Received: xxxxxxxx | Revised: xxxxxxxx | Accepted: xxxxxxx

DOI: xxxxx/xxxxx
ORIGINAL RESEARCH

Video Frame Classification using Custom CNN: A Comparative Analysis with Optimal Model Deployment via Gradio Interface

Abir Mahmud Shahariar[1]

1 Department of Computer Science and Engineering, International Standard University, Dhaka, Bangladesh

Sanjana Islam Kasfia[2]

2 Department of Computer Science and Engineering, International Standard University, Dhaka, Bangladesh

Hasibul Islam Peyal[3]

3 Department of Computer Science and Engineering, International Standard University, Dhaka, Bangladesh

Abstract

The rapid growth of deepfake technology has become a serious cybersecurity risk, allowing the creation of harmful, very realistic fake media that can trick both smart systems and people. This tendency raises crucial considerations regarding online misinformation, social engineering, and identity theft. This research introduces a deepfake recognition system for video frame-level classification utilizing a custom Convolutional Neural Network (CNN). A readily accessible Kaggle dataset featuring more than 16,000 face-cropped frames, utilizing Celeb-DF and FaceForensics++, was used to train the model. To appraise its efficacy, we undertook an evaluation of four prominent pretrained models: MobileNetV2, ResNet50, VGG19, and DenseNet121. The proposed custom CNN achieved a superior precision of 92.4%, surpassing ResNet50 (91%), VGG19 (89%), DenseNet121 (83%), and MobileNetV2 (73%). For practical implementation, the model was incorporated into a Gradio-based web interface for real-time inference. The findings demonstrate the viability of lightweight, task-based CNNs in mitigating deepfake threats and establishing reliable, user-oriented multimedia fidelity assessment systems.

Keywords— Video Frame Classification, Deepfake Detection, FaceForensics++, Synthetic Media Forensics, Convolutional Neural Network (CNN), Gradio Interface, Computer Vision, Deep Learning, Image Authenticity

1 | INTRODUCTION

Prior to the widespread integration of Artificial Intelligence (AI) and Machine Learning (ML), early developments in deepfake technology were achieved through a combination of conventional image processing, computer graphics, and video editing tools, rather than data-driven learning models. Software such as Adobe After Effects, Elastic Reality, and FaceGen enabled individuals to perform frame-by-frame manipulations, relying heavily on manual input and traditional algorithms. These systems utilized a variety of techniques, including Lucas-Kanade tracking and optical flow for motion stabilization, masking and rotoscoping for facial region segmentation, and Delaunay triangulation with mesh warping for facial morphing. Color accuracy and seamless blending were often achieved using alpha blending and histogram equalization, while image relighting and 3D texture mapping enhanced realism. Despite the absence of AI-driven automation, these methods demonstrated considerable sophistication through their reliance on low-level image manipulation, affine transformations, and geometry-based modeling. However, with the advent of generative AI models—particularly autoencoders and Generative Adversarial Networks (GANs)—deepfakes have evolved into highly realistic, automatically generated media, significantly increasing the difficulty of distinguishing them from authentic content.

As deepfakes make it increasingly difficult to distinguish between fact and fiction, this pervasive trend has raised significant concerns regarding the legitimacy and reliability of digital visual content. These artificial videos are now rendered with hyper-realistic accuracy, owing to advanced deep learning techniques, often making them indistinguishable to the human eye. Consequently, there is a growing demand for effective and scalable detection systems that can address the

limitations of current approaches in terms of accuracy, latency, and scalability for practical deployment [1]. Reliable and efficient classification methods are becoming increasingly vital as online video content continues to grow at an exponential rate. However, large datasets often introduce noise and computational challenges [2]. Using a face-cropped video frame dataset, our research addresses this issue by developing a simple yet effective frame-level identification technique for real-time deepfake detection. Like other computer vision frame-level classification tasks, deepfake recognition must account for visual diversity and frame-to-frame noise. Under such conditions, traditional frameworks often struggle to make accurate inferences, particularly when dependent on handcrafted features [3][4][5]. Instead, our approach leverages a deep learning framework designed to learn robust spatial patterns directly from the data, enhancing its resilience to such inconsistencies. We also acknowledge that not every frame in the Kaggle face-cropped video frame dataset contributes equally to classification accuracy, due to potential issues like motion blur, low resolution, or occlusion. Future improvements could benefit from incorporating a frame-filtering mechanism to exclude inadequate or non-informative inputs.

