IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 3, June 2025

Impact Factor: 7.67

Deep Fake Image Detection Using GAN

Prof. Bhagwati Galande[1], Ritika Sonawane[2], Sonali Gade[3], Anushka Mandlik[4], Priyanka Harnawal[5]

Assistant Professor, Department of Information Technology[1]

Students, Department of Information Technology, College [2,3,4,5]

Smt. Kashibai Navale College of Engineering, Pune, Maharashtra, India

Abstract: *In today's digital age, the rise of deepfake technology has become a significant concern due to its ability to generate hyper-realistic but fake media content. This has led to a serious threat to the authenticity of visual data and trust in digital platforms. Traditional detection methods often fail to adapt to the rapidly evolving nature of generative models.. Our system aims to tackle this issue using advanced algorithms and techniques such as GAN (Generative Adversarial Networks), CNN (Convolutional Neural Networks), Image Processing, and Deep Learning. The system works by extracting visual features from images and detecting inconsistencies using trained deep learning models. The proliferation of deepfake technology poses a significant threat to digital content authenticity and public trust. Deepfakes utilize advanced generative models to create highly realistic but manipulated media, making it increasingly difficult to distinguish between real and synthetic content. This paper presents a comprehensive deepfake detection system that leverages Generative Adversarial Networks (GANs), Convolutional Neural Networks (CNNs), and hybrid learning techniques to accurately classify AI-generated images. Our approach integrates a user-friendly web application developed using HTML, CSS, and JavaScript for frontend interaction, and a Python Flask backend for managing deep learning model inferences. The detection engine combines models trained on diverse datasets like CIFAKE and custom fruits datasets to increase robustness against various generative patterns. This system contributes a scalable and accessible solution for combating the growing concern of synthetic media proliferation..*

Keywords: Deepfake Detection, Generative Adversarial Networks (GANs), Convolutional Neural Networks (CNNs), AI-generated Images, Hybrid Deep Learning, Image Forgery Detection, Real-time Classification, Flask Web Application, MongoDB, Content Authentication

I. INTRODUCTION

The rapid emergence of deepfake technology has raised significant concerns about the trustworthiness and authenticity of digital media. Leveraging advanced generative frameworks such as Generative Adversarial Networks (GANs), deepfake methods create highly convincing yet entirely fabricated images and videos. These artificially generated media have the potential to spread false information, violate privacy rights, and diminish public confidence in digital content.Conventional detection approaches, which often depend on manual feature engineering or basic machine learning models, struggle to keep up with the ever-improving realism and complexity of deepfake content. As a result, there has been a growing shift towards employing deep learning methodologies, especially Convolutional Neural Networks (CNNs) and hybrid models that combine various learning strategies, to enhance detection performance.This research introduces a scalable and effective detection system that integrates GANs with CNN architectures in a hybrid framework. The system is capable of performing real-time analysis and classification of images, enabling it to distinguish authentic media from synthetic fabrications more accurately. Training the model on extensive and diverse datasets such as CIFAKE along with custom image collections further strengthens its ability to recognize a wide range of manipulation patterns and generative techniques.

DOI: 10.48175/568

II. RELATED WORK

The detection of deepfake images and videos has gained significant attention due to the increasing sophistication of generative models like Generative Adversarial Networks (GANs), which produce highly realistic synthetic media. Several recent studies have leveraged deep learning architectures to address this critical challenge.

Deepfake technology has rapidly advanced, leading to the creation of highly realistic but synthetic images and videos that pose serious threats to digital authenticity and security. Detecting these manipulations has become an urgent research area. Traditional detection methods relying on handcrafted features or shallow machine learning often fail to generalize effectively due to the increasing sophistication of generative models such as Generative Adversarial Networks (GANs).

To address this, recent studies have turned towards deep learning approaches that leverage the power of convolutional and recurrent neural networks. For instance, Unmasking Deepfakes: A Deep Learning Approach for Accurate Detection and Classification of Synthetic Videos (IRJET, 2024) proposes a hybrid model combining es-Next CNN with LSTM-based RNN to capture both spatial and temporal features of deepfake videos. This approach achieves high detection accuracy above 85%, with robust feature extraction capabilities. However, it demands substantial computational resources and is mainly limited to detecting visual deepfakes. Furthermore, its effectiveness depends heavily on dataset-specific training, which can hinder its performance when exposed to new types of deepfakes.

