Bidirectional Temporal-Sensitive Adaptation for Generalized Zero-Shot Temporal Action Localization

Mingkui Tan, *Member, IEEE*, Yihao Qian, Yirui Wang, Runhao Zeng, *Member, IEEE*, Victor C. M. Leung, *Life Fellow, IEEE*, Xiping Hu, *Member, IEEE*

***Abstract*—Zero-shot temporal action localization (ZSTAL) aims to localize and recognize action categories unseen during training. However, it assumes that test videos contain only unseen classes, which is unrealistic in practice where seen and unseen actions naturally co-exist. To bridge this gap, we introduce generalized ZSTAL (GZS-TAL), where models trained only on seen classes must handle both seen and unseen ones during testing. This setting highlights a critical challenge: a static, frozen model cannot adapt to the mixed distributions encountered at test time. To address this issue, we propose a Temporal-Sensitive Adaptation (TSA) module that equips TAL models with the ability to update themselves during testing. The key intuition is to use temporal dependency prediction as a self-supervised signal: TSA introduces an online-updatable memory optimized to reconstruct features of preceding segments from the current one, thereby embedding temporal dependencies into parameters and reusing them for adaptation at test time. To further enhance temporal modeling, we extend TSA into a Bi-directional TSA (Bi-TSA) mechanism that performs prediction in both forward and backward directions. By simultaneously exploiting historical and future contexts, Bi-TSA improves long-range temporal representation and yields more accurate boundary localization. Extensive experiments on THUMOS14 and ActivityNet-1.3 demonstrate that our approach achieves significant improvements over state-of-the-art methods under the GZS-TAL setting, validating its effectiveness and generalization ability.**

***Index Terms*—Temporal Action Localization, Generalized Zero-Shot Learning, Test-Time Adaptation.**

I. Introduction

Temporal Action Localization (TAL) is a fundamental problem in video understanding, which requires detecting and classifying action instances in long untrimmed videos with precise start and end boundaries. TAL has attracted growing research interest due to its broad applications in domains such as anomaly detection [1]–[3], sports video analysis and retrieval [4]–[7], and autonomous driving [8]. Despite remarkable progress, most existing methods are developed under the closed-set assumption, where the model can only recognize a fixed set of categories predefined during training. This assumption severely restricts their applicability in realistic open environments, where novel actions frequently emerge and cannot be fully enumerated in advance.

Fig. 1: Demonstration of the differences between ZSTAL and GZS-TAL. During training time, the model has access to visual information and semantic space of seen classes. During testing time, under the zero-shot learning setting, the model is judged only on unseen classes; whereas under the generalized zero-shot setting, the model is evaluated on the full label space, and must simultaneously localize and classify actions from both seen and unseen classes.

To relax this constraint, researchers have explored **Zero-Shot Temporal Action Localization** (ZSTAL) [9]–[13], which aims to localize and recognize action categories unseen during training. While ZSTAL highlights the potential of transferring semantic knowledge to novel categories, it relies on an idealized assumption: test videos contain only unseen classes. In practice, however, videos typically contain a mixture of seen and unseen classes. This discrepancy limits the usefulness of ZSTAL in real-world scenarios.

To bridge this gap, we introduce and systematically study the task of **Generalized Zero-Shot Temporal Action Localization (GZS-TAL)**. Unlike ZSTAL, in this setting, the training set contains only seen classes, while the test set includes both seen and unseen classes simultaneously (Fig. 1). This design reflects real-world applications more faithfully, since deployed systems inevitably encounter new actions alongside familiar ones. However, it also raises a unique challenge: the model must not only retain reliable recognition of seen actions but also leverage knowledge learned from seen classes to

This work was supported by the National Natural Science Foundation of China (NSFC) (Grant No. 62202311), Guangdong Basic and Applied Basic Research Foundation (Grant No. 2023A1515011512), Key Scientific Research Project of the Department of Education of Guangdong Province (Grant No. 2024ZDZX3012), the Joint Funds of the National Natural Science Foundation of China (Grant No. U24A20327), Shenzhen Science and Technology Foundation under (Grant No. JCYJ20250604173210013). (*Corresponding Author: Runhao Zeng*)

Mingkui Tan and Yihao Qian are with the School of Software Engineering, South China University of Technology, Guangzhou, 510000, China. Yirui Wang is with the School of Computer Science and Technology, Beijing Institute of Technology, Beijing, 100081, China. Runhao Zeng, Victor C. M. Leung and Xiping Hu are with the Artificial Intelligence Research Institute, Shenzhen MSU-BIT University, and Guangdong-Hong Kong-Macao Joint Laboratory for Emotional Intelligence and Pervasive Computing, Shenzhen, 518172, China. (E-mail: mingkuitan@scut.edu.cn; 202421045609@mail.scut.edu.cn; 3220241460@bit.edu.cn; runhaozeng.cs@gmail.com; vleung@ieee.org; huxp@bit.edu.cn)

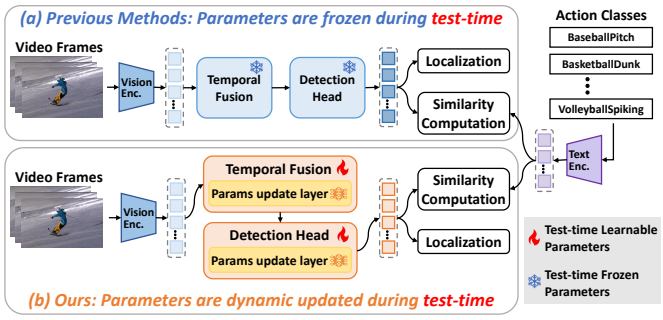


Fig. 2: Illustration of the (a) static and (b) adaptive Temporal Action Localization (TAL) paradigms during test-time. (a) depicts previous methods with parameters frozen during inference. In contrast, our approach (b) integrates parameter update layers into Temporal Fusion and Detection Head. This enables dynamic parameter updating during test-time, effectively transferring knowledge across the seen/unseen divide while ensuring robust performance on both categories.

recognize unseen ones. In other words, the central problem of GZS-TAL is how to transfer knowledge across the seen/unseen divide while ensuring robust performance on both.

Designing effective models for GZS-TAL naturally leads to the requirement of dynamic adaptation. A static model, trained once and frozen thereafter, cannot cope with the evolving test distributions where seen and unseen classes co-exist. Instead, the model should be capable of adjusting itself at test time to improve recognition of unseen actions while still preserving discriminative power on seen ones. Existing ZSTAL approaches, such as EffPrompt [9], STALE [10], and GAP [11], provide valuable insights and could in principle be modified for GZS-TAL. However, our empirical study shows that such extensions perform poorly, mainly because their models remain fixed during testing, as shown in Fig. 2(a). Without the ability to update themselves online, these methods cannot effectively adapt to the mixed distributions of seen and unseen classes required in GZS-TAL [14].

However, introducing dynamic adaptation is not straightforward and brings two new challenges: **1) Balancing adaptation and retention.** The model must dynamically incorporate unseen knowledge while avoiding catastrophic forgetting of seen classes. This trade-off is non-trivial, and our experiments explicitly evaluate forgetting to verify whether an approach can effectively mitigate it. **2) Unsupervised temporal modeling.** Since no annotations are available for unseen classes during testing, adaptation must be guided by the intrinsic temporal structure of videos. How to exploit temporal dynamics as a reliable self-supervised signal for model updating remains largely underexplored in TAL.

Inspired by recent advances in large language models (LLMs) [15]–[17], which demonstrate that self-supervised objectives can embed transferable knowledge into model parameters, and by the concept of Test-Time Training (TTT) [18], which updates model parameters during testing via a self-supervised loss, we extend these ideas to the temporal action localization domain. **1) To mitigate the problem of balancing adaptation and retention, we design a plug-and-play Temporal-Sensitive Adaptation (TSA) module.** TSA maintains an online-updatable parametric memory and is optimized by a specially designed temporal consistency loss, as shown in Fig. 2(b). This loss requires the model to reconstruct the features of preceding segments from the current one, thereby explicitly modeling temporal dependencies. By analogy to how LLMs acquire transferable knowledge through self-supervised training, our TSA enables TAL models to embed temporal knowledge into parameters during training and reuse it for dynamic adaptation at test time. In this way, the model can incorporate unseen knowledge while alleviating forgetting of seen classes. **2) To tackle the challenge of unsupervised temporal modeling, we further extend TSA into a Bi-directional TSA (Bi-TSA) mechanism.** By performing temporal prediction in both forward and backward directions, the model simultaneously exploits historical and future contexts. This bidirectional design strengthens temporal representations and improves the precision of action boundary localization, which are crucial for TAL in complex real-world videos. Extensive experiments on THUMOS14 and ActivityNet-1.3 demonstrate that, compared to state-of-the-art ZSTAL methods reproduced under the GZS-TAL setting, our framework achieves significant performance gains on both seen and unseen classes, showing superior generalization and robustness.

In summary, the main contributions of this paper include:

- We introduce the task of Generalized Zero-Shot Temporal Action Localization (GZS-TAL), which reflects realistic scenarios where seen and unseen classes co-exist at test time. Our benchmark and analysis establish GZS-TAL as a more practical and challenging direction beyond ZSTAL.
- To enable models to adapt dynamically in this new setting, we propose a plug-and-play Temporal-Sensitive Adaptation (TSA) module. Unlike prior frozen paradigms, TSA equips a trained TAL model with the ability to update itself during testing. With the help of a temporal consistency loss as a self-supervised signal, TSA transforms a static model into a dynamic one that can generalize to unseen actions without annotations.
- To further enhance temporal modeling without labels, we develop a Bi-directional TSA (Bi-TSA) mechanism. By introducing dual temporal prediction in both forward and backward directions, Bi-TSA exploits historical and future contexts simultaneously. This dual supervision significantly improves temporal representation quality and action boundary localization, bringing consistent gains under the GZS-TAL setting on THUMOS14 and ActivityNet-1.3.

II. Related Work

A. Zero-shot Temporal Action Localization

Video action analysis [19]–[22] has gained significant research attention. Within this field, Temporal Action Localization (TAL) detects and classifies action instances with precise temporal boundaries. Earlier works include two-stage

proposal–classification pipelines [23]–[26], one-stage anchor-free detectors [27]–[31], and query-based methods [32], [33], but all these methods, including certain weakly-supervised ones [34]–[36], assume a closed-set setting where training and test share the same classes. To relax this constraint, recent works extend TAL to Zero-Shot TAL (ZSTAL). Eff-Prompt [9] pioneered this direction by leveraging CLIP [37] to transfer knowledge to unseen classes. Subsequent works can be grouped into two families. Training-based methods [9]–[11], [13], [14], [38] train a TAL backbone on seen classes and then freeze it for inference. For instance, STALE [10] combines frame-level detectors and classifiers in a one-stage design. GAP [11] employs a query-based detector integrating static and dynamic cues, while mProTEA [13] tackles ZSTAL with multimodal prompt learning and text-enhanced actionness modeling. In contrast, training-free methods [12], [39] skip model training and adapt pre-trained vision–language models directly at test time. Examples include T3AL [12], which iteratively refines pseudo-labels frame by frame, and FreeZAD [39], which aggregates video-level semantics into prototypes. Although attractive, training-free approaches are generally less effective than training-based ones and remain applicable only in limited scenarios.

