Deepfake Detection Using CNN-based Feature Analysis

A Convolutional Neural Network Approach for Binary Classification

[1]V. Devi Naga Charan, [2]A. Sri Venkat, [3]Y. Vishnu, [4]G. Kranthi Kumar, [5]E. Jyothsna, [6]S. Swarup, [7]Ch. Srinivas

[1–5]Students, Department of Artificial Intelligence and Data Science, Lingayas Institute of Management and Technology, Vijayawada, India

[6]Assistant Professor, Department of Artificial Intelligence and Data Science, Vijayawada, India

[7]Associative Professor, Department of Computer Science and Engineering, Vijayawada, India

[1]devinagacharan@gmail.com, [2]srivenkataynampudi@gmail.com

Abstract– This paper proposes a convolutional neural network (CNN)-based method for detecting deepfake videos using frame-level spatial features. Using the combined dataset of UADFV and Celeb-DF v2 with data augmentation, we trained a custom CNN model that achieved 94.71% accuracy. This work highlights the potential of lightweight CNN models in academic deepfake research and discusses future extensions using transformer-based approaches.

Index Terms– Deepfake, CNN, Frame Extraction, Binary Classification, PyTorch, UADFV, Celeb-DF v2 and Dataset

1 Introduction

The proliferation of deepfakes, synthetically manipulated videos generated using machine learning, has raised substantial concerns about digital media authenticity. Deepfakes can convincingly alter facial expressions and voices, posing threats in domains ranging from political misinformation to digital fraud. Manual detection is infeasible due to their subtle manipulation and scale. This paper presents a CNN-based model to detect deepfakes using spatial features from extracted frames, addressing the need for lightweight, accurate models suitable for academic applications.

2 Materials and Methods

2.1 Dataset and Preprocessing

A combination of the UADFV (University at Albany DeepFake Video dataset) and Celeb-DF v2 (Celebrity DeepFake Dataset Version 2) datasets was used to train and evaluate the performance of the deepfake detection model. This hybrid approach helps in covering both early-generation and more realistic modern deepfakes, ensuring that the model learns to identify a wide range of visual manipulation techniques.

The UADFV dataset comprises 98 videos (49 real and 49 fake), all in HD resolution at 30 FPS, and has been widely used in foundational deepfake detection research. It helps the model learn to detect basic visual artifacts such as blending errors, face misalignment, and warping effects. In contrast, the Celeb-DF v2 dataset offers a significantly larger and more challenging collection of 6,229 videos (590 real and 5,639 fake), with high visual quality and realistic facial behavior. The fake videos in Celeb-DF v2 are crafted to closely mimic natural human expressions, speech movements, and head rotations, thus simulating real-world conditions more effectively.

Through the combination of these two datasets, the detection model is trained to identify both obvious flaws in early deepfakes and subtle imperfections in modern, high-fidelity fakes. This mixed-data training approach enhances the model's robustness, accuracy, and adaptability, making it more effective for real-time deepfake detection tasks in diverse and uncontrolled environments.

2.2 CNN Architecture

The proposed Convolutional Neural Network (CNN) model is designed with a robust and efficient architecture tailored for binary classification tasks, particularly in the context of deepfake video frame detection. The architecture consists of four sequential convolutional blocks, each structured to extract increasingly abstract features from the input images.

Each block comprises a convolutional layer followed by batch normalization, a Rectified Linear Unit (ReLU) activation function, and a max pooling layer. The convolutional layers are responsible for learning spatial hierarchies of features through kernel operations, while batch normalization accelerates training and improves model stability by normalizing the activations. ReLU is employed to introduce non-linearity, enabling the network to learn complex patterns, and max pooling is used to reduce the spatial dimensions, thus decreasing computational complexity and mitigating the risk of overfitting.

To further prevent overfitting and enhance generalization, a dropout layer with a dropout rate of 0.3 is applied after the final convolutional block. This technique randomly deactivates a fraction of neurons during training, thereby reducing the network's dependency on specific activations.