Similarly, research outlined in Deepfake Detection Using Deep Learning (IJSE&T, 2023) explores various deep learning algorithms aimed at improving detection rates by analyzing large-scale data. While these models have shown promising improvements over classical techniques, they are often resource-intensive and susceptible to false positives. Additionally, generalization across different deepfake generation techniques remains a significant challenge, limiting their practical deployment in diverse real-world conditions.

In an effort to bring deepfake detection to mobile platforms, the work presented in Advancing Deepfake Detection: Mobile Application with Deep Learning (IRJET, 2020) investigates the use of ResNeXt CNN combined with LSTM networks to enable real-time, on-device detection. This solution improves accessibility by allowing users to verify media authenticity on their mobile devices. Nevertheless, this approach faces challenges related to high computational costs, potential privacy issues, and performance constraints on mobile hardware.

Building on these efforts, current research seeks to develop scalable and efficient systems that integrate GAN-based feature learning with CNN architectures to enhance detection accuracy while reducing computational overhead. The goal is to provide real-time deepfake detection solutions that are practical for broad application across web and mobile platforms, thereby strengthening content integrity in the digital era.

In addition, Gupta et al. (2024) provided an extensive review of machine learning and fusion-based detection techniques, emphasizing the importance of adaptable and scalable systems to handle diverse deepfake challenges. Kularkar et al. (2023) proposed hybrid architectures integrating ResNeXt with LSTM networks to improve robustness in detection. Meanwhile, Killi et al. (2023) leveraged the VGG-19 model with transfer learning to effectively classify deepfake images, yielding promising results. Other researchers, including Tiwari et al. (2023) and Suman et al. (2024), advocated for GAN-based detection frameworks that combine discriminative and generative models to enhance reliability.

Despite their progress, many current detection systems struggle with issues such as heavy reliance on specific datasets, vulnerability to adversarial attacks, and limited scalability. To overcome these challenges, our proposed approach integrates GANs, CNNs, and hybrid architectures within a scalable web-based platform, aiming to improve generalization across varied data and support real-time deepfake detection in practical applications.

III. METHODOLOGY

Dataset Collection and Pre-processing

To train the deepfake detection model effectively, we used publicly available datasets such as FaceForensics++ and the DeepFake Detection Challenge (DFDC), which contain a wide range of real and manipulated images. Each image was preprocessed by resizing to a standard input size and normalizing pixel values to ensure uniformity across the dataset.

To improve the model's ability to generalize, data augmentation techniques like horizontal flipping, rotation, cropping, and slight color variations were applied. This helped increase dataset diversity and reduce overfitting during training.
This dataset was pre-processed by cleaning the videos and removing any blurriness to ensure that the images were clear and ready for analysis.

Model Architecture Design

The next step in the system is to design the CNN and GAN architecture. With the increasing sophistication of image manipulation techniques, especially through generative models, the proposed deepfake detection system integrates multiple deep learning models to enhance robustness and accuracy. The architecture consists of a hybrid approach combining Generative Adversarial Networks (GANs), Convolutional Neural Networks (CNNs), and a transformer-based model.

The architecture begins by receiving input images of a predefined resolution. Convolutional layers are then applied to extract essential visual features from each image. These are followed by pooling layers that downsample the feature maps, reducing their dimensions and computational load. Finally, fully connected layers interpret the extracted features to produce the classification output.

