Varendra International Journal for Interdisciplinary Research 2025: 1(2), 42-51

ISSN: 3078-1017 (Online). 3078-7009 (Print)
© Center for Interdisciplinary Research, Varendra University, Rajshahi, Bangladesh
https://doi.org/10.64296/vijir.v1i2.01
Received: June 10, 2025; Revised: September 15, 2025; Accepted: October 21, 2025

FusionNet: A Comprehensive Framework for Automated Deepfake Detection

Md. Karimul Islam[1], Md. Rafsan Yeasir[1], Mst. Saima Rahman Mou[1], Md. Fatin Nibbrash Nakib[2], Md. Taufiq Khan[1], Md. Arafat Ibna Mizan[2]

[1]Department of Computer Science and Engineering, Varendra University, Rajshahi, Bangladesh
[2]Department of Computer Science and Engineering, Bangladesh Army University of Engineering & Technology, Natore-6431, Bangladesh
[*]Corresponding Author: khantaufiq2001@gmail.com

ABSTRACT

Advancement in deep generative models have led to the spread of deepfake media, which poses a serious danger to information authenticity, privacy, and trust in digital media forensics. In order to meet the increasing demand for reliable deepfake picture identification, this study uses the publicly accessible Kaggle Deepfake Dataset to investigate hybrid deep learning approaches. Initial experiments with standalone Keras models, including DenseNet121, RestNet50, Xception, NASNet-Mobile, VGG16, and InceptionV3, achieved accuracies ranging from 89% to 90% over 20 epochs. To push the boundaries of performance, we developed a novel hybrid model that concatenates feature maps extracted from DenseNet121 and DenseNet169. This method greatly improved classification performance, obtaining a 91.22% accuracy rate with high precision 90.82%, 94.37% recall, and 92.56% F1-score with error rate 8.78%. Confusion matrices, classification reports, and ROC curves were used to thoroughly assess the model, which showed that it was effective at differentiating real photos from ones that had been altered, with an AUC score of 0.98. Our findings underscore the effectiveness of combining pre-trained CNN architectures for deepfake detection and contribute to the advancement of scalable and reliable solutions for safeguarding digital media integrity.

Keywords: Deepfake detection, Image classification, Deep Learning, Hybrid feature extraction, Convolutional Neural Network (CNN)

1. Introduction

Deepfakes are fake media produced by artificial intelligence (AI) that covertly adds a person to a photo or video where they are not. Development of deepfakes used to require a great deal of time and experience, but advances in machine learning have made it simpler and faster. In order to create realistic looks and expressions, neural networks are trained on vast amounts of human data. Although they have contributed to deepfakes, techniques such as GANs are not necessarily the main approach. AI and conventional graphics are used in modern systems to provide visually appealing outcomes [1]. They are used to produce harmful media, such as phony pornography directed at celebrities and women. They enable deception in the business world, such as posing as executives to embezzle money. Deepfakes also undermine privacy and trust, increasing psychological abuse and suffering. Although there are certain valid uses, including cost-effective marketing, the risks frequently exceed the advantages [2]. Deep learning improves deepfake detection by using neural networks to identify patterns and subtle anomalies in digital media that indicate manipulation. Techniques like facial and body movement analysis, audio analysis (such as voice modulation and synthesis), and behavioral analysis help spot deepfakes by examining discrepancies in expressions, voice, and actions. Current advancements include the use of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) for training models on large datasets and the integration of deepfake detection tools into social media platforms for real-time identification [3]. This work makes the following contributions to the field:

- Suggested a hybrid framework that combines the DenseNet121 and DenseNet169 models to detect deepfakes.
- Exceeded standalone models in terms of performance parameters, achieving 91.22% accuracy with a lower error rate of 0.0878.

The structure of the paper is as follows: Section 2 examines relevant research on deepfake identification and contemporary issues. The suggested technique and hybrid model are described in depth in Section 3. The experimental design and execution are explained in Section 4. The results and analysis are presented in Section 5, and the paper is concluded and future directions are discussed in Section 7.

