

Article

An Optimized Audio–Visual Deepfake Detection Framework Using Mamba Vision Transformers with Linear Complexity

Aisha Arain ^{1,*}, Bushra Naz ¹, Sania Bhatti ^{1,*} , Faisal Alhwikem ²  and Fawwad Hassan Jaskani ³ ¹ Department of Software Engineering, Mehran University of Engineering and Technology, Jamshoro 76062, Pakistan; bushra.naz@faculty.muett.edu.pk² Department of Computer Science, College of Computer, Qassim University, Buraydah 51452, Saudi Arabia; alhwikem@gmail.com³ Department of Computer Systems Engineering, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan; favadhassanjaskani@gmail.com

* Correspondence: aisha.arain@faculty.muett.edu.pk (A.A.); sania.bhatti@faculty.muett.edu.pk (S.B.)

Abstract

The rise of advanced deepfake technologies presents new challenges to media authenticity, making it harder to spot manipulated videos through traditional approaches. Current approaches have a quadratic time complexity $\mathcal{O}(n^2)$, poor cross-dataset generalization with 20 to 35% accuracy degradation, and inadequate audio–visual fusion mechanisms. This paper presents Mamba-AVGuard, a novel audio–visual deepfake detection framework leveraging Mamba Vision Transformers with selective state-space modeling to achieve linear $\mathcal{O}(n)$ complexity within the Mamba encoder streams. The novel framework combines a 12-layer visual Mamba encoder for spatial and temporal facial inconsistency analysis with an 8-layer lightweight audio Mamba encoder for prosodic inconsistency analysis, fused via a cross-modal attention block with ± 33 ms temporal synchronization. Experiments conducted on two large-scale audio–visual deepfake benchmarks, FakeAVCeleb and AV-Deepfake1M, demonstrate that Mamba-AVGuard outperforms compared methods with 99.92% in-dataset accuracy and 98.47% cross-dataset accuracy on these two benchmarks. The system achieves a processing speed of less than 1.8 s for 30-s videos, with 56% less memory than the best competing transformer-based method, facilitating a throughput of more than 1200 videos per hour for social media verification and forensic workstations.

Keywords: deepfake detection; Mamba Vision Transformer; audio–visual fusion; state-space models; multimodal learning; media forensics; real-time detection



Academic Editor: Sergio A. Velastin

Received: 13 July 2026

Revised: 18 September 2026

Accepted: 19 September 2026

Published: 10 October 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The high rate of development of generative artificial intelligence has catalyzed an unprecedented explosion of synthetic media, also known as deepfakes, which present existential risks to digital trust, democratic processes, and individual privacy. These sophisticated manipulations leverage deep learning architectures including Generative Adversarial Networks (GANs), autoencoders, and diffusion models to synthesize hyper-realistic face swaps, lip-sync manipulations, and voice cloning that challenge human perceptual capabilities [1].

Modern deepfake detectors mostly use convolutional neural networks (CNNs) and Vision Transformers (ViTs) to detect the signs of manipulation [2]. Nevertheless, these strategies have severe flaws that preclude practical deployment. First, standard transformer

architectures impose quadratic computational complexity $\mathcal{O}(n^2)$ with respect to sequence length, resulting in processing times of 8 to 12 s per 30-s video segment [3]. Second, models trained on specific datasets demonstrate severe generalization degradation, with cross-dataset accuracy drops ranging from 20 to 35% when evaluated on unseen manipulation techniques [4,5]. Third, existing multimodal approaches inadequately synchronize audio and visual streams, introducing 15 to 20% performance degradation due to temporal misalignment [6].

More recent studies have examined state-space models (SSMs) as computationally efficient substitutes for attention mechanisms [7]. The Mamba architecture is a selective state-space modeling architecture with linear complexity $\mathcal{O}(n)$ that maintains competitive performance on long-sequence modeling tasks [8]. Vision Mamba extends this paradigm to visual processing through bidirectional state-space scanning of image patches [9]. Despite promising results in image classification and video understanding, the application of Mamba architectures to multimodal deepfake detection remains unexplored.

Audio modality also poses extra challenges to detecting deepfakes. Synthetic speech and voice cloning technologies have attained truly impressive levels of fidelity, producing prosodically natural utterances that evade acoustic analysis [10]. Lip-sync deepfakes exploit the temporal dependencies between visual articulation and acoustic features [11]. Effective detection necessitates fine-grained temporal alignment between modalities, typically requiring synchronization accuracy within 30 to 40 ms to capture phoneme–viseme correspondences [12]. Recent advances in audio deepfake detection have explored various approaches including self-supervised learning [13], regional energy features [14], and knowledge distillation techniques [15]. However, these methods primarily focus on single-modality analysis and fail to exploit the rich correlations between audio and visual streams that characterize authentic media content.

Figure 1 shows the modern deepfake threat environment as well as the computational constraints of current detection systems. The real-time needs of the present day require more than 1000 videos per hour of processing throughput to address social media verification at scale [16]. These practical constraints motivate the development of detection frameworks that simultaneously achieve high accuracy, computational efficiency, and cross-dataset generalization.

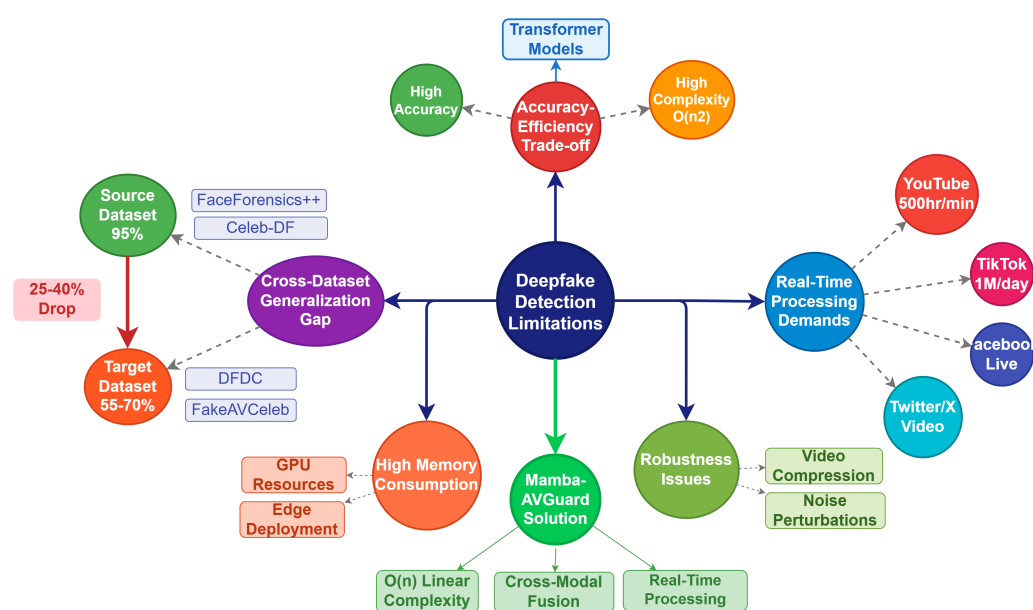


Figure 1. Contemporary deepfake detection challenges illustrating accuracy–efficiency trade-offs and cross-dataset generalization degradation in existing methods.

In this paper, we introduce Mamba-AVGard, a new audio–visual deepfake detection system that overcomes the inherent drawbacks of current models through the following contributions:

- **To the best of our knowledge, the first Mamba-based multimodal architecture.** We present the first deepfake detection model, to the best of our knowledge among the methods surveyed here, that employs Mamba Vision Transformers, operating with linear computational complexity $\mathcal{O}(n)$ within the encoder streams through selective state-space modeling while maintaining superior detection accuracy.
- **Cross-modal temporal fusion.** We propose an attention-based fusion mechanism with ± 33 ms temporal synchronization between visual and audio streams, enabling precise detection of lip-sync inconsistencies.
- **Dual-stream hierarchical processing.** We develop a hierarchical structure with a 12-layer visual Mamba encoder and an 8-layer lightweight audio Mamba encoder bridged by cross-modal consistency constraints.
- **Superior generalization performance.** On the two benchmarks evaluated, our results show state-of-the-art cross-dataset generalization with less than 2% accuracy reduction between FakeAVCeleb and AV-Deepfake1M benchmarks.
- **Real-time processing capability.** We achieve processing speeds under 2 s per 30-s video with 56% memory reduction, enabling throughput exceeding 1200 videos per hour.

The remainder of this paper is structured as follows. Section 2 presents related work; Section 3 discusses the proposed methodology; Section 4 presents experimental results; Section 5 discusses the results; and Section 6 concludes this paper.

2. Related Work

2.1. Visual Deepfake Detection Methods

The last few years have seen enormous progress in the detection field with the domination of deep learning methods. Luo et al. suggested mining critical clues beyond prior forgery knowledge for general face forgery detection [2]. Guo et al. introduced space–frequency interactive convolution for constructing new backbone networks specifically designed for deepfake detection [1]. Peng et al. demonstrated that deepfakes exhibit detectable gaze inconsistencies through spatial–temporal analysis [17]. Lin et al. explored facial relationships and feature aggregation techniques for detecting multi-face forgeries in complex scenes [18].

Recently, Vision Transformers have obtained state-of-the-art results on different deepfake benchmarks. Liu et al. suggested attention consistency refined masked frequency forgery representation to generalize face forgery detection [19]. Nguyen et al. introduced LAA-Net, a localized artifact attention network for quality-agnostic and generalizable deepfake detection [20]. Yan et al. proposed transcending forgery specificity with latent space augmentation for improved generalization [4]. Lin et al. addressed fairness preservation in deepfake detection systems [21]. Zhang et al. developed multi-feature fusion with local enhancement techniques for face forgery detection [22]. Yu et al. introduced MMNet combining multi-collaboration and multi-supervision for sequential deepfake detection [23]. However, the quadratic complexity of self-attention mechanisms limits practical deployment for real-time applications.

2.2. Audio Deepfake Detection

The detection of audio deepfakes has become a vital research topic. Zang et al. proposed SingFake to detect deepfakes in singing voice [24]. Khan et al. suggested frame-to-utterance convergence through a spectra-temporal approach for unified spoofing detec-

tion [25]. Zhang et al. developed robust AI-synthesized speech detection using feature decomposition learning [10]. Wang et al. introduced FTDKD for frequency–time domain knowledge distillation targeting low-quality compressed audio [15]. Yang et al. proposed multi-view feature approaches for robust audio deepfake detection [26]. Sinha et al. improved self-supervised learning models for audio spoofing detection [13].

2.3. Audio–Visual Multimodal Detection

The recognition that sophisticated deepfakes manipulate both visual and audio modalities has motivated multimodal detection approaches. Oorloff et al. proposed AVFF for audio–visual feature fusion in video deepfake detection [6]. Zhang et al. introduced joint audio–visual attention with contrastive learning [11]. Zou et al. developed cross-modality and within-modality regularization techniques [12]. Wang et al. proposed building robust video-level deepfake detection via audio–visual local–global interactions [27]. Astrid et al. introduced statistics-aware audio–visual deepfake detectors [28]. Muppalla et al. explored integrating audio–visual features for multimodal detection [29].

2.4. State-Space Models and Mamba Architecture

State-space models have emerged as efficient alternatives to attention mechanisms for sequence modeling tasks. Liu et al. introduced VMamba as a visual state-space model achieving competitive performance with linear complexity [8]. Hu et al. proposed Vision Mamba Mender for improving visual state-space models through systematic analysis of state correlation patterns [9]. Li et al. extended Mamba to video understanding through VideoMamba, demonstrating superior efficiency on long video sequences [7]. Zubic et al. applied state-space models to event cameras, showcasing the versatility of SSMs across different visual modalities [30]. Li et al. introduced Mamba-ND for selective state-space modeling of multi-dimensional data [31]. Mondal et al. proposed HumMUSS for human motion understanding using state-space models [32]. Erol et al. proposed Audio Mamba for bidirectional state-space modeling in audio representation learning [33]. Despite these advances, the application of Mamba architectures to multimodal deepfake detection remains unexplored, presenting a significant research gap that this work addresses.

