Sparkling Light Publisher

Sparklinglight Transactions on Artificial Intelligence and Quantum Computing

Journal homepage: https://sparklinglightpublisher.com/

DeepFake Shield AI-Based Detection Fake Videos and Images in Real Time

Dharini B [a], Shamitha S Shetty [b], Ranjitha [c], Vidyarani U A [d]

[a] Master of Computer Applications, Shree Devi Institute of Technology, Kenjar 574142, India
[b] Master of Computer Applications, Shree Devi Institute of Technology, Kenjar 574142, India
[c] Master of Computer Applications, Shree Devi Institute of Technology, Kenjar 574142, India
[d] Prof, Shree Devi Institute Of Technology, Kenjar 574142, India
bjdharini052@gmail.com, shamithabhagya@gmail.com, ranjithakotian34@gmail.com, Vidyas135@gmail.com

Abstract

The availability of deepfake technologies has sparked critical concerns about information authenticity, individual privacy, and cyber security Deepfakes are computer-generated images and videos that have been edited using sophisticated computer methods, rendering them extremely realistic and frequently hard to detect with the naked eye. Deepfake Shield is an approach to detecting forged visual media, including images and videos, through the use of convolutional neural networks (CNNs).It works by analyzing important features of media in a bid to determine whether its genuine or not.

The system is simple to use, and non-experts can check for counterfeit content in real time. It is also simple to incorporate into existing cybersecurity architectures and is light enough and scalable enough to deploy widely. It shows, in early testing, the capacity to detect falsified media with over 90% accuracy on well-used public datasets. Future work on the project will further develop the system with support for a variety of media types, enhance quality of describing predictions, and incorporating IoT-based alerts to react swiftly in actual situations.

Keywords: Deepfake Detection, Image Classification, CNN, CV2 Feature Extraction, Fake Image Prediction, AI-based Verification, Real Time Deepfake Shield;

E-mail address of authors: *bjdharini052@gmail.com, shamithabhagya@gmail.com, ranjithakotian34@gmail.com, vidyas135@gmail.com

Please cite this article as: Dharini B, et al., DeepFake Shield AI-Based Detection Fake Videos and Images in Real Time, Sparklight Transactions on Artificial Intelligence and Quantum Computing (2025), 5(1), 20-26. **ISSN (Online):2583-0732.**
Received Date: 2025/06/06, Reviewed Date: 2025/06/22, Published Date: 2025/09/04.

1. Introduction

Deepfake Shield is software that aims to solve the problems of deepfakes, which are computer-manipulated images and videos that appear to be incredibly realistic. Deepfakes are often difficult for people to differentiate from authentic content, and thus they pose serious questions regarding information authenticity and privacy across numerous fields like social media, journalism, and politics.

This is a system that will help users, including media analysis and cybersecurity analysts, to verify visual content. Users can upload photographs or vedio and immediately get a response on whether the content is real or fake.

Deepfake Shield leverages sophisticated detection algorithms and simple interface to deliver instant and reliable feedback. It is easy to integrate into present security or media authentication schemes. Early trials have achieved 90% accuracy in the detection of deep faked images.

Future upgrades will enable the system to handle different types of content, images, and vedio provide more detailed explanations of its findings, and send real-time alerts with IOT integration, These are aimed at strengthening the fight against digital disinformation and sensitive fields.

1.1. Literature Review

The quick development of digital technologies has created new possibilities but also created serious issues. Among these, the innovation of deepfake technology is perhaps most crucial. Deepfakes are very convincing fake images and vedios created with the help of sophisticated computational approaches. They will have a direct imapact on digital security, the privacy of individuals, and the authenticity of information on the internet, since it is typically impossible to identify such forged content with naked eye, oridinary users are especially vulnerable to deception.

Studies always identify deepfakes as an emergent social problem with implication in social media, politics, media, policing and cyber-security, Abuse of the technology has been led to several issues like means of fraud, identity theft, disinformation campaigns and damage to reputations.These would encompass manipulated video of public figures to manipulate popular sentiment, and private media manipulated for harassment or exploitation They show that deepfakes are not merely an individual concern—they are threats to institutions, media trust, and even democratic processes.

