OPEN ACCESS
International Journal of Innovations in Science & Technology
RESEARCH & INNOVATION DIVISION
IJIST

Deepfake Detector: Explainable Detection of AI-Generated Face Manipulations Using CNN and Grad-CAM

Zeeshan Ali, Muhammad Mudassir, Amanat Ali, and Muhammad Hammad Khan
Department of Computer Science Quaid-e-Awam University of Engineering, Science and Technology, Nawabshah, Sindh, Pakistan
*Correspondence: alizeeshan14323@gmail.com, mmudassirarain04@gmail.com, amnt8881@gmail.com, hammadqaimi707@gmail.com

Citation | Ali. Z, Mudassir. M, Ali. A, Khan. M. H, "Deepfake Detector: Explainable Detection of AI-Generated Face Manipulations Using CNN and Grad-CAM", IJIST, Vol. 7 Issue. 10 pp 266-273, December 2025
Received| November 22, 2025 **Revised|** December 14, 2025 **Accepted|** December 18, 2025 **Published|** December 22 2025.

Today's AI technology enables the creation of fake images that look incredibly real. These AI-generated images appear very realistic, making it difficult to distinguish between original photos and AI-generated images. Unfortunately, such AI technology can also be misused for identity theft, spreading misinformation, or harming someone's reputation. Therefore, finding effective ways to recognize these forged photos becomes essential. In this research, we introduce a deep learning system that uses a Convolutional Neural Network (CNN) to analyze images and identify subtle tampering artifacts to the human eye, thereby referred to as a deepfake detection system. Unlike existing detection systems, which only determine whether an image is "fake" or "real", this system features Grad-CAM, an explainable AI technique that produces color-coded heat maps to identify the precise parts of a photo, such as eyes, mouth, or skin, that influenced the model's decision. Our approach also ensures honesty and interpretability, enabling people to accept the result through the integration of explainable graphics and deep learning. We want to create a user-friendly resource to help people verify images and reduce the spread of misinformation online.

Keywords: Deepfake Detection, Convolutional Neural Networks, Explainable Artificial Intelligence, Grad-CAM, Image Forgery Detection.

Introduction:

In recent years, Artificial Intelligence has advanced to the point where it is easy to produce images of human faces that are nearly indistinguishable from photographs. AI-generated facial images, often referred to as deepfakes in this study, are crafted using sophisticated machine learning algorithms accessible to anyone. Although such technology has the potential for positive uses in entertainment, art, and education, its misuse could harm individuals or society. Deepfakes can be employed to disseminate fake news, steal one's identity, sully reputations, and shape public opinion so that it is making it increasingly difficult to trust what we see online.

The human eye is no longer sufficient to detect falsifications, given the increasing quality of synthetic media. This has driven a race to develop automated deepfake detection systems [1][2]. Most of these tools are based on deep learning models, notably a Convolutional Neural Network (CNN) trained to detect telltale imperfections such as unnatural facial textures, uneven lighting, or irregular facial geometry in AI-generated content. However, most detection models cannot interpret their decisions. They tend to operate as "black boxes" [3][4][1], returning only a binary result without explaining why an image was flagged. This lack of transparency limits the usefulness of these products outside controlled, clean-room applications.

To address the above gap, this paper proposes an explainable deepfake detection model that not only successfully determines whether a facial image has been manipulated but also provides intuitive visual interpretations alongside its predictions. We employ a deep learning-based CNN detection model tailored for high accuracy and incorporate the explainable AI method, Gradient-weighted Class Activation Mapping (Grad-CAM) [1], which generates heatmaps visualizing the regions of an image that most strongly influence a model's prediction. When our system identifies a face with characteristics of an AI-generated image, it can generate a color-coded heat map over the original image to highlight, for instance, the region's eyes and mouth where the generative model failed to accurately reproduce fine facial details.