Overall, existing ZSTAL methods assume test videos contain only unseen classes, an unrealistic simplification. Moreover, their frozen inference paradigm prevents models from adapting to the mixed distributions of seen and unseen classes encountered in practice. This motivates our study of GZS-TAL and the development of adaptive mechanisms that can update models during inference.

B. Generalized Zero-shot Learning

The generalized zero-shot learning (GZSL) paradigm addresses the more realistic scenario where test sets contain both seen and unseen classes. The goal is to transfer knowledge from seen to unseen via semantic representations while maintaining performance on seen classes [40]. Two central challenges are: (i) ensuring reliable knowledge transfer, and (ii) mitigating the bias toward seen classes when both sets coexist. GZSL has been extensively studied in image classification [41], [42], with applications ranging from healthcare to cross-domain recognition. In video understanding, prior works have considered GZSL for video classification [43], [44] and action recognition [45]–[48]. For example, Hong et al. [44] generate fine-grained video–text features for unseen classes, while Huang et al. [47] employ a dual-GAN framework to synthesize unseen action features.

While GZSL has been actively explored in classification and recognition tasks, no prior work has extended it to the temporal localization setting. To the best of our knowledge, we are the first to define and systematically study Generalized Zero-Shot Temporal Action Localization (GZS-TAL), which requires simultaneous detection of seen and unseen actions.

C. Test-time Adaptation

Deep networks often degrade when deployed under distribution shifts. Test-Time Adaptation (TTA) addresses this by updating models online with lightweight objectives. For example, TENT [49] minimizes entropy by adjusting normalization layers. EATA [50] efficiently adapts models at test time by selectively updating on reliable samples, while SAR [51] uses sharpness-aware reliable entropy minimization. Beyond general classification, recent research has extended TTA to the video domain [52]–[54], addressing challenges such as video classification and video restoration. Although effective in these contexts, applying conventional TTA to TAL is non-trivial. Existing methods typically rely on indirect statistics or pseudo-labels, which fail to improve temporal dynamics modeling. A stronger paradigm is Test-Time Training (TTT) [18], [55], which uses self-supervised proxy tasks to update parameters during inference, with applications in segmentation [56] and video generation [57]. However, most existing video TTT methods [58] are uni-directional, ignoring future context that is crucial for precise temporal localization.

We address these gaps with the proposed Bi-directional Temporal-Sensitive Adaptation (Bi-TSA). Unlike prior frozen ZSTAL or uni-directional TTT, Bi-TSA enables test-time adaptation through a self-supervised reconstruction objective while processing clips in both forward and backward orders. This design allows the model to continuously update itself with temporal dependencies, exploit both past and future cues, and better handle the co-existence of seen and unseen classes in GZS-TAL.

III. Proposed Method

A. Problem Definition

The temporal action localization (TAL) task aims to detect and classify action instances in long untrimmed videos. Given an untrimmed video V, we represent it as a sequence of feature vectors $X = \{x_1, x_2, \ldots, x_T\}$ over discrete time steps $t = 1, \ldots, T$, where the length T varies across videos. Each video is annotated with a set of action instances

$$Y_{\text{gt}} = \{(s_i, e_i, a_i)\}_{i=1}^{N_{\text{gt}}}. \tag{1}$$

where s_i and e_i denote the start and end timestamps, $a_i \in D$ is the action category, D is the action category space and N_{gt} is the number of ground-truth instances. The goal of TAL is to predict a set of instances

$$Y_{\text{pred}} = \{(\widetilde{s}_i, \widetilde{e}_i, \widetilde{a}_i)\}_{i=1}^{N_{\text{pred}}}. \tag{2}$$

where $\widetilde{a}_i \in D$, ensuring both precise temporal boundaries and correct category assignments.

In this paper, we focus on a more realistic extension of this task—**Generalized Zero-Shot Temporal Action Localization (GZS-TAL)**—where the test phase involves both seen and unseen classes. Specifically, the category space D is partitioned into two disjoint subsets: seen classes D_{seen} and unseen classes D_{unseen}, such that $D_{\text{seen}} \cup D_{\text{unseen}} = D$ and $D_{\text{seen}} \cap D_{\text{unseen}} = \emptyset$. During training, supervision is restricted to categories in D_{seen}; that is, the annotated label space is $D_{\text{train}} = D_{\text{seen}}$ and no positive supervision is provided for D_{unseen}. We make no assumption about whether instances from D_{unseen} physically occur in the training videos; if present, they are unlabeled and thus provide no class-specific supervision.

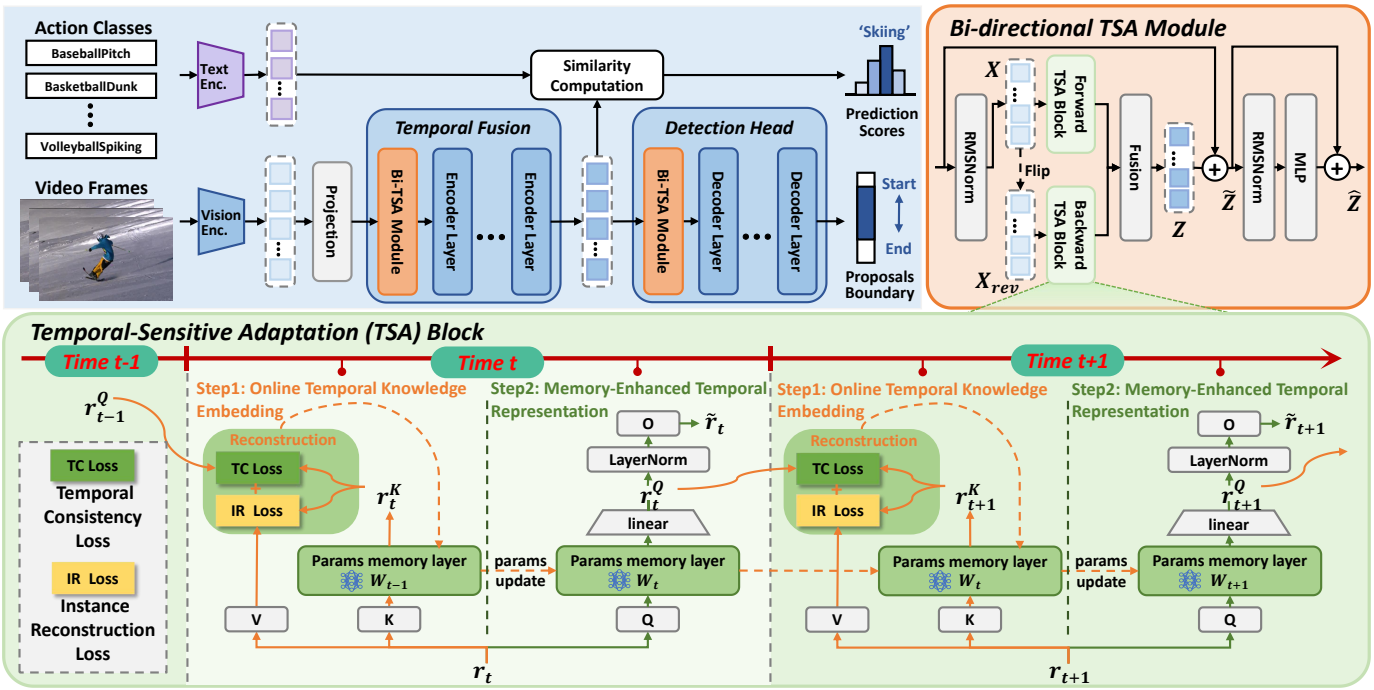


Fig. 3: Overview of the proposed framework with Bi-TSA for GZS-TAL. The framework (Sec. III-E) incorporates the Bi-directional TSA (Bi-TSA) module (Sec. III-D), which is integrated within both the Temporal Fusion and the Detection Head. Within the Bi-TSA module, the Temporal-Sensitive Adaptation (TSA) Block (Sec. III-C) processes forward and backward fragment sequences in parallel. The TSA Block updates the parameters W through self-supervised reconstruction loss to memorize the current input r_t, and then uses historical information to make predictions.

During testing, the model is evaluated on the full label space, i.e., $D_{\text{test}} = D_{\text{seen}} \cup D_{\text{unseen}}$, and must simultaneously localize and classify actions from both sets.

B. Method Overview

The GZS-TAL setting is considerably more challenging than standard ZSTAL because the model must retain its discriminative ability on seen classes while simultaneously adapting to previously unseen ones. A static network, trained once and frozen thereafter, is fundamentally inadequate for such scenarios. Instead, the model must acquire the ability to update itself dynamically at test time to cope with evolving input distributions.

To address this, we propose a plug-and-play **Temporal-Sensitive Adaptation (TSA)** block (Sec. III-C), which can be seamlessly inserted into the temporal modeling layers of existing TAL frameworks [10], [11], [27]. The central idea of the TSA block is to replace the static temporal representation cache with a parametric memory layer W that is continuously updated during both training and inference via a carefully designed self-supervised learning objective. This update mechanism enables the model to (i) learn how to adapt during training, and (ii) perform actual adaptation during testing without requiring labels.

To capture full action context and sharpen boundaries, we further introduce **Bi-directional TSA (Bi-TSA)** (Sec. III-D): two TSA streams process forward ($r_1 \to r_N$) and backward ($r_N \to r_1$) clip orders in parallel and fuse their outputs, so each time step benefits from both historical and future cues. In this way, our framework directly addresses the problem of test-time adaptation under GZS-TAL, embodies the core idea of self-supervised parametric memory updated online, and delivers a robust architecture with enhanced temporal fusion and detection capabilities through dynamic parameter updates and bidirectional temporal supervision.

C. Temporal-Sensitive Adaptation (TSA) Block

Conventional TAL models suffer from two fundamental limitations when applied to the GZS-TAL setting. First, they rely on static representations. Once training is complete, model parameters are frozen, preventing any adaptation to unseen classes during testing. Second, their temporal modeling capacity is inherently constrained. Transformer-based approaches preserve history in key–value caches with quadratic cost in sequence length, which are computationally demanding and unsuitable for online updating. RNN-based approaches, on the other hand, compress temporal context into low-dimensional hidden vectors, which lack expressiveness and cannot be effectively optimized at test time. As a result, both paradigms are unable to capture long-term dependencies while remaining adaptive in deployment.