In this study, we aim to assess and compare several Convolutional Neural Network (CNN) frameworks for deepfake detection at the video frame level. Alongside a custom-designed lightweight CNN, we implemented and evaluated several prominent pretrained models, including ResNet50, VGG19, MobileNetV2, and DenseNet121, using a consistent dataset comprising over 16,000 face-cropped frames sourced from the FaceForensics++ and Celeb-DF datasets. All models were evaluated under identical experimental conditions to ensure fairness and objectivity in the assessment of results.

Inspired by the growing concern over deepfakes and the lack of easily accessible yet reliable detection technologies, our study aims to address existing gaps through the following key contributions:

(i) Development of a custom CNN model that offers competitive accuracy while remaining computationally efficient,

(ii) A comparative analysis of widely recognized CNN architectures for effective deepfake image classification

(iii) Deployment of the proposed system via a Gradio-based web interface, enabling real-time, user-oriented interaction.

The remaining sections of this paper are organized as follows: Section 2 outlines the methodology, including dataset collection and splitting, data augmentation, model architecture, and training procedures. It also presents the Gradio interface used to deploy the model with the best performance. Section 3 details the comparative analysis and experimental findings. Section 4 discusses related work and contextualizes our results within existing research. Finally, Section 5 concludes the study and suggests future directions for deepfake detection and multimedia forensics.

2 | METHODOLOGIES

In this study, we developed a custom CNN model designed to classify whether a video frame is real or fake. Figure 1 illustrates the overall procedure employed in our research.

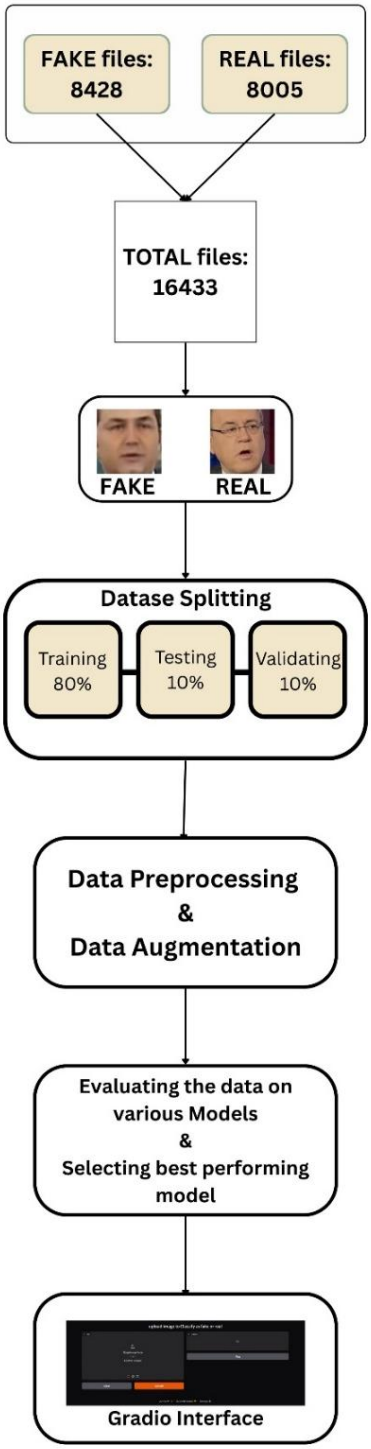


Fig. 1. Working Procedure

2.1 | Dataset Collection and Splitting

All face-cropped frames were sourced from a publicly available Kaggle dataset containing pre-extracted images from two benchmark video forensics datasets: FaceForensics++ and Celeb-DF [6]. The frames are categorized into two classes:

Real: 8,005 images cropped from 200 real videos

Fake: 8,428 images cropped from 200 deepfake videos

In total, the dataset comprises 16,433 images, with a relatively balanced distribution between real and fake classes. To prevent data leakage, the train/test split was performed at the video level—320 videos (160 real, 160 fake) were allocated for training, and 80 videos (40 real, 40 fake) for testing. This results in:

Training set: 13,146 frames

Validation set: 1,644 frames

Test set: 1,644 frames

An 80:20 split ratio was used to ensure robust evaluation while maintaining sufficient diversity for effective model training. Representative samples from both classes are shown in Figure 2.

