International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 13 Issue: 04 | Apr 2026 www.irjet.net p-ISSN: 2395-0072

Deepfake Detection Using Convolutional Neural Networks (CNN)

Vijay Chakole[1], Akshita Lanjewar[2], Astha Jadhao[3], Pallavi Chikate[4], Mayuri Sawalakhe[5]

[1]Professor, Dept. of Electronics and Telecommunication Engineering, K.D.K. college of eng.,Nagpur, India
[234]Student, Dept. of Electronics and Telecommunication Engineering, K.D.K. college of eng.,Nagpur, India

---

Abstract -*Deepfake technology has rapidly evolved with the advancement of artificial intelligence, enabling the creation of highly realistic synthetic images and videos. While this technology has useful applications, it also raises serious concerns related to misinformation, identity misuse, and digital security. Detecting such manipulated content has become increasingly challenging due to the complexity of modern generation techniques.*

This paper presents a deep learning-based approach for detecting deepfake images using multiple Convolutional Neural Network (CNN) architectures. Models including GoogLeNet, InceptionV3, VGG16, DenseNet121, and Xception were implemented using transfer learning. The dataset was preprocessed, augmented, and divided into training, validation, and testing sets to ensure reliable evaluation.

The models were compared based on accuracy and performance metrics, where VGG16 achieved the highest accuracy among all models. The final system was deployed using a Flask-based web interface, allowing users to upload images and obtain real-time predictions. The results demonstrate the effectiveness of CNN-based approaches for deepfake detection and their potential for real-world applications.

Key Words: Synthetic media analysis, Artificial intelligence in forensics, Convolutional neural networks (CNNs) ,Transfer learning techniques, Image classification models, Digital content authenticity ,Deep learning architectures ,Data augmentation methods .

1.INTRODUCTION

In recent years, the rapid advancement of artificial intelligence and deep learning has led to the emergence of deepfake technology, which enables the creation of highly realistic manipulated images and videos. These deepfakes are generated using sophisticated algorithms that can replace or alter facial features, making it difficult to distinguish between real and fake content. While such technology has useful applications in fields like entertainment and virtual reality, it also introduces serious risks, including misinformation, identity theft, and privacy violations.

Traditional image processing techniques are often inadequate for detecting deepfakes due to their inability to capture complex and subtle visual patterns. In contrast, deep learning models, especially Convolutional Neural Networks (CNNs), have proven to be highly effective in image classification tasks. These models can automatically learn important features from images, making them suitable for identifying manipulated content.

This research focuses on developing a deepfake detection system using multiple CNN architectures. By implementing and comparing models such as GoogLeNet, InceptionV3, VGG16, DenseNet121, and Xception, the study aims to identify the most effective architecture for accurate detection. Additionally, the project extends beyond model development by deploying the best-performing model using a web-based interface, making the system practical for real-time use.

2.LITERATURE REVIEW

Hany Farid conducted significant research in the field of digital image forensics, proposing methods to detect manipulated media by analyzing inconsistencies in visual and statistical patterns. These techniques laid the foundation for forgery detection; however, they are less effective when applied to highly realistic deepfake images generated using advanced deep learning models.

Andreas Rössler et al. introduced the FaceForensics++ dataset and evaluated several deep learning approaches for detecting manipulated facial images. Their work demonstrated that Convolutional Neural Networks (CNNs) outperform traditional methods in identifying deepfakes, although their performance depends heavily on dataset quality and diversity.

Karen Simonyan and Andrew Zisserman developed the VGG16 architecture, which uses deep convolutional layers to extract detailed image features. Due to its strong feature extraction capability, VGG16 has been widely adopted in image classification and deepfake detection tasks.

Christian Szegedy et al. proposed the Inception architecture, including GoogLeNet and InceptionV3, which utilize multi-scale convolution filters to capture features at different resolutions. This approach improves computational efficiency while maintaining high performance.