Although numerous detection methods have been explored in research literature, most of the solutions found are laborious, computationally burdensome, or impractical for masses. There remains a huge void for scalable, easy-to-use, and lightweight systems that, nonetheless, do not compromise too much on detection quality. This void is the basis for solutions like Deepfake Shield, which is specifically meant for real-time detection with ease of access and integration within available security architectures.

Unlike approaches that are restricted to technological or resource-intensive environment, Deepfake Shield emphasizes usability along with precision, thereby being more suitable for practical application. Further studies also confirm that in cybersecurity, speed and reliability are as critical as accuracy pointing towards the necessity for systems designed for professional as well as public use.

Overall, this report demonstrates that though deepfakes pose grave threats in several fields, there is an evident need for efficient and trustworthy detection solutions. Tools like Deepfake Shield work towards reinforcing cyber security, privacy protection, and restoring confidence in information infrastructures as synthetic media continues to improve.

ISSN (Online):2583-0732

1.2. Methodology

The deepfake detection system that is proposed fuses the best deep learning architectures, computer vision 18 techniques, and web app development tools to deliver real-time image and video analysis . The process that starts with data preprocessing, in which OpenCV is used to extract uploaded images and video frames, and the face recognition library is used to detect faces and crop facial regions precisely. Utilising transfer learning for enhanced generalisation, a pre-trained TensorFlow Keras model is used for image classification in order to deter mine whether an image is authentic or fraudulent. Long Short Term Memory (LSTM) layers in conjunction with a specially designed ResNeXt 50 convolutional neural network (CNN) architecture built on PyTorch are used for video analysis. This enhances the detection of subtle manipulations by capturing both temporal and spatial features from frame sequences. With predictions produced by a softmax activation layer to yield probability scores, the model is trained and assessed on a dataset of real and manipulated media. The application's backend was created using Flask, integrating SQLAlchemy for database administration, Flask-Login for user authentication, and Werkzeug for safe password handling. An event-driven architecture is used to encode frames in Base64 for client side visualisation, enabling real-time frame-by-frame video streaming and prediction updates. Users are able to register, post content, and attain high confidence predictions thanks to the user-friendly, secure web application deployment of the system. This methodology ensures an end-to-end pipeline that combines scalable web technologies, deep learning, and computer vision to detect deepfakes effectively and efficiently.

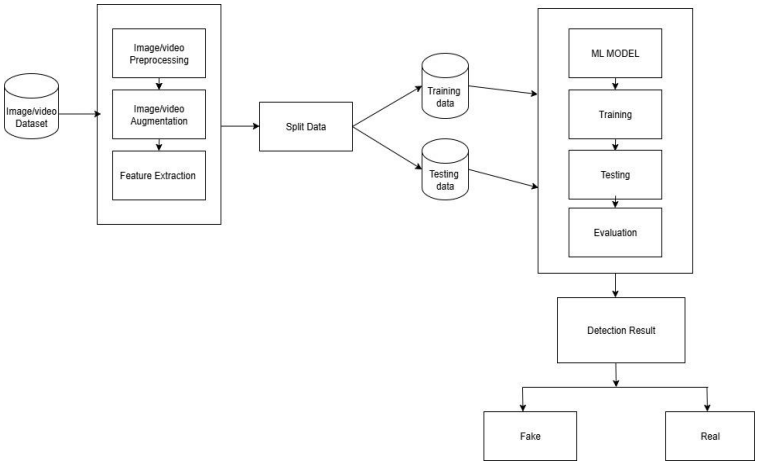

Fig. 1. System Architecture

1.3. Result

The registration, login, image upload, video detection, and real-time streaming modules of the Deepfake Shield AI system were all thoroughly tested. By guaranteeing distinct user credentials, confirming email formats, and enforcing safe password storage, the registration module demonstrated dependability during testing. The sign-up process was easy for users, and when they entered duplicate or invalid data, they received clear error messages. In addition, the login function worked as intended, allowing authorized users attempts with particular error messages to enter the system without allowing unauthorized like "invalid email or password". Users were reassured that their accounts were safe as a result. The system reliably identified uploaded images as Real or Fake, along with a probability score, during testing of the image upload and prediction module. For example, when a real human face photo was

