iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XI **Month of publication:** November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75458

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 XI Nov 2025- Available at www.ijraset.com

DeepFake Detection Using Convolutional Neural Networks

Swati Shilaskar[1], Sanika Jadhav[2], Yuvraj Jadhav[3], Jai Ughade[4]
Dept of E&TC Engineering, Vishwakarma Institute of Technology, Pune, India

Abstract: ***DeepFakes, or facial modifications produced by artificial intelligence, pose serious threats to the integrity of digital content and public trust. This paper proposes a method founded on Convolutional Neural Networks (CNNs) for binary facial image classification, separating real from fake. A properly processed and prebalanced dataset is utilized, and multiple data augmentation techniques are applied to ensure the model's maximum generalization capability. The proposed CNN structure, although relatively simple, is a sequence of several convolutional and pooling layers followed by dense layers with dropout regularization applied. Model training is performed with binary cross-entropy loss and optimized using the Adam optimizer. The experiment outcomes indicate that the model performs effectively on unseen test samples, thus demonstrating its capability to detect manipulated facial images and thus contributing to the overall goal of media integrity assurance.***

Index Terms: ***DeepFake Detection, Convolutional Neural Network (CNN), Image Classification, Deep Learning, Media Authenticity***

I. INTRODUCTION

The growing availability of generative models and deep learning libraries has resulted in a dramatic surge in the production of DeepFakes-synthetic media where a subject's face is manipulated or replaced digitally to produce a false representation. Such manipulated images and videos are produced with the help of sophisticated methods like autoencoders, Generative Adversarial Networks (GANs), and encoder-decoder models. DeepFakes are frequently indistinguishable from real media and raise important issues across digital forensics, disinformation campaigns, political manipulation, cybercrime, and online content integrity.

With the accelerating pace of development in DeepFake generation techniques, it is important that there be trustworthy detection systems to identify manipulated and real facial media. Though most current solutions tend to follow the transfer learning path with the aid of large-scale pretrained models like XceptionNet or EfficientNet, the latter may not always be a practical solution considering computational intensity and the lack of extensive customization.

This work introduces a lightweight and interpretable CNN-based detector that is specifically trained on DeepFake and actual face photos. In contrast to transfer learning approaches, the CNN is implemented from scratch using Keras and trained on a cleaned and augmented dataset. Data augmentation is utilized prominently in this system, increasing the variability of training samples and enhancing the model's capability to generalize to novel inputs. The detection process is based on the extraction of discriminative spatial features from face images and classification as real or fake based on patterns learned.

The model proposed here seeks to offer an easy and efficient solution for DeepFake detection in static images, possibly used in content verification systems, social media moderation, and digital authentication platforms.

II. LITERATURE REVIEW

Deepfake detection has been an essential area of research because synthetic media misuse has been on the rise in cybercrime, misinformation, and identity theft. Researchers have tried different approaches from traditional machine learning (ML) methods to sophisticated deep learning models such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and Vision Transformers (ViTs). This literature overview provides an inclusive summary of present research work on deepfake detection, integrating different methods and delineating their strong and weak aspects.

CNN-based deep models had shown ample possibilities in recognizing media manipulation. VGG16 and CNN model architectures used to identify deepfaked images and improved over the baseline transfer models, including Xception, NAS-Net, and MobileNet, and showed precision of 95% and accuracy of 94%.[1] EfficientNet and Capsule Networks (CapsNet) used for classifying deepfake images with a very high 99.64% validation accuracy on a 140,000 real and synthetic image dataset. Hybrid deep models had also become highly prominent in detecting deepfakes.[2] CNN-LSTM model created, where CNNs used spatial features and LSTMs used temporal dependency, with a 98.21% precision on DFDC and Ciplab datasets. [3] Similarly, [4] combined ResNet50 with

LSTM for deepfake detection using videos, which increased the accuracy from 84.75% to 87.48% for 40 epochs. These experiments suggested that combining the spatial and temporal analysis enhanced deepfake detection performance.

