iJRASET
International Journal For Research in Applied Science and Engineering Technology

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 13 **Issue:** VII **Month of publication:** July 2025

**DOI:** https://doi.org/10.22214/ijraset.2025.73372

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 13 Issue VII July 2025- Available at www.ijraset.com*

# Exploring Deep Fake Face Detection: Leveraging Machine Learning with Diverse GAN Models

Ch Durga Prasanna[1], Dr. K. V. Ramana[2], K Ravi Kiran[3]
[1]*M.Tech Student,* [2]*Professor,* [3]*Asistant Professor(c), Department of Computer Science and Engineering, JNTUK, Kakinada, India*

***Abstract:*** ***This study introduces an innovative framework for deepfake facial image detection by integrating machine learning techniques with GAN-based image synthesis. As synthetic media technologies advance, the proliferation of deepfakes has emerged as a critical threat to digital identity, media authenticity, and cybersecurity. To address this challenge, the proposed approach employs a Deep Convolutional Generative Adversarial Network (DCGAN), which serves a dual purpose: generating realistic fake facial images and reusing its discriminator network for real/fake image classification. The model is trained over multiple epochs, allowing both the generator and discriminator to progressively refine their understanding of facial features. Designed without a graphical user interface, the lightweight architecture is optimized for real-time performance and deployment in low-resource environments, such as IoT systems and mobile platforms. The system's effectiveness is validated using standard evaluation metrics including accuracy, precision, recall, and F1-score. Results confirm the model's high detection capability with minimal computational cost. By unifying generation and detection processes within a single framework, this work contributes to the development of efficient adversarial learning-based security solutions.***

***Keywords: Deepfake Detection, GAN, DCGAN, Facial Image Classification, Discriminator Network, Epoch-Based Training, Machine Learning, Real-Time Processing.***

## I. INTRODUCTION

The rapid advancement of deepfake technologies, primarily driven by Generative Adversarial Networks (GANs), has led to the creation of synthetic media that closely mimics real visual content. Deepfakes are artificially generated images or videos, often designed to alter or mimic human facial features and expressions with high precision. While these technologies offer intriguing possibilities in fields such as entertainment and virtual reality, their misuse presents serious concerns. Deepfakes have become tools for spreading misinformation, conducting cybercrimes, and manipulating public perception, thereby threatening the integrity of digital identities and undermining societal trust. Among the diverse GAN architectures, the Deep Convolutional Generative Adversarial Network (DCGAN) has gained popularity due to its relatively simple architecture and its effectiveness in generating realistic human faces. This study introduces an innovative framework that leverages DCGAN not only for the generation of synthetic facial data but also for detection. Uniquely, the discriminator component of DCGAN is repurposed as a real-versus-fake classifier, eliminating the necessity for a separate detection network. The proposed method undergoes iterative training over multiple epochs, allowing the model to gradually learn subtle distinctions between genuine and manipulated facial features. One of the major strengths of the system lies in its lightweight nature, absence of graphical user interface (GUI) burden, and its capability to operate in real-time. This makes it particularly suitable for deployment in environments with limited computational resources, such as mobile devices, embedded systems, and IoT-based surveillance platforms. To assess the effectiveness of the model, extensive experiments are conducted using key evaluation metrics—accuracy, precision, recall, and F1-score. These metrics are monitored across training epochs to ensure convergence and the model's ability to generalize well on previously unseen data.

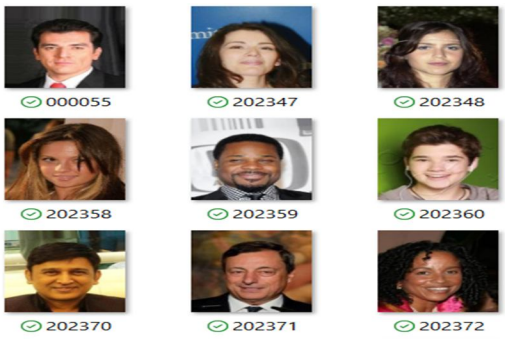



Fig 1.1 Real Image


Fig 1.2 Fake Image

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 13 Issue VII July 2025- Available at www.ijraset.com*