Gao Huang et al. introduced DenseNet, an architecture that connects each layer to every previous layer, allowing better feature reuse and improved gradient flow. This design enhances learning efficiency and reduces common training issues such as vanishing gradients.

François Chollet proposed the Xception architecture, which uses depthwise separable convolutions to improve model efficiency and performance. This model has shown promising results in various image classification and deepfake detection tasks.

3. PROPOSED SYSTEM

The proposed system is designed to detect whether an input image is real or fake using deep learning techniques. The system follows a structured pipeline that includes data preprocessing, model training, evaluation, and deployment. The primary objective is to build an accurate and efficient deepfake detection system by comparing multiple Convolutional Neural Network (CNN) architectures.

The system begins with collecting and organizing a dataset containing real and fake images. These images are then preprocessed to match the input requirements of different CNN models. Data augmentation techniques are applied to increase dataset diversity and improve model generalization.

Multiple pretrained CNN models—GoogLeNet, InceptionV3, VGG16, DenseNet121, and Xception—are implemented using transfer learning. In this approach, the base layers of each model are retained, and custom classification layers are added for binary classification. This helps reduce training time while maintaining high performance.

After training, the models are evaluated using performance metrics such as accuracy, confusion matrix, precision, recall, and F1-score. A comparative analysis is performed to determine the best-performing model.

Finally, the selected model is deployed using a Flask-based web application. This allows users to upload images and obtain predictions in real time, making the system practical and user-friendly.

4. METHODOLOGY

The proposed methodology follows a systematic approach to develop and evaluate a deepfake detection system using multiple CNN models.

4.1 Dataset Collection and Organization

The dataset used in this project consists of images categorized into two classes: real and fake. The dataset is divided into three subsets:

- Training set
- Validation set
- Testing set

This structured division ensures proper training, validation during learning, and final evaluation of model performance.

4.2 Data Preprocessing

Before training, all images are resized according to model requirements:

224 × 224 × 3 for GoogLeNet, InceptionV3, VGG16, and DenseNet121
299 × 299 × 3 for Xception

Additionally, pixel values are normalized to the range [0,1], which helps in faster convergence and stable training of the models.

4.3 Data Augmentation

To improve model generalization and reduce overfitting, data augmentation techniques are applied using ImageDataGenerator. These include:

- Rotation
- Zooming
- Width and height shifting
- Shearing
- Horizontal flipping

These transformations increase dataset variability and help the model learn robust features.

Fig-5.3: Data Augmentation

4.4 Model Development

Five pretrained Convolutional Neural Network (CNN) architectures are implemented using transfer learning:

GoogLeNet – Uses Inception modules for multi-scale feature extraction

InceptionV3 – Improved version of Inception with optimized computation

VGG16 – Deep architecture with strong feature extraction capability

DenseNet121 – Uses dense connections for efficient feature reuse

Xception – Uses depthwise separable convolutions for better performance

In all models, pretrained weights from ImageNet are used, and custom classification layers are added for binary classification.

4.5 Model Training

The models are trained using the following configuration:

- Optimizer: Adam
- Learning Rate: 0.0001
- Loss Function: Binary Crossentropy
- Epochs: 10
- Batch Size: 32

The validation dataset is used during training to monitor performance and prevent overfitting.

4.6 Model Evaluation

The performance of the models is evaluated using:

Accuracy
Confusion Matrix
Precision, Recall, and F1-score

These metrics provide a comprehensive understanding of how well the model performs in classifying real and fake images.

4.7 Prediction and Deployment

The trained model is used to classify new input images as real or fake. The final system is deployed using a Flask-based web application, allowing users to upload images and receive predictions in real time.

5. RESULTS AND PERFORMANCE ANALYSIS

The performance of the implemented CNN models was evaluated using accuracy and classification metrics. A comparative analysis was conducted to determine the most effective model for deepfake detection.

