SEAM-DETR: TRANSITION- AND QUALITY-AWARE DECODER FOR TEMPORAL ACTION DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Query-based detectors have become a strong family for temporal action detection, yet their localization quality remains unsatisfactory. We trace the problem to the two sides of the decoder. On the input side, the memory attended by the decoder marks a boundary only as a change between moments rather than as content at any one of them, which makes transitions hard to perceive, especially gradual ones. On the supervision side, one-to-one matching assigns the positive label to a single proposal per instance and treats its near-duplicates as background, so the learned scores reflect the matching assignments rather than the actual overlap with the ground truth. We propose Seam-DETR, which restores localization quality from both sides. Transition-Aware Memory (TAM) adds two parameter-free views to the memory that the decoder reads for localization, one measuring how much the content changes at each moment and the other the direction of the change, so transitions become perceptible. Match-Aware Ranking (MAR) prevents unmatched near-duplicates from being suppressed as background and ranks same-class proposals by their overlap with the ground truth, making confidence scores more localization-aware. TAM adds two lightweight adapters, and MAR introduces no additional parameters or inference cost. Extensive experiments demonstrate that Seam-DETR achieves state-of-the-art performance among query-based methods on THUMOS14, ActivityNet-1.3, and HACS-Segment.

1 INTRODUCTION

Temporal action detection (TAD) localizes the start and end times of every action instance in an untrimmed video and recognizes its category. Existing detectors form three families, anchor-based (Zhao et al., 2017; Lin et al., 2018; 2019; Xu et al., 2020), anchor-free (Lin et al., 2021; Zhang et al., 2022; Shi et al., 2023; Yang et al., 2024a), and query-based (Liu et al., 2022; Tan et al., 2021; Shi et al., 2022; Kim et al., 2023; 2024; 2025a). Query-based detectors, inspired by DETR (Carion et al., 2020; Zhu et al., 2021), have attracted interest because they remove hand-crafted components such as anchors and dense proposals from the pipeline. A set of queries attends to the encoded video features through cross-attention in a decoder, and one-to-one matching trains the set to predict action instances directly.

Despite these advances, localization quality is still the bottleneck. The accuracy of query-based detectors drops sharply as the temporal intersection over union (tIoU) threshold tightens, and strict thresholds are exactly where precise boundaries and a trustworthy ranking matter. Many detections find the action without pinning down where it begins and ends, or without being ranked above the ones that do. We therefore ask where a query-based detector loses its precision.

We find that a query-based detector loses precision on two sides of the decoder, transition blindness on the input side and confidence mismatch on the supervision side, and Fig. 1 measures both on three query-based detectors. The memory that the decoder attends to describes what is happening at each moment; a boundary appears in it only as a change from one moment to the next, while the decoder reads what is happening at the moments it samples. An abrupt transition packs that change into an instant right at the boundary, while a gradual one spreads it over seconds, as when a javelin thrower walks into the run-up, so no single moment stands out. We call the time this change takes in the video the transition duration of a boundary. Fig. 1(a) sorts action boundaries by transition duration and shows that a tightly fitting proposal becomes rarer toward the gradual end. On the supervision

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.

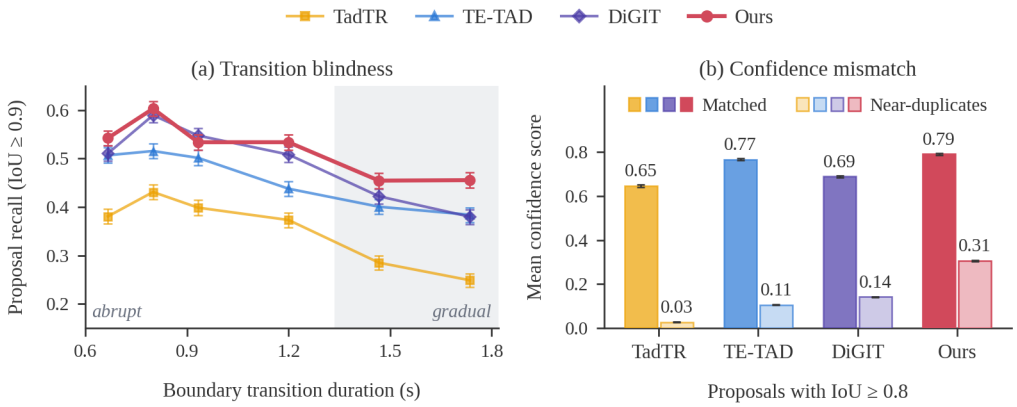


Figure 1: Two problems of query-based detectors, measured on THUMOS14. (a) Transition blindness. Proposal recall at IoU ≥ 0.9 falls toward long transition durations (Sec. A.2), and Ours keeps more tight proposals at the gradual end (shaded). (b) Confidence mismatch. Among proposals with IoU ≥ 0.8, the near-duplicates that the matching leaves unmatched score far below the matched proposal, and Ours raises their scores. Error bars show standard errors.

side, one-to-one matching marks a single query per instance as positive and treats every other query as background, including near-duplicates that fit the instance as well as the matched one or better. The classification loss therefore pushes the score of a proposal toward zero for being unmatched, not for being loose. Fig. 1(b) compares proposals that overlap an instance closely and shows that the near-duplicates among them score far lower on average than the matched proposals. A score learned this way only weakly tracks how well a detection fits, so a tight detection is not reliably ranked above a loose one.

Neither side tells the decoder how well a proposal fits. We propose Seam-DETR, which restores localization quality on both sides of the decoder and leaves the decoder architecture and the inference procedure unchanged. Transition-Aware Memory (TAM) makes change explicit in what the decoder reads for localization. It adds two parameter-free views of the memory. One measures how much the content changes at each moment, and the other contrasts the past with the future to expose the direction of the change. Two lightweight adapters feed the views in and start at zero, so the detector begins as the original model and learns to use the views only where they help. Match-Aware Ranking (MAR) makes the confidence score follow localization quality. It stops the training from suppressing near-duplicates and adds two ranking losses that order same-class proposals by their overlap with the ground truth. The two components act on the input side and the supervision side independently and add no parameters beyond the two adapters. Every query-based detector has these two sides, so both components carry over to other detectors of this family, as Sec. A.4 shows.

We summarize our contributions as follows:

- We identify two problems that limit the localization quality of query-based temporal action detectors, transition blindness on the input side of the decoder and confidence mismatch on its supervision side.
- We propose Transition-Aware Memory, which makes transitions perceptible to the decoder through two parameter-free temporal views of the memory.
- We propose Match-Aware Ranking, which makes confidence scores follow localization quality by sparing near-duplicates from suppression and ranking proposals by overlap.
- Seam-DETR achieves state-of-the-art performance among query-based methods on three benchmarks, gaining most at strict tIoU thresholds.

2 Related Work

Temporal Action Detection. Anchor-based detectors (Lin et al., 2018; 2019; Xu et al., 2020; Qing et al., 2021) generate candidate segments from predefined anchors or from boundary probabilities

predicted at every time step, and then classify and refine them. Anchor-free detectors (Lin et al., 2021; Zhang et al., 2022; Shi et al., 2023; Yang et al., 2024a; Shao et al., 2023) predict a segment directly at each time step, and several of them model boundaries explicitly. AFSD (Lin et al., 2021) refines coarse segments with salient features pooled near their boundaries, TriDet (Shi et al., 2023) models each boundary as a distribution over nearby positions, and ASL (Shao et al., 2023) reweights each frame by how much it matters for classification and for localizing the boundaries.

Query-Based Detectors. Inspired by DETR (Carion et al., 2020), query-based detectors represent instances with a set of queries trained by one-to-one matching, which removes hand-crafted anchors and dense proposals. RTD-Net (Tan et al., 2021) and ReAct (Shi et al., 2022) brought this formulation to TAD, and TadTR (Liu et al., 2022) added deformable attention and an actionness head. Self-DETR (Kim et al., 2023) guides the collapsed self-attention with cross-attention maps, and Pred-DETR (Kim et al., 2025b) guides the attention with the overlaps among its predictions. TE-TAD (Kim et al., 2024) expresses coordinates in absolute time so that one pass covers a whole video, DualDETR (Zhu et al., 2024) adds boundary-level queries, and DiGIT (Kim et al., 2025a) redesigns the encoder and the decoder sampling regions. With features from large video models (Wang et al., 2024), the strongest of these detectors matches or exceeds the best methods that train only the detection head. We keep the architecture of such a detector and revise the two sides of its decoder that all of them share, namely the memory it attends to and the supervision from one-to-one matching.

Supervision in DETR. One-to-one matching gives each object a single positive query, and DETR variants enrich this supervision along two lines. One adds positives, through extra query groups or auxiliary heads (Chen et al., 2023; Zong et al., 2023) or denser targets (Huang et al., 2025) or quality-aware one-to-many assignment (Kang et al., 2026). The other makes the classification score reflect localization quality. Dense detectors do so with IoU-aware targets (Zhang et al., 2021; Li et al., 2020; Feng et al., 2021) and ranking losses (Chen et al., 2019; Oksuz et al., 2021), and DETR variants with quality-based query selection (Zhao et al., 2024) and ranking-oriented matching and losses (Pu et al., 2023; Tong et al., 2025). MAR follows the second line. Whereas IoU-aware targets change only the target of the matched query, MAR exempts its unmatched near-duplicates from the classification loss and ranks them by overlap with the rank-and-sort losses (Oksuz et al., 2021).