To improve robustness, scientists had combined Convolutional Neural Networks (CNNs) with Recurrent Neural Networks (RNNs). [5] presented an RNN-CNN combination model, in which ResNext50 handled feature extraction, whereas LSTM picked up temporal anomalies among video frames with a validation accuracy of 95.54%. InceptionResNetV2 and LSTM used and reached a 91.48% accuracy on FaceForensics++, DFDC, and Celeb-DF datasets.[6]

Further work had tried to use transformer-based models for deepfake detection. [7] proposed a Vision Transformer (ViT)-based model with Multi-Task Cascaded Convolutional Networks (MTCNN) for face extraction. The research achieved a 98.22% accuracy, surpassing CNN-based architectures. CNNs with Convolutional Vision Transformers (CVTs) combined, where the CNN-based model was 97% accurate and the CVT-based model achieved 85% accuracy with enhanced detection robustness.

Although image and video-based deepfake detection had been extensively researched, detecting deepfake audio was still a big challenge. Various researches solved this problem by inspecting spectral features and using deep learning models.[8] The researchers proposed a Hybrid CNN-LSTM-based deepfake audio detection framework, using MFCC, spectral contrast, spectral flatness, and chromagram features, and attained 94.73% accuracy on WaveFake and Release in the Wild datasets.

Self-supervised learning had been applied to boost deepfake audio detection. [9] A model presented with WavLM (self-supervised learning) and an MFA classifier, with state-of-the-art performance on ASVspoof 2021.[10] BTS-E, a detection mechanism using breathing rhythms in human speech was proposed, which increased classification accuracy by 46%, demonstrating the difficulty of generating natural breathing sounds during deepfake speech synthesis.[11]

Multimodal methods had also been explored to detect deepfakes. [12] implemented a real-time deepfake detector by combining auditory and visual indicators, with an accuracy of 90% and as a plugin for the Chrome browser. Eye movement examination with deep models (MesoNet4 and ResNet101) proposed and showcased enhanced real-time detection on FaceForensics++, CelebV1, and CelebV2 data sets with a range of 96.89% to 98.73% accuracy. Apart from deep learning, researchers had also tried traditional machine learning (ML) methods owing to their lower computational complexity.[13] Support Vector Machine (SVM), Random Forest (RF), and Extremely Randomized Trees (ERT) could be utilized effectively to detect deepfakes, which were 99.84% accurate on FaceForensics++ and 99.38% accurate on DFDC.[14] SVM coupled with frequency domain analysis used and 99.76% accurate in the CelebA dataset was achieved. Comparative study, [16] compared various classifiers including Custom CNN, VGG19, and DenseNet-121 and concluded that VGG19 was the most accurate with a 95% accuracy rate and thus was the best model to be used for forensic purposes.

Researchers had also examined transfer learning methods to improve model generalization among various datasets. [17] presented CLRNet, a Residual Network that employed Convolutional LSTM, and it largely outperformed five existing state-of-the-art methods on the FaceForensics++ dataset. Transfer learning with autoencoders studied and produced 98.69% accuracy using EfficientNet-based CNNs.[18]

A generalization study on dataset examined first- and second-generation deepfake datasets and concluded that the eye area was the most discriminative region in detecting deepfakes.[19] Further introduced a method that used image source anomaly and resulted in an HTER of 1.22% in cross-dataset evaluation, exhibiting strong generalization performance. While these advancements were notable, there were still challenges to expanding the real-world usability of deepfake detection.[20] A survey pointed out the accuracy vs. scalability tradeoff, where there is a focus on developing strong models with real-world noise and speech accent diversity.[21]

The literature reviewed showed that CNN-based models, hybrid CNN-LSTM networks, and transformers had been widely applied for deepfake detection with high accuracy rates. Although deep learning-based models like EfficientNet, ResNet, and Xception performed well, machine learning models like SVM and Random Forest were still competitive because of their efficiency and interpretability. Generalizable, real-time deepfake detection systems were much needed, so further research had to be pursued in transfer learning, multimodal methods, and adversarial resilience to effectively address deepfake menaces.

III. RESEARCH GAP

Despite much progress made so far in DeepFake detection by employing state-of-the-art deep learning models like CNNs, LSTMs, Vision Transformers, and hybrid multimodal architectures, the majority of the current methods depend on computationally heavy pretrained networks or intricate ensemble designs that demand big-data training and high-end hardware. Furthermore, though hybrid and transformer-based models possess high accuracy levels, their interpretation and deployment effectiveness in lightweight or real-time applications are still minimal.