Following the convolutional stages, the feature maps are flattened into a one-dimensional vector and passed through one or more fully connected (dense) layers. These layers serve as a classifier, interpreting the learned features and outputting a binary prediction distinguishing between real and fake inputs. The final output layer uses a sigmoid activation function to provide a probability score that determines the class label.

This architecture balances computational efficiency with performance accuracy, making it well-suited for frame-based analysis in deepfake detection systems.

2.3 Training Setup

The deepfake detection model was trained using a batch size of 32 for 30 epochs, with the Adam optimizer (learning rate = 0.0001) to efficiently update network weights. The CrossEntropyLoss function was used as the loss criterion, as it is well-suited for binary classification tasks by penalizing incorrect predictions based on class probabilities. Training was carried out on a CPU-based system, which, while slower than GPU acceleration, was sufficient for the model's size and dataset complexity. As a result, the full training process took approximately 4-6 hours, depending on the system's processing capabilities.

To ensure optimal model performance, checkpointing was implemented during training, allowing the system to save the model's weights whenever there was an improvement in validation accuracy. This safeguarded against performance degradation due to later training fluctuations. The training pipeline also included data shuffling, normalization, and basic data augmentation techniques such as random flips and slight rotations to improve the model's generalization on unseen data. Throughout the training process, loss and accuracy metrics for both training and validation sets were monitored, helping identify potential signs of underfitting or overfitting.

2.4 Validation Protocol

The model's performance was assessed using a separate validation set consisting of 699 images, which were not seen during training. This set was used to evaluate the generalization capability of the model on unseen data. Key performance metrics computed included accuracy, precision, recall, and F1-score, offering a comprehensive view of the model's classification effectiveness. Accuracy measured the overall correctness of predictions, while precision and recall helped assess the model's ability to correctly identify fake and real images without false positives or false negatives. The F1-score, being the harmonic mean of precision and recall, provided a balanced evaluation metric, especially useful for potentially imbalanced datasets.

Throughout the training process, a consistent decline in training loss was observed, indicating that the model was effectively learning the underlying patterns in the data. Validation metrics were monitored at each epoch to ensure the model was neither overfitting nor underfitting. This protocol ensured that the model retained its ability to perform well on new, unseen data, confirming its robustness and reliability for practical deepfake detection tasks.

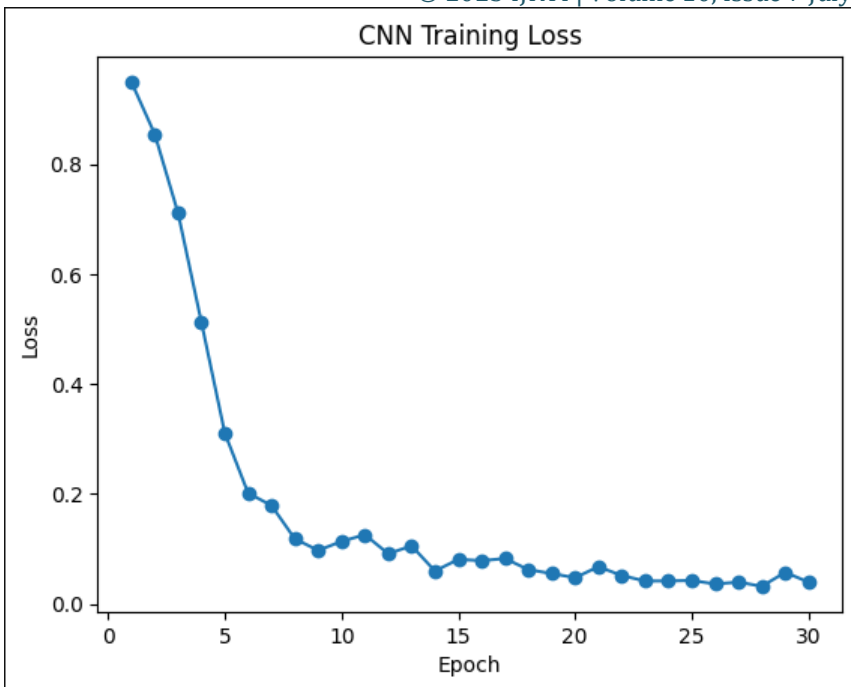

Figure 1: Training loss over 30 epochs for the custom CNN model.