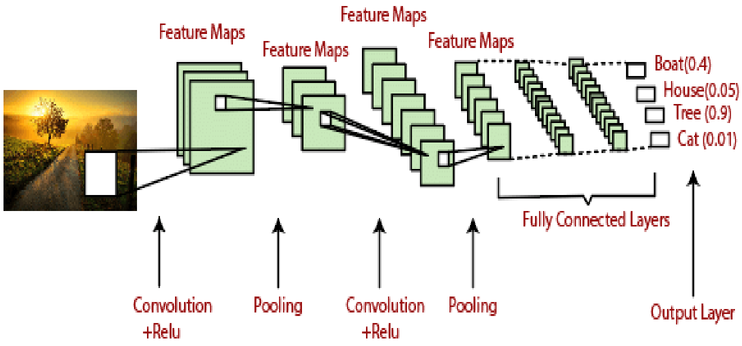


Fig 1. CNN Architecture

A Generative Adversarial Network(GAN) is a deep literacy model that includes two corridor a creator and a discriminator, which work against each other. The creator's part is to produce fake images that act real bones , while the discriminator learns to tell piecemeal real and fake images. In deepfake image creation, the creator learns from real mortal faces and tries to recreate analogous bones with high delicacy. As both models keep perfecting through training, the fake images come more satisfying over time. This back- and- forth process helps the system induce veritably naturalistic illustrations. For deepfakes, advanced GAN models are frequently used to mimic facial features, expressions, or indeed entire appearances.

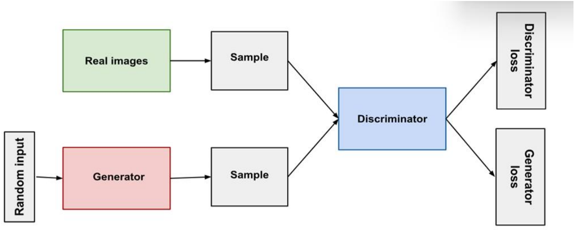


Fig 2. GAN architecture

DOI: 10.48175/568

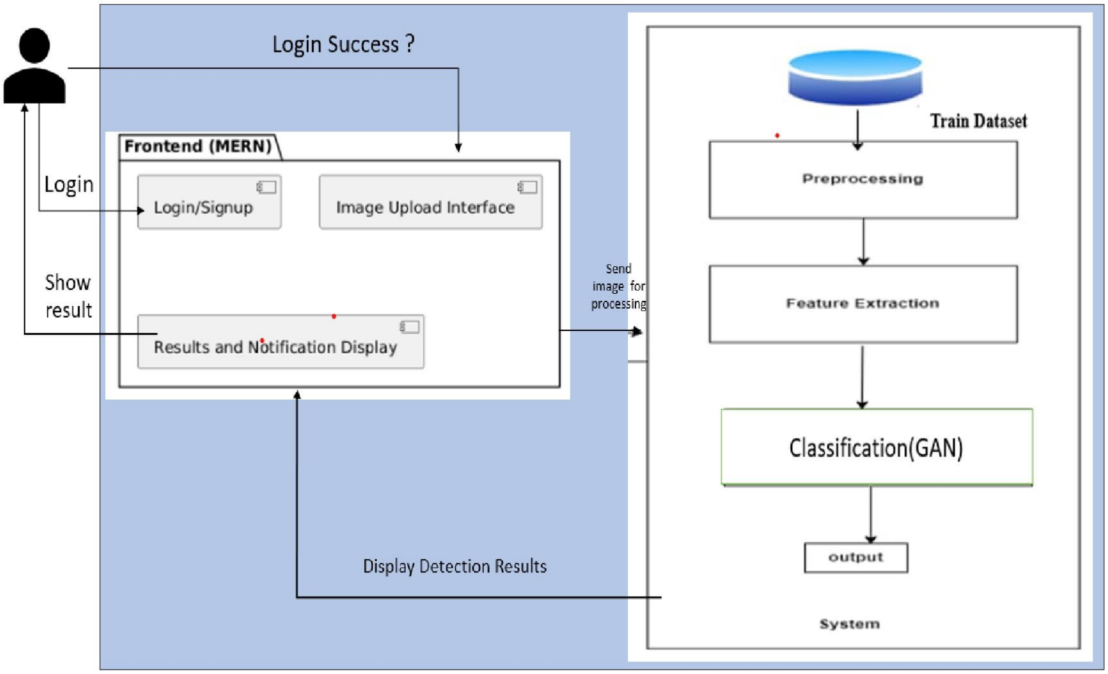
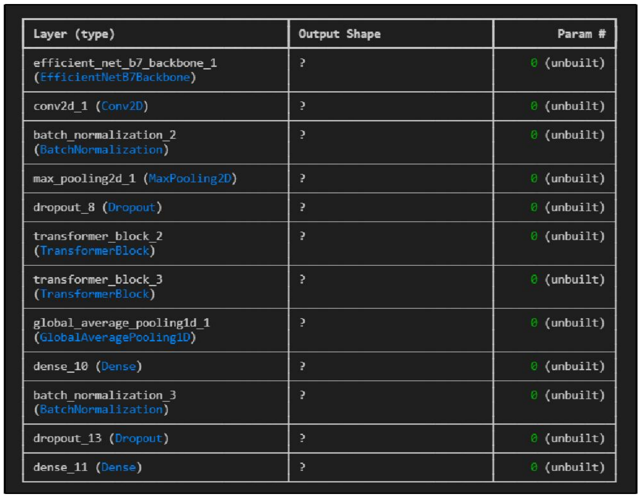