The accuracy obtained by each model is presented in

Table 1: Accuracy comparison of CNN models

Model	Accuracy
GoogLeNet	59%
InceptionV3	65%
VGG16	72.16%
DenseNet121	69-70%
Xception	67.7%

From the results, it is observed that VGG16 achieved the highest accuracy, indicating its strong ability to extract detailed features from images. DenseNet121 and Xception also performed well due to their efficient architecture and feature reuse mechanisms. InceptionV3 showed moderate performance, while GoogLeNet achieved lower accuracy but maintained faster computation.

The training and validation accuracy graphs show that the models learned effectively over epochs, with minimal overfitting due to the use of data augmentation techniques. Confusion matrices further demonstrate the model's capability to correctly classify real and fake images.

Overall, the results confirm that deeper architectures combined with transfer learning provide better performance in detecting deepfake images.

6. SYSTEM IMPLEMENTATION AND OUTPUT

6.1 Prediction Results

The trained model was tested on unseen images to evaluate its performance in real-world scenarios. The system successfully classifies input images into two categories: Real and Fake, based on probability scores generated by the model.

In the demonstrated output, the uploaded image is predicted as REAL with a confidence score of approximately 67.33%, indicating the model's ability to generalize and make reliable predictions on new data.

6.2 User Interface (Flask Application)

To make the system practically usable, a web-based interface was developed using the Flask framework. This

interface provides an easy and interactive way for users to test the model.
The key features of the interface include:

- Selection of trained models
- Image upload functionality
- Automatic preprocessing of the input image
- Real-time prediction output
- Display of confidence score

After uploading an image and clicking the Predict button, the system processes the image and displays the result as either Real or Fake along with the corresponding confidence value.