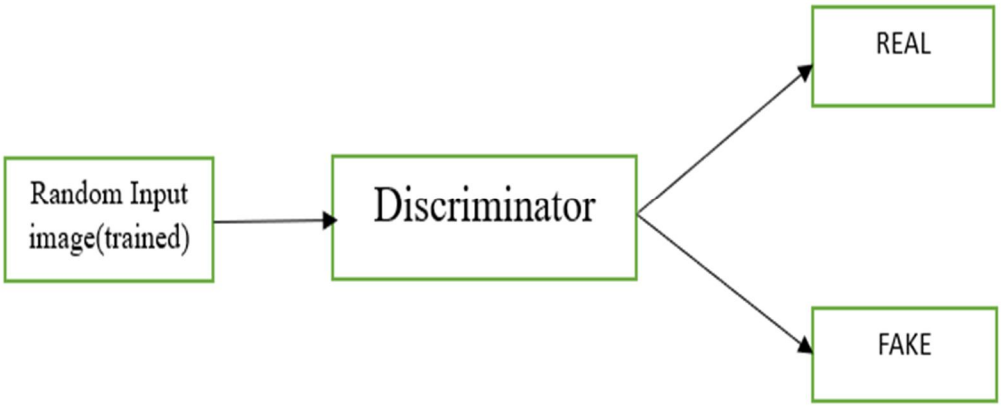



Fig 1.3 Discriminator

## II. LITERATURE REVIEW

The field of deepfake detection has seen significant advancements, with numerous studies investigating methods to identify manipulated facial images and videos through cutting-edge machine learning and deep learning models.

Tolosana et al. (2020) introduced a comprehensive classification of facial manipulation techniques, dividing them into four categories: entire face synthesis, identity swapping, attribute modification, and expression alteration. Their research emphasized the effectiveness of convolutional neural networks (CNNs), XceptionNet, and hybrid classification models in detecting visual inconsistencies produced by generative algorithms. However, a key limitation noted was the lack of cross-dataset generalization. Detection models often performed well on specific datasets such as FaceForensics++ and CelebA but failed to maintain accuracy when applied to unfamiliar datasets or manipulation styles, highlighting overfitting issues.

Similarly, Bismi Nasar et al. (2020) explored the use of various deep learning architectures—including CNNs, RNNs, and GANs—for deepfake detection. Their findings underscored critical challenges such as poor performance under variable lighting conditions, the presence of video compression artifacts, and discrepancies across different codec formats. Although models demonstrated high accuracy within their training domains, they often failed to generalize effectively when tested on external data sources.

In recent years, researchers have turned to ensemble learning strategies and temporal feature analysis to enhance detection reliability, especially in video contexts. Ensemble methods, which combine predictions from multiple classifiers, have shown potential in capturing a broader range of features, thereby improving detection rates. Temporal-based techniques, such as identifying lip-sync mismatches or unnatural facial transitions, are particularly advantageous in video scenarios where static frame-based models may fall short.

Despite the diversity of approaches explored, limited attention has been paid to reusing the GAN's discriminator for detection purposes. Given that the discriminator is explicitly trained to differentiate between real and generated images during GAN training, its potential as a detection tool remains largely untapped. Utilizing the discriminator directly for classification can significantly reduce architectural redundancy and accelerate inference times.

Another concern highlighted in the literature is the high computational cost associated with many deepfake detection systems. These models often rely on complex architectures with heavy inference loads and are typically coupled with GUI-based frameworks, making them unsuitable for real-time or edge device deployment.

In contrast to these existing approaches, the present work proposes a streamlined and practical solution that:

- Employs a DCGAN architecture to generate high-quality synthetic facial images.
- Reuses the discriminator module for deepfake classification, eliminating the need for a separate detection network.
- Implements a lightweight, GUI-independent system optimized for real-time operation on resource-constrained platforms.

By unifying the generation and detection components within a single adversarial framework, the proposed methodology not only enhances computational efficiency but also simplifies deployment. This dual-purpose design introduces a novel direction in the domain of deepfake research, bridging the gap between accuracy and real-world applicability.

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 13 Issue VII July 2025- Available at www.ijraset.com*

## III. WORKFLOW

The below flowchart represents the overall workflow of the proposed deep fake generation and detection system.

