TVTA: TRAJECTORY-AWARE VISEME-GUIDED TEMPORAL AGGREGATION FOR EVENT-BASED LIP READING

Anonymous authors
Paper under double-blind review

ABSTRACT

Event-based lip reading has recently emerged as a promising direction for visual speech recognition, benefiting from the high temporal resolution and motion sensitivity of event cameras. However, existing methods typically perform spatial compression before sufficient temporal modeling, which may suppress sparse and localized motion trajectories that are crucial for distinguishing similar lip movements. Moreover, most current approaches optimize temporal representations mainly at the word-classification level, leaving the underlying articulatory structure weakly constrained. To address these limitations, we propose a temporal aggregation framework that preserves location-wise trajectories in the encoder feature map prior to global spatial aggregation and introduces viseme-level sequence guidance. First, we introduce Trajectory-Aware Differential Aggregation (TDA), which performs temporal modeling at each encoded spatial location before adaptive global spatial aggregation. Second, we propose Viseme-Guided Aggregation (VGA), which introduces viseme-level sequence supervision to guide temporal aggregation for word recognition. Third, we employ EMA teacher-student consistency to regularize training and improve generalization. Extensive experiments on DVS-Lip and DVS-LRW100 show that the proposed method establishes new best results under the corresponding evaluation settings, while qualitative analyses further reveal that the learned temporal representations capture meaningful viseme-aware structure from event streams.

1 INTRODUCTION

Lip reading infers spoken content from visual articulations and complements automatic speech recognition in noisy or privacy-sensitive scenarios. For such visual-only perception, event cameras offer an attractive sensing modality. Compared with conventional frame-based cameras, they asynchronously record brightness changes with microsecond temporal resolution, high dynamic range, and sparse event streams, making them particularly suitable for capturing subtle and rapidly evolving lip motions. Nevertheless, effectively exploiting event streams for lip reading remains challenging because discriminative cues are encoded in subtle, localized, and continuously evolving articulatory patterns.

To address these challenges, existing event-based lip-reading methods have explored a variety of spatio-temporal modeling strategies. Early studies introduced multi-grained spatio-temporal representations to capture event dynamics across different temporal scales (Tan et al., 2022; 2024), establishing the basic paradigm of jointly modeling spatial appearance and temporal evolution from event lip streams. Subsequent works further improved this paradigm from complementary perspectives. Temporal granularity alignment was introduced to better coordinate short- and long-term motion patterns (Zhang et al., 2025a), while event-specific motion analysis sought to characterize articulatory changes through structured geometric representations (Ju et al., 2026). More recent studies adopted state-space models to strengthen long-range temporal dependencies (Zhang et al., 2025d; Ren et al., 2025), and explored frequency-aware feature enhancement to capture motion information beyond conventional spatial-domain processing (Zhang et al., 2025b;f; Ren et al., 2026). Collectively, these methods have substantially advanced the modeling of event-based visual speech by enriching temporal scales, motion structures, and feature interactions.

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.

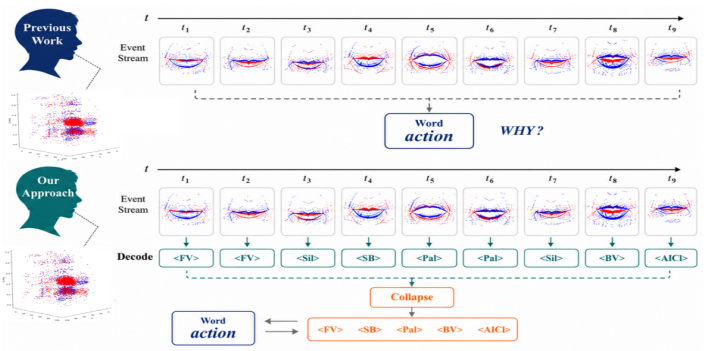


Figure 1: Comparison between previous word-level modeling and our viseme-supervised temporal decoding. Previous work directly aggregates the event stream into a word prediction, while our approach first decodes segment-level viseme labels and then collapses them into a compact viseme sequence that explains the final word recognition result.

Despite their architectural diversity, however, they generally follow a similar processing paradigm: event streams are first converted into voxelized or structured spatial representations, processed by spatial encoders, and then progressively aggregated before downstream temporal reasoning and word recognition. This shared formulation provides a convenient interface for conventional visual backbones, but it also means that fine-grained local event dynamics are often compressed before their temporal evolution is explicitly modeled. Moreover, temporal representations are typically optimized through the final word-level recognition objective, providing limited supervision for the underlying visual articulation progression.

These observations motivate strengthening temporal learning at two complementary levels. **At the local level**, temporal evolution should be modeled over the spatial lattice retained by the encoder before its final global aggregation, allowing each encoded location to maintain a separate temporal sequence up to that point. **At the sequence level**, temporal representations should be guided not only by the final word-level objective, but also by structured intermediate supervision that characterizes the progression of visual articulations. Recent advances in Connectionist Temporal Classification (CTC)-based sequence modeling (Graves et al., 2006), including consistency-regularized optimization (Yao et al., 2025) and intermediate CTC objectives (Altinok, 2025), further indicate the potential of richer sequence supervision for learning temporally structured representations.

Motivated by these insights, we propose TVTA, a temporal aggregation framework for event-based lip reading. For local temporal learning, we introduce Trajectory-Aware Differential Aggregation (TDA), which models temporal evolution at each encoded spatial location before the final global spatial aggregation. For sequence-level temporal learning, we propose Viseme-Guided Aggregation (VGA), which introduces viseme-aware CTC supervision and exploits articulation cues to guide temporal feature aggregation for word recognition, as illustrated in Fig. 1. Finally, we employ EMA teacher–student consistency to regularize training and improve generalization. Experiments on both DVS-Lip and DVS-LRW100 verify the effectiveness of the proposed framework. Ablation studies further validate the contributions of TDA, VGA, and teacher–student consistency. The main contributions of this paper are summarized as follows:

- We propose TVTA, a temporally enhanced framework for event-based lip reading that strengthens temporal learning at both the local trajectory level and the sequence supervision level.
- We introduce Trajectory-Aware Differential Aggregation (TDA) and Viseme-Guided Aggregation (VGA), which respectively model location-wise trajectories on the encoder feature map before global spatial aggregation and incorporate viseme-aware sequence supervision to improve temporal representation learning for word recognition.
- Extensive experiments on both DVS-Lip and DVS-LRW100 demonstrate consistent improvements over previous state-of-the-art methods. Comprehensive ablation studies and qualitative analyses further validate the effectiveness of each proposed component.

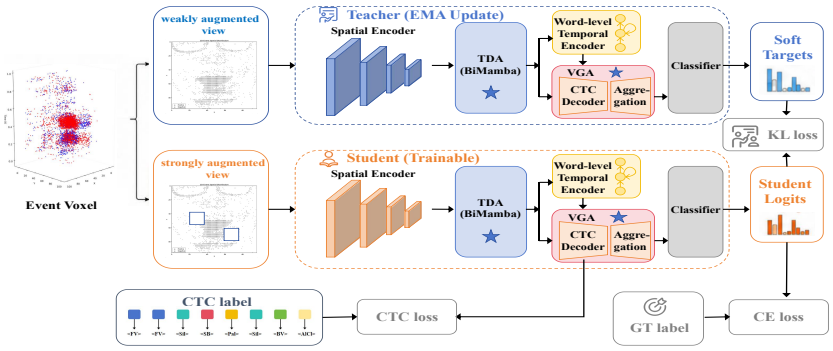


Figure 2: Overall framework of the proposed event-based lip-reading system. A weakly augmented view is processed by the EMA teacher, while a strongly augmented view is processed by the student. Both branches share the same architecture, consisting of a spatial encoder, TDA, a word-level temporal encoder, VGA, and a classifier. The teacher parameters are updated as an exponential moving average (EMA) of the student parameters. The student is jointly optimized by cross-entropy, CTC supervision, and KL consistency.

2 Related Work

2.1 Event-Based Lip Reading

Event cameras offer microsecond temporal resolution and high dynamic range, making them ideal for capturing fast and subtle lip motions. Existing event-based lip-reading methods typically convert asynchronous streams into structured representations followed by spatio-temporal feature learning. Early frameworks like MSTP (Tan et al., 2022) and its extension MSTP++ (Tan et al., 2024) demonstrated the value of multi-temporal and fine-grained representations, while MTGA (Zhang et al., 2025a) introduced graph structures to preserve local spatio-temporal details. Recent studies in event-based lip reading have further advanced the field by integrating state-space modeling for long-range dependencies (Zhang et al., 2025d) and enhancing high-frequency lip-region details (Zhang et al., 2025f;b). Despite these advances, most methods perform temporal modeling *after* spatial encoding or aggregation, inevitably compressing localized spatial responses and weakening the sparse, rapidly changing motion cues crucial for event-based lip reading.