2. Related Works

In order to unify the assessment of face alteration detection methods and to meet the growing concerns about the authenticity of digital material, Afchar et al. [4] present Meso-4 and MesoInception-4, two lightweight deep learning models for identifying face video forgeries, particularly Face2Face and Deepfake manipulations. The networks achieve remarkable detection rates of 98% and 95% for Deepfake and Face2Face forgeries, respectively, by concentrating on mesoscopic picture attributes and striking a balance between computational economy and performance. Although robustness to real-world video compression is highlighted by Afchar et al., results degrade at high compression settings.

Ahmed et al. [5] offer a rationale-augmented convolutional neural network (CNN) for deepfake identification, which achieves an impressive 95.77% accuracy on the Kaggle DeepFake Video dataset. Using MATLAB, their approach focuses on effective real-time processing and addresses issues like facial reconstruction for security applications using webcams and surveillance cameras. Ahmed et al. show how reliable the model is and how it can be used to identify edited videos with little loss of data. Limitations, however, include the need for high-quality datasets for efficient performance and the computational cost of real-time implementation. By advancing the field of deep learning-based Deepfake detection, the work offers a viable method for forensic and security applications.

Raza et al. [6] present a unique deepfake predictor (DFP) method that detects altered photos from the Kaggle dataset with 94% accuracy and 95% precision by combining VGG16 and convolutional neural network layers. Their analysis demonstrates how the suggested DFP model outperformed other approaches, including NAS-Net, Xception, MobileNet, and the conventional VGG16, which produced inferior performance measures. The model's shortcomings include its dependence on a small dataset (1081 actual and 960 fake photos) and its difficulty generalizing to big, diverse datasets, despite the fact that it shows promise for cybersecurity applications. The study emphasizes how hybrid architectures can be used for reliable and effective deepfake detection.

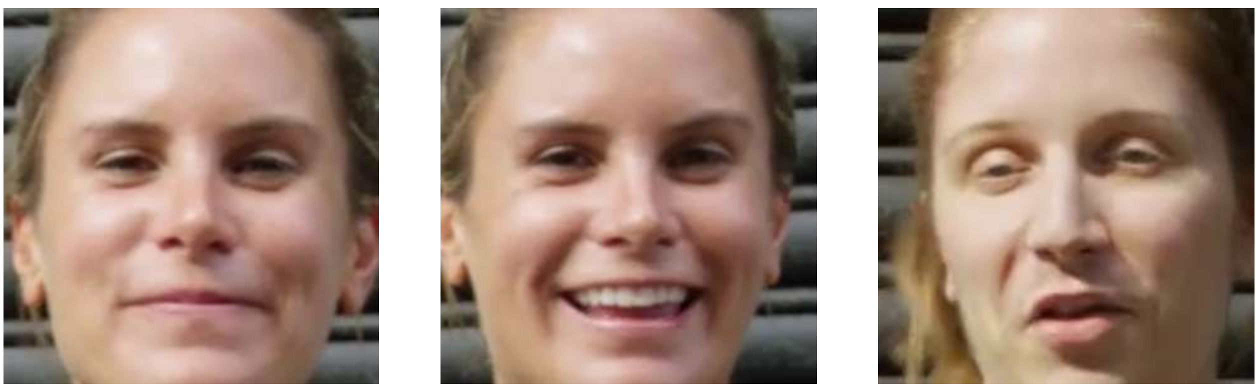
Rössler et al. [7] introduced FaceForensics++, a large-scale benchmark dataset intended to address growing concerns about facial image modification and its societal repercussions. More than 1.8 million photos produced with four cutting-edge alteration techniques—Face2Face, FaceSwap, DeepFakes, and NeuralTextures—are included in this collection. The study's main contribution is the establishment of an automated benchmark for assessing manipulation detection techniques in practical settings, such as compression and different resolutions. With XceptionNet reaching the greatest accuracy of 99.26% on raw videos, the authors also showed how utilizing domain-specific knowledge greatly improves the performance of deep learning models. The paper does point out several drawbacks, though, such as difficulties identifying GAN-based manipulations like NeuralTextures, where model performance degrades under severe compression.

3. Methodology

To detect deepfake photos with high accuracy, the suggested architecture uses a hybrid deep learning technique. This work improves the integration of multiple pre-trained deep learning models and assesses their performance to enhance classification results.

3.1. Dataset

We used the publicly accessible Kaggle Deepfake Dataset [9], which includes 11,407 images divided into real and deepfake categories, for our investigation. The dataset was initially split into an 80:20 ratio, creating separate training and testing sets. Subsequently, 80% of the training data was further divided into training and validation subsets using an 80:20 ratio. A visual representation of the dataset samples is provided in Fig. 1.