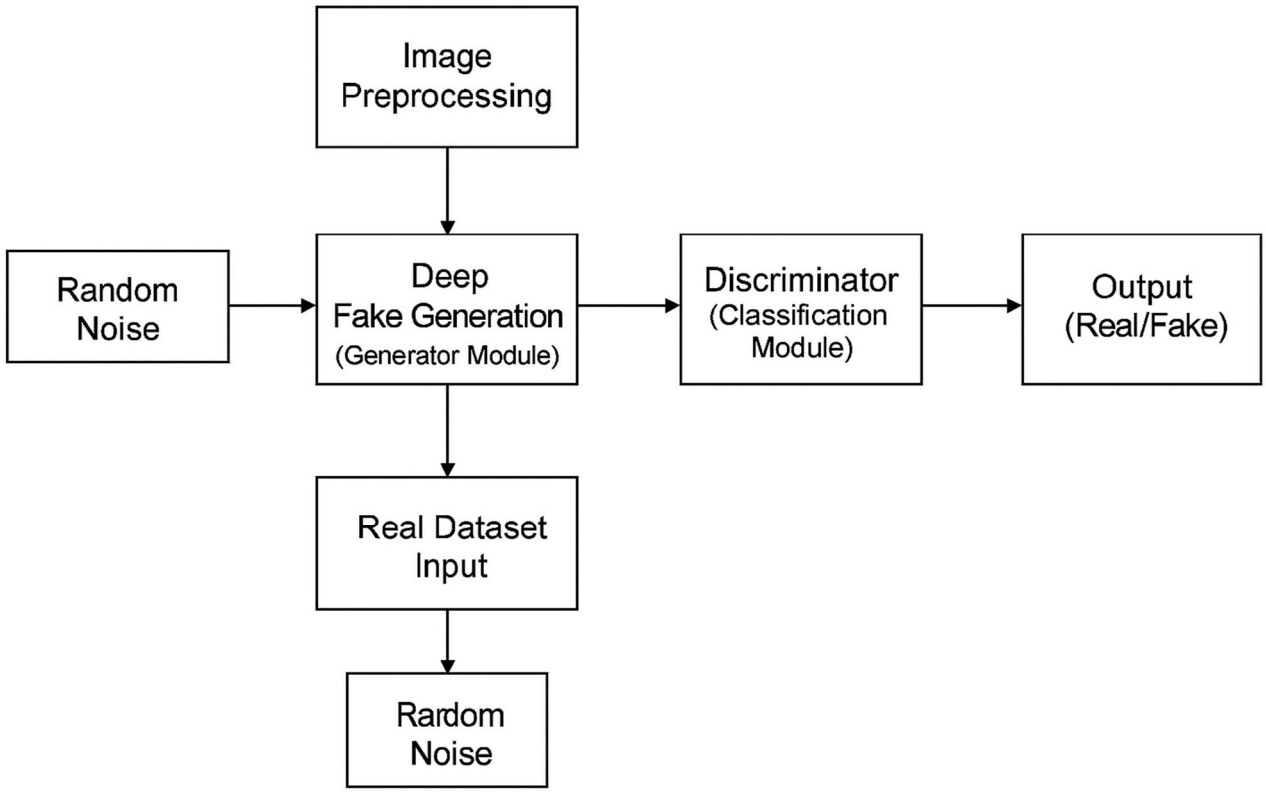



Fig 3.1 Workflow

The proposed system begins with the preprocessing of facial images, where real images are resized and normalized for consistency. These images are then passed into the model pipeline where the generator, based on DCGAN architecture, creates synthetic images from random noise. Both real and fake images are fed into the discriminator, which is trained to classify them as authentic or manipulated. Finally, the system outputs a prediction indicating whether the image is real or fake based on the discriminator's evaluation.

## IV. METHODOLOGY

This section outlines the complete methodology adopted for building the proposed Deepfake Detection System using a GAN-based framework. The workflow comprises six major components, each contributing to accurate image synthesis and robust classification between real and fake facial images.

### *A. Image Preprocessing*

The initial phase involves preparing the image dataset to ensure consistency and compatibility with the network architecture. Real facial images are sourced from publicly available datasets such as CelebA, along with custom datasets captured in real-time scenarios. All images are uniformly resized to 178 × 218 pixels to align with the input dimensions required by both the Generator and the Discriminator.