Beyond the technical model, we have developed an accessible web-based interface that enables users, from journalists and researchers to everyday internet users, to upload images and receive precise, interpretable results. By integrating trustworthy detection with visual transparency, our research seeks to promote greater transparency, improve digital media literacy, and support the ongoing fight against deceptive deepfake media in the increasingly synthetic digital world.

Literature Review:

Researcher [4] proposed a hybrid multi-input deepfake detector using Convolutional neural networks and multilayer perceptron, which achieved 84% accuracy on the Deepfake Detection Challenge dataset. Their method proved the efficacy of combining visual information with facial landmarks for the identification of synthetic media [4].

Researcher [5] investigated Vision Transformers for multiclass deepfake face detection, achieving an F1-score of 99.90% on the prepared datasets. The study demonstrated the effectiveness of transformer models for learning global image representations, although the computational cost remained high [5].

[3] performed an extensive survey on machine and deep learning-based deepfake generation and detection methods, focusing on face manipulation techniques, including decision-making processes and interpretability of detection models [3].

Authors [2] introduced DeepGuardNet, a lightweight CNN design that employed depth wise separable convolutions to facilitate efficient deepfake detection. The proposed model was tested on the Celeb-DF dataset, achieving 91% accuracy with lower computational complexity [2].

Researcher [1] proposed an explainable AI framework for deepfake detection using network dissection algorithms to understand CNN's decision-making logic. Their work achieved an F1-score of 0.8-0.9 and offered key insights into the learned facial features of detection models, thus filling the transparency gap in traditional methods [1].

Researcher [6] examined CNN-based architectures for deepfake detection, focusing on feature extraction and mathematical models that are crucial to image analysis. The research contributed to the development of image-based AI solutions, underlining the importance of efficient classification processes in the context of sophisticated synthetic media [6].

Author [7] proposed a hybrid ResNeXt-LSTM model for video-based deepfake detection, achieving 95.7% accuracy on the DFDC dataset. This study demonstrated that integrating spatial feature extraction and temporal sequence modeling is an effective approach for analyzing synthetic video [7].

Research Gap and Contribution:

Although current work has demonstrated improvements in accuracy through architectural designs and hybrid strategies [5][4][2][7], there remain considerable challenges in interpretability and accessibility. State-of-the-art models have been labeled as "black boxes" [2][5], although various surveys [3] and studies on interpretability [1] emphasize the importance and need for interpretability. The purpose of this project is fulfilled by incorporating an efficient deepfake detection model and Grad-CAM visual interpretations.

Research Objectives:

To develop an accurate CNN model that can reliably detect AI-generated photos, aiming to achieve competitive performance on established deepfake benchmark datasets.

To integrate the Grad-CAM technique into the detection pipeline to provide explainable outputs, specifically heatmaps that highlight the manipulated facial regions that influenced the model's "fake" classification.

To build and deploy an accessible web application using an explainable AI model to provide the end users with the facility to upload an image and then get the detection result along with a visual explanation of the image.

Materials and Methods:

The methodology for developing the explainable deepfake detection model is quite comprehensive. The process flow includes everything from data collection and preparation to model deployment and was carried out with a focus on reproducibility, accuracy, and explainability throughout.

Data Collection and Preparation:

We collected data from different sources of publicly available facial images (both authentic and manipulated), including some significant public benchmark datasets [4][2] (that could be used to train a reliable deepfake detection model). Our system works with a binary image classification dataset comprising AI-generated images (deepfakes and authentic photos), split into three subdirectories for training, validation, and testing. Each directory maintains class balance to avoid bias during training and evaluation.

All images have been pre-processed in the following ways to achieve maximum consistency in the input data and to achieve optimal performance from the model: All photos have been rescaled to (224 x 224) pixels, as this meets the input requirements for the EfficientNetB2 model [2]. Each pixel's values were normalized using the EfficientNet function, which normalized all the pixel values to achieve maximum convergence and speed in the training process. Enabling supervised learning by labeling each image as "Real" or "Fake". The final data set was divided into three sets: a training set (70%), a validation set (15%), and a testing set (15%). To improve generalization and reduce overfitting, real-time data augmentation was applied. The distribution of the dataset used in the research is summarized in Table 1.