Also, there is little research on from-scratch, domain-specific CNN architectures with high-quality augmentation approaches, especially with regards to instances involving constrained computer environments. This work fills the gap by suggesting a light, interpretable CNN model trained from scratch on facial images, fortified with strong data augmentation methods, to attain competitive accuracy without the need for transfer learning or large computational resources.

IV. METHODOLGY

The proposed system primarily works on detection of DeepFake facial images. These stages include dataset preparation, data preprocessing, data augmentation, architectural design of the Convolutional Neural Network (CNN), model training, and performance evaluation. Each step is implemented systematically to ensure model generalization, stability, and robustness on unseen data.

A. Dataset Description and Preprocessing

The dataset utilized in this research is organized into three distinct subsets: training, validation, and testing. The training set is used to optimize the model parameters during learning, the validation set is employed to tune hyperparameters and monitor overfitting, and the test set is reserved strictly for final model evaluation. Each subset contains facial images divided into two categories: Real**,** which includes authentic, unaltered facial images, and Fake**,** which consists of manipulated facial images generated using DeepFake techniques. The dataset is assumed to be balanced, with an equal distribution of both classes, and contains preprocessed face crops that are centered and resized to ensure uniformity in spatial dimensions.

Preprocessing is to improve model convergence and computational efficiency. All images are normalized by scaling pixel intensities to a range of [0,1], and resized to a fixed resolution compatible with the CNN input layer. Facial regions are assumed to be cropped in advance to eliminate irrelevant background noise and focus the model on discriminative facial features.

B. Data Augmentation

To enhance the diversity of the training data and mitigate the risk of overfitting, a comprehensive data augmentation pipeline is applied exclusively to the training images. Augmentation techniques simulate real-world variations in facial appearance by introducing controlled randomness. The transformations employed include random rotations, width and height shifts, shearing, zooming, horizontal flipping, and rescaling. These augmentations effectively expand the training set size and expose the CNN to a wider range of spatial transformations,

thereby improving its generalization capability. Importantly, the validation and test sets are not subjected to augmentation to maintain evaluation consistency.

C. System Architecture

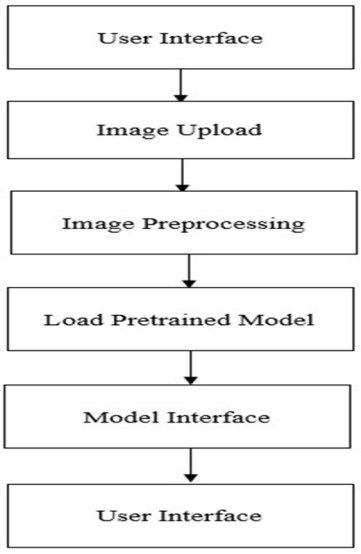


Fig1. System Architecture

The working model of the DeepFake detection system is illustrated in the flowchart shown in Figure 1. The process begins with the User Interface, where the user can upload an image. Once the image is uploaded, it is sent to the Image Preprocessing module, where resizing, normalization, and format conversion are performed to prepare the input for further analysis. The preprocessed image is then sent to the backend, where a Pretrained CNN Model is utilized to perform classification operations. The Model Interface controls the prediction process and returns a decision (Real or Fake) and a confidence score. The result is then sent back to the User Interface, thus offering an end-to-end smooth interaction for real-time DeepFake detection. The proposed system is structured as a sequential CNN pipeline that transforms an input image into a classification output indicating whether the image is real or fake. The block diagram of the architecture comprises three core components: an input layer**,** a sequence of convolutional and pooling layers**,** and a fully connected classifier**.** The input to the system is a facial image, which is then processed through multiple convolutional layers to extract hierarchical spatial features. These features are downsampled using pooling operations and passed through dense layers that output a probability score via a sigmoid activation function.

This pipeline allows the network to learn meaningful representations from the input images and distinguish between subtle differences inherent in real and DeepFake manipulations. The block diagram of the proposed CNN architecture is illustrated in Fig. 1, which outlines the sequential flow from raw input to binary classification output.

