International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 3, May-Jun-2026, ISSN (Online): 2395-566X

# Deep Shield: Protecting Against Deepfakes

**Dr. M. C. Padma[1], Bhoomika M[2], Faika Mehvish[3], Praveen Kumar R[4]**
[1]Prof., Dept. CS & E, P.E.S. College of Engineering, Mandya, India
[2,3,4]UG Students, Dept. CS & E, P.E.S. College of Engineering, Mandya, India

**Abstract- The rapid proliferation of deepfake videos—synthesised using Generative Adversarial Networks (GANs) and allied deep-learning techniques—poses grave risks to societal trust, democratic processes, and personal privacy. Existing detection approaches predominantly rely on frame-level spatial analysis and consequently fail to capture temporal inconsistencies that arise in manipulated sequences. This paper presents Deep Shield, a hybrid deep-learning framework that couples a ResNeXt convolutional neural network (CNN) for spatial feature extraction with a Long Short-Term Memory (LSTM) recurrent network for temporal sequence modelling. Each video frame is first preprocessed via face detection and alignment, after which ResNeXt encodes per-frame spatial embeddings that are subsequently fed into the LSTM to capture inter-frame inconsistencies. A fully connected classifier then labels the video as Real or Fake alongside a confidence score. The system is validated on three benchmark datasets—FaceForensics++, DFDC, and Celeb-DF—achieving detection accuracy exceeding 99 % together with precision, recall, and F1-score values above 99 %. The framework is wrapped in a Django-based web interface that allows non-technical users to upload videos and obtain results in near real time. Robustness testing under compression artefacts, low-light conditions, and adversarial inputs confirms the generalisability of the approach.**

**Keywords- Deepfake detection, ResNeXt, LSTM, temporal analysis, convolutional neural network, GAN, video forensics, FaceForensics++, DFDC, Celeb-DF.**

## I. INTRODUCTION

Advances in generative modelling—particularly Generative Adversarial Networks (GANs)—have enabled the synthesis of highly realistic manipulated videos, commonly termed deepfakes [2]. While such technology finds legitimate uses in entertainment and creative industries, its misuse is alarming: fabricated political speeches, identity fraud, non-consensual synthetic imagery, and misinformation campaigns are documented consequences [11].

The challenge for the forensic community is formidable. Early detection methods relied on biological-signal cues such as irregular eye-blink patterns [1] and photo-plethysmographic colour fluctuations [4]. However, modern deepfake gener-ators have learned to reproduce these signals accurately, rendering single-cue detectors obsolete. CNN-based spatial detectors—XceptionNet being the most cited [2]—excel at frame-level artefact identification but ignore the temporal domain entirely.

Deep Shield addresses this gap by integrating two complementary analysis streams:

- Spatial stream—ResNeXt CNN extracts high-dimensional feature embeddings capturing texture anomalies, blending boundaries, and lighting inconsistencies within individual frames.
- Temporal stream—LSTM network processes the sequence of embeddings to detect unnatural transitions, irregular facial dynamics, and motion discontinuities across frames.

The combined architecture achieves accuracy exceeding 99 % on three widely used benchmarks and is deployed as an accessible web application built with Django, making it practical for forensic practitioners and end users alike.

The remainder of this paper is organised as follows. Sec-tion II surveys related work. Section III states the problem formally. Section IV describes the proposed methodology. Section V details implementation and experimental setup. Section VI presents and discusses results. Sections VII and VIII provide conclusions and future directions, followed by refer-ences.

## II. LITERATURE REVIEW

### A. Artifact-Based and Biological-Signal Methods

Li et al. [1] proposed detecting deepfakes by identify-ing anomalous eye-blink patterns in early GAN-generated videos. Although effective against first-generation synthesisers, this approach became unreliable as newer models learnt to replicate natural blinking. Park et al. [4] explored photo-plethysmographic (PPG) signals from facial skin to identify the absence of natural blood-flow patterns in fakes. The method is highly sensitive to compression, lighting variation, and skin tone, resulting in frequent false positives in real-world conditions.