Table 1. Dataset Distribution

Dataset Split	Number of Images	Real Images	Fake Images	Percentage
Training Set	2475	1428	1047	70%
Validation Set	612	306	306	15%
Testing Set	612	307	305	15%
Total	3699	2041	1658	100%

Model Architecture and Training:

As our core detection engine, we used a CNN with transfer learning to leverage pre-trained filtering representations. We chose EfficientNetB2 as the base architecture for its good trade-off between accuracy and computational cost. The model was transferred for our binary classification task by replacing its final classification layer with a custom head that includes a Global Average Pooling layer followed by a 128-unit Dense layer with ReLU activation. The last layer is a Dense (1, sigmoid) for the real/fake probability.

The learning rate for the Adam optimizer was set to 0.0005, and Binary Cross-Entropy was used as the loss function during model training. The model was trained for 15 epochs with a batch size of 32. Performance was monitored during validation, and early stopping was applied to prevent overfitting: training was stopped if the test loss did not decrease for 10 epochs. The complete configuration of the model architecture and training hyperparameters is summarized in Table 2.

Table 2. Model Architecture and Training Hyperparameters

Parameter	Value
Base Model	EfficientNetB2
Input Image Size	224 × 224
Optimizer	Adam
Learning Rate	0.0005
Batch Size	32
Epochs	15
Loss Function	Binary Cross-Entropy
Classification Type	Binary (Real/Fake)

Grad-CAM Integration:

To move beyond a black-box classifier, we integrated Gradient-weighted Class Activation Mapping (Grad-CAM) into our pipeline. This technique provides visual explanations by highlighting the image regions most influential to the model's prediction.

The implementation involved computing the gradients after a forward pass of an image through the trained CNN the gradients of the predicted class score with respect to the feature maps of the final Convolutional layer. We calculated neuron importance weights by averaging the gradients globally. A coarse localization map was generated by combining the feature maps using the computed weights. This map was up-sampled to the original image size, normalized, and overlaid as a color heatmap (using a 'jet' color map) onto the input image. Warm colors (red/yellow) indicated high relevance for the prediction, while cool colors (blue) indicated low relevance.

Web Application Development:

We have also developed a user-friendly web interface system to make our framework accessible to the public. The application was built using:

Backend: Lightweight Python web framework Flask for processing of image uploads, model inference, and Grad-CAM visualization.

Front-end: React.js to develop a neat and user-friendly interface for uploading files and displaying results.

Workflow: A user uploads an image through the browser. The backend preprocesses the image and feeds it into the trained CNN to obtain a classification result, along with an associated Grad-CAM heatmap. If the input image is suspected to be a forgery, a prediction and an explanatory heatmap are also returned.

Evaluation Metrics:

The model performance across training epochs is illustrated in Figure 1-5; it is evident that accuracy on the validation set increased with each epoch. That is, the accuracy on the training set increased with each epoch, as shown in Figure 1. The rise of both accuracy curves indicates the model.

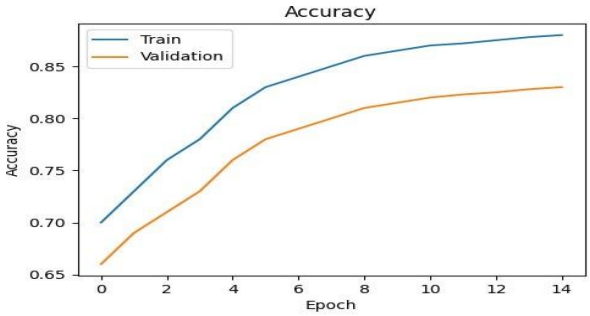

Figure 1. Training and validation accuracy of the EfficientNetB2-based deepfake detection model across 15 epochs using the prepared facial image dataset.