(a) Image 1 (b) Image 2 (c) Image 3

Figure 1: Samples from Dataset

3.2. Data Preprocessing & Augmentation

In order to ensure uniformity and improve the stability of the training process, normalization was used during the preprocessing step to scale the pixel values of all images within the range [0, 1]. A number of data augmentation strategies were used to increase the model's ability to generalize to fresh and untested data. TABLE 1 illustrates the augmentation techniques implemented in this study.

3.3. Training Models

3.3.1. DenseNet121

A pre-trained version of the model, initialized with ImageNet weights, was adjusted for the deepfake detection task in order to maximize DenseNet121's performance. Custom layers designed for binary classification were used in place of the original architecture's completely connected layers. These comprised a global average pooling layer, dense layers with dropout regularization to avoid overfitting, and ReLU activation. In order to speed up and stabilize the training process, batch normalization was added. Lastly, to produce probability scores for binary classification, a thick layer with a sigmoid activation function was included.

Table 1: Transformation for data augmentation

Technique	Value
Rotation Range	20°
Width Shift Range	0.2
Height Shift Range	0.2
Shear Range	0.2
Zoom Range	0.2
Horizontal Flip	True
Vertical Flip	True

3.3.2. DenseNet169

The same method was used to refine DenseNet169 for deepfake detection. This model sought to extract more complicated features from the dataset by utilizing its deeper architecture. In order to improve performance and generalization, the fully connected layers were adjusted to fit the classification job, utilizing batch normalization, dropout, and dense layers.

3.3.3. Proposed Fusion Model

By combining the features taken from DenseNet121 and DenseNet169, a fusion model was produced that capitalized on the complementing advantages of both architectures. The fusion model's robustness and classification accuracy were enhanced by this fusion strategy, which enabled it to identify a wider variety of patterns in the dataset. Fig 2 illustrates how the fusion structure was created by combining two pretrained models.

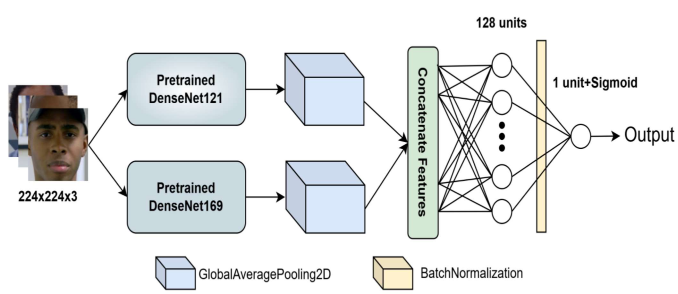


Figure 2: Feature-level fusion framework combining DenseNet121 and DenseNet169 models for enhanced classification.

4. Experimental Setup and Implementation

The study was conducted on a Kaggle environment utilizing a GPU accelerator with 30GB of RAM, and Python was the primary programming language employed for coding. To determine which combination of hyperparameters produces the greatest results on a validation set, we methodically explored a variety of values in our study. The chosen hyperparameters used in our investigation are displayed in Table 2.