To overcome these issues, we propose the **Temporal-Sensitive Adaptation (TSA)** block. TSA introduces a *parametric memory layer* that serves as a dynamic hidden state. Instead of storing context explicitly (as in Transformers) or compressing it into limited vectors (as in RNNs), TSA embeds

temporal knowledge into model parameters through continuous online updating—where "online" refers to the timing and mechanism of parameter updates during test-time [49]–[51], rather than to a strict streaming data access pattern. At each time step, the parametric memory is optimized by a self-supervised temporal objective, enabling the model to internalize temporal dependencies into its parameters. This design is both efficient and expressive: memory updates are lightweight, yet they progressively accumulate temporal knowledge. The intuition is analogous to large language models, which acquire transferable knowledge through self-supervised pretraining—the difference is that TSA performs such updates online, allowing TAL models to dynamically adapt under the GZS-TAL setting.

Step 1: Online Temporal Knowledge Embedding. Let the feature sequence of a video be $X = \{x_1, x_2, \ldots, x_T\}$, which is partitioned into non-overlapping clips $\{r_t\}_{t=1}^{N}$. At time step t, given the current input clip r_t, the TSA block updates its parametric memory W, represented as the weights of a small network f. Specifically, W_{t-1} encodes the historical context up to step $t-1$, and is updated into W_t by minimizing a self-supervised reconstruction loss:

$$W_t = W_{t-1} - \eta_t \cdot \nabla_{W_{t-1}} \mathcal{L}_{\text{recon}}(r_t; W_{t-1}). \tag{3}$$

where the learning rate $\eta_t = \sigma(W_{\text{lr}} \cdot x_t)$ is dynamically adapted to the current input. The reconstruction loss is defined as a weighted combination of two complementary terms:

$$\mathcal{L}_{\text{recon}}(r_t; W_{t-1}) = \alpha \cdot \mathcal{L}_{\text{inst}}(r_t; W_{t-1}) + \beta \cdot \mathcal{L}_{\text{temp}}(r_t; W_{t-1}). \tag{4}$$

The **Instance Reconstruction Loss** $\mathcal{L}_{\text{inst}}$ ensures that the model extracts useful information from the current input by reconstructing its projected features:

$$\mathcal{L}_{\text{inst}}(r_t; W_{t-1}) = \left| f(\theta_K r_t; W_{t-1}) - \theta_V r_t \right|^2. \tag{5}$$

where θ_K and θ_V are learnable projections. The **Temporal Consistency Loss** $\mathcal{L}_{\text{temp}}$ enforces explicit temporal dependency by reconstructing the features of the previous step from the current input:

$$\mathcal{L}_{\text{temp}}(r_t; W_{t-1}) = \left| f(\theta_K r_t; W_{t-1}) - r_{t-1}^{Q} \right|^2. \tag{6}$$

where r_{t-1}^{Q} denotes the unnormalized output of the TSA block at step $t-1$. Through these dual objectives, the parametric memory W integrates both current information and temporal continuity, progressively embedding temporal knowledge into its parameters. This enables the model to update itself online, a property crucial for adapting to unseen classes in the GZS-TAL setting.

Step 2: Memory-Enhanced Temporal Representation. Once the parameters have been updated to W_t, the TSA block uses them to generate a feature representation for the current input segment r_t. The process proceeds in two stages. First, the input is projected into the query space through a learnable matrix θ_Q, passed through the network $f(\cdot; W_t)$ parameterized by the updated memory, and mapped by a linear layer L to obtain the raw intermediate features:

$$r_t^{Q} = L(f(\theta_Q r_t; W_t)), \tag{7}$$

These raw features r_t^{Q} capture both the current input and the temporal context encoded in W_t. Importantly, they are also reused in the reconstruction objective to enforce temporal consistency across adjacent segments, thereby guiding the self-supervised update of the parametric memory. Second, to obtain the final representation for downstream action localization, the raw features are normalized and further projected:

$$\tilde{r}_t = \theta_O \cdot LN(r_t^{Q}). \tag{8}$$

where LN denotes Layer Normalization and θ_O is a learnable projection matrix. The final output $\tilde{r}_t$ thus integrates information from the current segment with accumulated temporal knowledge in the parametric memory, producing a memory-enhanced representation that serves as the basis for robust action classification and precise boundary localization under GZS-TAL.

D. Bi-directional TSA Module

While the TSA block introduces dynamic online updates, its temporal modeling remains essentially unidirectional, relying only on past context to guide adaptation. This limitation is particularly problematic under GZS-TAL, where unseen actions may exhibit complex and unfamiliar dynamics. In such cases, access to both historical and future context is crucial for precise boundary localization and robust representation learning. To alleviate this limitation, we design a *Bi-directional TSA (Bi-TSA)* that processes the sequence in forward and backward directions and fuses the two streams at matched time indices.

Forward stream. Without loss of generality, let $X = \{x_1, \ldots, x_T\} \in \mathbb{R}^{T \times C}$ denote the time-ordered feature stream at the TSA insertion point (TSA is plug-and-play and may be placed at any temporal layer; here X simply refers to the features fed to that module). We partition X into sequential segments $\{r_t\}_{t=1}^{N}$ and feed them in order to a forward TSA block. The segment-level outputs are concatenated chronologically to produce $Z_{\text{fw}} \in \mathbb{R}^{T \times C}$, preserving one-to-one alignment with the original timeline.

Backward stream. To exploit reverse dependencies, we construct a temporally flipped view by reversing only the valid frames of X (padding, if present, is masked and left unchanged), yielding $X_{\text{rev}} = \{x_T, \ldots, x_1\}$. Using the *same* segmentation scheme, the segments of X_{rev} are processed sequentially by a backward TSA block. The resulting outputs are then restored to chronological order, giving $Z_{\text{bw}} \in \mathbb{R}^{T \times C}$ that is index-aligned with Z_{fw} at every time step.

Bidirectional Fusion and Post-Fusion Refinement. After obtaining the forward representation Z_{fw} and the backward representation Z_{bw}, we perform feature-level fusion to construct a unified temporal representation. Specifically, we compute the element-wise mean:

$$Z = \tfrac{1}{2}(Z_{\text{fw}} + Z_{\text{bw}}), \tag{9}$$

which leverages the inherent symmetry of the two directions while avoiding directional bias.

To further enhance representational quality, the fused feature Z is integrated with the original input sequence X through a

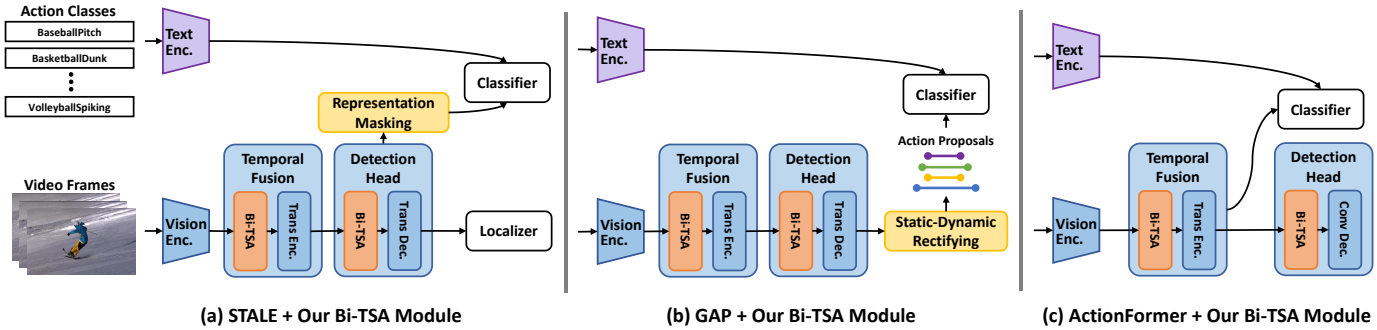


Fig. 4: Illustration of our Bi-TSA module's application to different Temporal Action Localization (TAL) frameworks. Bi-TSA (orange) is incorporated into the Temporal Fusion and Detection Head modules of (a) STALE+Ours, (b) GAP+Ours, and (c) ActionFormer+Ours (with a convolutional decoder).

residual connection: $\tilde{Z} = X + Z$. The residual path ensures preservation of the raw temporal cues from X. The output $\tilde{Z}$ is then normalized via Root Mean Square Normalization (RMSNorm) [59] and passed through a Multi-Layer Perceptron (MLP) to refine temporal semantics. A second residual connection adds the MLP output back to its input:

$$\hat{Z} = \text{MLP}(\text{RMSNorm}(\tilde{Z})) + \tilde{Z}. \tag{10}$$

This dual-residual design enhances gradient propagation, mitigates information degradation, and yields the final bidirectionally enriched representation $\hat{Z}$, which is passed to subsequent modules.

E. Integration of Bi-TSA into Existing TAL Frameworks

The proposed Bi-TSA is a plug-and-play adaptor that can be seamlessly integrated into existing TAL frameworks without modifying their overall structure. In practice, Bi-TSA can be embedded at two key stages: (i) temporal fusion (TF) modules, where sequential features are aggregated into higher-level representations, and (ii) detection heads (DH), where action classification and boundary regression are performed. As illustrated in Fig. 4, these two integration paradigms are instantiated for both two-stage [11] and one-stage [10], [27] TAL frameworks.

Integration into temporal fusion modules. Many TAL methods (e.g., transformer-based models such as [10], [11], [27]) include temporal fusion stages that progressively combine local clip-level features into contextual representations, which range from single-scale (as integrated into STALE [10] and GAP [11], shown in Fig. 4(a)(b)) to multi-scale (as integrated into ActionFormer [27], shown in Fig. 4(c)) hierarchies.To enhance this process under the GZS-TAL setting, we embed Bi-TSA at the entry of such modules, typically before the first temporal block. This design allows Bi-TSA to refine fine-grained local dependencies while later layers focus on semantic abstraction.

To support this, we adopt a streaming input design. Given a video feature sequence $X = \{x_1, \ldots, x_T\}$, we partition it into N non-overlapping segments $\{r_t\}_{t=1}^{N}$ of length L_W. These segments are fed sequentially into Bi-TSA, which incrementally updates its parametric memory. This streaming scheme reduces computational overhead, ensures temporal continuity, and provides natural steps for test-time adaptation. By embedding Bi-TSA in the fusion stage, the backbone gains bidirectional, adaptive temporal modeling, substantially improving its ability to handle unseen actions.

Integration into detection heads. In addition, Bi-TSA can be inserted into detection heads that directly predict action categories and temporal boundaries. For classification branches (as integrated into STALE [10] and GAP [11], shown in Fig. 4(a)(b)), Bi-TSA refines temporal features before alignment with text embeddings, improving discriminability between seen and unseen classes. For regression branches (as integrated into ActionFormer [27], shown in Fig. 4(c)), it enriches proposal-level features with bidirectional context, enhancing boundary precision. This insertion is lightweight and requires no change to the detection head's original objective.

Summary. Through integration into temporal fusion modules and detection heads, Bi-TSA equips existing TAL frameworks with online, bidirectional temporal adaptation, enabling them to localize both seen and unseen actions more robustly under the challenging GZS-TAL setting.