D. Convolutional Neural Network (CNN) Architecture

Algorithm 1: DeepFake Detection Using Convolutional Neural Network

Input: Dataset D = {Images, Labels}
Output: Trained CNN model M
1: Split D into training, validation, and test sets
2: Preprocess all images:
  a. Resize images to fixed dimensions (e.g., 128×128)
  b. Normalize pixel values to [0,1]
3: Apply data augmentation on training set:
  a. Random rotation, flipping, shifting, zooming, etc.
4: Define CNN architecture:
  a. Input → Conv2D → ReLU → MaxPooling
  b. Repeat Conv2D → ReLU → MaxPooling
  c. Flatten → Dense(512) → Dropout(0.5) → Output (Sigmoid)
5: Compile the model using:
  a. Loss = Binary Cross-Entropy
  b. Optimizer = Adam
6: Train the model on training set:
  a. Use early stopping with patience = 5 epochs
  b. Save best model weights using checkpointing
7: Evaluate model on test set:
  a. Compute accuracy, loss, and visualize performance
8: Return the final trained model M

The algorithm starts by partitioning the dataset which comprises facial images and respective labels, into training, validation, and testing subsets. Uniform dimensions are assigned to all the images with normalization to guarantee uniform input formatting. In an effort to enhance generalization and avoid overfitting, the training set experiences heavy data augmentation by undergoing intense geometric and intensity-based transformations like rotation, flipping, shifting, and zooming. The CNN architecture is then specified as stacked convolutional layers with ReLU activation functions followed by max pooling layers. The feature maps are flattened and fed into a fully connected dense layer with dropout regularization. An output neuron activated by sigmoid functions does binary classification.

The model is trained with binary cross-entropy loss and Adam optimizer. Training is done with early stopping to avoid overfitting and checkpointing to save the best model weights. The model is tested on the test set after training, and its performance is quantified in terms of classification accuracy and loss. The final result is a trained CNN model that can differentiate real and DeepFake images.

The CNN architecture designed for this task is implemented using the Keras Sequential API and is composed of multiple stages, each optimized to perform specific feature extraction and classification operations. The architecture begins with four convolutional layers with increasing filter sizes of 32, 64, 128, and 128, respectively. Each convolution layer utilizes the Rectified Linear Unit (ReLU) activation function to introduce non-linearity and facilitate gradient flow during backpropagation. Following each convolution operation, a MaxPooling2D layer is applied to reduce spatial dimensions and computational complexity while preserving dominant features. The output feature maps are then flattened into a one-dimensional vector and passed to a Dense layer consisting of 512 neurons activated by ReLU. A Dropout layer with a 50% dropout rate is included to prevent overfitting by randomly deactivating neurons during training. The final layer is a single neuron with a sigmoid activation function, responsible for producing a probability score between 0 and 1, indicating the likelihood of the input image being a DeepFake. The model is compiled with the binary cross-entropy loss function**,** the Adam optimizer**,** and accuracy as the primary evaluation metric.

E. Model Training Procedure

Training is conducted over a maximum of 30 epochs, with the model processing the entire training dataset iteratively to adjust weights using backpropagation. To ensure optimal performance and prevent overfitting, early stopping is employed. This mechanism halts training if the validation loss fails to improve for five consecutive epochs. Additionally, model checkpointing is utilized to save the model weights corresponding to the highest validation accuracy during training, ensuring the retention of the best-performing model configuration. Training and validation accuracy and loss values are recorded after each epoch to monitor learning progress. These metrics are visualized using training curves, which offer insights into the model's convergence behavior. A balanced batch size (typically 32) is used during training to ensure stable gradient updates while maintaining computational efficiency.

F. Evaluation Strategy

Once training is completed, the final model is evaluated on the held-out test set to determine its performance on unseen data. The evaluation includes calculating the test accuracy, final loss, and analyzing classification metrics. The training history is plotted and saved to a visual report file, providing empirical evidence of the model's learning trends and performance stability.

V. RESULTS AND DISCUSSION

To test the efficacy of the proposed CNN-based DeepFake detection model, experiments were performed sequentially on a real and synthetically created facial image dataset. The dataset was split into training, validation, and test sets with class balance.

Upon training, the model was tested on unseen data to assess its performance and resilience in detecting AI-manipulated content. Training and Validation Performance The model was trained for up to 30 epochs with early stopping set to watch validation loss. Training curves revealed smooth convergence with rising training accuracy and validation accuracy plateauing without overfitting. Use of data augmentation strategies, dropout regularization, and model checkpointing were central to overfitting and loss of performance prevention.