5. Experimental Results and Analysis

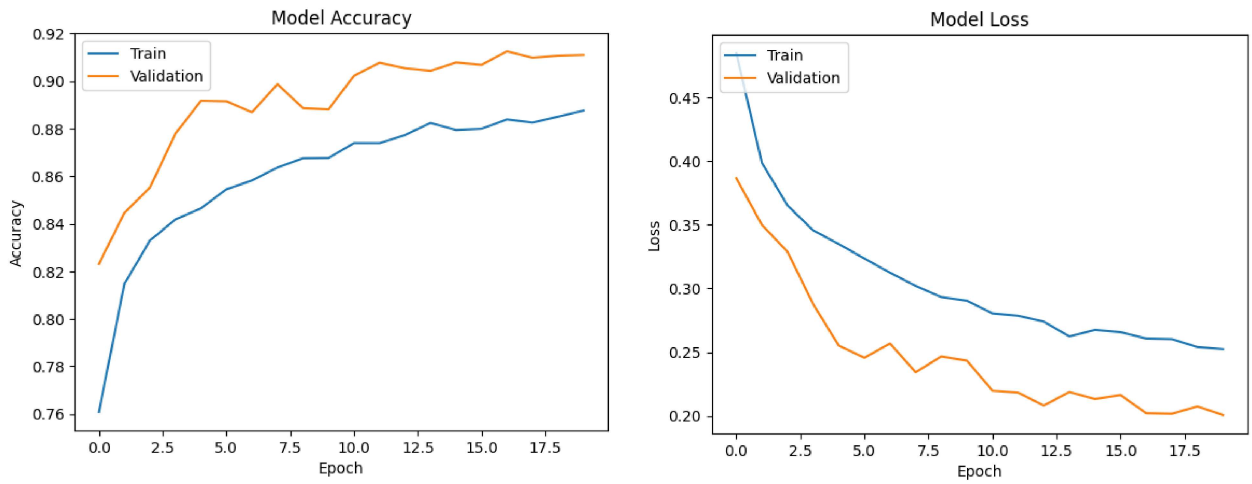
To evaluate the effectiveness of the proposed models, experiments were conducted using the enriched dataset. Together with the results of the separate models, DenseNet121 and DenseNet169, Table 3 also shows the results of some pretrained models.

Table 2: Parameters used in the pre-trained deep learning models

Parameter	**Value**
batch size	32
number of epoch	20
optimizer	adam
learning rate	0.0001
output classifier layer	sigmoid
activation function	relu

Table 3: Experimental Results

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 (%)**	**Error rate**	**AUC**
DenseNet121	89.53	90.69	91.28	90.99	0.1048	0.97
DenseNet169	89.75	**92.01**	90.13	91.06	0.1025	0.97
ResNet50	58.01	57.99	**99.83**	73.37	0.4199	0.71
Xception	88.70	89.40	91.40	90.40	0.1130	0.96
VGG16	85.25	82.26	95.05	88.19	0.1475	0.94
VGG19	80.85	88.05	77.47	82.46	0.1915	0.90
NASNetMobile	82.73	80.31	92.99	86.19	0.1727	0.92
InceptionV3	87.88	87.92	87.13	87.44	0.1212	0.95
DenseNet121+DenseNet169	**91.22**	90.82	94.37	**92.56**	**0.0878**	**0.98**



(a) Accuracy Curve

(b) Loss Curve

Figure 3: Accuracy and Loss Curve for Proposed Model.

Summary—The fused model (DenseNet121+DenseNet169) reaches 91.22% accuracy and 92.56% F1-score, improving over the best single backbone (DenseNet169: 89.75% / 91.06%). The error rate drops from 0.1025 to 0.0878 ($a \approx b$ 14.4% relative), and AUC rises to 0.98 (Table 3).

Confusion matrices—DenseNet121 tends to produce more false positives on *real*, while DenseNet169 shows more false negatives on *fake*; the fusion balances both (Figs. 5 and 6).

Learning curves—Training and validation curves track closely without late-epoch divergence (Fig. 3).

Per-class report—Class-wise scores are close (macro/weighted F1 = 0.91; Table 4). The ROC is steep with AUC = 0.98 (Fig. 4).

The findings demonstrate that the proposed model outperformed the individual models, achieving the highest accuracy and F1-score. Even though DenseNet121 and DenseNet169 performed well in classification when used separately. The confusion matrix, displayed in Fig. 5, illustrates the classification capabilities of individual models, whereas Fig. 6 of the proposed model distinguishes between deepfake and real images.

Table 4: Classification Report of the Proposed Model

	Precision	**recall**	**f1-score**	**support**
real	0.91	0.94	0.93	6608
fake	0.92	0.87	0.89	4799
Accuracy			0.91	11407
macro avg	0.91	0.91	0.91	11407
weighted avg	0.91	0.91	0.91	11407

The accuracy and loss curves in Fig. 3 show how the proposed model converged throughout training and validation. Furthermore, Table 4 shows the Classification Report. The findings show that the proposed model performs better than the individual models on a number of evaluation parameters. Table 4 displays a comprehensive classification report for the proposed model. And Fig. 4 displays a ROC curve of the proposed model.