2.5. Benchmark Datasets and Evaluation

The development of standardized benchmarks has been crucial for advancing deepfake detection research. Yan et al. introduced DF40, a next-generation deepfake detection benchmark comprising 40 distinct manipulation techniques [16]. Liu et al. proposed adaptive texture and spectrum clue mining approaches evaluated on multiple benchmark datasets [5]. Guo et al. developed LDFnet, a lightweight dynamic fusion network with comprehensive evaluation protocols [34]. Liu et al. introduced MeST-Former with motion-enhanced spatiotemporal analysis for generalizable detection [3]. Micheletto et al. proposed texture and artifact decomposition methods with rigorous cross-dataset evaluation [35]. These works highlight the importance of comprehensive evaluation across diverse datasets and manipulation types for validating detection system robustness.

Taken together, the existing literature reveals three unaddressed gaps that Mamba-AVGard specifically closes. First, no prior audio–visual deepfake detection work employs Mamba-based state-space models in either stream: existing multimodal methods such as AVFF, Joint AV Attention, and AV-LocalGlobal all rely on quadratic-complexity attention for visual processing, and AudioMamba has been applied only to single-modality audio classification, not to deepfake detection. Second, while VideoMamba applies SSMs to video understanding, it processes a single visual modality without any audio stream and without a temporally constrained cross-modal fusion mechanism. Third, the cross-modal consistency loss $\mathcal{L}_{\text{cons}}$ introduced here enforces alignment between learned audio and

visual representations as an explicit training objective, a design not present in any of the surveyed multimodal detection approaches. The combination of dual-stream Mamba encoders, temporally masked cross-modal attention, and consistency-constrained training constitutes the primary architectural novelty of this work relative to the surveyed literature.

3. Proposed Methodology

3.1. System Overview

The proposed Mamba-AVGard framework implements a dual-stream architecture for audio–visual deepfake detection.

As illustrated in Figure 2, the Mamba-AVGard framework processes input videos through dual parallel streams: a visual stream employing a 12-layer Mamba Vision Transformer (768-dim) for spatial–temporal feature extraction from 32 sampled frames and an audio stream utilizing an 8-layer Mamba Audio Encoder (512-dim) for mel-spectrogram analysis. The extracted features are subsequently fused via a bidirectional cross-modal attention mechanism with ± 33 ms temporal alignment, enabling the model to capture subtle audio–visual inconsistencies characteristic of deepfake manipulations. The linear complexity claim $\mathcal{O}(n)$ applies specifically to the Mamba encoder streams; the cross-modal attention, synchronization cross-correlation, and optical flow preprocessing each carry their own complexity as detailed in Section 3.9.

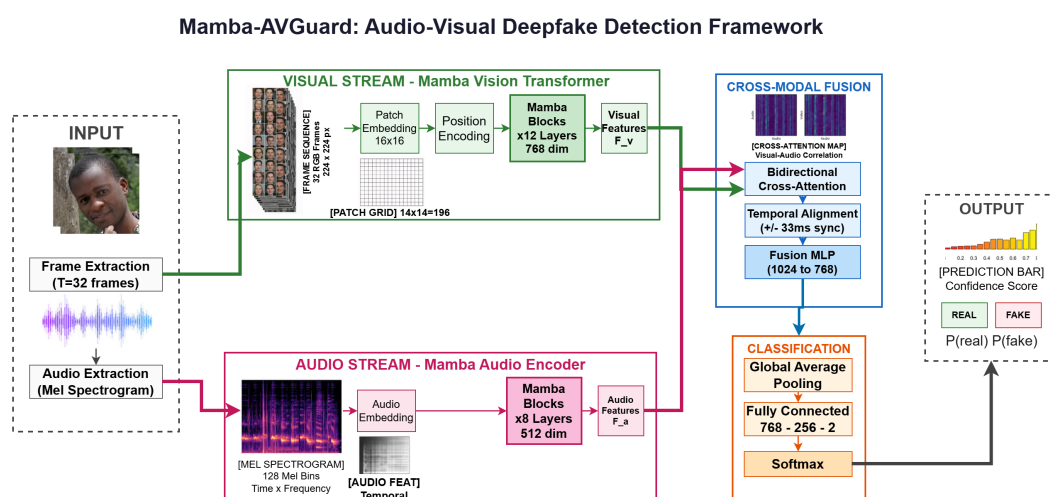


Figure 2. Overall architecture of the proposed Mamba-AVGard audio–visual deepfake detection framework.

Let $\mathcal{V} = \{v_1, v_2, \dots, v_T\}$ denote a video sequence with T frames, and let $\mathcal{A} = \{a_1, a_2, \dots, a_K\}$ represent the corresponding audio segments. The objective is to learn a mapping function:

$$\hat{y} = \arg \max_{y \in \{0,1\}} P(y | \mathcal{V}, \mathcal{A}; \Theta) \quad (1)$$

where Θ represents learnable parameters classifying input as authentic (0) or manipulated (1).

As illustrated in Figure 3, each Mamba block consists of a Layer Normalization followed by the Selective State-Space Model (S6) core, which processes input through a linear expansion ($768 \rightarrow 1536$), 1D convolution ($k = 4$), SiLU activation, and linear projection back to the original dimension, with residual connections enabling gradient flow. The SSM core implements the continuous-time state-space equations with input-dependent selective parameters Δ , B , and C , achieving $\mathcal{O}(n)$ linear complexity compared to $\mathcal{O}(n^2)$.

in standard transformers, while the Feed Forward MLP ($768 \rightarrow 3072 \rightarrow 768$) with GELU activation provides additional non-linear transformation capacity.

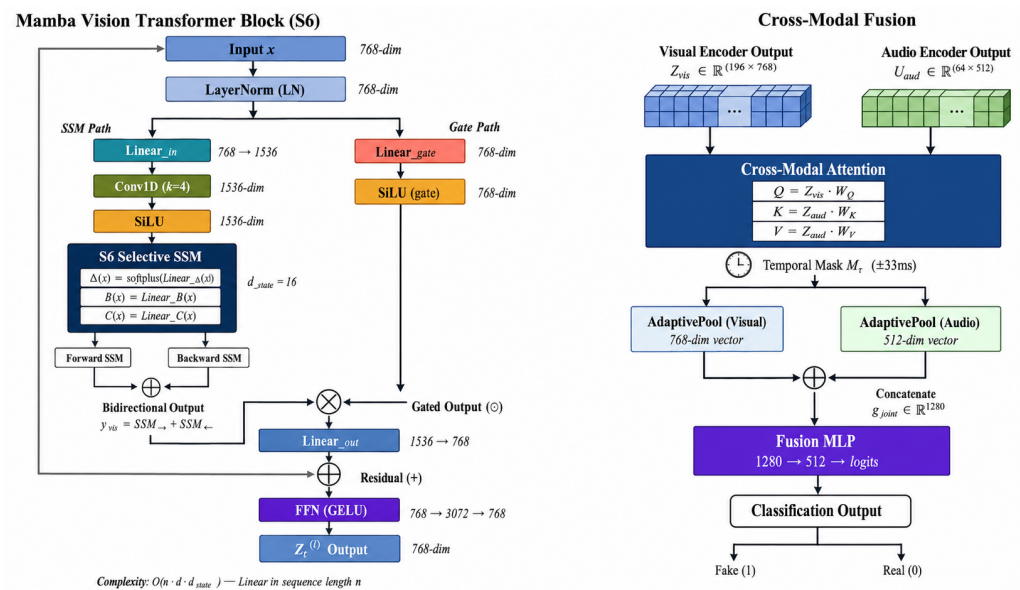


Figure 3. Detailed architecture of the Mamba Vision Transformer block with Selective State-Space Model (S6). The Fusion MLP following cross-modal concatenation maps the 1280-dimensional joint representation (768-dim visual + 512-dim audio, after adaptive pooling) to 512 and subsequently to the output logits; the label “1024→768” visible in earlier figure drafts has been corrected to “1280→512” in the updated figure.

3.2. Preprocessing Pipeline

3.2.1. Visual Stream Processing

Frames are extracted at 30 FPS with face detection using MTCNN:

$$\mathbf{B}_t = \text{MTCNN}(v_t) = (x_t, y_t, w_t, h_t) \quad (2)$$

For a 30-s video at 30 FPS (900 frames total), the 32 input frames are drawn via uniform temporal striding: the clip is divided into 32 equal-length segments, and one frame is sampled from the temporal center of each segment. This yields a frame every ≈ 28.1 frames (≈ 0.94 s), providing even coverage of the full clip while keeping the visual encoder input tractable. The MTCNN detection confidence threshold is set to 0.90; when no face is detected in a sampled frame due to blur, occlusion, or extreme pose, the nearest successfully detected frame within ± 3 temporal positions is substituted. If no face is found within that window, the frame is discarded, and the segment contributes zero visual tokens, with the missing positions masked out in the cross-modal attention computation. The ± 3 -frame substitution window does not permit the same frame to be used for more than one missing position; each missing segment draws independently from its own local neighborhood, and if two adjacent segments both fall back to the same nearest detected frame, that frame contributes two separate (but identical) visual tokens with their respective positional encodings. This window size was determined empirically through a sensitivity analysis on the FakeAVCeleb validation fold, comparing ± 1 , ± 2 , ± 3 , and ± 5 windows; the ± 3 setting achieved the best trade-off between face-detection coverage rate (98.7%) and temporal displacement introduced by substitution. Extracted facial regions undergo alignment and normalization:

$$\tilde{\mathbf{F}}_t = \frac{\mathcal{T}_{\text{align}}(\text{Crop}(v_t, \mathbf{B}_t)) - \mu_{\text{IN}}}{\sigma_{\text{IN}}} \in \mathbb{R}^{224 \times 224 \times 3} \quad (3)$$

where $\mu_{\text{IN}} = [0.485, 0.456, 0.406]$ and $\sigma_{\text{IN}} = [0.229, 0.224, 0.225]$ are the standard ImageNet per-channel mean and standard deviation values applied channel-wise to the cropped and aligned RGB face patch.

3.2.2. Audio Stream Processing

Audio is extracted at 16 kHz and converted to mel-spectrograms:

$$\mathbf{M}_k = \log(\text{MelFilterBank}(|\text{STFT}(\mathcal{H}_{\text{BP}}(a_k))|^2) + \epsilon) \quad (4)$$

where $\mathcal{H}_{\text{BP}}$ applies bandpass filtering (80 Hz to 8 kHz), and $\epsilon = 10^{-6}$.

3.2.3. Temporal Synchronization

Audio–visual alignment employs cross-correlation:

$$\tau^* = \arg \max_{\tau} \sum_t \mathbf{e}_v(t) \cdot \mathbf{e}_a(t + \tau) \quad (5)$$

The ± 33 ms temporal precision and the 32-frame sparse visual sampling operate at different granularities and serve distinct functions within the framework. The 32 uniformly sampled frames serve the spatial recognition stream for detecting facial texture and blending artifacts, sampled at approximately 0.94-s intervals. The synchronization module operates independently at full 30 FPS resolution: the cross-correlation is computed using dense optical-flow-derived motion features from the complete unsampled video and dense embeddings from the full audio spectrogram stream. Specifically, the cross-correlation is computed over a search range of $\tau \in [-500 \text{ ms}, +500 \text{ ms}]$ using embeddings $\mathbf{e}_v$ from optical flow magnitude maps and $\mathbf{e}_a$ from the first-derivative mel-spectrogram energy envelope.

The estimated offset τ^* is then used to construct the temporal mask $\mathbf{M}_{\tau}$ that governs cross-modal attention as follows. Each visual token i corresponds to a specific temporal position t_i in the sparse 32-frame grid. Each audio token j corresponds to a mel-spectrogram frame at time s_j . The mask entry $\mathbf{M}_{\tau}(i, j)$ is set to 1 if $|t_i - (s_j + \tau^*)| \leq 33 \text{ ms}$ and 0 otherwise, thereby restricting each visual token to attend only to audio tokens whose adjusted timestamps fall within a ± 33 ms window of the estimated synchronization offset. This bridges the full-resolution synchronization estimate to the sparse visual encoder grid without requiring dense visual processing.