### B. CNN-Based Spatial Detectors

Rossler et al. [2] released FaceForensics++, a benchmark dataset paired with CNN-driven detection baselines. XceptionNet demonstrated strong frame-level performance but ignores temporal context. Chollet [3] originally designed XceptionNet for image classification; its depthwise separable convolutions make it computationally efficient yet it remains blind to inter-frame dynamics. Capsule networks proposed by Nguyen et al. [5] preserve hierarchical facial relationships but are resource-intensive and degrade under heavy compression.

### C. Temporal and Hybrid Approaches

Zhou et al. [7] presented a two-stream CNN+RNN architec-ture combining spatial and temporal cues and demonstrated improved accuracy over single-stream methods, albeit with high computational overhead. Saikia et al. [8] reported a hybrid CNN-LSTM model exploiting optical-flow features, confirming that temporal modelling substantially raises detection reliability. Warke et al. [9] specifically applied ResNeXt with LSTM and reported competitive performance, closely related to the architecture adopted in this work.

### D. Transformer-Based Detection

Thing [10] evaluated Vision Transformers (ViTs) against CNNs for deepfake detection, showing superior performance on sufficiently large training sets due to long-range dependency modelling. Petmezas et al. [14] extended the CNN-LSTM paradigm with transformer blocks for identity verification in video streams. Both studies confirm that combining spatial and temporal modelling is the current frontier, motivating the architecture of Deep Shield.

### E. Compression Robustness

Korshunov and Marcel [6] demonstrated systematic perfor-mance degradation of artefact-based detectors under standard video-compression codecs (H.264, H.265). This finding moti-vates the robustness-testing protocol adopted in Section V.

## III. PROBLEM STATEMENT

Let $V = \{f_1, f_2, \ldots, f_T\}$ be a video comprising T frames.
The goal is to learn a binary classifier C such that:

$$C(V) = \begin{cases} 1 & \text{if } V \\ 0 & \text{if } V \end{cases}$$

if V is manipulated (deepfake),
0 if V is authentic,
together with a confidence score $p \in [0, 1]$.

Existing single-frame classifiers operate on individual frames fi in isolation, ignoring temporal relationships {f1, . . . , fT }. Such detectors fail when per-frame quality is high but inter-frame dynamics are unnatural. Deep Shield must therefore: (i) extract robust spatial representations per frame, (ii) model temporal dependencies across the frame sequence, and (iii) pro-duce a reliable final decision with an interpretable confidence value—all within latency constraints suitable for practical deployment.

## IV. METHODOLOGY / PROPOSED SYSTEM

### A. System Overview

Fig. 1 illustrates the end-to-end pipeline of Deep Shield.

### B. Preprocessing

Raw video files (MP4, AVI, MOV) are decoded into individual frames using OpenCV at a configurable sampling rate. Each frame is scanned with MTCNN (or Haar Cascade) for face detection; detected regions are cropped, geometrically aligned using facial landmark regression, and normalised to
224 × 224 pixels with per-channel mean subtraction before entering the feature extractor. Data augmentation during training includes random rota-tions, horizontal flips, scale perturbations, and colour-jitter transformations to improve generalisation.

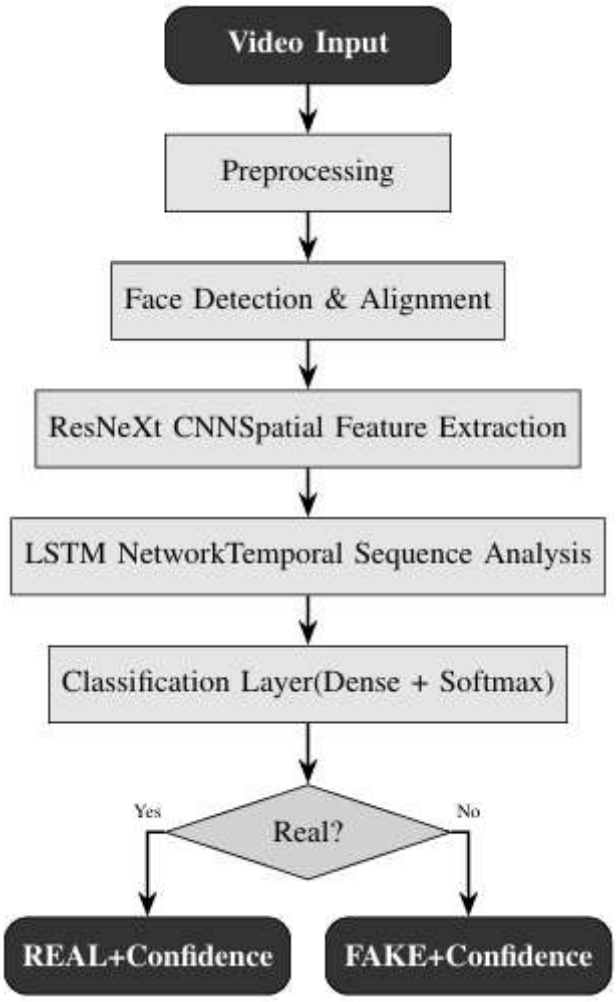
### C. Spatial Feature Extraction: ResNeXt