2.2 | Data Augmentation

To enhance model robustness and generalization, on-the-fly data augmentation was applied exclusively to the training set. This technique effectively increases the size and diversity of training samples, helping prevent overfitting and improving the model's performance on unseen data. Before augmentation, all images were resized to 100×100 pixels with three color channels. During training, 32 images were processed per batch.

The following transformations were applied using Keras's ImageDataGenerator:

(i) Horizontal Flip: Randomly flips images along the vertical axis, simulating variations in left-right facial orientation.

(ii) Vertical Flip: Randomly flips images along the horizontal axis, introducing top-bottom variations.

In this study, no additional geometric distortions—such as rotations, shifts, shears, or zooms—were applied. The use of vertical and horizontal flips alone provided adequate variability in facial orientations across the frame-level dataset.

2.3 | Website Work Procedure

The core purpose of the developed website is to present deepfake detection capabilities to users through an interactive interface. Figure 3 consists the entire working procedure of the application which allows users to upload a facial image and receive real-time classification results. The interface comprises three main components: the Upload Panel, Submit Button, and Output Panel, which is illustrated in the Figure 4. The web application is built using Gradio's Interface module and integrates the trained custom CNN model.

Fig. 2. Real & Fake Image

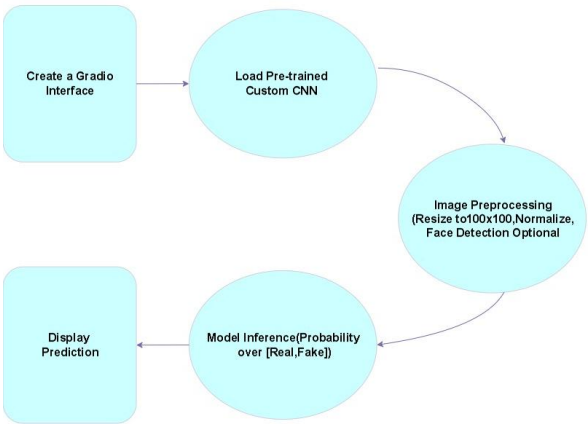


Fig.3. Website's Working Procedure

In the backend, the pre-trained model is loaded into memory using TensorFlow. This is followed by image preprocessing, model inference, and output display. The system delivers results within approximately one second per image, providing users with immediate feedback.

The application can be run locally or deployed on platforms such as Hugging Face Spaces, Render, or Heroku. Although the lightweight model is capable of running on CPUs, GPU usage is recommended to further accelerate inference times.

3 | RESULTS

3.1 | Quantitative Analysis

1. Training, Validation, and Test Performance:

Table 1: Evaluation of Proposed Model on Training, Validation, and Test Data

Dataset	Accuracy (%)	Loss
Training	94	0.2041
Validation	92.45	0.236
Test	92.88	0.2448

The custom CNN architecture was trained on 12,800 face-cropped frames and validated/tested on 3,200 unseen frames. The model showed good convergence across all training and validation datasets, with only a minimal degree of overfitting.

Fig. 4. Website's User Interface

The final losses and accuracies for each dataset split (training, validation, and test) are summarized in below from Table 1.

(i) Training Loss & Accuracy: The training loss converged to 0.2041, with an accuracy of 94.00%.

(ii) Validation Loss & Accuracy: The validation loss stabilized at 0.2360, and the validation accuracy remained steady at 92.45%, indicating minimal overfitting.

(iii) Test Loss & Accuracy: The test loss was 0.2448, with an accuracy of 92.88% on unseen data, demonstrating strong generalization of the model.

2. Confusion Matrix Analysis:

Figure 7 displays the confusion matrix, illustrating the distribution of the model's predictions on the test set of 1,644 face-cropped frames. A high number of true positives and true negatives suggests that the model effectively distinguishes between real and manipulated frames.