F. Training and Inference Details

Training. Our framework employs a dual-loop training strategy that jointly captures global semantic objectives and local temporal adaptability. This design is conceptually related to meta-learning, where an inner loop adapts to sample-specific information while an outer loop enforces global generalization. Specifically:

1) Inner Loop (Local Adaptation). At the video-segment level, the Bi-TSA module updates its parametric memory through gradient descent on the self-supervised reconstruction loss. This allows the model to embed fine-grained temporal dependencies into its evolving parameters. To avoid overfitting and ensure stability, the updated memory is reset after each video, ensuring that the adaptation remains sample-specific rather than accumulating across videos.

2) Outer Loop (Global Optimization). At the full-video level, the backbone network is optimized end-to-end following the standard TAL training objectives, including classification and boundary regression. This loop enforces semantic consistency and discriminative capacity across seen classes,

complementing the temporal refinements introduced by the inner loop.

3) Synergy of Dual Loops. The inner loop equips the model with locally adaptive temporal modeling, while the outer loop ensures global task alignment. Their synergy enables multi-granularity temporal learning, improving both generalization to unseen classes and localization accuracy.

Inference. At test time, our framework diverges from prior TAL methods [9]–[11], which rely on fixed parameters after training. Instead, Bi-TSA dynamically updates its parametric memory online, guided by the same reconstruction loss (Eqn. 4). This enables the model to adapt to unseen classes while retaining performance on seen ones. As in training, the memory is reset after each video, preventing cross-video drift and ensuring adaptation remains sample-specific. Through this mechanism, our model achieves robust test-time adaptation, delivering improved temporal precision and stronger generalization under GZS-TAL.

IV. Experiments

A. Datasets

THUMOS14 [60] contains 20 action classes, with 200 validation and 213 test videos. Each video includes on average 15.5 action instances of varying duration. Following prior works [10], [11], [27], we train on the validation set and evaluate on the test set.

ActivityNet-1.3 [61] is a large-scale dataset encompassing 200 action classes, with the training set containing 10,024 videos, the validation set containing 4,926 videos, and the test set containing 5,044 videos. It contains around 1.5 action instances per video. As the ground-truth labels for the test set are not publicly available, consistent with prior studies, we employ the videos from the training set for training and those from the validation set for inference.

For the GZS-TAL task, we adopt the seen-unseen class split strategy from ZSTAL [9]–[11], dividing the data according to 75%-seen 25%-unseen and a 50%-seen 50%-unseen split setting. During the training phase, like ZSTAL, we train the model using the training data that includes only seen classes. Differently, in the inference phase, we evaluate the model using the entire test data, comprising both seen and unseen classes.

B. Evaluation Metrics

Following prior work [10], [11], [27], we report the mean average precision (mAP) at multiple temporal intersection over union (tIoU) thresholds for all datasets, as mAP is the standard evaluation metric for TAL task. Specifically, for a given tIoU threshold, the mAP is calculated as the mean precision across all action category predictions. The average mAP is computed as the mean of mAPs over several tIoU thresholds. For evaluation, to ensure a fair comparison with each baseline method, we adhere to the specific evaluation protocols used in their original papers. Following prior work [9], [10], we use tIoU settings of [0.3:0.1:0.7] for THUMOS14 and [0.5:0.05:0.95] for ActivityNet-1.3.

C. Implementation Details

Video and Text Feature Extractor. We adopt the pre-trained CLIP [37] (ViT/B-16) encoders to extract video and text features with dimension $C = 512$. We also report results using InternVideo [62] (InternVideo-MM-L-14) encoders with $C = 768$, which provide stronger video representations through masked video modeling and video-text pretraining. For THUMOS14, video features are extracted from 8-frame segments with a stride of 8; for ActivityNet-1.3, we use 8-frame segments with a stride of 16.

Integration with existing TAL methods. We integrate our Bi-TSA module into three representative TAL frameworks. STALE [10] and GAP [11], originally designed for ZSTAL, can be directly adapted to GZS-TAL by adjusting the label space during inference. ActionFormer [27], though not designed for open-set scenarios, is extended by replacing its classification head with a vision-language contrastive head based on CLIP or InternVideo encoders. For STALE, we follow [10] and train on ActivityNet-1.3 using Adam with a learning rate of 1×10^{-5} for 15 epochs. Its THUMOS14 implementation is not publicly available, thus results on that dataset are not reported. For GAP, we follow [11] and train with AdamW on both datasets for 100 epochs, with learning rates 1×10^{-4} (THUMOS14) and 5×10^{-5} (ActivityNet-1.3). For ActionFormer, we follow [27], training with Adam and warm-up. On THUMOS14, we train for 100 epochs with learning rate 1×10^{-4}; on ActivityNet-1.3, we train for 40 epochs with learning rate 1×10^{-3}. For the Bi-TSA module, the reconstruction loss coefficients are set to $\alpha = 0.8$ and $\beta = 0.2$. Furthermore, the Bi-TSA module's parametric memory W, implemented as a small MLP network, was initialized with weights from a normal distribution and zero biases across all integrated frameworks. All experiments are run on a single NVIDIA RTX A800 GPU.

D. Comparison with the State-of-the-Arts

Since no prior work explicitly addresses the GZS-TAL setting, we adapt representative ZSTAL methods (STALE [10], GAP [11], and ActionFormer [27] variants) as baselines and integrate our Bi-TSA into them. For fairness, we strictly follow the original configurations and evaluation protocols of each baseline, and all baseline results are reproduced under the GZS-TAL protocol.

*1) **Results with CLIP Encoder**:* Table I reports results using CLIP as the vision-language encoder. Across all datasets and split settings, integrating TSA in isolation consistently improves performance over the baselines, while incorporating our full Bi-TSA module yields further gains. On the **THUMOS14** dataset, under the 75%-25% split, ActionFormer+TSA achieves a +1.0% gain in average mAP (27.4% $\rightarrow$ 28.4%), while GAP+TSA improves by +1.2% (24.0% $\rightarrow$ 25.2%). This improvement trend is maintained under the more challenging 50%-50% split, where GAP+TSA surpasses the baseline by +0.8% (21.3% $\rightarrow$ 22.1%). On the **ActivityNet-1.3** dataset, our TSA module shows similar effectiveness. Under the 75%-25% split, ActionFormer+TSA and GAP+TSA reach 21.7% and 24.3% average mAP, respectively, both

TABLE I: Generalized Zero-Shot Temporal Action Localization performance (mAP in %) on THUMOS14 and ActivityNet-1.3. Our proposed Bi-TSA module is applied to three baseline methods under two different encoders. The notation '+TSA' indicates the performance with only our TSA module integrated, while '+Bi-TSA' indicates the performance with our Bi-TSA module integrated. '-' indicates that results are not reported for STALE on THUMOS14 due to the lack of an official implementation for this dataset.

Split	Encoder	Methods	THUMOS14						ActivityNet-1.3			
			0.3	0.4	0.5	0.6	0.7	AVG	0.5	0.75	0.95	AVG
75% Seen 25% Unseen	CLIP	STALE	-	-	-	-	-	-	39.0	23.0	3.8	23.2
		STALE+TSA	-	-	-	-	-	-	39.3	23.5	4.0	23.7
		STALE+Bi-TSA	-	-	-	-	-	-	**39.9**	**24.1**	**4.6**	**24.3**
		GAP	39.0	33.2	24.0	15.2	8.6	24.0	35.4	24.4	5.2	23.6
		GAP+TSA	**40.9**	33.5	25.0	**16.8**	8.9	25.0	36.3	24.1	5.6	23.7
		GAP+Bi-TSA	40.6	**33.6**	**25.3**	16.6	**9.6**	**25.2**	**36.6**	**25.1**	**5.8**	**24.3**
		ActionFormer	40.1	35.7	28.5	21.1	11.6	27.4	31.9	19.9	3.2	19.6
		ActionFormer+TSA	40.3	35.4	**29.3**	21.9	**14.2**	28.2	32.1	20.7	3.8	20.4
		ActionFormer+Bi-TSA	**41.1**	**36.6**	29.2	**22.1**	13.2	**28.4**	**34.0**	**22.1**	**4.2**	**21.7**
	InternVideo	STALE	-	-	-	-	-	-	41.0	23.2	4.0	23.8
		STALE+TSA	-	-	-	-	-	-	41.2	24.3	3.9	24.5
		STALE+Bi-TSA	-	-	-	-	-	-	**41.2**	**24.8**	**4.2**	**24.6**
		GAP	38.3	32.9	25.2	16.9	8.5	24.4	39.8	27.7	4.7	26.6
		GAP+TSA	38.3	32.2	25.2	16.7	9.4	24.4	39.7	27.1	**7.4**	26.7
		GAP+Bi-TSA	**39.5**	**33.4**	**25.4**	**17.2**	**9.8**	**25.0**	**40.6**	**27.7**	5.7	**27.1**
		ActionFormer	39.4	35.3	29.7	22.7	14.3	28.3	31.9	20.4	**4.8**	20.7
		ActionFormer+TSA	40.0	25.9	30.2	23.2	15.7	29.0	33.9	21.8	4.3	21.4
		ActionFormer+Bi-TSA	**41.6**	**37.2**	**31.0**	**23.7**	**15.7**	**29.8**	**35.0**	**22.1**	4.7	**22.0**
50% Seen 50% Unseen	CLIP	STALE	-	-	-	-	-	-	39.7	22.8	3.6	23.0
		STALE+TSA	-	-	-	-	-	-	38.6	22.4	**4.5**	23.2
		STALE+Bi-TSA	-	-	-	-	-	-	**39.2**	**23.3**	3.7	**23.3**
		GAP	36.2	28.3	21.2	13.7	6.8	21.3	34.1	23.3	5.0	22.7
		GAP+TSA	35.8	29.6	22.1	**14.8**	**8.5**	**22.2**	**36.0**	24.1	4.3	23.3
		GAP+Bi-TSA	**36.7**	**30.2**	**22.2**	14.4	7.1	22.1	35.5	**24.6**	**5.0**	**23.8**
		ActionFormer	29.4	26.6	22.3	17.2	10.4	21.1	18.4	11.3	0.9	11.4
		ActionFormer+TSA	29.2	26.4	22.9	17.6	11.2	21.5	19.7	12.2	2.6	12.2
		ActionFormer+Bi-TSA	**30.1**	**27.6**	**23.5**	**18.2**	**11.3**	**22.1**	**21.1**	**13.5**	**2.7**	**13.4**
	InternVideo	STALE	-	-	-	-	-	-	40.1	22.0	4.2	22.9
		STALE+TSA	-	-	-	-	-	-	38.5	**23.0**	**5.0**	23.3
		STALE+Bi-TSA	-	-	-	-	-	-	**39.0**	22.7	4.4	**23.5**
		GAP	32.7	27.4	21.0	13.9	7.6	20.5	39.7	26.1	6.2	25.9
		GAP+TSA	**35.0**	**28.3**	21.0	13.8	7.5	21.1	**39.4**	26.5	**7.0**	26.1
		GAP+Bi-TSA	34.1	28.2	**21.5**	**14.0**	**8.0**	**21.2**	39.2	**27.2**	5.7	**26.4**
		ActionFormer	28.7	26.1	22.4	17.4	11.5	21.2	20.9	13.2	2.8	13.3
		ActionFormer+TSA	28.5	26.1	21.9	17.1	**11.7**	21.1	22.1	14.0	2.9	14.0
		ActionFormer+Bi-TSA	**29.3**	**26.8**	**23.0**	**17.6**	11.0	**21.5**	**22.6**	**14.6**	**2.9**	**14.3**

outperforming their original versions. Due to the lack of an official configuration for THUMOS14, STALE cannot be directly compared on this dataset. Nevertheless, its gains on ActivityNet-1.3 further validate the general effectiveness of our approach. These results indicate that TSA, as a versatile plug-and-play adaptation module, effectively compensates for the temporal modeling limitations of image-centric encoders and synergizes with advanced video foundation models, consistently advancing state-of-the-art performance in the challenging GZS-TAL setting, with the bidirectional mechanism further strengthening its long-range temporal modeling.