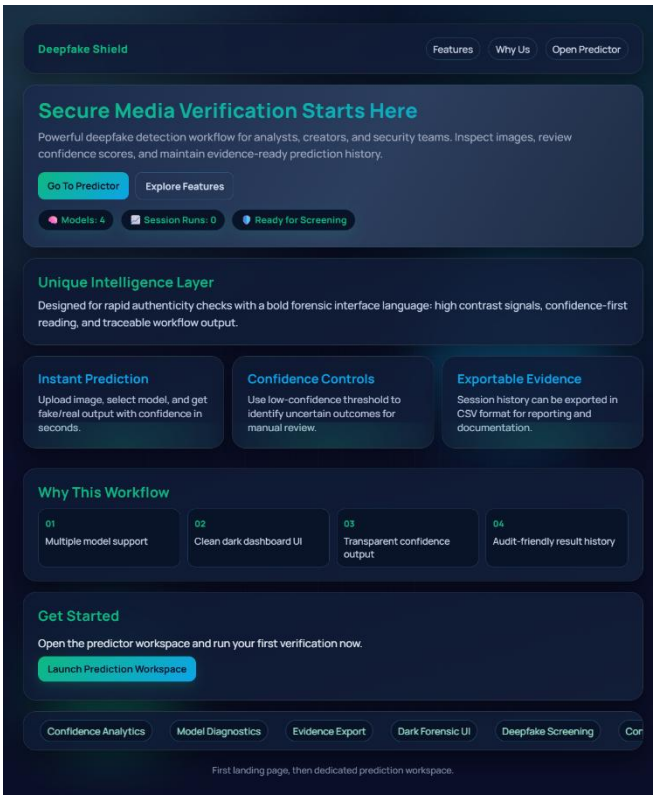

Fig -6.2 .1: Final Interface

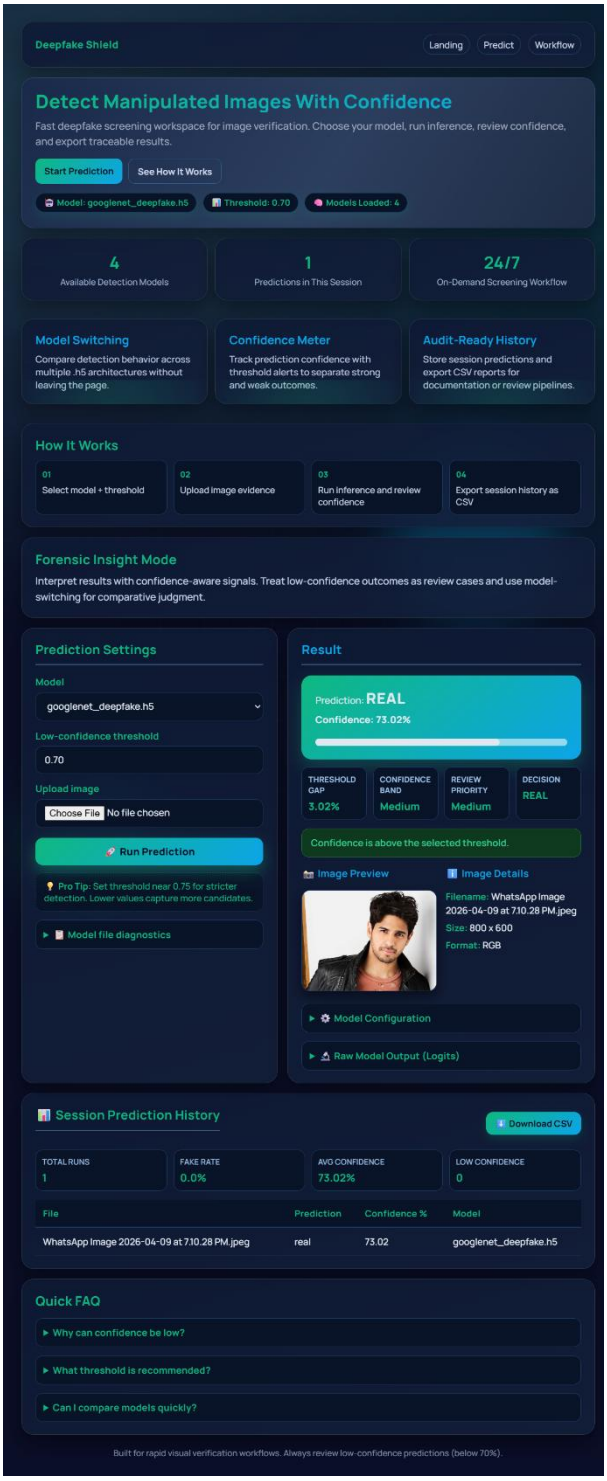

Fig-6.2 .2: Final Output

6.3 System Workflow

The overall workflow of the system is as follows:

Input Image → Preprocessing → CNN Model → Prediction → Output Display

This pipeline ensures efficient processing and enables real-time deepfake detection through a user-friendly interface.

7. DISCUSSION

The comparative evaluation of different CNN architectures highlights the importance of model design in deepfake detection. Among all the implemented models, VGG16 achieved the highest accuracy, which can be attributed to its deep architecture and ability to capture fine-grained image features. This makes it more effective in identifying subtle differences between real and manipulated images.

DenseNet121 and Xception also demonstrated strong performance due to their advanced architectural designs. DenseNet121 improves feature reuse through dense connections, while Xception enhances efficiency using depthwise separable convolutions. These characteristics contribute to better learning and improved classification results. In contrast, Inception-based models such as GoogLeNet and InceptionV3 showed comparatively lower accuracy, indicating a trade-off between computational efficiency and detection performance.

The use of transfer learning played a crucial role in improving model performance, as pretrained models were able to leverage previously learned features. Additionally, data augmentation helped increase dataset variability, reducing overfitting and improving generalization on unseen data.

The deployment of the system using Flask demonstrates its practical applicability. By providing a simple interface for users to upload images and receive predictions, the system bridges the gap between theoretical models and real-world usage. This makes it suitable for applications in digital media verification and cybersecurity.