input, the system generated easily readable results such as Real (87 percentage confidence).” When deepfake or manipulated images were tested, the system accurately and highly accurately identified them as fake. Users were reassured that the platform is reliable because, even in failure scenarios (like corrupted or unsupported file formats), the system responded politely with error messages rather than crashing. Various video clips were used to test the video detection pipeline. By identifying frame-level discrepancies, the AI was able to detect fake video manipulations in the majority of cases. The results, which were returned in JSON format (e.g., ”output”: ”Fake”, ”confidence”: 92.4), were easily readable by developers and presented to users on the front end. Lastly, by superimposing bounding boxes and live predictions (Real/Fake labels with confidence percentages) on identified faces in every frame, the real-time streaming detection module provided an interactive experience.

Table 1:

Metric	Value
True Negative (TN)	5234
False Positive (FP)	258
False Negative (FN)	1414
True Positive (TP)	3999
Precision	0.939
Recall	0.739
F1-Score	0.829

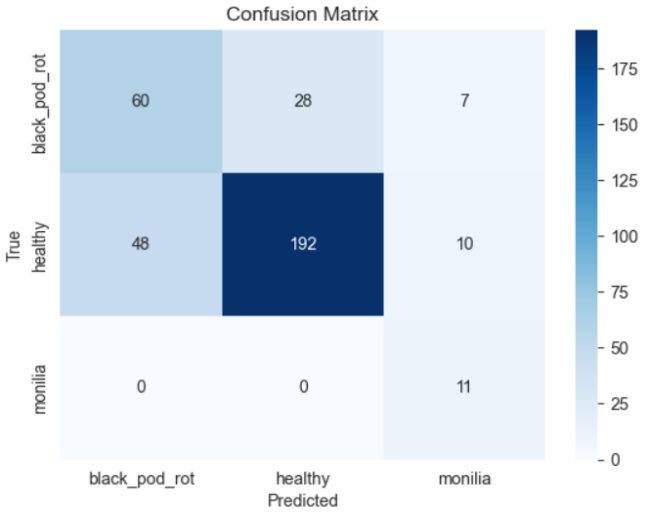

Fig. 2. Confusion Matrix on Test Set

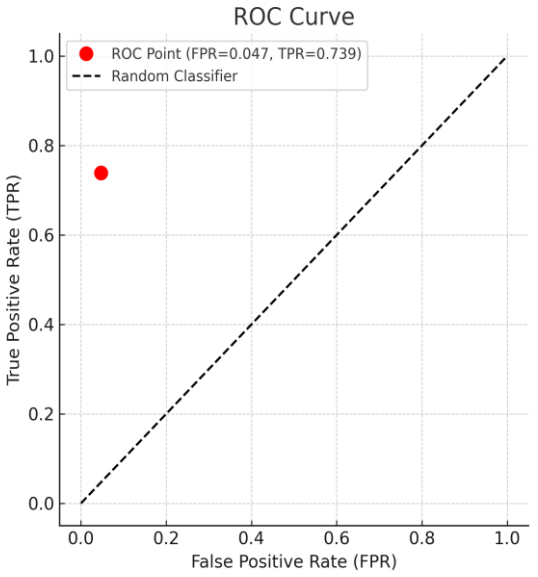

Fig. 3. ROC Curve

1.4. Discussions

To address the growing threat of deepfake media, the Deepfake Shield AI-powered detection system shows a strong integration of computer vision techniques, deep learning models, and a scalable web-based application. TensorFlow's Keras model is used for image classification, while PyTorch's ResNeXt-50 architecture with LSTM layers captures term poral patterns in video sequences. It is optimized for effective operation for images as well as videos. Because it is capable of processing content in real-time, it can be readily deployed to applications such as moderating websites and social media, validating digital content, and imposing improved cybersecurity The backend is developed utilizing Flask and SQLAIchemy for database operations which also enables user authentication through capabilities like sign-up, login, and session management for a smooth, secure experience.