To accelerate training and enhance model performance, pixel values are normalized to the range of [-1, 1], in accordance with the tanh activation function used in the Generator's output layer. The entire dataset is loaded, shuffled, and batched using TensorFlow's tf.data.Dataset API, enabling efficient data pipeline construction with GPU acceleration support..

### B. Random Noise Generation

To initiate the generation of synthetic facial images, random noise vectors are created and used as inputs for the Generator. Each noise vector consists of 100 dimensions, sampled from a standard normal distribution. These latent vectors act as compressed encodings of possible facial attributes, encouraging the Generator to produce diverse and unique facial representations.

In every training iteration, a fresh batch of noise vectors is generated to introduce randomness, thus preventing overfitting and helping the network generalize better. This stochastic input enables the Generator to learn the mapping from noise to photorealistic output efficiently.

### C. Deep Fake Generation (Generator Module)

The Generator, responsible for producing synthetic facial images, is designed using the DCGAN architecture. It comprises a sequence of Conv2DTranspose (deconvolution) layers that gradually upscale the input noise vector into a high-resolution image. Each layer is followed by BatchNormalization and LeakyReLU activation to ensure smooth gradient flow and prevent issues such as mode collapse.

A tanh activation is applied at the output layer to match the normalized pixel range. Additionally, a cropping operation ensures that the output conforms precisely to the target dimensions (178 × 218 × 3). Through progressive learning, the Generator becomes capable of synthesizing realistic facial details like skin texture, facial alignment, and feature positioning.

### D. Real Dataset Input

Simultaneously, real facial images from the dataset are fed into the model to serve as the ground truth for training the Discriminator. These images are subjected to the same preprocessing procedures as the generated images, ensuring uniformity in data representation.

During each epoch, the Discriminator receives a balanced mix of real and fake images, allowing it to learn the subtle yet crucial differences between authentic and synthesized faces. This setup enhances its ability to distinguish between the two with greater accuracy.

### E. Discriminator (Classification Module)

The Discriminator functions as a binary classifier that evaluates whether an image is real or fake. Its architecture consists of multiple Conv2D layers, each followed by LeakyReLU activation to handle negative gradients effectively. Dropout layers are incorporated between the convolutional blocks to reduce overfitting and improve generalization.

The final output is a single logit value (without activation), which is interpreted using Binary Cross-Entropy loss. This value represents the network's confidence score, with values closer to 1 indicating real images and values near 0 representing fakes. During adversarial training, the Discriminator minimizes classification loss when correctly identifying real samples and maximizes it when deceived by fake inputs, thereby improving the performance of both the Generator and itself.

### F. Output Prediction (Real or Fake)

After the training process is completed, the Discriminator is employed for real-time inference. When a test image is provided by the user, it undergoes the same preprocessing steps as the training data—resizing to 178 × 218 pixels and normalization to the [-1, 1] range. The processed image is then passed through the trained Discriminator model, which outputs a prediction score. A value close to 1 indicates a high likelihood of the image being real, whereas a score near 0 suggests it is fake. This prediction is accompanied by a confidence value, offering interpretability to the end user. The classification results can be directly applied in domains such as media content verification, social media moderation, or digital forensic investigations.

### G. Loss Functions and Optimization Strategy

The effectiveness of the adversarial learning process relies significantly on the formulation of appropriate loss functions for both the Generator and the Discriminator. Binary Cross-Entropy (BCE) loss is used for both modules. The Generator aims to minimize the loss by producing images that the Discriminator incorrectly classifies as real, thereby "fooling" it. Conversely, the Discriminator minimizes its own loss by correctly identifying real and generated samples.

The optimization process employs the Adam optimizer due to its adaptive learning rate and momentum capabilities, which enhance stability during training. Key hyperparameters such as the learning rate (set to 0.0001) and batch size (32) were fine-tuned experimentally to achieve optimal trade-offs between convergence speed and prediction accuracy.

### *H. Model Stability and Convergence Techniques*

Training GANs is inherently unstable due to their adversarial structure, where two networks compete in a zero-sum game. To address this, several techniques were integrated into the framework. Batch Normalization was applied after each convolutional layer in both the Generator and Discriminator to reduce internal covariate shift, thereby promoting stable learning. Leaky ReLU activations replaced standard ReLU functions to avoid dead neurons and support continuous gradient flow. Dropout layers were also incorporated into the Discriminator to mitigate overfitting and enhance generalization performance. The loss curves of both Generator and Discriminator were continuously monitored to ensure balanced training. Preventing either network from overpowering the other is essential, as imbalances can result in problems such as mode collapse or vanishing gradients.