3 METHOD

Fig. 2 gives an overview of Seam-DETR. It keeps the query-based detector reviewed in Sec. 3.1 and modifies it on the two sides of the decoder. On the input side, Transition-Aware Memory (TAM) adds two views of change to the memory that the decoder reads for localization. On the supervision side, Match-Aware Ranking (MAR) supervises the confidence scores to follow localization quality. Sec. 3.2 first measures the two problems on the base detector, Secs. 3.3 and 3.4 present the two components, and Sec. 3.5 describes training and inference.

3.1 PRELIMINARIES

Query-Based TAD Framework. Given the feature sequence of an untrimmed video, extracted by a pre-trained video backbone, a query-based detector encodes it into a memory $M \in \mathbb{R}^{T \times D}$, a sequence of T tokens with D channels. Many detectors then downsample M into S levels $\mathcal{M} = \{M_1, \ldots, M_S\}$, halving the length from one level to the next, and let the decoder attend to all levels. Each query q of the decoder carries an embedding and a reference segment, the center and width of the segment it currently predicts. Both are either learned, as in Deformable DETR (Zhu et al., 2021), or taken from the top-scoring positions of the encoder output, as in two-stage detectors (Kim et al., 2024; 2025a), and the decoder layers refine them in turn. Writing Q^l for the query embeddings after layer l, each layer computes

$$Q^l = \mathrm{FFN}\big(\mathrm{CrossAttn}(\mathrm{SelfAttn}(Q^{l-1}), \mathcal{M})\big), \tag{1}$$

where self-attention lets the queries exchange information with one another and the feed-forward network updates the embedding. In the common deformable form (Zhu et al., 2021), each head of the cross-attention predicts a few sampling positions around the reference segment of a query and a weight for each position. It projects the memory into values, reads the values at these positions by linear interpolation, and takes their weighted sum. Every layer ends with a classification head that

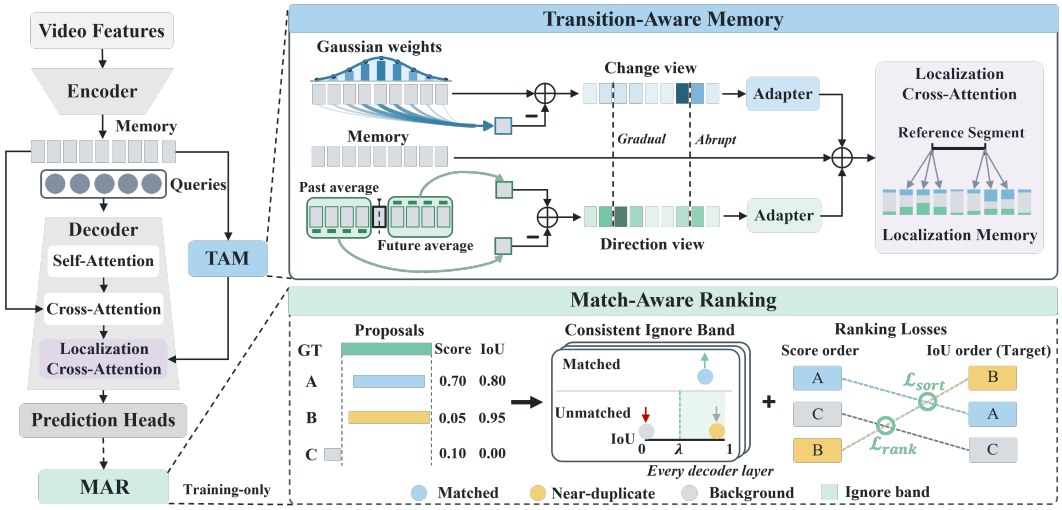


Figure 2: Overview of Seam-DETR. TAM computes a change view and a direction view of the memory, passes each through an adapter, and adds them to the memory. The resulting localization memory feeds the localization cross-attention of every decoder layer. MAR is used only in training. Its ignore band stops the classification loss from suppressing unmatched near-duplicates, and its ranking losses order proposals of the same class by their overlap with the ground truth.

outputs logits z_{qc} for query q and class c, and a regression head that outputs a segment b_q, which serves as the reference segment of the next layer. The outputs of the last layer are the detections, each scored by $s_{qc} = \mathrm{sigmoid}(z_{qc})$.

Label Assignment and Losses. For each training video, Hungarian matching (Kuhn, 1955) assigns every ground-truth instance to exactly one query. Let $\hat{\pi}(i)$ denote the query matched to instance i, with class y_i and segment g_i. The loss of one decoder layer is

$$\mathcal{L}_{\mathrm{det}} = \sum_{q,c} \mathrm{FL}\big(s_{qc}, t_{qc}\big) + \sum_{i} \mathcal{L}_{\mathrm{IoU}}\big(b_{\hat{\pi}(i)}, g_i\big), \tag{2}$$

where FL is the focal loss (Lin et al., 2017), $\mathcal{L}_{\mathrm{IoU}} = 1 - \mathrm{tIoU}(b_{\hat{\pi}(i)}, g_i)$ is the IoU loss (Liu et al., 2022), and the classification target t_{qc} is one if $q = \hat{\pi}(i)$ and $c = y_i$ for some instance i and zero otherwise. Every unmatched query is therefore pushed toward background for every class, however well its segment fits an instance. The same loss applies to the output of every earlier decoder layer as auxiliary supervision, and two-stage detectors also supervise the encoder output from which they select the queries.

3.2 Transition Blindness and Confidence Mismatch

Before describing the two components, we measure both problems on our base detector, DiGIT (Kim et al., 2025a), and trace each to its side of the decoder. Both measurements use THUMOS14.

Transition Blindness. Fig. 3 asks whether a boundary stands out, first in the video features and then in the memory that the decoder attends to. For each annotated boundary we take a four-second window centered on it and compute the cosine similarity between every pair of time steps in the window. The figure averages these similarity matrices over all boundaries, and a control repeats the measurement around random time steps. In the video features a boundary is barely more distinct than any other moment, since the two sides of a boundary window are almost as similar to each other as the two halves of a random window (0.80 against 0.86). The encoder changes this. In the memory the two sides of a boundary window are nearly orthogonal (0.05 against 0.54), so the decoder does receive a mark at the boundary. The mark, however, is a change of content across the boundary rather than a property of the content at it, while the cross-attention reads the content at the positions it samples. A query that regresses a boundary must therefore infer the change from the tokens it samples instead of reading it off them, and it has no landmark that marks the boundary itself.

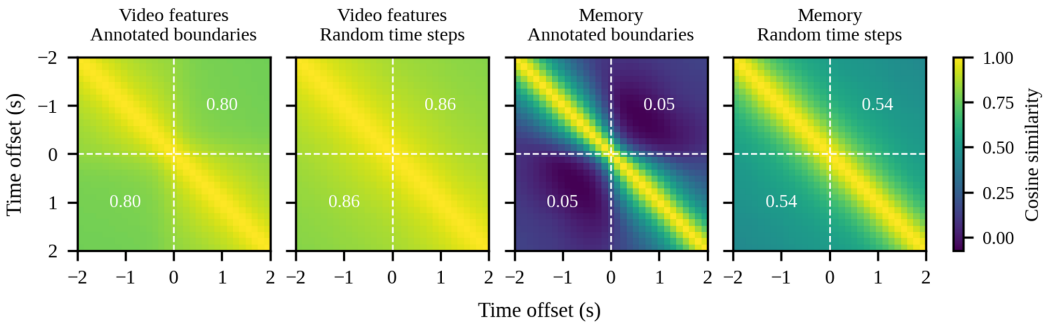


Figure 3: Average cosine similarity between all pairs of time steps in a four-second window centered on annotated boundaries and on random time steps, on the video features and on the memory the decoder attends to (THUMOS14, base detector). The numbers give the mean similarity between the two sides of the window.

Confidence Mismatch. Fig. 4 isolates what the matching does to the score. We run the training-time Hungarian matching with the trained base detector on the test videos. For each instance this yields one matched query and a set of unmatched near-duplicates, the queries whose segments overlap the same instance by at least 0.5. We split these proposals into loose ($0.5 \leq \text{IoU} < 0.8$) and tight ($\text{IoU} \geq 0.8$) ones and plot, in one panel each, the score distributions of the matched queries and of the near-duplicates. In both panels the two distributions barely overlap, with matched queries clustering around 0.8 and near-duplicates near zero. The tight panel shows equally well-localized proposals receiving opposite scores, and across the two panels the matched score barely moves with IoU. The score therefore records the outcome of the matching more than the overlap, and focal loss keeps pushing the unmatched near-duplicates toward zero, so a well-localized proposal scores low when the matching did not pick it.

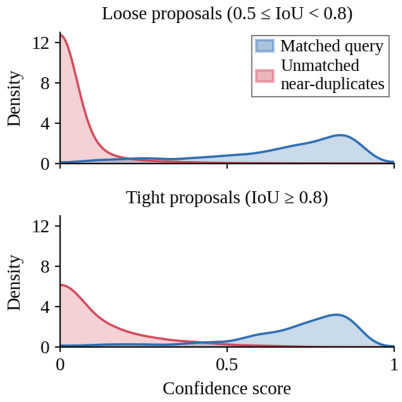


Figure 4: Confidence score distributions of matched queries and of their unmatched near-duplicates.

