Novel solution for Deepfake Image Detection using CNN and ResNet50 Architecture

Daksh Sanghvi
Thadomal Shahani Engineering College, Mumbai

Vansh Solanki
Thadomal Shahani Engineering College, Mumbai

Akhilesh Sonarikar
Thadomal Shahani Engineering College, Mumbai

Rishabh Jaiswal
Thadomal Shahani Engineering College, Mumbai

ABSTRACT

The proliferation of deepfake technology has raised significant concerns regarding the manipulation and authenticity of digital media. Addressing the urgent need for reliable detection methods, this research explores the application of deep learning techniques in identifying image-based deepfakes.[12] Specifically, this study delves into the utilization of convolutional neural networks (CNNs) and ResNet50, a residual neural network architecture, for discerning manipulated visual content.[3]

The research methodology involves training and evaluating these deep learning models on diverse datasets comprising authentic and manipulated images. Through the implementation of transfer learning, the pre-trained ResNet50 model is fine-tuned on a deepfake-specific dataset to enhance its capacity for accurate detection.

Key factors influencing the efficacy of these detection methods, such as dataset size, model architecture, and training parameters, are thoroughly analyzed and discussed. Evaluation metrics encompassing accuracy, precision, recall, and F1 scores are employed to assess the performance of the models in differentiating between real and deepfake images.

The findings underscore the robustness of ResNet50 and CNN-based approaches in detecting image-based deepfakes, exhibiting promising results in identifying manipulated content across various contexts. Furthermore, insights into the limitations and potential areas for improvement in deepfake detection using these methodologies are presented, paving the way for future research endeavors in this critical domain.

Keywords:

Deepfake Detection, ResNet50, CNN, Deep learning

1. INTRODUCTION

In a digital era dominated by rapid technological advancements, the rise of deepfakes has sparked a cat-and-mouse game between creators and detectors. Deepfakes, convincingly manipulated multimedia content often fueled by powerful neural networks, have infiltrated the online landscape, posing challenges to the authenticity of visual information.

Imagine you have a tool that can take a video or an image and make it seem like someone else is saying or doing things they never actually did. That's essentially what a deepfake is. A deepfake is a type of manipulated media where artificial intelligence, specifically deep learning algorithms, are used to alter or replace images or videos to make them appear convincingly real.[10] It can be used to superimpose someone's face onto another person's body in a video or make them say things they never said. Essentially, it's a way of creating fake content that looks incredibly authentic, making it hard to tell if what you're seeing or hearing is genuine or manipulated. These technologies are powerful because they can analyze patterns in images or videos, learning how a person looks or speaks, and then use that information to create incredibly realistic forgeries.

The boost of deepfake technology has introduced a myriad of real world negative effects, prompting concerns across various domains [9]. Misinformation campaigns leveraging deepfakes have the potential to sow confusion and manipulate public opinion, posing a direct threat to the democratic process and the integrity of information. The risk of identity theft and privacy violations has escalated, as individuals may fall victim to convincing impersonations, leading to reputational damage and emotional distress. Beyond personal implications, deepfakes contribute to the erosion of trust in media, making it increasingly challenging for individuals to discern between authentic and manipulated content. Businesses and organizations face the risk of malicious actors exploiting deepfakes for social engineering attacks, further jeopardizing cybersecurity. Additionally, the potential weaponization of deepfakes in international relations poses a global security threat, with the technology being harnessed for political manipulation and destabilization.

As we dive into the intricacies of our study, we embark on a mission to dissect the efficacy of CNN and ResNet50 in the realm of deepfake detection. These two stalwart architectures, each bearing its unique prowess, hold the promise of unmasking the virtual masquerade orchestrated by the creators of synthetic media. In our pursuit of clarity amidst the digital fog, we will traverse through datasets teeming with manipulated content, scrutinizing the performance of CNN and ResNet50 under the relentless gaze of diverse evaluation metrics. It's a journey that takes us beyond the lines of code, exploring the practical implications of our findings and paving the way for future advancements in the ongoing battle against the deceptive allure of deepfakes.

2. Literature Review

Existing literature reveals a shift towards leveraging advanced neural network architectures, particularly Convolutional Neural Networks (CNNs) and ResNet50, to counter the escalating threat of manipulated multimedia content.

The utilization of CNNs enables effective feature extraction from visual data, while the ResNet50's residual learning mechanism enhances the model's ability to discern subtle manipulations, resulting in improved accuracy compared to traditional methods.