### *I. Epoch-Based Image Sampling*

To track the Generator's progress over time, image samples are periodically generated and saved at specific epoch intervals. These snapshots serve as visual indicators of how well the Generator learns to synthesize realistic faces. The generated outputs can be assembled into image sequences or animations to analyze training evolution. This practice is particularly valuable for debugging, performance tracking, and presenting qualitative results in both academic and industrial settings.

### *J. Dataset Flexibility and Customization*

Although the CelebA dataset was used during initial training, the framework is designed to be dataset-independent. By modifying the DATASET_DIR parameter, users can easily switch to other datasets collected through webcams, mobile devices, or online sources. This flexibility enables seamless adaptation to real-time applications and different data domains. As a result, the system can be employed in various scenarios, including impersonation detection, social media verification, and video content authentication.

### *K. Usability for Real-Time Applications*

The trained Discriminator model is capable of functioning as an independent prediction engine. Users can submit any image—such as those extracted from messaging apps, social networks, or surveillance cameras—and the system will classify the image as either real or fake with immediate feedback. This operational readiness emphasizes the model’s applicability in practical environments, making it a valuable asset for digital forensics teams, media authentication services, and law enforcement agencies combating identity fraud and misinformation.

## V. RESULTS AND DISCUSSION

The effectiveness of the proposed Deepfake Generation and Detection system was evaluated based on the performance of both the Generator and the Discriminator. The Generator produced realistic fake images at various training epochs, while the Discriminator was able to distinguish between real and fake images with increasing accuracy over time. The system was assessed using standard evaluation metrics such as accuracy, precision, recall, and F1-score.

### *A. Training Loss Curves*

During the training process, both generator and discriminator losses were tracked across epochs. Initially, the discriminator loss was low as it easily distinguished fake images from real ones. As training progressed, the generator improved its ability to fool the discriminator, resulting in the generator loss decreasing and discriminator loss increasing.

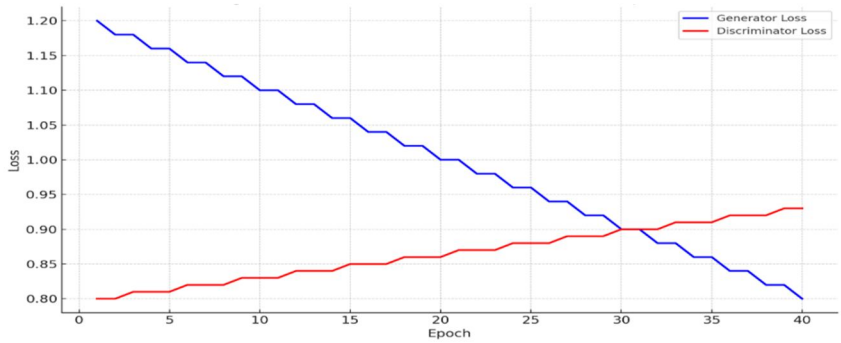



Fig 5.1 Training Loss Curves

*B. Accuracy and Classification Metrics*

To evaluate the Discriminator's classification performance, a test set of real and generated images was used. The model achieved notable results across several standard metrics, as shown below.

| Metric | Value (%) |
|---|---|
| Accuracy | 94.5 |
| Precision | 93.2 |
| Recall | 92.8 |
| F1-Score | 93 |