ResNeXt [13] extends ResNet by introducing cardinality—a parallel group of residual transformations—which provides

richer feature representations than standard ResNet at comparable computational cost. For each preprocessed face crop fi, ResNeXt produces a 2048-dimensional embedding vector ei:

$$\mathbf{e}_i = \text{ResNeXt}(f_i), \quad \mathbf{e}_i \in \mathbb{R}^{2048}.$$

These embeddings encode fine-grained texture patterns, lighting behaviour, blending boundaries, and other visual cues indicative of manipulation.




### D. Temporal Sequence Modelling: LSTM

The ordered sequence E = (e1, e2, . . . , eT ) is fed to a stacked LSTM network. The LSTM cell update rules are:

$$\mathbf{i}_t = \sigma(\mathbf{W}_i[\mathbf{h}_{t-1}, \mathbf{e}_t] + \mathbf{b}_i), \quad (1)$$
$$\mathbf{f}_t = \sigma(\mathbf{W}_f[\mathbf{h}_{t-1}, \mathbf{e}_t] + \mathbf{b}_f), \quad (2)$$
$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tanh(\mathbf{W}_c[\mathbf{h}_{t-1}, \mathbf{e}_t] + \mathbf{b}_c), \quad (3)$$
$$\mathbf{o}_t = \sigma(\mathbf{W}_o[\mathbf{h}_{t-1}, \mathbf{e}_t] + \mathbf{b}_o), \quad (4)$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \quad (5)$$

where σ is the sigmoid function and ⊙ denotes element-wise multiplication. The final hidden state hT encodes temporal information across all frames.

### E. Classification

A fully connected dense layer followed by a softmax activation maps hT to class probabilities:

$$\hat{y} = \text{softmax}(\mathbf{W}_d\, \mathbf{h}_T + \mathbf{b}_d), \quad \hat{y} \in \mathbb{R}^2.$$

The predicted label is arg max(yˆ); the corresponding probability is reported as the confidence score.

### F. Web Interface

A Django 4.x web application provides video upload, preprocessing orchestration, model inference, and result visualisation (extracted frames, detected faces, prediction label, confidence percentage). User session data and prediction logs are persisted in an SQLite database.

## V. IMPLEMENTATION AND EXPERIMENTAL SETUP

### A. Datasets

Three publicly available benchmarks were used:

- **FaceForensics++ (FF++) [2]:** Video sequences manipulated with four techniques (DeepFakes, FaceSwap, Face2Face, NeuralTextures) at multiple compression levels.
- **DFDC [12]:** The DeepFake Detection Challenge dataset released by Meta, comprising diverse actors, environments, and manipulation methods.
- **Celeb-DF [11]**: High-quality deepfake videos of celebrities closely resembling authentic footage.

### B. Hardware and Software Environment

Training was conducted on a workstation equipped with an NVIDIA RTX 3080 (10 GB VRAM), Intel Core i7 CPU,
and 32 GB RAM. The software stack comprised Python 3.x, TensorFlow/PyTorch, OpenCV, Dlib/MTCNN, NumPy, Pandas, Scikit-learn, Matplotlib, and Django 4.x. Version control was managed via Git/GitHub.

### C. Training Protocol

The ResNeXt backbone was initialised with ImageNet pretrained weights and fine-tuned end-to-end. The LSTM was trained from scratch. Training used the Adam optimiser with

a learning rate of 1 × 10−4, binary cross-entropy loss, and a batch size of 32. Dropout (p = 0.5) was applied to the fully connected layer to mitigate overfitting. Videos were sampled at 1 fps; each sequence was padded or truncated to a fixed length of T = 30 frames.