2.2 Temporal Modeling and Sequence Supervision

While RNNs (Hochreiter & Schmidhuber, 1997; Cho et al., 2014) suffer from sequential bottlenecks and Transformers (Vaswani et al., 2017) incur quadratic complexity, State Space Models (SSMs) like Mamba (Gu & Dao, 2024) offer efficient, content-adaptive linear-time alternatives (Gu et al., 2022; Smith et al., 2023). Although SSMs have been applied to visual tasks (Zhu et al., 2024; Liu et al., 2024; Li et al., 2024) and event-based lip reading (Zhang et al., 2025d), they predominantly perform temporal modeling *after* spatial aggregation, leaving the fine-grained temporal evolution of localized responses *before* compression underexplored.

Recent visual speech recognition (VSR) studies increasingly favor intermediate sequence decoding over direct global classification. For example, explicit intermediate representations (Thomas et al., 2025) and consistency-regularized CTC modeling (Yao et al., 2025) have proven effective, especially in limited-data regimes (Gimeno-Gómez & Martínez-Hinarejos, 2024). However, existing event-based lip-reading methods still rely on global sequence aggregation for isolated-word classification (Tan et al., 2022; 2024; Zhang et al., 2025a;d;b), leaving the latent articulatory structure weakly constrained. Together, these two gaps motivate us to unify pre-aggregation temporal modeling via trajectory-aware differential aggregation and viseme-aware sequence supervision via viseme-guided temporal aggregation.

2.3 TEACHER–STUDENT CONSISTENCY LEARNING

Consistency regularization and self-distillation have emerged as powerful paradigms for improving model robustness. In self-supervised learning, methods like SimCLR (Chen et al., 2020), BYOL (Grill et al., 2020), and MoCo (He et al., 2020) align latent representations across augmented views to avoid representational collapse, whereas supervised approaches such as R-Drop (Liang et al., 2021) and self-distillation (Furlanello et al., 2018; Zhang et al., 2019; Zhang & Sabuncu, 2020; Yun et al., 2020; Shen et al., 2022) encourage prediction consistency across perturbed branches. Related ideas have also recently been explored in CTC-based speech recognition, where CR-CTC (Yao et al., 2025) enforces consistency between augmented views to regularize the CTC distribution itself.

However, existing methods rarely address the vulnerability of event-based lip-reading to strong spatio-temporal perturbations. We bridge this gap by repurposing the EMA teacher-student paradigm as a fully supervised regularizer to enforce perturbation-invariant representations under event-specific augmentations.

3 METHOD

3.1 OVERALL FRAMEWORK

Figure 2 illustrates the overall framework. We convert each asynchronous event stream into a fixed-length dual-polarity voxel representation $\mathbf{V} \in \mathbb{R}^{T\times C_{\text{in}}\times H\times W}$, where positive and negative events are accumulated in separate channels. During training, two views are generated from the same sample: a weakly augmented view for the EMA teacher and a strongly augmented view for the student.

Both branches share the same architecture. The voxel sequence is processed by an embedding layer and a DropPath-regularized ResNet encoder, followed by the proposed Trajectory-Aware Differential Aggregation module to produce frame-level features. These features are then fed into two parallel temporal branches: a word-level temporal encoder for semantic sequence modeling and the proposed Viseme-Guided Aggregation for viseme-aware temporal aggregation. VGA consists of a CTC decoder and a viseme-guided aggregation pathway, where the former extracts segment-level visual speech structure and the latter uses the resulting viseme segment context to score, reweight, and fuse word-level temporal features before classification.

During training, the student is optimized by word-level classification, viseme-level CTC supervision, and KL consistency against the EMA teacher. At inference time, since the teacher and student share the same model architecture and differ only in parameters, either branch can be used for performance evaluation.

3.2 TRAJECTORY-AWARE DIFFERENTIAL AGGREGATION

An embedding layer and four-stage ResNet-18 spatial encoder process the voxel input, with DropPath (Huang et al., 2016) applied to residual blocks. Its spatial convolutions mix local information; TDA therefore does not preserve raw, unmixed pixel trajectories. Rather, unlike a pipeline that immediately collapses the encoder's spatial lattice, TDA treats each location in the encoded feature map as a separate temporal trajectory before the final global spatial aggregation. In this way, temporal evolution is explicitly modeled for each encoded location before those responses are globally compressed.

As illustrated in Fig. 3, TDA contains spatial-to-temporal rearrangement, shared temporal encoding, and differential spatial aggregation. Given $\mathbf{X} \in \mathbb{R}^{B\times C\times T\times H\times W}$, we flatten its spatial lattice into $N = H \times W$ trajectories and encode each trajectory along time:

$$\tilde{\mathbf{X}}_s = \mathcal{R}(\mathbf{X}) + \mathbf{P}, \qquad \mathbf{G} = \Psi_{\text{BiMamba}}(\tilde{\mathbf{X}}_s), \qquad \tilde{\mathbf{X}}_s, \mathbf{G} \in \mathbb{R}^{B\times N\times T\times C}. \tag{1}$$

Here, $\mathcal{R}(\cdot)$ denotes rearrangement, $\mathbf{P} \in \mathbb{R}^{1\times N\times 1\times C}$ is a learnable spatial positional embedding shared over time, and $\Psi_{\text{BiMamba}}(\cdot)$ is a bidirectional state-space encoder shared across all trajectories. Thus, temporal information is injected at each location before cross-location aggregation, while the parameter cost remains independent of N.

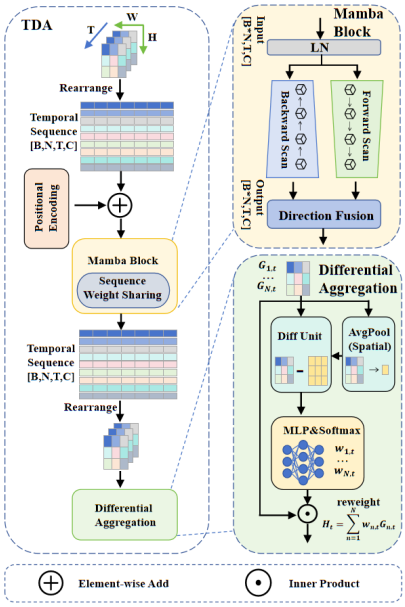


Figure 3: Overview of the proposed Trajectory-Aware Differential Aggregation module. Each spatial location is first converted into a temporal token sequence, processed by a shared bidirectional Mamba block, and then aggregated by differential spatial aggregation.

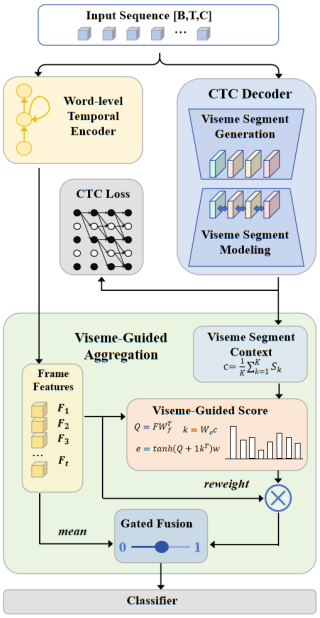


Figure 4: Overview of Viseme-Guided Aggregation. VGA consists of a CTC decoder and a viseme-guided aggregation pathway. The decoder provides viseme supervision and segment context, and the aggregation pathway uses this context to score, reweight, and fuse word-level temporal features.

To aggregate these temporally enriched responses, we construct a differential spatial weighting mechanism. We compute the global spatial prototype:

$$\bar{\mathbf{G}}_t = \frac{1}{N}\sum_{n=1}^{N}\mathbf{G}_{n,t}, \qquad \Delta_{n,t} = \mathbf{G}_{n,t} - \bar{\mathbf{G}}_t, \tag{2}$$

where $\bar{\mathbf{G}}_t$ summarizes the average spatial response at time step t, and $\Delta_{n,t}$ measures how much the n-th location deviates from this prototype. TDA scores locations from these differential cues because absolute activations alone cannot distinguish truly informative regions from responses that merely follow the dominant mouth motion. Measuring deviation from the global prototype therefore suppresses common components and highlights distinctive local temporal variations:

$$a_{n,t} = \phi(\Delta_{n,t}), \qquad w_{n,t} = \frac{\exp(a_{n,t}/\tau)}{\sum_{j=1}^{N}\exp(a_{j,t}/\tau)}, \tag{3}$$

where $\phi(\cdot)$ is a two-layer MLP that maps the differential feature to a scalar importance logit and τ is a learnable temperature. The final frame-level temporal representation is then obtained by

$$\mathbf{H}_t = \sum_{n=1}^{N} w_{n,t}\mathbf{G}_{n,t}, \qquad \mathbf{H} \in \mathbb{R}^{B\times T\times C}. \tag{4}$$

As a result, the downstream word-level temporal encoder receives a frame sequence formed after location-wise temporal modeling on the encoder feature map, rather than one obtained by immediately averaging that spatial map.