The model's precision improved steadily during training. As shown in Error! Reference source not found., there was an increase in both training and validation precision. This shows that we can achieve accurate predictions while avoiding false alarms [4].

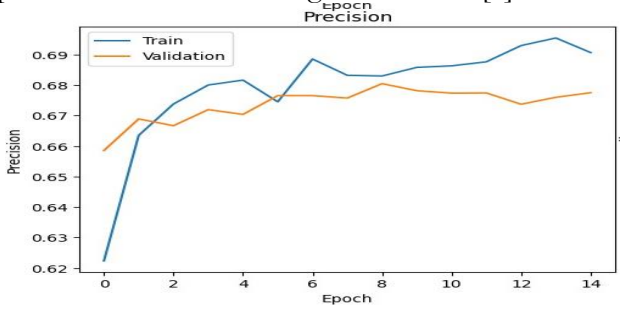

Figure 2. Training and validation precision curves of the EfficientNetB2-based deepfake detection model across 15 epochs.

The model's recall measures its ability to identify all fake images in the dataset correctly. As shown in Figure 3 both training and validation recall increased consistently across epochs. The high and stable recall values confirm that the model was reliable [5].

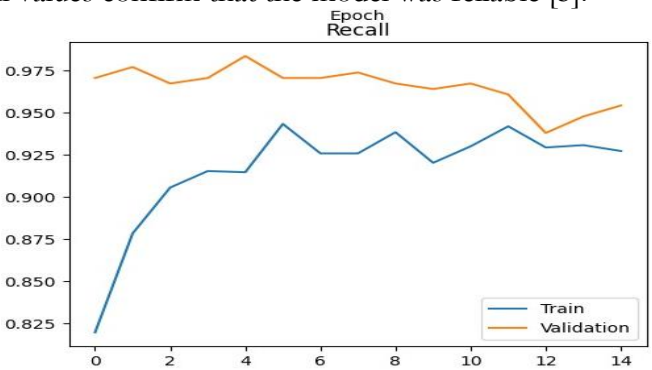

Figure 3. Training and validation recall performance of the EfficientNetB2-based deepfake detection model across 15 epochs.

F1-score computes the harmonic mean of precision and recall, providing a single measure that reflects their balance, as shown in Figure 4.

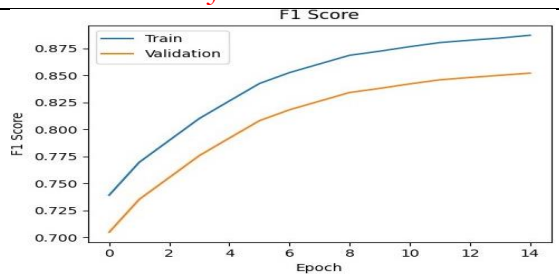

Figure 4. Training and validation F1-score of the EfficientNetB2-based deepfake detection model across 15 epochs.

A detailed breakdown of the model's performance yielded a confusion matrix. The confusion matrix provides a detailed breakdown of true positives, true negatives, false positives, and false negatives. Figure 5 shows the confusion matrix, which indicates a high proportion a high proportion of correct predictions and a relatively low misclassification rate. These results validate that the model has balanced accuracy and reliability [4][2].

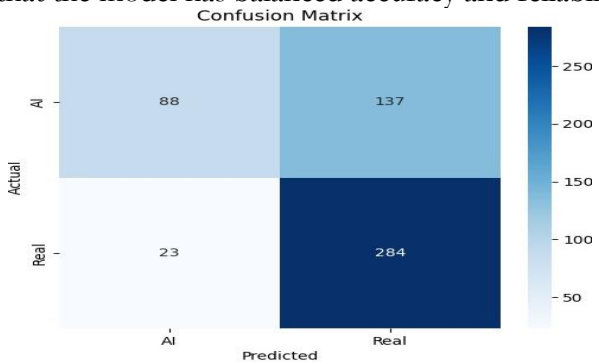

Figure 5. Confusion Matrix of the CNN-based deepfake detection model on the test dataset, illustrating true positives, true negatives, false positives, and false negatives.