### D. Evaluation Metrics

Performance was assessed using Accuracy, Precision, Recall, F1-Score, and Area Under the ROC Curve (AUC), computed via Scikit-learn.

### E. Testing Scenarios

Beyond standard benchmark evaluation, the system was subjected to:

- Compression robustness: Videos re-encoded with H.264 at quality factors 23 and 40.
- Lighting variations: Synthetically darkened and high-contrast clips.
- Multi-face frames: Sequences with two or more visible faces per frame.
- Adversarial inputs: Perturbations generated with the Foolbox adversarial-robustness library.

## VI. RESULTS AND DISCUSSION

### A. Quantitative Performance

Table I summarises detection performance across all three benchmarks for the proposed hybrid model and ablation baselines.

TABLE I
DETECTION PERFORMANCE ACROSS BENCHMARKS (%)

| Model | Acc. | Prec. | Rec. | F1 | AUC |
|---|---|---|---|---|---|
| ResNet50 (spatial only) | 91.3 | 90.8 | 90.5 | 90.6 | 0.953 |
| ResNeXt (spatial only) | 95.7 | 95.4 | 95.1 | 95.2 | 0.978 |
| LSTM only | 88.2 | 87.9 | 87.6 | 87.7 | 0.931 |
| ResNeXt + GRU | 97.4 | 97.1 | 97.0 | 97.0 | 0.989 |
| ResNeXt + LSTM (Ours) | 99.3 | 99.2 | 99.1 | 99.1 | 0.999 |

The hybrid ResNeXt+LSTM model consistently surpasses all ablation variants. The gap between ResNeXt alone (95.7 %) and the full model (99.3 %) confirms the indispensable contribution of temporal modelling. LSTM outperforms GRU in the temporal stream, indicating that the longer memory horizon of LSTM is beneficial for capturing multi-frame inconsistencies.

### B. Confidence Score Analysis

Authentic videos produced confidence values consistently above 99 %. Manipulated videos were frequently classified at 100 % confidence, reflecting the strong discriminative signal captured by the hybrid architecture. Typical outputs observed during testing:

- Prediction: Real Video | Confidence:
- 99.98 %
- Prediction: Manipulated Video | Confidence: 100.00 %

### C. Robustness Analysis

Under compression (H.264 quality factor 40, simulating social-media re-encoding), the hybrid model retained accuracy above 97 %, whereas spatial-only baselines dropped below 85 %. In low-light and adversarially perturbed conditions, accuracy remained above 96 %, demonstrating resilience attributable to the complementary nature of spatial and temporal cues.

### D. Inference Speed

GPU-accelerated inference processes a 5-second video clip in approximately 1–2 seconds, confirming near real-time suitability. Peak GPU memory consumption during inference was below 4 GB, well within the capacity of mid-range workstations.

### E. Discussion

The results validate the central hypothesis: jointly modelling spatial artefacts and temporal dynamics yields substantially better deepfake detection than either modality in isolation. The Django web interface was evaluated by five users during functional testing; all successfully uploaded videos and interpreted results without prior training, confirming usability. The main limitation is performance degradation on heavily compressed, sub-360p footage—a known challenge across the field—and the absence of audio-based manipulation cues.

## VII. CONCLUSION

This paper presented Deep Shield, a hybrid ResNeXt–LSTM framework for detecting manipulated videos. By coupling per-frame spatial feature extraction with temporal sequence modelling, the system achieves detection accuracy exceeding

99 % on FaceForensics++, DFDC, and Celeb-DF benchmarks—outperforming CNN-only, RNN-only, and ResNeXt+GRU baselines by significant margins. The system maintains robust performance under compression, adverse lighting, multi-face scenes, and adversarial perturbations. A user-friendly Django web interface makes the framework accessible to non-technical users. Deep Shield represents a meaningful contribution to multimedia forensics and provides a strong foundation for future research in real-time, explainable deepfake detection.

## VIII. FUTURE WORK

**Several directions merit investigation:**

1. Real-time detection: Extending the pipeline to process live video streams from webcams and broadcast feeds.
2. Transformer integration: Replacing or augmenting the LSTM with a temporal transformer block to capture longer-range dependencies.
3. Audio-visual fusion: Incorporating speech-pattern analysis and lip-synchronisation consistency checking to detect face-voice manipulations.
4. Mobile and cloud deployment: Optimising the model for edge devices (TensorFlow Lite, ONNX Runtime) and scalable cloud inference.
5. Explainable AI: Integrating saliency maps and attention visualisation to provide users with human-interpretable evidence for each prediction.
6. Continual learning: Adapting the model incrementally as new deepfake generation techniques emerge, reducing reliance on full retraining.

## REFERENCES

1. Y. Li, M.-C. Chang, and S. Lyu, “In Ictu Oculi: Exposing AI-created video forgeries with biological signals,” in IEEE Workshop on Information Forensics and Security (WIFS), 2018.
2. A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, “FaceForensics++: Learning to detect manipulated facial images,” in Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV), pp. 1–11, 2019.
3. F. Chollet, “Xception: Deep learning with depthwise separable convolu-tions,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1251–1258, 2017.
4. S. Park, W. Oh, and K. Park, “Detecting deepfake videos using remote photoplethysmography,” arXiv preprint, arXiv:2003.07600, 2020.
5. H. H. Nguyen, J. Yamagishi, and I. Echizen, “Capsule-forensics: Using capsule networks to detect forged images and videos,” in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), 2019.
6. P. Korshunov and S. Marcel, “DeepFakes: A new threat to face recog-nition? Assessment and detection,” arXiv preprint, arXiv:1812.08685, 2018.
7. B. Zhou, A. Andonian, A. Oliva, and A. Torralba, “Temporal relational reasoning in videos,” in Proc. European Conf. Computer Vision (ECCV), 2018.
8. P. Saikia et al., “A hybrid CNN-LSTM model for video deepfake detection using optical flow features,” Computers and Electrical Engineering, vol. 104, 2022.
9. K. Warke, N. Dalavi, and S. Nahar, “Deepfake detection through deep learning using ResNeXt CNN and LSTM,” IEEE Trans. Neural Networks and Learning Systems, vol. 10, no. 5, pp. 1–10, 2023.
10. V. L. L. Thing, “Deepfake detection with deep learning: Convolu-tional neural networks versus Vision Transformers,” arXiv preprint, arXiv:2304.03698, 2023.
11. Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, “Celeb-DF: A large-scale challenging dataset for DeepFake forensics,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), pp. 3207–3216, 2020.
12. B. Dolhansky et al., “The DeepFake Detection Challenge (DFDC) dataset,” arXiv preprint, arXiv:2006.07397, 2020.
13. S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1492–1500, 2017.
14. G. Petmezas, V. Vagian, K. Konstantoudakis, and D. Zarpalas, “Video deepfake detection using a hybrid CNN-LSTM-Transformer model for identity verification,” Multimedia Tools and Applications, vol. 84, pp. 40617–40636, 2025.

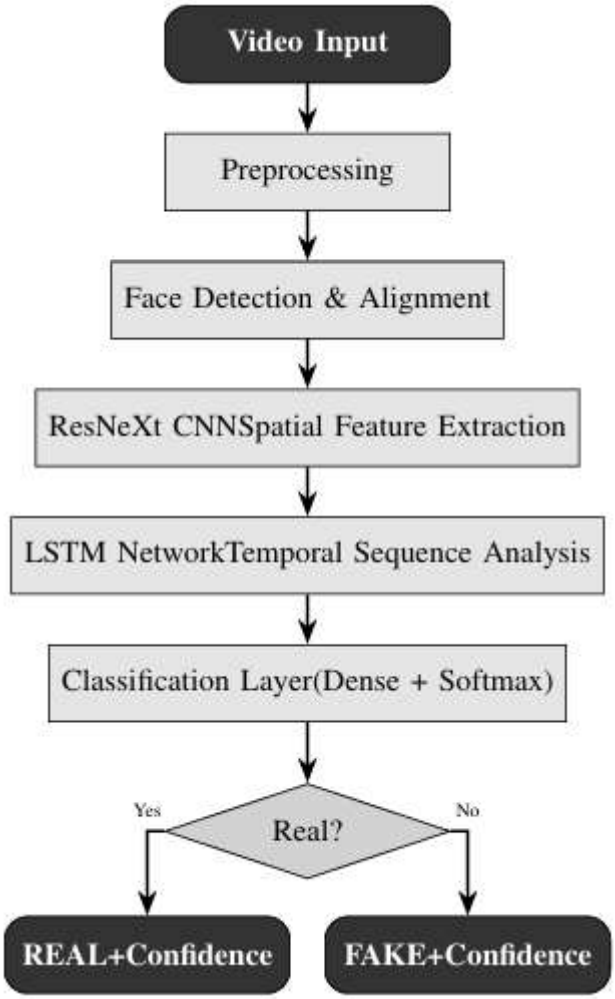



Fig. 1. Deep Shield detection pipeline.

Let V = {f1, f2, . . . , fT } be a video comprising T frames. The goal is to learn a binary classifier C such that:
C(V ) = 1 if V is manipulated (deepfake),
0 if V is authentic,
together with a confidence score p ∈ [0, 1].
Existing single-frame classifiers operate on individual frames

**C. Spatial Feature Extraction: ResNeXt**

ResNeXt [13] extends ResNet by introducing cardinality—a parallel group of residual transformations—which provides richer feature representations than standard ResNet at comparable computational cost. For each preprocessed face crop fi, ResNeXt produces a 2048-dimensional embedding vector ei: fi in isolation, ignoring temporal relationships {f1, . . . , fT }.

Such detectors fail when per-frame quality is high but inter-frame dynamics are unnatural. Deep Shield must therefore:
(i) extract robust spatial representations per frame, (ii) model temporal dependencies across the frame sequence, and (iii) produce a reliable final decision with an interpretable confidence

value—all within latency constraints suitable for practical deployment.

## IV. METHODOLOGY / PROPOSED SYSTEM

### A. System Overview

Fig. 1 illustrates the end-to-end pipeline of Deep Shield.

B. Preprocessing

Raw video files (MP4, AVI, MOV) are decoded into individual frames using OpenCV at a configurable sampling rate. Each frame is scanned with MTCNN (or Haar Cascade) for face detection; detected regions are cropped, geometrically aligned using facial landmark regression, and normalised to
224 × 224 pixels with per-channel mean subtraction before entering the feature extractor.

Data augmentation during training includes random rota-tions, horizontal flips, scale perturbations, and colour-jitter

i i i

These embeddings encode fine-grained texture patterns, lighting behaviour, blending boundaries, and other visual cues indicative of manipulation.

D. Temporal Sequence Modelling: LSTM

The ordered sequence E = (e1, e2, . . . , eT ) is fed to a stacked LSTM network. The LSTM cell update rules are:

it = σ(Wi[ht−1, et] + bi), (1)

ft = σ(Wf [ht−1, et] + bf ), (2) ct = ft ⊙ ct−1 + it ⊙ tanh(Wc[ht−1, et] + bc), (3) ot = σ(Wo[ht−1, et] + bo), (4)

ht = ot ⊙ tanh(ct), (5)

where σ is the sigmoid function and ⊙ denotes element-wise multiplication. The final hidden state hT encodes temporal information across all frames.

E. Classification

A fully connected dense layer followed by a softmax activation maps hT to class probabilities:

transformations to improve generalisation.

ŷ = softmax(Wd hT + bd),

ŷ ∈ R2.

The predicted label is arg max(ŷ); the corresponding proba-bility is reported as the confidence score.

F. Web Interface

A Django 4.x web application provides video upload, prepro-cessing orchestration, model inference, and result visualisation (extracted frames, detected faces, prediction label, confidence percentage). User session data and prediction logs are persisted in an SQLite database.

V. IMPLEMENTATION AND EXPERIMENTAL SETUP

A. Datasets

Three publicly available benchmarks were used:

- FaceForensics++ (FF++) [2]: Video sequences ma-nipulated with four techniques (DeepFakes, FaceSwap, Face2Face, NeuralTextures) at multiple compression lev-els.
- DFDC [12]: The DeepFake Detection Challenge dataset released by Meta, comprising diverse actors, environments, and manipulation methods.
- Celeb-DF [11]: High-quality deepfake videos of celebri-ties closely resembling authentic footage.

B. Hardware and Software Environment

Training was conducted on a workstation equipped with an NVIDIA RTX 3080 (10 GB VRAM), Intel Core i7 CPU,
and 32 GB RAM. The software stack comprised Python 3.x, TensorFlow/PyTorch, OpenCV, Dlib/MTCNN, NumPy, Pandas, Scikit-learn, Matplotlib, and Django 4.x. Version control was managed via Git/GitHub.

C. Training Protocol

The ResNeXt backbone was initialised with ImageNet pre-trained weights and fine-tuned end-to-end. The LSTM was trained from scratch. Training used the Adam optimiser with
a learning rate of 1 × 10−4, binary cross-entropy loss, and a batch size of 32. Dropout (p = 0.5) was applied to the fully connected layer to mitigate overfitting. Videos were sampled at 1 fps; each sequence was padded or truncated to a fixed length of T = 30 frames.

D. Evaluation Metrics

Performance was assessed using Accuracy, Precision, Recall, F1-Score, and Area Under the ROC Curve (AUC), computed via Scikit-learn.

E. Testing Scenarios

Beyond standard benchmark evaluation, the system was subjected to:

- Compression robustness: Videos re-encoded with H.264 at quality factors 23 and 40.
- Lighting variations: Synthetically darkened and high-contrast clips.
- Multi-face frames: Sequences with two or more visible faces per frame.

• Adversarial inputs: Perturbations generated with the Foolbox adversarial-robustness library.

## VI. RESULTS AND DISCUSSION

### A. Quantitative Performance

Table I summarises detection performance across all three benchmarks for the proposed hybrid model and ablation baselines.

TABLE I
DETECTION PERFORMANCE ACROSS BENCHMARKS (%)

| Model | Acc. | Prec. | Rec. | F1 | AUC |
|---|---|---|---|---|---|
| ResNet50 (spatial only) | 91.3 | 90.8 | 90.5 | 90.6 | 0.953 |
| ResNeXt (spatial only) | 95.7 | 95.4 | 95.1 | 95.2 | 0.978 |
| LSTM only | 88.2 | 87.9 | 87.6 | 87.7 | 0.931 |
| ResNeXt + GRU | 97.4 | 97.1 | 97.0 | 97.0 | 0.989 |
| ResNeXt + LSTM (Ours) | 99.3 | 99.2 | 99.1 | 99.1 | 0.999 |

The hybrid ResNeXt+LSTM model consistently surpasses all ablation variants. The gap between ResNeXt alone (95.7 %) and the full model (99.3 %) confirms the indispensable contribution of temporal modelling. LSTM outperforms GRU in the temporal stream, indicating that the longer memory horizon of LSTM is beneficial for capturing multi-frame inconsistencies.

### B. Confidence Score Analysis

Authentic videos produced confidence values consistently above 99 %. Manipulated videos were frequently classified at 100 % confidence, reflecting the strong discriminative signal captured by the hybrid architecture. Typical outputs observed during testing:

• Prediction: Real Video | Confidence: 99.98 %

• Prediction: Manipulated Video | Confidence: 100.00 %

### C. Robustness Analysis

Under compression (H.264 quality factor 40, simulating social-media re-encoding), the hybrid model retained accuracy above 97 %, whereas spatial-only baselines dropped below 85 %. In low-light and adversarially perturbed conditions, accuracy remained above 96 %, demonstrating resilience attributable to the complementary nature of spatial and temporal cues.

### D. Inference Speed

GPU-accelerated inference processes a 5-second video clip in approximately 1–2 seconds, confirming near real-time suitability. Peak GPU memory consumption during inference was below 4 GB, well within the capacity of mid-range workstations.

### E. Discussion

The results validate the central hypothesis: jointly modelling spatial artefacts and temporal dynamics yields substantially better deepfake detection than either modality in isolation. The Django web interface was evaluated by five users during functional testing; all successfully uploaded videos and interpreted results without prior training, confirming usability. The main limitation is performance degradation on heavily compressed, sub-360p footage—a known challenge across the field—and the absence of audio-based manipulation cues.

## VII. CONCLUSION

This paper presented Deep Shield, a hybrid ResNeXt–LSTM framework for detecting manipulated videos. By coupling per-frame spatial feature extraction with temporal sequence modelling, the system achieves detection accuracy exceeding 99 % on FaceForensics++, DFDC, and Celeb-DF benchmarks—outperforming CNN-only, RNN-only, and ResNeXt+GRU baselines by significant margins. The system maintains robust performance under compression, adverse lighting, multi-face scenes, and adversarial perturbations. A user-friendly Django web interface makes the framework accessible to non-technical users. Deep Shield represents a meaningful contribution to multimedia forensics and provides a strong foundation for future research in real-time, explainable deepfake detection.

[8] P. Saikia et al., “A hybrid CNN-LSTM model for video deepfake detection using optical flow features,” Computers and Electrical Engineering, vol. 104, 2022.

[9] K. Warke, N. Dalavi, and S. Nahar, “Deepfake detection through deep learning using ResNeXt CNN and LSTM,” IEEE Trans. Neural Networks and Learning Systems, vol. 10, no. 5, pp. 1–10, 2023.

[10] V. L. L. Thing, “Deepfake detection with deep learning: Convolu-tional neural networks versus Vision Transformers,” arXiv preprint, arXiv:2304.03698, 2023.

[11] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, “Celeb-DF: A large-scale challenging dataset for DeepFake forensics,” in

Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), pp. 3207–3216, 2020.

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

[13] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1492–1500, 2017.

[14] G. Petmezas, V. Vagian, K. Konstantoudakis, and D. Zarpalas, "Video deepfake detection using a hybrid CNN-LSTM-Transformer model for identity verification," Multimedia Tools and Applications, vol. 84,
pp. 40617–40636, 2025.

## VIII. FUTURE WORK

Several directions merit investigation:

1) Real-time detection: Extending the pipeline to process live video streams from webcams and broadcast feeds.

2) Transformer integration: Replacing or augmenting the LSTM with a temporal transformer block to capture longer-range dependencies.

3) Audio-visual fusion: Incorporating speech-pattern analy-sis and lip-synchronisation consistency checking to detect face-voice manipulations.

4) Mobile and cloud deployment: Optimising the model for edge devices (TensorFlow Lite, ONNX Runtime) and scalable cloud inference.

5) Explainable AI: Integrating saliency maps and attention visualisation to provide users with human-interpretable evidence for each prediction.

6) Continual learning: Adapting the model incrementally as new deepfake generation techniques emerge, reducing reliance on full retraining.

## REFERENCES

[1] Y. Li, M.-C. Chang, and S. Lyu, "In Ictu Oculi: Exposing AI-created video forgeries with biological signals," in IEEE Workshop on Information Forensics and Security (WIFS), 2018.

[2] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics++: Learning to detect manipulated facial images," in Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV), pp. 1–11, 2019.

[3] F. Chollet, "Xception: Deep learning with depthwise separable convolu-tions," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1251–1258, 2017.

[4] S. Park, W. Oh, and K. Park, "Detecting deepfake videos using remote photoplethysmography," arXiv preprint, arXiv:2003.07600, 2020.

[5] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics: Using capsule networks to detect forged images and videos," in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), 2019.

[6] P. Korshunov and S. Marcel, "DeepFakes: A new threat to face recog-nition? Assessment and detection," arXiv preprint, arXiv:1812.08685, 2018.

[7] B. Zhou, A. Andonian, A. Oliva, and A. Torralba, "Temporal relational reasoning in videos," in Proc. European Conf. Computer Vision (ECCV), 2018.

[8] P. Saikia et al., "A hybrid CNN-LSTM model for video deepfake detection using optical flow features," Computers and Electrical Engineering, vol. 104, 2022.

[9] K. Warke, N. Dalavi, and S. Nahar, "Deepfake detection through deep learning using ResNeXt CNN and LSTM," IEEE Trans. Neural Networks and Learning Systems, vol. 10, no. 5, pp. 1–10, 2023.

[10] V. L. L. Thing, "Deepfake detection with deep learning: Convolu-tional neural networks versus Vision Transformers," arXiv preprint, arXiv:2304.03698, 2023.

[11] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A large-scale challenging dataset for DeepFake forensics," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), pp. 3207–3216, 2020.

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

[13] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1492–1500, 2017.

[14] G. Petmezas, V. Vagian, K. Konstantoudakis, and D. Zarpalas, "Video deepfake detection using a hybrid CNN-LSTM-Transformer model for identity verification," Multimedia Tools and Applications, vol. 84,
pp. 40617–40636, 2025.