As the development of CNN and ResNet50-based deepfake detection systems progresses, ethical considerations come to the forefront. Privacy concerns, potential biases, and the responsible deployment of these technologies are actively discussed.

The literature calls for continued research into future directions, emphasizing the exploration of emerging technologies and methodologies to ensure the ongoing effectiveness of deepfake detection in a rapidly evolving landscape.

The rapid evolution of deepfake technology poses a serious threat to the credibility of multimedia content, generating concerns in diverse fields such as media, politics, and cybersecurity.

Traditional detection methods struggle to match the sophistication of evolving deepfake techniques, emphasizing the critical need for cutting-edge solutions to counter the manipulation of videos and images.

Exploring the integration of Convolutional Neural Networks (CNNs) and the ResNet50 architecture by leveraging CNNs' prowess in image analysis and ResNet50's unique features, this approach aims to enhance detection accuracy in the face of emerging challenges posed by deepfake technology.

In creating an innovative deepfake detection system, the methodology involves assembling a diverse dataset and preprocessing it for model robustness. The approach integrates Convolutional Neural Networks (CNNs) and ResNet50 through transfer learning. The hybrid model undergoes training, validation, and hyperparameter tuning, with performance evaluated against baseline models. Adversarial testing gauges resilience, and considerations for real-time deployment are addressed. The entire process is documented for reproducibility and refinement.

The evaluation of the developed model's performance encompasses widely recognized metrics, including accuracy for an overall assessment, precision to measure the proportion of correctly identified positive instances, recall to gauge the model's ability to capture all actual positive instances, and the F1 score, which strikes a balance between precision and recall, providing a comprehensive measure of classification effectiveness.

3. Methodology

The methodology for this research composed of stages like selecting the dataset, understanding the architectures of CNN and ResNet50, model building by applying training to the dataset using CNN and ResNet50 and analysing the performance using metrics. The different stages are explained as follows.
The dataset used for methodology is designed for implementing the task of deepfake detection, a crucial aspect in the field of computer vision and artificial intelligence. The dataset is organized into three main folders: train, test, and validation, each serving a specific purpose in the training and evaluation of deepfake detection models. Within each of these folders, there are two labelled subfolders, namely "real" and "fake," representing authentic and manipulated images, respectively.[5]

In the train folder, the "real" subfolder comprises authentic, unaltered images, forming the basis for training the deepfake detection model on genuine visual content. Concurrently, the "fake" subfolder within the train set consists of manipulated images, essential for training the model to discern between authentic and deepfake content. The test folder mirrors this structure, with "real" containing unseen genuine images and "fake" presenting manipulated images for assessing the model's performance on previously unseen data. The validation set serves as an additional checkpoint, with "real" and "fake" subfolders providing a diverse set of images to evaluate the model's generalization capabilities and refine its performance.

The primary objective of this dataset is to facilitate the development and evaluation of deepfake detection models. Researchers and practitioners can use the provided images to train their models to distinguish between real and manipulated content. The division into training, testing, and validation sets ensures a robust evaluation of model performance, fostering the development of accurate and reliable Deep Fake detection algorithms.

3.1 Architectures and model building using CNN and RESNET50

Convolutional Neural Networks (CNNs) represent a pivotal advancement in the domain of computer vision, particularly in tasks such as image recognition and classification. A CNN architecture is characterized by its ability to automatically and adaptively learn spatial hierarchies of features.[11]

Application of convolutional and pooling layers. At its core, a CNN consists of convolutional layers that convolve input images with learnable filters, enabling the extraction of local features and patterns. These convolutional layers are often followed by activation functions, such as Rectified Linear

Units (ReLU), to introduce non-linearity. Pooling layers, commonly in the form of max pooling, are strategically placed to down sample and retain essential information while reducing computational complexity. The network typically concludes with fully connected layers for high-level feature integration and decision-making. CNNs excel at capturing hierarchical representations of input data, learning low-level features like edges and textures in early layers and progressing to more complex, abstract features in deeper layers [11]**.**