The presence of errors is balanced, and the dominance of entries along the diagonal confirms strong class discrimination with minimal bias. This indicates the model's ability to accurately classify both real and fake images with few misclassifications

As shown in the charts, the custom CNN's training and validation performance across more than 200 epochs demonstrates a steady and effective learning process.

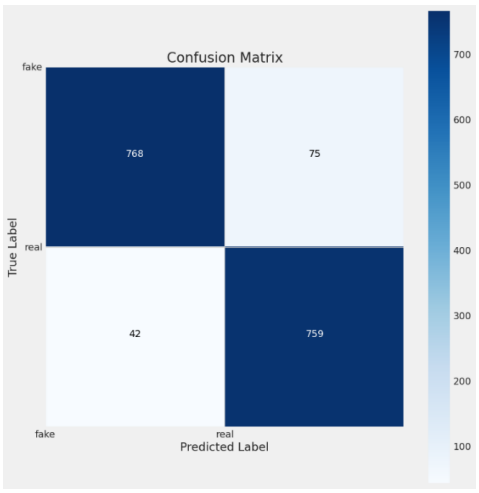
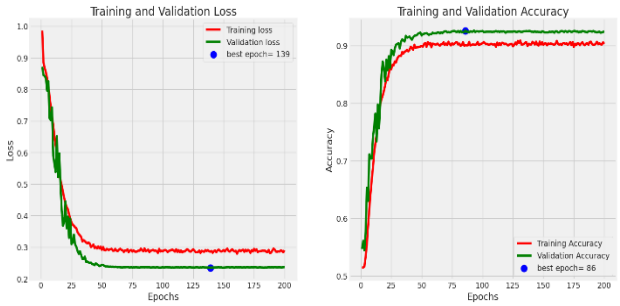


Fig. 7. Confusion Matrix for Proposed CNN

3. Training and Validation Performance:



(A) **(B)**

Fig. 8. Training and Validation Loss/Accuracy for Proposed CNN

In the Figure 8 (A),the left subplot reveals that the training and validation losses consistently decrease during the early epochs, indicating effective tuning. Around epoch 50, the training loss begins to plateau but continues to decrease steadily. Notably, the blue pointer marks the point at which validation loss stabilizes at epoch 50 and reaches its minimum at epoch 139. Since the gap between the training and validation losses remains small and constant, this trend suggests that the model maintains substantial adaptability without overfitting.

On the other hand, in the Figure 8(B),the right subplot illustrates an abrupt increase in accuracy during the initial training phase. While training accuracy steadily exceeds 90%, validation accuracy spikes at epoch 86. This shows that the system is well regularized and can acquire discriminative features, as it continuously performs better on the validation set than on the training data. The model's robustness is further evidenced by the consistent convergence of the accuracy curves.

4. Classification Report:

To further assess the model's performance, we calculated the precision, recall, and F1-score for both classes. These metrics offer a more nuanced evaluation beyond overall accuracy.

Table 2: Classification Report for Proposed CNN

Class	Precision	Recall	F1-Score	Support
Fake	0.95	0.91	0.93	843
Real	0.91	0.95	0.93	801
Accuracy	-	-	0.93	1644
Macro Avg	0.93	0.93	0.93	1644
Weighted Avg	0.93	0.93	0.93	1644

Table 2 presents the classification performance of the proposed CNN model in distinguishing between fake and real images. The model achieved a precision of 0.95 for fake images, indicating that 95% of the images predicted as fake were correctly classified, while the recall of 0.91 shows that 91% of all actual fake images were successfully identified. For real images, the model yielded a precision of 0.91 and a recall of 0.95, demonstrating balanced performance across both classes. The F1-score for both classes is 0.93, reflecting a strong harmonic mean of precision and recall. The overall accuracy of the model is 93% on a test set of 1,644 images. Both the macro average (which treats all classes equally) and the weighted average (which accounts for class imbalance) of precision, recall, and F1-score are also 0.93, indicating consistent and reliable performance across the dataset.