8. CONCLUSION

This research presents a comprehensive deepfake detection system using multiple Convolutional Neural Network (CNN) architectures. The study combines a review of existing methods with the implementation and evaluation of different models, providing a complete analysis of deepfake detection techniques.

Among the implemented models, VGG16 achieved the highest accuracy, demonstrating its strong capability in extracting detailed image features. DenseNet121 and Xception also showed competitive performance, highlighting the effectiveness of advanced architectures in improving detection accuracy. The use of transfer learning significantly reduced training time while maintaining high performance, and data augmentation techniques further enhanced model generalization.

In addition to model evaluation, the system was successfully deployed using a Flask-based web application. This allows users to upload images and receive real-time predictions, making the system practical and user-friendly. The integration of model performance and real-world implementation strengthens the overall contribution of this work.

Overall, the results confirm that deep learning approaches are highly effective for detecting deepfake images. The study also emphasizes the importance of selecting appropriate architectures and preprocessing techniques to achieve better accuracy and reliability.

9. FUTURE WORK

Although the proposed system demonstrates effective performance in detecting deepfake images, there are several areas where further improvements can be made. Future work can focus on enhancing both the accuracy and scalability of the system.

One possible direction is to use larger and more diverse datasets, which can help the model generalize better and improve its robustness against different types of deepfake techniques. Additionally, advanced architectures such as EfficientNet or Vision Transformers can be explored to achieve higher accuracy and better feature representation.

Another important extension is the implementation of video-based deepfake detection, where temporal information and frame-level analysis can be utilized to detect manipulated videos more effectively. This would significantly increase the practical applicability of the system.

The system can also be improved by developing a real-time detection application for mobile or web platforms, making it more accessible to users. Furthermore, integrating explainable AI techniques can help in understanding model decisions and improving transparency.

Overall, future enhancements can make the system more accurate, efficient, and suitable for real-world deployment in areas such as cybersecurity, digital forensics, and media verification.

10. ACKNOWLEDGEMENT

The authors sincerely acknowledge the contributions of the research community in the fields of artificial intelligence and digital forensics that supported this work. We also extend our thanks to our guides and institution for their valuable support and encouragement during the project.

11. REFERENCES

[1] J. J. Bird and A. Lotfi, "CIFAKE: Image Classification and Explainable Identification of AI-Generated Synthetic Images," IEEE Access, vol. 12, 2024.

[2] H. Farid, "Image Forgery Detection: A Survey," IEEE Signal Processing Magazine, vol. 26, no. 2, pp. 16–25, Mar. 2009.

[3] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies and M. Nießner, "FaceForensics++: Learning to Detect Manipulated Facial Images," Proc. IEEE International Conference on Computer Vision (ICCV), 2019, pp. 1–11.

[4] Y. Li and S. Lyu, "Exposing DeepFake Videos By Detecting Face Warping Artifacts," Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.

[5] D. Güera and E. J. Delp, "Deepfake Video Detection Using Recurrent Neural Networks," Proc. IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS), 2018, pp. 1–6.

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going Deeper with Convolutions," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[8] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Proc. International Conference on Learning Representations (ICLR), 2015.

[9] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely Connected Convolutional Networks," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[10] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[11] B. Dolhansky et al., "The Deepfake Detection Challenge Dataset," arXiv preprint arXiv:2006.07397, 2020.

[12] H. Nguyen, J. Yamagishi and I. Echizen, "Capsule-Forensics: Using Capsule Networks to Detect Forged Images and Videos," Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019.

[13] X. Yang, Y. Li and S. Lyu, "Exposing Deep Fakes Using Inconsistent Head Poses," Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019.

[14] S. Agarwal et al., "Protecting World Leaders Against Deepfakes," Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.

[15] Z. Wang et al., "CNN-generated Images are Surprisingly Easy to Spot… for Now," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.