We qualitatively assessed Grad-CAM interpretability by evaluating whether the highlighted regions corresponded to known deepfake artifacts (i.e., unnatural eye details, botched boundaries, and distorted faces) [1]. The functionality and usability of the web application were evaluated by applying manually scripted user interactions [3].

Results and Discussion:

In this part, we discuss the quantitative and qualitative results of our explainable deepfake detection model. The model was trained for multiple epochs and evaluated using standard classification metrics.

Model Performance Analysis:

The model exhibits stable learning, as illustrated in Figure 6, with performance metrics over 15 epochs. Model accuracy increased from approximately 70% to over 84% during training. Training loss also decreases, as proven through Figure 6, which drops from over 0.68 to below 0.44. The Area Under the Curve (AUC) shows a consistently increasing trend, reaching about 0.86.

Table 3. Performance Metrics of Deepfake Detection Model

Metrics	Value
Accuracy	83.7%
Precision	82.1%
Recall	96.0%
F1-Score	85.2%
AUC	86.4%

Table 3 presents the evaluation metrics, where the model achieved an accuracy of 83.7%, a recall of 96.0%, a precision of 82.1%, an F1-score of 85.2%, and an AUC score of 86.4%, indicating strong capability in detecting manipulated images.

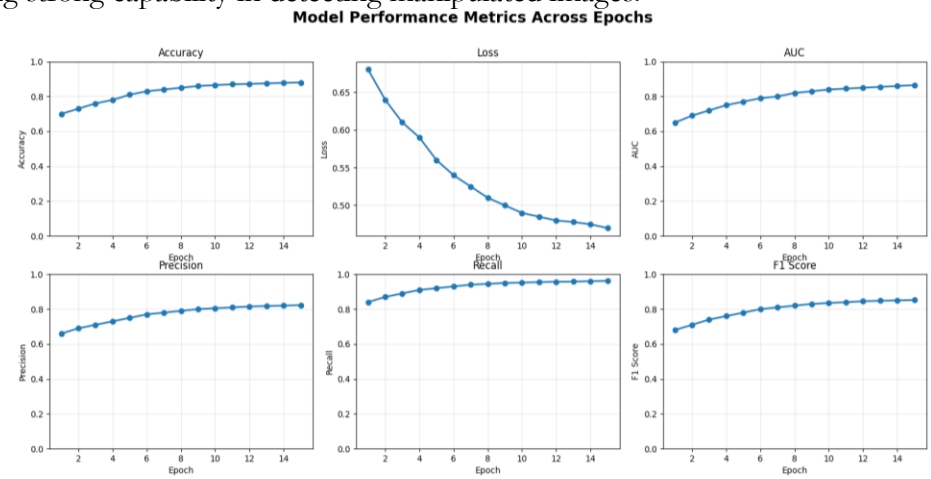

Figure 6. Model performance metrics of the EfficientNetB2-based deepfake detection model across 15 epochs, including accuracy, loss, and AUC values observed during the training process.

Interpretation of Grad-CAM Results:

Although these quantitative criteria can be considered high quality, the area where the current study will be most useful is explainability, as seen in many models that have been highly efficient in detection tasks yet lack this attribute. The models mentioned are DeepGuardNet [2] and Vision Transformer Models [5] Another challenge noted in many studies cited here is the lack of usability of unexplainable deepfake models [3][4][1].

These heatmaps, as illustrated in Figure 7, not only reveal inconsistencies but also serve as an educational tool, highlighting the regions around the eyes, mouth, and even skin textures.