In order to analyze the model's decision-making capability in a better manner, two example images were passed through a special evaluation interface that performs pixel-level and artifact-based evaluation. Evaluation criteria include image coherence, facial consistency, lighting analysis, and artifact detection.

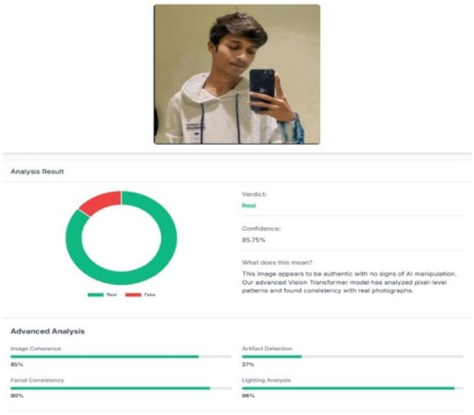


Fig.2 DeepFake detection result for real image (Verdict: Real)

In Fig. 1, the system classified another image and labeled it as Real with 85.75% confidence. The image had high facial consistency (90%) and good illumination (96%), which is common in real images. Low artifact detection (27%) and high image coherence score (85%) also indicated that the image was real.

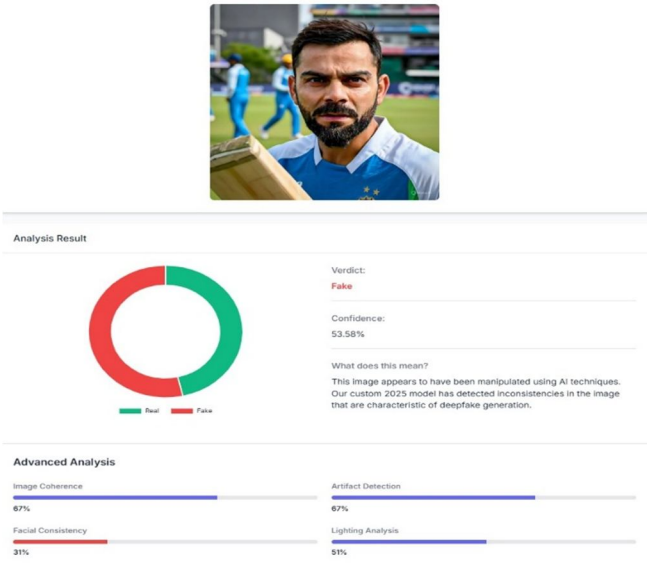


Fig 3. DeepFake detection result for synthetic image (Verdict: Fake)

As shown in Fig2, the model classified the image as Fake with a confidence of 53.58%. The higher measures of analysis indicated moderate image coherence (67%) and lighting analysis (51%), but very low facial consistency (31%). High artifact detection (67%) indicates the occurrence of anomalies common in GAN-generated images, e.g., unnatural blending or edge distortions. These are characteristic of deepfake manipulation patterns.

The experiments demonstrate that the proposed CNN model, despite its simplicity, can differentiate between real and manipulated facial images. Visual observation of predictions highlights the ability of the model to detect subtle inconsistencies such as poor facial blending and irregular lighting. The system's performance also demonstrates that handcrafted CNNs, when adequately trained and augmented, can achieve similar performance as more complex transfer learning-based methods.

