RETHINKING TEMPORAL FORGERY LOCALIZATION: FROM SEMANTIC-ORIENTED DENSE DETECTION TO FORGERY-SENSITIVE SPARSE LOCALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Temporal Forgery Localization (TFL) aims to identify the temporal boundaries of manipulated segments within otherwise authentic content. Given the commonalities between TFL and Temporal Action Localization (TAL), many TFL methods build on TAL-derived feature modeling and localization designs, with further forgery-specific adaptations. However, these adaptations often address representation learning and localization separately, leaving the preservation of manipulation-sensitive responses and the modeling of sparse interval structure insufficiently explored jointly. We propose **DiST-TFL**, a Diffusion-guided Sparse Temporal Forgery Localization framework that jointly adapts representation learning and temporal localization to TFL-specific characteristics. Diffusion-based feature refinement provides a reference for revealing manipulation-sensitive responses, which are propagated across scales through a hierarchical temporal feature pyramid. A sparse query-based localization module then predicts a compact set of forged intervals without Non-Maximum Suppression (NMS). Experiments on LAV-DF and AV-Deepfake1M demonstrate substantial improvements in overall localization performance, while ablation studies validate the complementary roles of diffusion-guided feature modeling and sparse localization.

1 INTRODUCTION

Recent advances in generative models have substantially improved the realism of manipulated audio-visual content while lowering the barrier to its creation, raising growing concerns over misinformation, identity fraud, and media authenticity. While conventional deepfake detection primarily determines whether an entire video is real or fake, recent studies have highlighted a more challenging audio-visual scenario, where manipulations affect only short temporal segments of one or both modalities within otherwise authentic content (Cai et al., 2022; 2024). In such cases, whole-video deepfake detection cannot identify which temporal segments are forged or delineate their precise boundaries. This has motivated increasing attention to *Temporal Forgery Localization* (TFL), which predicts the start and end timestamps of forged segments.

Temporal Forgery Localization has received increasing attention in recent years. BA-TFD (Cai et al., 2022) and BA-TFD+ (Cai et al., 2023) introduced audio-visual temporal localization frameworks with explicit boundary modeling, establishing strong baselines for locating manipulated segments. UMMAFormer (Zhang et al., 2023) subsequently improved audio-visual interaction and long-range temporal modeling through multi-scale representations. More recent studies have introduced further forgery-specific adaptations. AuViRe (Koutlis & Papadopoulos, 2026), for instance, exploits cross-modal reconstruction discrepancy as evidence of manipulation, while RegQAV (Zhu et al., 2025) reformulates temporal localization with query-based set prediction. These studies show that TFL has already evolved beyond a direct reuse of generic temporal localization techniques, with task-specific adaptations introduced to better model forgery-specific characteristics. Despite these advances, an important question remains: how can subtle forgery evidence be effectively preserved while supporting compact and direct temporal localization?

This remaining gap can be understood from two aspects, as illustrated in Fig. 1. **At the feature-representation level** (Fig. 1(a)), representative TFL methods, such as UMMAFormer (Zhang et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

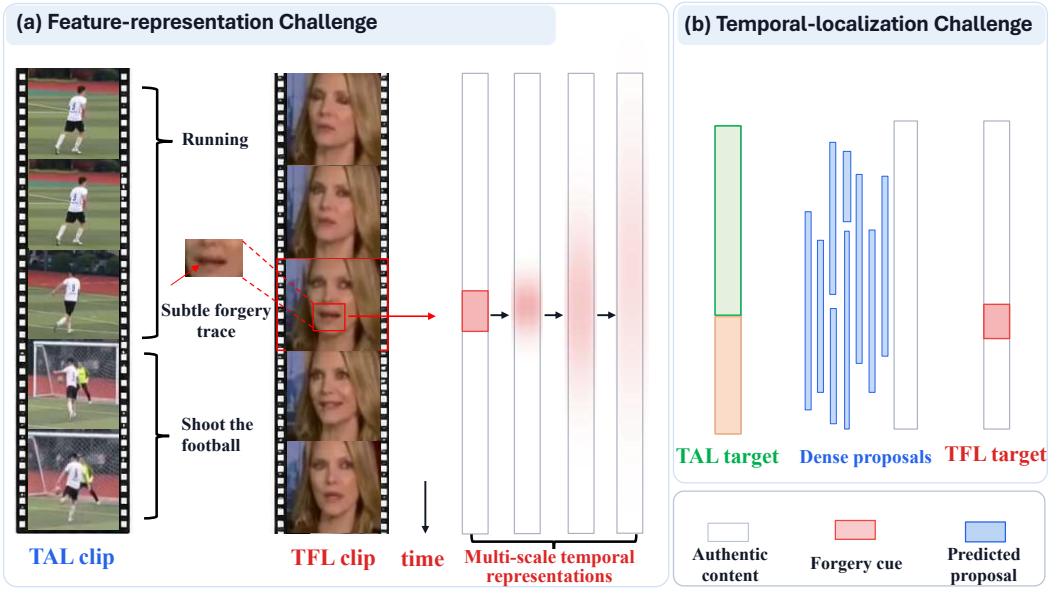


Figure 1: **Motivation for joint adaptation in Temporal Forgery Localization.** Manipulation-sensitive responses may become less distinguishable in multi-scale temporal representations, while dense prediction can be redundant for typically sparse forged intervals. This motivates jointly considering forgery-sensitive representation learning and compact temporal prediction.

2023) and AuViRe (Koutlis & Papadopoulos, 2026), construct multi-scale temporal representations through progressive temporal aggregation and feature-pyramid fusion. While effective for modeling broader context, TFL often differs from TAL in that the manipulated content may remain semantically similar to authentic content, with only subtle local forgery traces indicating manipulation. As a result, manipulation-sensitive responses to these traces may become less distinguishable in coarser representations, and cross-scale fusion does not explicitly guarantee their preservation. **At the temporal-localization level** (Fig. 1(b)), dense prediction provides broad coverage but also produces many overlapping candidates. This design is natural for temporally extended semantic targets, but may be unnecessarily redundant for TFL, where forged intervals are typically sparse. RegQAV (Zhu et al., 2025) has introduced query-based localization to TFL, demonstrating the potential of direct set prediction, but it does not explicitly address the preservation of weak manipulation cues across temporal scales. These observations motivate jointly considering forgery-sensitive representation learning and compact temporal prediction.

Based on these observations, we propose **DiST-TFL, a Diffusion-guided Sparse Temporal Forgery Localization framework** that jointly addresses forgery-sensitive representation learning and compact temporal prediction. For feature modeling, **Diffusion-guided Hierarchical Feature Modeling** uses diffusion-based refinement as a reference for the original audio-visual features. Cross-attention between the original and diffusion-refined reference representations highlights forgery-related discrepancies before temporal aggregation, which are then propagated across scales through hierarchical modeling and feature-pyramid fusion. For temporal localization, **Sparse Query-based Temporal Localization** directly predicts a compact set of forged intervals with learnable queries, whose temporal references are initialized from high-confidence locations in the multi-scale memory. One-to-one matching establishes direct prediction-to-target assignments during training, eliminating the need for NMS at inference.

We evaluate DiST-TFL on two widely used audio-visual temporal forgery localization benchmarks, LAV-DF and AV-Deepfake1M. The results show that DiST-TFL improves overall localization performance and is particularly effective under constrained prediction budgets, supporting the use of sparse set prediction for TFL. Ablation studies further validate the complementary effects of Diffusion-guided feature modeling and query-based localization. Our main contributions are summarized as follows:

- We revisit the design of Temporal Forgery Localization and identify two remaining challenges: preserving subtle forgery evidence across multi-scale temporal representations and reducing redundant dense prediction for typically sparse forged intervals.
- We propose **DiST-TFL**, a Diffusion-guided Sparse Temporal Forgery Localization framework that couples diffusion-guided hierarchical feature modeling with sparse set prediction.
- Extensive experiments on LAV-DF and AV-Deepfake1M demonstrate the effectiveness of DiST-TFL, while ablation studies confirm the complementary roles of diffusion-guided feature modeling and sparse temporal localization.

2 RELATED WORK

2.1 MULTIMODAL DEEPFAKE DETECTION AND TEMPORAL FORGERY LOCALIZATION

Multimodal deepfake detection leverages audio-visual correspondences to identify manipulated videos. Representative methods model cross-modal inconsistency (Yang et al., 2023), combine inter- and intra-modal regularization (Zou et al., 2024), or learn audio-visual correspondence through self-supervised objectives (Oorloff et al., 2024). More recently, audio-visual prompt learning and frame-level cross-modal matching have also been explored (Miao et al., 2025). These methods mainly produce video-level authenticity predictions and do not explicitly localize manipulated intervals.

Temporal Forgery Localization (TFL) extends deepfake detection to temporal interval prediction. MDS (Chugh et al., 2020) first studies TFL through audio-visual modality dissonance. BA-TFD (Cai et al., 2022) and BA-TFD+ (Cai et al., 2023) introduce boundary-aware localization, while UMMAFormer (Zhang et al., 2023) improves audio-visual interaction and temporal modeling with reconstruction-based anomaly attention and a multi-scale feature pyramid. DiMoDif (Koutlis & Papadopoulos, 2024) and AuViRe (Koutlis & Papadopoulos, 2026) further exploit cross-modal representation discrepancies, while RegQAV (Zhu et al., 2025) explores query-based end-to-end localization. Despite these advances, jointly coupling forgery-sensitive representation learning with sparse temporal prediction remains less explored.

2.2 QUERY-BASED DETECTION AND DIFFUSION MODELS

Query-based detection originates from object detection, where DETR (Carion et al., 2020) reformulates detection as direct set prediction using learnable object queries and bipartite matching, removing the need for anchor generation and NMS. This paradigm was subsequently extended to temporal action localization. TadTR (Liu et al., 2022) predicts action intervals with temporal queries and deformable attention, while TE-TAD (Kim et al., 2024) further develops fully end-to-end temporal detection with adaptive query selection. In TFL, RegQAV (Zhu et al., 2025) first introduces query-based localization, using register-enhanced audio-visual representations for query generation and temporal decoding. These studies demonstrate the effectiveness of set prediction for temporal localization. However, existing query-based TFL mainly focuses on the prediction formulation, whereas our work focuses on coupling sparse prediction with representations that preserve manipulation-sensitive responses.

Diffusion models (Ho et al., 2020; Song et al., 2020) have also been extended beyond generation to discriminative tasks, including anomaly detection (Wyatt et al., 2022), object detection (Chen et al., 2023), and temporal action localization (Nag et al., 2023). More recently, ActivityForensics (Bao et al., 2026) introduces TADiff for visual-only temporal forgery localization, applying diffusion regularization to multi-scale temporal features to strengthen forgery cues. However, TADiff operates after the multi-scale Transformer network and retains dense per-location prediction, while manipulation-sensitive responses may already have been weakened by earlier temporal downsampling. In contrast, DiST-TFL applies diffusion refinement to audio-visual features before hierarchical temporal modeling and couples the resulting multi-scale representations with sparse query-based localization.

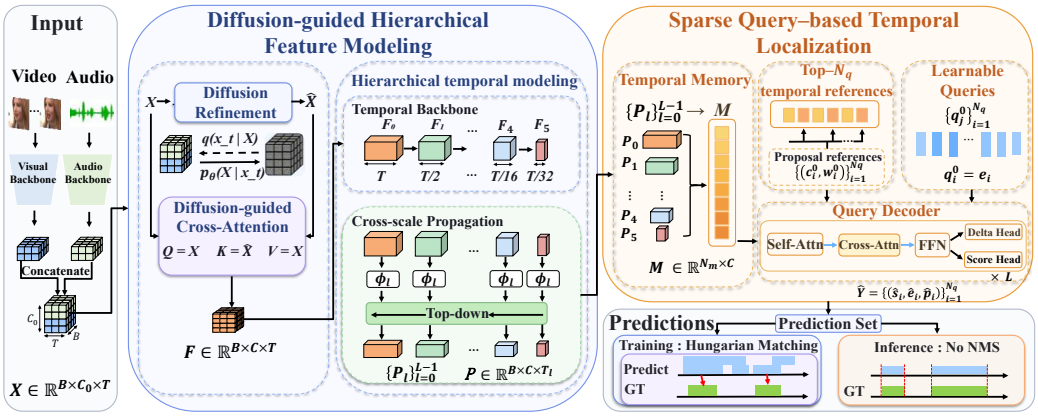


Figure 2: Overview of the DiST-TFL Framework.

3 METHODOLOGY

3.1 OVERVIEW FRAMEWORK OF DIST-TFL

Let $\mathbf{X} \in \mathbb{R}^{B\times C_0\times T}$ denote the input audio-visual representation, with batch size B, feature dimension C_0, and temporal length T. As illustrated in Fig. 2, DiST-TFL consists of diffusion-guided hierarchical feature modeling and sparse query-based temporal localization. Diffusion refinement produces a reference $\hat{\mathbf{X}}$ from the input $\mathbf{X}$ for diffusion-guided feature interaction. The resulting features are processed by a hierarchical temporal backbone with feature-pyramid fusion to obtain multi-scale representations $\{\mathbf{P}_l\}_{l=0}^{L-1}$. These features are organized into a temporal memory $\mathcal{M}$, from which high-confidence locations initialize temporal references for a fixed set of N_q learnable queries $\{\mathbf{q}_i\}_{i=1}^{N_q}$. The queries are progressively refined into a compact prediction set $\hat{\mathcal{Y}} = \{(\hat{s}_i, \hat{e}_i, \hat{p}_i)\}_{i=1}^{N_q}$. Training uses one-to-one matching, enabling direct prediction without NMS at inference.

3.2 DIFFUSION-GUIDED HIERARCHICAL FEATURE MODELING

This module is designed to preserve manipulation-sensitive responses across temporal resolutions. Diffusion-based feature refinement first provides a manipulation-sensitive reference at the original resolution, which is then integrated into hierarchical temporal modeling and propagated across scales through feature-pyramid fusion.

Diffusion-based Feature Refinement. We employ a lightweight diffusion-inspired feature refinement process, following the noise perturbation and deterministic refinement scheme of DDPM/DDIM (Ho et al., 2020; Song et al., 2020). Given the input representation $\mathbf{X}$, a noisy feature $\mathbf{X}_t$ is obtained at diffusion step t by adding Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. A timestep-conditioned denoiser D_θ then predicts a task-oriented perturbation and produces a refined reference representation:

$$\mathbf{X}_t = \sqrt{\bar{\alpha}_t}\mathbf{X} + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \qquad \hat{\mathbf{X}}^{(t)} = \frac{\mathbf{X}_t - \sqrt{1-\bar{\alpha}_t}D_\theta(\mathbf{X}_t, t)}{\sqrt{\bar{\alpha}_t}}. \tag{1}$$

The coefficient $\bar{\alpha}_t$ is the cumulative product of the noise-schedule coefficients up to step t. In our implementation, the refinement process is initialized at the fixed timestep $t = 2$ and follows the deterministic reverse sequence $t = 2, 1, 0$ during both training and inference. The final output of this reverse process is denoted by $\hat{\mathbf{X}}$.

We do not introduce an additional diffusion reconstruction loss; instead, D_θ is jointly optimized with the entire localization network through the downstream localization objective in Eq. 9. Rather than explicitly measuring a reconstruction residual, we use $\hat{\mathbf{X}}$ as a reference for feature interaction. After feature projection and temporal positional encoding, the original representation provides the

query and value, while the **refined reference representation** provides the key:

$$\mathbf{F} = \mathrm{Attn}\left(\mathbf{Q} = \mathbf{X}, \mathbf{K} = \hat{\mathbf{X}}, \mathbf{V} = \mathbf{X}\right), \tag{2}$$

where $\mathrm{Attn}(\cdot)$ denotes cross-attention and $\mathbf{F}$ is the resulting diffusion-guided representation. In this way, the diffusion-refined reference interacts with the original representation without replacing its content. This interaction allows the original features to attend to a noise-conditioned reference, yielding representations that are more effective for downstream localization.

Hierarchical Temporal Representation. Following the hierarchical feature-pyramid design of ActionFormer (Zhang et al., 2022), we organize the diffusion-guided representation $\mathbf{F}$ into a multi-scale temporal hierarchy. Successive temporal blocks progressively reduce the temporal resolution while enlarging the receptive field, producing features $\{\mathbf{F}_l\}_{l=0}^{L-1}$, where $\mathbf{F}_l \in \mathbb{R}^{B\times C\times T_l}$, $T_l = T/s_l$, and s_l denotes the temporal stride at level l. Since diffusion-guided interaction is performed before hierarchical modeling, manipulation-sensitive responses are introduced before successive temporal downsampling and carried into the resulting multi-scale representations.

The hierarchical features are further fused through a top-down feature pyramid. Each level is first projected to a common feature dimension, and higher-level information is progressively propagated to finer temporal resolutions:

$$\mathbf{H}_l = \phi_l(\mathbf{F}_l) + \mathrm{Up}\left(\mathbf{H}_{l+1}\right), \qquad l = L-2, \ldots, 0, \tag{3}$$

with $\mathbf{H}_{L-1} = \phi_{L-1}(\mathbf{F}_{L-1})$, where $\phi_l(\cdot)$ denotes the level-wise feature projection and $\mathrm{Up}(\cdot)$ denotes temporal upsampling to match the resolution of level l. A temporal convolution and normalization are then applied at each level to obtain the final pyramid representations $\{\mathbf{P}_l\}_{l=0}^{L-1}$. In this way, manipulation-sensitive responses induced by diffusion-guided interaction are propagated across temporal resolutions while broader temporal context is incorporated, providing the multi-scale features for subsequent sparse temporal localization.

3.3 Sparse Query-based Temporal Localization

This module reduces redundant dense prediction by formulating temporal localization as sparse set prediction. The multi-scale features are organized into a unified temporal memory, which provides temporal references for query initialization and subsequent iterative refinement.

Multi-scale Temporal Memory. Given the pyramid representations $\{\mathbf{P}_l\}_{l=0}^{L-1}$, each level is converted into temporal tokens augmented with sinusoidal positional encodings and learnable level embeddings. Tokens from all levels are concatenated to form a unified memory $\mathcal{M} \in \mathbb{R}^{B\times N_m\times C}$, where N_m is the total number of tokens. Each memory token $\mathbf{m}_j$ is associated with a normalized temporal coordinate $r_j \in [0, 1]$, preserving its temporal location and providing references for query initialization.

Proposal-guided Query Initialization. Inspired by the query-based localization paradigm of RegQAV (Zhu et al., 2025), we represent forged intervals with a fixed set of N_q queries. Unlike deriving query content from input representations, we decouple query content from temporal initialization. Specifically, a foreground seed scoring head $g_{\mathrm{seed}}(\cdot)$ predicts a score $a_j = g_{\mathrm{seed}}(\mathbf{m}_j)$ for each memory token $\mathbf{m}_j$, indicating its likelihood of serving as a foreground temporal seed. The seed scoring head g_{seed} is directly supervised with sigmoid focal loss, where foreground/background targets are assigned according to one-to-one Hungarian matching: matched proposals are labeled as foreground and unmatched proposals as background. We then select the N_q memory tokens with the highest foreground scores as temporal seeds, and denote their memory indices by $\{\mathcal{I}_i\}_{i=1}^{N_q}$. The normalized temporal coordinate of each selected seed initializes the proposal center, while the corresponding memory feature is used to predict the initial width:

$$c_i^0 = r_{\mathcal{I}_i}, \qquad w_i^0 = g_w(\mathbf{m}_{\mathcal{I}_i}), \qquad \mathbf{q}_i^0 = \mathbf{e}_i, \qquad i = 1, \ldots, N_q, \tag{4}$$

where c_i^0 and w_i^0 denote the initial center and width of the i-th temporal proposal, respectively, $g_w(\cdot)$ is the width prediction head, and $\mathbf{e}_i$ is the learnable embedding that initializes the query content $\mathbf{q}_i^0$. Thus, the selected memory features determine the initial temporal references but are not injected into the query content.

Proposal-guided Iterative Refinement. The queries are refined through a temporal decoder with self-attention, memory cross-attention, and feed-forward layers. At decoder layer k, each query attends only to memory tokens within a local region around its current proposal:

$$\mathcal{W}_i^k = \left\{ j \;\middle|\; |r_j - c_i^k| \le \gamma w_i^k \right\}, \tag{5}$$

where $\mathcal{W}_i^k$ denotes the set of memory-token indices attended by the i-th query at decoder layer k, and γ controls the temporal context range around the current proposal (c_i^k, w_i^k).

At each layer, the proposal is updated using the predicted center and log-scale width offsets:

$$c_i^{k+1} = c_i^k + \Delta c_i^k w_i^k, \qquad w_i^{k+1} = w_i^k \exp(\Delta w_i^k), \tag{6}$$

where Δc_i^k and Δw_i^k denote the predicted center and log-scale width offsets, respectively. The refined proposals are constrained to valid normalized temporal ranges.

After N_d decoder layers, each center-width proposal $(c_i^{N_d}, w_i^{N_d})$ is converted to normalized boundaries $(\hat{s}_i, \hat{e}_i)$. Together with the forgery confidence $\hat{p}_i$, the predictions form $\hat{\mathcal{Y}} = \{(\hat{s}_i, \hat{e}_i, \hat{p}_i)\}_{i=1}^{N_q}$. At inference, they are mapped to the original timeline and filtered by confidence without NMS.

3.4 Training and Inference

Training. We train the sparse localization module with one-to-one bipartite matching between the N_q predictions and N_{gt} ground-truth forged intervals. Let $\hat{y}_i = (\hat{p}_i, \hat{\mathbf{b}}_i)$ denote the i-th prediction, where $\hat{p}_i$ is the forgery confidence predicted by the decoder classification head and $\hat{\mathbf{b}}_i = (\hat{s}_i, \hat{e}_i)$ is the predicted temporal interval. Let $y_n = (1, \mathbf{b}_n)$ denote the n-th ground-truth target, where $\mathbf{b}_n = (s_n, e_n)$ is its temporal interval and 1 indicates the foreground label, with $n = 1, \ldots, N_{\mathrm{gt}}$.

For each prediction–target pair, we define the matching cost as

$$\mathcal{C}(\hat{y}_i, y_n) = \lambda_{\mathrm{cls}}^m \mathcal{C}_{\mathrm{cls}}(\hat{p}_i) + \lambda_{\mathrm{L1}}^m \left\| \hat{\mathbf{b}}_i - \mathbf{b}_n \right\|_1 + \lambda_{\mathrm{GIoU}}^m \left[1 - \mathrm{GIoU}(\hat{\mathbf{b}}_i, \mathbf{b}_n) \right], \tag{7}$$

where $\mathcal{C}_{\mathrm{cls}}$ denotes the sigmoid focal matching cost, while λ_{cls}^m, λ_{L1}^m, and $\lambda_{\mathrm{GIoU}}^m$ control the contributions of the classification, boundary ℓ_1, and temporal GIoU terms, respectively.

The optimal assignment is obtained using the Hungarian algorithm (Kuhn, 1955):

$$\hat{\sigma} = \arg\min_{\sigma} \sum_{n=1}^{N_{\mathrm{gt}}} \mathcal{C}\left(\hat{y}_{\sigma(n)}, y_n\right), \tag{8}$$

where σ denotes a one-to-one assignment from the ground-truth targets to a subset of the predictions, and $\hat{\sigma}$ is the optimal assignment.

Since TFL has a single foreground category, the decoder classification head is supervised with sigmoid focal loss, with matched queries treated as foreground and unmatched queries as background. Matched intervals are further supervised by an ℓ_1 boundary loss and a temporal GIoU loss. The overall localization objective is

$$\mathcal{L} = \lambda_{\mathrm{cls}} \mathcal{L}_{\mathrm{cls}} + \lambda_{\mathrm{L1}} \mathcal{L}_{\mathrm{L1}} + \lambda_{\mathrm{GIoU}} \mathcal{L}_{\mathrm{GIoU}}, \tag{9}$$

where $\mathcal{L}_{\mathrm{cls}}$ is the sigmoid focal classification loss over decoder queries, while $\mathcal{L}_{\mathrm{L1}}$ and $\mathcal{L}_{\mathrm{GIoU}}$ denote the boundary ℓ_1 loss and temporal GIoU loss over matched intervals, respectively. The coefficients λ_{cls}, λ_{L1}, and λ_{GIoU} control the contributions of the three loss terms.

The same matching-and-supervision scheme is applied independently to the initial seed proposals and intermediate decoder outputs. At the seed stage, the foreground scores produced by g_{seed} are supervised using the same sigmoid focal loss formulation, with matched seed proposals labeled as foreground and unmatched ones as background; matched seed intervals are additionally supervised by the boundary losses. All components, including the diffusion denoiser, are jointly optimized end-to-end.

Inference. At inference, only predictions from the final decoder layer are retained. The normalized boundaries $(\hat{s}_i, \hat{e}_i)$ are mapped back to the original video timeline together with their forgery confidence $\hat{p}_i$. Predictions with $\hat{p}_i < \tau$, where τ denotes the confidence threshold, are discarded. The remaining intervals are returned directly without NMS.

Table 1: Comparison with existing methods on LAV-DF and AV-Deepfake1M (%). * denotes baselines reproduced with AV-HuBERT; others use reported results. **Bold**: best; underline: second best.

Dataset	Method	mAP	AP@0.50	AP@0.75	AP@0.95	AR@5	AR@10	AR@20
LAV-DF	BA-TFD*	31.94	63.67	26.77	0.09	50.05	53.00	55.32
	BA-TFD+*	54.98	76.85	63.42	1.82	69.80	71.01	72.23
	UMMAFormer*	63.05	64.75	64.18	55.66	91.15	**94.77**	**96.67**
	DiMoDif	–	95.50	87.90	20.60	–	91.40	92.70
	RegQAV	69.90	94.10	88.10	27.60	–	91.70	91.80
	AuViRe*	87.24	**97.72**	94.06	45.75	90.53	92.10	93.01
	DiST-TFL (Ours)	**93.13**	96.88	**94.44**	**81.59**	**93.01**	94.64	95.29
AV-Deepfake1M	BA-TFD*	8.75	27.11	3.20	0.01	30.52	34.69	38.95
	BA-TFD+*	26.10	56.94	19.14	0.12	46.79	49.03	51.68
	UMMAFormer*	62.98	68.49	67.01	36.48	85.49	89.56	**91.92**
	DiMoDif	–	86.93	75.95	5.43	76.64	78.84	80.25
	RegQAV	56.66	90.24	81.86	12.57	85.97	86.95	87.27
	AuViRe*	73.97	95.48	86.62	9.53	81.96	83.51	84.36
	DiST-TFL (Ours)	**84.17**	**96.11**	**91.18**	**37.06**	**88.94**	**90.05**	90.98

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Dataset. We evaluate our method on LAV-DF (Cai et al., 2023) and AV-Deepfake1M (Cai et al., 2024). LAV-DF contains 78,703 training videos and 26,100 test videos, with annotations for forged temporal boundaries and modification types. AV-Deepfake1M (AV-DF1M) is a larger-scale audio-visual deepfake benchmark containing over one million videos from more than 2K subjects, covering video-only, audio-only, and audio-visual manipulations.

Evaluation Metrics. We report AP at temporal IoU thresholds of 0.50, 0.75, and 0.95, together with mAP averaged from 0.50 to 0.95 at intervals of 0.05. Recall is measured by AR@K, where at most K predictions are retained per video. We use AR@5, AR@10, and AR@20 for the main comparison, and AR@1–3 for evaluation under instance-matched prediction budgets.

Implementation Details. We use AV-HuBERT (Shi et al., 2022) features as the audio-visual input, with 1024-dimensional visual and audio representations concatenated into a 2048-dimensional feature. The temporal backbone produces a six-level feature hierarchy with a common feature dimension of 256. For diffusion-based refinement, we use a three-step schedule indexed by $t \in \{0, 1, 2\}$, with a linear noise schedule from 10^{-4} to 0.02. Both training and inference start from the fixed timestep $t = 2$ and use deterministic DDIM refinement with $\eta = 0$. The sparse localization module uses $N_q = 30$ learnable queries. We train the model for 30 epochs with a batch size of 16 using AdamW with a weight decay of 0.05. At inference, we use a confidence threshold of $\tau = 0.003$ and do not apply NMS. We use dataset-specific optimization and decoder configurations for LAV-DF and AV-Deepfake1M. The complete hyperparameter settings are provided in the Appendix.

4.2 COMPARISON WITH STATE-OF-THE-ART METHODS

We compare DiST-TFL with existing temporal forgery localization methods on LAV-DF and AV-Deepfake1M, as summarized in Table 1. For BA-TFD, BA-TFD+, UMMAFormer (UMMA), and AuViRe, we replace their original input features with the same AV-HuBERT representations used by DiST-TFL while otherwise retaining their published model and hyperparameter settings. For RegQAV and DiMoDif, we report the results from their original papers because their pretrained representations are integral to their formulations. Accordingly, Table 1 provides a transparent comparison based on reproduced and originally reported results rather than a strictly feature-controlled evaluation. DiST-TFL achieves the highest mAP on both datasets, reaching 93.13 on LAV-DF and 84.17 on AV-Deepfake1M, improving over the second-best AuViRe by 5.89 and 10.20 points, respec-

Table 2: Component ablation of DiST-TFL on LAV-DF and AV-Deepfake1M (%) under a unified configuration. **Bold**: best; underline: second best.

Variant	Diff.	FPN	Query	mAP	AP@0.50	AP@0.75	AP@0.95	AR@5	AR@10
LAV-DF									
w/o Diff. & Query	–	✓	–	87.090	88.221	87.619	83.140	**97.346**	**98.408**
w/o Query	✓	✓	–	88.972	90.270	89.477	**84.797**	96.845	97.946
w/o Diffusion	–	✓	✓	76.418	90.890	88.011	14.866	86.640	90.442
w/o FPN Fusion	✓	–	✓	67.808	87.655	79.807	5.985	80.210	84.038
DiST-TFL	✓	✓	✓	**93.130**	**96.879**	**94.438**	81.589	93.009	94.643
AV-Deepfake1M									
w/o Diff. & Query	–	✓	–	69.193	76.246	73.707	39.310	85.512	89.462
w/o Query	✓	✓	–	70.262	76.897	74.661	41.144	86.810	89.897
w/o Diffusion	–	✓	✓	83.165	93.159	89.135	**42.097**	88.760	**90.671**
w/o FPN Fusion	✓	–	✓	75.349	87.609	82.399	28.342	86.107	88.471
DiST-TFL	✓	✓	✓	**84.170**	**96.111**	**91.182**	37.059	**88.938**	90.047

Table 3: Efficiency comparison on LAV-DF and AV-Deepfake1M (AV-DF1M).

Dataset	Method	Param. (M)	FLOPs (G)	Mem. (GB)	Fwd. (ms)	NMS (ms)	Total (ms)
LAV-DF	UMMA	21.7	27.1	0.159	73.1	6.9 / 3474.2	78.2 / 3547.3
	Late Diff.	31.3	32.5	0.198	125.3	–	125.4
	DiST-TFL	64.4	85.3	0.348	89.0	–	89.0
AV-DF1M	UMMA	21.7	27.1	0.159	73.6	10.4 / 6536.7	83.7 / 6610.4
	Late Diff.	29.0	31.5	0.188	110.8	–	110.9
	DiST-TFL	62.0	84.3	0.339	71.7	–	71.8

tively. On LAV-DF, DiST-TFL reaches 81.59 AP@0.95, substantially higher than the second-best result of 55.66, while on AV-Deepfake1M it ranks first across all reported AP thresholds. DiST-TFL also attains the best AR@5 on both datasets and remains competitive as K increases in AR@K. Component ablations under a unified training and evaluation configuration are provided in Table 2.

These results align with the two challenges motivating DiST-TFL: preserving manipulation-sensitive responses across multi-scale temporal representations and reducing redundant dense prediction for typically sparse forged intervals. On LAV-DF, DiST-TFL achieves substantially higher overall mAP while maintaining competitive recall, with UMMAFormer showing only slightly higher recall at larger prediction budgets. On AV-Deepfake1M, DiST-TFL likewise improves mAP while maintaining competitive recall with a compact prediction set. Together, these results indicate that broader candidate coverage does not necessarily translate into better localization quality.

4.3 Ablation Studies

Table 2 examines the main components of DiST-TFL under a unified configuration. The results indicate that their effects are interdependent rather than independently additive. Diffusion refinement and FPN fusion provide complementary benefits in representation modeling: the former improves localization by introducing manipulation-sensitive responses before temporal aggregation, while the latter helps maintain manipulation-sensitive information across temporal resolutions through cross-scale feature fusion. This coupling is reflected by the substantial degradation when FPN fusion is removed on both datasets and by the large drop after removing diffusion from the full model on LAV-DF. Additional results in Appendix Table 10 further support this design choice, showing that applying refinement before hierarchical modeling is more effective than applying it after multi-scale features have already been formed. Sparse query prediction then operates on the resulting representations, producing a compact set of intervals and improving mAP over the variant without queries on both datasets. Its lower AP@0.95, however, indicates that the overall gain does not extend to the strictest boundary threshold.

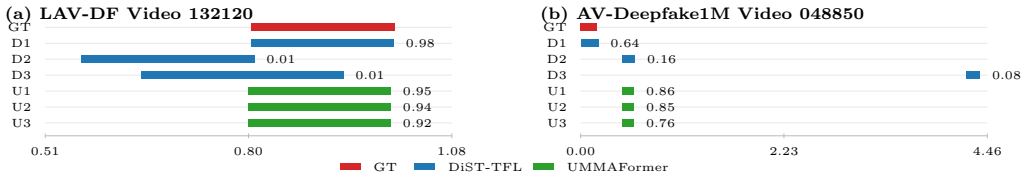


Figure 3: Qualitative comparison of DiST-TFL and UMMAFormer.

4.4 Analysis of Sparse Temporal Prediction

To further analyze sparse temporal prediction, we evaluate recall under a ground-truth-count-matched prediction budget. For a video containing K forged segments, only the top-K predictions are retained and AR@K is reported. As shown in Table 4, DiST-TFL achieves the best AR@1 on LAV-DF and remains close to UMMAFormer on AV-Deepfake1M. Its advantage becomes more evident when multiple forged intervals are present: DiST-TFL reaches 74.02 AR@2 on LAV-DF, outperforming AuViRe and UMMAFormer by 4.30 and 20.43 points, and achieves 84.57 AR@2 and 83.33 AR@3 on AV-Deepfake1M, exceeding AuViRe by 8.29 and 10.11 points. These results indicate that DiST-TFL maintains strong coverage when the prediction budget matches the number of forged instances, supporting the use of a compact set-prediction formulation rather than relying on dense, overlapping candidates for recall.

Table 4: Recall under a matched prediction budget (the top-K predictions for K forged segments).

Dataset	K	UMMA	AuViRe	DiST-TFL
LAV-DF	1	98.85	92.79	**99.07**
	2	53.59	69.72	**74.02**
AV-DF1M	1	**88.99**	78.58	87.18
	2	63.03	76.28	**84.57**
	3	59.13	73.22	**83.33**

4.5 Efficiency and Qualitative Analysis

Table 3 compares inference efficiency on both datasets with identical AV-HuBERT inputs. For UMMAFormer, we report NMS and total latency using compiled C++ and Python Soft-NMS implementations. DiST-TFL introduces higher model-side complexity through diffusion refinement. Despite the increased computational cost, NMS-free sparse prediction maintains low end-to-end latency on both datasets. Notably, late diffusion uses fewer parameters and FLOPs than DiST-TFL, yet incurs higher latency on both datasets, indicating a gap between theoretical complexity and practical latency. Compared with UMMAFormer, DiST-TFL is faster than Python Soft-NMS and remains competitive with compiled C++ Soft-NMS, being faster on AV-DF1M but slower on LAV-DF. Overall, DiST-TFL combines NMS-free inference with competitive end-to-end latency, while early diffusion is faster than late diffusion under our implementation.

Fig. 3 shows that DiST-TFL produces compact and well-aligned predictions for sparse forged intervals, whereas the dense baseline tends to generate redundant or shifted candidates. This observation is consistent with our motivation of preserving manipulation-sensitive responses and avoiding redundant dense prediction in TFL. More qualitative visualizations are provided in the Appendix.

5 Conclusion

In this work, we revisit Temporal Forgery Localization from two remaining challenges: preserving manipulation-sensitive responses across multi-scale temporal representations and reducing redundant dense prediction for typically sparse forged intervals. DiST-TFL addresses these challenges by combining diffusion-guided hierarchical feature modeling with sparse query-based localization. Experiments on LAV-DF and AV-Deepfake1M demonstrate consistent improvements, with ablation studies confirming the complementary roles of the two components. Future work could explore stronger forgery-aware representations, more efficient sparse localization with improved boundary precision under strict overlap criteria such as AP@0.95, and greater robustness to unseen manipulation methods and real-world distribution shifts.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

We used generative AI tools to assist with code implementation and editing, data cleaning and reformatting, and translation. We also used these tools to improve the wording and readability of parts of the manuscript. The research questions, methodology, experimental design, and interpretation of results were developed by the authors. The authors reviewed the AI-assisted code, data processing, translations, and text, verified the reported results against the experimental outputs, and take responsibility for the final submission.

REPRODUCIBILITY STATEMENT

We make several efforts to support reproducibility. The implementation code is included in the supplementary materials. The main paper describes the proposed architecture, learning objectives, evaluation protocol, and experimental setup, and the Appendix provides additional implementation details, hyperparameter settings, and ablation configurations. For the datasets used in our experiments, we further document the complete preprocessing and AV-HuBERT feature extraction pipeline for LAV-DF and AV-Deepfake1M, including audio/video preprocessing and the construction of the audio-visual input representations. These materials collectively provide the necessary information for reproducing our experimental results.

REFERENCES

Peijun Bao, Anwei Luo, Gang Pan, Alex C Kot, and Xudong Jiang. Activityforensics: A comprehensive benchmark for localizing manipulated activity in videos. *arXiv preprint arXiv:2604.03819*, 2026.

Zhixi Cai, Kalin Stefanov, Abhinav Dhall, and Munawar Hayat. Do you really mean that? content driven audio-visual deepfake dataset and multimodal method for temporal forgery localization. In *2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–10. IEEE, 2022.

Zhixi Cai, Shreya Ghosh, Abhinav Dhall, Tom Gedeon, Kalin Stefanov, and Munawar Hayat. Glitch in the matrix: A large scale benchmark for content driven audio–visual forgery detection and localization. *Computer Vision and Image Understanding*, 236:103818, 2023.

Zhixi Cai, Shreya Ghosh, Aman Pankaj Adatia, Munawar Hayat, Abhinav Dhall, Tom Gedeon, and Kalin Stefanov. Av-deepfake1m: A large-scale llm-driven audio-visual deepfake dataset. In *Proceedings of the 32nd ACM international conference on multimedia*, pp. 7414–7423, 2024.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 19773–19786. IEEE, 2023.

Komal Chugh, Parul Gupta, Abhinav Dhall, and Ramanathan Subramanian. Not made for each other-audio-visual dissonance-based deepfake detection and localization. In *Proceedings of the 28th ACM international conference on multimedia*, pp. 439–447, 2020.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Ho-Joong Kim, Jung-Ho Hong, Heejo Kong, and Seong-Whan Lee. Te-tad: Towards full end-to-end temporal action detection via time-aligned coordinate expression. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18837–18846. IEEE, 2024.

Christos Koutlis and Symeon Papadopoulos. Dimodif: Discourse modality-information differentiation for audio-visual deepfake detection and localization. *arXiv preprint arXiv:2411.10193*, 2024.

Christos Koutlis and Symeon Papadopoulos. Auvire: Audio-visual speech representation reconstruction for deepfake temporal localization. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 7168–7177. IEEE, 2026.

Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

Xiaolong Liu, Qimeng Wang, Yao Hu, Xu Tang, Shiwei Zhang, Song Bai, and Xiang Bai. End-to-end temporal action detection with transformer. *IEEE Transactions on Image Processing*, 31: 5427–5441, 2022.

Hui Miao, Yuanfang Guo, Zeming Liu, and Yunhong Wang. Multi-modal deepfake detection via multi-task audio-visual prompt learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 612–621, 2025.

Sauradip Nag, Xiatian Zhu, Jiankang Deng, Yi-Zhe Song, and Tao Xiang. Difftad: Temporal action detection with proposal denoising diffusion. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10328–10340. IEEE, 2023.

Trevine Oorloff, Surya Koppisetti, Nicolò Bonettini, Divyaraj Solanki, Ben Colman, Yaser Yacoob, Ali Shahriyari, and Gaurav Bharaj. Avff: Audio-visual feature fusion for video deepfake detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27092–27102. IEEE, 2024.

Bowen Shi, Wei-Ning Hsu, Kushal Lakhotia, and Abdelrahman Mohamed. Learning audio-visual speech representation by masked multimodal cluster prediction. *arXiv preprint arXiv:2201.02184*, 2022.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Julian Wyatt, Adam Leach, Sebastian M Schmon, and Chris G Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 649–655. IEEE, 2022.

Wenyuan Yang, Xiaoyu Zhou, Zhikai Chen, Bofei Guo, Zhongjie Ba, Zhihua Xia, Xiaochun Cao, and Kui Ren. Avoid-df: Audio-visual joint learning for detecting deepfake. *IEEE Transactions on Information Forensics and Security*, 18:2015–2029, 2023.

Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *European conference on computer vision*, pp. 492–510. Springer, 2022.

Rui Zhang, Hongxia Wang, Mingshan Du, Hanqing Liu, Yang Zhou, and Qiang Zeng. Ummaformer: A universal multimodal-adaptive transformer framework for temporal forgery localization. In *Proceedings of the 31st ACM international conference on multimedia*, pp. 8749–8759, 2023.

Xiaodong Zhu, Suting Wang, Junqi Yang, Yuhong Yang, Weiping Tu, and Zhongyuan Wang. Query-based audio-visual temporal forgery localization with register-enhanced representation learning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 8547–8556, 2025.

Heqing Zou, Meng Shen, Yuchen Hu, Chen Chen, Eng Siong Chng, and Deepu Rajan. Cross-modality and within-modality regularization for audio-visual deepfake detection. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4900–4904. IEEE, 2024.

6 Appendix

6.1 Audio-Visual Feature Extraction

Pre-trained Backbone. We extract audio-visual representations using the large AV-HuBERT model with the publicly released `self_large_vox_433h.pt` checkpoint. The model is pre-trained in a self-supervised manner on LRS3 and VoxCeleb2 and is loaded through the Fairseq

toolkit. We use the output of the final encoder layer, yielding a 1024-dimensional representation at each temporal position.

Video Preprocessing. For each video, faces are detected frame by frame using Dlib. The facial region is cropped, resized to 88×88, and converted to grayscale. Pixel values are first scaled to $[0, 1]$ and then standardized using a mean of 0.421 and a standard deviation of 0.165, following the preprocessing configuration used for AV-HuBERT feature extraction.

Audio Preprocessing. The audio track is extracted using FFmpeg as a single-channel 16-kHz waveform. Log filter-bank features are computed using a 25-ms analysis window and a 10-ms frame shift with 26 filters, producing approximately 100 audio frames per second. Every four consecutive audio frames are stacked to match the 25-fps video stream. Zero padding is applied when the number of audio frames is not divisible by four, and the resulting audio sequence is further aligned with the video sequence by padding or truncation when necessary. Audio features are layer-normalized before being passed to the AV-HuBERT encoder.

Feature Views and Model Input. For each video, we extract three feature views by controlling the input modalities to AV-HuBERT: a visual-only view obtained by masking the audio input, an audio-only view obtained by masking the visual input, and a multimodal view obtained from the complete audio-visual input. Each view consists of a sequence of 1024-dimensional representations and is stored in the same `.npz` file. DiST-TFL uses the visual-only and audio-only representations during training and inference. Their temporal sequences are truncated to a common length and concatenated along the feature dimension, producing a 2048-dimensional audio-visual representation at each temporal position. The multimodal view is retained during feature extraction but is not used as input to DiST-TFL.

Dataset Processing. The same feature extraction pipeline is applied to LAV-DF and AV-Deepfake1M without dataset-specific changes to the preprocessing procedure. We follow the official train, validation, and test partitions.

6.2 Additional Implementation Details

All experiments are conducted on a single NVIDIA GeForce RTX 4080 GPU. The six pyramid levels use temporal strides $\{1, 2, 4, 8, 16, 32\}$, and the maximum input sequence length is set to 768. Each decoder layer uses four attention heads with a 1024-dimensional feed-forward layer and a dropout rate of 0.1. We adopt a cosine learning-rate schedule with a five-epoch warm-up, a drop-path rate of 0.1, and model exponential moving average.

For one-to-one bipartite matching, the matching weights are shared across datasets and set to $\lambda^m_{\text{cls}} = 2$, $\lambda^m_{\text{L1}} = 5$, and $\lambda^m_{\text{GIoU}} = 2$. For the training objective, the classification and boundary loss weights are set to $\lambda_{\text{cls}} = 2$ and $\lambda_{\text{L1}} = 5$, respectively, while the GIoU loss weight λ_{GIoU} is dataset-specific.

We use dataset-specific optimization and decoder configurations for LAV-DF and AV-Deepfake1M, as summarized in Table 5. Specifically, the two datasets use different learning rates, numbers of decoder layers, proposal ranges, and GIoU loss weights.

Table 5: Dataset-specific hyperparameter settings.

Hyperparameter	**LAV-DF**	**AV-Deepfake1M**
Learning rate	1×10^{-3}	5×10^{-4}
Decoder layers N_d	4	2
Proposal range γ	1.0	0.5
GIoU loss weight λ_{GIoU}	2.0	1.0

6.3 Additional Ablation and Sensitivity Analysis

We provide additional experiments to examine the sensitivity of DiST-TFL to optimization and diffusion settings. Table 6 evaluates the w/o Query variant under two learning rates on AV-Deepfake1M.

Table 6: Learning-rate sensitivity of the w/o Query variant on AV-Deepfake1M. Results are in %.

Learning Rate	mAP	AP@0.50	AP@0.75	AP@0.95	AR@5	AR@10
5×10^{-4}	67.307	74.514	72.156	36.081	**86.883**	89.770
1×10^{-3}	**70.262**	**76.897**	**74.661**	**41.144**	86.810	**89.897**

Table 7: Sensitivity to the number of diffusion steps on AV-Deepfake1M. Results are in %.

Steps	mAP	AP@0.50	AP@0.75	AP@0.95	AR@5	AR@10
2	**84.387**	96.483	**91.857**	36.277	**89.073**	**90.268**
3	84.170	96.111	91.182	37.059	88.938	90.047
4	84.310	**96.701**	91.691	**37.321**	84.467	89.686
5	**84.387**	96.627	91.465	36.308	87.845	89.916

Increasing the learning rate from 5×10^{-4} to 1×10^{-3} improves mAP and all reported AP metrics, while recall remains similar. This confirms that the comparison used in the main ablation is not disadvantaged by an under-tuned learning rate.

Table 7 examines the sensitivity of DiST-TFL to the number of diffusion refinement steps. The results remain stable across two to five steps, with mAP varying by less than 0.25 points. This indicates that the method is not highly sensitive to this hyperparameter. We use three steps in the main experiments as a fixed representative setting, without separately tuning the number of refinement steps for each dataset.

Table 10 further studies where diffusion refinement is introduced in the temporal modeling pipeline. Applying diffusion before hierarchical feature modeling consistently outperforms applying it after the multi-scale hierarchy has been formed. On AV-Deepfake1M, early diffusion improves mAP from 83.301 to 84.170 and also yields higher AP@0.95 and AR@5. The difference is substantially larger on LAV-DF, where early diffusion increases mAP from 22.284 to 93.130. These results show that diffusion placement is important to the effectiveness of the refinement process and support our design of performing diffusion-guided interaction before hierarchical temporal modeling.

6.4 Additional Efficiency Analysis

Tables 8 and 9 compare the computational cost and inference latency of DiST-TFL and its variants with UMMAFormer on AV-Deepfake1M and LAV-DF. DiST-TFL introduces higher model-side complexity due to diffusion refinement, resulting in more parameters, FLOPs, and memory consumption than UMMAFormer and the variants without early diffusion. The reported total latency is measured independently over the complete inference pipeline and is therefore not necessarily equal to the sum of the separately measured forward and NMS latency.

Despite its higher theoretical complexity, DiST-TFL does not incur proportionally higher end-to-end latency. On AV-Deepfake1M, DiST-TFL requires 71.76 ms per sample, while the late-diffusion variant requires 110.87 ms despite using fewer parameters and FLOPs. A similar trend is observed on LAV-DF, where DiST-TFL takes 89.04 ms compared with 125.35 ms for late diffusion. Under our implementation, these results show that applying diffusion before hierarchical temporal modeling yields lower latency than independently refining the constructed multi-scale features.

Compared with UMMAFormer, DiST-TFL avoids the additional NMS stage through direct sparse set prediction. Using the compiled C++ Soft-NMS implementation, UMMAFormer requires 83.66 ms on AV-Deepfake1M and 78.20 ms on LAV-DF, whereas DiST-TFL requires 71.76 ms and 89.04 ms, respectively. DiST-TFL is therefore faster on AV-Deepfake1M and slightly slower on LAV-DF despite its higher model-side computation. The Python Soft-NMS implementation incurs substantially greater post-processing latency, illustrating the implementation-dependent overhead of dense localization pipelines. Overall, DiST-TFL trades additional representation-side computation for NMS-free inference while maintaining competitive end-to-end latency.

Table 8: Efficiency comparison on AV-Deepfake1M (AV-DF1M). For UMMAFormer (UMMA), NMS and total inference times are reported as C++ / Python.

Method	Param. (M)	FLOPs (G)	Mem. (GB)	Fwd. (ms)	NMS (ms)	Total (ms)
UMMA	21.689	27.10	0.159	73.62	10.43 / 6536.67	83.66 / 6610.39
Ours w/o Diff	24.20	26.13	0.169	66.01	–	66.08
Ours (Late Diff.)	28.96	31.53	0.188	110.78	–	110.87
Ours	61.98	84.29	0.339	71.69	–	71.76

Table 9: Efficiency comparison on LAV-DF. For UMMAFormer (UMMA), NMS and total inference times are reported as C++ / Python.

Method	Param. (M)	FLOPs (G)	Mem. (GB)	Fwd. (ms)	NMS (ms)	Total (ms)
UMMA	21.689	27.10	0.159	73.06	6.92 / 3474.15	78.20 / 3547.31
Ours w/o Diff	26.57	27.13	0.179	70.82	–	70.88
Ours (Late Diff.)	31.33	32.54	0.198	125.27	–	125.35
Ours	64.35	85.29	0.348	88.96	–	89.04

6.5 Diffusion Refinement Details

The diffusion refinement module uses a lightweight one-dimensional residual denoiser D_θ to predict a task-oriented perturbation. Given an input feature of dimension C, the denoiser consists of a 1×1 input projection, a depthwise temporal convolution with kernel size 3, a pointwise 1×1 convolution, Group Normalization, and a final 1×1 output projection. Timestep information is encoded using sinusoidal embeddings followed by a two-layer MLP and injected into the denoiser through adaptive scale-and-shift modulation of the normalized features. Following the noise-prediction parameterization, the denoiser predicts a noise-like perturbation rather than the clean feature directly.

We use a three-step schedule indexed by $t \in \{0, 1, 2\}$, with β_t linearly ranging from 10^{-4} to 0.02. During both training and inference, the noisy representation is initialized at the fixed timestep $t = 2$, and deterministic DDIM refinement is performed over the reverse sequence $t = 2, 1, 0$ with $\eta = 0$. At each step, the clean representation is estimated from the current noisy feature and the predicted perturbation, followed, when applicable, by a deterministic DDIM update to the preceding timestep. The final clean estimate is clipped to $[-3, 3]$ for numerical stability.

No separate diffusion reconstruction loss is introduced. Instead, the refined representation is passed to the downstream temporal localization network, and gradients from the localization objective are back-propagated through the complete refinement process, including the denoiser and timestep-conditioning modules.

6.6 Qualitative Visualization and Analysis

Figures 4 and 5 present qualitative results of DiST-TFL on the LAV-DF and AV-Deepfake1M test sets, respectively. The examples cover forged intervals with different durations and temporal locations. On both datasets, the highest-confidence prediction generally aligns closely with the ground-truth forged interval, while the remaining predictions tend to receive substantially lower confidence scores. This behavior is observed for both short manipulated segments and forged intervals appearing at different positions in the video timeline, illustrating the boundary quality and compact prediction behavior of DiST-TFL at the individual-video level.

Figures 7 and 6 provide the corresponding UMMAFormer results under the same AV-HuBERT input representation. While UMMAFormer can also localize forged regions accurately, its high-ranked predictions often form multiple overlapping hypotheses around the same temporal region. We further compare the two methods on identical test videos in Figures 8 and 9. This paired comparison makes the difference more explicit. DiST-TFL typically produces a dominant prediction that is closely aligned with the ground truth, whereas UMMAFormer often exhibits larger boundary deviations at the sample level. Moreover, its redundant predictions tend to share similar boundary offsets, forming clusters of highly overlapping hypotheses that are shifted in nearly the same di-

Table 10: Effect of diffusion placement on AV-Deepfake1M and LAV-DF. Results are in %.

Dataset	Placement	mAP	AP@0.95	AR@5
AV-Deepfake1M	Late	83.301	35.980	88.224
	Early (DiST-TFL)	**84.170**	**37.059**	**88.938**
LAV-DF	Late	22.284	3.178	81.599
	Early (DiST-TFL)	**93.130**	**81.589**	**93.009**

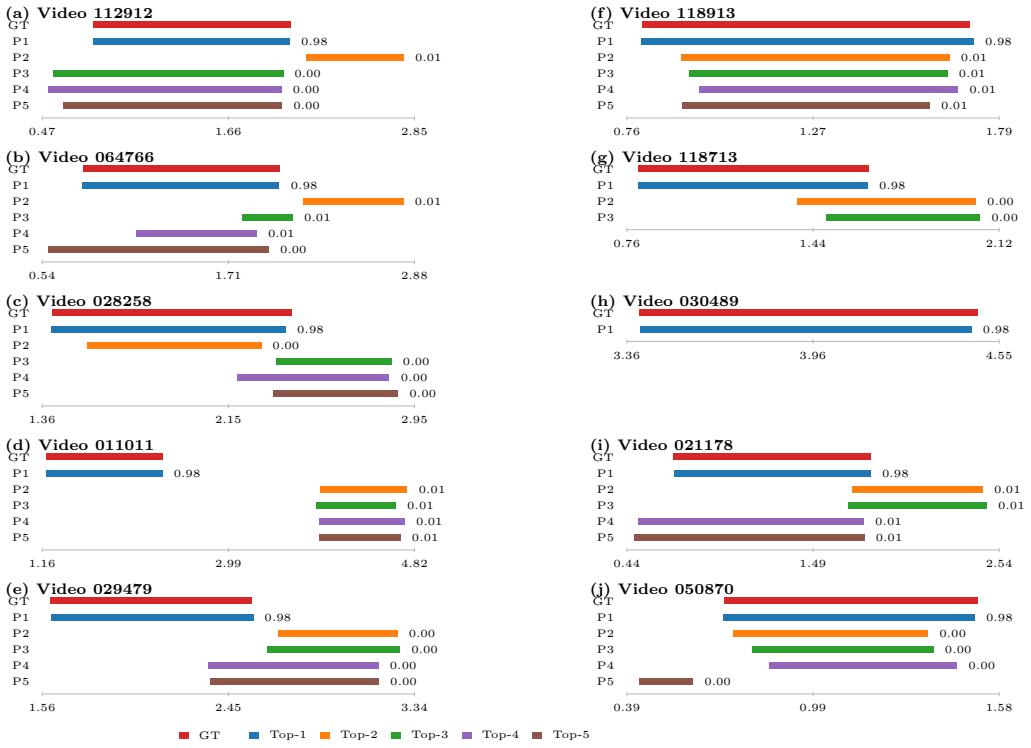


Figure 4: Qualitative temporal localization results on the LAV-DF test set. Ground-truth forged intervals are shown in red, and the top-ranked predictions are shown with their confidence scores.

rection rather than providing complementary coverage of the ground-truth interval. This effect is particularly evident on AV-Deepfake1M, where several shared examples show accurate DiST-TFL predictions despite consistently shifted UMMAFormer candidates. On LAV-DF, the gap is smaller, but DiST-TFL still tends to provide tighter boundary alignment with fewer redundant hypotheses. Together, these examples qualitatively support the distinction between dense candidate generation and the sparse set-prediction formulation adopted by DiST-TFL.

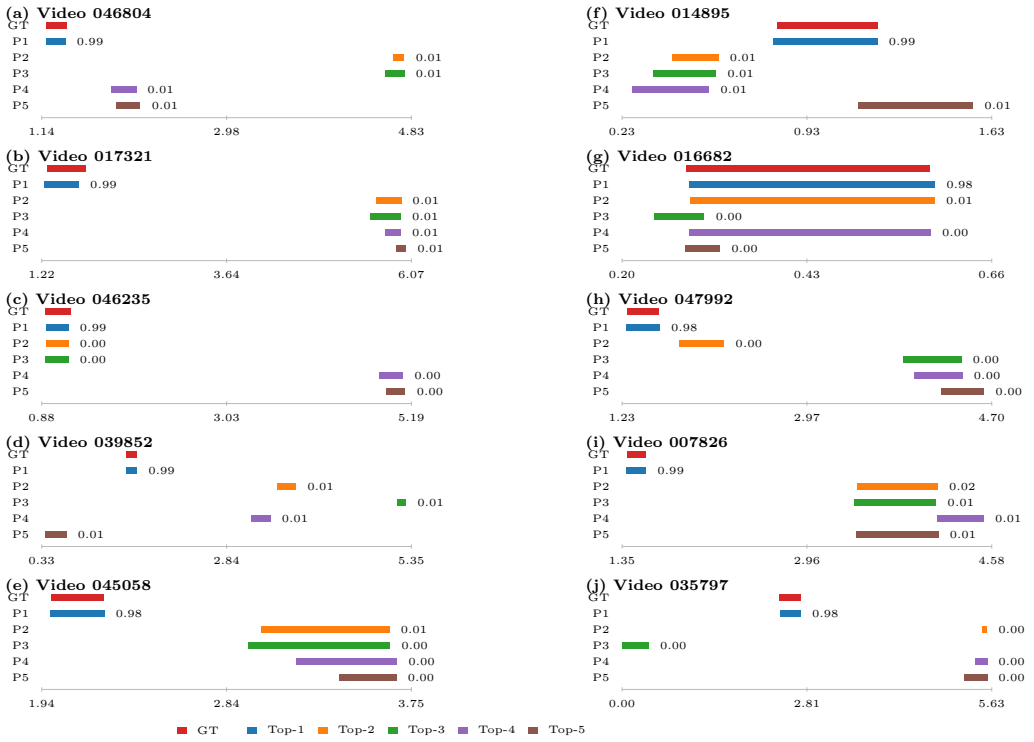


Figure 5: Qualitative temporal localization results on the AV-Deepfake1M test set. Ground-truth forged intervals are shown in red, and the top-ranked predictions are shown with their confidence scores.

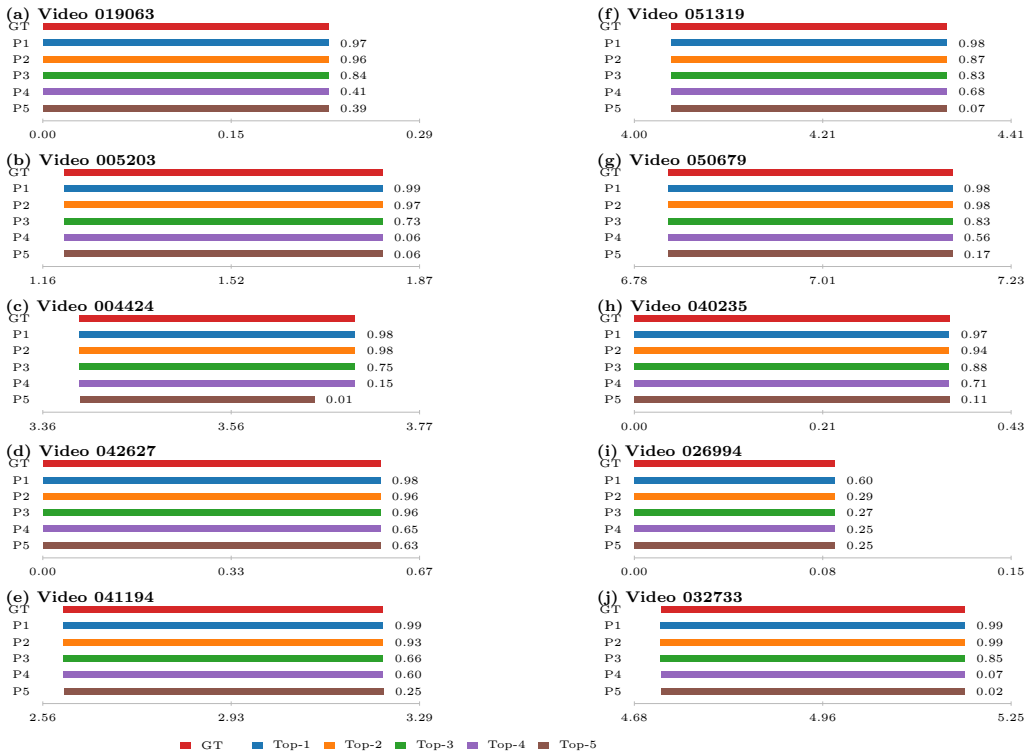


Figure 6: Qualitative temporal localization results of UMMAFormer on the AV-Deepfake1M test set. Ground-truth forged intervals are shown in red, and the top-ranked predictions are shown with their confidence scores.

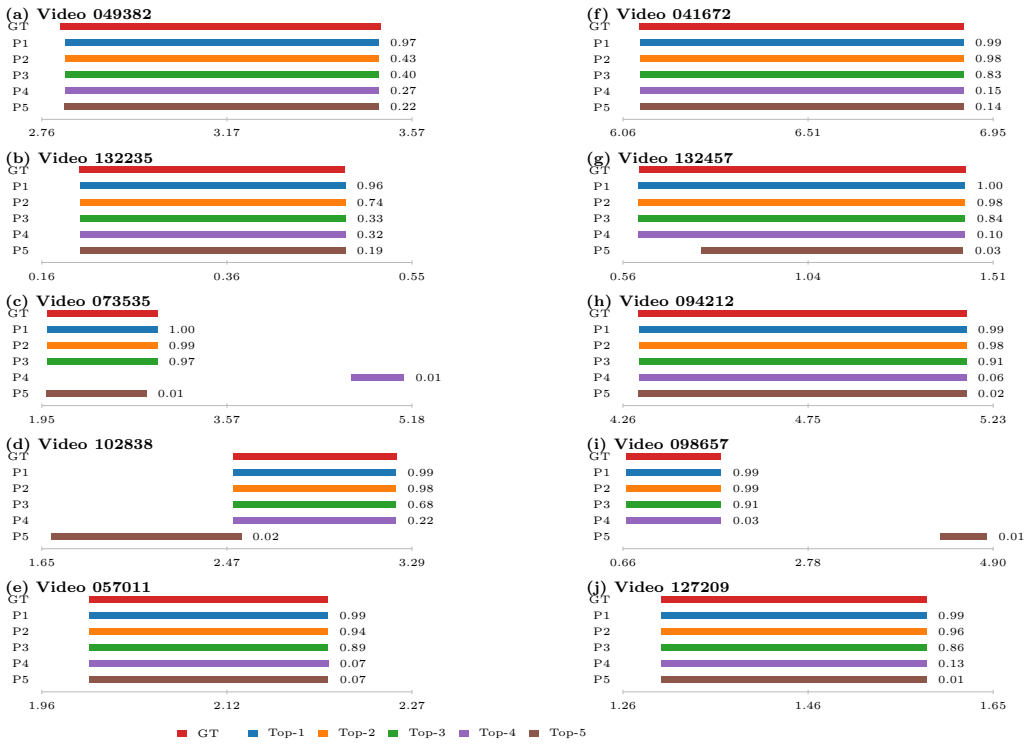


Figure 7: Qualitative temporal localization results of UMMAFormer on the LAV-DF test set. Ground-truth forged intervals are shown in red, and the top-ranked predictions are shown with their confidence scores.

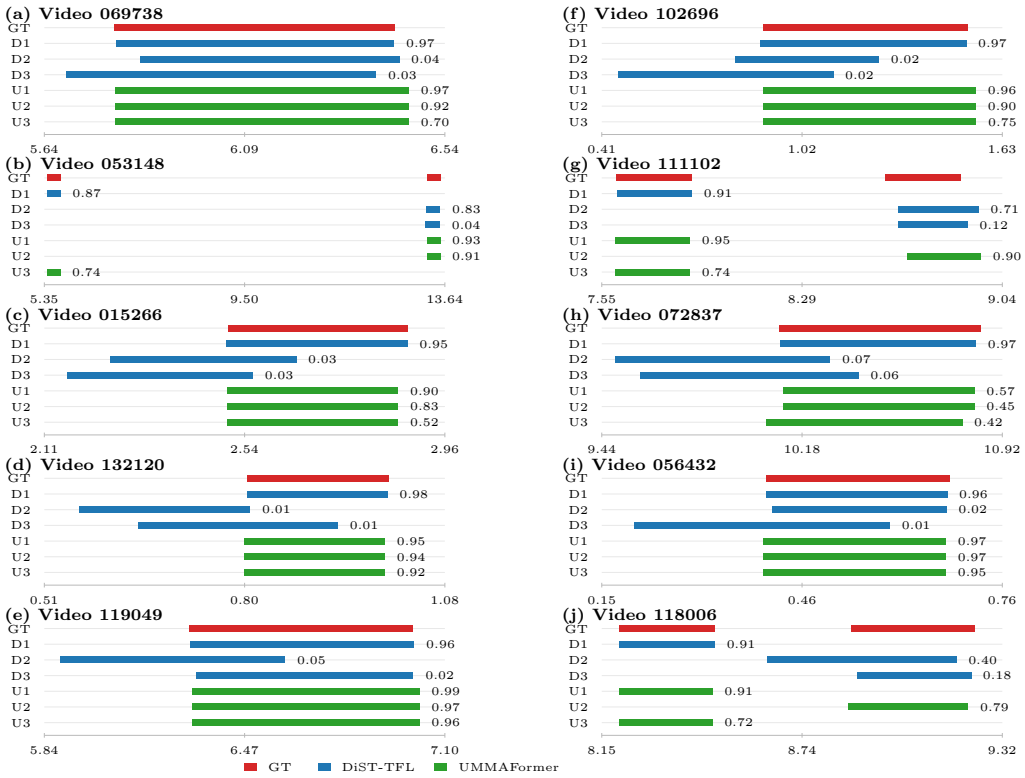


Figure 8: Qualitative comparison between DiST-TFL and UMMAFormer on selected LAV-DF test videos. Ground-truth forged intervals are shown in red.

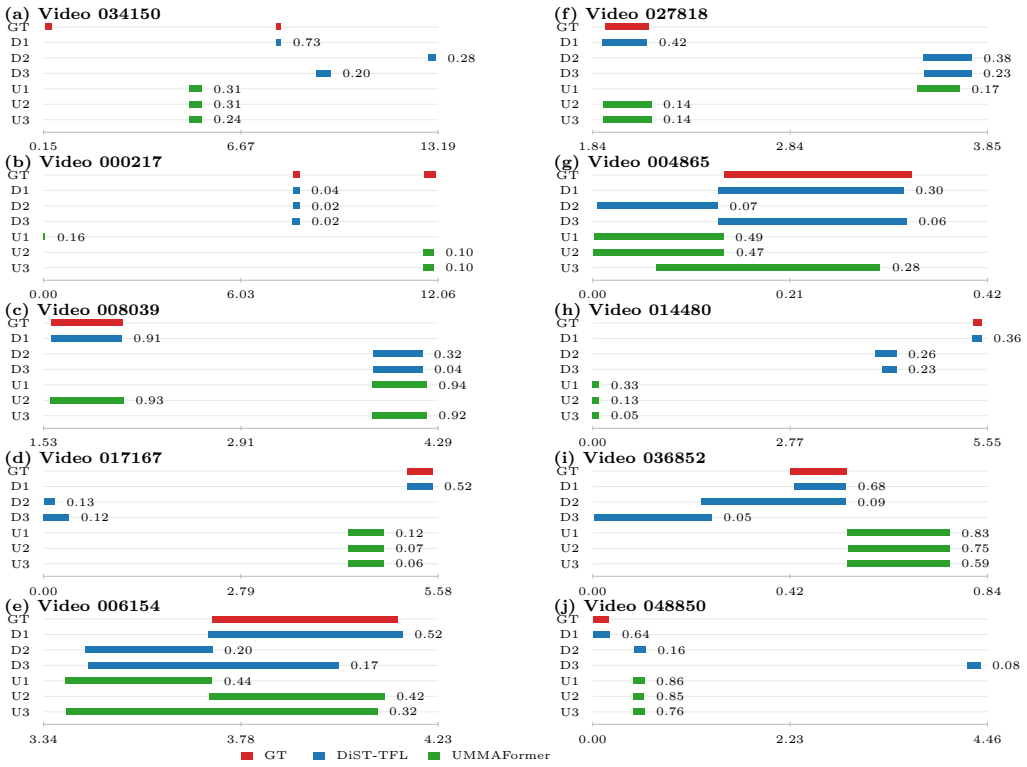


Figure 9: Qualitative comparison between DiST-TFL and UMMAFormer on selected AV-Deepfake1M test videos. Ground-truth forged intervals are shown in red.