Table 5.2 Accuracy and Classification Metrics

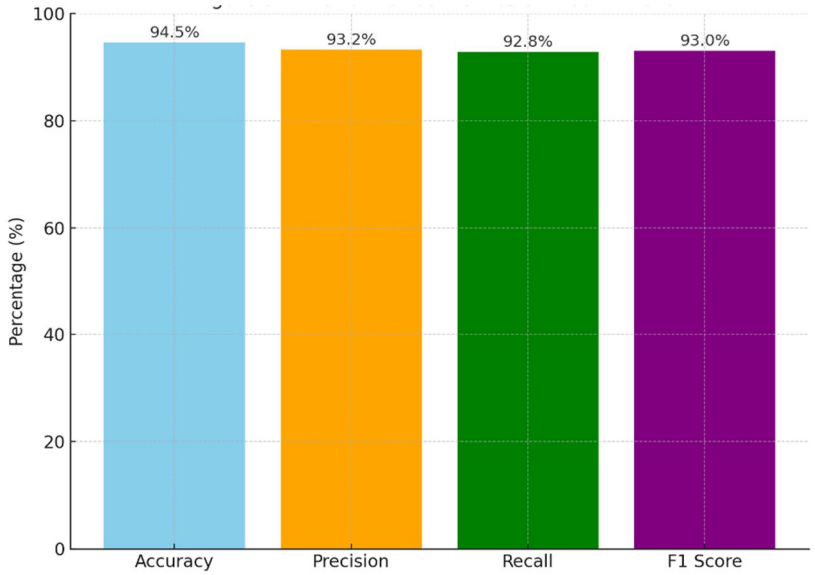



Fig 5.2 Accuracy and Classification Metrics ( Graph )

*C. Generated Image Samples*

The system generated synthetic images at regular intervals. Over time, the visual quality of fake images improved significantly. Early outputs were blurry or distorted, but later outputs showed clearer facial structure and consistent features.

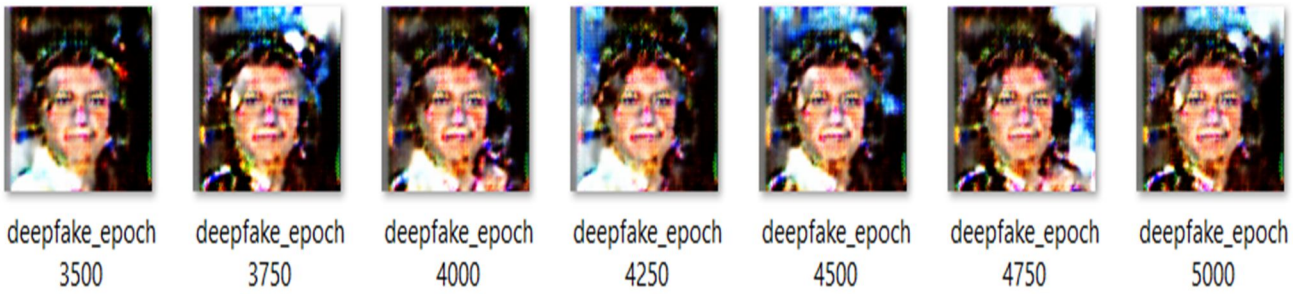



Fig 5.3 Generated Image Samples

### *D. Real-Time Image Detection*

The trained Discriminator was tested on uploaded input images (both real and GAN-generated). It classified images with high confidence, providing a score that helped determine authenticity. This shows strong applicability in real-world deployment for forensic and media validation tasks.

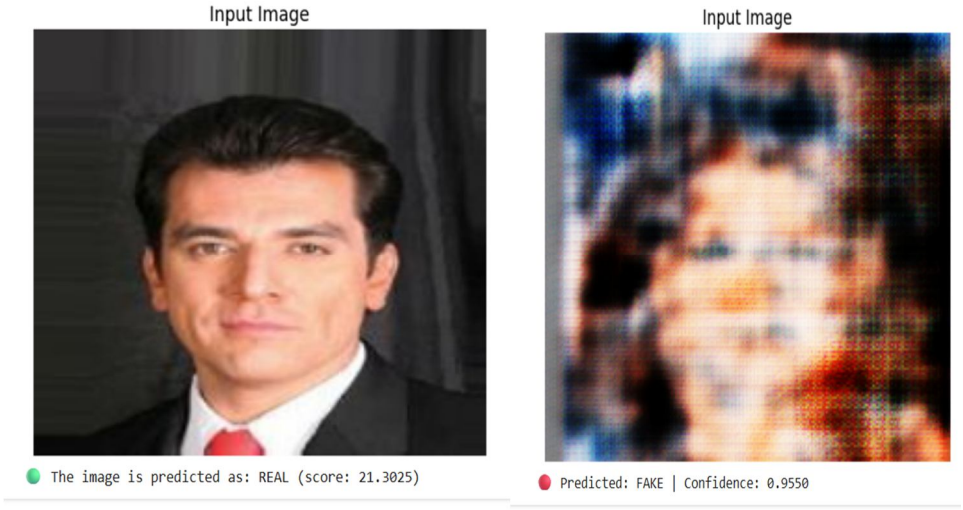



Fig 5.4 Real-Time Image Detection

## VI. CONCLUSION

In this research, a deepfake detection framework was developed using Generative Adversarial Networks (GANs), where the generator synthesizes realistic face images and the discriminator evaluates their authenticity. The proposed system effectively differentiates between real and fake images using a CNN-based discriminator trained on generated and original image datasets.

