition using deep learning’, J. Bank. Financ. Technol., vol. 3, no. 1, pp. 59–69, Apr. 2019, doi: 10.1007/s42786-018-00007-1.

[6] M. Laavanya and V. Vijayaraghavan, ‘Real time fake currency note detection using deep learning’, Int J Eng Adv TechnolIJEAT, vol. 9, 2019.

[7] K. Bhavsar, K. Jani, and R. Vanzara, ‘Indian Currency Recognition from Live Video Using Deep Learning’, in Computing Science, Communication and Security, N. Chaubey, S. Parikh, and K. Amin, Eds., in Communications in Computer and Information Science, vol. 1235. Singapore: Springer Singapore, 2020, pp. 70–81. doi: 10.1007/978-981-15-6648-6_6.

[8] M. A. Khan, R. Etminani-Ghasrodashti, A. Shahmoradi, S. Kermanshachi, J. M. Rosenberger, and A. Foss, ‘Integrating Shared Autonomous Vehicles into Existing Transportation Services: Evidence from a Paratransit Service in Arlington, Texas’, Int. J. Civ. Eng., vol. 20, no. 6, pp. 601–618, Jun. 2022, doi: 10.1007/s40999-021-00698-6.

[9] B. Shilpa, S. Neha, B. Prerana, U. Ananya, and P. H. Ashwini, ‘FAKE CURRENCY DETECTION USING DEEP LEARNING’.

[10] A. Saxena, P. K. Singh, G. P. Pal, and R. K. Tewari, ‘Fake currency detection using image processing’, Int. J. Eng. Technol., vol. 7, pp. 199–205, Jan. 2018, doi: 10.17577/IJERTV8IS120143.

[11] S. Shahani, A. Jagiasi, and P. R., ‘Analysis of Banknote Authentication System using Machine Learning Techniques’, Int. J. Comput. Appl., vol. 179, no. 20, pp. 22–26, Feb. 2018, doi: 10.5120/ijca2018916343.

[12] C. G. Pachón, D. M. Ballesteros, and D. Renza, ‘Fake Banknote Recognition Using Deep Learning’, Appl. Sci., vol. 11, no. 3, p. 1281, Jan. 2021, doi: 10.3390/app11031281.

[13] L. Guarnera, O. Giudice, and S. Battiato, ‘DeepFake Detection by Analyzing Convolutional Traces’, in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA: IEEE, Jun. 2020, pp. 2841–2850. doi: 10.1109/CVPRW50498.2020.00341.

[14] J. D’cruz, M. Jose, M. Eldhose, and B. Jose, ‘FAKE INDIAN CURRENCY DETECTION USING DEEP LEARNING’.

[15] G. Gebremeskel, T. A. Tadele, D. W. Girmaw, and A. O. Salau, ‘Developing a Model for Detection of Ethiopian Fake Banknote Using Deep Learning’, In Review, preprint, Dec. 2022. doi: 10.21203/rs.3.rs-2282764/v1.

[16] S. Gopane and R. Kotecha, ‘Indian Counterfeit Banknote Detection Using Support Vector Machine’, SSRN Electron. J., 2020, doi: 10.2139/ssrn.3568724.

[17] M. Jadhav, Y. Sharma, and G. Bhandari, ‘Forged Multinational Currency Recognition System Using Convolutional Neural Network’, in Proceedings of 6th International Conference on Recent Trends in Computing, R. P. Mahapatra, B. K. Panigrahi, B. K. Kaushik, and S. Roy, Eds., in Lecture Notes in Networks and Systems, vol. 177. Singapore: Springer Singapore, 2021, pp. 471–479. doi: 10.1007/978-981-33-4501-0_43.

[18] R. Samuel, B. D. Nico, P. Moritz, and O. Joerg, ‘Wasserstein GAN: Deep Generation applied on Bitcoins financial time series’, 2021, doi: 10.48550/ARXIV.2107.06008.

[19] H. Rashid, M. A. Tanveer, and H. Aqeel Khan, ‘Skin Lesion Classification Using GAN based Data Augmentation’, in 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany: IEEE, Jul. 2019, pp. 916–919. doi: 10.1109/EMBC.2019.8857905.

[20] A. S. Alene, ‘Ethiopian Paper Currency Recognition System: An Optimal Feature Extraction’, vol. 7, no. 8, 2019.

[21] R. K. Kaliyar, A. Goswami, and P. Narang, ‘FakeBERT: Fake news detection in social media with a BERT-based deep learning approach’, Multimed. Tools Appl., vol. 80, no. 8, pp. 11765–11788, Mar. 2021, doi: 10.1007/s11042-020-10183-2.

[22] Y. Liu, H. Pu, and D.-W. Sun, ‘Efficient extraction of deep image features using convolutional neural network (CNN) for applications in detecting and analysing complex food matrices’, Trends Food Sci. Technol., vol. 113, pp. 193–204, Jul. 2021, doi: 10.1016/j.tifs.2021.04.042.

[23] J. Odili and M. Kahar, ‘Numerical Function Optimization Solutions Using the African Buffalo Optimization Algorithm (ABO)’, Br. J. Math. Comput. Sci., vol. 10, no. 1, pp. 1–12, Jan. 2015, doi: 10.9734/BJMCS/2015/17145.

[24] B. Almonacid, F. Aspée, and F. Yimes, ‘Autonomous Population Regulation Using a Multi-Agent System in a Prey–Predator Model That Integrates Cellular Automata and the African Buffalo Optimization Metaheuristic’, Algorithms, vol. 12, no. 3, p. 59, Mar. 2019, doi: 10.3390/a12030059.

[25] P. S. Muhuri, P. Chatterjee, X. Yuan, K. Roy, and A. Esterline, ‘Using a Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) to Classify Network Attacks’, Information, vol. 11, no. 5, p. 243, May 2020, doi: 10.3390/info11050243.