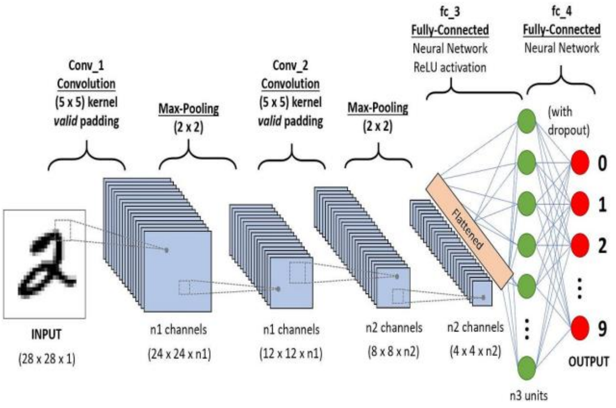
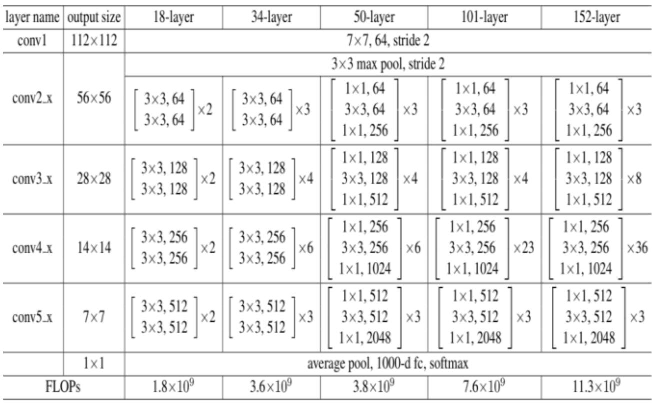


Fig.1 Convolutional Neural Network [14]

ResNet50, short for Residual Network with 50 layers, is a deep convolutional neural network architecture renowned for its exceptional performance in image classification tasks. Developed by Kaiming He and his team, ResNet50 is a variant of the original ResNet, introducing a stack of 50 layers that includes residual connections. The fundamental innovation of ResNet lies in its residual learning framework, which tackles the challenge of training very deep networks by introducing skip connections that circumvent the vanishing gradient problem. In ResNet50, the architecture comprises a series of convolutional layers, batch normalization, and rectified linear unit (ReLU) activations [1]. The network is divided into multiple residual blocks, each containing a shortcut connection that skips one or more layers. This enables the network to learn residual functions, facilitating the training of exceedingly deep models. The use of bottleneck architectures in ResNet50 optimizes computational efficiency, allowing the network to strike a balance between depth and complexity. With its proven ability to outperform previous architectures on benchmark datasets, ResNet50 has become a cornerstone in various computer vision applications, demonstrating its efficacy in feature learning and representation across a spectrum of visual recognition tasks



layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		[3×3, 64 3×3, 64] ×2	[3×3, 64 3×3, 64] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3
conv3_x	28×28	[3×3, 128 3×3, 128] ×2	[3×3, 128 3×3, 128] ×4	[1×1, 128 3×3, 128 1×1, 512] ×4	[1×1, 128 3×3, 128 1×1, 512] ×4	[1×1, 128 3×3, 128 1×1, 512] ×8
conv4_x	14×14	[3×3, 256 3×3, 256] ×2	[3×3, 256 3×3, 256] ×6	[1×1, 256 3×3, 256 1×1, 1024] ×6	[1×1, 256 3×3, 256 1×1, 1024] ×23	[1×1, 256 3×3, 256 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512 3×3, 512] ×2	[3×3, 512 3×3, 512] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Fig. 2. The diagram shows different layers of ResNet50 [15]

3.1.1 Training the model using CNN

The training procedure of Convolutional Neural Networks (CNNs) for deepfake detection involves a systematic process to enable the model to learn and generalize patterns indicative of real and fake content.[1]

The process begins with the preparation of a labelled dataset comprising authentic and manipulated images or frames from videos. These images serve as input to the CNN, and the network undergoes an iterative training process. During training, the CNN adjusts its internal parameters, or weights, by comparing its predictions to the ground truth labels using a predefined loss function, often binary cross-entropy in the case of binary classification for deep fake detection.[9] The optimization algorithm, commonly gradient descent, is then employed to minimize this loss by updating the network's weights. The training dataset is typically divided into batches to facilitate efficient computation and parallel processing. As the CNN progresses through epochs, representing complete passes through the entire dataset, it refines its ability to extract hierarchical features from the images, learning to discern subtle artifacts or inconsistencies introduced by deepfake generation techniques.