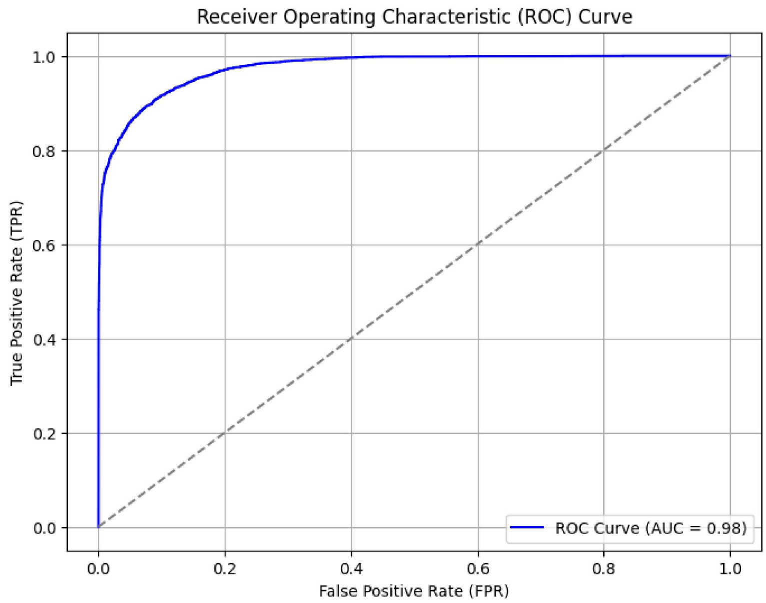


Figure 4: ROC Curve of Proposed Model

6. Model Interpretation

To enhance the transparency and interpretability of the proposed deepfake detection models, we employed the Local Interpretable Model-agnostic Explanations (LIME) framework. LIME is a post-hoc explainable AI (XAI) technique that approximates the behavior of a complex model in the vicinity of a given prediction using a simpler, interpretable model such as a linear classifier. This allows us to understand which regions of an input image contribute most strongly to the model's decision.

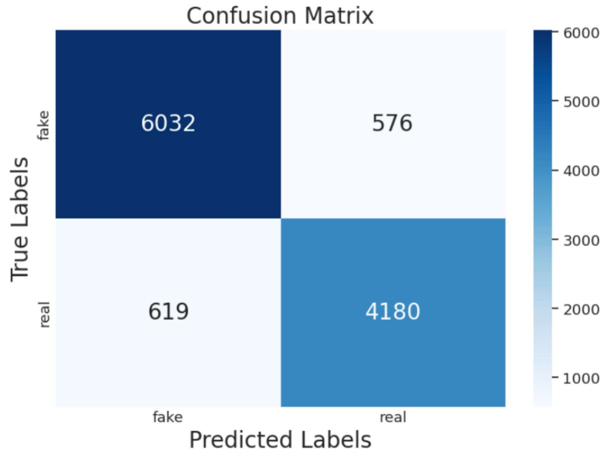
Formally, for a model f and an instance x, LIME seeks a locally interpretable model $g \in G$ (e.g., linear) that minimizes the following objective:

$$\mathcal{L}(f, g, \pi_x) = \arg\min_{g \in G} \sum_{z \in Z} \pi_x(z)\big(f(z) - g(z)\big)^2 + \Omega(g) \tag{1}$$

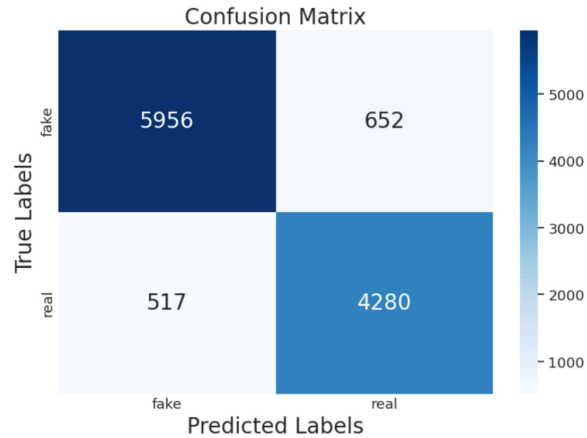
where Z is a set of perturbed samples around x, $\pi_x(z)$ is a proximity measure between x and z, and $\Omega(g)$ is a complexity term penalizing non-interpretable explanations.