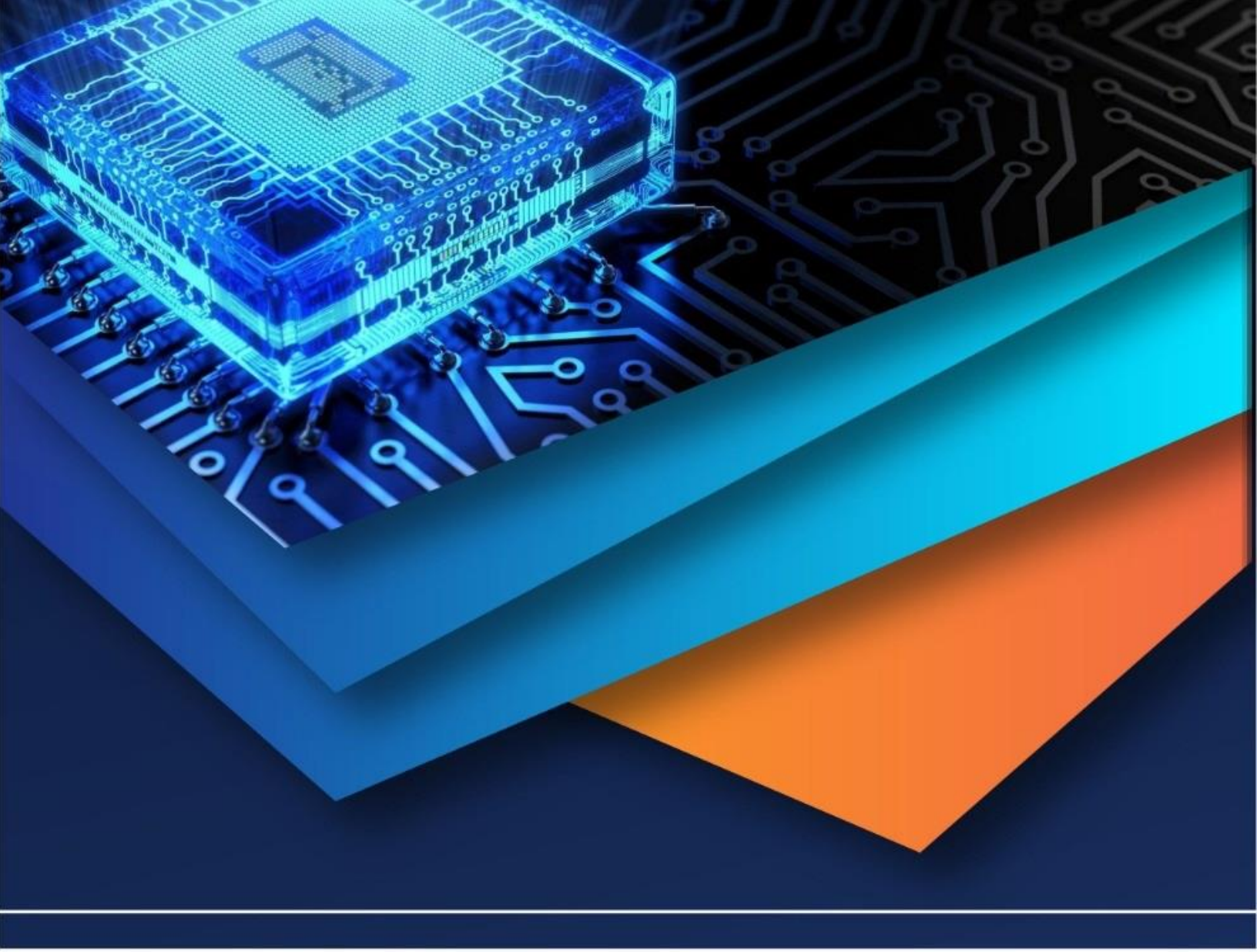
VI. CONCLUSION

This paper presents an approach for detecting DeepFake images based on a Convolutional Neural Network (CNN) which was developed and trained from scratch, independent of pretrained networks. Empirical tests verified the model's effectiveness in detecting inaccuracies characteristic of synthetic media like the addition of artifacts and facial abnormalities. The paper points out that even simple CNN architectures, provided they are properly trained, can potentially provide competitive efficiency in the prevention of actual-world DeepFakes, particularly in resource-constrained environments. While the present study is primarily centered on static images, the approach can be adapted for video-based DeepFake detection by adding temporal analysis. the increasing threat of synthetic media.

REFERENCES

[1] Chitale, Madhura, Aakanksha Dhawale, Manushree Dubey, and Sunil Ghane. "A Hybrid CNN-LSTM Approach for Deepfake Audio Detection." In 2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT), pp. 1-6. IEEE, 2024.
[2] Tariq, S., S. Lee, and S. S. Woo. "A convolutional lstm based residual network for deepfake video detection. arXiv 2020." arXiv preprint arXiv:2009.07480 (2020).
[3] Al-Dulaimi, Omar Alfarouk Hadi Hasan, and Sefer Kurnaz. "A hybrid CNN-LSTM approach for precision deepfake image detection based on transfer learning." Electronics 13, no. 9 (2024): 1662.
[4] Tipper, Sarah, Hany F. Atlam, and Harjinder Singh Lallie. "An Investigation into the Utilisation of CNN with LSTM for Video Deepfake Detection." Applied Sciences 14, no. 21 (2024): 9754.
[5] Ali, Raza, Munir Kashif, and Almutairi Mubarak. "A novel deep learning approach for deepfake image detection." Applied Sciences 12, no. 19 (2022): 9820.
[6] Almutairi, Zaynab, and Hebah Elgibreen. "A review of modern audio deepfake detection methods: challenges and future directions." Algorithms 15, no. 5 (2022): 155.