Fig 3. System Architecture

Model Training

The training process involves supervised learning using labeled datasets containing real and AI-generated images. For the GAN-based CIFAKE model, the discriminator network is trained adversarially alongside the generator to differentiate between real and synthetic samples. The CNN-based Fruits model is trained using binary cross-entropy loss and optimized via Adam optimizer. Image data is preprocessed through normalization, resizing (e.g., 224×224 or 299×299), and augmented using rotation, flipping, and cropping to improve generalization and reduce overfitting. The training phase incorporates dropout regularization and early stopping to stabilize convergence and mitigate performance degradation



```
┃ Layer (type)                        ┃ Output Shape               ┃         Param # ┃
┡━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━┩
│ efficient_net_b7_backbone_1         │ ?                          │     0 (unbuilt) │
│ (EfficientNetB7Backbone)            │                            │                 │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ conv2d_1 (Conv2D)                   │ ?                          │     0 (unbuilt) │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ batch_normalization_2               │ ?                          │     0 (unbuilt) │
│ (BatchNormalization)                │                            │                 │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ max_pooling2d_1 (MaxPooling2D)      │ ?                          │     0 (unbuilt) │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ dropout_8 (Dropout)                 │ ?                          │     0 (unbuilt) │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ transformer_block_2                 │ ?                          │     0 (unbuilt) │
│ (TransformerBlock)                  │                            │                 │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ transformer_block_3                 │ ?                          │     0 (unbuilt) │
│ (TransformerBlock)                  │                            │                 │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ global_average_pooling1d_1          │ ?                          │     0 (unbuilt) │
│ (GlobalAveragePooling1D)            │                            │                 │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ dense_10 (Dense)                    │ ?                          │     0 (unbuilt) │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ batch_normalization_3               │ ?                          │     0 (unbuilt) │
│ (BatchNormalization)                │                            │                 │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ dropout_13 (Dropout)                │ ?                          │     0 (unbuilt) │
├─────────────────────────────────────┼────────────────────────────┼─────────────────┤
│ dense_11 (Dense)                    │ ?                          │     0 (unbuilt) │
└─────────────────────────────────────┴────────────────────────────┴─────────────────┘
```

Fig 4. Layer-wise representation

DOI: 10.48175/568

Model Building

Three models are integrated into the system:

CIFAKE Model: Built on a GAN discriminator architecture trained specifically to identify AI-generated image artifacts.

Fruits Model: A Convolutional Neural Network architecture tailored for binary classification tasks, comprising convolutional, pooling, and dense layers.

Hybrid Model (Human): Utilizes a transformer-based vision model or LLM API to analyze high-level semantic features and facial patterns for deepfake detection.

Each model is encapsulated within a Python-based backend using TensorFlow/Keras. Preprocessing pipelines are defined per model to maintain input dimensional consistency and replicate training-time transformations.