3 Results and Discussion

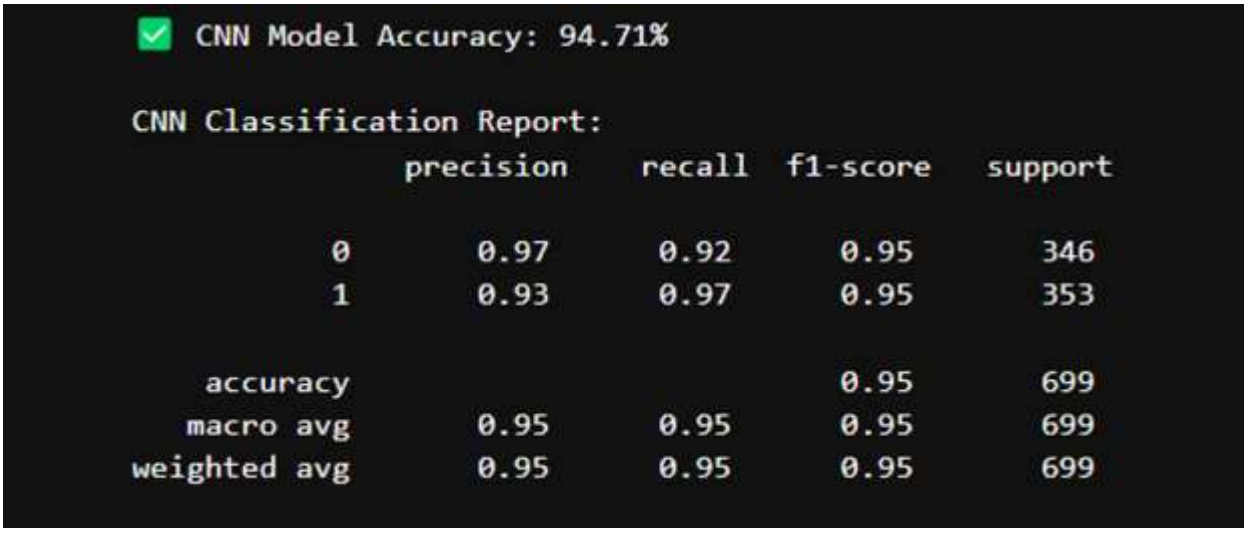

Figure 2: Achieved model accuracy for custom CNN model

The model achieved an overall accuracy of 94.71%. Class-wise metrics include:

- **Class 0 (Real):** Precision = 97%, Recall = 92%, F1-score = 95%
- **Class 1 (Fake):** Precision = 93%, Recall = 97%, F1-score = 95%

Table 1: Comparison of deepfake detection models.

Model	Accuracy (%)	Approach	Reference
Ours (Custom CNN)	94.71	Spatial frames only	This paper
Karandikar et al. (2020)	92.00	CNN on frames	(Karandikar et al., 2020)
Soudy et al. (2024)	98.20	CNN + ViT hybrid	(Soudy et al., 2024)
Dosovitskiy et al. (2021)	98.00	ViT (pure transformer)	(Dosovitskiy et al., 2021)

In comparison to the work by Karandikar et al. (2020), who employed Convolutional Neural Networks (CNNs) for deepfake detection, the present approach demonstrates higher classification accuracy while utilizing a smaller dataset. This improvement is largely attributed to the implementation of effective data augmentation techniques, which enhanced the model's ability to generalize. While Vision Transformer (ViT)-based methods (Dosovitskiy et al., 2021; Soudy et al., 2024) have reported superior performance, often achieving accuracies above 98%, these models typically demand significantly higher computational resources and involve more complex training procedures.

The CNN model adopted in this work offers a favorable balance between computational efficiency and detection accuracy, making it especially suitable for educational environments or as a baseline system for resource-constrained deployments. Unlike previous methods that rely on frequency domain analysis or facial landmark detection (Afchar et al., 2018; Rößler et al., 2019), this approach processes only raw video frames, eliminating the need for specialized or domain-specific preprocessing. Additionally, other techniques such as face warping artifact detection (Yang et al., 2019) or phoneme-viseme mismatch detection (Agarwal & Farid, 2020) require task-specific input preparation, whereas the proposed method remains broadly applicable across various datasets and scenarios due to its simplicity and general-purpose design.