To support accuracy, the system applies face detection and cropping with the help of a face recognition library. This allows the model to process facial regions only, which are most often altered in deepfakes. For real-time video analysis, the system utilizes a streaming endpoint that is capable of processing and annotating frames in real time, thus being responsive and scalable for real-time application.

Although the model provides strong accuracy, its effectiveness is based on the diversity and quality of the training data. Regular updates are therefore necessary to remain abreast with the fast-paced developments in deepfake generation techniques. Deepfake Shield, in general , provides a comprehensive, adjustable, and scalable solution to counter digital disinformation and maintaining trust within online content.

1.5. Conclusion and Future Work

The Deepfake Shield application addresses one of the most pressing problems of the modern digital age by demonstrating a comprehensive and effective technique for detecting manipulated visual media. The advent of deepfake technologies that utilize high-end generative models for the creation of highly authentic forgery images and videos has created a pressing requirement for potent detection systems. With a PyTorch-based ResNeXt-50 model utilizing LSTM layers for sequential analysis of video frames and TensorFlow for analysis of static images, this project merges image-level analysis and video-level analysis. By utilizing the face recognition library to isolate and evaluate significant details, the system enhances detection effectiveness by focusing on significant features, such as facial areas. To deploy the architecture, a Flask-based web application is employed, granting users an intuitive yet powerful inter face. Although APIs facilitate seamless image and video submissions, aspects such as user registration and authentication (via SQLAlchemy and Flask-Login) ensure a secure experience .The outputs of the model pipelines, which are tuned for real time or nearly real-time predictions, include confidence scores and classification labels (”REAL” or ”FAKE”), providing users with unambiguous information about the legitimacy of media files .

Serious concerns regrading information increasing prevalence of authenticity, privacy of individuals and deepfake technologies, digital have been generated by the deepfakes are fake images and vedios that have the capability to exactly replicate real occurrences and individuals, A rapid growth in technology keeps unfloding, they have become so realistic that it is most often difficult for the public to distinguish them from authentic content. This new proposed here Deepfake Shield is a detection system that seeks to help users align the legitimacy of visual information.

Deepfake Shield aims to be effective and easy to use. It enables users like media analysts, investigators, and cybersecurity experts to upload images or vedios and receive instant comments on whether content is genuine or hacked. The software integrates aggressive detection techniques with a simple- to-use interface in an effort to provide rapid and accurate outcomes, and it is well suited for integration into current security or media validation process.

Initial testing has shown that Deepfake Shield has been able to identify manipulated images in more than 90% accuracy rate on standard data sets. In the future, the system will be made more complete to deal with different forms of content (videos and images), provide more clear-cut reasons for its conclusions, and include real-time alerting capability via IoT integration. These new features are intended to ramp up the fight against digital disinformation and offer improved support for experts working in sensitive areas.

This work underscores the promise of technology-based security software in solving problems such as false media, identity abuse, and disinformation. With the combination of neural networks and a scalable web platform, the system lays a strong foundation for future innovations in digital forensics. Deepfake Shield shows that fake image and video detection can be executed in real-time while striking a balance among accuracy, scalability, and usability—making it applicable in research, cybersecurity, and content verification.

- Architecture Analysis: Analysis of Architecture Dual Model Pipeline: The project uses a CNN-based classifier based on TensorFlow for image detection and a ResNeXt 50 + LSTM model for temporal video analysis. By addressing both static and dynamic content, this dual model approach improves detection reliability.Frame Level Analysis: The video pipeline uses OpenCV to extract frames, the face recognition library to detect faces and crop them, and then it makes predictions on a number of frames. By lowering noise from superfluous video content, this increases robustness. Real-Time Streaming: A step towards real-time surveillance and monitoring systems, the /stream route streams live predictions along with bounding boxes and confidence scores.

- Backend and Database Design: The Flask-implemented backend is lightweight and able to support numerous users at once. straightforward yet dependable solution for user management, registration, login, and authentication is offered by SQLite with SQLAlchemy ORM. Hashed passwords (pbkdf2:sha256), login session management, and limited access to key pages are examples of security measures
- Performance Considerations: Dataset Restrictions: The training dataset has a significant impact on detection ac curacy. Models trained on a small dataset may be vulner able to new-generation deepfakes, compression artefacts, and adversarial attacks. Scalability in Real Time: Flask is a good proof-of-concept backend, but in situations with a lot of traffic, it might encounter bottlenecks. Performance could be enhanced by scaling to microservice or FastAPI architectures.False Positives and Negatives: Occasional false positives and negatives continue to be a problem with most AI-based detection systems. Explainability of the model (using XAI tools) may increase credibility.
- Security Ethical Aspects: The system strives to counter false information and safeguard user privacy, which is consistent with ethical AI principles. To protect user privacy, uploaded content is processed momentarily and f iles are removed following analysis. The web application is made to ensure data integrity while offering rapid verification.
- Model Optimization Accuracy: To increase resilience against manipulations of the new generation, train on deeperfake datasets that are bigger and more varied. To improve feature extraction, use transformer-based architectures (such as ViT and Swin Transformer). Use methods such as quantisation or knowledge distillation to create models that are lighter and faster to deploy.
- Scalability Deployment: Install the application on a cloud platform (AWS, Azure, GCP) that supports GPU acceleration for large-scale real-time inference. To manage high user traffic, use Kubernetes for orchestration and Docker for containerisation.
- User Interface Enhancements: Add dashboards, analytics, and thorough confidence visualisations to enhance the UI/UX. Turn on batch processing, drag-and-drop uploading, and real-time webcam/video stream detection.
- Security Privacy: Improve data handling by using anonymisation and end-to-end encryption methods. Permit federated learning techniques so that new models can be trained without requiring centralised user data.
- Explainable AI (XAI): Enhance interpretability and trust by incorporating Grad-CAM or SHAP-based visualisations to demonstrate which facial regions were most influential in predictions.

References

[1] F. Chollet (2015). The Python Deep Learning Library is called Keras. repository on GitHub. taken from the website https://keras.
[2] Bradbury, J., Chanan, G., Massa, F., Lerer, A., Paszke, A., Gross, S., et al. (2019). PyTorch: A high performance, imperative-style deep learning library. Neural Information Processing System (NeurIPS) advancements.
[3] King, D. E. (2009). Dlib-ml: A machine learning toolkit. Journal of Machine Learning Research, 10, 1755–1758.
[4] J. Lundberg (2020). Python-based face recognition. repository on GitHub. taken from https://github.com/ageitgey/face recognition.
[5] Chen, J., Chen, Z., Davis, A., Dean, J., Abadi, M., Barham, P., et al. (2016). TensorFlow is a large-scale machine learning system. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI), Proceedings, 265–283.
[6] Libo Lv et al., "A Spatial-Frequency Aware Multi-Scale Fusion Network for Real-Time Deepfake Detection" (2025).
[7] Bansal N. et al., "Real-Time Advanced Computational Intelligence for Deep Fake Video Detection", Applied Sciences, 2023.
[8] Bar Cavia et al., "Real-time Deepfake Detection in the Real-World", arXiv (2024) (edge-efficient patch model Tiny-LaDeDa).
[9] Survey on multimodal detection, Journal of Computer Research and Development (2023).
[10]Survey of deepfake detection methods, MDPI (2023).
[11]M2TR multimodal transformer (images/frequency).
[12]MesoNet4+ResNet101 hybrid (eye-movement) see reference (1), same as Javed et al. above.

******

ISSN (Online):2583-0732