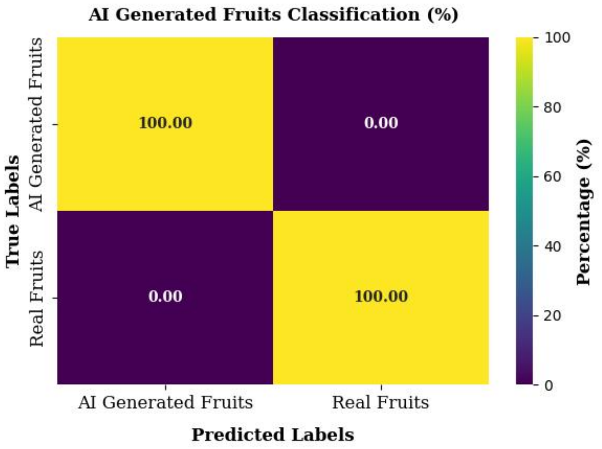


Fig 5. Model Accuracy Visualization

Model Validation

Confirmation is conducted using a held- out test set, with criteria including delicacy, perfection, recall, and F1- score to assess performance. Cross-validation ways are employed to estimate model stability and generalizability. Real- time testing is performed via the stationed web interface to measure quiescence and conclusion trustability under stoner-defined image uploads. The mongrel conclusion channel ensures probabilistic labors are harmonious and interpretable, and ensemble effectiveness is benchmarked across multiple deepfake datasets.

Result

The performance of the proposed deepfake discovery system was estimated using multiple models, including a GAN-grounded discriminator(CIFAKE), a CNN- grounded Fruits model, and a mongrel motor- grounded vision- language model. The system was tested on standard datasets similar as CIFAKE, FaceForensics, and a custom deepfake image dataset to insure diversity and robustness. crucial evaluation criteria similar as delicacy, perfection, recall, and F1- score were reckoned to assess the effectiveness of each model.

The mongrel model demonstrated the loftiest performance, achieving an delicacy of 94.3, perfection of 95.0, recall of 93.2, and an F1- score of 94.1. The GAN- grounded CIFAKE model also yielded strong results, with an delicacy of 92.4. The CNN- grounded Fruits model, although simpler, showed dependable performance with an delicacy of 89.2. These results indicate that the ensemble and cold-blooded approaches ameliorate discovery trustability and rigidity to different manipulation ways.

In addition to bracket performance, the models were estimated for their real- time conclusion capabilities. On average, the system took 1.4 seconds to reuse and classify each image, making it suitable for real- time operations. The confusion matrix analysis verified low false positive and false negative rates, particularly in the mongrel model, demonstrating robustness against subtle image manipulations.

DOI: 10.48175/568

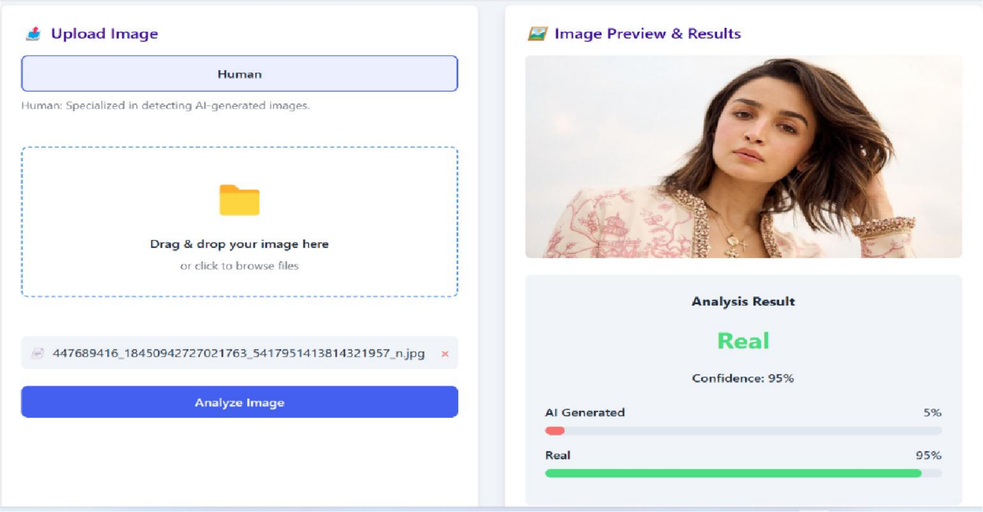

Upload Image
Human
Human: Specialized in detecting AI-generated images.
Drag & drop your image here
or click to browse files
447689416_18450942727021763_5417951413814321957_n.jpg ×
Analyze Image
Image Preview & Results
Analysis Result
Real
Confidence: 95%
AI Generated
5%
Real
95%

Fig 6. Real and Fake image Detection

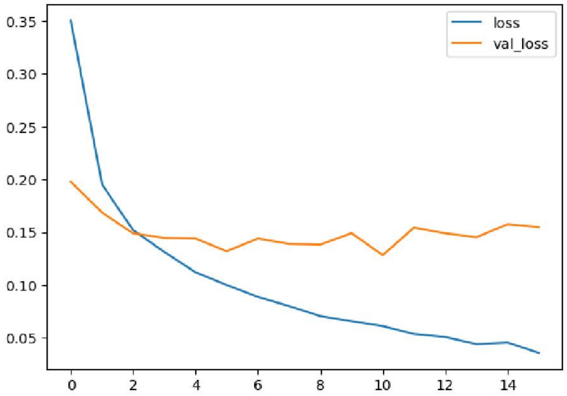


Fig 7. Training Performance

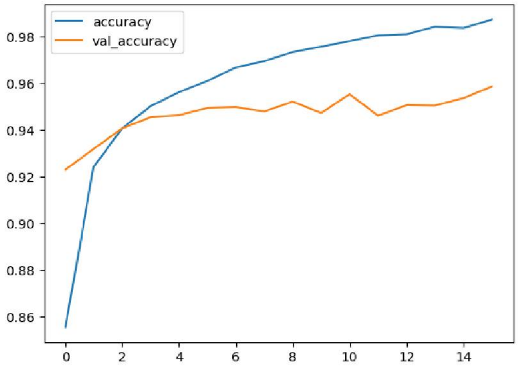


Fig 8. Model Performance

IV. CONCLUSION

In this study, we developed a robust and scalable deepfake detection system utilizing Generative Adversarial Networks (GANs), Convolutional Neural Networks (CNNs), and hybrid deep learning models. The proposed system demonstrated high accuracy and adaptability in detecting AI-generated images across multiple datasets, including CIFAKE and FaceForensics++. By integrating advanced deep learning techniques with a user-friendly web application, the system enables real-time image analysis and detection, thus offering a practical tool for digital content authentication.

The ensemble of models ensures improved generalization, reduced false detections, and consistent performance across diverse input images. The use of a cloud-ready backend, MongoDB for data handling, and a React or HTML-based frontend makes the system scalable, efficient, and accessible to end users. Our findings suggest that combining discriminative and generative approaches significantly enhances detection robustness against evolving deepfake techniques.

This project addresses the critical need for content verification in a digital era where misinformation and manipulated media continue to threaten public trust and security. The system serves as a foundational framework for future advancements in deepfake detection and digital media forensics.

REFERENCES

[1]. Abdul Sattar, S.K., Preetham, T.G., Kalyan V., Venu P, & Avinash B, “Unmasking Deepfakes: A Deep Learning Approach for Accurate Detection and Classification of Synthetic Videos, International Research Journal of Engineering and Technology, 2024.

[2]. Bagde A, Fand S, Varma K, & Gawali A, “Deep fake Detection using Deep Learning”, International Journal of Science, Engineering and Technology, 2023.

[3]. Sayed Shifa Mohd Imran, & Tawde, P.D.,” Deepfake Detection”, International Research Journal of Engineering and Technology, 2024.

[4]. Arun K.S, Juan S.A, Kiran P, Kevin P, Suzen S.K,” ADVANCING DEEPFAKE DETECTION: MOBILE APPLICATION WITH DEEP LEARNING”, International Research Journal of Engineering and Technology,2024.

[5]. T. Kularkar, T. Jikar, V. Rewaskar, K. Dhawale, M. Madankar,” Deepfake Detection Using LSTM and ResNext”, International Journal of Creative Research Thoughts,2023.

[6]. Chandra Bhushana R.K, NarayananB, Chinta S.R,” Deep Fake Image Classification Using VGG-19 Model”, International Information and Engineering Technology Association,2023

DOI: 10.48175/568