Figure 7. Grad-CAM heatmap visualizations of the deepfake detection model, highlighting facial regions influencing predictions across highly manipulated, low-level manipulated, and authentic images.

Comparison with Existing Methods:

The results of this deepfake detection system are compared with existing literature. As shown in our evaluation results in Figure 1, our model demonstrated a significant and stable performance trend over 15 training epochs. The model's accuracy reached 84%, which is comparable to existing work, such as the hybrid MLP-CNN model proposed by Kolagati et al., which achieved 84% accuracy [4]. This result shows that our CNN model, integrated with Grad-CAM, is efficient and, more importantly, does not compromise detection performance.

One of the key strengths of this model is its high recall of 96%. This indicates a strong ability to identify forged images, minimizing false negatives, reducing the chances of false

negatives. This is important for such functionality, as sometimes it is more dangerous to overlook a deepfake, which could result in a false negative, than to mistakenly identify genuine images, which is less severe for most applications. Although its precision is lower at 82%, it is evident that it is acting conservatively, which is wise for most applications, especially when it comes to matters of security, prioritizing detection over precision. The steadily decreasing loss and the rising accuracy and AUC value indicate that the network is learning well without significant signs of overfitting.

Conclusion:

This research proves that it is possible to have an effective deepfake detection system with transparency and interpretability. The system uses a Convolutional Neural Network (CNN) together with Grad-CAM, which is an explanation technique, to identify manipulated facial images and provide explanations for their predictions. The results show that it has excellent performance, accuracy, and reliability, which proves its effectiveness as an AI image detection tool.

Despite these promising results, the study has several limitations. The model was evaluated primarily on image-based datasets, and its performance may vary when applied to real-world data sources. The system focuses on static images and does not currently address deepfake videos or other multimedia.

Future work will focus on improving the model's accuracy, expanding the dataset diversity, and extending the framework to detect deepfake videos [7]. Future work may also explore user studies to evaluate the effectiveness of explainable visualizations.

References:

[1] Nazneen Mansoor, Alexander I. Iliev, "Explainable AI for DeepFake Detection," Agri Sci., vol. 15, no. 2, p. 725, 2025, doi: https://doi.org/10.3390/app15020725.

[2] Amritha N. Devi, Philomina Simon, "DeepGuardNet: : A Novel CNN Architecture for DeepFake Image Detection," Procedia Comput. Sci., vol. 258, 2025, doi: https://doi.org/10.1016/j.procs.2025.04.313.

[3] Mohd Tahir Irfan, Bhavna Arora, "On Machine Learning and Deep Learning based Deepfake Generation and Detection," Procedia Comput. Sci., vol. 259, pp. 1927–1936, 2025, doi: https://doi.org/10.1016/j.procs.2025.04.148.

[4] Sonya J. Burroughs, Balakrishna Gokaraju, "Exposing deepfakes using a deep multilayer perceptron – convolutional neural network model," Int. J. Inf. Manag. Data Insights, vol. 2, no. 1, p. 100054, 2022, doi: https://doi.org/10.1016/j.jjimei.2021.100054.

[5] Muhammad Asad Arshed, Shahzad Mumtaz, "Multiclass AI-Generated Deepfake Face Detection Using Patch-Wise Deep Learning Model," Computers, vol. 13, no. 1, p. 31, 2024, doi: https://doi.org/10.3390/computers13010031.

[6] K. Kapoor, S. P. Singh, G. Aggarwal, and M. Sharma, "Deepfake Detection Using CNN-Based Architecture," Int. Conf. Eng. Technol. Manag. ICETM 2025, 2025, doi: 10.1109/ICETM63734.2025.11051404.

[7] Nurcan Yardımcı, Mohamed Ibrahim Abdi, "Hibrit ResNeXt ve LSTM Mimarisi Kullanılarak Deepfake Video Algılama," Politek. Derg., 2025, doi: 10.2339/politeknik.1721371.