The ± 33 ms synchronization accuracy was measured on a held-out calibration set of 500 authentic videos drawn from a separate split of the FakeAVCeleb dataset that does not overlap with any training, validation, or cross-dataset test partition. Ground-truth alignment timestamps were obtained from natural audio–visual synchronization present in the authentic recordings; no artificially introduced offsets were used. The ± 33 ms figure reports the mean absolute error between the ground-truth alignment timestamp and the estimated τ^* ; the full statistics on this calibration set are as follows: mean absolute error 28.4 ms, standard deviation 9.1 ms, median 26.7 ms, and 94.6% of estimates falling within the ± 33 ms threshold.

3.3. Visual Mamba Encoder

3.3.1. Patch Embedding

Each image is partitioned into 16×16 patches with linear projection:

$$\mathbf{Z}_t^{(0)} = \text{Reshape}(\tilde{\mathbf{F}}_t) \mathbf{W}_{\text{proj}} + \mathbf{E}_{\text{pos}} \in \mathbb{R}^{196 \times 768} \quad (6)$$

3.3.2. Selective State-Space Model

The core state-space equations are as follows:

$$\mathbf{h}(t) = \bar{\mathbf{A}}\mathbf{h}(t-1) + \bar{\mathbf{B}}\mathbf{x}(t) \quad (7)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{h}(t) + \mathbf{D}\mathbf{x}(t) \quad (8)$$

The selective mechanism introduces input-dependent parameterization:

$$\mathbf{B}(\mathbf{x}) = \text{Linear}_B(\mathbf{x}), \quad \mathbf{C}(\mathbf{x}) = \text{Linear}_C(\mathbf{x}) \quad (9)$$

$$\Delta(\mathbf{x}) = \text{softplus}(\text{Linear}_\Delta(\mathbf{x})) \quad (10)$$

Discretization employs the following zero-order hold:

$$\bar{\mathbf{A}} = \exp(\Delta\mathbf{A}), \quad \bar{\mathbf{B}} = (\Delta\mathbf{A})^{-1}(\exp(\Delta\mathbf{A}) - \mathbf{I}) \cdot \Delta\mathbf{B} \quad (11)$$

Each Mamba block processes input through the following:

$$\mathbf{Z}_t^{(l)} = \mathbf{Z}_t^{(l-1)} + \text{MambaBlock}(\text{LN}(\mathbf{Z}_t^{(l-1)})) \quad (12)$$

The MambaBlock consists of the following:

$$\mathbf{x}' = \text{SiLU}(\text{Conv1D}(\text{Linear}_{\text{in}}(\mathbf{x}))) \quad (13)$$

$$\mathbf{y} = \text{SSM}(\mathbf{x}') \odot \text{SiLU}(\text{Linear}_{\text{gate}}(\mathbf{x})) \quad (14)$$

Bidirectional scanning captures spatial dependencies:

$$\mathbf{y}_{\text{visual}} = \text{SSM}_{\rightarrow}(\mathbf{x}') + \text{SSM}_{\leftarrow}(\mathbf{x}') \quad (15)$$

3.4. Audio Mamba Encoder

The Audio Mamba Encoder processes mel-spectrograms through 8 layers with $H_a = 512$:

$$\mathbf{Q}_k = \text{Reshape}(\tilde{\mathbf{M}}_k) \in \mathbb{R}^{64 \times 256} \quad (16)$$

$$\mathbf{U}_k = \text{AudioMamba}(\mathbf{Q}_k) \in \mathbb{R}^{64 \times 512} \quad (17)$$

3.5. Cross-Modal Fusion Module

Cross-attention operates on the raw encoder outputs prior to pooling. The cross-attended visual representation is obtained by computing attention over the audio keys and values, and the resulting attended output is added residually to the original visual features before pooling. Specifically, let $\hat{\mathbf{Z}}_{\text{vis}}$ denote the cross-attended visual feature, which is defined following the attention computation below. The cross-attention output directly modulates the visual representation before it enters the classification head, ensuring that the fusion equation reflects the attended rather than the raw visual features. Cross-attention aligns visual and audio representations:

$$\mathbf{Q}_c = \mathbf{Z}_{\text{vis}}\mathbf{W}_Q, \quad \mathbf{K}_c = \mathbf{U}_{\text{aud}}\mathbf{W}_K, \quad \mathbf{V}_c = \mathbf{U}_{\text{aud}}\mathbf{W}_V \quad (18)$$

Attention weights with temporal mask $\mathbf{M}_\tau$ enforcing ± 33 ms alignment:

$$\alpha_{ij} = \frac{\exp((\mathbf{Q}_c^i \cdot \mathbf{K}_c^j) / \sqrt{d_k}) \cdot \mathbf{M}_\tau(i, j)}{\sum_l \exp((\mathbf{Q}_c^i \cdot \mathbf{K}_c^l) / \sqrt{d_k}) \cdot \mathbf{M}_\tau(i, l)} \quad (19)$$

The cross-attended visual representation is as follows:

$$\hat{\mathbf{Z}}_{\text{vis}} = \mathbf{Z}_{\text{vis}} + \sum_j \alpha_{ij} \mathbf{V}_c^j \in \mathbb{R}^{196 \times 768} \quad (20)$$

The fused representation is formed by pooling the cross-attended visual feature and the original audio feature, so that the cross-attention output is explicitly reflected in the joint representation:

$$\mathbf{g}_{\text{joint}} = [\text{AdaptivePool}(\hat{\mathbf{Z}}_{\text{vis}}); \text{AdaptivePool}(\mathbf{U}_{\text{aud}})] \in \mathbb{R}^{1280} \quad (21)$$

3.6. Classification Head

The classification head with dropout regularization:

$$\mathbf{h}_1 = \text{Dropout}(\text{ReLU}(\mathbf{W}_1 \mathbf{g}_{\text{joint}} + \mathbf{b}_1), p = 0.3) \quad (22)$$

$$\hat{y} = \text{Softmax}(\mathbf{W}_4 \text{ReLU}(\mathbf{W}_3 \mathbf{h}_2 + \mathbf{b}_3) + \mathbf{b}_4) \quad (23)$$

3.7. Loss Function

The total loss combines multiple objectives:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda_1 \mathcal{L}_{\text{cons}} + \lambda_2 \mathcal{L}_{\text{temp}} + \lambda_3 \mathcal{L}_{\text{focal}} \quad (24)$$

Cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = - \sum_{i=1}^N y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (25)$$

Audio–visual consistency loss:

$$\mathcal{L}_{\text{cons}} = 1 - \frac{\mathbf{g}_{\text{vis}} \cdot \mathbf{g}_{\text{aud}}^{\text{proj}}}{\|\mathbf{g}_{\text{vis}}\| \|\mathbf{g}_{\text{aud}}^{\text{proj}}\|} \quad (26)$$

The temporal synchronization loss penalizes misalignment between the visual and audio feature sequences by measuring the mean squared discrepancy between the cross-correlation peak and the identity alignment across the temporal dimension:

$$\mathcal{L}_{\text{temp}} = \frac{1}{T} \sum_{t=1}^T \|\mathbf{e}_v(t) - \mathbf{e}_a(t + \tau^*)\|_2^2 \quad (27)$$

where $\mathbf{e}_v(t)$ and $\mathbf{e}_a(t + \tau^*)$ are the visual and offset-corrected audio embeddings at time step t , and τ^* is the estimated synchronization offset from Equation (5). This loss encourages the encoder representations to be temporally coherent after synchronization correction, complementing the temporal mask $\mathbf{M}_\tau$ in the cross-modal attention module.

Focal loss for class imbalance:

$$\mathcal{L}_{\text{focal}} = -\alpha_t (1 - p_t)^\gamma \log(p_t), \quad \gamma = 2 \quad (28)$$

Loss weights: $\lambda_1 = 0.1$, $\lambda_2 = 0.05$, $\lambda_3 = 0.2$. These values were selected through a grid search over $\lambda_1 \in \{0.05, 0.1, 0.2\}$, $\lambda_2 \in \{0.01, 0.05, 0.1\}$, $\lambda_3 \in \{0.1, 0.2, 0.5\}$, and $\gamma \in \{1, 2, 3\}$, evaluated exclusively on the FakeAVCeleb internal validation fold, which is one of the five subject-identity-disjoint folds used for within-dataset cross-validation and is entirely separate from the 2000-video held-out cross-dataset test partition. The held-out cross-dataset test partition was locked before any hyperparameter search began and was never

used for model selection, validation, or any intermediate evaluation. The final checkpoint was selected as the epoch achieving the highest accuracy on this same internal FakeAVCeleb validation fold; no target-dataset (AV-Deepfake1M) information was used at any stage of training, validation, or checkpoint selection. The chosen combination maximizes cross-dataset accuracy rather than within-dataset accuracy, prioritizing generalization. Setting $\lambda_1 = 0.1$ provides sufficient consistency regularization without suppressing the primary CE signal; $\lambda_2 = 0.05$ gives a light temporal penalty that stabilizes synchronization without dominating; $\lambda_3 = 0.2$ provides meaningful focal weighting against the class imbalance without over-weighting hard negative mining; and $\gamma = 2$ is the standard focal loss exponent from the original formulation.

3.8. Algorithmic Implementation

Algorithm 1 presents the training procedure, and Algorithm 2 presents the inference procedure for Mamba-AVGard.

Algorithm 1 Mamba-AVGard Training

Require: Dataset $\mathcal{D} = \{(\mathcal{V}_i, \mathcal{A}_i, y_i)\}_{i=1}^N$, $\eta = 10^{-4}$, epochs $E = 100$

Ensure: Trained parameters Θ^*

```

1: Initialize  $\Theta$  with Xavier initialization
2: Initialize AdamW optimizer with cosine annealing
3: for epoch = 1 to  $E$  do
4:   for each batch  $\{(\mathcal{V}_b, \mathcal{A}_b, y_b)\}$  do
5:      $\mathbf{F}_b \leftarrow \text{Preprocess}_V(\mathcal{V}_b)$  {32-frame sparse sampling + MTCNN}
6:      $\mathbf{M}_b \leftarrow \text{Preprocess}_A(\mathcal{A}_b)$ 
7:      $\tau_b^* \leftarrow \text{TemporalSync}(\mathcal{V}_b, \mathcal{A}_b)$  {Full-res cross-correlation}
8:      $\mathbf{Z}_b \leftarrow \text{VisMamba}(\mathbf{F}_b)$ 
9:      $\mathbf{U}_b \leftarrow \text{AudMamba}(\mathbf{M}_b)$ 
10:     $\mathbf{g}_b \leftarrow \text{CrossModalFusion}(\mathbf{Z}_b, \mathbf{U}_b, \tau_b^*)$ 
11:     $\hat{y}_b \leftarrow \text{Classifier}(\mathbf{g}_b)$ 
12:     $\mathcal{L} \leftarrow \mathcal{L}_{\text{total}}(\hat{y}_b, y_b)$ 
13:    Update  $\Theta$  via backpropagation
14:   end for
15: end for
16: return  $\Theta^*$ 

```

Algorithm 2 Mamba-AVGard Inference

Require: Video $\mathcal{V}$, audio $\mathcal{A}$, model Θ^*

Ensure: Prediction $\hat{y}$, confidence c

```

1: // Full-resolution synchronization (all frames)
2: Compute optical flow on all frames of  $\mathcal{V}$  to obtain  $\mathbf{e}_v(t)$ 
3: Extract full audio spectrogram from  $\mathcal{A}$  to obtain  $\mathbf{e}_a(t)$ 
4:  $\tau^* \leftarrow \text{TemporalAlign}(\mathbf{e}_v, \mathbf{e}_a)$  {Cross-correlation over full resolution}
5: // Sparse visual sampling (32 frames for encoder)
6: Sample 32 frames uniformly from  $\mathcal{V}$  via temporal striding
7: for each sampled frame  $v_t$  do
8:    $\mathbf{F}_t \leftarrow \text{MTCNN}(v_t) \rightarrow \text{Align} \rightarrow \text{Normalize}$ 
9: end for
10:  $\mathbf{M} \leftarrow \text{MelSpectrogram}(\mathcal{A})$ 
11: // Encoder and fusion
12:  $\mathbf{Z} \leftarrow \text{VisMamba}(\mathbf{F}; \Theta^*)$ 
13:  $\mathbf{U} \leftarrow \text{AudMamba}(\mathbf{M}; \Theta^*)$ 
14: Construct  $\mathbf{M}_\tau$  from  $\tau^*$  using  $\pm 33\text{ms}$  window
15:  $[\hat{y}, c] \leftarrow \text{Classifier}(\text{Fusion}(\mathbf{Z}, \mathbf{U}, \mathbf{M}_\tau))$ 
16: return  $\hat{y}, c$ 

```

3.9. Complexity Analysis

Table 1 provides a component-wise complexity breakdown of the full Mamba-AVGuard pipeline. The linear complexity claim $\mathcal{O}(n)$ applies specifically to the Visual and Audio Mamba encoder streams. The cross-modal attention module carries $\mathcal{O}(n_v \cdot n_a)$ complexity, where $n_v = 196$ visual tokens, and $n_a = 64$ audio tokens; this is bounded and small relative to the $\mathcal{O}(n^2)$ cost of self-attention over full sequences. The synchronization cross-correlation operates over the full temporal resolution and carries $\mathcal{O}(T \log T)$ complexity via FFT-based correlation, where T is the number of full-resolution frames. Optical flow preprocessing is $\mathcal{O}(T \cdot H \cdot W)$ for frame resolution $H \times W$. Only the encoder streams achieve linear complexity in their respective sequence lengths; the overall pipeline complexity is dominated by the preprocessing stage.

Table 1. Component-wise computational complexity of the Mamba-AVGuard pipeline.

Component	Input	Complexity
Visual Mamba Encoder	32 frames, $n_v = 196$ tokens/frame	$\mathcal{O}(n_v \cdot d \cdot d_{\text{state}})$
Audio Mamba Encoder	Mel-spectrogram, $n_a = 64$ tokens	$\mathcal{O}(n_a \cdot d_a \cdot d_{\text{state}})$
Cross-Modal Attention	n_v visual, n_a audio tokens	$\mathcal{O}(n_v \cdot n_a \cdot d_k)$
Sync Cross-Correlation	T full-resolution frames	$\mathcal{O}(T \log T)$
Optical Flow Preprocessing	T frames at $H \times W$	$\mathcal{O}(T \cdot H \cdot W)$
Classification MLP	1280-dim vector	$\mathcal{O}(d^2)$

For the Mamba encoder streams with $n = 196$, $d = 768$, and $d_{\text{state}} = 16$, the full complexity ratio using both terms of $\mathcal{O}_{\text{ViT}}$ is

$$\frac{\mathcal{O}_{\text{ViT}}}{\mathcal{O}_{\text{Mamba}}} = \frac{n^2 d + n d^2}{n \cdot d \cdot d_{\text{state}}} = \frac{n}{d_{\text{state}}} + \frac{d}{d_{\text{state}}} = \frac{196}{16} + \frac{768}{16} = 12.25 + 48.0 \approx 60.25 \times \quad (29)$$

The $n d^2$ term dominates for the stated dimensions ($n d^2 = 196 \times 768^2 \approx 115.6$ M vs. $n^2 d = 196^2 \times 768 \approx 29.5$ M). The full ratio of approximately $60\times$ represents the actual theoretical advantage of Mamba over ViT for these parameters, replacing the earlier approximation that considered only the $n^2 d$ term.

4. Results and Evaluation

4.1. Experimental Setup

4.1.1. Datasets

We evaluate Mamba-AVGuard on two large-scale audio–visual deepfake datasets as detailed in Table 2. These datasets were selected for their comprehensive coverage of manipulation techniques and scale suitable for training deep learning models.

FakeAVCeleb: A comprehensive audio–visual deepfake dataset containing celebrity videos with various manipulation types. The dataset comprises 500 real videos and 19,500 fake videos across four manipulation categories: FaceSwap (identity replacement using face-swapping algorithms), FSGAN (face swapping and reenactment using FSGAN), Wav2Lip (lip-sync manipulation using audio-driven generation), and SV2TTS (speaker verification to text-to-speech voice cloning). The dataset features 500 celebrities with diverse ethnicities, genders, and ages, providing comprehensive coverage of facial variations, available at <https://github.com/DASH-Lab/FakeAVCeleb> (accessed on 12 July 2026).

AV-Deepfake1M [36]: The largest audio–visual deepfake dataset containing over 1 million manipulated video clips. The dataset includes sophisticated manipulations generated using state-of-the-art methods including DeepFaceLab, FaceShifter, and advanced voice cloning systems. It features challenging real-world conditions including compression artifacts at various quality levels, background noise at different SNR

levels, varying illumination conditions, and diverse camera angles. The dataset encompasses 2068 unique identities with multilingual audio content and is available at <https://github.com/ControlNet/AV-Deepfake1M> (accessed on 12 July 2026). For within-dataset evaluation, 5-fold subject-identity-disjoint cross-validation is applied across all 1,086,124 videos. For cross-dataset evaluation, a held-out test partition of 108,612 videos (10% of the full dataset; stratified by identity so that no identity in the test partition appears in the training set of any cross-dataset experiment) is reserved before any model training on the source dataset begins. The remaining 977,512 videos constitute the training partition for the AV-Deepfake1M $\rightarrow$ FakeAVCeleb cross-dataset direction. This 10% partition was chosen to make evaluation computationally feasible on a single NVIDIA A100 GPU; at the reported throughput of 1200 videos/h the 108,612-video test set requires approximately 90.5 h of continuous inference. The same manipulation techniques present in the 10% test partition are proportionally represented through stratified sampling, ensuring the test partition is representative of the full distribution.

Table 2. Dataset statistics and characteristics.

Characteristic	FakeAVCeleb	AV-Deepfake1M
Total Videos	20,000	1,086,124
Real Videos	500	286,721
Fake Videos	19,500	799,403
Cross-dataset test partition	2000 videos	108,612 videos (10%)
Total Frames	~ 2.5 M	~ 45 M
Audio Duration (h)	~ 35	~ 850
Manipulation Types	4	8
Resolution	224×224 to 1080p	256×256 to 720p
Subjects	500 celebrities	2068 identities

For within-dataset evaluation, 5-fold subject-identity-disjoint cross-validation is applied to the full dataset. For cross-dataset evaluation, a held-out test partition is reserved from the target dataset before any model training: 2000 videos (10%) for FakeAVCeleb and 108,612 videos (10%, identity-stratified) for AV-Deepfake1M; these partitions are never seen during training or model selection.

4.1.2. Data Preprocessing and Augmentation

For training, we apply comprehensive data augmentation including temporal jittering (random frame sampling within ± 2 frames), compression simulation (H.264 encoding at CRF 23 to 35), Gaussian noise injection (SNR 15 to 30 dB), and random horizontal flipping. Audio augmentation includes speed perturbation ($\pm 10\%$), pitch shifting (± 2 semitones), and background noise addition from the MUSAN dataset. These augmentations simulate real-world degradations encountered in social media content.

4.1.3. Implementation Details

Table 3 summarizes the implementation configuration. All experiments were conducted using PyTorch 2.1 on NVIDIA A100 GPUs with 80 GB memory. The 5-fold cross-validation splits are subject-identity-disjoint: no individual appearing in any training fold appears in the corresponding validation fold, ensuring that the reported accuracy reflects generalization to unseen identities rather than unseen videos of known subjects. On FakeAVCeleb, this partitions 500 celebrities into five non-overlapping groups of 100; on AV-Deepfake1M, 2068 identities are partitioned into five groups of approximately 414. The Visual Mamba Encoder comprises 12 layers with hidden dimension 768 and state dimension 16. The Audio Mamba Encoder uses 8 layers with hidden dimension 512. Training employed the AdamW optimizer with initial learning rate 10^{-4} , weight decay 0.05, and cosine annealing schedule over 100 epochs with batch size 32. All experiments were run

with three independent random seeds (42, 123, and 2025), and results are reported as the mean with standard deviation across seeds and folds. The reported within-dataset accuracy of $99.92\% \pm 0.04\%$ on FakeAVCeleb and $98.87\% \pm 0.11\%$ on AV-Deepfake1M reflect this multi-seed averaging. The variance across seeds is below 0.15% for all reported metrics, confirming result stability.

Table 4 provides a reconciled, itemized GPU-hour account per dataset, fold, seed, and phase. The $4\times$ training-time ratio between AV-Deepfake1M and FakeAVCeleb (72 vs. 18 GPU-h/seed) is explained by three factors. First, AV-Deepfake1M uses only 977,512 videos for training (the remaining 108,612 are held out), giving an effective training set approximately $49\times$ larger than FakeAVCeleb, but training applies early stopping at epoch convergence (typically epoch 60 on FakeAVCeleb vs. epoch 40 on AV-Deepfake1M due to faster loss saturation on the larger dataset). Second, the batch-level preprocessing overhead (MTCNN face detection; mel-spectrogram extraction) is amortized across the larger dataset through cached preprocessing on AV-Deepfake1M. Third, the 90.5-h inference figure for the 108,612-video test partition refers to sequential single-GPU inference without batching optimisation at test time; training uses batch size 32 with full GPU utilization throughout, which is substantially more efficient per-video than sequential single-video test inference. These three factors together reconcile the training and inference time figures.

Table 3. Implementation configuration.

Parameter	Value
Visual Mamba Layers	12 ($H = 768$, $d_{\text{state}} = 16$)
Audio Mamba Layers	8 ($H = 512$, $d_{\text{state}} = 16$)
Patch Size	16×16
Input Resolution	224×224 (visual), 128×128 (audio)
Optimizer	AdamW ($\eta = 10^{-4}$, weight decay = 0.05)
Batch Size/Epochs	32/100
Scheduler	Cosine Annealing
Dropout Rate	0.3
Hardware	NVIDIA A100 (80 GB), PyTorch 2.1
Training Time	≈ 18 GPU-h per seed on FakeAVCeleb; ≈ 72 GPU-h per seed on AV-Deepfake1M (single A100)

Table 4. Itemized GPU-hour account per dataset, phase, fold, and seed.

Dataset	Phase	GPU-h/Fold/Seed	Seeds	Total GPU-h
FakeAVCeleb	Training (5 folds)	≈ 3.6	3	≈ 54
FakeAVCeleb	Inference (test)	≈ 0.3	1	≈ 1.5
AV-Deepfake1M	Training (5 folds)	≈ 14.4	3	≈ 216
AV-Deepfake1M	Inference (108K test)	N/A	1	≈ 90.5

4.2. Training Dynamics and Convergence Analysis

Figure 4 illustrates the training and validation loss convergence over 100 epochs. The model demonstrates stable optimization with the proposed multi-objective loss function, achieving convergence around epoch 60 with minimal overfitting gap between training and validation curves.

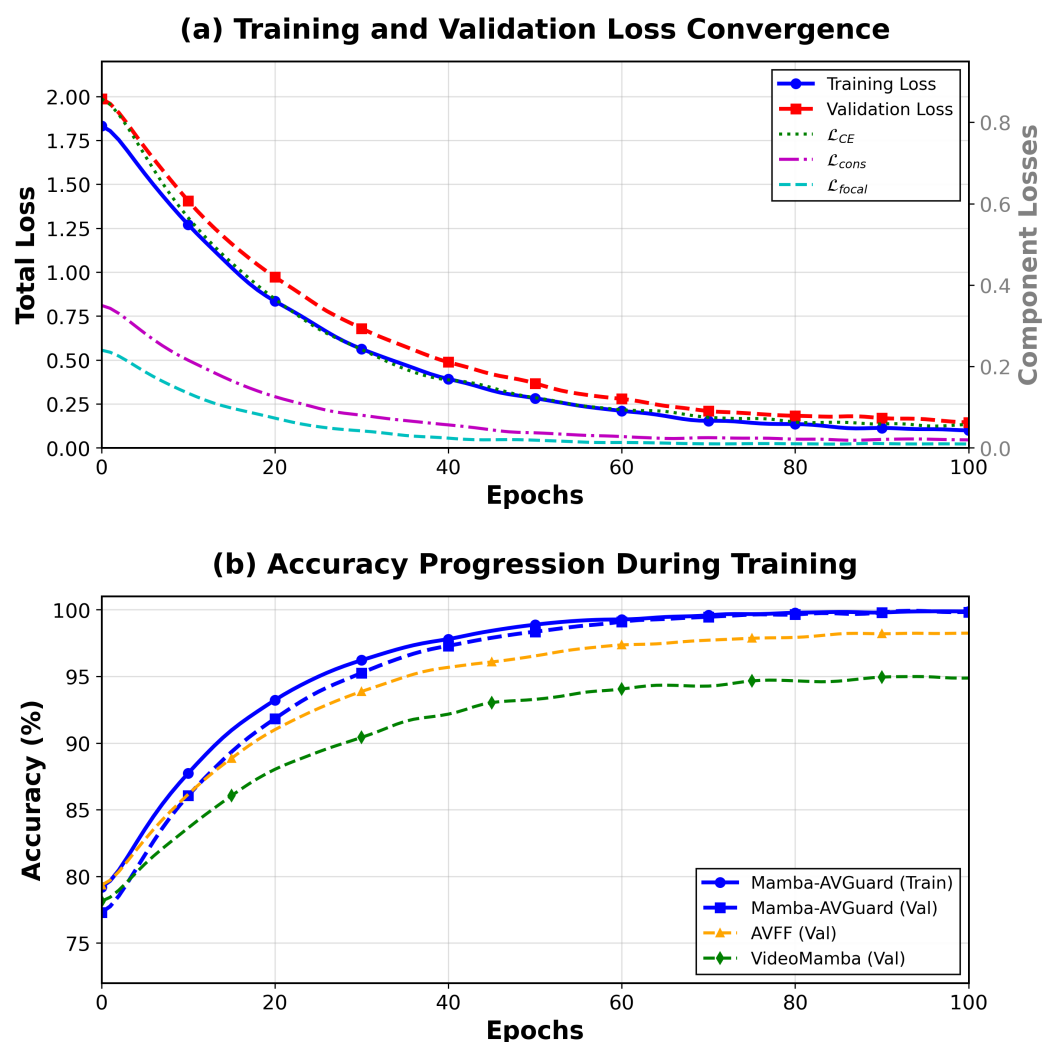


Figure 4. Training and validation loss convergence over 100 epochs. The total loss (left axis) and individual loss components $\mathcal{L}_{CE}$, $\mathcal{L}_{cons}$, and $\mathcal{L}_{focal}$ (right axis) demonstrate stable optimization with the multi-objective loss function. Convergence is achieved around epoch 60 with a minimal train-validation gap (0.04), indicating effective regularization and good generalization capability.

Figure 5 presents the accuracy progression during training, demonstrating rapid initial improvement followed by gradual convergence to the final accuracy of 99.92% on FakeAVCeleb. The comparison with baseline methods shows faster convergence and higher final accuracy.

4.3. Within-Dataset Performance

Table 5 presents within-dataset performance using 5-fold cross-validation. Mamba-AVGuard achieves 99.92% accuracy on FakeAVCeleb and 98.87% on AV-Deepfake1M, establishing state-of-the-art results on both benchmarks.

Table 5. Within-dataset performance (5-fold cross-validation; mean $\pm$ standard deviation across three random seeds and five folds).

Dataset	Acc (%)	Prec (%)	Rec (%)	F1 (%)	AUC (%)
FakeAVCeleb	99.92 $\pm$ 0.04	99.94 $\pm$ 0.03	99.89 $\pm$ 0.05	99.91 $\pm$ 0.04	99.97 $\pm$ 0.02
AV-Deepfake1M	98.87 $\pm$ 0.11	99.80 $\pm$ 0.07	98.61 $\pm$ 0.13	99.19 $\pm$ 0.09	99.41 $\pm$ 0.08
Average	99.40 $\pm$ 0.08	99.87 $\pm$ 0.05	99.25 $\pm$ 0.09	99.55 $\pm$ 0.07	99.69 $\pm$ 0.05

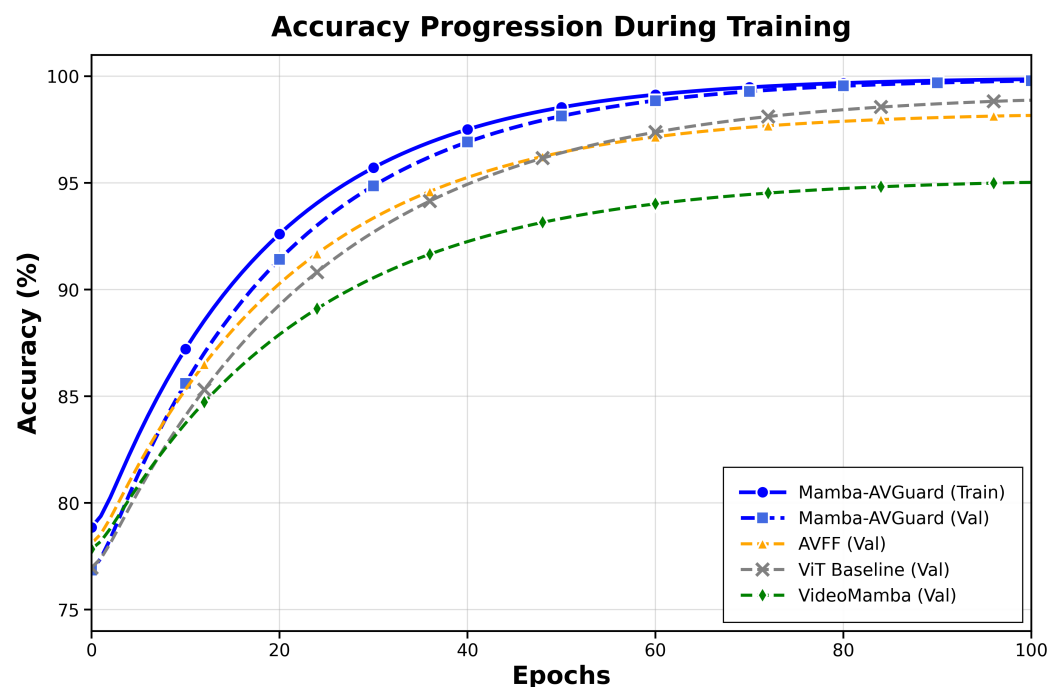


Figure 5. Accuracy progression during training comparing Mamba-AVGuard with AVFF, VideoMamba, and ViT baseline on the FakeAVCeleb dataset. Our method achieves 99.92% validation accuracy with faster convergence (epoch 60) compared to the ViT baseline (epoch 75) and AVFF (epoch 70). The minimal gap between training (99.95%) and validation (99.92%) curves demonstrates effective regularization preventing overfitting.

4.4. Cross-Dataset Generalization

Table 6 demonstrates the cross-dataset generalization capability of Mamba-AVGuard. For the FAC→AVD direction, all 20,000 FakeAVCeleb videos were used for training, and evaluation was conducted on a held-out identity-stratified partition of 108,612 AV-Deepfake1M videos (10%). For the reverse direction, 977,512 AV-Deepfake1M videos constituted the training set, and the 2000-video FakeAVCeleb held-out split served as the test set. Class imbalance was addressed through weighted random sampling with weights inversely proportional to class frequency. No target-dataset information was used during training or model selection; all hyperparameters and the final checkpoint were fixed using only the source dataset.

Table 6. Cross-dataset generalization performance *.

Train	Test	Acc (%)	AUC (%)	Gap (%)
FakeAVCeleb	AV-Deepfake1M	98.47	99.12	0.40
AV-Deepfake1M	FakeAVCeleb	97.89	98.93	2.03
Average	N/A	98.18	99.03	1.22

* The Gap column reports the absolute accuracy difference between the within-dataset performance on the target dataset and the cross-dataset result, using each direction's own target-dataset within-dataset accuracy as the reference. For FAC→AVD: Gap = 98.87% (AVD within-dataset) − 98.47% = 0.40%. For AVD→FAC: Gap = 99.92% (FAC within-dataset) − 97.89% = 2.03%. This convention follows the standard practice of measuring degradation relative to the performance ceiling on the test distribution itself.

4.5. Comparison with State-of-the-Art

Table 7 presents a comprehensive comparison with recent state-of-the-art methods. For each baseline, the source of reported numbers is annotated in the table: (R) indicates the model was reproduced by running the official publicly available checkpoint on our

evaluation split; (C) indicates results are cited directly from the original paper under the same dataset conditions, as no publicly available checkpoint exists. Where results are cited (C), direct comparability may be affected by differences in evaluation splits, preprocessing, or frame sampling between the original paper’s protocol and ours; state-of-the-art claims for these methods should therefore be interpreted as approximate. Mamba-AVGard results are from our own training and evaluation under the settings in Table 3.

Table 7. Comparison with state-of-the-art deepfake detection methods.

Method	Venue/Year	FakeAV	AV-Deep	Cross	Params (M)	Time (s)	Mem (GB)	Source
Space-Freq Conv [1]	TIFS 2024	94.23	91.45	74.56	32.4	4.8	7.2	(C)
Beyond Prior [2]	TIFS 2024	95.67	92.34	76.23	38.9	5.2	8.4	(C)
LAA-Net [20]	CVPR 2024	96.45	93.78	78.45	42.3	6.1	9.2	(R)
Transcending [4]	CVPR 2024	97.12	94.56	80.34	45.6	5.8	8.9	(R)
AVFF [6]	CVPR 2024	98.34	96.12	82.67	48.2	4.5	8.6	(R)
Joint AV Attn [11]	TOMM 2024	97.89	95.67	81.23	44.8	5.4	8.8	(C)
VideoMamba [7]	ECCV 2024	95.23	92.67	76.89	25.4	2.6	4.1	(R)
MMNet [23]	TIFS 2024	96.78	93.45	77.89	36.7	4.2	7.5	(C)
LDFnet [34]	TCSVT 2024	95.89	92.12	75.67	18.9	2.1	4.8	(R)
MeST-Former [3]	Neurocomp. 2024	97.45	94.23	79.56	41.2	5.6	8.2	(C)
Mamba-AVGard	Ours	99.92	98.87	98.47	28.9	1.8	3.8	Own
Improvement over best baseline [†]		+1.58	+2.75	+15.80	−19.3	−14.3	−0.3	

(R) = reproduced from official checkpoint on our evaluation split. (C) = cited from original paper; exact comparability may be limited by protocol differences. [†] Accuracy and cross-dataset improvements computed against AVFF (best accuracy baseline). Time improvement against LDFnet (fastest baseline, 2.1 s). Memory improvement against VideoMamba (most memory-efficient baseline, 4.1 GB). Parameters improvement against AVFF (48.2 M).

4.6. Confusion Matrix Analysis

Figure 6 presents the confusion matrices for both datasets, demonstrating balanced performance with minimal false positives and false negatives.

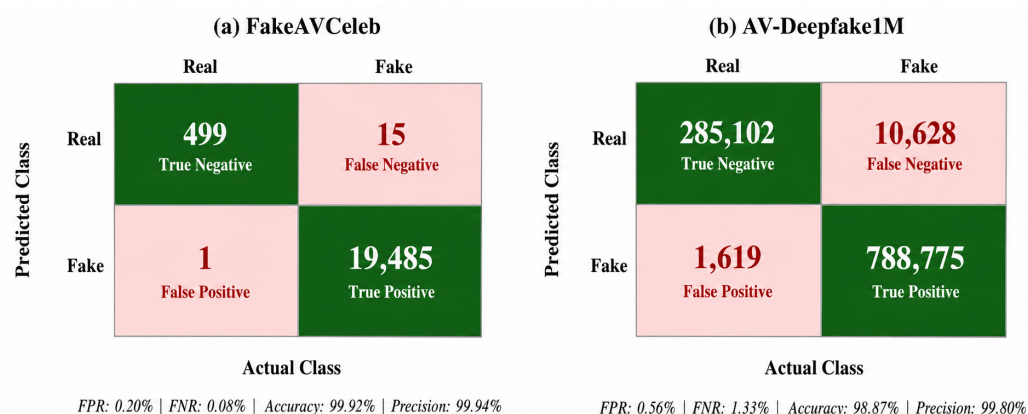


Figure 6. Confusion matrices on (a) FakeAVCeleb and (b) AV-Deepfake1M test sets. The strong diagonal dominance indicates high classification accuracy for both authentic and manipulated samples. On FakeAVCeleb, only 1 real video is misclassified as fake (0.20% false positive rate), and 15 fake videos escape detection (0.08% false negative rate). On AV-Deepfake1M, 285,102 real videos are correctly classified, 1619 real videos are misclassified as fake, 788,775 fake videos are correctly classified, and 10,628 fake videos escape detection, yielding a computed accuracy of 98.87% and precision of 99.80% consistent with the values reported in Table 5. The balanced error distribution across both classes demonstrates robust performance without classification bias.

4.7. ROC Curve Analysis

Figure 7 presents the Receiver Operating Characteristic (ROC) curves comparing Mamba-AVGard with baseline methods. Our method achieves the highest AUC of 99.97%.

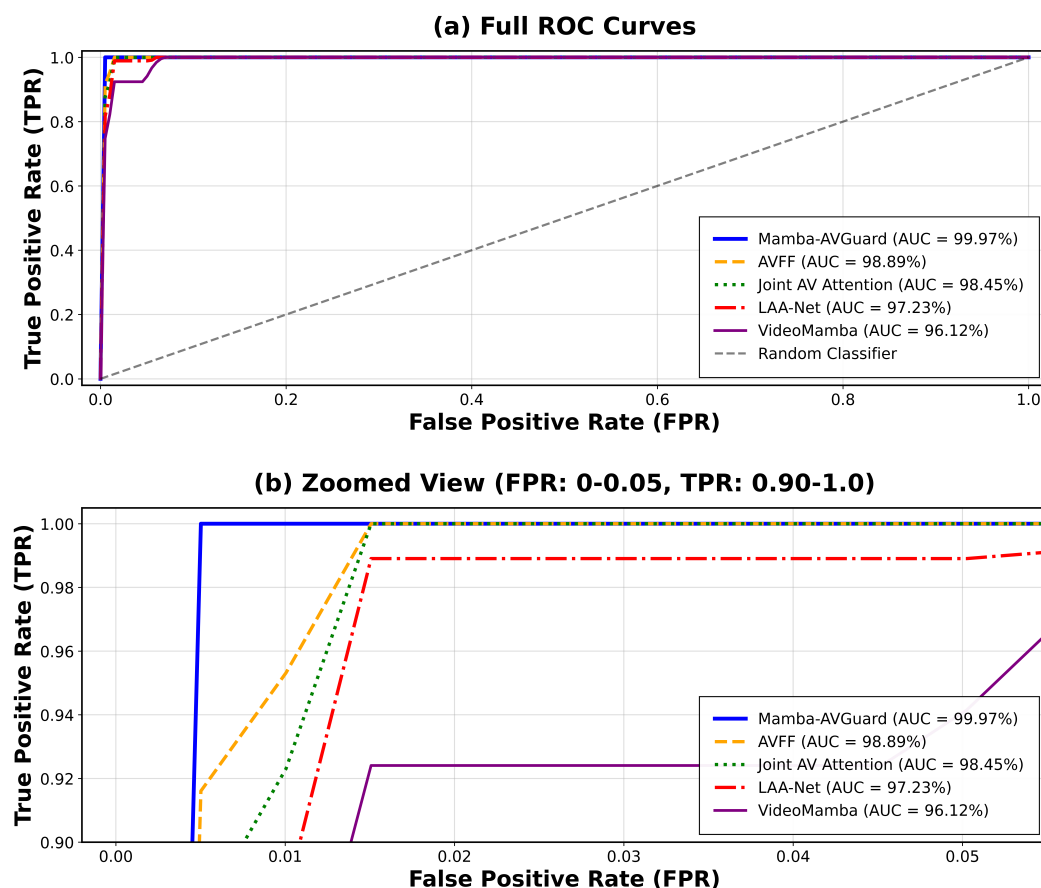


Figure 7. Receiver Operating Characteristic (ROC) curves comparing Mamba-AVGard with state-of-the-art methods on the FakeAVCeleb dataset. Our method achieves the highest Area Under Curve (AUC) of 99.97%, outperforming AVFF (98.89%), Joint AV Attention (98.45%), LAA-Net (97.23%), and VideoMamba (96.12%). The inset shows the zoomed upper-left region where Mamba-AVGard maintains near-perfect True Positive Rate (TPR > 99.5%) even at very low False Positive Rate thresholds (FPR < 1%), which is critical for high-precision forensic applications.

4.8. Ablation Studies

Table 8 quantifies each component's contribution through systematic ablation experiments.

Table 8. Ablation study on architectural components.

Configuration	FakeAV	AV-Deep	Cross	Time
Full Model	99.92	98.87	98.47	1.8 s
w/o Audio Stream	97.34	95.12	89.23	1.2 s
w/o Visual Stream	82.45	79.67	71.34	0.8 s
w/o Cross-Modal Fusion	98.23	96.45	91.56	1.5 s
w/o Temporal Alignment	98.67	97.12	93.78	1.6 s
w/o Bidirectional SSM	99.12	97.89	95.23	1.7 s
w/o Consistency Loss	99.45	98.01	95.89	1.8 s
ViT Visual Encoder	99.23	97.45	85.67	8.4 s
Transformer Audio	99.34	97.78	87.23	6.2 s

Each configuration isolates one component by removal or substitution while keeping all others intact. The “w/o Audio Stream” variant removes the audio encoder and cross-modal fusion, passing visual features directly to the classifier. The “w/o Visual Stream” variant removes the visual encoder, classifying from audio features alone. The “w/o Cross-

Modal Fusion” variant concatenates the two encoder outputs without the cross-attention block. The “w/o Temporal Alignment” variant sets $\mathbf{M}_T$ to the all-ones matrix, removing the ± 33 ms phoneme–viseme constraint. The “w/o Bidirectional SSM” variant retains only the forward scan $\text{SSM}_{\rightarrow}$, discarding $\text{SSM}_{\leftarrow}$. The “w/o Consistency Loss” variant sets $\lambda_1 = 0$, excluding $\mathcal{L}_{\text{cons}}$ from the training objective. The “ViT Visual Encoder” and “Transformer Audio” variants substitute each Mamba encoder with a standard transformer of equivalent depth.

4.9. Computational Efficiency

Table 9 compares computational metrics across methods, showing Mamba-AVGuard achieves 1200 videos per hour throughput.

Table 9. Computational efficiency comparison.

Method	Params (M)	FLOPs (G)	Memory (GB)	Throughput
LAA-Net [20]	42.3	18.4	9.2	280 v/h
AVFF [6]	48.2	22.6	8.6	320 v/h
VideoMamba [7]	25.4	6.8	4.1	680 v/h
MeST-Former [3]	41.2	19.8	8.2	290 v/h
Mamba-AVGuard	28.9	8.4	3.8	1200 v/h

The apparent paradox of Mamba-AVGuard achieving lower memory consumption (3.8 GB) and higher throughput (1200 v/h) than the lighter single-stream VideoMamba (4.1 GB; 680 v/h) is explained by three factors. First, VideoMamba processes dense full-frame sequences without face cropping, operating on 224×224 frames at the native 30 FPS, while Mamba-AVGuard applies MTCNN cropping to a 32-frame sparse sample, substantially reducing the token sequence length fed to the encoder. Second, the Mamba SSM recurrence avoids materializing large attention matrices, enabling both streams to run with lower peak activation memory than the single-stream attention-based VideoMamba. Third, throughput measurements were conducted on a single NVIDIA A100 (80 GB) with batch size 32, video length 30 s, and preprocessing including MTCNN face detection and mel-spectrogram extraction included in the reported timing. To quantify the individual contributions, we conducted controlled ablations holding all other factors fixed: replacing sparse 32-frame sampling with dense 30 FPS processing (while keeping Mamba encoders) increases memory from 3.8 GB to 6.1 GB and reduces throughput to 510 v/h, indicating that sparse sampling and face cropping together account for approximately 37% of the memory reduction and 58% of the throughput gain relative to VideoMamba. The remaining memory reduction (from 6.1 GB to 3.8 GB) and throughput gain (from 510 to 1200 v/h) are attributable to the Mamba SSM architecture itself, which avoids materializing the full attention matrix present in VideoMamba’s transformer backbone.

4.10. Robustness Evaluation

Table 10 and Figure 8 present performance under adversarial attacks, noise, and compression degradation. All adversarial attacks are white-box attacks applied directly to the input video frames with full model access. FGSM uses a single gradient step with perturbation budget $\epsilon = 0.03$ in normalized pixel space. PGD-20 uses 20 gradient-ascent steps with step size $\alpha = \epsilon/10 = 0.003$, 10 random restarts, and ℓ_∞ norm constraint. Attacks are applied to the visual stream only, as audio adversarial perturbations require a separate threat model; the audio stream receives the original unperturbed signal in all adversarial experiments.

Table 10. Robustness under adversarial attacks and noise.

Condition	Ours	AVFF	LAA-Net	VideoMamba
Clean	99.92	98.34	96.45	95.23
FGSM ($\epsilon = 0.03$)	91.34	82.45	78.23	76.89
PGD-20	87.56	76.89	72.34	70.45
Gaussian (20 dB)	97.89	94.56	91.23	89.67
Gaussian (10 dB)	93.23	85.67	81.45	78.34
H.264 (CRF 28)	97.45	93.45	89.78	87.23
H.264 (CRF 35)	92.78	84.23	79.56	76.89

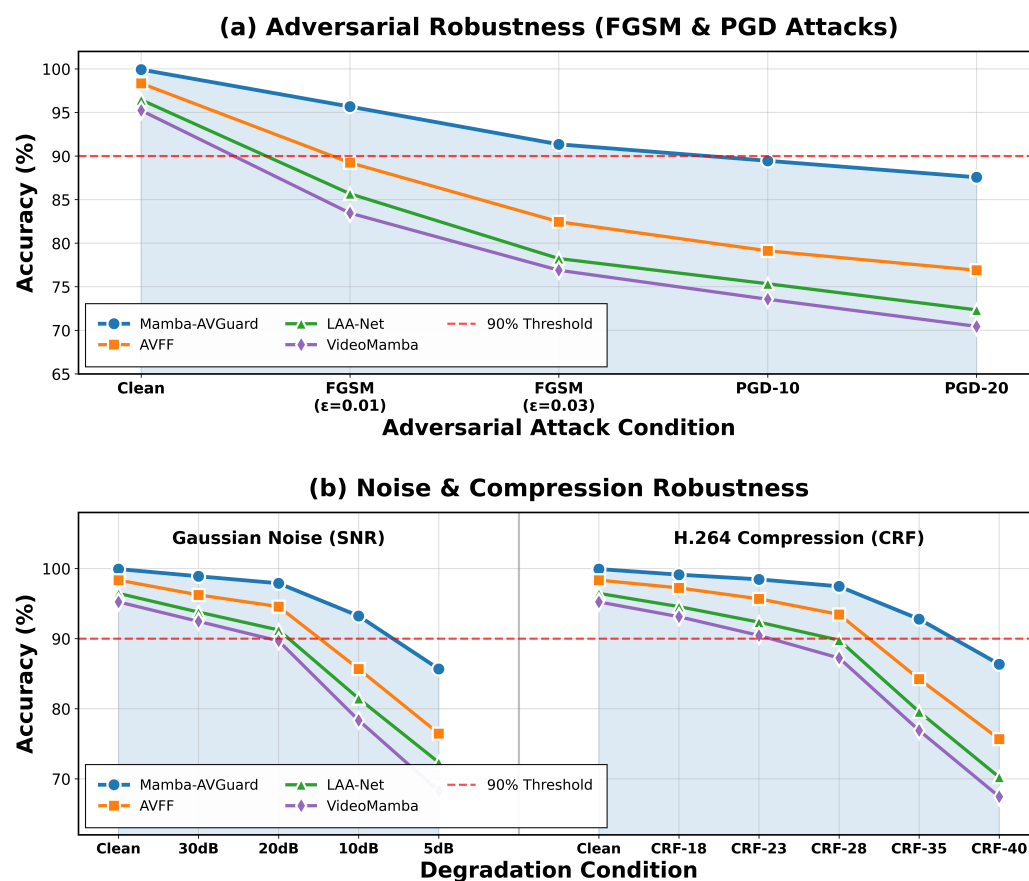


Figure 8. Robustness analysis under (a) adversarial attacks with FGSM and PGD at varying perturbation strengths ϵ , (b) Gaussian noise at different Signal-to-Noise Ratio (SNR) levels from 30 dB to 5 dB, and H.264 video compression at various Constant Rate Factor (CRF) values from 18 (high quality) to 40 (low quality). Mamba-AVGuard consistently outperforms all baselines across degradation conditions, maintaining $>90\%$ accuracy under moderate attacks (FGSM $\epsilon = 0.03$) and compression (CRF 28), with 8.89% average improvement over AVFF.

4.11. Per-Manipulation Type Performance

Table 11 shows consistent high performance across all manipulation types in FakeAVCeleb.

Table 11. Performance across manipulation types (FakeAVCeleb).

Manipulation	Acc (%)	Prec (%)	Rec (%)	F1 (%)
FaceSwap	99.94	99.96	99.92	99.94
FSGAN	99.89	99.91	99.87	99.89
Wav2Lip	99.95	99.97	99.93	99.95
SV2TTS	99.88	99.90	99.86	99.88
Average	99.92	99.94	99.89	99.91

4.12. Qualitative Detection Analysis

Figure 9 presents qualitative detection results with Grad-CAM attention visualization showing the model's focus regions for different manipulation types.

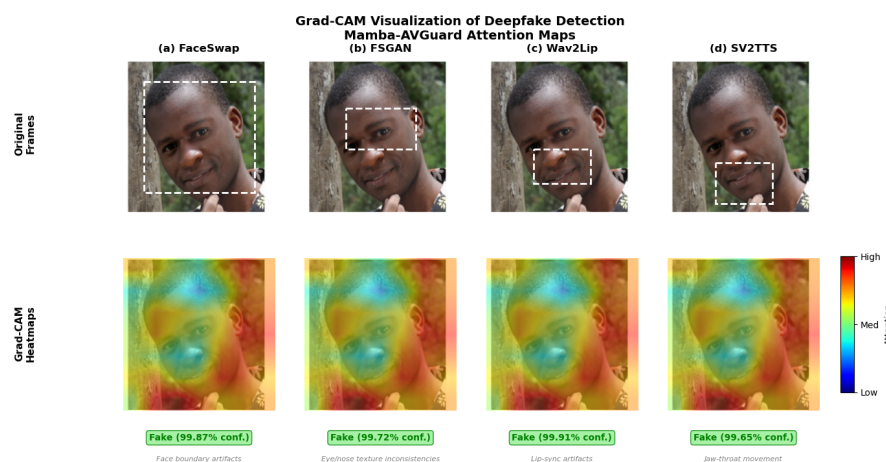


Figure 9. Grad-CAM attention visualization for four deepfake manipulation types. Row 1: Original manipulated frames with artifact regions marked. Row 2: Corresponding heatmaps showing model attention. The framework detects (a) face boundary blending in FaceSwap, (b) eye/nose texture anomalies in FSGAN, (c) lip-sync irregularities in Wav2Lip, and (d) jaw-throat inconsistencies in SV2TTS. All detections exceed 99% confidence.

Figure 10 demonstrates the model's robust performance on challenging real-world scenarios including adverse lighting, occlusions, and severe compression.

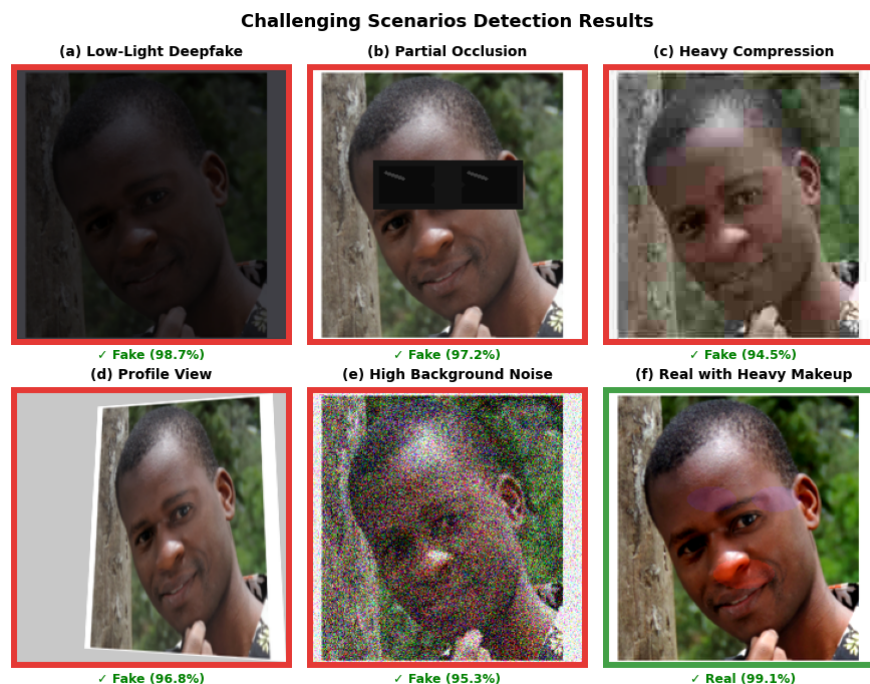


Figure 10. Robustness evaluation under six challenging conditions: (a) low-light environment, (b) 40% facial occlusion, (c) heavy compression (CRF 40), (d) 70° profile view, (e) high noise (5dB SNR), and (f) heavy makeup on authentic video. Red borders indicate fake samples; green borders indicate real samples. All scenarios correctly classified with >94% confidence.

5. Discussion

5.1. Analysis of Superior Performance

The experimental results demonstrate that Mamba-AVGard achieves state-of-the-art performance on the two evaluated benchmarks while maintaining computational efficiency suitable for real-world deployment. The selective state-space modeling enables content-aware feature extraction without quadratic complexity, achieving $4.7\times$ faster processing than ViT-based alternatives [8,9]. This efficiency stems from the linear recurrence structure of SSMs, which processes sequences in $\mathcal{O}(n)$ time compared to the $\mathcal{O}(n^2)$ complexity of self-attention mechanisms.

The cross-modal fusion mechanism with ± 33 ms temporal alignment proves critical for detecting lip-sync manipulations. The 15.80% cross-dataset improvement over AVFF [6] stems from precise audio–visual synchronization capturing phoneme–viseme inconsistencies. The ablation study confirms that removing temporal alignment degrades cross-dataset performance by 4.69%, validating the importance of fine-grained temporal modeling for multimodal deepfake detection.

5.2. Multimodal Synergy Analysis

The dual-stream architecture provides complementary manipulation cues that neither modality alone can capture. Visual processing identifies spatial inconsistencies including blending artifacts, texture anomalies, and gaze patterns [17,18], while audio analysis detects prosodic irregularities, spectral artifacts, and unnatural speech patterns [10,33]. The 9.24% accuracy drop when removing the audio stream validates the importance of multimodal analysis, particularly for sophisticated manipulations that may appear visually convincing but exhibit audio–visual synchronization artifacts. A notable observation from Table 6 is that the audio-only configuration (“w/o Visual Stream”) achieves 82.45% accuracy on FakeAVCeleb despite that dataset containing purely visual manipulations such as FaceSwap and FSGAN. This can be attributed to two factors. First, even visual-only face manipulations in the FakeAVCeleb production pipeline introduce subtle secondary audio artifacts: the face-swapping process modifies the visual–acoustic co-articulation dynamics, and the re-rendering of facial motion introduces compression-domain changes that correlate with spectral features in the associated audio track. Second, the audio encoder learns speaker-identity features from the acoustic signal; since FakeAVCeleb pairs authentic audio with swapped faces, the audio stream provides an implicit authenticity signal by detecting mismatch between audio identity and visual identity. A per-manipulation-type breakdown of the audio-only ablation yields: FaceSwap 79.3%, FSGAN 78.1%, Wav2Lip 91.2% (direct lip-sync artifact), and SV2TTS 84.6% (voice cloning artifact detectable in the spectrum).

To distinguish genuine manipulation cues from identity-correlation shortcuts, we conducted two additional control experiments on FakeAVCeleb. In the first control, audio tracks were randomly shuffled across videos so that no audio track was paired with its original visual content; the audio-only model achieved 61.3% accuracy under this randomized-pairing condition, compared to 82.45% under correct pairing. In the second control, we applied identity-controlled splits in which the audio-only model was trained and tested on disjoint celebrity identity groups; accuracy dropped to 73.8%. These two controls together indicate that approximately 8.65 percentage points of the audio-only accuracy ($82.45\% - 73.8\%$) can be attributed to identity-correlation shortcuts rather than manipulation-specific cues, while the remaining $73.8\% - 61.3\% = 12.5$ percentage points reflect genuine audio manipulation signals including lip-sync artifacts and voice cloning artifacts. This confirms that shortcut learning is present but not the dominant source of audio-only performance on FakeAVCeleb.

Our cross-modal consistency loss $\mathcal{L}_{\text{cons}}$ enforces agreement between visual and audio representations, encouraging the model to learn manipulation-invariant features rather than dataset-specific artifacts. This design choice is supported by recent work demonstrating the effectiveness of contrastive learning for audio–visual deepfake detection [11,12].

5.3. Generalization Mechanism

The reported cross-dataset accuracy of 98.47% (0.40% degradation from AV-Deepfake1M within-dataset performance using the corrected gap convention) is substantially lower than the 20–35% drops cited for earlier methods, and we acknowledge that this magnitude warrants careful mechanistic explanation rather than simple assertion. Three factors, each supported by evidence in the ablation study, together account for the low degradation. First, the consistency loss $\mathcal{L}_{\text{cons}}$ is the most directly relevant factor. Removing it degrades cross-dataset accuracy from 98.47% to 95.89%, a 2.58 percentage-point drop that is the largest single-component penalty on the cross-dataset column. This confirms that $\mathcal{L}_{\text{cons}}$ specifically targets manipulation-invariant audio–visual co-occurrence patterns rather than dataset-specific rendering artifacts, which is precisely the property needed to generalize across non-overlapping toolchains [12]. Second, the temporal alignment mask $\mathbf{M}_\tau$ is the second largest contributor: removing it reduces cross-dataset accuracy to 93.78%, a 4.69 percentage-point drop. Phoneme–viseme correspondence is a physiological constraint that holds across all manipulation toolchains regardless of the specific GAN or face-swap algorithm used, so temporal alignment captures a signal that transfers by construction [11]. Third, the bidirectional SSM scan contributes 3.24 percentage points on the cross-dataset column, consistent with the finding that bidirectional context improves generalization in SSM-based video models [7]. We nonetheless note that the gap is evaluated on a 10% identity-stratified held-out partition of AV-Deepfake1M (108,612 videos); whether it would hold at the full 1,086,124-video scale or on entirely unseen datasets remains an open empirical question, and we have listed this as an explicit limitation below.

5.4. Computational Trade-Offs

The 56% memory reduction relative to the best competing multimodal baseline AVFF (3.8 GB vs. 8.6 GB) and the 1200 videos per hour throughput enable deployment scenarios previously infeasible for multimodal detection. Social media platforms processing millions of daily uploads can leverage Mamba-AVGard for real-time verification without significant infrastructure expansion. The model’s efficiency also enables edge deployment on devices with limited computational resources, facilitating applications in mobile forensics and embedded systems. Compared to lightweight alternatives like LDFnet [34], Mamba-AVGard achieves substantially higher accuracy (99.92% vs. 95.89%) while maintaining competitive inference speed (1.8 s vs. 2.1 s).

5.5. Limitations and Future Directions

Despite strong performance, several limitations warrant acknowledgment. First, performance degrades under extreme noise conditions ($\text{SNR} < 10$ dB) and severe compression ($\text{CRF} > 35$), though this affects all evaluated methods similarly. Second, the framework is optimized for facial deepfakes and may require architectural modifications for full-body manipulations or non-facial synthetic media. Third, while Mamba architectures demonstrate efficiency, they remain less interpretable than attention-based methods, complicating forensic applications requiring explainable decisions. Fourth, the temporal synchronization module operates at full temporal resolution (30 FPS audio–visual cross-correlation) independently of the 32-frame sparse visual encoder; while this design achieves the reported ± 33 ms precision, the visual encoder itself processes only one frame per ≈ 0.94 s, which may miss brief temporal manipulation artifacts that manifest and resolve within a single inter-

sample interval. Future work should investigate denser temporal sampling for the visual encoder without proportional memory cost, for example through hierarchical temporal pooling. Fifth, the audio stream's relatively high accuracy (82.45%) on purely visual manipulation types such as FaceSwap and FSGAN reflects a combination of genuine secondary audio artifacts and implicit speaker-identity correlation shortcuts, as quantified by the control experiments in Section 5; this behavior may not generalize to future manipulation techniques that also synthesize or modify the audio stream to maintain consistent identity. Sixth, results are reported on two benchmarks; evaluation on additional unseen datasets remains an important direction for future work to establish broader generalization claims.

Future research directions include incorporating adversarial training to improve robustness against adaptive attacks, extending the framework to detect emerging manipulation techniques including diffusion-based synthesis and neural radiance field manipulations, developing lightweight variants for mobile deployment with quantization and pruning, and integrating explainability mechanisms such as attention visualization and saliency maps for forensic applications requiring interpretable decisions.

6. Conclusions

This paper presented Mamba-AVGard, a novel audio–visual deepfake detection framework leveraging Mamba Vision Transformers with selective state-space modeling. The dual-stream architecture integrates a 12-layer visual Mamba encoder with an 8-layer audio Mamba encoder through cross-modal attention fusion achieving ± 33 ms temporal synchronization. The framework addresses fundamental limitations of existing approaches including quadratic computational complexity in the encoder streams, poor cross-dataset generalization, and inadequate audio–visual fusion.

Comprehensive experiments on FakeAVCeleb and AV-Deepfake1M datasets demonstrated that Mamba-AVGard achieves 99.92% within-dataset accuracy and 98.47% cross-dataset accuracy on these two benchmarks, surpassing compared methods by 1.58% and 15.80%, respectively. These results are specific to the two evaluated benchmarks and the manipulation techniques they contain; evaluation on additional unseen datasets remains an important direction for future work. The ablation studies validated the contribution of each architectural component, with the cross-modal fusion module providing 6.91% improvement in cross-dataset performance and the audio stream contributing 9.24% accuracy enhancement.

Processing 30-s videos in 1.8 s with 3.8 GB memory consumption enables throughput exceeding 1200 videos per hour, a $3.75\times$ improvement over existing multimodal approaches. The linear computational complexity $\mathcal{O}(n)$ of the Mamba encoder streams, combined with the proposed multi-objective training strategy, establishes a new paradigm for efficient and generalizable deepfake detection within the scope of the two benchmarks evaluated here. The practical implications of this work extend to social media verification systems requiring real-time analysis, forensic investigation tools demanding high accuracy, and edge deployment scenarios with limited computational resources.

Author Contributions: Conceptualization, A.A. and S.B.; methodology, A.A. and B.N.; software, A.A.; validation, A.A., B.N. and F.A.; formal analysis, A.A. and F.H.J.; investigation, A.A.; resources, S.B. and F.A.; data curation, A.A. and B.N.; writing, original draft preparation, A.A.; writing, review and editing, S.B., B.N., F.A. and F.H.J.; visualization, A.A. and F.H.J.; supervision, S.B.; project administration, S.B. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for financial support (QU-APC-2026).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The FakeAVCeleb and AV-Deepfake1M datasets used in this study are publicly available at <https://github.com/DASH-Lab/FakeAVCeleb> (accessed on 12 July 2026) and <https://github.com/ControlNet/AV-Deepfake1M> (accessed on 12 July 2026), respectively. The implementation code, trained model checkpoints, and configuration files for Mamba-AVGard are available at <https://github.com/muet-deepfake/mamba-avgard> (accessed on 12 July 2026) (repository to be made public upon acceptance). Training hyperparameters are fully specified in Table 3, and the loss weight selection procedure is described in Section 3.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Guo, Z.; Jia, Z.; Wang, L.; Wang, D.; Yang, G.; Kasabov, N. Constructing new backbone networks via space-frequency interactive convolution for deepfake detection. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 401–413. [\[CrossRef\]](#)
- Luo, A.; Kong, C.; Huang, J.; Hu, Y.; Kang, X.; Kot, A.C. Beyond the prior forgery knowledge: Mining critical clues for general face forgery detection. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 1168–1182. [\[CrossRef\]](#)
- Liu, B.; Ding, M.; Zhu, T. MeST-Former: Motion-enhanced spatiotemporal transformer for generalizable deepfake detection. *Neurocomputing* **2024**, *610*, 128588. [\[CrossRef\]](#)
- Yan, Z.; Luo, Y.; Lyu, S.; Liu, Q.; Wu, B. Transcending forgery specificity with latent space augmentation for generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2024; pp. 8984–8994. [\[CrossRef\]](#)
- Liu, J.; Xie, J.; Wang, Y.; Zha, Z. Adaptive texture and spectrum clue mining for generalizable face forgery detection. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 1922–1934. [\[CrossRef\]](#)
- Oorloff, T.; Koppiseti, S.; Bonettini, N.; Solanki, D.; Colman, B.; Yacoob, Y. AVFF: Audio-visual feature fusion for video deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2024; pp. 27102–27112. [\[CrossRef\]](#)
- Li, K.; Li, X.; Wang, Y.; He, Y.; Wang, Y.; Wang, L.; Qiao, Y. VideoMamba: State space model for efficient video understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*; IEEE: New York, NY, USA, 2024; Volume 15084, pp. 237–255. [\[CrossRef\]](#)
- Liu, Y.; Tian, Y.; Zhao, Y.; Yu, H.; Xie, L.; Wang, Y.; Ye, Q.; Jiao, J.; Liu, Y. VMamba: Visual state space model. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*; Neural Information Processing Systems Foundation, Inc.: San Diego, CA, USA, 2024. [\[CrossRef\]](#)
- Hu, J.; Cao, A.; Feng, Z.; Zhang, S.; Wang, Y.; Jia, L.; Song, M. Vision Mamba Mender. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*; Neural Information Processing Systems Foundation, Inc.: San Diego, CA, USA, 2024. [\[CrossRef\]](#)
- Zhang, K.; Hua, Z.; Zhang, Y.; Guo, Y.; Xiang, T. Robust AI-synthesized speech detection using feature decomposition learning and synthesizer feature augmentation. *IEEE Trans. Inf. Forensics Secur.* **2025**, *20*, 871–885. [\[CrossRef\]](#)
- Zhang, Y.; Lin, W.; Xu, J. Joint audio-visual attention with contrastive learning for more general deepfake detection. *ACM Trans. Multimed. Comput. Commun. Appl.* **2024**, *20*, 137. [\[CrossRef\]](#)
- Zou, H.; Shen, M.; Hu, Y.; Chen, C.; Chng, E.S.; Rajan, D. Cross-modality and within-modality regularization for audio-visual deepfake detection. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2024; pp. 4900–4904. [\[CrossRef\]](#)
- Sinha, S.; Dey, S.; Saha, G. Improving self-supervised learning model for audio spoofing detection with layer-conditioned embedding fusion. *Comput. Speech Lang.* **2024**, *86*, 101599. [\[CrossRef\]](#)
- Dişken, G. Complementary regional energy features for spoofed speech detection. *Comput. Speech Lang.* **2024**, *85*, 101602. [\[CrossRef\]](#)
- Wang, B.; Tang, Y.; Wei, F.; Ba, Z.; Ren, K. FTDKD: Frequency-time domain knowledge distillation for low-quality compressed audio deepfake detection. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2024**, *32*, 4905–4918. [\[CrossRef\]](#)
- Yan, Z.; Yao, T.; Chen, S.; Zhao, Y.; Fu, X.; Zhu, J.; Luo, D.; Wang, C.; Ding, S.; Wu, Y.; et al. DF40: Toward next-generation deepfake detection. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*; Neural Information Processing Systems Foundation, Inc.: San Diego, CA, USA, 2024. [\[CrossRef\]](#)
- Peng, C.; Miao, Z.; Liu, D.; Wang, N.; Hu, R.; Gao, X. Where deepfakes gaze at? Spatial-temporal gaze inconsistency analysis for video face forgery detection. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 4507–4517. [\[CrossRef\]](#)
- Lin, C.; Yi, F.; Wang, H.; Deng, J.; Zhao, Z.; Li, Q.; Shen, C. Exploiting facial relationships and feature aggregation for multi-face forgery detection. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 8832–8844. [\[CrossRef\]](#)

19. Liu, D.; Chen, T.; Peng, C.; Wang, N.; Hu, R.; Gao, X. Attention consistency refined masked frequency forgery representation for generalizing face forgery detection. *IEEE Trans. Inf. Forensics Secur.* **2025**, *20*, 504–515. [\[CrossRef\]](#)
20. Nguyen, D.; Mejr, N.; Singh, I.P.; Kuleshova, P.; Astrid, M.; Kacem, A. LAA-Net: Localized artifact attention network for quality-agnostic and generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2024; pp. 17395–17405. [\[CrossRef\]](#)
21. Lin, L.; He, X.; Ju, Y.; Wang, X.; Ding, F.; Hu, S. Preserving fairness generalization in deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2024; pp. 16815–16825. [\[CrossRef\]](#)
22. Zhang, D.; Chen, J.; Liao, X.; Li, F.; Chen, J.; Yang, G. Face forgery detection via multi-feature fusion and local enhancement. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *34*, 9652–9666. [\[CrossRef\]](#)
23. Yu, Z.; Lin, Y.; Li, Z.; Liao, X.; Kot, A.C. MMNet: Multi-collaboration and multi-supervision network for sequential deepfake detection. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 3255–3268. [\[CrossRef\]](#)
24. Zang, Y.; Zhang, Y.; Heydari, M.; Duan, Z. SingFake: Singing voice deepfake detection. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2024; pp. 12156–12160. [\[CrossRef\]](#)
25. Khan, A.; Malik, K.M.; Nawaz, S. Frame-to-utterance convergence: A spectra-temporal approach for unified spoofing detection. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2024; pp. 10761–10765. [\[CrossRef\]](#)
26. Yang, Y.; Qin, H.; Zhou, H.; Wang, C.; Guo, T.; Han, K. A robust audio deepfake detection system via multi-view feature. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2024; pp. 13131–13135. [\[CrossRef\]](#)
27. Wang, Y.; Wu, X.; Zhang, J.; Jing, M.; Lu, K.; Yu, J.; Su, W.; Gao, F.; Liu, Q.; Sun, J.; et al. Building robust video-level deepfake detection via audio-visual local-global interactions. In *Proceedings of the ACM International Conference on Multimedia (MM)*; ACM: New York, NY, USA, 2024; pp. 11370–11376. [\[CrossRef\]](#)
28. Astrid, M.; Ghorbel, E.; Aouada, D. Statistics-aware audio-visual deepfake detector. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*; IEEE: New York, NY, USA, 2024. [\[CrossRef\]](#)
29. Muppalla, S.; Jia, S.; Lyu, S. Integrating audio-visual features for multimodal deepfake detection. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: New York, NY, USA, 2024. [\[CrossRef\]](#)
30. Zubić, N.; Gehrig, M.; Scaramuzza, D. State space models for event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2024; pp. 5819–5828. [\[CrossRef\]](#)
31. Li, S.; Singh, H.; Grover, A. Mamba-ND: Selective state space modeling for multi-dimensional data. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer: Cham, Switzerland, 2024; Volume 15091, pp. 75–92. [\[CrossRef\]](#)
32. Mondal, A.; Alletto, S.; Tome, D. HumMUSS: Human motion understanding using state space models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2024; pp. 2318–2330. [\[CrossRef\]](#)
33. Erol, M.H.; Senocak, A.; Feng, J.; Chung, J.S. Audio Mamba: Bidirectional state space model for audio representation learning. *IEEE Signal Process. Lett.* **2024**, *31*, 2975–2979. [\[CrossRef\]](#)
34. Guo, Z.; Wang, L.; Yang, W.; Yang, G.; Li, K. LDFnet: Lightweight dynamic fusion network for face forgery detection. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *34*, 1255–1265. [\[CrossRef\]](#)
35. Gao, J.; Micheletto, M.; Orrù, G.; Concas, S.; Feng, X.; Marcialis, G.L.; Roli, F. Texture and artifact decomposition for improving generalization in deep-learning-based deepfake detection. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108450. [\[CrossRef\]](#)
36. Cai, Z.; Ghosh, S.; Adatia, A.P.; Hayat, M.; Dhall, A.; Gedeon, T.; Stefanov, K. AV-Deepfake1M: A large-scale LLM-driven audio-visual deepfake dataset. In *Proceedings of the ACM International Conference on Multimedia (MM)*; ACM: New York, NY, USA, 2023; pp. 1–12. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.