In our study, LIME was applied to visualize the areas of facial images that the models considered most relevant for classification. Fig. 7 illustrates the output of LIME highlighting the most contributing areas to make the model declare it as a fake image. Even from a human inspection perspective, the marked portions should be the ones that indicates it is a fake. So, it is evident that the model is deciding based on the expected features. As a result, it is safe to say that model is learning effectively.

Overall, incorporating LIME into our analysis confirmed that the proposed framework makes decisions based on meaningful and explainable features, thereby increasing trust in the system and supporting its potential for real-world deployment.



(a) For DenseNet121 model



(b) For DenseNet169 model

Figure 5: Confusion Matrix of both models

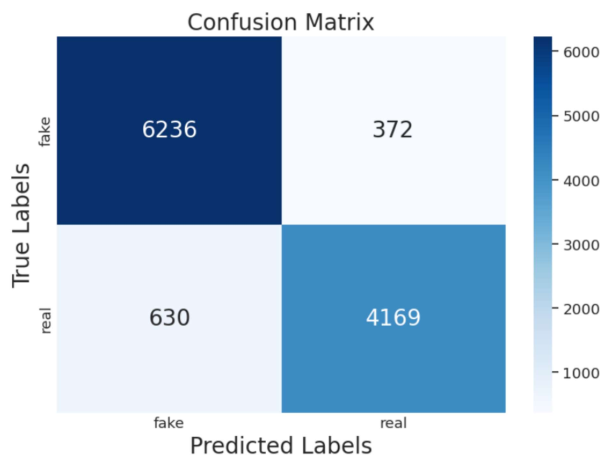


Figure 6: Confusion Matrix of proposed fusion model

7. Conclusion & Future Work

The usefulness of the system is shown by the assessment metrics for deepfake detection utilizing several models, such as the proposed fusion model. When compared to separate models, the proposed fusion model—which combines DenseNet121 and DenseNet169—achieved a high accuracy of 91% and improved precision, recall, and F1 score. These findings highlight how combining different architectures' feature extraction capabilities might improve classification performance. The proposed framework establishes a strong foundation for addressing deepfake detection challenges, though there remains significant potential for further development. Future research could explore several promising avenues, including the application of explainable AI techniques [10] to enhance model interpretability and reliability, and the design of lightweight convolutional neural network (CNN) architectures to improve scalability and real-time efficiency. The proposed framework has the potential to greatly progress the field of deepfake detection by tackling these issues, making it more dependable, effective, and accessible for a variety of applications.

Figure 7: Applying LIME on A Fake Image

Acknowledgments

The author thanks ChatGPT, an AI language model by OpenAI, for assistance in writing and refining parts of the manuscript.

References

[1] A. Kaur, A. N. Hoshyar, V. Saikrishna, S. Firmin, and F. Xia, "Deepfake video detection: challenges and opportunities," *Artificial Intelligence Review*, vol. 57, no. 6, pp. 1–47, 2024.

[2] C. Barnes and T. Barraclough, "Deepfakes and synthetic media," in *Emerging technologies and international security*, pp. 206–220, Routledge, 2020.

[3] Q. Zhao, X. Wang, W. Jiang, and Y. Xiong, "Deep learning for deepfake detection: A comprehensive review," *ACM TOMM*, vol. 17, no. 2, pp. 1–24, 2021.

[4] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in *2018 IEEE WIFS*, pp. 1–7.

[5] S. R. A. Ahmed and E. Sonuç, "Retracted article: Deepfake detection using rationale-augmented convolutional neural network," *Applied Nanoscience*, vol. 13, no. 2, pp. 1485–1493, 2023.

[6] A. Raza, K. Munir, and M. Almutairi, "A novel deep learning approach for deepfake image detection," *Applied Sciences*, vol. 12, no. 19, p. 9820, 2022.

[7] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *ICCV*, pp. 1–11, 2019.

[8] H. Zhang, J. Wu, S. Liu, and S. Han, "A pre-trained multi-representation fusion network for molecular property prediction," *Information Fusion*, vol. 103, p. 102092, 2024.

[9] A. Basaran, “Kaggle dataset.” `https://www.kaggle.com/datasets/abdurahimbaaran/deepfake-dataset`. Accessed: Dec. 31, 2024.

[10] R. Dwivedi, D. Dave, H. Naik, *et al.*, “Explainable AI (XAI): Core ideas, techniques, and solutions,” *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–33, 2023.