[7] Guo, Yinlin, Haofan Huang, Xi Chen, He Zhao, and Yuehai Wang. "Audio deepfake detection with self-supervised wavlm and multi-fusion attentive classifier." In ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 12702-12706. IEEE, 2024.
[8] Parikh, Aman, Kristen Pereira, Pranav Kumar, and Kailas Devadkar. "Audio-Visual Deepfake Detection System Using Multimodal Deep Learning." In 2023 3rd International Conference on Intelligent Technologies (CONIT), pp. 1-6. IEEE, 2023.
[9] Doan, Thien-Phuc, Long Nguyen-Vu, Souhwan Jung, and Kihun Hong. "BTS-E: Audio deepfake detection using breathing-talking-silence encoder." In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1-5. IEEE, 2023.
[10] Taeb, Maryam, and Hongmei Chi. "Comparison of deepfake detection techniques through deep learning." Journal of Cybersecurity and Privacy 2, no. 1 (2022): 89-106.
[11] Soudy, Ahmed Hatem, Omnia Sayed, Hala Tag-Elser, Rewaa Ragab, Sohaila Mohsen, Tarek Mostafa, Amr A. Abohany, and Salwa O. Slim. "Deepfake detection using convolutional vision transformers and convolutional neural networks." Neural Computing and Applications 36, no. 31 (2024): 19759-19775.
[12] Yadav, Priti, Ishani Jaswal, Jaiprakash Maravi, Vibhash Choudhary, and Gargi Khanna. "DeepFake detection using InceptionResNetV2 and LSTM." In International Conference on Emerging Technologies: AI, IoT, and CPS for Science Technology Applications. 2021.
[13] Sudharson, S., Priyanka Kokil, D. Sasikala, and Penta Aravinda Swamy. "Deepfake detection system using a hybrid model." In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-6. IEEE, 2023.
[14] Saini, Madan Lal, Arnav Patnaik, Dayal Chandra Sati, and Ratish Kumar. "Deepfake Detection System Using Deep Neural Networks." In 2024 2nd International Conference on Computer, Communication and Control (IC4), pp. 1-5. IEEE, 2024.
[15] Rana, Md Shohel, Mohammad Nur Nobi, Beddhu Murali, and Andrew H. Sung. "Deepfake detection: A systematic literature review." IEEE access 10 (2022): 25494-25513.
[16] Rajalaxmi, R. R., P. P. Sudharsana, A. M. Rithani, S. Preethika, P. Dhivakar, and E. Gothai. "Deepfake detection using inception-resnet-v2 network." In 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), pp. 580-586. IEEE, 2023.
[17] Rana, Md Shohel, Beddhu Murali, and Andrew H. Sung. "Deepfake detection using machine learning algorithms." In 2021 10th International Congress on Advanced Applied Informatics (IIAI-AAI), pp. 458-463. IEEE, 2021
[18] Agarwal, Harsh, and Ankur Singh. "Deepfake detection using svm." In 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 1245-1249. IEEE, 2021.
[19] Ashok, V., and Preetha Theresa Joy. "Deepfake Detection Using XceptionNet." In 2023 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE), pp. 1-5. IEEE, 2023.
[20] Singh, Richa, K. Ashwini, B. Chandu Priya, and K. Pavan Kumar. "Deepfake Face Extraction and Detection Using MTCNN-Vision Transformers." In 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), pp. 01-08. IEEE, 2024.
[21] Wang, Yufei, and Guangjun Liao. "Deepfake Video Detection Based on Image Source Anomaly." In 2024 IEEE 2nd International Conference on Image Processing and Computer Applications (ICIPCA), pp. 397-401. IEEE, 2024.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY