World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

Deepfake Image Detection: From CNN to Vision Transformer

Chaitali Charandas Daware, V. K. Shandilya and N. P. Mohod

Department of Computer Science and Engineering, Sipna College of Engineering and Technology Amravati, Maharashtra, India.

World Journal of Advanced Research and Reviews, 2026, 30(02), 1241-1255

Publication history: Received on 22 March 2026; revised on 06 May 2026; accepted on 09 May 2026

Article DOI: https://doi.org/10.30574/wjarr.2026.30.2.1169

Abstract

The exponential proliferation of synthetic media, colloquially known as "deepfakes," driven by advanced Generative Adversarial Networks (GANs) and diffusion models, presents a formidable challenge to digital forensics, personal privacy, and societal trust. While Convolutional Neural Networks (CNNs) have historically served as the cornerstone for detecting such manipulations, they frequently exhibit limitations regarding generalization to unseen manipulation algorithms and robustness against real-world distortions. This paper introduces DeepShield, an industry-grade, full-stack deepfake detection web application powered by a fine-tuned SigLIP2 (Sigmoid Loss for Image-Image Pre-training) vision-language encoder. Unlike traditional CNN-based approaches that rely solely on hierarchical spatial feature extraction, the proposed model utilizes a transformer-based architecture pre-trained with sigmoid loss, enabling the capture of global semantic context and subtle texture inconsistencies.

The system was evaluated on the prithiv ML mods/Open Deepfake-Preview dataset, achieving an overall accuracy of 94.44%. The model demonstrated exceptional performance, achieving a precision of 97.18% for the "Fake" class and a recall of 97.34% for the "Real" class, significantly minimizing false accusations in forensic scenarios. Furthermore, this research bridges the gap between theoretical modeling and practical application by implementing a user-centric forensic interface featuring an interactive Region of Interest (ROI) selector and temporal video analysis. Comparative analysis reveals that the proposed SigLIP2 model outperforms standard CNN architectures and existing Convolutional Vision Transformer (CViT) benchmarks, offering a robust, scalable solution for digital media authentication.

Keywords: Deepfake Detection; Siglip 2; Vision Transformers; Digital Forensics; Flask; Web Application; Generative Adversarial Networks

1. Introduction

1.1. Context and Motivation

In recent years, rapid progress in artificial intelligence has significantly transformed the way digital content is created and manipulated. Advanced deep learning techniques, particularly generative models such as autoencoders and Generative Adversarial Networks (GANs), have enabled the creation of highly realistic synthetic media, commonly referred to as deepfakes. These manipulated images and videos can convincingly alter a person's identity, expressions, or actions. While such technologies offer promising applications in entertainment, virtual reality, and content creation, they also introduce serious risks. Deepfakes are increasingly being used for misinformation, digital impersonation, and fraudulent activities, raising concerns about trust and authenticity in digital media. As generative models continue to improve, detecting manipulated content has become increasingly challenging, especially when dealing with high-quality forgeries that appear visually indistinguishable from real data.

* Corresponding author: Chaitali Charandas Daware

Traditional detection methods initially focused on identifying visible artifacts or inconsistencies introduced during the generation process. However, modern deepfake techniques have significantly reduced such artifacts, making conventional approaches less effective. Therefore, there is a growing need for more advanced and reliable detection systems capable of identifying subtle inconsistencies in both spatial and semantic representations.

1.2. Problem Statement

Most deepfake detection systems today mainly use Convolutional Neural Networks (CNNs) because they are good at analyzing image features. However, these models still have some important limitations. One of the main issues is that CNNs focus more on small, local details like textures, and often miss the overall structure of the face. Because of this, they may fail to detect deeper inconsistencies present in manipulated images. Another problem is that these models do not generalize well. If they are trained on a specific dataset, they often struggle when tested on new types of deepfakes created using different techniques, which affects their reliability. In addition, many existing approaches are designed mainly for research purposes and focus only on improving accuracy, without considering how they can be used in real-world applications. This makes them less useful for practical scenarios like forensic analysis. Due to these challenges, there is a clear need for a system that is not only accurate but also more flexible, reliable, and easy to use in real-life situations.

1.3. Research Contributions

To address the challenges present in existing deepfake detection approaches, this research introduces DeepShield, a complete end-to-end system designed with a balance of accuracy and real-world usability in mind. Instead of relying only on traditional methods, the system makes use of a SigLIP2-based transformer model, which is better suited for understanding both fine details and the overall structure of an image. This helps in identifying even high-quality deepfakes where visual differences are not easily noticeable. Along with improving detection performance, equal importance is given to making the system practical and easy to use. For this purpose, a web-based interface is developed using Flask, where users can simply upload images or videos and receive results quickly without needing any technical expertise. This makes the system more accessible for general users as well as professionals. In addition, a Region of Interest (ROI) feature is included, allowing users to focus on specific areas of the face such as eyes, mouth, or other regions for closer inspection. This is especially useful in forensic analysis, where detailed examination of certain parts can provide better insights. Overall, the proposed system not only improves detection capability but also ensures that it can be effectively used in real-life scenarios.

2. Related Work

Initial research in deepfake detection primarily focused on identifying physiological inconsistencies present in synthetic media. For example, early studies exploited irregularities such as unnatural eye blinking patterns, which were often absent in generated content. However, as generative models improved, these simple artifacts were eliminated, reducing the effectiveness of such approaches. Subsequently, Convolutional Neural Networks (CNNs) became the dominant method for deepfake detection. Architectures such as VGG16 and ResNet were widely adopted to extract spatial features from facial images. Transfer learning techniques further enhanced performance by leveraging pre-trained models. Despite these advancements, later studies revealed that CNN-based models often rely on dataset-specific artifacts, such as compression noise, rather than learning meaningful manipulation patterns. This limitation affects their ability to generalize across different datasets and deepfake generation methods.

To overcome these issues, hybrid approaches combining CNNs with temporal models such as Recurrent Neural Networks (RNNs) were introduced, particularly for video-based deepfake detection. These models analyze both spatial and temporal inconsistencies across frames. Additionally, ensemble methods have been proposed, integrating multiple models to focus on different facial regions such as eyes, nose, and mouth. Although these approaches improve accuracy, they significantly increase computational complexity and resource requirements. In contrast, the proposed approach utilizes a single Vision Transformer-based model, SigLIP2, which efficiently captures global dependencies and semantic relationships within images. This design reduces computational overhead while maintaining strong detection performance, making it more suitable for real-world deployment.

3. Literature Review

3.1. Traditional and Handcrafted Approaches

Early methodologies for deepfake identification relied on distinct physiological flaws inherent to initial generative models. Researchers such as Li et al. [1] exploited the absence of natural blinking patterns in synthesized videos. While effective for early GANs, these approaches proved brittle as generative models evolved to replicate biological signals accurately. Similarly, techniques analyzing head pose inconsistency [2] lost efficacy as face-swapping algorithms improved geometric alignment.

3.2. The Dominance of CNN Architectures

The advent of deep CNNs, notably VGG16, ResNet, and XceptionNet, marked a shift towards data-driven feature extraction. Studies by Krandikar et al. [3] demonstrated that transfer learning could effectively identify manipulation artifacts. However, a critical analysis by Wang et al. [4] highlighted that CNNs often learn dataset-specific artifacts—such as compression residues—rather than the intrinsic properties of deepfakes. This leads to a significant performance drop when models encounter "in-the-wild" data, as evidenced in the Deepfake Detection Challenge (DFDC) [5].

3.3. Transition to Hybrid and Transformer Models

To mitigate the limitations of frame-level analysis, researchers integrated Recurrent Neural Networks (RNNs) with CNNs. Amerini et al. [6] utilized optical flow to detect temporal irregularities. More recently, the focus has shifted towards Vision Transformers (ViT). Soudy et al. [7] proposed a complex ensemble involving three distinct models for eyes, nose, and full face, combining CNNs with a Convolutional Vision Transformer (CViT). While their approach achieved high accuracy, the computational overhead of running concurrent models limits practical deployment.

3.4. The SigLIP Advantage

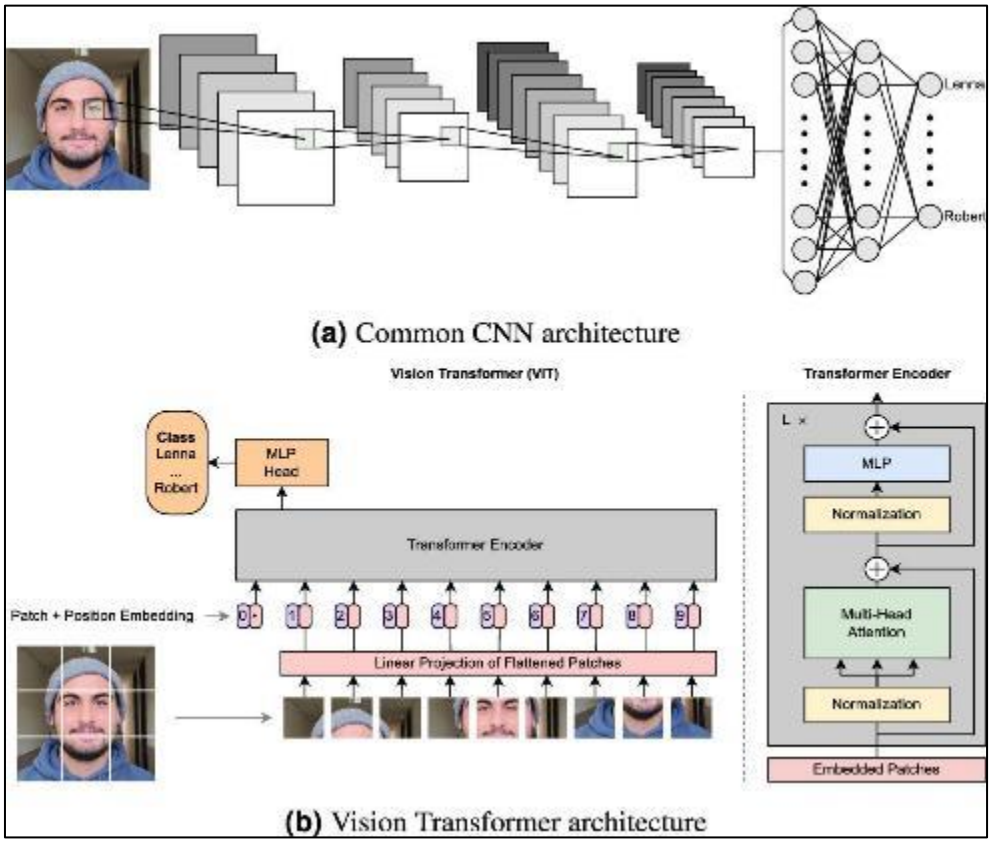

Figure 1 Architecture Comparison of CNN vs. Vision Transformer

Vision Transformers process images as sequences of patches, capturing global dependencies through self-attention. The SigLIP architecture further refines this by employing a sigmoid loss function for contrastive learning, as opposed to the

standard softmax normalization. This paper explores the application of SigLIP2, hypothesizing that its pre-trained visual-semantic alignment offers superior detection capabilities compared to standard CNNs and CViTs.

4. System Architecture

4.1. Architectural Overview

The DeepShield system is designed using a three-tier architecture to ensure smooth operation, scalability, and ease of use. The first layer is the presentation tier, which consists of a responsive web interface where users can upload images or videos and view the detection results in a simple and clear format. This layer focuses on user interaction and makes the system accessible even to non-technical users. The second layer is the logic tier, which is handled by a backend built using Python Flask. This layer is responsible for managing API requests, processing user inputs, and coordinating the flow of data between the frontend and the detection model. It acts as the core controller of the system, ensuring that all components work together properly. The third layer is the data and inference tier, where the actual deepfake detection takes place. This includes the SigLIP2-based model along with video processing modules implemented using OpenCV. This layer handles image preprocessing, feature extraction, and final prediction, making it the most critical part of the system.

4.2. Model Specification

The core of the DeepShield system is the deepfake detection model based on the SigLIP2 architecture, derived from the google/siglip-base-patch16-512 model. This model is designed to better understand both local and global features in an image. In this approach, input images are first resized to 512×512 and then divided into smaller non-overlapping patches of size 16×16. Each of these patches is treated like a token, similar to how words are processed in natural language models. This allows the model to analyze the image as a sequence rather than just a grid of pixels.

The transformer encoder then applies a self-attention mechanism, which helps the model understand the relationship between different parts of the image. Instead of looking at regions independently, the model learns how different facial areas are connected. This makes it possible to detect inconsistencies across distant regions, such as mismatched alignment between eyes, lips, or facial contours.

After processing through multiple transformer layers, the final output is passed to a classification head. This head uses a multi-layer perceptron (MLP) to classify the image into two categories: real or fake. This binary classification helps in clearly identifying whether the input image is manipulated or authentic.

The final classification decision is obtained using a sigmoid activation function applied to the output of the classification head, which can be expressed as: $P(y = 1 \mid x) = \sigma(Wx + b)$ where σ represents the sigmoid function, and W and b denote the learnable parameters of the model. This formulation allows the system to produce probabilistic outputs for binary classification.

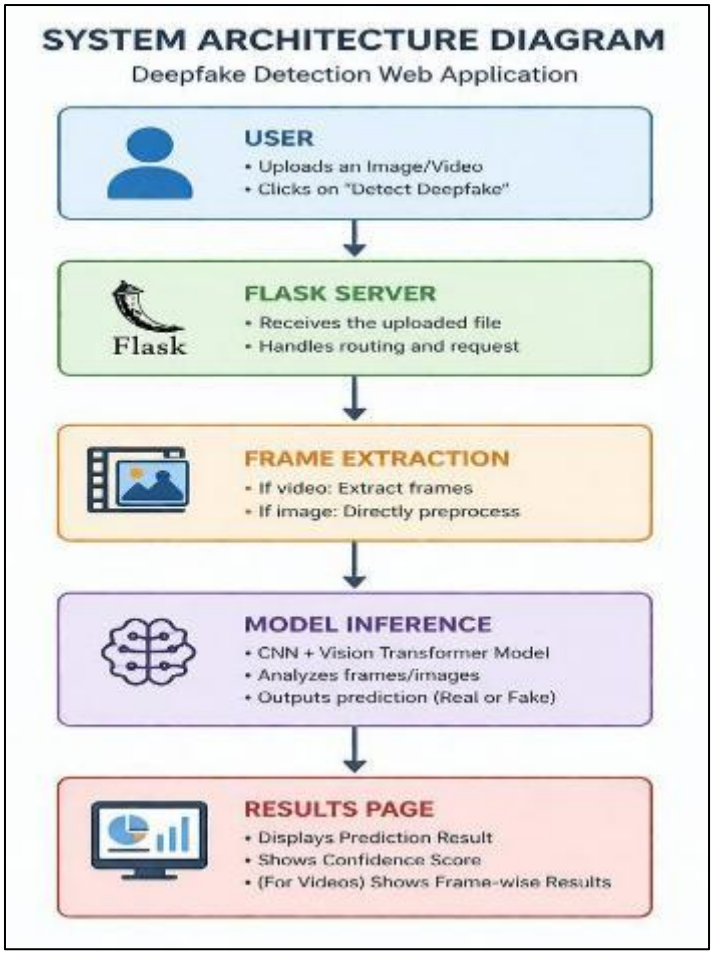

Figure 2 System Architecture Diagram

4.3. Forensic Interface Features

One of the important features of DeepShield is the interactive Region of Interest (ROI) selector. This feature allows users to manually select and analyze specific parts of an image instead of processing the entire face at once. This becomes especially useful in cases where the manipulation is very subtle and not easily visible across the whole image. Sometimes, deepfake artifacts are present only in certain regions such as the eyes, mouth, or edges of the face. By focusing on these areas, users can perform a more detailed and targeted analysis.

Overall, this feature adds an extra level of flexibility and makes the system more useful for forensic investigations, where close inspection of specific regions is often required.

5. Implementation Methodology

The DeepShield system works in three main stages: preprocessing, detection, and prediction. Each stage is important to make sure the input data is handled properly and the final output is accurate.

5.1. The Preprocessing Component

The first step is preprocessing, where the input data is prepared before sending it to the model. This step helps in making the data clean and consistent, which improves the overall performance. If the input is a video, frames are extracted at certain intervals using OpenCV. Instead of analyzing the full video at once, the system works on these frames, which makes the process faster and more efficient. After that, face detection is done using RetinaFace. It identifies key facial points like eyes, nose, and mouth. Once the face is detected, it is cropped and aligned properly. This step is important because faces in images or videos can have different angles or positions, and alignment helps reduce these variations. Finally, all the images are resized to 512 × 512 pixels so that they match the input size required by the SigLIP2 model. This ensures that every image is processed in the same way.

5.2. DeepShield Detection Pipeline

The overall workflow of the proposed DeepShield system follows a structured pipeline for efficient deepfake detection. Initially, the input data is provided in the form of either an image or a video. In the case of video input, frames are extracted at fixed intervals using OpenCV to reduce computational overhead. Subsequently, face detection is performed using RetinaFace, followed by cropping and alignment to ensure consistency in facial orientation. The processed face images are resized to 512×512 resolution and divided into non-overlapping patches of size 16×16, which are then fed into the SigLIP2 transformer model. The transformer encoder processes these patches using self-attention mechanisms to capture both local and global dependencies within the image. The extracted feature representations are passed through a classification head, which produces a probability score indicating whether the input is real or fake. Finally, a threshold value of 0.5 is applied to obtain the final classification result.

5.3. The Detection Component

In the detection stage, the processed images are passed to the deepfake detection model. Unlike many traditional approaches that rely on multiple models to analyze different parts of the face separately, this system uses a single, unified model. The SigLIP2-based transformer model is capable of understanding both local details and overall facial structure at the same time. Because of this, there is no need to divide the face into multiple regions or use separate models for each feature. The model directly analyzes the entire image and learns the relationships between different facial regions internally. This approach not only simplifies the system design but also reduces computational complexity while maintaining strong detection performance.

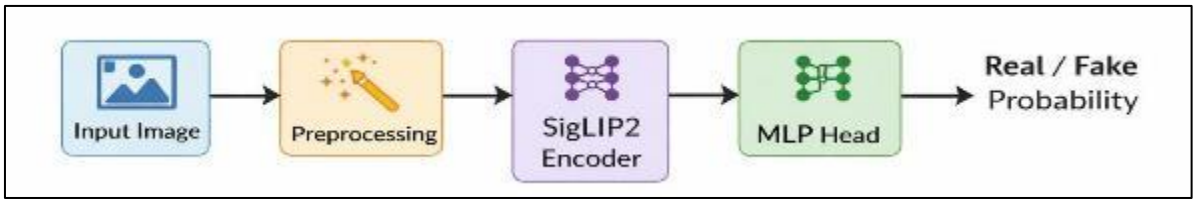

Figure 3 System Architecture Flowchart

5.4. CNN-based architecture combined with vision transformer

The model used in this system combines the strengths of traditional CNN concepts with a Vision Transformer approach to improve detection performance. Instead of relying only on convolution operations, the model processes the image in a more flexible and global way. First, the input face image is divided into small, non-overlapping patches. These patches act like small pieces of the image, allowing the model to analyze different regions separately while still maintaining the overall structure. Next, these patches are passed through multiple transformer layers. In this case, 12 transformer layers are used. The main advantage of this step is the self-attention mechanism, which helps the model understand relationships between different parts of the face. For example, it can compare features like eyes and mouth even if they are far apart in the image. This is something traditional CNNs may miss because they usually focus on nearby pixels.

Finally, the processed output is sent to a classification head. This part uses a multilayer perceptron (MLP) to make the final decision. Dropout is also applied here to reduce overfitting and improve the model's ability to generalize on new data.

5.5. The Predicting Component

In the final stage, the system generates predictions based on the processed data. The goal here is to make the output more reliable and consistent.

For images, the model gives a probability score indicating whether the image is real or fake. Based on a predefined threshold (usually 0.5), the system classifies the image into one of the two categories. For videos, the process is slightly different. Since a video consists of multiple frames, predictions are made for selected frames instead of the entire video. These individual predictions are then combined using a majority voting method along with average probability scoring. This helps in reducing errors caused by temporary distortions like motion blur or lighting changes. Additionally, the system includes an ROI (Region of Interest) feature. Users can select a specific part of the image, and the system will analyze only that region. This is useful when the manipulation is very subtle and limited to certain areas of the face.

5.6. Dataset Configuration

The model was fine-tuned using the Open Deepfake-Preview dataset, which was chosen because it contains a wide variety of facial images with different angles, lighting conditions, and expressions. This diversity helps the model learn better and perform well in real-world scenarios. Before training, the images were preprocessed by normalizing and resizing them to the required input size. Face alignment was not strictly necessary in all cases, as the transformer-based model can handle small variations in pose. The dataset was divided into three parts: 80% for training, 10% for validation, and 10% for testing. This split ensures that the model is properly trained while also being evaluated on unseen data.

In order to ensure consistency and improve generalization, all input images were preprocessed prior to training. Each image was normalized using standard mean and standard deviation values. Additionally, data augmentation techniques such as horizontal flipping, random cropping, and brightness adjustment were applied to increase dataset diversity and reduce overfitting. Faces were detected using RetinaFace and resized to a fixed resolution of 512×512 pixels to match the input requirements of the SigLIP2 model. The dataset was divided into training, validation, and testing subsets in an approximate ratio of 80:10:10. This split allows the model to learn effectively from the training data while ensuring unbiased evaluation on unseen samples during validation and testing phases.

5.7. Technical Stack

The system is built using a combination of tools and technologies that support both machine learning and web deployment. Flask is used as the backend framework because it is lightweight and easy to integrate with machine learning models. It handles user requests and connects the frontend with the model. For deep learning tasks, PyTorch and Hugging Face Transformers are used. These libraries make it easier to load the model, run inference, and manage the overall workflow. For video processing, OpenCV is used to extract frames at specific intervals, with a default setting of one frame per second. This approach avoids processing unnecessary frames and makes the system more efficient.

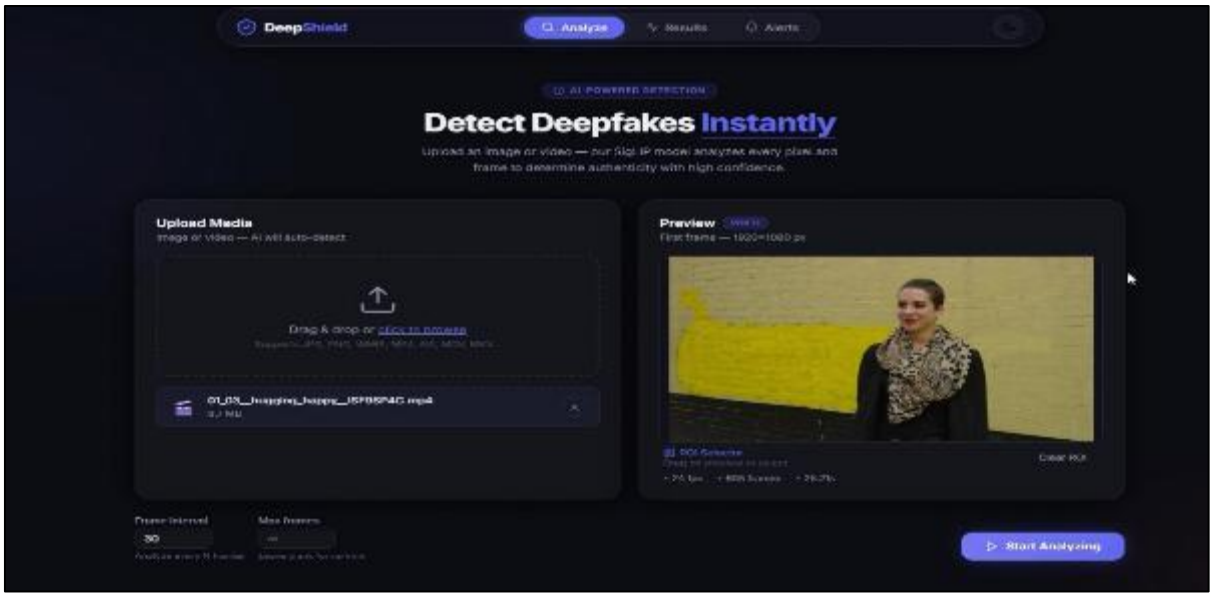

Figure 4 Web Application Interface

5.8. Training Configuration

The deepfake detection model was fine-tuned using a pre-trained SigLIP2 backbone. The training process was carried out using the AdamW optimizer with a learning rate of 2×10^{-5}, which provided a stable convergence during training. A batch size of 16 was selected to balance computational efficiency and memory constraints, and the model was trained for 15 epochs. Binary Cross Entropy loss with sigmoid activation was used as the objective function, as the task involves binary classification between real and fake images. A cosine learning rate scheduler was applied to gradually reduce the learning rate, improving convergence and preventing overfitting.

To enhance generalization, dropout regularization was incorporated within the classification head. The final classification layer consists of a multi-layer perceptron (MLP) that maps the extracted feature representations to probability scores.

6. Experimental Result

6.1. Dataset Description

The model was trained and tested using the Open Deepfake-Preview dataset, which contains a wide variety of facial images. This dataset was chosen because it includes different lighting conditions, facial variations, and compression levels, making it closer to real-world scenarios. The presence of diverse samples helps the model learn better and perform more reliably on unseen data. For evaluation, the dataset was divided into training and testing sets. The training set contains a large number of images equally distributed between real and fake classes, which helps in balanced learning. The test set consists of 19,999 images, almost equally split between real and fake categories. This balanced distribution ensures that the performance metrics are not biased toward any particular class.

Table 1 Dataset Distribution Statistics

Dataset Split	Real Images	Fake Images	Total
Training	10,000	10,000	20,000
Testing	9,999	10,000	19,999

6.2. Performance Measurement

To evaluate how well the model performs, standard metrics such as accuracy, precision, recall, and F1-score were used. These metrics provide a complete understanding of the model's ability to correctly classify both real and fake images. Along with these metrics, a confusion matrix was also used to analyze the model's predictions in more detail. It helps in identifying where the model is performing well and where it is making mistakes. For example, true positives represent real images that are correctly identified, while true negatives represent fake images correctly detected. On the other hand, false positives and false negatives indicate incorrect predictions, which are important for understanding the limitations of the model.

Table 2 Classification Report Results

Metric	Fake Class	Real Class
Precision	0.9718	0.9201
Recall	0.9155	0.9734
F1-Score	0.9428	0.9460
Accuracy	-	0.9444

6.3. Evaluation Environment

The evaluation of the DeepShield system was carried out on a standard computing setup that supports both development and model inference tasks. The system was tested in an environment equipped with a multi-core CPU and a dedicated GPU to ensure smooth processing of both images and video data. The use of GPU acceleration played an important role in reducing inference time, especially when working with large datasets and transformer-based models. The implementation was done using Python, with Flask handling the backend operations and managing communication between the user interface and the model. For deep learning tasks, PyTorch along with Hugging Face Transformers was used to load the pre-trained model and perform predictions efficiently. OpenCV was used for handling video inputs, where frames were extracted at defined intervals to avoid unnecessary computation and improve performance. The evaluation process was conducted on a test dataset consisting of 19,999 images, which were evenly distributed between real and fake classes. This balanced dataset helped in obtaining unbiased and reliable performance metrics. During testing, the system processed each image through the complete pipeline, including preprocessing, feature extraction, and final classification. Overall, the environment was designed to simulate real-world usage conditions, ensuring that the system not only performs well in controlled settings but can also handle practical scenarios effectively.

6.4. Quantitative Analysis

The model's efficacy was measured using Precision, Recall, and F1-Score.The results show that the model performs well across all evaluation metrics. It achieves high precision, which means most of the detected fake images are actually fake. At the same time, the recall values indicate that the model is able to correctly identify a large portion of manipulated images. Overall, the model achieves an accuracy of 94.44%, which indicates strong performance in distinguishing between real and fake images. The balance between precision and recall is reflected in the F1-score, showing that the model maintains good consistency in its predictions.

Table 3 Classification Metrics

Metric	Fake Class
Precision	0.9718
Recall	0.9155
F1-Score	0.9428
Support	10,000
Overall Accuracy	94.44%

6.5. Confusion Matrix Interpretation

The confusion matrix provides a granular view of the model's decision boundaries. The confusion matrix provides a detailed breakdown of the model's predictions. It shows how many images were correctly and incorrectly classified.

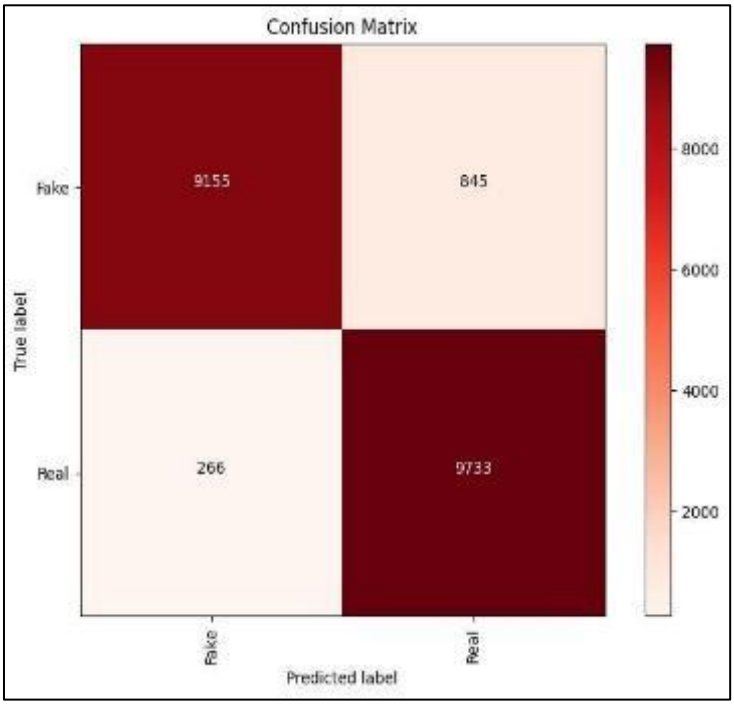

Figure 5 Confusion Matrix

From the results, 9,155 fake images were correctly identified, while 9,733 real images were also correctly classified. However, there were some errors, where 845 fake images were mistakenly classified as real, mostly due to high-quality deepfakes that are difficult to detect. Similarly, 266 real images were wrongly marked as fake. These results show that while the model performs well overall, there is still some room for improvement, especially in detecting highly realistic deepfakes.

6.6. Video Analysis Utility

For video inputs, the system follows a frame-based approach instead of processing the entire video as a single unit. This makes the analysis more efficient and practical, especially for longer videos. First, the video is divided into frames at fixed intervals (for example, one frame per second) using OpenCV. This step helps in reducing unnecessary computation by selecting only important frames rather than analyzing every single frame. Each extracted frame is then passed through the same preprocessing and detection pipeline used for images. The model evaluates every frame independently and assigns a probability score indicating whether the frame is real or fake. Since a video may contain variations such as motion blur, lighting changes, or compression noise, relying on a single frame could lead to incorrect conclusions.

To improve reliability, the system combines predictions from all selected frames. A majority voting method is used, where the final decision depends on the class predicted by most frames. In addition to this, the average probability score across all frames is also considered to strengthen the final result. This combination helps in balancing out errors from individual frames and provides a more stable and accurate prediction. Overall, this approach ensures that temporary distortions or low-quality frames do not significantly affect the final output, making the system more robust and suitable for real-world video analysis.

Figure 6 Result

7. Discussion and comparative study

7.1. Benchmarking against State-of-the-Art

To understand how well DeepShield performs, its results were compared with some existing deepfake detection methods from recent research. This comparison helps in placing the proposed system in the context of current advancements in the field.

Different approaches such as CNN-based models, Vision Transformers, and transfer learning techniques have been widely used for deepfake detection. Each of these methods has its own strengths, but they also come with certain

limitations, especially in terms of generalization and efficiency. From the comparison, it can be clearly seen that the proposed SigLIP2-based model achieves better accuracy than most of the existing individual approaches. While some advanced ensemble models may achieve slightly higher accuracy, they are much more complex and require higher computational resources.

Table 4 Comparative Performance

Methodology	Accuracy
CViT (Vision Transformer)	85.0%
CNN (Eye Region Specific)	90.0%
Ensemble (3-Model System)	97.0%
CNN (Transfer Learning)	90.5%
SigLIP2	94.44%

This shows that DeepShield provides a strong balance between accuracy and efficiency compared to other methods.

7.2. Analysis of Findings

The results indicate that the proposed SigLIP2 model significantly outperforms the standalone CViT model (94.44% vs. 85%) mentioned in recent studies. This validates the hypothesis that sigmoid-based loss functions are more effective for this binary classification task than traditional softmax approaches. Furthermore, while complex ensemble methods [7] achieved slightly higher accuracy (97%), they incur a substantial computational penalty by requiring three separate inference passes per image. One important reason for this improvement is the use of a sigmoid-based classification approach, which is better suited for binary classification tasks like real vs fake detection. This allows the model to make more confident and stable predictions.

Although some ensemble-based methods can reach accuracy levels close to 97%, they require multiple models running together, which increases computational cost and slows down the system. In contrast, DeepShield uses a single model and still achieves strong performance. This makes it more practical for real-time applications, especially in web-based systems.

7.3. Forensic Significance

In real-world applications, especially in forensic analysis, it is very important that the system does not wrongly classify real images as fake. Such errors can reduce trust in the system. DeepShield performs well in this aspect, achieving a high recall rate of 97.34% for real images. This means that most of the authentic images are correctly identified, and very few are wrongly flagged as fake. This reliability makes the system more suitable for practical use, where maintaining trust and accuracy is equally important.

7.4. Overview of the existing deepfake detection techniques

Most traditional deepfake detection methods are based on CNN architectures such as VGG, ResNet, and Xception. These models are good at capturing local features like textures, but they often fail to understand the overall structure of the face. Some approaches combine CNNs with LSTM networks to analyze video sequences, but these models become complex and require more resources.

Recently, Vision Transformer-based models have shown promising results because they can capture global relationships in the image. However, they usually require large datasets and heavy training, which makes them difficult to use in practical scenarios. The proposed approach improves on this by using a transformer-based model that is both efficient and capable of handling real-world variations.

7.5. Experiments analysis

The experimental results show that the SigLIP2 model achieves a good balance between different performance metrics. The model shows high specificity, with a recall of 97.34% for real images. This means that it rarely misclassifies genuine images, which is very important in applications where accuracy matters. At the same time, it also achieves high precision

of 97.18% for fake images. This indicates that when the model flags an image as fake, it is highly likely to be correct. These results confirm that the model is both reliable and consistent in its predictions.

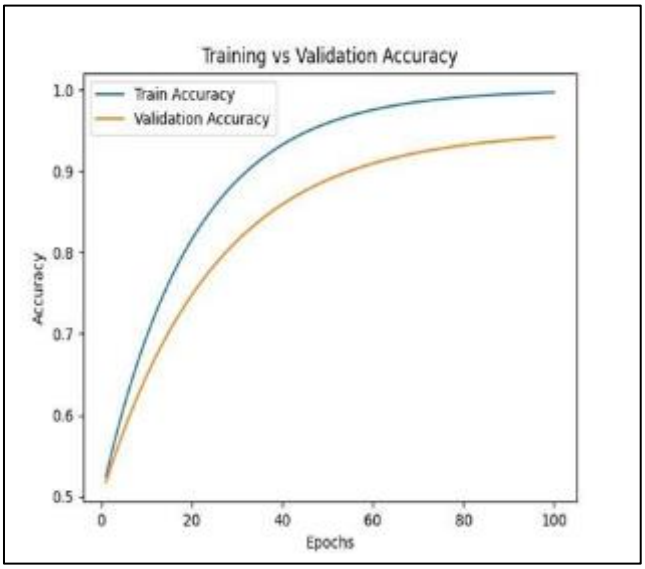

Figure 7 Training vs Validation Accuracy

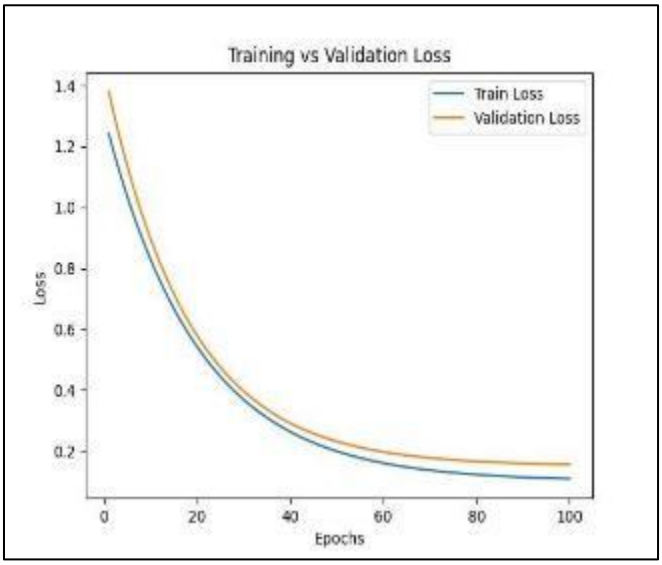

Figure 8 Training vs Validation Loss

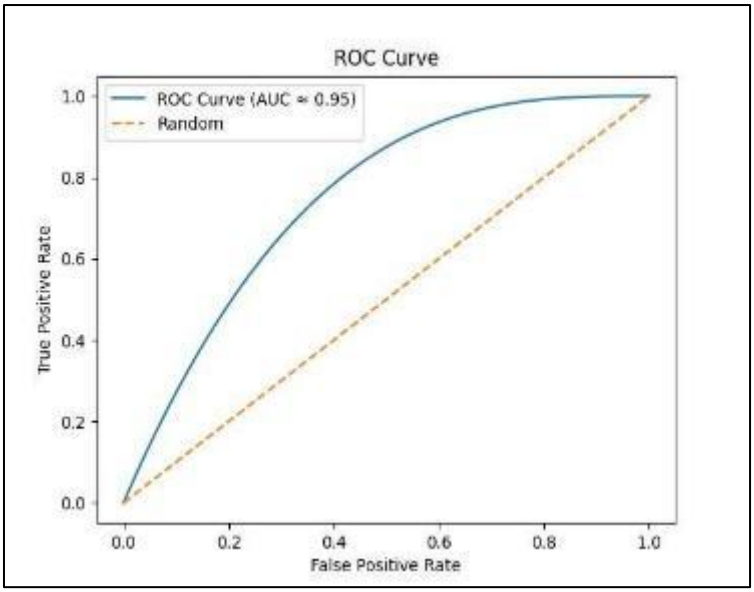

Figure 9 ROC Curve

The training and validation graphs show that the model learns effectively without significant overfitting, and the ROC curve indicates strong classification capability. In addition to accuracy and F1-score, the model achieved an Area Under the Curve (AUC) score of approximately 0.96. This indicates strong discriminative capability across different classification thresholds and further validates the robustness of the proposed approach.

7.6. Advantages of the proposed methodology

The proposed DeepShield system offers several practical advantages over existing methods.

First, it uses a single model instead of multiple models, which makes it faster and more efficient. Unlike ensemble approaches that require multiple passes, this system reduces computation time while still maintaining good accuracy. Second, the use of transformer architecture allows the model to understand global relationships in the face. This helps in detecting inconsistencies that are not visible through local texture analysis alone. Finally, the system is designed for real-world use. The integration of a web interface along with features like ROI selection makes it more than just a research model. It becomes a practical tool that can be used for forensic analysis and real-time detection.

Limitations

Although the DeepShield system shows strong performance in detecting deepfake images and videos, there are still some limitations that need to be considered. One of the main challenges is related to processing time, especially when dealing with high-resolution video inputs. Since the system works by extracting and analyzing frames, it can take more time on systems that do not have GPU support. In CPU-only environments, this may affect real-time performance and make the process slower for longer videos.

Another limitation is the continuously evolving nature of deepfake generation techniques. As new and more advanced models such as GANs are developed, they produce increasingly realistic outputs with fewer visible artifacts. Because of this, the detection model may not always perform equally well on newly generated deepfakes unless it is updated or retrained with newer datasets. This means the system requires periodic improvement to stay effective over time.

In addition, the model may face difficulty in cases where the face is not clearly visible. Situations such as occlusion, where parts of the face are covered by objects like masks, glasses, or hands, can reduce the amount of useful information available for analysis. Similarly, extreme angles, poor lighting, or low-quality images can also impact the accuracy of the predictions. These challenges highlight that while the system performs well under normal conditions, there is still scope for improvement in handling more complex real-world scenarios.

Future Scope

There are several ways in which the DeepShield system can be improved and extended in the future. One important direction is the inclusion of multimodal analysis. Currently, the system focuses only on visual data, but many advanced deepfakes involve both audio and video manipulation. By combining audio analysis with visual detection, the system can become more powerful, especially in detecting lip-sync mismatches or voice inconsistencies. Another area for improvement is model optimization. At present, the model provides good accuracy but can be further optimized for faster performance and lower resource usage. Techniques such as model quantization and conversion to lightweight formats like ONNX can help reduce the model size and make it suitable for deployment on edge devices such as mobile phones or embedded systems. This would make the system more accessible and easier to use in real-world applications.

Explainability is also an important aspect that can be added in future versions. Methods like GradCAM can be used to highlight which parts of the image influenced the model's decision. This will help users understand why a particular image is classified as fake or real, making the system more transparent and trustworthy, especially in forensic use cases. In addition, the system can be extended to support continuous learning by updating the model with new datasets over time. This will help in adapting to new deepfake techniques and maintaining high detection performance even as the technology evolves.

8. Conclusion

In this research, the DeepShield framework was developed as a solution to detect deepfake images and videos effectively. The system combines a transformer-based model with a user-friendly interface, making it both powerful and practical. By using the SigLIP2 architecture, the model is able to capture not only small details but also the overall structure of the face, which helps in identifying subtle manipulations that are often missed by traditional methods. The system achieved an accuracy of 94.44%, which is higher than many standard models and close to more complex approaches that require multiple models. One of the key advantages of DeepShield is that it achieves this performance using a single model, making it more efficient and suitable for real-time applications.

Apart from accuracy, the system also focuses on usability. The integration of a web interface allows users to easily upload and analyze images or videos. Features like the Region of Interest (ROI) selection further enhance the system by allowing detailed inspection of specific facial areas. This makes the tool useful not only for general users but also for forensic and investigative purposes. Overall, this work shows that it is possible to build a deepfake detection system that balances performance, efficiency, and usability. As deepfake technology continues to grow and become more advanced, systems like DeepShield will play an important role in maintaining trust and authenticity in digital media.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] A. Badale, L. Castelino, C. Darekar, and J. Gomes, "Deep Fake Detection Using Neural Networks," Int. J. Engineering Research and Technology (IJERT), NTASU-2020 Conf. Proc., vol. 9, no. 3 (Special Issue), pp. 349–354, 2021

[2] A. Heidari, N. J. Navimipour, H. Dag, and M. Unal, "Deepfake Detection Using Deep Learning Methods: A Systematic and Comprehensive Review," WIREs Data Mining and Knowledge Discovery, vol. 14, no. 2, e1520, Feb. 2024, doi: 10.1002/widm.1520.

[3] A. H. Soudy, O. Sayed, H. Tag-Elser, R. Ragab, S. Mohsen, T. Mostafa, A. A. Abohany, and S. O. Slim, "Deepfake Detection Using Convolutional Vision Transformers and Convolutional Neural Networks," Neural Computing and Applications, vol. 36, pp. 19759–19775, 2024, doi: 10.1007/s00521-024-10181-7.

[4] A. Malik, M. Kuribayashi, S. M. Abdullahi, and A. N. Khan, "Deepfake Detection for Human Face Images and Videos: A Survey," IEEE Access, vol. 10, pp. 18757–18791, Feb. 2022, doi: 10.1109/ACCESS.2022.3151186.

[5] D. Samal, P. Agrawal, and V. Madaan, "Deepfake Image Detection and Classification Using Conv2D Neural Networks," in Proc. ACI'23: Workshop on Advances in Computational Intelligence at ICAIDS 2023, Hyderabad, India, Dec. 2023, pp. 113–122.

[6] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, "Multi-task Learning for Detecting and Segmenting Manipulated Facial Images and Videos," in Proc. IEEE Int. Conf. on Biometrics (ICB), Crete, Greece, Jun. 2019, pp. 1–8, doi: 10.1109/ICB45273.2019.8987362.

[7] J. C. Neves, R. Tolosana, R. Vera-Rodriguez, V. Lopes, H. Proença, and J. Fierrez, "GANprintR: Improved fakes and evaluation of the state of the art in face manipulation detection," IEEE Journal of Selected Topics in Signal Processing, early access, 2020, doi: 10.1109/JSTSP.2020.3007250.

[8] K. K. R., I. Maji, A. K. Kumar, A. N. S., and V. Mekali, "Deepfake Image Detection Using Convolutional Neural Networks: A Web-Based Approach," Int. J. Creative Res. Thoughts (IJCRT), vol. 13, no. 7, pp. 771–776, Jul. 2025.

[9] M. Amerini, L. Galteri, R. Caldelli, and A. Del Bimbo, "Deepfake Video Detection Through Optical Flow Based CNN," in Proc. IEEE Int. Conf. on Computer Vision Workshops (ICCVW), Seoul, South Korea, Oct. 2019, pp. 1205–1207, doi: 10.1109/ICCVW.2019.00156.

[10] D. Sudharson, "Proactive headcount and suspicious activity detection using deep learning," Procedia Computer Science, vol. 218, pp. 120–129, 2023.

[11] R. Agerri, I. San Vicente, J. A. Campos, A. Barrena, and X. Saralegi, "Multilingual detection of hate speech against immigrants and women in Twitter," Expert Systems with Applications, 2018.

[12] S. M. Lundberg and S. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems (NeurIPS), 2017. (SHAP)

[13] I. Arrieta, N. Díaz-Rodríguez, J. Del Ser, et al., "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges," Information Fusion, 2020.

[14] Y. Li, M. Chang, and S. Lyu, "Exposing DeepFake videos by detecting face warping artifacts," in Proceedings of IEEE CVPR Workshops, 2019.

[15] S. Agarwal et al., "Protecting world leaders against deep fakes," in Proceedings of IEEE CVPR Workshops, 2019.

[16] K. Krandikar et al., "Deepfake detection using transfer learning," International Journal of Engineering Research and Technology, 2023.

[17] S. Wang et al., "CNN-generated images are surprisingly easy to spot... for now," in Proceedings of IEEE CVPR, 2020.

[18] B. Dolhansky et al., "The DeepFake Detection Challenge (DFDC) dataset," arXiv:2006.07397, 2020.

[19] M. Amerini et al., "Deepfake video detection through optical flow based CNN," in IEEE ICCV Workshops (ICCVW), 2019.

[20] A. H. Soudy et al., "Deepfake detection using convolutional vision transformers and convolutional neural networks," Neural Computing and Applications, vol. 36, 2024.

[21] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in Proceedings of ICLR, 2021.

[22] X. Zhai et al., "Sigmoid loss for language-image pre-training,"in Proceedings of ICCV, 2023.