*2) **Results with InternVideo Encoder**:* To further examine the generality of TSA, we replace CLIP with InternVideo, a stronger video-language foundation model pre-trained with large-scale video-text data. InternVideo provides richer temporal dynamics than CLIP, and our TSA module continues to deliver additional improvements. A similar trend of improvement is observed when using the CLIP encoder: integrating TSA in isolation generally outperforms the baseline, and Bi-TSA further improves performance. As shown in Table I, on **THUMOS14** under the 75%-25% split, ActionFormer+TSA improves average mAP from 28.3% to 29.8%, achieving a new state-of-the-art. GAP+TSA also gains +1.4%. On **ActivityNet-1.3**, under the most challenging 50%-50% split, TSA enhances STALE from 22.9% to 23.5%. These results confirm that the TSA module learns robust temporal representations through dynamic and self-supervised adaptation, while bidirectional modeling provides complementary advantages by capturing both historical and future contexts, thereby yielding more stable temporal representations and more precise boundary

TABLE II: Comparison with several representative TTA methods on THUMOS14 and ActivityNet-1.3 datasets using ActionFormer. '+Ours' indicates the performance with our Bi-TSA module integrated.

Split	Encoder	Methods	THUMOS14						ActivityNet-1.3			
			0.3	0.4	0.5	0.6	0.7	AVG	0.5	0.75	0.95	AVG
75% Seen 25% Unseen	CLIP	ActionFormer	40.1	35.7	28.5	21.1	11.6	27.4	31.9	19.9	3.2	19.6
		ActionFormer+TENT [49]	39.3	35.0	29.0	21.6	13.0	27.6	32.0	19.5	3.0	19.7
		ActionFormer+EATA [50]	39.1	34.6	28.6	21.2	12.8	27.3	32.1	19.6	3.1	19.9
		ActionFormer+SAR [51]	39.0	34.8	28.6	20.9	12.4	27.1	31.5	19.1	3.2	19.4
		ActionFormer+Ours	**41.1**	**36.6**	**29.2**	**22.1**	**13.2**	**28.4**	**34.0**	**22.1**	**4.2**	**21.7**
	InternVideo	ActionFormer	39.4	35.3	29.7	22.7	14.3	28.3	31.9	20.4	**4.8**	20.7
		ActionFormer+TENT [49]	39.2	35.3	29.6	23.3	16.2	28.7	33.9	20.9	3.3	20.8
		ActionFormer+EATA [50]	39.0	35.2	29.5	23.2	**16.3**	28.7	34.5	21.5	3.3	21.3
		ActionFormer+SAR [51]	38.0	34.3	28.9	22.2	15.8	27.9	33.0	20.5	3.4	20.4
		ActionFormer+Ours	**41.6**	**37.2**	**31.0**	**23.7**	15.7	**29.8**	**35.0**	**22.1**	4.7	**22.0**
50% Seen 50% Unseen	CLIP	ActionFormer	29.4	26.6	22.3	17.2	10.4	21.1	18.4	11.3	0.9	11.4
		ActionFormer+TENT [49]	29.5	26.9	22.8	17.5	10.9	21.5	20.6	12.1	1.7	12.4
		ActionFormer+EATA [50]	28.7	26.1	22.2	17.1	10.5	20.9	20.0	12.2	1.7	12.2
		ActionFormer+SAR [51]	28.4	26.1	21.5	16.2	9.7	20.4	19.8	11.9	1.6	12.1
		ActionFormer+Ours	**30.1**	**27.6**	**23.5**	**18.2**	**11.3**	**22.1**	**21.1**	**13.5**	**2.7**	**13.4**
	InternVideo	ActionFormer	28.7	26.1	22.4	17.4	**11.5**	21.2	20.9	13.2	2.8	13.3
		ActionFormer+TENT [49]	29.2	26.7	22.1	17.3	11.2	21.3	21.5	12.7	1.8	13.0
		ActionFormer+EATA [50]	28.5	26.1	22.0	17.1	11.4	21.0	21.9	13.0	1.8	13.3
		ActionFormer+SAR [51]	27.6	25.1	21.6	16.9	11.3	20.5	20.9	12.9	1.8	13.0
		ActionFormer+Ours	**29.3**	**26.8**	**23.0**	**17.6**	11.0	**21.5**	**22.6**	**14.6**	**2.9**	**14.3**

localization.

*3) **Comparison with TTA methods**:* To comprehensively evaluate the performance of our method relative to existing Test-Time Adaptation (TTA) paradigms, we conducted extensive comparative experiments. We integrated three widely-used and representative general TTA methods (TENT [49], EATA [50], and SAR [51]) into the ActionFormer backbone. The results in Table II consistently show that these general TTA methods yield only marginal and often unstable performance changes compared to the baseline. For example, on ActivityNet-1.3 with the InternVideo encoder, EATA improved the average mAP by only within 0.5%, while TENT and SAR even led to performance degradation. In contrast, our Bi-TSA achieved a stable, significant gain of 1.3% under the same conditions. Similar trends were observed across all dataset configurations.

We attribute this to a fundamental difference in adaptation objectives: general TTA methods such as TENT are primarily designed for image classification and rely on entropy minimization or pseudo-labeling. These approaches do not explicitly model the temporal dependencies that are crucial for TAL. Conversely, our proposed temporal consistency loss explicitly enforces cross-segment consistency and reconstructs historical context, enabling the model to learn transferable temporal knowledge that is vital for adapting to unseen actions and refining boundary localization.

*4) **Comparison with distillation-based methods**:* To further investigate the distinction between our method and classical knowledge distillation [63]–[65], we conduct additional experiments under the GZS-TAL setting. We introduce two comparative variants: **1) +TTA-KD.** For a fair comparison, following references [66], [67], we integrate Knowledge Distillation with Test-Time Adaptation. Specifically, the trained model serves as both the frozen Teacher model and the initial state of the updatable Student model. During test-time, the Student model is updated online by minimizing a composite loss that combines a knowledge distillation loss (using KL-divergence to align classification with the Teacher and GIoU loss for boundary regression alignment [68]) and the entropy minimization loss from TENT [49]. **2) +Ours (KL).** To explore connections with knowledge distillation, we replace the L2 form of $\mathcal{L}_{\text{temp}}$ in our Bi-TSA module with its KL-divergence counterpart, which is a representative loss function in knowledge distillation.

As the results shown in Table III, the +TTA-KD method yields marginal improvements, often hovering around the Baseline. This indicates that while TTA-KD implicitly uses temporal information through classification and boundary alignment, it does not explicitly model temporal dependencies. In contrast, our TSA module explicitly embeds temporal continuity into parameters via a self-supervised reconstruction of preceding segments, thereby directly and more strongly leveraging the temporal structure of videos for online adaptation. This allows TSA to better handle the dynamic characteristics of unseen actions. Furthermore, both the +Ours (KL) variant and our original method significantly outperform the Baseline. Notably, +Ours (KL) achieves slightly better performance than the original method in some settings (e.g., 22.0% → 23.3% on

TABLE III: Comparison with distillation-based methods on THUMOS14 and ActivityNet-1.3 datasets using ActionFormer.

Split	Encoder	Methods	THUMOS14						ActivityNet-1.3			
			0.3	0.4	0.5	0.6	0.7	AVG	0.5	0.75	0.95	AVG
75% Seen 25% Unseen	CLIP	ActionFormer	40.1	35.7	28.5	21.1	11.6	27.4	31.9	19.9	3.2	19.6
		ActionFormer+TTA-KD	38.9	35.0	28.6	21.4	13.0	27.4	31.4	19.0	3.2	19.3
		ActionFormer+Ours (KL)	40.2	35.1	28.8	21.9	**13.5**	27.9	**34.4**	22.0	4.2	**22.0**
		ActionFormer+Ours (L2)	**41.1**	**36.6**	**29.2**	**22.1**	13.2	**28.4**	34.0	**22.1**	**4.2**	21.7
	InternVideo	ActionFormer	39.4	35.3	29.7	22.7	14.3	28.3	31.9	20.4	4.8	20.7
		ActionFormer+TTA-KD	39.2	35.3	29.6	23.3	**16.2**	28.7	34.0	21.0	3.3	20.9
		ActionFormer+Ours (KL)	41.1	36.3	29.0	22.7	15.0	28.8	**36.6**	**23.5**	**4.8**	**23.3**
		ActionFormer+Ours (L2)	**41.6**	**37.2**	**31.0**	**23.7**	15.7	**29.8**	35.0	22.1	4.7	22.0
50% Seen 50% Unseen	CLIP	ActionFormer	29.4	26.6	22.3	17.2	10.4	21.1	18.4	11.3	0.9	11.4
		ActionFormer+TTA-KD	29.0	26.5	22.5	17.3	10.7	21.2	19.8	12.1	1.7	12.1
		ActionFormer+Ours (KL)	29.6	27.0	23.1	17.8	11.1	21.7	**21.2**	**13.7**	**2.8**	**13.4**
		ActionFormer+Ours (L2)	**30.1**	**27.6**	**23.5**	**18.2**	**11.3**	**22.1**	21.1	13.5	2.7	13.4
	InternVideo	ActionFormer	28.7	26.1	22.4	17.4	11.5	21.2	20.9	13.2	2.8	13.3
		ActionFormer+TTA-KD	28.9	26.3	22.1	17.1	11.3	21.1	21.6	12.7	1.8	13.1
		ActionFormer+Ours (KL)	**29.5**	26.4	22.7	**17.8**	**11.5**	**21.6**	22.2	14.1	2.6	14.1
		ActionFormer+Ours (L2)	29.3	**26.8**	**23.0**	17.6	11.0	21.5	**22.6**	**14.6**	**2.9**	**14.3**

TABLE IV: Ablation study on the placement of our Bi-TSA module. All experiments are performed on the THUMOS14 dataset under the 75%-Seen 25%-Unseen split. (TF) and (DH) denote applying our Bi-TSA module to the Temporal Fusion and Detection Head. Best results are in **bold**.