4 | COMPARATIVE ANALYSIS

Table 3 provides a comparative analysis of the proposed custom CNN model against popular pre-trained models—MobileNetV2, DenseNet121, VGG19, and ResNet50—in terms of classification performance and computational efficiency. The proposed model achieved the highest accuracy of 92.4% and an F1-score of 93, outperforming all other models. While ResNet50 came close with 91% accuracy and a 92 F1-score, it required significantly more parameters (23.5 million) and longer training time per epoch (52 seconds). In contrast, the proposed CNN demonstrated superior efficiency, requiring only 1.25 million parameters and just 18 seconds per epoch, making it the most lightweight and computationally efficient model in the comparison. This highlights the effectiveness of the proposed model not only in terms of predictive performance but also in reducing computational cost, making it highly suitable for real-world deployment, especially in resource-constrained environments.

5 | CONCLUSION

This research proposes a custom-designed Convolutional Neural Network (CNN) for video frame deepfake detection, aiming to establish an efficient and effective framework. The proposed approach outperforms well-known pre-trained models such as ResNet50, VGG19, DenseNet121, and MobileNetV2, achieving an impressive classification accuracy of 92.4% using a face-cropped frame dataset sourced from reputable benchmarks—FaceForensics++ and Celeb-DF. While pre-trained models like VGG19 and ResNet50 yielded competitive results, our custom architecture surpassed them in both efficiency and precision. Importantly, this work bridges the gap between theoretical research and practical application by deploying the trained model within an interactive Gradio-based interface, ensuring broad accessibility and enabling real-time detection of manipulated media.

Despite its strong performance, the model still has room for improvement—particularly in terms of the dataset's size and diversity. A limited dataset may hinder the model's ability to generalize effectively to more complex, subtle, or previously unseen types of deepfake manipulations. This can affect performance when applied to real-world scenarios where deepfakes vary significantly in quality, style, and context. Expanding the dataset to include a broader range of manipulation techniques, lighting conditions, facial expressions, and background variations would enhance the model's robustness and adaptability. Additionally, rigorous testing across more realistic and diverse environments would improve the model's reliability and ensure its practical effectiveness in real-world deepfake detection tasks.

Future work could explore ensemble methods, transformer-based architectures, and Explainable AI (XAI) tools like SHAP and LIME to enhance transparency. Additionally, upgrading the user interface, enabling video-level detection, and applying more diverse data augmentations (e.g., shifts, rotations, zooms) could boost accuracy and robustness. Overall, this study lays the groundwork for efficient and adaptable deepfake detection tools.

Table 3: Comparative Results of Model Performance and Computational Complexity

Model	MobileNetV2	DenseNet121	VGG19	ResNet50	Proposed Custom CNN
Accuracy	73	83	89	91	92.4
F1-score	72	84	89	92	93
Time Per Epoch (s)	34	49	63	52	18
Parameter (M)	3.4	8.1	143	23.5	1.25

REFERENCES

[1] D. L. T. Bale, L. C. Ochei, and C. Ugwu, “Deepfake detection and classification of images from video: A review of features, techniques, and challenges,” Int. J. Intell. Inf. Syst., vol. 13, no. 2, Apr. 2024. [Online]. Available: https://doi.org/10.11648/j.ijiis.20241302.11

[2] A. Karpenko and P. Aarabi, “Tiny videos: A large data set for nonparametric video retrieval and frame classification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 3, pp. 518–527, May 2011, doi: 10.1109/TPAMI.2010.118.

[3] S. Manivannan, R. Wang, E. Trucco, and A. Hood, “Automatic normal-abnormal video frame classification for colonoscopy,” in Proc. 2013 IEEE 10th Int. Symp. Biomed. Imaging: From Nano to Macro, San Francisco, CA, USA, 2013, pp. 644–647.

[4] J. Oh, S. Hwang, J. Lee, W. Tavanapong, J. Wong, and P. C. de Groen, “Informative frame classification for endoscopy video,” Med. Image Anal., vol. 11, no. 2, pp. 110–127, Apr. 2007, doi: 10.1016/j.media.2006.10.003.

[5] N. Rangseekajee and S. Phongsuphap, “Endoscopy video frame classification using edge-based information analysis,” in Proc. 2011 Comput. Cardiol., Hangzhou, China, 2011, pp. 549–552.

[6] J. Yohannan and N. Sasikumar, “Face Forensic++ & Celeb-DF Combined Deepfake Data,” Kaggle. [Online]. Available: https://www.kaggle.com/datasets/nanduncs/1000-videos-split/data [Accessed: May 5, 2025].