3.3 Viseme-Guided Aggregation for Sequence-Level Temporal Learning

As illustrated in Fig. 4, $\mathbf{H}$ is dispatched to two parallel temporal branches: a word-level temporal encoder and VGA. VGA introduces sequence-level supervision through a CTC decoder and a viseme-guided aggregation pathway.

Viseme-aware CTC decoder. Since DVS-Lip provides only word labels rather than frame-aligned viseme annotations, CTC (Graves et al., 2006) is a natural choice for introducing sequence structure without requiring manual alignment. Following prior work (Hazen et al., 2004; Zhang et al., 2025b), each word label is first converted to a pronunciation sequence and then mapped to a viseme sequence through a fixed conversion table. VGA first constructs a compact sequence of temporal segments and then models their interactions:

$$\mathbf{S} = \Psi_{\text{seg}}\big(\Gamma_{\text{seg}}(\mathbf{H})\big), \qquad \mathbf{c} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{S}_k, \tag{5}$$

where $\Gamma_{\text{seg}}(\cdot)$ denotes the segment generation operator, $\Psi_{\text{seg}}(\cdot)$ denotes the bidirectional segment modeling module, $\mathbf{S} \in \mathbb{R}^{B\times K\times D_s}$ are the refined segment states, $\mathbf{c} \in \mathbb{R}^{B\times D_s}$ is the global viseme segment context, and $K = T/\ell$ is the number of segments. In implementation, $\Gamma_{\text{seg}}(\cdot)$ is realized by temporal convolution and average pooling, while $\Psi_{\text{seg}}(\cdot)$ is instantiated as a bidirectional temporal sequence model.

The decoder then projects each segment state into a viseme distribution:

$$\mathbf{P} = \Omega_{\text{vi}}(\mathbf{S}), \qquad \mathbf{P} \in \mathbb{R}^{B\times K\times C_{\text{vi}}}, \tag{6}$$

where $\Omega_{\text{vi}}(\cdot)$ denotes the viseme prediction head and C_{vi} is the number of viseme categories including the CTC blank. Let $\Pi(\mathbf{y}^{\text{vi}})$ denote the valid CTC alignment paths for target sequence $\mathbf{y}^{\text{vi}}$. Its conditional probability and the corresponding auxiliary objective are:

$$p(\mathbf{y}^{\text{vi}} \mid \mathbf{H}) = \sum_{\boldsymbol{\pi}\in\Pi(\mathbf{y}^{\text{vi}})}\prod_{k=1}^{K} p(\pi_k \mid \mathbf{S}_k), \qquad \mathcal{L}_{\text{ctc}} = -\log p(\mathbf{y}^{\text{vi}} \mid \mathbf{H}), \tag{7}$$

Since multiple words can share the same viseme sequence, directly decoding a word from the CTC output is ambiguous. We therefore use the CTC-supervised segment representation to condition temporal aggregation, allowing viseme structure to contribute to the fused feature for word recognition.

Viseme-guided aggregation pathway. In parallel, the word-level temporal encoder transforms $\mathbf{H}$ into $\mathbf{F}$ and retains its temporal mean as the baseline representation:

$$\mathbf{F} = \Psi_{\text{word}}(\mathbf{H}), \qquad \mathbf{f}_{\text{mean}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{F}_t. \tag{8}$$

Here, $\mathbf{F} \in \mathbb{R}^{B\times T\times D_w}$, and $\Psi_{\text{word}}(\cdot)$ is the same three-layer bidirectional GRU used in the baseline. However, mean pooling treats all time steps uniformly and cannot explicitly emphasize the visually critical articulation phases highlighted by the CTC decoder. VGA instead uses the viseme segment context $\mathbf{c}$ to modulate temporal aggregation.

Specifically, let $\mathbf{F} \in \mathbb{R}^{T\times D_w}$ denote the word-level feature matrix of a sample, and let $\mathbf{1} \in \mathbb{R}^{T}$ be an all-ones vector. We use the CTC-supervised segment context $\mathbf{c}$ to compute temporal weights over $\mathbf{F}$ and aggregate the weighted features:

$$\begin{aligned} &\mathbf{Q} = \mathbf{F}\mathbf{W}_f^{\top}, \qquad \mathbf{k} = \mathbf{W}_c\mathbf{c}, \\ &\boldsymbol{\alpha} = \text{Softmax}\big(\tanh\big(\mathbf{Q} + \mathbf{1}\mathbf{k}^{\top}\big)\mathbf{w}\big), \qquad \mathbf{f}_{\text{vga}} = \sum_{t=1}^{T}\alpha_t\mathbf{F}_t, \end{aligned} \tag{9}$$

where $\mathbf{W}_f \in \mathbb{R}^{D_h\times D_w}$ and $\mathbf{W}_c \in \mathbb{R}^{D_h\times D_s}$ project the word-level features and segment context into a common hidden space, and $\mathbf{w} \in \mathbb{R}^{D_h}$ scores each fused temporal state. The resulting $\boldsymbol{\alpha} \in \Delta^{T-1}$ forms a distribution over the T word-level features, whose weighted sum gives $\mathbf{f}_{\text{vga}}$.

Finally, VGA combines the baseline mean-pooled representation and the viseme-guided representation through a learnable gate:

$$\gamma = \sigma(g) \in (0, 1), \qquad \mathbf{r} = \mathbf{f}_{\text{mean}} + \gamma\big(\mathbf{f}_{\text{vga}} - \mathbf{f}_{\text{mean}}\big), \tag{10}$$

where g is a trainable scalar. The final word logits are produced by a classifier applied to $\mathbf{r}$. Thus, the CTC decoder not only regularizes learning through viseme supervision, but also reshapes the temporal aggregation used for word recognition.

Model	Acc1	Acc2	Acc
I3D (Carreira & Zisserman, 2017)	58.24	77.68	67.94
TANet (Liu et al., 2021)	58.36	79.17	68.74
ACTION-Net (Wang et al., 2021)	58.32	79.41	68.84
GET (Peng et al., 2023)	58.96	80.82	69.80
MSTP (Tan et al., 2022)	62.17	82.07	72.10
E-Mamba (Zhang et al., 2025d)	63.75	82.71	73.23
SSE-Net (Zhang et al., 2025e)	63.06	84.45	73.73
MTGA (Zhang et al., 2025a)	63.90	86.38	75.08
STCNet (Zhang et al., 2025c)	65.86	87.43	76.62
HFR-Lip (Zhang et al., 2025f)	66.47	87.67	77.06
MSTP++ (Tan et al., 2024)	67.43	89.28	78.33
FAST-HG(30 bins) (Zhang et al., 2025b)	68.81	<u>89.97</u>	<u>79.37</u>
FAST-HG(210 bins) (Zhang et al., 2025b)	<u>69.11</u>	89.46	79.26
Ours	67.23	87.79	77.49
Ours + Pretrain	**69.31**	**90.81**	**80.04**

Table 1: Performance comparison on the DVS-Lip dataset. Best results are shown in bold and second-best results are underlined.

3.4 Perturbation-Consistent Teacher–Student Training

We adopt an exponential moving average (EMA) teacher–student strategy following prior work (Tarvainen & Valpola, 2017; Caron et al., 2021) to regularize training and improve generalization. Concretely, this consistency is imposed between weakly and strongly augmented views of the same event sample.

The teacher shares the same architecture as the student, but its parameters are updated by EMA:

$$\theta'_t \leftarrow \mu(e)\theta'_{t-1} + (1-\mu(e))\theta_t, \tag{11}$$

where θ_t and θ'_t denote the student and teacher parameters at training step t, respectively, and $\mu(e)$ is the EMA decay rate at epoch e. In practice, we use a two-stage EMA decay schedule: a smaller decay in the early stage allows the teacher to follow the supervised student more rapidly, while a larger decay in the later stage makes the teacher evolve more smoothly and provide stabler soft targets.

During training, both branches are derived from the same event sample and share the same base augmentation. Specifically, we first apply a random spatial crop and a random horizontal flip, and then voxelize the result to obtain a base tensor $\mathbf{x}^{\text{base}}$. The teacher directly takes this tensor as its weakly augmented input, i.e., $\mathbf{x}^w = \mathbf{x}^{\text{base}}$. In contrast, the student input is obtained by sequentially applying spatial cutout, random zooming, and temporal masking to the same base tensor:

$$\mathbf{x}^s = \mathcal{A}_{\text{mask}}\big(\mathcal{A}_{\text{zoom}}\big(\mathcal{A}_{\text{cutout}}(\mathbf{x}^{\text{base}})\big)\big)\,. \tag{12}$$

Thus, the teacher input is weakly augmented rather than unaugmented, while the student receives three additional voxel-level perturbations. Their logits are computed as:

$$\mathbf{z}_{\text{tea}} = f_{\theta'}(\mathbf{x}^w), \quad \mathbf{z}_{\text{stu}} = f_\theta(\mathbf{x}^s), \tag{13}$$

where f_θ and $f_{\theta'}$ denote the student and teacher networks, respectively. The training objective is:

$$\mathcal{L} = \lambda_{\text{ce}}\mathcal{L}_{\text{ce}} + \lambda_{\text{kl}}(e)\mathcal{L}_{\text{kl}} + \lambda_{\text{ctc}}\mathcal{L}_{\text{ctc}}, \tag{14}$$

where $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss, $\mathcal{L}_{\text{ctc}}$ is the auxiliary CTC loss introduced in VGA, and λ_{ce} and λ_{ctc} are balancing weights. Following Mean Teacher (Tarvainen & Valpola, 2017), the KL term uses a detached teacher target and an epoch-dependent ramp-up weight:

$$\mathcal{L}_{\text{kl}} = D_{\text{KL}}\left(\text{Softmax}(\mathbf{z}_{\text{tea}}) \parallel \text{Softmax}(\mathbf{z}_{\text{stu}})\right), \qquad \lambda_{\text{kl}}(e) = \lambda_{\max} g(e), \tag{15}$$

where $\lambda_{\max}$ is the maximum consistency weight and $g(e)$ is a ramp-up function that increases from 0 to 1.