To prevent overfitting, regularization techniques such as dropout or batch normalization may be applied.[11] The training process concludes when the model exhibits satisfactory performance on a separate validation dataset. The effectiveness of the trained CNN is subsequently evaluated on a testing dataset, and metrics like accuracy, precision, recall, and F1 score are analyzed to assess the model's ability to distinguish between authentic and manipulated content.[13] Fine-tuning or adjusting hyperparameters may be performed to optimize performance further.

3.1.2 Training the model using ResNet50

The training procedure for deepfake detection using ResNet50 involves several key steps to enable the model to effectively identify patterns associated with manipulated content. Initially, a labelled dataset is prepared, comprising authentic and deepfake images or video frames.[15]

The ResNet50 architecture is then configured, featuring multiple residual blocks that facilitate the flow of information through deep layers.

The choice of a binary cross-entropy loss function and an optimization algorithm, often stochastic gradient descent (SGD), aims to minimize the disparity between predicted and actual class labels during training. The model undergoes iterative training, processing batches of images and adjusting its weights through back propagation. [15] To mitigate over fitting, regularization techniques such as dropout and batch normalization are applied. Fine-tuning of hyperparameters, including learning rates and batch sizes, is performed to optimize model performance. The training process spans multiple epochs, allowing ResNet50 to learn hierarchical features and intricacies associated with deep fake generation.[9]

Performance monitoring on a validation dataset guides the decision to terminate training when the model achieves satisfactory results or shows signs of convergence. Following training, the model is evaluated on a separate testing dataset, and key metrics such as accuracy, precision, recall, and F1 score are computed to gauge its effectiveness in discerning between authentic and manipulated content. [13]

3.2 Evaluation metrics:

Accuracy is a commonly evaluation metric used for evaluating the performance of a classification model, including those designed for deepfake detection. [13] [16]
It is a straightforward metric that quantifies the percentage of correctly classified instances out of the total number of instances in the dataset.

$$\textbf{Accuracy} = \frac{\textbf{Correct Predictions}}{\textbf{All Predictions}}$$

Fig. 3 Performance Metrics – Accuracy [16]

4. Results & Discussion

Our research endeavours to compare the early-stage performance of ResNet50 and CNN architectures in the task of deepfake detection, specifically evaluated at Epoch 10. The obtained accuracy scores provide an initial glimpse into the models' capabilities within the context of our labelled dataset, comprising both real and fake data.

ResNet50 Performance:
At Epoch 10, ResNet50 achieved an accuracy score of 0.7066. This score suggests a respectable level of discernment between real and fake content, showcasing the model's ability to capture relevant features even in the early stages of training.

CNN Performance:
In contrast, the CNN model exhibited a higher accuracy score of 0.9067 at Epoch 10. This indicates a superior early-stage performance compared to ResNet50 within our dataset, emphasizing the CNN's efficacy in capturing discriminative features associated with deepfake content.

The observed accuracy scores highlight the divergence in early-stage performance between ResNet50 and CNN for deepfake detection. The CNN architecture, with its accuracy of 0.9067, demonstrates a notable advantage over ResNet50 (0.7066) at Epoch 10. This distinction suggests that, within the limited training epochs considered, the CNN model exhibits a more rapid and effective learning of discriminative features relevant to differentiating real and fake content in our labelled dataset.

5. Limitations & Future Scope

The limitations and future scope of proposed work as explained as follows

5.1 Limitations

Incomplete Model Training:
The primary limitation remains the reliance on early-stage accuracy scores, which may not fully represent the models' ultimate performance. Neural networks typically require more training epochs to converge and achieve optimal accuracy. Future work should involve a thorough evaluation across the entire training duration to provide a comprehensive understanding of the models' capabilities.

Dataset Specificity:
The results are contingent on the characteristics of our specific dataset, which includes labelled real and fake data. The performance of the models may vary when applied to different datasets with distinct features and distributions. Extending the analysis to diverse datasets will enhance the generalizability of our findings.

Potential Overfitting:
While the accuracy scores are encouraging, there is a risk of overfitting, particularly in the early stages of training. Future investigations should monitor the models for signs of overfitting and consider implementing regularization techniques to ensure the models generalize well to unseen data.

5.2 Future Scope

Extended Training:

Both CNN and ResNet50 models should be further trained across a more extensive range of epochs to observe their convergence patterns and determine the point at which performance stabilizes.

Fine-Tuning Hyperparameters:

Additional experimentation with hyperparameters, such as learning rates and batch sizes, could optimize the models' performance and potentially narrow the performance gap observed after two epochs.