3.3 Transition-Aware Memory

Sec. 3.2 shows that the memory marks a boundary as a change between the tokens on its two sides rather than as content at any one of them. At a gradual transition this change is also spread over seconds. TAM computes the change explicitly at two scales. The change view contrasts each token with its local neighborhood, and the direction view contrasts the seconds before a moment with the seconds after it. Both are fixed linear filters on M. Their responses pass through two lightweight adapters, the only learnable parts of TAM, and are added to the memory that the decoder reads for localization.

Change View. Let G_σ denote a Gaussian kernel along time with standard deviation σ in time steps. The change view measures how much the content changes at each moment,

$$V_\Delta = M - G_\sigma * M. \tag{3}$$

Convolving with G_σ is a low-pass filter along time that keeps the content, and V_Δ is its high-pass complement. It responds to any local change, is sharpest at abrupt boundaries, and vanishes where the content is steady.

Direction View. The direction view contrasts the past against the future over W steps on each side and exposes the direction of the change,

$$V_\rightarrow(t) = \frac{1}{W}\sum_{k=1}^{W} M(t-k) - \frac{1}{W}\sum_{k=1}^{W} M(t+k). \tag{4}$$

This is the difference of two box averages, a step-shaped filter that in the frequency domain passes a band tuned to transitions spanning about W steps. Because it integrates over $2W$ steps, $V_{\rightarrow}$ keeps responding when the change is spread over a gradual transition, which is exactly where the high-pass V_{Δ} weakens. The two views are therefore complementary.

Localization Memory. Our base detector splits the cross-attention of each layer into a part that samples inside the reference segment and a part that samples around its two ends. We call the second part the localization cross-attention. Two adapters ϕ_{Δ} and $\phi_{\rightarrow}$, each a two-layer MLP with a ReLU in between and the same width as M, map the views into the memory space, and the localization cross-attention reads

$$\tilde{M} = M + \phi_{\Delta}(V_{\Delta}) + \phi_{\rightarrow}(V_{\rightarrow}), \tag{5}$$

while the first part keeps reading M. In a detector with a single cross-attention, the same attention reads $\tilde{M}$ a second time with the same sampling positions and weights, and only the regression head uses this second output. The views are computed once at the finest scale, pass through the same resizing that builds the multi-scale memory, and enter every decoder layer before the value projection of the localization cross-attention. The last layer of each adapter starts at zero, so the detector begins training as the original model and learns to use the views only where they help. In $\tilde{M}$ the sampled tokens themselves carry the transition, the change view marking where the content changes and the direction view marking which way it changes, so a query that regresses a boundary has a landmark to align to.

3.4 Match-Aware Ranking

Sec. 3.2 shows that the score s_{qc} records the outcome of the matching more than how well a proposal fits. MAR changes its supervision in two ways. It stops suppressing unmatched near-duplicates as background, and it orders proposals of the same class by how well they localize. Both rely on a label of localization quality that exists for every query.

Localization Quality. For query q and class c, let

$$l_{qc} = \max_{i:\, y_i = c} \mathrm{tIoU}(b_q, g_i) \tag{6}$$

be the largest overlap between the segment of q and a ground-truth instance i of class c, and $l_{qc} = 0$ when the video holds no such instance. Where the matching target t_{qc} marks a single query per instance, l_{qc} grades every query, matched or not. It is read off the current segments and enters the losses as a constant, so MAR reshapes the scores rather than the segments.

Consistent Ignore Band. We multiply the negative term of the focal loss for the pair (q, c) by

$$w_{qc} = \begin{cases} 0, & q \text{ unmatched and } l_{qc} \geq \lambda, \\ 1, & \text{otherwise.} \end{cases} \tag{7}$$

A proposal that overlaps an instance by at least λ but does not win the matching is neither pushed toward zero nor pulled toward one, and in the last layer the ranking losses below set its score by overlap. Unmatched proposals below λ are still pushed toward background. The band applies in every decoder layer, including the auxiliary losses of the earlier layers, so that the earlier layers do not keep suppressing what the last layer exempts.

Ranking Losses. Within each training video and each class c that appears in it, we define, by overlap rather than by the matching, the positive set $\mathcal{P}_c = \{q : l_{qc} \geq \tau_p\}$ and the negative set $\mathcal{N}_c = \{q : l_{qc} < \tau_n\}$. Proposals with intermediate overlap belong to neither set. They are neither tight enough to count as positives nor far enough to count as clear background, so the ranking losses enforce no order against them. Following the rank-and-sort formulation (Oksuz et al., 2021), we compare two scores through a soft step of half-width δ,

$$H(x) = \mathrm{clip}\big(x/(2\delta) + 1/2,\, 0,\, 1\big), \tag{8}$$

and apply two losses to the scores s_{qc},

$$\mathcal{L}_{\mathrm{rank}} = \sum_{q \in \mathcal{P}_c} \frac{\sum_{q' \in \mathcal{N}_c} H(s_{q'c} - s_{qc})}{\sum_{q' \in \mathcal{N}_c} H(s_{q'c} - s_{qc}) + \sum_{q' \in \mathcal{P}_c} H(s_{q'c} - s_{qc})}, \tag{9}$$

$$\mathcal{L}_{\text{sort}} = \sum_{q \in \mathcal{P}_c} \Big[\frac{\sum_{q' \in \mathcal{P}_c} H(s_{q'c} - s_{qc})\,(1 - l_{q'c})}{\sum_{q' \in \mathcal{P}_c} H(s_{q'c} - s_{qc})} - \frac{\sum_{q' \in \mathcal{P}_c,\, l_{q'c} \geq l_{qc}} H(s_{q'c} - s_{qc})\,(1 - l_{q'c})}{\sum_{q' \in \mathcal{P}_c,\, l_{q'c} \geq l_{qc}} H(s_{q'c} - s_{qc})} \Big]_+ . \tag{10}$$

Here, $[x]_+ = \max(x, 0)$. The rank loss asks every positive to score above every negative, and the sort loss compares, for each positive q, the localization error of the positives ranked above it with that of the positives that deserve to be above it. Both losses act on the class scores s_{qc} of the last decoder layer, so we add no quality head and leave inference unchanged.

3.5 Training and Inference

Training. MAR compares proposals by their overlap with the ground truth, and the comparison is informative only once the detector produces reasonable proposals. We therefore train Seam-DETR in two stages. The first stage trains the detector with TAM from scratch under $\mathcal{L}_{\text{det}}$. The second stage continues training with MAR under the objective

$$\mathcal{L}_{\text{MAR}} = \mathcal{L}_{\text{det}} + \eta_{\text{rank}}\,\mathcal{L}_{\text{rank}} + \eta_{\text{sort}}\,\mathcal{L}_{\text{sort}}, \tag{11}$$

where $\mathcal{L}_{\text{det}}$ now includes the ignore band, the two ranking losses are summed over the classes of each video and averaged over the positives in the batch, and η_{rank} and η_{sort} ramp linearly from zero over the first epochs of the second stage.

Inference. Following previous methods (Kim et al., 2025a), we use the predictions of the last decoder layer and score each detection by s_{qc}, and remove duplicates with non-maximum suppression (NMS). TAM runs as in training, and MAR only changes the losses. The post-processing of each dataset is given in Sec. A.1.

4 Experiments

4.1 Experimental Setup

We evaluate on THUMOS14 (Idrees et al., 2017), ActivityNet-1.3 (Caba Heilbron et al., 2015), and HACS-Segment (Zhao et al., 2019). Following convention, we train on the validation set of THUMOS14 and evaluate on its test set, and we train on the training sets of the other two and evaluate on their validation sets. We report mAP at tIoU thresholds [0.3:0.7:0.1] and their average on THUMOS14, and at $\{0.5, 0.75, 0.95\}$ together with the average over [0.5:0.95:0.05] on the other two. We build on DiGIT (Kim et al., 2025a) with pre-extracted InternVideo2-6B features (Wang et al., 2024). Dataset statistics, the evaluation protocol, and the implementation details of each dataset are given in Sec. A.1.

4.2 Comparison with the State of the Art

THUMOS14. Tab. 1 provides a comparison with the state-of-the-art methods on THUMOS14. With InternVideo2 features, Seam-DETR improves over TadTR, TE-TAD, and DiGIT at every tIoU threshold, and the margin over DiGIT widens as the threshold tightens, from 0.8% at tIoU 0.3 to 2.4% at 0.7. Strict thresholds are where boundary precision and the ranking of detections decide the score, which is what TAM and MAR target. Seam-DETR also outperforms the anchor-free detectors and approaches the strongest full-training methods, which fine-tune the video backbone end to end.

ActivityNet-1.3. Following the conventional approach, the external classification score is combined with class-agnostic predictions to obtain class labels. As shown in Tab. 1, Seam-DETR improves over TadTR, TE-TAD, and DiGIT, and over the anchor-free detectors on the same features in the average mAP. The gain is smaller than on THUMOS14, which reflects the characteristics of the dataset. Most instances of ActivityNet-1.3 are long and each video holds few of them, so gradual transitions and competing proposals matter less.