Model	Acc1	Acc2	Acc
I3D (Carreira & Zisserman, 2017)	69.44	81.92	75.68
TANet (Liu et al., 2021)	70.40	82.00	76.20
ACTION-Net (Wang et al., 2021)	69.36	82.40	75.88
MSTP (Tan et al., 2022)	72.68	84.84	78.76
MSTP++ (Tan et al., 2024)	75.08	87.48	81.28
FAST-HG(30 bins) (Zhang et al., 2025b)	<u>77.27</u>	**89.52**	<u>83.40</u>
FAST-HG(210 bins) (Zhang et al., 2025b)	77.25	89.02	83.14
Ours	**78.08**	<u>89.08</u>	**83.58**

Table 2: Performance comparison on the DVS-LRW100 dataset. Best results are shown in bold and second-best results are underlined.

DropPath	EMA	TDA	VGA	Acc (%)
–	–	–	–	74.09
✓	–	–	–	74.96
✓	✓	–	–	75.78/75.82*
✓	–	✓	–	75.59
✓	–	–	✓	76.58
✓	✓	✓	–	76.54/76.58*
✓	✓	✓	✓	77.49/76.82*

Table 3: Ablation study of the proposed modules. Entries marked with * denote teacher accuracy; unmarked values denote student accuracy.

4 Experiments

4.1 Dataset Settings

DVS-Lip Dataset. DVS-Lip was introduced by MSTP (Tan et al., 2022) as the first event-based lip-reading dataset. It contains 19,871 word-level samples captured by a DAVIS-346 event camera from 40 volunteers. The vocabulary consists of 100 words selected from LRW (Chung et al., 2017), including visually similar word pairs and common words.

We follow the official split of DVS-Lip, using 14,896 samples for training and 4,975 samples for testing. The official test set is divided into two subsets. The first subset, denoted as Acc1, contains visually confusing word pairs and is therefore more challenging. The second subset, denoted as Acc2, contains the remaining common words. We report Acc1, Acc2, and the overall accuracy Acc for comparison.

DVS-LRW100 Dataset. DVS-LRW100 was introduced by MSTP++ (Tan et al., 2024) as a synthetic event-based lip-reading dataset generated from LRW using the v2e simulator. It adopts the same 100 word classes as DVS-Lip and contains 107,000 samples, including 97,000 for training, 5,000 for validation, and 5,000 for testing. The samples are balanced across visually confusable and non-confusable subsets.

4.2 Main Results

Table 1 shows state-of-the-art performance on DVS-Lip. When pretrained on DVS-LRW100 and fine-tuned on DVS-Lip, **Ours + Pretrain** reaches **69.31%** on Acc1, **90.81%** on Acc2, and **80.04%** in overall accuracy, each ranking first in Table 1. It is the first result to exceed 90% on Acc2 and 80% in overall accuracy on DVS-Lip. These gains show that the proposed temporal aggregation design benefits substantially from large-scale event-based pretraining and transfers effectively from the synthetic DVS-LRW100 data to the real DVS-Lip benchmark. For completeness, the non-pretrained model already reaches 77.49%, confirming that the proposed design remains highly competitive even without cross-dataset pretraining.

Table 2 shows state-of-the-art performance on DVS-LRW100 under the single-branch setting. Our method attains **78.08%** on Acc1, 89.08% on Acc2, and **83.58%** in overall accuracy. Since Acc1 measures performance on the visually confusable word subset, the best Acc1 indicates strong discriminative ability on the most challenging examples, while the highest overall accuracy shows that the proposed temporal aggregation design transfers effectively to the larger synthetic dataset. Although Acc2 is not highest, it remains second-best and close to the strongest competing method. These results further support the effectiveness and generalization of the proposed design beyond DVS-Lip.

4.3 Ablation Study

Module Ablation. Table 3 verifies each component's contribution. Relative to the 74.09% baseline, DropPath improves accuracy to 74.96%, showing that moderate stochastic regularization benefits event-based lip reading. Teacher–student training further raises performance to 75.78% for the

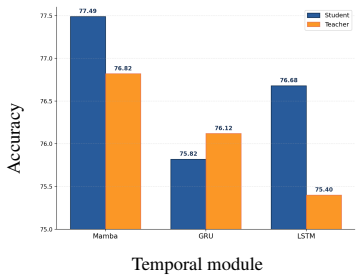


Figure 5: Temporal-module ablation in TDA. Blue and orange bars denote student and teacher accuracy, respectively.

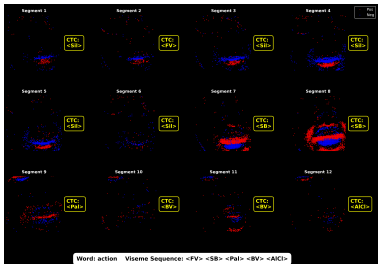


Figure 6: Qualitative visualization of VGA decoding on an event voxel sample of the word "action".

student and 75.82% for the teacher, confirming the effectiveness of perturbation-consistent supervision under strong event-level augmentations. With a single temporal enhancement, TDA alone improves the student accuracy to 75.59%, while VGA alone yields a clearly stronger result of 76.58%.

The combined results further clarify the complementarity among the proposed components. When TDA is added on top of DropPath and teacher–student training, the student accuracy reaches 76.54%, validating the benefit of performing temporal modeling before spatial compression. After further introducing VGA, the student attains the best result of 77.49%, corresponding to a total gain of 3.40 points over the baseline. Notably, the standalone VGA result is already close to the combination of TDA and teacher–student training, while the full model still produces the strongest overall performance. These observations indicate that TDA and VGA contribute from different aspects of temporal modeling: TDA strengthens local trajectory encoding before aggregation, whereas VGA further refines the final word-level representation through viseme-aware supervision and viseme-guided scoring, reweighting, and fusion.

Temporal Module Ablation In TDA. Fig. 5 compares different temporal modeling modules used inside TDA. Replacing Mamba with GRU leads to a clear drop in student accuracy from 77.49% to 75.82%, while LSTM achieves 76.68%, which is better than GRU but still below Mamba. The same advantage is also reflected in the teacher results. This trend suggests that the temporal branch inside TDA benefits not only from recurrence, but more importantly from a stronger ability to model long-range dependencies while preserving fine-grained local variation. Compared with GRU and LSTM, Mamba performs selective state-space propagation over the full trajectory and can adaptively emphasize informative event responses at different time steps, which is particularly suitable for sparse lip-motion dynamics. As a result, Mamba is better aligned with the goal of TDA, namely injecting temporally discriminative information into each spatial location before spatial aggregation.

4.4 Qualitative Analysis

Fig. 6 provides a qualitative example of the decoding behavior of VGA. For the word "action", the model partitions the event voxel into 12 visual segments and produces the ordered viseme sequence ⟨FV⟩ ⟨SB⟩ ⟨Pal⟩ ⟨BV⟩ ⟨AICI⟩ after removing blank predictions. Despite the absence of explicit frame-level alignment, the decoded outputs still follow a consistent visual articulation progression of the target word, further indicating that the CTC decoder learns meaningful viseme-aware structure.

5 Conclusion

We present TVTA, an event-based lip-reading framework that strengthens temporal learning before spatial aggregation and during sequence aggregation. TDA models location-wise encoder trajectories with shared bidirectional Mamba and differential spatial weighting. VGA uses CTC-supervised viseme segment context to reweight and fuse word-level temporal features, while EMA teacher–student consistency improves robustness to strong event augmentations. Experiments on DVS-Lip and DVS-LRW100 establish new best results under the corresponding evaluation settings, and ablations verify the complementary gains from TDA, VGA, and consistency training.

AI USE STATEMENT

In this work, we used generative AI tools to polish the language of the manuscript and improve its readability. We did not use generative AI tools to generate research ideas, methods, experimental results, analyses or figures. We reviewed all AI-assisted edits to ensure technical accuracy and preserve the intended meaning. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Duygu Altinok. Boosting ctc-based asr using llm-based intermediate loss regularization. In *International Conference on Text, Speech, and Dialogue*, pp. 275–288. Springer, 2025.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Kyunghyun Cho, Bart Van Merriënboer, Çağlar Gulçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1724–1734, 2014.

Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

Manon Dampfhoffer and Thomas Mesquida. Neuromorphic lip-reading with signed spiking gated recurrent units. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2024.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *International conference on machine learning*, pp. 1607–1616. PMLR, 2018.

David Gimeno-Gómez and Carlos-D Martínez-Hinarejos. Comparison of conventional hybrid and ctc/attention decoders for continuous visual speech recognition. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 3628–3638, 2024.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *Neural Information Processing Systems*, 2020.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First conference on language modeling*, 2024.

Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

Timothy J Hazen, Kate Saenko, Chia-Hao La, and James R Glass. A segment-based audio-visual speech recognizer: Data collection, development, and initial experiments. In *Proceedings of the 6th international conference on Multimodal interfaces*, pp. 235–242, 2004.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9726–9735. IEEE, 2020.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput*, 9(8):1735–1780, 1997.

Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *14th European Conference on Computer Vision, ECCV 2016*, pp. 646–661. Springer Verlag, 2016.

Hao Ju, Zhedong Zheng, Xu Zheng, Wenyue Chen, Lin Wang, Dong Wang, Huchuan Lu, and Xu Jia. Event-based lip reading with triplane fusion network. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2026.

Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. In *European conference on computer vision*, pp. 237–255. Springer, 2024.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. R-drop: regularized dropout for neural networks. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, pp. 10890–10905, 2021.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: visual state space model. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, pp. 103031–103063, 2024.

Zhaoyang Liu, Limin Wang, Wayne Wu, Chen Qian, and Tong Lu. Tam: Temporal adaptive module for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 13708–13718, 2021.

Yansong Peng, Yueyi Zhang, Zhiwei Xiong, Xiaoyan Sun, and Feng Wu. Get: Group event transformer for event-based vision. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6015–6025. IEEE, 2023.

Hongwei Ren, Yue Zhou, Jiadong Zhu, Xiaopeng Lin, Haotian Fu, Yulong Huang, Yuetong Fang, Fei Ma, Hao Yu, and Bojun Cheng. Rethinking efficient and effective point-based networks for event camera classification and regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Hongwei Ren, Fei Ma, Yuetong Fang, Hongxiang Huang, Yue Zhou, Yulong Huang, Haotian FU, Ziyi Yang, Youxin Jiang, Xiangqian Wu, et al. Scalable event cloud network for event-based classification. In *Forty-third International Conference on Machine Learning*, 2026.

Yiqing Shen, Liwu Xu, Yuzhe Yang, Yaqian Li, and Yandong Guo. Self-distillation from the last mini-batch for consistency regularization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11943–11952, 2022.

Jimmy T.H. Smith, Andrew Warrington, and Scott Linderman. Simplified state space layers for sequence modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Ai8Hw3AXqks.

Ganchao Tan, Yang Wang, Han Han, Yang Cao, Feng Wu, and Zheng-Jun Zha. Multi-grained spatio-temporal features perceived network for event-based lip-reading. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20094–20103, 2022.

Ganchao Tan, Zengyu Wan, Yang Wang, Yang Cao, and Zheng-Jun Zha. Tackling event-based lip-reading by exploring multigrained spatiotemporal clues. *IEEE Transactions on Neural Networks and Learning Systems*, 36(5):8279–8291, 2024.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 1195–1204, 2017.

Marshall Thomas, Edward Fish, and Richard Bowden. Vallr: Visual asr language model for lip reading. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2846–2856, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Zhengwei Wang, Qi She, and Aljosa Smolic. Action-net: Multipath excitation for action recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13209–13218. IEEE, 2021.

Zengwei Yao, Wei Kang, Xiaoyu Yang, Fangjun Kuang, Liyong Guo, Han Zhu, Zengrui Jin, Zhaoqing Li, Long Lin, and Daniel Povey. Cr-ctc: Consistency regularization on ctc for improved speech recognition. In *The Thirteenth International Conference on Learning Representations*, 2025.

Sukmin Yun, Jongjin Park, Kimin Lee, and Jinwoo Shin. Regularizing class-wise predictions via self-knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13876–13885, 2020.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3713–3722, 2019.

Wenhao Zhang, Jun Wang, Yong Luo, Lei Yu, Wei Yu, Zheng He, and Jialie Shen. Mtga: multi-view temporal granularity aligned aggregation for event-based lip-reading. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, pp. 10176–10184, 2025a.

Xueyi Zhang, Jialu Sun, Chengwei Zhang, Xianghu Yue, Tianfang Xiao, Siqi Cai, Mingrui Lao, and Haizhou Li. Eventlip: Enhancing event-based lip reading via frequency-aware spatiotemporal hypergraph modeling. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 8263–8272, 2025b.

Xueyi Zhang, Jialu Sun, Peiyin Zhu, Bowen Wang, Mingrui Lao, and Yanming Guo. Event-based binary neural networks for efficient and accurate lip reading. In *International Conference on Machine Learning and Intelligent Computing*, pp. 212–221. PMLR, 2025c.

Xueyi Zhang, Jialu Sun, Peiyin Zhu, Tianfang Xiao, Mingrui Lao, and Yanming Guo. Mamba-based temporal modeling for event-based lip reading. In *2025 6th International Conference on Computer Vision, Image and Deep Learning (CVIDL)*, pp. 926–930. IEEE, 2025d.

Yi Zhang, Xiuping Liu, Hongchen Tan, and Xin Li. Spectrum-guided spatial feature enhancement network for event-based lip-reading. *Neurocomputing*, 636:129974, 2025e.

Yi Zhang, Xiuping Liu, Jun Zhou, Yang Liu, and Hongchen Tan. Semantics-aware high-frequency enhancement for event-based lip-reading. *Information Sciences*, pp. 123026, 2025f.

Zhilu Zhang and Mert R Sabuncu. Self-distillation as instance-specific label smoothing. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, pp. 2184–2195, 2020.

Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. In *International Conference on Machine Learning*, pp. 62429–62442. PMLR, 2024.

Word	Viseme Sequence
accused	BV SB Y RV AlCl SB
action	FV SB Pal BV AlCl
allow	BV L RV
allowed	BV L RV SB
america	BV LCI FV R BV SB BV
american	BV LCI FV R BV SB BV AlCl
another	BV AlCl BV SB R
around	R RV AlCl SB
attacks	BV SB FV SB AlCl
banks	LB FV VlCl SB AlCl
become	LB OV SB BV LCI
being	LB OV OV VlCl
benefit	LB FV AlCl BV LFr OV SB
benefits	LB FV AlCl BV LFr OV SB AlCl
between	LB OV SB RV OV AlCl
billion	LB OV L Y BV AlCl
called	SB RV L SB
capital	SB FV LB BV SB BV L
challenge	Pal FV L BV AlCl Pal
change	Pal FV AlCl Pal
chief	Pal OV LFr
couple	SB BV LB BV L
court	SB RV R SB
death	SB FV SB
described	SB OV AlCl SB R FV LB SB

Table 4: Word-to-viseme label table used in VGA (Part I).

Word	Viseme Sequence
difference	SB OV LFr R BV AlCl AlCl
different	SB OV LFr R BV AlCl SB
during	SB RV R OV VlCl
economic	FV SB BV AlCl BV LCI OV SB
education	FV Pal BV SB FV Pal BV AlCl
election	OV L FV SB Pal BV AlCl
england	OV VlCl SB L BV AlCl SB
evening	OV LFr AlCl OV VlCl
everything	FV LFr R OV SB OV VlCl
exactly	OV SB AlCl FV SB SB L OV
general	Pal FV AlCl R BV L
germany	Pal R LCI BV AlCl OV
giving	SB OV LFr OV VlCl
ground	SB R RV AlCl SB
happen	FV FV LB BV AlCl
happened	FV FV LB BV AlCl SB
having	FV FV LFr OV VlCl
heavy	FV FV LFr OV
house	FV RV AlCl
hundreds	FV BV AlCl SB R BV SB AlCl
immigration	OV LCI BV SB R FV Pal BV AlCl
judge	Pal BV Pal
labour	L FV LB R
leaders	L OV SB R AlCl
legal	L OV SB BV L

Table 5: Word-to-viseme label table used in VGA (Part II).

A Dataset Details

Datasets. We use two event-based lip-reading benchmarks in this work. DVS-Lip is the main real-world benchmark used throughout the paper. DVS-LRW100 is a larger synthetic benchmark generated from LRW and is used both for direct evaluation and for the pretraining stage of the cross-dataset transfer experiment reported on DVS-Lip. Unless otherwise specified, all data splits follow the official protocol of each dataset.

Viseme label table. For reproducibility, Tables 4–7 list the 100 word classes together with the corresponding viseme label sequences used as CTC supervision targets in VGA. The same word-to-viseme targets are used consistently for both DVS-Lip and DVS-LRW100.

Word	Viseme Sequence
little	L OV SB BV L
london	L BV AlCl SB BV AlCl
majority	LCI BV Pal RV R BV SB OV
meeting	LCI OV SB OV VlCl
military	LCI OV L BV SB FV R OV
million	LCI OV L Y BV AlCl
minutes	LCI OV AlCl BV SB AlCl
missing	LCI OV AlCl OV VlCl
needs	AlCl OV SB AlCl
number	AlCl BV LCI LB R
numbers	AlCl BV LCI LB R AlCl
paying	LB FV OV VlCl
perhaps	LB R FV FV LB AlCl
point	LB RV AlCl SB
potential	LB BV SB FV AlCl Pal BV L
press	LB R FV AlCl
price	LB R FV AlCl
question	SB RV FV AlCl Pal BV AlCl
really	R OV L OV
right	R FV SB
russia	R BV Pal BV
russian	R BV Pal BV AlCl
saying	AlCl FV OV VlCl
security	AlCl OV SB Y RV R BV SB OV
several	AlCl FV LFr R BV L

Table 6: Word-to-viseme label table used in VGA (Part III).

Word	Viseme Sequence
should	Pal RV SB
significant	AlCl BV SB AlCl OV LFr OV SB BV AlCl SB
spend	AlCl LB FV AlCl SB
spent	AlCl LB FV AlCl SB
started	AlCl SB BV R SB OV SB
still	AlCl SB OV L
support	AlCl BV LB RV R SB
syria	AlCl OV R OV BV
syrian	AlCl OV R OV BV AlCl
taken	SB FV SB BV AlCl
taking	SB FV SB OV VlCl
terms	SB R LCI AlCl
these	SB OV AlCl
thing	SB OV VlCl
think	SB OV VlCl SB
times	SB FV LCI AlCl
tomorrow	SB BV LCI BV R RV
under	BV AlCl SB R
warning	RV RV R AlCl OV VlCl
water	RV RV SB R
welcome	RV FV L SB BV LCI
words	RV R SB AlCl
worst	RV R AlCl SB
years	Y OV R AlCl
young	Y BV VlCl

Table 7: Word-to-viseme label table used in VGA (Part IV).

B Implementation Details

Event representation. Each cropped event stream is converted into a dual-channel voxel tensor. Let an event stream be denoted as

$$\mathcal{E} = \{(t_i, x_i, y_i, p_i)\}_{i=1}^{M},$$

where t_i, (x_i, y_i), and p_i denote the timestamp, spatial coordinate, and polarity of the i-th event, respectively. The event stream is voxelized as

$$\mathbf{V} = \Phi(\mathcal{E}; T, H, W) \in \mathbb{R}^{T\times 2\times H\times W}, \tag{16}$$

where $\Phi(\cdot)$ denotes voxelization, $H = W = 88$, and $T = \texttt{seq_len} \times \texttt{num_bins}$. In the main setting, $\texttt{seq_len} = 30$ and $\texttt{num_bins} = 2$, hence $T = 60$.

Following the implementation, timestamps are first normalized into discrete temporal bins:

$$\begin{aligned} \tilde{t}_i &= \frac{t_i - \min_j t_j}{\max_j t_j - \min_j t_j + \varepsilon}, \\ b_i &= \min\big(T-1, \lfloor T\tilde{t}_i \rfloor\big), \end{aligned} \tag{17}$$

where ε is a small constant for numerical stability and $b_i \in \{0, \ldots, T-1\}$ denotes the temporal bin index of the i-th event. The two channels then store positive and negative event counts separately:

$$\begin{aligned} \mathbf{V}_{t,0,y,x} &= \sum_{i=1}^{M} \mathbf{1}[b_i = t,\ (x_i, y_i) = (x, y),\ p_i > 0], \\ \mathbf{V}_{t,1,y,x} &= \sum_{i=1}^{M} \mathbf{1}[b_i = t,\ (x_i, y_i) = (x, y),\ p_i = 0], \end{aligned} \tag{18}$$

where the dataset stores negative-polarity events with $p_i = 0$. Therefore, the model input is a single voxel tensor with two polarity channels rather than two separate spatial branches.

During training, the teacher and student receive two differently augmented versions of the same dual-channel voxel tensor, denoted by $\mathbf{V}^{\mathrm{tea}}$ and $\mathbf{V}^{\mathrm{stu}}$, respectively. During inference, the model directly takes the clean voxel tensor $\mathbf{V}$ as input.

Training protocol. For the main experiments, the model is trained end-to-end with a word-level classification objective, an auxiliary viseme-level CTC objective, and a teacher–student consistency objective. For the transferred DVS-Lip result, the model is first pretrained on DVS-LRW100 and then fine-tuned on DVS-Lip.

Data augmentation. We follow the augmentation setting used in prior event-based lip-reading work (Dampfhoffer & Mesquida, 2024). Each 128×128 event sample is first center-cropped to a 96×96 mouth region. During training, it is randomly cropped to 88×88, and horizontal flipping is then applied with probability 0.5. The cropped events are voxelized into the dual-channel tensor $\mathbf{V} \in \mathbb{R}^{T\times 2\times 88\times 88}$. For EMA teacher–student training, both branches are constructed from the same cropped-and-flipped base tensor. Therefore, the teacher input is not strictly clean: it shares the same random crop and random flip as the student.

The teacher branch directly uses this base tensor without further perturbation. The student branch applies three additional voxel-level augmentations sequentially: spatial cutout, random zooming, and temporal masking. In code, Cutout uses 4 random holes with maximum side length 20, Zoom uses a maximum spatial scale of 26 pixels, and temporal masking applies 6 random temporal masks with maximum length $T/5$. During testing, the event stream is center-cropped from 96×96 to 88×88 and then voxelized into a clean dual-channel tensor $\mathbf{V} \in \mathbb{R}^{T\times 2\times 88\times 88}$ without any random augmentation. Therefore, the teacher–student asymmetry is introduced after the shared crop/flip stage rather than before voxelization.

Optimization settings. We use Adam, following prior event-based lip-reading practice; we also observed that replacing Adam with AdamW did not yield better results in our experiments. In both pretraining and fine-tuning, the learning rate is warmed up during the first epoch and then annealed by cosine decay.

DropPath is applied to the residual branch of each ResBlock. The drop probabilities are increased stage by stage and set to 0.05, 0.10, 0.15, and 0.20 for the four residual stages, respectively.

TDA and VGA settings. VGA uses temporal segment modeling together with a gated combination of mean aggregation and viseme-guided aggregation. In the implementation, the segment generation stage is realized by a lightweight temporal convolutional block: the first temporal convolution enlarges the channel dimension and aggregates short-range temporal context, after which a second temporal convolution performs non-overlapping temporal downsampling to form segment tokens. The resulting segment sequence is then processed by the segment modeling module described in the main paper. In TDA, the default temporal bin number is 60 in the main setting. Unless otherwise stated, the word-level temporal encoder follows the same implementation as the baseline.

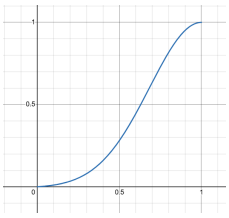


Figure 7: Ramp-up curve of the consistency weight $g(e)$ used in teacher–student training.

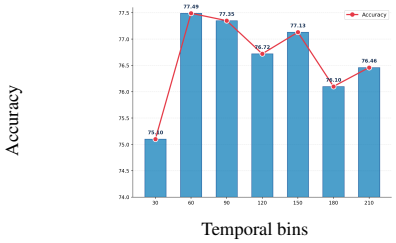


Figure 8: Ablation on temporal-bin settings.

Setting	Pretrain	Fine-tune
Dataset	DVS-LRW100	DVS-Lip
Epochs	100	100
Batch size	256	256
Learning rate	$3 \times 10^{-4} \rightarrow 1 \times 10^{-5}$	$2 \times 10^{-4} \rightarrow 1 \times 10^{-6}$
Weight decay	2×10^{-4}	1×10^{-3}
λ_{ctc}	1.0	2.0
KL weight	0.5	1.0
Pool gate init	0.0	0.0
Segment length	5	5
Segment dim	1024	1024
Segment dropout	0.5	0.5
Attention dim	512	512
Pool dropout	0.5	0.5
EMA decay	$0.99 \rightarrow 0.998$	$0.99 \rightarrow 0.998$
e_{ramp}	40	20
Seed	42	42

Table 8: Main settings used in the DVS-LRW100 pretraining stage and the DVS-Lip fine-tuning stage.

Teacher–student configuration. Following Mean Teacher (Tarvainen & Valpola, 2017), the consistency weight is controlled by a ramp-up function $g(e)$, where $x = (e-1)/(e_{\mathrm{ramp}}-1)$:

$$g(e) = \begin{cases} \dfrac{\exp\left(-5(1-x)^2\right) - \exp(-5)}{1-\exp(-5)}, & e \le e_{\mathrm{ramp}}, \\ 1, & e > e_{\mathrm{ramp}}, \end{cases} \tag{19}$$

Here, e_{ramp} controls the duration of the ramp-up stage. As illustrated in Fig. 7, this schedule increases very slowly at the beginning and rises more rapidly only after the teacher becomes more stable. This design is important because the EMA teacher is still immature in the early stage and cannot yet provide sufficiently reliable soft targets; assigning a large KL weight too early would therefore inject noisy supervision into the student. After the model has learned a more stable decision boundary, the consistency term is strengthened to make the student follow the now more reliable teacher predictions.

C Reproducibility Notes

Randomness control. Unless otherwise specified, all experiments use random seed 42. We seed Python random, NumPy, PyTorch CPU/GPU generators, and data-loader workers consistently, where worker seeds are derived from `torch.initial_seed()`.

Parameter settings. The detailed parameter settings used in pretraining and fine-tuning are reported separately in Table 8.

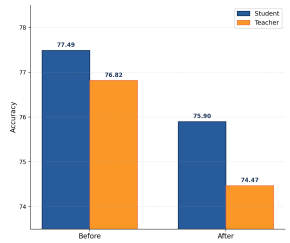


Figure 9: Comparison of temporal modeling placement in TDA. The proposed design applies temporal modeling before differential aggregation, whereas the alternative performs it after spatial aggregation.

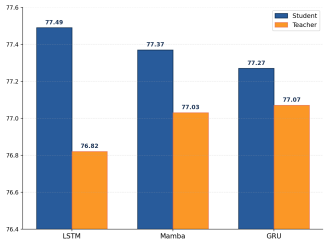


Figure 10: VGA segment model ablation.

$\lambda_{\text{ctc}}/\lambda_{\text{ce}}$	**Original**	**Fine-tune**
0.5	76.34 / 76.10*	79.94 / 79.06*
1.0	77.49 / 76.82*	79.72 / 78.77*
2.0	76.22 / 76.54*	80.04 / 79.02*

Table 9: CTC-to-CE weight ratio ablation. Entries marked with * denote teacher accuracy.

D Additional Ablation Results

D.1 Temporal-Bin Ablation

Fig. 8 further studies the effect of temporal bin settings. The best accuracy is achieved at 60 bins, and the performance shows an overall downward trend as the number of bins increases further. When the temporal resolution becomes too fine, each bin contains fewer events and the representation becomes increasingly sparse, which weakens stable temporal aggregation. This trend suggests that a moderate temporal discretization is preferable, as it preserves sufficient motion dynamics without over-fragmenting the event stream.

D.2 Placement of Temporal Modeling in TDA

To directly validate the design motivation of TDA, we compare temporal modeling *before* versus *after* differential aggregation. This comparison is more informative than changing the temporal module alone, because it tests the core hypothesis behind TDA rather than only the choice of sequence model. Specifically, the proposed design performs bidirectional Mamba modeling on each spatial trajectory *before* differential aggregation, whereas the alternative first compresses spatial responses and only then applies temporal modeling. If temporal modeling is postponed until after aggregation, the local motion evolution from different lip regions has already been mixed, making it harder for the temporal branch to recover discriminative fine-grained dynamics.

Fig. 9 shows a clear advantage for pre-aggregation temporal modeling. When bidirectional Mamba is applied before differential aggregation, the model reaches 77.49% student accuracy and 76.82% teacher accuracy. In contrast, moving temporal modeling after aggregation reduces the performance to 75.90% and 74.47%, respectively. The drop is substantial for **both branches**, confirming that the gain of TDA does not come merely from adding a Mamba block, but from placing temporal modeling at the **correct stage**. These results directly support the motivation of TDA: temporal dynamics should be injected into each spatial location **before spatial compression**, so that localized motion trajectories are preserved rather than prematurely mixed by aggregation.

D.3 CTC-to-CE Weight Ratio Ablation

Table 9 examines the balance between the CTC loss and the cross-entropy loss in both the original training setting and the fine-tuning setting. In the original setting, the best student result is achieved

$\lambda_{\mathrm{kl}}/\lambda_{\mathrm{ce}}$	**Original**	**Fine-tune**
0.5	76.38 / 76.58*	79.76 / 79.20*
1.0	77.49 / 76.82*	80.04 / 79.02*
2.0	76.84 / 76.04*	79.54 / 78.27*

Table 10: KL-to-CE weight ratio ablation. Entries marked with * denote teacher accuracy.

Segment Ratio	**Acc (%)**
$1/5$	77.49 / 76.82*
$1/3$	76.86 / 76.48*
$1/2$	76.95 / 76.24*
$1/1$	76.76 / 75.98*

Table 11: VGA segment ratio ablation. Entries marked with * denote teacher accuracy.

at $\lambda_{\mathrm{ctc}}/\lambda_{\mathrm{ce}} = 1.0$, while both decreasing the ratio to 0.5 and increasing it to 2.0 lead to lower accuracy. This indicates that the original training stage prefers a balanced trade-off between word-level supervision and viseme-level supervision. In contrast, the fine-tuning stage reaches the strongest student result at $\lambda_{\mathrm{ctc}}/\lambda_{\mathrm{ce}} = 2.0$, whereas the teacher result is slightly higher at 0.5 than at 2.0. Overall, these results suggest that after pretraining, fine-tuning can benefit from a relatively stronger viseme-aware constraint on the student branch during adaptation to DVS-Lip.

D.4 KL-to-CE Weight Ratio Ablation

Table 10 evaluates the weight of KL consistency regularization. In the original training setting, $\lambda_{\mathrm{kl}}/\lambda_{\mathrm{ce}} = 1.0$ again gives the best student accuracy, whereas both weaker and stronger consistency weights are suboptimal. The same pattern is observed in the fine-tuning setting, where the best student result is also obtained at 1.0. Compared with 0.5 and 2.0, this setting provides a better balance between perturbation-consistent regularization and task-specific optimization. These results suggest that the KL term should remain moderate: if it is too small, the consistency signal is underused; if it is too large, the optimization becomes overly constrained by teacher–student agreement.

D.5 VGA Segment Ratio Ablation

Table 11 studies the temporal segment ratio used in VGA. This ablation is reported in the original training setting, since the segment ratio is a module-specific design choice rather than a transfer-stage optimization hyper-parameter. The $1/5$ setting performs best. As the segmentation becomes denser, the sequence is partitioned into more sparse or nearly blank segments, which weakens the stability of segment-level modeling and reduces the usefulness of the decoded context.

D.6 VGA Segment Modeling Ablation

Fig. 10 compares different segment modeling modules inside VGA. The highest accuracies obtained by the three alternatives are all very close: LSTM reaches 77.49% on the student branch, Mamba reaches 77.37%, and GRU reaches 77.27%. This indicates that, unlike TDA, the segment modeling stage in VGA is not highly sensitive to the specific temporal module choice. In other words, once the segment sequence is constructed, replacing the internal modeling unit changes the final performance only marginally. This behavior suggests that VGA benefits mainly from the segment-level viseme supervision and context-guided aggregation mechanism itself, whereas the exact choice of segment modeling module plays a comparatively secondary role.

E Additional Theoretical Note

CTC principle. CTC is particularly suitable for the present setting because DVS-Lip only provides word labels, while frame-level viseme boundaries are unavailable. Let $\mathcal{V}$ denote the viseme vocabulary and $\varnothing$ the blank symbol. For a target viseme sequence $\mathbf{y}^{\mathrm{vi}} = (y_1, \ldots, y_U)$ and a segment sequence of length K, CTC considers all monotonic paths $\boldsymbol{\pi} \in (\mathcal{V} \cup \{\varnothing\})^K$ that collapse to $\mathbf{y}^{\mathrm{vi}}$ under the many-to-one mapping $\mathcal{B}(\cdot)$, which removes blanks and merges repeated labels. The valid path set is

$$\Pi(\mathbf{y}^{\mathrm{vi}}) = \left\{ \boldsymbol{\pi} \in (\mathcal{V} \cup \{\varnothing\})^K \;\middle|\; \mathcal{B}(\boldsymbol{\pi}) = \mathbf{y}^{\mathrm{vi}} \right\}. \tag{20}$$

Given segment states $\mathbf{S}_1, \ldots, \mathbf{S}_K$, the viseme head produces per-segment probabilities, and the probability of one path is

$$\begin{aligned} \mathbf{P}_k &= \Omega_{\mathrm{vi}}(\mathbf{S}_k), \\ p(\boldsymbol{\pi} \mid \mathbf{H}) &= \prod_{k=1}^{K} \mathrm{Softmax}(\mathbf{P}_k)_{\pi_k}. \end{aligned} \tag{21}$$

The target-sequence likelihood marginalizes over all valid paths:

$$\begin{aligned} p(\mathbf{y}^{\mathrm{vi}} \mid \mathbf{H}) &= \sum_{\boldsymbol{\pi} \in \Pi(\mathbf{y}^{\mathrm{vi}})} p(\boldsymbol{\pi} \mid \mathbf{H}), \\ \mathcal{L}_{\mathrm{ctc}} &= -\log p(\mathbf{y}^{\mathrm{vi}} \mid \mathbf{H}). \end{aligned} \tag{22}$$

For efficient evaluation, CTC introduces the blank-augmented target

$$\begin{aligned} \mathbf{y}' &= (\varnothing, y_1, \varnothing, y_2, \ldots, \varnothing, y_U, \varnothing), \\ |\mathbf{y}'| &= 2U + 1, \end{aligned} \tag{23}$$

and computes the likelihood by dynamic programming. Let $\alpha(k, u)$ denote the forward probability of all valid prefixes ending at state u of $\mathbf{y}'$ at segment step k. Its recursion is

$$\phi_{k,u} = \mathrm{Softmax}(\mathbf{P}_k)_{y'_u}, \tag{24}$$

$$\alpha_{k,u} = \big(\alpha_{k-1,u} + \alpha_{k-1,u-1}\big)\phi_{k,u}, \qquad \delta_u = 0, \tag{25}$$

$$\alpha_{k,u} = \big(\alpha_{k-1,u} + \alpha_{k-1,u-1} + \alpha_{k-1,u-2}\big)\phi_{k,u}, \qquad \delta_u = 1. \tag{26}$$

where $\delta_u = 1$ if $y'_u \neq \varnothing$ and $y'_u \neq y'_{u-2}$, and $\delta_u = 0$ otherwise. In implementation, `nn.CTCLoss` carries out this forward–backward summation in log-space, so the model is trained without requiring explicit segment-to-viseme alignment. Therefore, multiple monotonic alignments may correspond to the same target viseme string, and the loss optimizes their total probability jointly rather than enforcing a unique framewise annotation. During decoding, repeated labels and blank symbols are removed by the standard collapse operation. In the following example, $\langle \mathrm{Sil} \rangle$ denotes the CTC blank symbol $\varnothing$. A prediction path such as

$$\langle \mathrm{FV} \rangle, \langle \mathrm{FV} \rangle, \langle \mathrm{Sil} \rangle, \langle \mathrm{SB} \rangle, \langle \mathrm{Pal} \rangle, \langle \mathrm{Pal} \rangle, \langle \mathrm{Sil} \rangle, \langle \mathrm{BV} \rangle, \langle \mathrm{AlCl} \rangle$$

is collapsed into the shorter viseme sequence

$$\langle \mathrm{FV} \rangle, \langle \mathrm{SB} \rangle, \langle \mathrm{Pal} \rangle, \langle \mathrm{BV} \rangle, \langle \mathrm{AlCl} \rangle.$$

This mechanism allows VGA to learn meaningful intermediate viseme structure without requiring manually aligned annotations. In other words, the decoder is only asked to satisfy monotonic sequence consistency with the target viseme string, while the precise alignment between segment positions and viseme units is inferred automatically during training.

Mamba principle. TDA adopts bidirectional Mamba because it provides content-adaptive state-space modeling with linear-time sequence complexity (Gu & Dao, 2024). Compared with recurrent architectures such as GRU and LSTM, Mamba does not rely on a fixed recurrent transition shared uniformly across all time steps; instead, it uses input-dependent state updates, which is better suited to event streams whose informative responses appear sparsely and irregularly over time. Given an input trajectory $\mathbf{x}_1, \ldots, \mathbf{x}_T$, Mamba updates the hidden state by

$$\mathbf{h}_t = \bar{\mathbf{A}}_t \, \mathbf{h}_{t-1} + \bar{\mathbf{B}}_t \, \mathbf{x}_t, \qquad \mathbf{y}_t = \mathbf{C}_t \, \mathbf{h}_t, \tag{27}$$

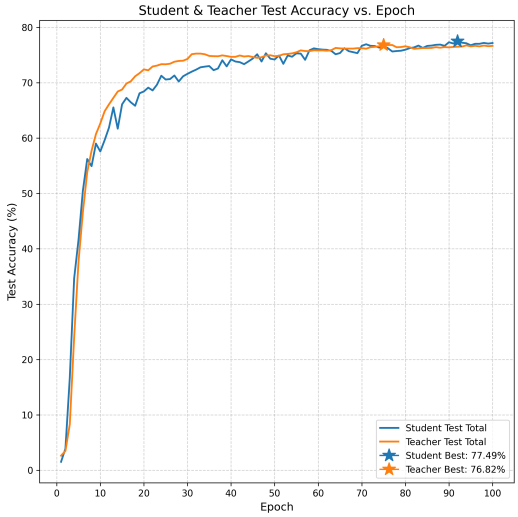


Figure 11: Student and teacher test accuracy curves on the original DVS-Lip training setting.

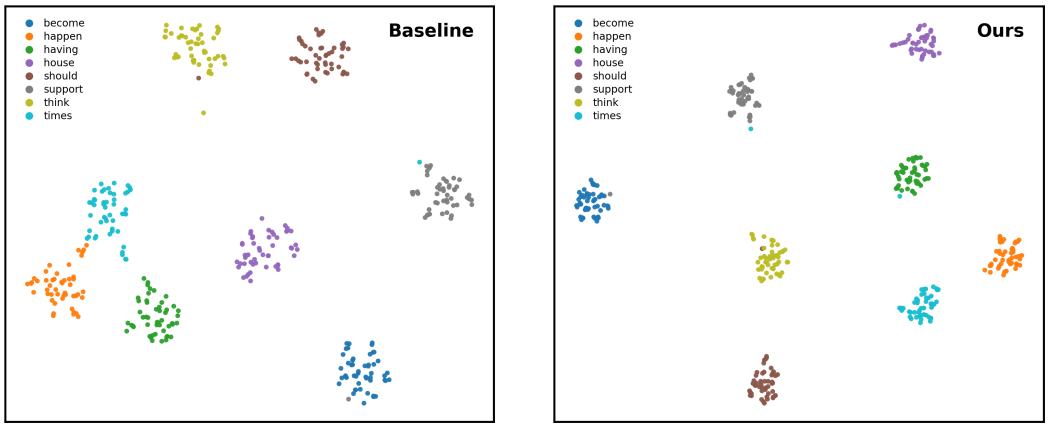
where the discretized transition $\bar{\mathbf{A}}_t$ and input projection $\bar{\mathbf{B}}_t$ depend on the current token through an input-dependent selection mechanism:

$$\Delta_t, \mathbf{B}_t, \mathbf{C}_t = s(\mathbf{x}_t), \qquad \bar{\mathbf{A}}_t = \exp(\Delta_t \mathbf{A}), \qquad \bar{\mathbf{B}}_t = \Delta_t \mathbf{B}_t. \tag{28}$$

Unlike recurrent units that repeatedly compress the sequence through a single hidden transition, this formulation allows the state transition and input injection to vary across time steps, so informative event responses can be selectively propagated along each local trajectory. In TDA, this property is particularly useful because lip-motion cues are sparse and non-uniform across both space and time. Applying Mamba before differential aggregation therefore equips each spatial location with temporally enriched features before cross-location compression.

F More Qualitative Analysis

Teacher–student training dynamics. Fig. 11 shows that the teacher branch is noticeably smoother than the student throughout training, with clearly smaller fluctuations. This is consistent with the EMA update mechanism and indicates that parameter averaging effectively alleviates the optimization noise introduced by strong augmentation. Since the teacher observes a weakly augmented view while the student is trained on a strongly augmented view, the teacher predictions are naturally more stable and therefore serve as a more suitable reference for consistency regularization. In the early and middle stages of training, the teacher remains consistently stronger than the student, suggesting that the weakly augmented branch forms a stable decision boundary earlier. As training proceeds, the student gradually improves under this cross-augmentation consistency constraint, becomes comparable to the teacher in the later stage, and finally attains a slightly higher best accuracy. This phenomenon indicates that the consistency supervision constructed from the weakly augmented view provides an effective training constraint for the strongly augmented branch.



(a) Baseline (b) Ours (fine-tuned)

Figure 12: t-SNE visualization of clean test features for the baseline and the proposed fine-tuned model.

Feature-space visualization. Fig. 12 compares the clean-test feature distributions of the baseline and the proposed fine-tuned model on eight word classes, namely *become*, *happen*, *having*, *house*, *should*, *support*, *think*, and *times*. The visualized points are the feature representations right before the final classifier, so the figure reflects the intermediate discriminative structure learned by the model. In the baseline, the features are already roughly separable, but several classes still show relatively large intra-class spread, while the distances between some class centers remain limited. By contrast, the proposed model yields a more regular distribution: samples from the same class are clustered more tightly, and the gaps between different class clusters are also enlarged. This indicates that before entering the final classifier, the proposed model has already formed a more discriminative representation space with improved intra-class compactness and inter-class separability.