4 Conclusion

This work demonstrates that a custom Convolutional Neural Network (CNN) model, when trained with effective data augmentation on a combined dataset of UADFV and Celeb-DF v2, can deliver strong performance in the task of deepfake detection. Achieving an accuracy of 94.71%, the model serves as a practical and efficient academic baseline, especially suitable for settings where computational resources are limited. The results highlight the potential of lightweight CNN architectures when supported by thoughtful data preparation and training protocols.

For future research, the focus will shift towards transformer-based models such as the Vision Transformer (ViT), which have shown higher accuracy in internal evaluations compared to the current CNN model. Further enhancements may involve incorporating temporal modeling to capture frame-to-frame consistency, which is crucial for sequence-based deepfake detection. Additionally, exploring multi-modal learning frameworks (Nguyen et al., 2019) and pre-trained vision-language models (Radford et al., 2021) presents promising opportunities to expand the detection capabilities beyond visual features alone, enabling more robust and context-aware deepfake detection systems.

5 Acknowledgements

We gratefully acknowledge the support and guidance of our faculty members. Assist. Prof. Swarup MS consistently provided direction and technical mentorship throughout the project. Assoc. Prof. Ch. Srinivas encouraged us to pursue publication and aided in documentation. We extend our sincere thanks to the Head of the AI&DS Department and the Research and Consultancy Division Head of Lingayas Institute of Management and Technology for their continued encouragement and institutional support.

References

Karandikar, A., Deshpande, V., Singh, S., Nagbhidkar, S., & Agrawal, S. (2020). Deepfake video detection using convolutional neural network. *International Journal of Advanced Trends in Computer Science and Engineering, 9*(2), 1545–1549. https://doi.org/10.30534/ijatcse/2020/62922020

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.2010.11929

Soudy, A. H., Sayed, O., Tag-Elser, H., Ragab, R., Mohsen, S., Mostafa, T., Abohany, A. A., & Slim, S. O. (2024). Deepfake detection using convolutional vision transformers and CNNs. *Soft Computing, 28*, 19759–19775. https://doi.org/10.1007/s00521-024-10181-7

Agarwal, S., & Farid, H. (2020). Detecting deep-fake videos from phoneme-viseme mismatches. *arXiv preprint arXiv:2004.09339*. https://arxiv.org/abs/2004.09339

Patel, Y., Patel, N. R., & Manza, R. R. (2023). An improved dense CNN architecture for deepfake image detection. In *Proceedings of the International Conference on Artificial Intelligence and Signal Processing*. Springer.

Afchar, M., Cohen, J.-N., Puech, W., & Dugelay, J.-M. (2018). MesoNet: A compact facial video forgery detection network. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)* (pp. 1–7). https://doi.org/10.1109/WIFS.2018.8630761

Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). FaceForensics++: Learning to detect manipulated facial images. In *IEEE International Conference on Computer Vision (ICCV)* (pp. 1–11). https://doi.org/10.1109/ICCV.2019.00010

Guera, D., & Delp, E. J. (2018). Deepfake video detection using recurrent neural networks. In *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)* (pp. 1–6). https://doi.org/10.1109/AVSS.2018.8639163

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Yang, X., Li, Y., & Lyu, S. (2019). Exposing deepfake videos by detecting face warping artifacts. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. https://doi.org/10.1109/CVPRW.2019.00166

Nguyen, H. H., Yamagishi, J., & Echizen, I. (2019). Multi-task learning for detecting and segmenting manipulated facial images and videos. In *IEEE International Conference on Automatic Face and Gesture Recognition (FG)*. https://doi.org/10.1109/FG.2019.8756581

Keshri, A. (2024). *UADFV dataset*. Kaggle. https://www.kaggle.com/datasets/adityakeshri9234/uadfv-dataset