HACS-Segment. Tab. 2 compares Seam-DETR with the state of the art on HACS-Segment. Seam-DETR improves over TadTR, TE-TAD, and DiGIT at every tIoU threshold and reaches the best average mAP among all methods. As on THUMOS14, Seam-DETR is strongest at the strict thresholds and is ahead of every method at 0.75 and 0.95.

Table 1: Comparison with the state of the art on THUMOS14 and ActivityNet-1.3. Full training fine-tunes the video backbone; the other methods train the detection head on pre-extracted features. Bold marks the best result in each group.

Type	Method	Feature	THUMOS14						ActivityNet-1.3			
			0.3	0.4	0.5	0.6	0.7	Avg.	0.5	0.75	0.95	Avg.
Full training	AFSD (Lin et al., 2021)	I3D	67.3	62.4	55.5	43.7	31.1	52.0	52.4	35.3	6.5	34.4
	TALLFormer (Cheng & Bertasius, 2022)	Swin-B	76.0	–	63.2	–	34.5	59.2	54.1	36.2	7.9	35.6
	ViT-TAD (Yang et al., 2024b)	ViT-B	85.1	80.9	74.2	61.8	45.4	69.5	55.8	38.5	8.8	37.4
	AdaTAD (Liu et al., 2024)	VideoMAEv2-g	89.7	86.7	80.9	71.0	56.1	76.9	61.7	43.4	10.9	41.9
	AdaTAD++ (Agrawal et al., 2025)	VideoMAEv2-g	**90.0**	**87.7**	**82.6**	**73.2**	**59.5**	**78.6**	**63.6**	**45.8**	**12.8**	**43.2**
Anchor-free	ActionFormer (Zhang et al., 2022)	I3D / R(2+1)D	82.1	77.8	71.0	59.4	43.9	66.8	54.7	37.8	8.4	36.6
	TriDet (Shi et al., 2023)	I3D / R(2+1)D	83.6	80.1	72.9	62.4	47.4	69.3	54.7	38.0	8.4	36.8
	DyFADet (Yang et al., 2024a)	I3D / R(2+1)D	84.0	80.1	72.7	61.1	47.9	69.2	58.1	39.6	8.4	38.5
	ActionFormer (Zhang et al., 2022)	InternVideo2	82.3	81.9	75.1	65.8	50.3	71.9	61.5	**44.6**	**12.7**	41.2
	ActionMamba (Chen et al., 2026)	InternVideo2	**86.9**	**83.1**	**76.9**	**65.9**	**50.8**	**72.7**	**62.4**	43.5	10.2	**42.0**
Query-based	RTD-Net (Tan et al., 2021)	I3D / TSN	68.3	62.3	51.9	38.8	23.7	49.0	47.2	30.7	8.6	30.8
	ReAct (Shi et al., 2022)	I3D / TSN	69.2	65.0	57.1	47.8	35.6	55.0	49.6	33.0	8.6	32.6
	Self-DETR (Kim et al., 2023)	I3D	74.6	69.5	60.0	47.6	31.8	56.7	52.3	33.7	8.4	33.8
	DualDETR (Zhu et al., 2024)	I3D	82.9	78.0	70.4	58.5	44.4	66.8	52.6	35.0	7.8	34.3
	Pred-DETR (Kim et al., 2025b)	I3D	80.0	73.5	64.6	52.3	37.6	61.6	54.2	36.4	9.5	36.0
	TadTR (Liu et al., 2022)	InternVideo2	84.8	79.3	70.4	58.2	43.8	67.3	57.1	38.8	11.0	38.2
	TE-TAD (Kim et al., 2024)	InternVideo2	84.3	81.1	73.7	62.6	49.5	70.3	61.3	41.8	10.9	41.1
	DiGIT (Kim et al., 2025a)	InternVideo2	87.6	84.2	77.6	67.3	52.5	73.8	**62.0**	43.1	11.3	42.0
	Seam-DETR	InternVideo2	**88.4**	**85.3**	**79.7**	**69.4**	**54.9**	**75.5**	61.5	**44.0**	**12.4**	**42.6**

Table 2: Comparison with the state of the art on HACS-Segment.

Type	Method	Feature	0.5	0.75	0.95	Avg.
Anchor-based	SSN (Zhao et al., 2017)	I3D	28.8	18.8	5.3	19.0
	G-TAD (Xu et al., 2020)	I3D	41.1	27.6	8.3	27.5
	BMN (Lin et al., 2019)	SlowFast	52.5	36.4	10.4	35.8
	TCANet (Qing et al., 2021)	SlowFast	**54.1**	**37.2**	**11.3**	**36.8**
Anchor-free	TALLFormer (Cheng & Bertasius, 2022)	Swin-B	55.0	36.1	11.8	36.5
	TriDet (Shi et al., 2023)	SlowFast	56.7	39.3	11.7	38.6
	TriDet (Shi et al., 2023)	VideoMAEv2-g	62.4	44.1	13.1	43.1
	DyFADet (Yang et al., 2024a)	SlowFast	57.8	39.8	11.8	39.2
	DyFADet (Yang et al., 2024a)	VideoMAEv2-g	**64.0**	44.8	**14.1**	44.3
	ActionFormer (Zhang et al., 2022)	InternVideo2	62.6	44.6	12.7	43.3
	ActionMamba (Chen et al., 2026)	InternVideo2	**64.0**	**45.7**	13.3	**44.6**
Query-based	TadTR (Liu et al., 2022)	InternVideo2	54.2	38.8	12.8	37.8
	TE-TAD (Kim et al., 2024)	InternVideo2	60.4	45.6	16.5	44.1
	DiGIT (Kim et al., 2025a)	InternVideo2	62.4	47.9	17.6	45.9
	Seam-DETR	InternVideo2	**63.4**	**49.1**	**18.4**	**47.0**

4.3 Ablation Study

We conduct ablation studies on THUMOS14 with InternVideo2 features to evaluate TAM and MAR and to examine their design choices. Base denotes our reproduction of DiGIT, and the default setting is shaded in gray in the tables. The appendix gives further results. Sec. A.2 repeats the probes of Sec. 3.2 on each model and adds an error analysis, Sec. A.3 extends the ablations of TAM and MAR, Sec. A.4 adds TAM and MAR to other query-based detectors, Sec. A.5 reports the computational cost, and Sec. A.6 gives a qualitative analysis.

Component Ablation. Tab. 3 adds the two components to the base detector one at a time. Each helps on its own, raising the average mAP by 1.9 and 1.4, and each helps in a different place. TAM gains most at tIoU 0.5 (+2.8), since once transitions are perceptible more proposals fit their instances well. MAR creates no tighter proposals but lets the tight proposals that the matching used to suppress rank high enough to count, and at tIoU 0.7 it gains more than TAM (+1.3 against +0.8). Together they improve over either alone at every threshold.

Table 3: Contribution of each component.

Method	0.3	0.5	0.7	Avg.
Base	86.9	76.0	52.7	72.8
+ TAM	88.0	78.8	53.5	74.7
+ MAR	87.8	77.6	54.0	74.2
+ TAM + MAR	88.4	79.7	54.9	75.5

Table 4: Ablation of TAM.

Setting	0.3	0.5	0.7	Avg.
Cross-attention				
Inside only	87.8	76.4	53.3	73.7
Both	87.7	78.2	54.0	74.6
Localization only (ours)	88.0	78.8	53.5	74.7
Filters				
Learnable	87.8	78.5	53.2	74.5
Fixed (ours)	88.0	78.8	53.5	74.7
View injection				
After value projection	87.4	77.4	52.9	74.0
Before value projection	88.0	78.8	53.5	74.7

Table 5: Ablation of MAR.

Setting	0.3	0.5	0.7	Avg.
TAM w/o MAR	88.0	78.8	53.5	74.7
Components				
Ignore band only	88.1	79.0	53.6	75.0
Ranking losses only	88.0	78.5	53.2	74.6
Both (ours)	88.4	79.7	54.9	75.5
Layers of the ignore band				
Last layer only	88.4	79.3	54.0	75.2
All layers (ours)	88.4	79.7	54.9	75.5
Ranking losses				
Rank only	88.3	79.4	54.0	75.1
Sort only	88.1	79.2	54.2	75.1
Rank + sort (ours)	88.4	79.7	54.9	75.5
Training				
Joint training	79.4	72.0	51.7	68.7
Fine-tuning (ours)	88.4	79.7	54.9	75.5
Alternative				
Post-hoc IoU head ($\gamma = 0.5$)	87.0	78.3	54.2	74.5

TAM Ablation. Tab. 4 varies the three choices that define TAM. *Which cross-attention reads the views.* Giving the views only to the cross-attention that samples inside the segment recovers about half of the gain, and giving them to both adds nothing over the localization cross-attention alone, so the views pay off where boundaries are regressed. *Fixed versus learnable filters.* Learnable filters, initialized to the same Gaussian and window, give 74.5, 0.2 below the fixed filters, so training the filters together with the detector brings no measurable gain and the two fixed filters suffice. *Where the views enter.* Injecting the views after the value projection gives up a third of the gain. Before the projection, each decoder layer maps the views through its own value projection together with the memory. After it, every layer adds the same view term to its values, and the drop shows that the per-layer projection matters.

MAR Ablation. Tab. 5 varies MAR on top of TAM. *Components.* The ignore band alone brings a small gain and the ranking losses alone bring none, yet together they bring more than their sum. While the focal loss keeps pushing well-localized near-duplicates toward zero, the ranking losses have nothing to order, and once the band releases them, the losses decide where they go. *Layers of the ignore band.* Restricting the band to the last layer gives up much of the gain at the strict threshold, which confirms that the earlier layers must release the same proposals. *Ranking losses.* Rank or sort alone reaches 75.1 and the two together reach 75.5. The rank loss orders positives above negatives, the sort loss orders the positives among themselves, and the score follows overlap only when both orderings hold. *Training.* Applying MAR from the start of training fails badly, because the overlap labels are computed from the proposals themselves and early in training those proposals are not worth ordering. *Alternative.* A post-hoc IoU head, trained on top of the TAM detector to predict the overlap of each proposal and used to rescale its score as $s \cdot \hat{\text{IoU}}^{\gamma}$, does not help at any exponent. The result already falls below TAM alone at $\gamma = 0.5$ and degrades to 72.8 as γ grows to 2. Predicting overlap after the fact leaves in place the ordering that the matching installed. Sec. A.3 further compares MAR with IoU-aware losses and tests its sensitivity to hyperparameters.

5 Conclusion

We identify two problems that limit the localization quality of query-based temporal action detectors, transition blindness on the input side of the decoder and confidence mismatch on its supervision side, and propose Seam-DETR to address both. TAM makes transitions explicit in the memory that the localization cross-attention reads, through two parameter-free views and two lightweight adapters. MAR supervises the confidence scores to follow localization quality and adds no parameters or inference cost. Seam-DETR achieves state-of-the-art performance among query-based methods on three benchmarks and gains most at strict tIoU thresholds.

REFERENCES

Tanay Agrawal, Abid Ali, Antitza Dantcheva, and Francois Bremond. Scaling action detection: AdaTAD++ with transformer-enhanced temporal-spatial adaptation. In *Proceedings of the International Conference on Computer Vision*, pp. 12222–12231, 2025.

Humam Alwassel, Fabian Caba Heilbron, Victor Escorcia, and Bernard Ghanem. Diagnosing error in temporal action detectors. In *Proceedings of the European Conference on Computer Vision*, pp. 256–272, 2018.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 961–970, 2015.

Zhi Cai, Songtao Liu, Guodong Wang, Zheng Ge, Xiangyu Zhang, and Di Huang. Align-DETR: Enhancing end-to-end object detection with aligned loss. In *Proceedings of the British Machine Vision Conference*, 2024.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, pp. 213–229, 2020.

Guo Chen, Yifei Huang, Jilan Xu, Baoqi Pei, Jiahao Wang, Zhe Chen, Zhiqi Li, Tong Lu, and Limin Wang. Video Mamba suite: State space model as a versatile alternative for video understanding. *International Journal of Computer Vision*, 134(1):20, 2026.

Kean Chen, Jianguo Li, Weiyao Lin, John See, Ji Wang, Lingyu Duan, Zhibo Chen, Changwei He, and Junni Zou. Towards accurate one-stage object detection with AP-loss. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 5119–5127, 2019.

Qiang Chen, Xiaokang Chen, Jian Wang, Shan Zhang, Kun Yao, Haocheng Feng, Junyu Han, Errui Ding, Gang Zeng, and Jingdong Wang. Group DETR: Fast DETR training with group-wise one-to-many assignment. In *Proceedings of the International Conference on Computer Vision*, pp. 6633–6642, 2023.

Feng Cheng and Gedas Bertasius. TALLFormer: Temporal action localization with a long-memory transformer. In *Proceedings of the European Conference on Computer Vision*, pp. 503–521, 2022.

Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R. Scott, and Weilin Huang. TOOD: Task-aligned one-stage object detection. In *Proceedings of the International Conference on Computer Vision*, pp. 3510–3519, 2021.

Shihua Huang, Zhichao Lu, Xiaodong Cun, Yongjun Yu, Xiao Zhou, and Xi Shen. DEIM: DETR with improved matching for fast convergence. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 15162–15171, 2025.

Haroon Idrees, Amir R. Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The THUMOS challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017.

Zhengjian Kang, Jun Zhuang, Kangtong Mo, Qi Chen, Rui Liu, and Ye Zhang. PaQ-DETR: Learning pattern and quality-aware dynamic queries for object detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 25504–25513, 2026.

Ho-Joong Kim, Jung-Ho Hong, Heejo Kong, and Seong-Whan Lee. TE-TAD: Towards full end-to-end temporal action detection via time-aligned coordinate expression. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 18837–18846, 2024.

Ho-Joong Kim, Yearang Lee, Jung-Ho Hong, and Seong-Whan Lee. DiGIT: Multi-dilated gated encoder and central-adjacent region integrated decoder for temporal action detection transformer. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 24286–24296, 2025a.

Jihwan Kim, Miso Lee, and Jae-Pil Heo. Self-feedback DETR for temporal action detection. In *Proceedings of the International Conference on Computer Vision*, pp. 10286–10296, 2023.

Jihwan Kim, Miso Lee, Cheol-Ho Cho, Jihyun Lee, and Jae-Pil Heo. Prediction-feedback DETR for temporal action detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 4266–4274, 2025b.

Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21002–21012, 2020.

Chuming Lin, Chengming Xu, Donghao Luo, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Learning salient boundary feature for anchor-free temporal action localization. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 3320–3329, 2021.

Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. BSN: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision*, pp. 3–19, 2018.

Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. BMN: Boundary-matching network for temporal action proposal generation. In *Proceedings of the International Conference on Computer Vision*, pp. 3889–3898, 2019.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the International Conference on Computer Vision*, pp. 2980–2988, 2017.

Shuming Liu, Chen-Lin Zhang, Chen Zhao, and Bernard Ghanem. End-to-end temporal action detection with 1B parameters across 1000 frames. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 18591–18601, 2024.

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

Kemal Oksuz, Baris Can Cam, Emre Akbas, and Sinan Kalkan. Rank & sort loss for object detection and instance segmentation. In *Proceedings of the International Conference on Computer Vision*, pp. 3009–3018, 2021.

Yifan Pu, Weicong Liang, Yiduo Hao, Yuhui Yuan, Yukang Yang, Chao Zhang, Han Hu, and Gao Huang. Rank-DETR for high quality object detection. In *Advances in Neural Information Processing Systems*, volume 36, pp. 16100–16113, 2023.

Zhiwu Qing, Haisheng Su, Weihao Gan, Dongliang Wang, Wei Wu, Xiang Wang, Yu Qiao, Junjie Yan, Changxin Gao, and Nong Sang. Temporal context aggregation network for temporal action proposal refinement. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 485–494, 2021.

Jiayi Shao, Xiaohan Wang, Ruijie Quan, Junjun Zheng, Jiang Yang, and Yi Yang. Action sensitivity learning for temporal action localization. In *Proceedings of the International Conference on Computer Vision*, pp. 13457–13469, 2023.

Dingfeng Shi, Yujie Zhong, Qiong Cao, Jing Zhang, Lin Ma, Jia Li, and Dacheng Tao. ReAct: Temporal action detection with relational queries. In *Proceedings of the European Conference on Computer Vision*, pp. 105–121, 2022.

Dingfeng Shi, Yujie Zhong, Qiong Cao, Lin Ma, Jia Li, and Dacheng Tao. TriDet: Temporal action detection with relative boundary modeling. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 18857–18866, 2023.

Jing Tan, Jiaqi Tang, Limin Wang, and Gangshan Wu. Relaxed transformer decoders for direct action proposal generation. In *Proceedings of the International Conference on Computer Vision*, pp. 13526–13535, 2021.

Xin Tong, Baojie Tian, Yufei Guo, and Zhe Ma. RANK++LETR: Learn to rank and optimize candidates for line segment detection. In *Advances in Neural Information Processing Systems*, volume 38, pp. 41739–41757, 2025.

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, Tianxiang Jiang, Songze Li, Jilan Xu, Hongjie Zhang, Yifei Huang, Yu Qiao, Yali Wang, and Limin Wang. InternVideo2: Scaling foundation models for multimodal video understanding. In *Proceedings of the European Conference on Computer Vision*, pp. 396–416, 2024.

Mengmeng Xu, Chen Zhao, David S. Rojas, Ali Thabet, and Bernard Ghanem. G-TAD: Sub-graph localization for temporal action detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 10156–10165, 2020.

Le Yang, Ziwei Zheng, Yizeng Han, Hao Cheng, Shiji Song, Gao Huang, and Fan Li. DyFADet: Dynamic feature aggregation for temporal action detection. In *Proceedings of the European Conference on Computer Vision*, pp. 305–322, 2024a.

Min Yang, Huan Gao, Ping Guo, and Limin Wang. Adapting short-term transformers for action detection in untrimmed videos. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 18570–18579, 2024b.

Chen-Lin Zhang, Jianxin Wu, and Yin Li. ActionFormer: Localizing moments of actions with transformers. In *Proceedings of the European Conference on Computer Vision*, pp. 492–510, 2022.

Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sünderhauf. VarifocalNet: An IoU-aware dense object detector. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 8514–8523, 2021.

Hang Zhao, Antonio Torralba, Lorenzo Torresani, and Zhicheng Yan. HACS: Human action clips and segments dataset for recognition and temporal localization. In *Proceedings of the International Conference on Computer Vision*, pp. 8668–8678, 2019.

Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. DETRs beat YOLOs on real-time object detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 16965–16974, 2024.

Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the International Conference on Computer Vision*, pp. 2914–2923, 2017.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021.

Yuhan Zhu, Guozhen Zhang, Jing Tan, Gangshan Wu, and Limin Wang. Dual DETRs for multi-label temporal action detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 18559–18569, 2024.

Zhuofan Zong, Guanglu Song, and Yu Liu. DETRs with collaborative hybrid assignments training. In *Proceedings of the International Conference on Computer Vision*, pp. 6748–6758, 2023.

A APPENDIX

This appendix reports the settings behind the experiments and the analyses that the main text refers to. Sec. A.1 gives the datasets, the evaluation protocol, the implementation on each dataset, and how the two scales of TAM are set. Sec. A.2 describes how the two problems of Sec. 3.2 are measured, repeats the measurements for each component, and adds an error analysis. Sec. A.3 adds ablations of the views of TAM and of alternatives to MAR, and a sensitivity analysis of MAR. Sec. A.4 adds TAM and MAR to two further query-based detectors, Sec. A.5 reports the inference cost, and Sec. A.6 shows what the two views respond to and two detection examples.

A.1 EXPERIMENTAL DETAILS

Datasets. We conduct experiments on THUMOS14 (Idrees et al., 2017), ActivityNet-1.3 (Caba Heilbron et al., 2015), and HACS-Segment (Zhao et al., 2019). THUMOS14 covers 20 sports action classes, and its 200 validation and 213 test videos contain 3,007 and 3,358 action instances, respectively. Following convention, we train on the validation set and evaluate on the test set. ActivityNet-1.3 contains 200 classes of daily activities with 10,024 training and 4,926 validation videos. HACS-Segment shares the 200 classes of ActivityNet-1.3 and provides 37,613 training and 5,981 validation videos. The test annotations of the latter two datasets are withheld, so we train on their training sets and evaluate on their validation sets. Actions in THUMOS14 last 5 seconds on average and each video holds about 15 of them, whereas actions in ActivityNet-1.3 and HACS-Segment last 48 and 33 seconds on average.

Evaluation Metric. We follow the standard protocol and report mean average precision (mAP) at several tIoU thresholds. On THUMOS14, the thresholds are [0.3:0.7:0.1]. On ActivityNet-1.3 and HACS-Segment, we report mAP at $\{0.5, 0.75, 0.95\}$ and the average mAP over [0.5:0.95:0.05]. Unless noted otherwise, mAP refers to the average over thresholds.

Implementation Details. Architecture. We build on DiGIT (Kim et al., 2025a) and keep its architecture, including the pre-extracted InternVideo2-6B (Wang et al., 2024) features, the six-layer encoder and decoder, and the number of queries on each dataset. The change view uses $\sigma = 4$ tokens on THUMOS14 and ActivityNet-1.3 and $\sigma = 2$ tokens on HACS-Segment. The direction view averages $W = 15$ tokens on each side. Both views account for padding. The Gaussian filter is normalized over the valid tokens, and the direction view averages each side over its valid tokens and is set to zero when either side has none.

Training. We train with AdamW without weight decay and evaluate the exponential moving average of the weights. On THUMOS14, the first stage runs for 60 epochs with a learning rate of 5×10^{-5} and a batch size of 4. The second stage runs for 20 epochs with a learning rate of 10^{-5}, $\lambda = 0.4$, $\tau_p = 0.5$, $\tau_n = 0.3$, and $\delta = 0.5$, and η_{rank} and η_{sort} reach 1 after 5 epochs. The ignore band also applies to the proposal classification loss of the encoder. On ActivityNet-1.3 and HACS-Segment, the first stage runs for 10 and 13 epochs, respectively, with a learning rate of 10^{-4} and a batch size of 16, and the second stage runs for 5 epochs with the same learning rate. On these two datasets, the weight w_{qc} of an unmatched proposal decreases linearly from 1 at $l_{qc} = 0.5$ to 0 at $l_{qc} = 0.95$ instead of dropping to zero at λ.

Inference. On THUMOS14 and HACS-Segment, we apply class-wise NMS with thresholds of 0.65 and 0.75, respectively. Following the previous method (Kim et al., 2025a), on HACS-Segment, we keep only the detections of the classes whose external video-level classification score (Wang et al., 2024) exceeds 0.05; on ActivityNet-1.3, we follow the binary protocol, in which the detector predicts class-agnostic segments that take their classes from the external classification scores. We combine the two scores as $p_{\text{det}} \cdot p_{\text{cls}}^{0.6}$ and apply soft-NMS with $\sigma = 1.6$.

Setting σ and W. The two scales are set per dataset on the base detector before TAM is trained. A token spans 0.13 s on THUMOS14 and 0.27 s on ActivityNet-1.3 and HACS-Segment, whose features are extracted with twice the stride. On THUMOS14 the Gaussian of the change view follows a cutoff period of four seconds, which gives $\sigma = 4$ tokens, and HACS-Segment keeps that cutoff with $\sigma = 2$ tokens. ActivityNet-1.3, whose instances are the longest of the three, doubles the cutoff to eight seconds, again $\sigma = 4$ tokens. For the direction view we measure two quantities

on the memory of the base detector, how often the magnitude of the view at a predicted boundary is larger for the better-localized of two overlapping proposals, and how strongly it correlates with the magnitude of the change view. Fig. 5 plots both against the window on THUMOS14. The discrimination falls slowly as the window grows, from 0.72 at half a second to 0.69 at two seconds, while the correlation falls from 0.58 to 0.04 and becomes negative for longer windows. We take $W = 15$ tokens, two seconds, the shortest window at which the two views carry almost independent responses.

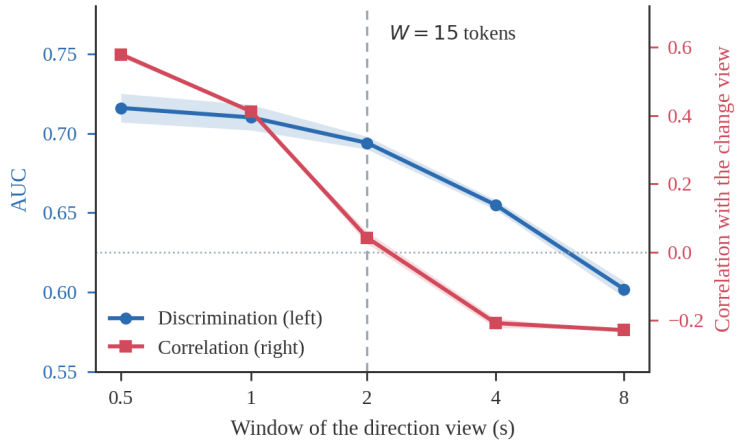


Figure 5: Setting the window of the direction view on THUMOS14. AUC is how often the magnitude of the direction view at a predicted boundary is larger for the better-localized of two overlapping proposals. Correlation is with the magnitude of the change view. Bands span the values at the start and at the end of an instance, and the dashed line marks the window we use.

A.2 Diagnostic Analysis

Measurement Details. The measurements of Fig. 3 use the THUMOS14 test videos that fit in the 4096-token window of the base detector, which gives 195 videos, 4957 annotated boundaries, and 2925 random time steps as the control. Each window spans four seconds, that is, 15 time steps of 0.133 s on each side of its center. The memory is the finest-scale input of the decoder cross-attention, taken after its projection and normalization. The transition duration of a boundary, used in Fig. 1(a) and Tab. 6, is measured on the video features. We project the features around the boundary onto the direction from the mean feature 0.4–2 s before it to the mean feature 0.4–2 s after it, and take the time that the projection needs to go from 10% to 90% of the way. Tab. 6 repeats the measurement for each sixth of the boundaries sorted by this duration, for the base detector and for TE-TAD (Kim et al., 2024), whose 2304-token window holds 178 videos. The boundary mark in a window is placed between the two adjacent time steps with the lowest cosine similarity, and the Offset and Within 0.3 s columns compare this position with the annotated boundary.

Table 6: Boundary marks in the memory by transition duration on THUMOS14 test. Rise is the median transition duration in each sixth. Cross is the mean cosine similarity between the two sides of the boundary window. Offset is the median distance in seconds between the boundary mark and the annotated boundary, and Within 0.3 s is the fraction of boundaries whose mark lies within 0.3 s of the annotation. The last column gives this fraction for the video features.

		Base detector			TE-TAD			Features
Sixth	Rise (s)	Cross	Offset	Within 0.3 s	Cross	Offset	Within 0.3 s	Within 0.3 s
1 (abrupt)	0.67	0.03	0.40	45%	0.23	0.67	31%	35%
2	0.80	0.00	0.27	53%	0.24	0.53	31%	43%
3	0.93	0.01	0.40	45%	0.24	0.53	35%	32%
4	1.07	0.02	0.40	43%	0.25	0.67	32%	24%
5	1.33	0.07	0.67	27%	0.29	0.80	25%	13%
6 (gradual)	1.73	0.16	0.93	17%	0.38	0.93	18%	7%

Probes on Each Component. Tab. 7 repeats the two probes of Fig. 1 on the base detector, on each component alone, and on the full model. TAM raises the recall of tight proposals at the most gradual boundaries from 0.38 to 0.46 and leaves the two scores nearly unchanged. MAR raises the score of matched proposals from 0.69 to 0.83 and that of their tight near-duplicates from 0.14 to 0.37, so the score of a well-fitting proposal depends less on whether the matching assigned it, while the recall changes little. Each component thus acts on the problem it was designed for, and the full model keeps both changes.

Table 7: The two probes of Fig. 1 for each component on THUMOS14 test. Recall is the fraction of boundaries in the most gradual sixth whose instance has a same-class proposal with IoU ≥ 0.9 before NMS. Matched and Near-duplicates are the mean scores of proposals with IoU ≥ 0.8 that the matching assigns to the instance and of those it leaves unassigned.

	Transition blindness	Confidence mismatch	
Method	Recall	Matched	Near-duplicates
Base	0.38	0.69	0.14
+ TAM	0.46	0.68	0.13
+ MAR	0.40	0.83	0.37
+ TAM + MAR	0.46	0.79	0.31

Score Distributions after MAR. Fig. 6 repeats Fig. 4 for the full model next to the base detector. Among tight proposals, the near-duplicates of the base detector concentrate near zero, whereas in the full model their scores spread over the whole range and the matched proposals move up, so the two distributions overlap more. Among loose proposals, the near-duplicates stay low in both models. MAR therefore does not lift every unmatched proposal but only those that fit the instance well.

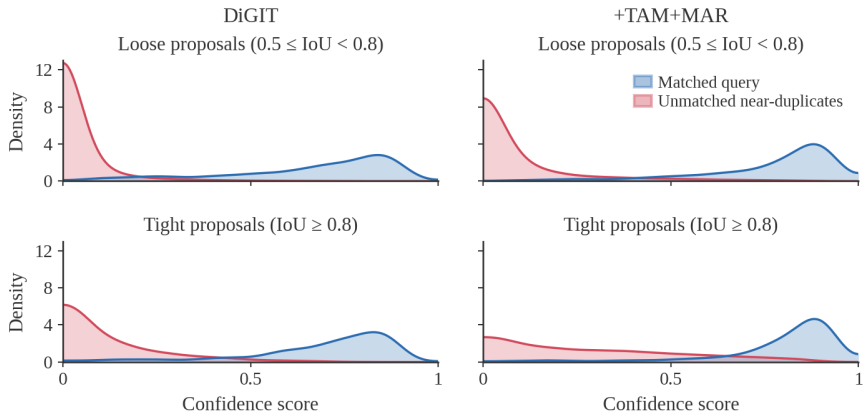
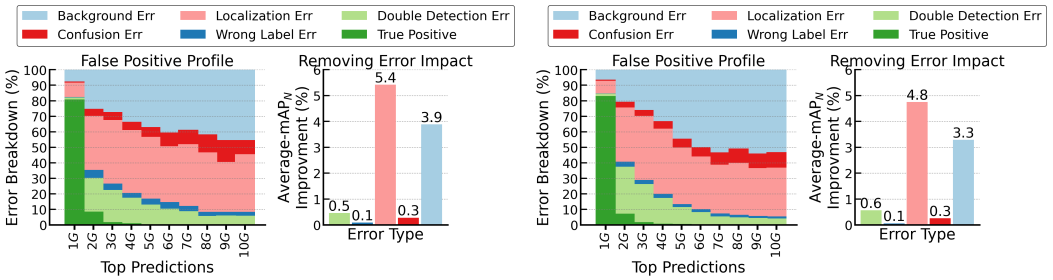


Figure 6: Confidence score distributions of matched queries and of their unmatched near-duplicates, for loose and tight proposals, on the base detector (left) and the full model (right), under the training-time matching on the THUMOS14 test videos.

Error Analysis. Fig. 7 shows the false-positive analysis of DETAD (Alwassel et al., 2018) on THUMOS14 for the base detector and the full model. Among the top-1G predictions, the share of true positives rises from 81% to 83%. Localization and background errors are by far the two largest error types, and both shrink. Removing them would raise the normalized average mAP (average-mAP$_N$) of the base detector by 5.4% and 3.9% and that of the full model by 4.8% and 3.3%. This is in line with the tighter proposals of TAM and with the rank loss of MAR, which places proposals that overlap an instance above background ones. The gain from removing double-detection errors grows slightly, from 0.5% to 0.6%, which is consistent with MAR raising the scores of near-duplicates.



(a) DiGIT (b) Seam-DETR

Figure 7: DETAD false-positive analysis on THUMOS14 at tIoU 0.5 for (a) the base detector and (b) the full model. G is the number of ground-truth instances.

A.3 Additional Ablations

Views of TAM. Tab. 8 keeps one view of TAM at a time. The last setting instead feeds the memory itself to the two adapters in place of the views. Either view alone improves over the base detector, the change view more than the direction view, and the two together improve over either. The two adapters with the memory alone leave the accuracy at the base level, so the gain comes from the views themselves.

Alternatives to MAR. Tab. 9 replaces MAR with two classification losses that tie the score to localization quality, VFL (Zhang et al., 2021) and the loss of Align-DETR (Cai et al., 2024). For Align-DETR we take only the loss and keep one-to-one matching. Both are applied in the same fine-tuning stage as MAR, starting from the TAM detector. Neither of the two losses moves the average mAP by more than 0.3, while MAR adds 0.8. Both losses set the target of the matched proposal by its overlap but keep every unmatched proposal as background, so the mean score of the tight near-duplicates stays at 0.10 and 0.09, against 0.13 before fine-tuning, whereas MAR raises it to 0.31. MAR thus acts where these losses do not, on the unmatched proposals.

Sensitivity of MAR. Fig. 8 changes one hyperparameter of MAR at a time from its default and fine-tunes the TAM detector once for each value. Across the 16 settings, the average mAP stays within 0.21 of the default, and the default is the best or on par with the best in every panel. Only the two smallest half-widths of the soft step, $\delta = 0.125$ and 0.25, cost about 0.2.

Table 8: Views of TAM on THUMOS14.

Setting	0.3	0.4	0.5	0.6	0.7	Avg.
Base	86.9	83.0	76.0	65.4	52.7	72.8
Change view only	87.2	83.8	78.2	68.3	53.1	74.1
Direction view only	87.0	83.2	77.3	67.8	52.8	73.6
Adapters only	87.3	83.3	76.7	65.6	51.8	72.9
Both views (ours)	88.0	84.6	78.8	68.8	53.5	74.7

Table 9: Alternatives to MAR on THUMOS14. Matched and Near-duplicates are the mean scores of proposals with IoU ≥ 0.8, as in Tab. 7.

Setting	0.3	0.5	0.7	Avg.	Matched	Near-duplicates
TAM w/o MAR	88.0	78.8	53.5	74.7	0.68	0.13
VFL (Zhang et al., 2021)	87.9	78.6	53.2	74.4	0.70	0.10
Align-DETR loss (Cai et al., 2024)	88.1	78.8	53.9	74.8	0.69	0.09
MAR (ours)	88.4	79.7	54.9	75.5	0.79	0.31

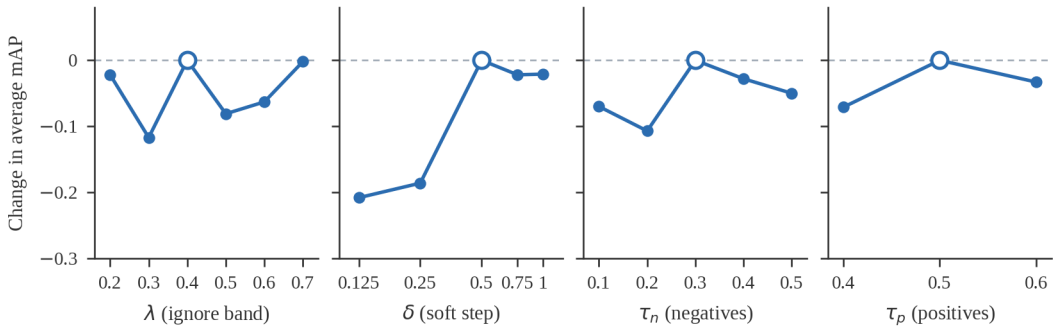


Figure 8: Sensitivity of MAR. Each point changes one hyperparameter from the default (hollow) and fine-tunes the TAM detector once. The vertical axis gives the change in average mAP.

A.4 Other Query-Based Detectors

TAM and MAR are meant for query-based detectors in general, so we add both to two further detectors on THUMOS14 with the same InternVideo2 features. TadTR (Liu et al., 2022) and TE-TAD (Kim et al., 2024) each have a single cross-attention, so TAM feeds the localization stream by the second read described in Sec. 3.3, and MAR then fine-tunes the detector as on DiGIT. Tab. 10 compares our reproduction of each detector with the same detector trained with TAM and fine-tuned with MAR. Both improve, by 2.2 on TadTR and 1.1 on TE-TAD, and both gain most at the strict thresholds, as on DiGIT. The gain is smaller on TE-TAD, whose memory marks boundaries less sharply than that of the base detector, as the higher cross-window similarity and the lower share of marks within 0.3 s in Tab. 6 show, so the two views of TAM have less change to expose.

Table 10: TAM and MAR on two further query-based detectors. Each detector is our reproduction.

Method	Feature	0.3	0.4	0.5	0.6	0.7	Avg.
TadTR (Liu et al., 2022)	InternVideo2	82.0	76.0	68.5	55.6	41.7	64.7
+ TAM + MAR	InternVideo2	81.3	77.4	70.8	60.7	44.4	**66.9**
TE-TAD (Kim et al., 2024)	InternVideo2	85.0	80.7	73.8	63.3	48.5	70.3
+ TAM + MAR	InternVideo2	86.2	81.3	74.6	64.4	50.4	**71.4**

The Two Problems in Other Detectors. Fig. 1 shows that TadTR and TE-TAD suffer from the same two problems as DiGIT. Tab. 11 repeats the two probes on these detectors with TAM and MAR added. In both, the two components raise the recall at the gradual end and the scores of near-duplicates, as on DiGIT in Tab. 7. TadTR scores its near-duplicates lowest, even though its score includes an actionness term trained to predict overlap.

Table 11: The two probes of Fig. 1 on TadTR and TE-TAD with and without TAM and MAR, on THUMOS14 test, defined as in Tab. 7.

	Transition blindness	Confidence mismatch	
Method	Recall	Matched	Near-duplicates
TadTR (Liu et al., 2022)	0.25	0.65	0.03
+ TAM + MAR	0.30	0.79	0.54
TE-TAD (Kim et al., 2024)	0.38	0.77	0.11
+ TAM + MAR	0.45	0.88	0.27

A.5 Computational Cost

Tab. 12 compares the inference cost on THUMOS14. TAM adds 1.05M parameters (0.5%) to DiGIT and 0.26M parameters to TE-TAD and TadTR, whose single cross-attention reads the localization memory in a second pass. It raises the detector time by 12% on DiGIT, 24% on TE-TAD and

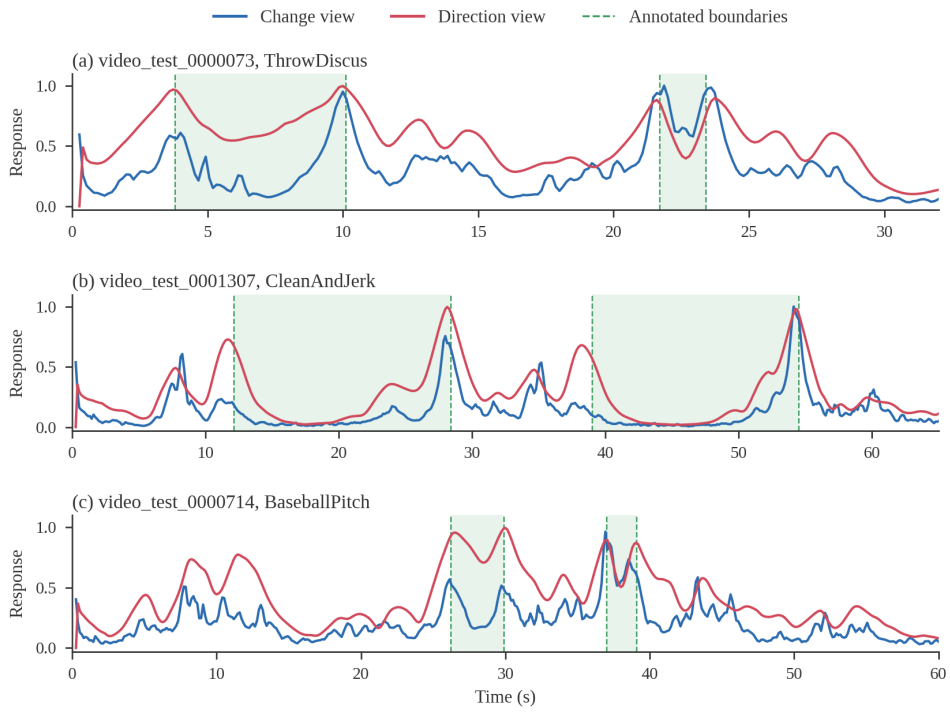


Figure 9: The change view and the direction view of TAM on three THUMOS14 test videos. Each curve is the magnitude of the view over time, scaled to its maximum within the video; shaded intervals are annotated instances.

34% on TadTR, since the cost of the views and of the second read is nearly fixed while the three detectors differ in size. Post-processing time and peak memory change little, and MAR adds no cost at inference. Seam-DETR remains faster than the anchor-free detectors and TadTR.

Table 12: Inference cost per video on THUMOS14 test with InternVideo2 features.

Method	Inference time (ms)			Peak memory
	Detector	Post-process	Total	(GB)
Anchor-free				
ActionFormer (Zhang et al., 2022)	42.47	31.75	74.22	3.04
ActionMamba (Chen et al., 2026)	28.76	31.75	60.51	2.88
Query-based				
TadTR (Liu et al., 2022)	92.07	17.39	109.46	1.24
TadTR + Ours	123.28	17.62	140.90	1.24
TE-TAD (Kim et al., 2024)	27.71	5.61	33.32	2.48
TE-TAD + Ours	34.44	5.41	39.85	2.74
DiGIT (Kim et al., 2025a)	31.27	8.08	39.35	3.52
Seam-DETR	35.11	8.61	43.72	3.68

A.6 Qualitative Analysis

Where the Views Respond. Fig. 9 plots the magnitude of the two views of TAM on three test videos, with the annotated instances shaded. Both views respond near the annotated boundaries. The change view is sharp and reacts to any local change, inside instances as well as at their boundaries. The direction view responds over a wider span and still responds where the change view is weak, for example at the start of the first instance in (a) and at the start of each instance in (b). The two views therefore mark a transition in different ways, which is why TAM uses both.

Detection Examples. Fig. 10 shows one example for each problem. In (a) the instance begins with a gradual transition of about 2 s. The best candidate of the base detector starts 0.9 s after the annotated start, at IoU 0.69. The full model produces a candidate that starts within 0.2 s of the annotated start, at IoU 0.85. In (b) the base detector gives its two highest scores, 0.62 and 0.56, to candidates that start about 2 s late, at IoU 0.54, and scores its two candidates at IoU 0.86 lower, at 0.37 and 0.22. The full model gives its highest score, 0.89, to a candidate at IoU 0.96.

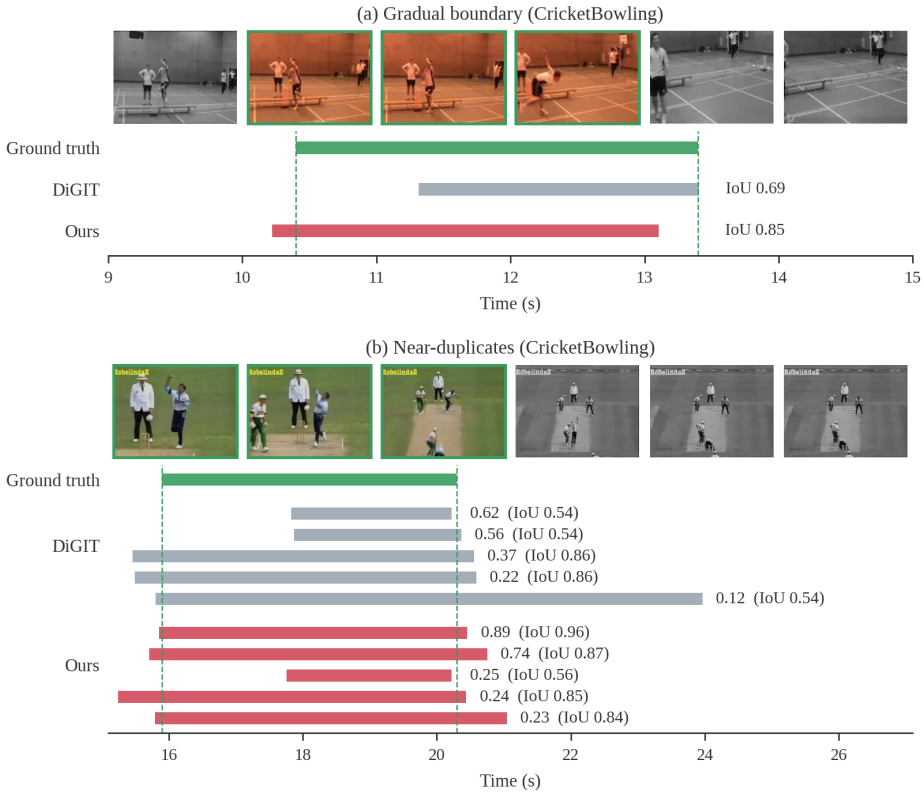


Figure 10: Two examples from THUMOS14 test. DiGIT is the base detector and Ours is the full model. (a) An instance with a gradual start and the best-fitting candidate of each model before NMS, labeled with its IoU. (b) For each model, the five highest-scoring same-class candidates before NMS among those at IoU above 0.5, labeled with their score and IoU. Frames above each panel are sampled at uniform intervals, and frames inside the instance are in color.

A.7 LLM USAGE

We used LLMs to assist with language editing and polishing, including improving clarity, concision, and wording. We also used LLMs for literature retrieval and discovery to identify potentially relevant related work. All suggested references were independently verified against the original sources. We take responsibility for the final content of this work.