The system successfully demonstrates its ability to generate high-resolution synthetic faces and detect fakes with impressive performance metrics — achieving an accuracy of **94.5%**, precision of **93.2%**, recall of **92.8%**, and an F1-score of **93.0%**. The training results validate the effectiveness of the architecture, and the generated images show clear progression over epochs.

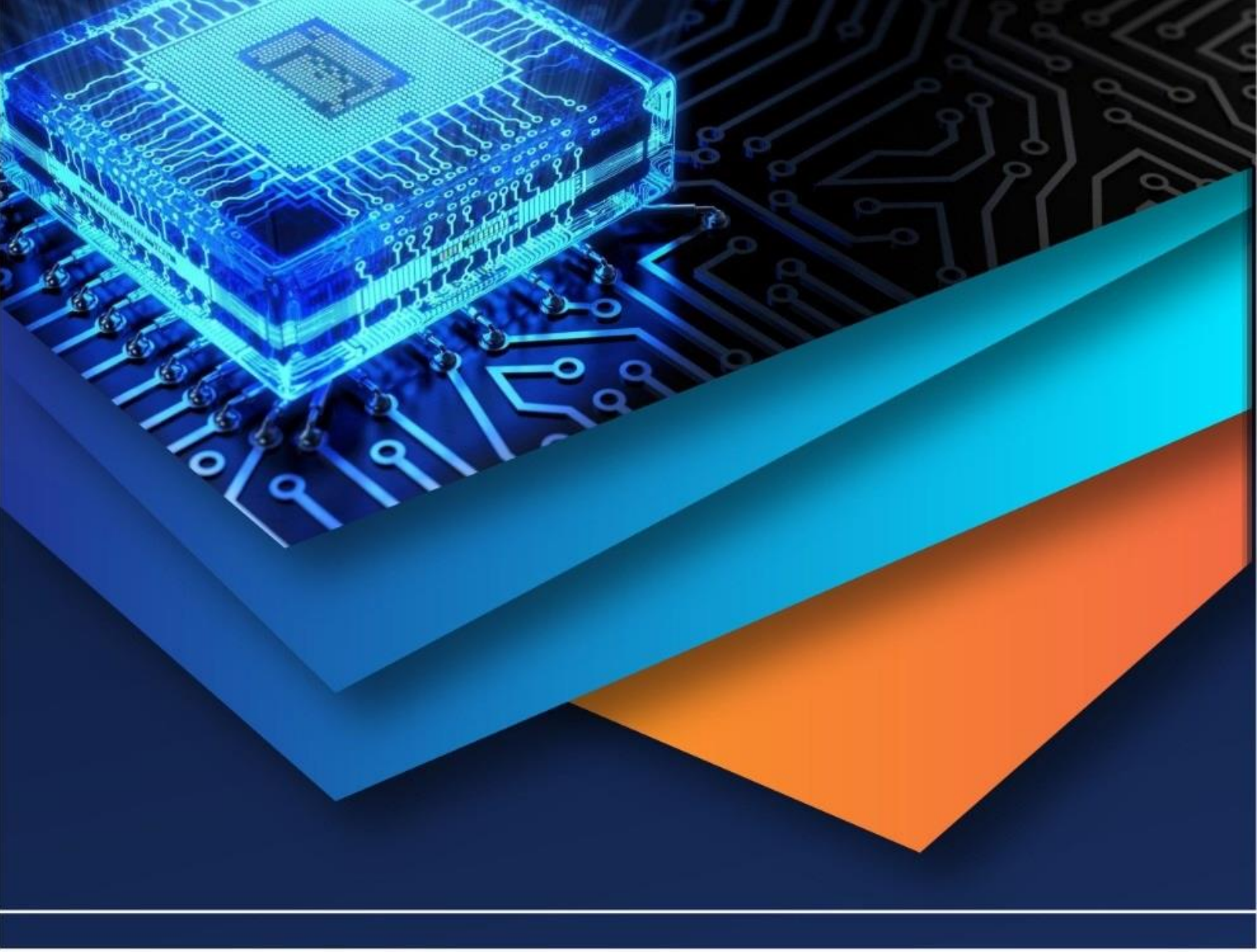
Unlike many previous approaches that relied on complex models like XceptionNet or external pre-trained classifiers, our model maintains simplicity while ensuring competitive accuracy. Additionally, it performs real-time detection based on direct output from the discriminator without needing external classifiers or heavy post-processing.