Exploration of Other Metrics:

Beyond accuracy, the exploration of other metrics such as precision, recall, and F1 score could provide a more nuanced understanding of each model's ability to correctly identify deepfake instances.

6. Conclusion

In the comparative analysis of deepfake detection using CNN and ResNet50 architectures, the early-stage results after two epochs revealed a notable performance difference. The CNN exhibited a higher accuracy of 0.9067 compared to ResNet50's accuracy of 0.7066 on a dataset comprising labelled real and fake instances. This suggests that, within the initial training epochs, the CNN model displayed a more rapid and effective learning capacity for distinguishing between authentic and manipulated content. However, it is crucial to interpret these findings with caution, considering the early stage of training. Future work should involve extended training, hyperparameter tuning, and evaluation on diverse datasets to provide a more comprehensive understanding of the comparative effectiveness of CNN and ResNet50 in real-world deepfake detection scenarios. Additionally, attention to potential overfitting and careful consideration of dataset characteristics are essential for a more nuanced interpretation of model performance.

In summary, even with the updated accuracy for the ResNet50 model, the CNN model still outperforms it in terms of accuracy, recall, and AUC. However, the ResNet50 model maintains a slightly higher F1 score and precision. The choice between these models would depend on the specific goals and priorities of your task, as different metrics may be more important depending on the context.

7. References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, June 2016.

[2] D. Li, D. Zhao, Y. Chen, and Q. Zhang, "Deepsign: Deep learning based traffic sign recognition," in 2018 international joint conference on neural networks (IJCNN), pp. 1–6, IEEE, 2018.

[3] T. T. Nguyen et al., "Deep learning for deepfakes creation and detection: A survey," Computer Vision and Image Understanding, vol. 223, pp. 1–20, Oct. 2022, doi: 10.1016/j.cviu.2022.103525.

[4] M. Westerlund. The emergence of deepfake technology: A review. Technology Innovation Management Review 9(11):39–52, 2019.

[5] A. Rössler, D. Cozzolino, L. Verdoliva, et al.FaceForensics++: Learning to detect manipulated facial images. [2023-05-31].

[6] Tariq, Shahroz, Sangyup Lee, Hoyoung Kim, Youjin Shin, and Simon S. Woo. "Detecting both machine and human created fake face images in the wild." In Proceedings of the 2nd international workshop on multimedia privacy and security, pp. 81-87. 2018.

[7] Taeb, Maryam, and Hongmei Chi. "Comparison of Deepfake Detection Techniques through Deep Learning." Journal of Cybersecurity and Privacy 2, no. 1, 2022.

[8] Guarnera, Luca, Oliver Giudice, Cristina Nastasi, and Sebastiano Battiato. "Preliminary fo-rensics analysis of deepfake images." In 2020 AEIT international annual conference (AEIT), pp. 1-6. IEEE, 2020.

[9] Hasin Shahed Shad, Md. Mashfiq Rizvee, Nishat Tasnim Roza, S. M. Ahsanul Hoq, Mohammad Monirujjaman Khan, Arjun Singh, Atef Zaguia and Sami Bourouis."Comparative Analysis of Deepfake detection method using Convolutional neural network"

[10] DEEPFAKES: THREATS AND COUNTERMEASURES SYSTEMATIC REVIEW Marwan Albahar, Jameel Almalki

[11] Conceptual Understanding of Convolutional Neural Networks – A deep learning approach. Sakshi Indolia , Anil
Kumar Goswami , S.P. Mishra , Pooja Asopa

[12] Deep learning for deepfake creation and detection: A survey. Thanh Thi Nguyena, Quoc Viet Hung Nguyenb, Dung Tien Nguyena, Duc Thanh Nguyena, Thien Huynh-Thec, Saeid Nahavandid, Thanh Tam Nguyene, Quoc-Viet Phamf , Cuong M. Nguyen

[13] https://neptune.ai/blog/performance-metrics-in-machine-learning-complete-guide Performance metrics in Machine learning

[14] https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53. A Comprehensive Guide to Neural Networks

[15] https://datagen.tech/guides/computer-vision/resnet-50/ Resnet50: The basics and quick tutorial.

[16] https://www.evidentlyai.com/classification-metrics/accuracy-precision-recall#:~:text=Accuracy%20is%20a%20metric%20that,the%20total%20number%20of%20predictions. Accuracy vs. Precision vs. Recall in machine learning.