Bi-TSA (TF)	Bi-TSA (DH)	mAP@IoU					
		0.3	0.4	0.5	0.6	0.7	AVG
×	×	39.4	35.3	29.7	22.7	14.3	28.3
✓	×	40.9	36.6	30.5	23.3	15.5	29.3
✓	✓	**41.6**	**37.2**	**31.0**	**23.7**	**15.7**	**29.8**

TABLE V: Ablation study on loss weight combinations of Bi-TSA module on THUMOS14 dataset under 75%-Seen 25%-Unseen split setting using ActionFormer with InternVideo encoder.

α ($\mathcal{L}_{\text{inst}}$)	β ($\mathcal{L}_{\text{temp}}$)	mAP@IoU					
		0.3	0.4	0.5	0.6	0.7	AVG
1.0	0	40.9	36.2	29.9	21.9	14.2	28.6
0.8	**0.2**	**41.6**	**37.2**	**31.0**	**23.7**	**15.7**	**29.8**
0.5	0.5	41.4	36.9	30.5	23.2	15.4	29.4
0.2	0.8	40.9	36.2	30.1	23.2	15.0	29.1

ActivityNet-1.3 under 75%-25% split). These demonstrate the TSA module's robustness to the choice of loss function. We ultimately chose the L2 loss for its computational simplicity and stability.

E. Ablation Study

*1) **Effect of Bi-TSA Placement**:* To investigate the specific role of our proposed Bi-TSA module, we present a detailed ablation study in Table IV. All experiments are based on the ActionFormer framework with InternVideo as the encoder under the 75%-25% split setting on THUMOS14, comparing the effects of integrating the Bi-TSA module into the temporal fusion (TF), the detection head (DH), or both.

Comparing the baseline +Bi-TSA (TF) shows that adding the Bi-TSA module to the temporal fusion alone improves the average mAP by 1.0%. This proves that self-supervised learning at the encoding stage enables the model to actively capture and adapt to a video's unique temporal dynamics, providing more robust and discriminative features for downstream tasks.

The last row shows that integrating Bi-TSA into both components achieves the best performance with an average mAP gain of 1.5%, significantly surpassing single-component applications. This indicates a synergy where the Bi-TSA in temporal fusion refines robust features from the video, while the Bi-TSA in detection head further optimizes these features for boundary prediction, maximizing the model's overall adaptive and temporal modeling capabilities.

*2) **Effect of Loss Weighting in Bi-TSA Module**:* To systematically determine the optimal balance between the **Instance Reconstruction Loss** $\mathcal{L}_{\text{inst}}$ and the **Temporal Consistency Loss** $\mathcal{L}_{\text{temp}}$, we conducted an ablation study on the THUMOS14 dataset (75%-seen 25%-unseen split) using the ActionFormer with the InternVideo encoder. We evaluated the performance of the Bi-TSA module under several weighting schemes (α, β). The experimental results in Table V show that the model achieves optimal temporal localization performance (mAP@0.5 = 31.0%) with the weight configuration $(\alpha = 0.8, \beta = 0.2)$. The experiments also reveal that all dual-loss configurations incorporating the temporal consistency loss significantly outperform the configuration using only the instantaneous reconstruction loss, and all exceed the baseline without TSA (mAP@0.5 = 29.7%). This confirms that instantaneous reconstruction alone captures only local information, whereas temporal consistency enforces coherence

TABLE VI: Comparison of different fusion strategies of Bi-TSA module. All experiments are performed on the THUMOS14 dataset under the 75%-Seen 25%-Unseen split setting.

Method	mAP@IoU					
	0.3	0.4	0.5	0.6	0.7	AVG
ActionFormer	39.4	35.3	29.7	22.7	14.3	28.3
Add	40.5	35.8	29.4	22.2	14.4	28.5
Max	40.3	35.9	30.0	22.4	14.1	28.5
Avg (Ours)	**41.6**	**37.2**	**31.0**	**23.7**	**15.7**	**29.8**

TABLE VII: Ablation study on test-time update strategies of Bi-TSA module at different stages on THUMOS14 dataset under 75%-Seen 25%-Unseen split setting using ActionFormer with InternVideo encoder.

Stage	Methods	mAP@IoU					
		0.3	0.4	0.5	0.6	0.7	AVG
Mid	w/o Test-time Update	39.7	35.4	28.3	22.1	13.0	27.7
	w/ Test-time Update (Cross-Video)	38.2	34.4	28.5	21.4	13.3	27.2
	w/ Test-time Update (Per-Video)	**40.5**	**35.8**	**29.7**	**22.9**	**14.3**	**28.6**
Post	w/o Test-time Update	40.0	35.3	28.8	22.7	14.4	28.2
	w/ Test-time Update (Cross-Video)	40.9	36.9	30.4	23.0	14.7	29.2
	w/ Test-time Update (Per-Video)	**41.6**	**37.2**	**31.0**	**23.7**	**15.7**	**29.8**

by reconstructing the preceding timestep. By preserving cross-segment dependencies during adaptation, this loss drives the model to learn smoother and more generalizable dynamics, thereby improving action localization performance.

*3) **Different Fusion Strategies of Bi-TSA Module**:* A critical design in the Bi-TSA module is the strategy for fusing the forward and backward temporal representations. To identify the most effective approach, we conduct an ablation study comparing three common fusion operations: element-wise addition (Add), max-pooling (Max), and average-pooling (Avg, our method). Experiments are performed on the THUMOS14 dataset under the 75%-Seen 25%-Unseen split setting.

The results in Table VI show that the AVG fusion strategy achieves the best performance, outperforming both Add and Max across all IoU thresholds, which is attributed to its inherent property of preserving information from both temporal directions equally. In contrast, the Add strategy tends to amplify features from one temporal direction, while the Max strategy may retain only local extremes and overlook relevant temporal information. Notably, regardless of different fusion strategies, our method consistently outperforms the original framework in average mAP. This demonstrates that the Bi-TSA module strengthens temporal representations and improves the precision of action boundary localization by integrating both historical and future temporal information.

*4) **Effect of Test-Time Update Strategy for Bi-TSA Module**:* To validate the role of test-time adaptation in Bi-TSA, we conduct an ablation study within the ActionFormer framework on THUMOS14 under the 75%-Seen 25%-Unseen split. Specifically, we compare three update strategies for Bi-TSA: **(1) No test-time update**, where the Bi-TSA module is updated only during training and remains frozen during inference; **(2) Continuous test-time update across videos**, where the parameter memory W is updated continually over consecutive test videos; **(3) Per-video reset test-time update**, our full method, in which the parameter memory W is reset before each new test video is processed.

TABLE VIII: Per-class AP@0.5 comparison of different test-time update strategies on THUMOS14 dataset under 75%-Seen 25%-Unseen split setting using ActionFormer with InternVideo encoder. 'U-A' and 'U-B' denote the two unseen classes "HammerThrow" and "ThrowDiscus" in THUMOS14, while 'S-A' and 'S-B' denote the two seen classes "BasketballDunk" and "Diving", under the 75%-25% split setting.

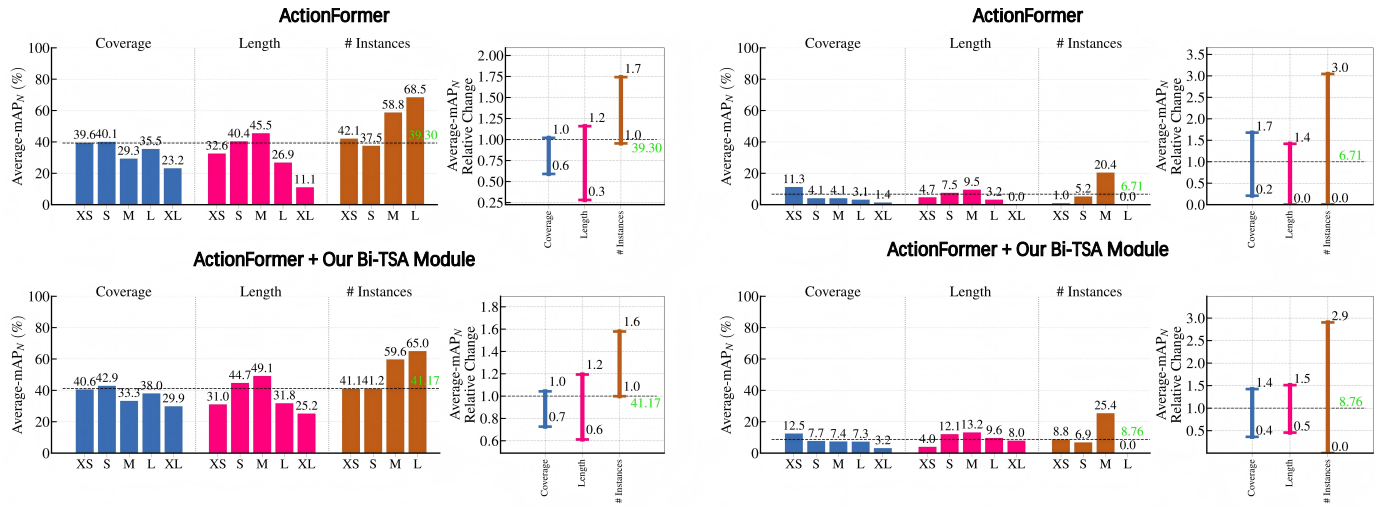
Methods	AP@0.5			
	U-A	U-B	S-A	S-B
w/o Test-time Update	26.5	4.4	69.5	70.2
w/ Test-time Update (Cross-Video)	27.4	4.0	67.7	72.5
w/ Test-time Update (Per-Video)	**34.3**	**6.2**	**71.3**	**74.2**

TABLE IX: Ablation study on weight sharing strategies of Bi-TSA module on THUMOS14 dataset under 75%-Seen 25%-Unseen split setting using ActionFormer with InternVideo encoder.

Method	mAP@IoU					
	0.3	0.4	0.5	0.6	0.7	AVG
Shared Weight	39.3	34.9	28.5	21.9	14.8	28.0
Non-shared Weight (Ours)	**41.6**	**37.2**	**31.0**	**23.7**	**15.7**	**29.8**

The results in Table VII show that methods with test-time updating outperform the no-update strategy, highlighting the necessity of equipping TAL models with online adaptation mechanisms in realistic open environments, where unseen actions emerge alongside seen ones and static models struggle to generalize effectively. Additionally, the per-video reset update yields a 0.6-point improvement in mAP@0.5 compared to cross-video updating. Furthermore, we separately analyzed two seen and two unseen classes in THUMOS14. As shown in Table VIII, the per-video reset update strategy improves AP@0.5 by 2–7 points over the cross-video update strategy on both class types, reaffirming its stronger generalization. Continuous cross-video updating may introduce cross-video drift due to variations in appearance and action rhythm across different videos, which can interfere with modeling the temporal consistency of the current video. In contrast, resetting the memory per video enables the model to focus more effectively on the dynamic patterns of the present video, thereby better leveraging temporal context for localization and feature refinement. This ultimately enhances generalization performance on both seen and unseen classes.

*5) **Effect of Parameter Sharing in Bi-TSA Module**:* To investigate the effect of parameter sharing in bidirectional modeling, we conducted ablation studies on the THUMOS14 dataset (75%-seen 25%-unseen split) with the ActionFormer with InternVideo encoder. We compared the Bi-TSA design with Non-shared weights (our method), where parameters are separate for each direction, against a parameter-sharing variant



(a) Effectiveness of Bi-TSA on seen classes under different characteristics (b) Effectiveness of Bi-TSA on unseen classes under different characteristics

Fig. 5: The detailed comparison of integrating Bi-TSA module on ActionFormer on THUMOS14 with 75%-Seen 25%-Unseen split setting, using [69]. Actions are grouped by three characteristics: Coverage (XS: (0,0.02], S: (0.02,0.04], M: (0.04,0.06], L: (0.06,0.08], XL: (0.08,1.0]); Length in seconds (XS: (0,3], S: (3,6], M: (6,12], L: (12,18], XL: >18); and the number of Instances (XS: 1, S: [2,40], M: [40,80], L: >80). (a-b) *Left*: The normalized mAP at tIoU=0.5 under different action characteristics. (a-b) *Right*: The relative normalized mAP change at tIoU=0.5 with respect to different characteristics of the ground truth instances.

TABLE X: Complexity comparison with the baseline method (ActionFormer) on both datasets with InternVideo encoder. 'mAP' denotes mAP@0.5 for THUMOS14 and the average mAP for ActivityNet-1.3.

Dataset	Methods	Param.	Mem.	MACs	FPS	mAP
THUMOS14	Baseline	184.0M	8.8G	72.7G	49	29.70
	+Ours	200.6M	14.8G	106.0G	47	31.04
ActivityNet-1.3	Baseline	184.3M	16.6G	6.1G	45	19.60
	+Ours	200.9M	20.6G	8.8G	40	21.73

of Bi-TSA, in which the forward and backward TSA modules share the same set of parameters. As shown in Table IX, the results indicate that the shared-weight Bi-TSA variant leads to a performance drop of 2.5% in mAP@0.5 (31.0% → 28.5%). This performance drop suggests that forward and backward streams capture distinct temporal patterns in videos. Our non-shared design preserves the flexibility to model bidirectional dependencies independently, which is crucial for robust temporal adaptation in GZS-TAL.

6) ***Complexity****:* To assess the real-world applicability of our method, we compare the ActionFormer with its corresponding versions integrated with our Bi-TSA module (denoted as +Ours) from four aspects. **1) Parameter Count (Param.)**: Total trainable parameters of the model. **2) GPU Memory Usage (Mem.)**: Peak memory consumption during inference. **3) Computational Complexity (MACs)**: Number of multiply-accumulate operations. **4) Frames Per Second (FPS)**: Inference speed measured on a single NVIDIA RTX A800 GPU.

Our experiments were conducted on both datasets using the InternVideo encoder. As shown in Table X, the analysis reveals that our Bi-TSA module increases the parameter count by less than 10%, with a commensurate rise in computational load due to the online parameter updates. Furthermore, after integrating the Bi-TSA module, the model's inference speed remains above 40 FPS. We argue that the significant performance improvement obtained in return well justifies this overhead. Our method explicitly considers resource constraints, facilitating deployment on edge devices. The parameter and memory overheads are minimal, ensuring practical feasibility. Its plug-and-play design allows flexible enabling or bypassing, and the memory update frequency is adjustable. These features make Bi-TSA a deployable solution suited for open-world scenarios involving unknown actions.

F. Analysis of Balancing Adaptation and Retention

To quantitatively assess the balance between adaptation and retention, we follow [69] and disentangle the GZS-TAL results on THUMOS14 into two metrics: seen-class mAP (s-mAP), reflecting knowledge retention from training, and unseen-class mAP (u-mAP), reflecting generalization to novel categories. We integrate Bi-TSA into the ActionFormer backbone and evaluate under the 75%–25% split. As shown in Fig. 5, Bi-TSA improves u-mAP@0.5 by 2.05% over the baseline, confirming its ability to dynamically adapt to unseen classes. At the same time, it raises s-mAP@0.5 by 1.87%, showing that adaptation is achieved without sacrificing performance on seen categories. This indicates that Bi-TSA not only alleviates catastrophic forgetting but also enhances discriminative power for both seen and unseen classes through stronger temporal modeling.

Beyond average performance, we further evaluate robustness across different action characteristics (coverage, length, and number of instances) as defined in [69]. As illustrated in Fig. 5, our method consistently improves results under all conditions, while exhibiting smaller variances than the

baseline. This confirms the stability of Bi-TSA across diverse action scenarios.

V. Conclusion

In this work, we have introduced the Generalized Zero-Shot Temporal Action Localization (GZS-TAL) task, which better reflects realistic open-world scenarios where seen and unseen classes co-exist at test time. To address the inherent challenges, we have proposed the Temporal-Sensitive Adaptation (TSA) block and its bidirectional extension (Bi-TSA), which enable test-time parameter updates guided by self-supervised temporal objectives. Extensive experiments on THUMOS14 and ActivityNet-1.3 have shown that integrating our modules into existing TAL frameworks has yielded consistent improvements across backbones and feature encoders. These results suggest that our approach has the potential to enhance generalization to unseen classes while retaining robustness on seen ones. We believe that this work has laid the foundation for exploring adaptive temporal localization in open-world video understanding, and we expect that it can inspire further research on test-time adaptation mechanisms and more flexible models for long video analysis.

References

[1] P. Wu, X. Zhou, G. Pang, Y. Sun, J. Liu, P. Wang, and Y. Zhang, "Open-vocabulary video anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 297–18 307.

[2] L. Zhu, L. Wang, A. Raj, T. Gedeon, and C. Chen, "Advancing video anomaly detection: A concise review and a new dataset," *Advances in Neural Information Processing Systems*, vol. 37, pp. 89 943–89 977, 2024.

[3] L. Zanella, W. Menapace, M. Mancini, Y. Wang, and E. Ricci, "Harnessing large language models for training-free video anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 527–18 536.

[4] M. Rodrigo, C. Cuevas, D. Berjón, and N. García, "Automatic highlight detection in videos of martial arts tricking," *Multimedia Tools and Applications*, vol. 83, no. 6, pp. 17 109–17 133, 2024.

[5] Z. Islam, S. Paul, and M. Rochan, "Unsupervised video highlight detection by learning from audio and visual recurrence," in *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2025, pp. 8702–8711.

[6] T. Li, Z. Sun, and X. Xiao, "Unsupervised modality-transferable video highlight detection with representation activation sequence learning," *IEEE Transactions on Image Processing*, vol. 33, pp. 1911–1922, 2024.

[7] A. Xarles, S. Escalera, T. B. Moeslund, and A. Clapés, "T-deed: Temporal-discriminability enhancer encoder-decoder for precise event spotting in sports videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 3410–3419.

[8] K. Zhang, S. Wang, N. Jia, L. Zhao, C. Han, and L. Li, "Integrating visual large language model and reasoning chain for driver behavior analysis and risk assessment," *Accident Analysis & Prevention*, vol. 198, p. 107497, 2024.

[9] C. Ju, T. Han, K. Zheng, Y. Zhang, and W. Xie, "Prompting visual-language models for efficient video understanding," in *European conference on computer vision*. Springer, 2022, pp. 105–124.

[10] S. Nag, X. Zhu, Y.-Z. Song, and T. Xiang, "Zero-shot temporal action detection via vision-language prompting," in *European conference on computer vision*. Springer, 2022, pp. 681–697.

[11] J.-R. Du, K.-Y. Lin, J. Meng, and W.-S. Zheng, "Towards completeness: A generalizable action proposal generator for zero-shot temporal action localization," in *International Conference on Pattern Recognition*. Springer, 2024, pp. 252–267.

[12] B. Liberatori, A. Conti, P. Rota, Y. Wang, and E. Ricci, "Test-time zero-shot temporal action localization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 18 720–18 729.

[13] A. Raza, B. Yang, and Y. Zou, "Zero-shot temporal action detection by learning multimodal prompts and text-enhanced actionness," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 11, pp. 11 000–11 012, 2024.

[14] Y. Lee, H.-J. Kim, and S.-W. Lee, "Text-infused attention and foreground-aware modeling for zero-shot temporal action detection," *Advances in Neural Information Processing Systems*, vol. 37, pp. 9864–9884, 2024.

[15] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019, pp. 4171–4186.

[16] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[17] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[18] Y. Sun, X. Li, K. Dalal, J. Xu, A. Vikram, G. Zhang, Y. Dubois, X. Chen, X. Wang, S. Koyejo *et al.*, "Learning to (learn at test time): Rnns with expressive hidden states," *arXiv preprint arXiv:2407.04620*, 2024.

[19] P. Chen, C. Gan, G. Shen, W. Huang, R. Zeng, and M. Tan, "Relation attention for temporal action localization," *IEEE Transactions on Multimedia*, vol. 22, no. 10, pp. 2723–2733, 2019.

[20] M. Qi, H. Ye, J. Peng, and H. Ma, "Action quality assessment via hierarchical pose-guided multi-stage contrastive regression," *IEEE Transactions on Image Processing*, vol. 34, pp. 6461–6474, 2025.

[21] W. Yun, M. Qi, F. Peng, and H. Ma, "Semi-supervised teacher-reference-student architecture for action quality assessment," in *European Conference on Computer Vision*. Springer, 2024, pp. 161–178.

[22] J. Chen, J. Peng, Y. Lu, J.-H. Lai, and A. J. Ma, "Vision-language adaptive clustering and meta-adaptation for unsupervised few-shot action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[23] Z. Qing, H. Su, W. Gan, D. Wang, W. Wu, X. Wang, Y. Qiao, J. Yan, C. Gao, and N. Sang, "Temporal context aggregation network for temporal action proposal refinement," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 485–494.

[24] J. Tan, J. Tang, L. Wang, and G. Wu, "Relaxed transformer decoders for direct action proposal generation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 13 526–13 535.

[25] C. Zhao, A. K. Thabet, and B. Ghanem, "Video self-stitching graph network for temporal action localization," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 13 658–13 667.

[26] R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, and C. Gan, "Graph convolutional networks for temporal action localization," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 7094–7103.

[27] C.-L. Zhang, J. Wu, and Y. Li, "Actionformer: Localizing moments of actions with transformers," in *European Conference on Computer Vision*. Springer, 2022, pp. 492–510.

[28] D. Shi, Y. Zhong, Q. Cao, L. Ma, J. Li, and D. Tao, "Tridet: Temporal action detection with relative boundary modeling," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 18 857–18 866.

[29] S. Liu, C.-L. Zhang, C. Zhao, and B. Ghanem, "End-to-end temporal action detection with 1b parameters across 1000 frames," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 18 591–18 601.

[30] J. Yang, P. Wei, and N. Zheng, "Cross time-frequency transformer for temporal action localization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 6, pp. 4625–4638, 2023.

[31] H. Liu, X. Li, B. Fan, and J. Xu, "Brtal: Boundary refinement temporal action localization via offset-driven diffusion models," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[32] H.-J. Kim, J.-H. Hong, H. Kong, and S.-W. Lee, "Te-tad: Towards full end-to-end temporal action detection via time-aligned coordinate expression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 837–18 846.

[33] Y. Zhu, G. Zhang, J. Tan, G. Wu, and L. Wang, "Dual detrs for multi-label temporal action detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 559–18 569.

[34] R. Zeng, C. Gan, P. Chen, W. Huang, Q. Wu, and M. Tan, "Breaking winner-takes-all: Iterative-winners-out networks for weakly supervised temporal action localization," *IEEE Transactions on Image Processing*, vol. 28, no. 12, pp. 5797–5808, 2019.

[35] G. Lim, H. Kim, J. Kim, and Y. Choi, "Probabilistic vision-language representation for weakly supervised temporal action localization," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 5507–5516.

[36] W. Yun, M. Qi, C. Wang, and H. Ma, "Weakly-supervised temporal action localization by inferring salient snippet-feature," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 38, no. 7, 2024, pp. 6908–6916.

[37] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[38] S. Yan, X. Xiong, A. Nagrani, A. Arnab, Z. Wang, W. Ge, D. Ross, and C. Schmid, "Unloc: A unified framework for video localization tasks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 13 623–13 633.

[39] C. Han, H. Wang, J. Kuang, L. Zhang, and J. Gui, "Training-free zero-shot temporal action detection with vision-language models," *arXiv preprint arXiv:2501.13795*, 2025.

[40] S. Liu, M. Long, J. Wang, and M. I. Jordan, "Generalized zero-shot learning with deep calibration network," *Advances in neural information processing systems*, vol. 31, 2018.

[41] D. Mahapatra, Z. Ge, and M. Reyes, "Self-supervised generalized zero shot learning for medical image classification using novel interpretable saliency maps," *IEEE Transactions on Medical Imaging*, vol. 41, no. 9, pp. 2443–2456, 2022.

[42] Y. Jia, X. Ye, P. Li, and S. Guo, "Contrastive adaptation on domain augmentation for generalized zero-shot side-scan sonar image classification," *IEEE Transactions on Instrumentation and Measurement*, 2025.

[43] M. Hong, G. Li, X. Zhang, and Q. Huang, "Generalized zero-shot video classification via generative adversarial networks," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 2419–2426.

[44] M. Hong, X. Zhang, G. Li, and Q. Huang, "Fine-grained feature generation for generalized zero-shot video classification," *IEEE Transactions on Image Processing*, vol. 32, pp. 1599–1612, 2023.

[45] K. Liu, W. Liu, H. Ma, W. Huang, and X. Dong, "Generalized zero-shot learning for action recognition with web-scale video data," *World Wide Web*, vol. 22, no. 2, pp. 807–824, 2019.

[46] D. Mandal, S. Narayan, S. K. Dwivedi, V. Gupta, S. Ahmed, F. S. Khan, and L. Shao, "Out-of-distribution detection for generalized zero-shot action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9985–9993.

[47] K. Huang, S. Mckeever, and L. Miralles-Pechuán, "Generalized zero-shot learning for action recognition fusing text and image gans," *IEEE Access*, vol. 12, pp. 5188–5202, 2024.

[48] T. Su, H. Wang, Q. Qi, L. Wang, and B. He, "Transductive learning with prior knowledge for generalized zero-shot action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 1, pp. 260–273, 2023.

[49] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," *arXiv preprint arXiv:2006.10726*, 2020.

[50] S. Niu, J. Wu, Y. Zhang, Y. Chen, S. Zheng, P. Zhao, and M. Tan, "Efficient test-time model adaptation without forgetting," in *International conference on machine learning*. PMLR, 2022, pp. 16 888–16 905.

[51] S. Niu, J. Wu, Y. Zhang, Z. Wen, Y. Chen, P. Zhao, and M. Tan, "Towards stable test-time adaptation in dynamic wild world," in *The Eleventh International Conference on Learning Representations*.

[52] R. Zeng, Q. Deng, H. Xu, S. Niu, and J. Chen, "Exploring motion cues for video test-time adaptation," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 1840–1850.

[53] R. Zeng, Q. Deng, R. Zhang, S. Niu, J. Chen, X. Hu, and V. C. Leung, "Exploring audio cues for enhanced test-time video model adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[54] J. Liu and Z. Yang, "Test-time adaptation for real-world video adverse weather restoration with meta batch normalization," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[55] Q. Deng, S. Niu, R. Zhang, Y. Chen, R. Zeng, J. Chen, and X. Hu, "Learning to generate gradients for test-time adaptation via test-time training layers," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 15, 2025, pp. 16 235–16 243.

[56] W. Liu, X. Shen, H. Li, X. Bi, B. Liu, C.-M. Pun, and X. Cun, "Depth-aware test-time training for zero-shot video object segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 218–19 227.

[57] K. Dalal, D. Koceja, J. Xu, Y. Zhao, S. Han, K. C. Cheung, J. Kautz, Y. Choi, Y. Sun, and X. Wang, "One-minute video generation with test-time training," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 17 702–17 711.

[58] R. Wang, Y. Sun, A. Tandon, Y. Gandelsman, X. Chen, A. A. Efros, and X. Wang, "Test-time training on video streams," *Journal of Machine Learning Research*, vol. 26, no. 9, pp. 1–29, 2025.

[59] B. Zhang and R. Sennrich, "Root mean square layer normalization," *Advances in neural information processing systems*, vol. 32, 2019.

[60] H. Idrees, A. R. Zamir, Y.-G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah, "The thumos challenge on action recognition for videos "in the wild"," *Computer Vision and Image Understanding*, vol. 155, pp. 1–23, 2017.

[61] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2015, pp. 961–970.

[62] Y. Wang, K. Li, Y. Li, Y. He, B. Huang, Z. Zhao, H. Zhang, J. Xu, Y. Liu, Z. Wang *et al.*, "Internvideo: General video foundation models via generative and discriminative learning," *arXiv preprint arXiv:2212.03191*, 2022.

[63] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[64] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[65] G. Li, D. Cheng, N. Wang, J. Li, and X. Gao, "Neighbor-guided pseudo-label generation and refinement for single-frame supervised temporal action localization," *IEEE Transactions on Image Processing*, vol. 33, pp. 2419–2430, 2024.

[66] S. Kim and M. Kim, "Test-time adaptation toward personalized speech enhancement: Zero-shot learning with knowledge distillation," in *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2021, pp. 176–180.

[67] L. Weijler, M. J. Mirza, L. Sick, C. Ekkazan, and P. Hermosilla, "Ttt-kd: Test-time training for 3d semantic segmentation through knowledge distillation from foundation models," in *2025 International Conference on 3D Vision (3DV)*. IEEE, 2025, pp. 1264–1274.

[68] X. Chen, Y. Guo, J. Liang, S. Zhuang, R. Zeng, and X. Hu, "Temporal action detection model compression by progressive block drop," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 29 225–29 236.

[69] H. Alwassel, F. C. Heilbron, V. Escorcia, and B. Ghanem, "Diagnosing error in temporal action detectors," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 256–272.

Mingkui Tan received the bachelor's degree in environmental science and engineering and the master's degree in control science and engineering from Hunan University, Changsha, China, in 2006 and 2009, respectively, and the Ph.D. degree in computer science from Nanyang Technological University, Singapore, in 2014. From 2014 to 2016, he was a Senior Research Associate on computer vision at the School of Computer Science, University of Adelaide, Adelaide, SA, Australia. He is currently a Professor with the School of Software Engineering, South China University of Technology, Guangzhou, China. His research interests include machine learning, sparse analysis, deep learning, and large-scale optimization.

Yihao Qian is currently a M.S. student in the College of Software Engineering, South China University of Technology. He obtained his B.S. degree in School of Computer Science and Technology from Zhejiang University of Technology in 2024. His research interests include machine learning, video understanding and test-time adaptation in computer vision.

Yirui Wang is currently a M.S. student in the School of Computer Science, Beijing Institute of Technology. She received her B.S. degree from Harbin Engineering University in 2024. Her research interests include test-time adaptation, computer vision, and temporal action localization.

Runhao Zeng received the PhD degree in software engineering from South China University of Technology, in 2021. He is currently an associate professor at the Artificial Intelligence Research Institute, Shenzhen MSU-BIT University. He has authored or coauthored several peer-reviewed papers on computer vision, machine learning on top-tier conferences and journals, including the Proceedings of NeurIPS, CVPR, ICCV, and TPAMI. His current research interests include machine learning, computer vision, with particular focus on video analysis.

Victor C. M. Leung (Life Fellow, IEEE) is currently a Professor in computer science and software engineering with Shenzhen MSU-BIT University, Shenzhen, China. He is also an Emeritus Professor in electrical and computer engineering and the Director of the Laboratory for Wireless Networks and Mobile Systems, The University of British Columbia (UBC), Vancouver, BC, Canada. He has coauthored more than 1300 journals/conference papers and book chapters. His research is in the broad areas of wireless networks and mobile systems. He is serving on the Editorial Board of the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE ACCESS, IEEE Network, and several other journals. He was a recipient of the IEEE Vancouver Section Centennial Award, 2011 UBC Killam Research Prize, 2017 Canadian Award for Telecommunications Research, and 2018 IEEE TCGCC Distinguished Technical Achievement Recognition Award. He has coauthored papers that won the 2017 IEEE ComSoc Fred W. Ellersick Prize, 2017 IEEE Systems Journal Best Paper Award, 2018 IEEE CSIM Best Journal Paper Award, and 2019 IEEE TCGCC Best Journal Paper Award. He is a fellow of the Royal Society of Canada, Canadian Academy of Engineering, and Engineering Institute of Canada. He is named in the current Clarivate Analytics list of "Highly Cited Researchers."

Xiping Hu received the Ph.D. degree from the University of British Columbia, Vancouver, BC, Canada. He is currently a professor with Beijing Institute of Technology, and with Shenzhen MSU-BIT University, China. He has more than 180 papers published and presented in prestigious conferences and journals, such as IEEE TPAMI/TMC/TPDS/TIP/JSAC, IEEE COMST, ACM MobiCom/MM/SIGIR/WWW, AAAI, and IJCAI. He has been serving as associate editor of IEEE TCSS, and the lead guest editors of IEEE IoT Journal and IEEE TASE etc. He has been granted several key research projects with more than 15 million USD as principal investigator. He was the Co-Founder and CTO of Bravolol Ltd., Hong Kong, a leading language learning mobile application company with over 300 million users, and listed as the top 2 language education platform globally. His research areas consist of mobile cyber-physical systems, crowdsensing and affective computing.