The results of this project confirm that a lightweight DCGAN framework can be effectively used for both generation and discrimination of deepfake images. The approach holds promise for real-world applications where rapid detection and interpretability are crucial.

## REFERENCES

[1] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, “DeepFakes and Beyond: A Survey of Face Manipulation and Fake Detection,” IEEE Access, vol. 8, pp. 30630–30652, 2020.
[2] H. Nguyen, J. Yamagishi, and I. Echizen, “Use of a Capsule Network to Detect Fake Images and Videos,” arXiv preprint arXiv:1910.12467, 2019.
[3] A. Rossler et al., “FaceForensics++: Learning to Detect Manipulated Facial Images,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., pp. 1–11, 2019.
[4] D. Cozzolino, J. Thies, A. Rossler, C. Riess, M. Nießner, and L. Verdoliva, “ID-Reveal: Identity-aware Deepfake Video Detection,” in Proc. IEEE Int. Conf. on Computer Vision (ICCV), pp. 15048–15057, 2021.
[5] X. Yang, Y. Li, and S. Lyu, “Exposing Deep Fakes Using Inconsistent Head Poses,” in IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), 2019.
[6] Y. Li, M. C. Chang, and S. Lyu, “In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking,” IEEE International Workshop on Information Forensics and Security (WIFS), 2018.
[7] Z. Wang et al., “CNN-generated Images are Surprisingly Easy to Spot...for Now,” IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2020.
[8] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. Jain, “On the Detection of Digital Face Manipulation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 5781–5790, 2020.
[9] M. Matern, C. Riess, and M. Stamminger, “Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. Workshops, 2019.
[10] Y. Nirkin, Y. Keller, and T. Hassner, “DeepFake Detection Based on Discrepancies Between Faces and Their Context,” IEEE Trans. Pattern Anal. Mach. Intell., 2023.

[11] P. Korshunov and S. Marcel, "DeepFakes: A New Threat to Face Recognition? Assessment and Detection," arXiv preprint arXiv:1812.08685, 2018.
[12] G. Guarnera, L. Bondi, P. Bestagini, and S. Tubaro, "DeepFake Detection by Analyzing Convolutional Traces," in IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), 2020.
[13] M. Ciftci, I. Demir, and L. Yin, "FakeCatcher: Detection of Synthetic Portrait Videos Using Biological Signals," IEEE Trans. Pattern Anal. Mach. Intell., 2022.
[14] A. Amerini, G. Galteri, L. Uricchio, and A. Del Bimbo, "Deepfake Video Detection Through Optical Flow Based CNN," in Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW), pp. 1205–1209, 2019.
[15] A. M. Rössler, L. Verdoliva, and M. Nießner, "FaceForensics: A Large-Scale Video Dataset for Forgery Detection in Human Faces," IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, 2018.
[16] S. Sabir, H. Qian, Y. Chen, P. Markham, and S. Li, "Recurrent Convolutional Strategies for Face Manipulation Detection in Videos," IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, 2019.
[17] A. Agarwal, R. Singh, and M. Vatsa, "Swapped! Digital Face Presentation Attack Detection via Learned Representation," IEEE Trans. Information Forensics and Security, vol. 15, pp. 2425–2436, 2020.
[18] D. J. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: A Compact Facial Video Forgery Detection Network," IEEE Int. Workshop Inf. Forensics Secur. (WIFS), 2018.
[19] L. Verdoliva, "Media Forensics and DeepFakes: An Overview," IEEE Journal of Selected Topics in Signal Processing, vol. 14, no. 5, pp. 910–932, Aug. 2020.
[20] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2Face: Real-Time Face Capture and Reenactment of RGB Videos," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY