MULTI-SIGNAL CASCADED GROUNDING FOR AUDIO MOMENT RETRIEVAL FROM LONG AUDIO

Technical Report

Seungdeok Choi, Yong-Hwa Park

Korean Advanced Institute of Science and Technology
Mechanical Engineering
Daejeon, South Korea
{haroldchoi6,yhpark}@kaist.ac.kr

ABSTRACT

We describe our submission to DCASE 2026 Task 6, audio moment retrieval (AMR) from long audio. Building on the query-dependent DETR (QD-DETR) detection-transformer baseline, we replace the MS-CLAP feature extractor with frozen M2D-CLAP embeddings and augment the detector with a sequence of complementary, training-only supervision signals: a multi-resolution coarse auxiliary branch, a span-query re-ranking regularizer, a per-boundary InfoNCE contrast, and a DN-DETR denoising task, together with a bidirectional Mamba audio encoder. A lightweight cascade refinement decoder then adds a second, localized detection stage that reads per-frame audio crops around each first-stage prediction and emits bounded boundary corrections. At inference we combine a two-seed score-level ensemble with a saliency-as-proposals stream that repurposes the saliency head as a parallel moment generator. Our four submitted systems form a monotone progression on the development-testing split, the strongest reaching Recall1@0.7 = 33.70, a large gain over the official baseline.

***Index Terms*—** audio moment retrieval, cross-modal grounding, detection transformer, state-space models, contrastive learning

1. INTRODUCTION

Audio moment retrieval (AMR) localizes the start/end timestamps of the moments in a long audio recording that match a free-form natural-language query [1]. Unlike clip-level audio–text retrieval, AMR must reason over minutes of audio and produce temporally precise boundaries, which makes long-range sequence modeling and fine-grained cross-modal alignment the central difficulties of DCASE 2026 Task 6 [2]. The task is evaluated with recall and mean-average-precision (mAP) under an intersection-over-union (IoU) criterion; the primary ranking metric is Recall1@0.7, the fraction of queries whose single most confident predicted moment overlaps a ground-truth moment with IoU ≥ 0.7.

The official baseline casts AMR as set prediction with a DETR-style detector [3, 4], specifically QD-DETR [5], on top of MS-CLAP sliding-window features [6]. We retain this detection formulation but make three sets of changes. First, we upgrade the frozen feature extractor to M2D-CLAP [7] and replace the audio self-attention encoder with a bidirectional Mamba state-space block [8]. Second, we add four training-only supervision channels — a coarse multi-resolution branch, a span-query re-ranking head, a per-boundary InfoNCE contrast [9], and a DN-DETR denoising task [10] — none of which alters the inference graph. Third, we introduce a cascade refinement decoder that performs a localized second detection stage, and two inference-time aggregation schemes (multi-seed ensembling and saliency-as-proposals).

We submit four systems that layer these components incrementally (Section 4), so that each slot is a strict superset of the previous one and the contribution of every component can be read directly from the progression. All four share the backbone of Section 2; the added components are described in Section 3.

Replace with your funding acknowledgment, if any.

2. SHARED BACKBONE

2.1. Feature extractor

All systems consume pre-extracted, *frozen* M2D-CLAP features [7]. Audio is processed with sliding one-second windows at 16 kHz and projected to a 768-dimensional CLAP embedding per second; the text query is encoded token-wise with the model's BERT-based text branch [11] to a sequence of 768-dimensional embeddings (the unpooled `last_hidden_state`). The audio sequence is concatenated with two-dimensional temporal endpoint features (TEF), i.e. the normalized start/end position of each frame, before projection. Because the features are pre-computed, the M2D-CLAP backbone contributes no trainable parameters.

2.2. QD-DETR detector

The detector follows QD-DETR [5], common to all four slots. Audio and text are each mapped to a hidden width $d = 256$ by a two-layer projection with input dropout 0.5. A two-layer text-to-video (T2V) cross-attention encoder injects the query into the audio stream, after which a two-layer audio encoder produces the audio memory $M \in \mathbb{R}^{L\times d}$. A two-layer conditional-DETR decoder [12] with $N_q = 10$ learnable queries predicts, for each query, a moment as a normalized center–width pair $(c, w) \in [0, 1]^2$ decoded as a residual to a reference point [13], together with a foreground/background logit. Two saliency projections produce a per-frame relevance score used by the saliency loss. Multi-head attention uses 8 heads and a feed-forward width of 1024; sinusoidal position encoding is used throughout.

We replace the audio self-attention encoder with a *single* bidirectional Mamba block [8]: a forward selective state-space scan and

a reverse-sequence scan are summed and layer-normalized (state size 16, local convolution width 4, expansion 2). The selective state-space primitive provides content-dependent, linear-time temporal mixing while preserving boundary-localized information; we found (Section 4.3) that a single scan is preferable to deeper stacks, which over-smooth position-specific token content.

2.3. Base training objective

Predictions are matched to ground-truth moments by Hungarian assignment, and matched pairs are supervised by an L_1 term and a generalized-IoU term [14] on (c, w), a cross-entropy term on the foreground/background class, and an InfoNCE-style per-frame saliency contrast:

$$\mathcal{L}_{\text{det}} = \lambda_1 \mathcal{L}_{L_1} + \lambda_g \mathcal{L}_{\text{gIoU}} + \lambda_c \mathcal{L}_{\text{cls}} + \lambda_s \mathcal{L}_{\text{sal}}, \tag{1}$$

with $(\lambda_1, \lambda_g, \lambda_c, \lambda_s) = (10, 1, 4, 1)$, a background class weight of 0.1, and a saliency margin of 0.2. A DETR auxiliary loss is applied to the intermediate decoder layer.

2.4. Training protocol

We use a two-stage schedule: a 200-epoch Clotho-Moment pre-training run followed by a 200-epoch CASTELLA fine-tune. Optimization uses AdamW [15] (learning rate 10^{-4} for pre-training, 8×10^{-5} for fine-tuning; weight decay 10^{-4}; gradient clipping 0.1), batch size 32, at most 5 windows per clip, audio length capped at $L = 300$ frames and queries at 32 tokens. The default seed is 2023; slot 4 additionally trains a second seed (42).

3. SUBMISSION COMPONENTS

The four submitted systems add components cumulatively. Slot 1 adds two auxiliary channels to the backbone; slot 2 adds the Mamba encoder and two further channels; slot 3 adds the cascade decoder; slot 4 adds two inference-time aggregation schemes.

3.1. Coarse auxiliary branch (slot 1)

To force the encoder toward representations that are stable across temporal resolutions, a training-only branch average-pools the projected audio $2\times$ (kernel/stride 2), recomputes TEF on the coarse grid, and runs the *shared* encoder and heads on the half-length sequence, producing coarse moments supervised by the same Hungarian objective scaled by γ_c:

$$\mathcal{L}_{\text{coarse}} = \gamma_c \, \mathcal{L}_{\text{det}}\big(\text{pool}_2(M)\big), \qquad \gamma_c = 0.25. \tag{2}$$

The branch shares all weights with the main path and is discarded at inference. The coefficient has a U-shaped optimum at 0.25 (Section 4.3).

3.2. Span-query re-ranking (slot 1)

For each matched query i with predicted span s_i, we mean-pool the audio memory over s_i (detached), project and ℓ_2-normalize it to g_i, and compute a cosine similarity with the ℓ_2-normalized mean-pooled projected text q. A binary cross-entropy term against the foreground/background label y_i supervises the projections:

$$\mathcal{L}_{\text{rr}} = \text{BCE}\big(\tfrac{1}{2}\big(\langle g_i, q\rangle + 1\big),\, y_i\big)\,. \tag{3}$$

At inference the foreground score may be blended as $\alpha\, p_{\text{fg}} + (1-\alpha)\,\frac{1}{2}(\langle g_i, q\rangle + 1)$; with $\alpha = 1$ the blend is a pass-through and Eq. (3) acts purely as a training-time encoder regularizer (weight 0.5).

3.3. Boundary contrast (slot 2)

To sharpen the audio representation across event boundaries — where the information for IoU ≥ 0.7 lives — we add a per-boundary InfoNCE term [9]. For each ground-truth boundary side $b \in \{\text{start}, \text{end}\}$ we take the $K = 3$ frames just inside the boundary (set $\mathcal{I}_b$) and the K frames just outside (set $\mathcal{O}_b$). With a text projection q and audio projections $\tilde{a}_t$ (ℓ_2-normalized), and the normalized inside pool $\bar{a}_b = \text{norm}\big(\frac{1}{|\mathcal{I}_b|}\sum_{t\in\mathcal{I}_b}\tilde{a}_t\big)$,

$$\mathcal{L}_{\text{BC}} = -\frac{1}{|\mathcal{B}|}\sum_{b\in\mathcal{B}} \log \frac{\exp(\langle q, \bar{a}_b\rangle/\tau)}{\exp(\langle q, \bar{a}_b\rangle/\tau) + \sum_{t\in\mathcal{O}_b}\exp(\langle q, \tilde{a}_t\rangle/\tau)}, \tag{4}$$

with temperature $\tau = 0.1$ and weight 0.2; $\mathcal{B}$ ranges over all boundary sides of all ground-truth spans. The term touches neither the decoder nor the inference graph.

3.4. DN-DETR denoising (slot 2)

We add a DN-DETR denoising task [10] to stabilize bipartite matching. For each ground-truth span (c, w) we generate $M = 5$ noised copies

$$\tilde{c} = \text{clip}(c + \varepsilon_c), \quad \tilde{w} = \text{clip}(w + \varepsilon_w), \quad \varepsilon_{\{c,w\}} \sim \mathcal{N}(0, \sigma^2), \tag{5}$$

with $\sigma = 0.05$, which enter the decoder as extra queries (after an inverse-sigmoid reference-point transform). An attention mask isolates the denoising queries from the learnable queries, and each denoising query is supervised one-to-one (no Hungarian assignment) to reconstruct its source span. The task is removed at inference (weight 1.0). The noise scale has a U-shaped optimum at $\sigma = 0.05$ (Section 4.3).

3.5. Cascade refinement decoder (slot 3)

Slot 3 adds a second detection stage that refines each first-stage moment from a localized audio crop. The stage-1 last-layer query state h_i and span $s_i = (c_i, w_i)$ are *detached*; we convert s_i to a frame interval, widen it by a margin $m = 10$ frames, and gather the corresponding crop of the audio memory with an intra-crop learnable relative position embedding. A single cross-attention+FFN layer (4 heads) reads this crop with h_i as query, and a zero-initialized head emits a bounded residual:

$$\delta_i = \kappa\, \tanh\big(W_\delta\, \tilde{h}_i\big), \qquad s_i' = \text{clip}(s_i + \delta_i), \qquad \kappa = 0.05. \tag{6}$$

Because W_δ is zero-initialized, $\delta_i = 0$ at step 0 and the refinement starts *exactly* from the identity, preserving stage-1 quality; detaching the inputs prevents cascade gradients from disturbing the encoder/decoder. Both stage-1 and refined spans are supervised by $L_1 + \text{gIoU}$ under the same Hungarian indices (weight 1.0), and the refined spans are used at inference. The bound $\kappa = 0.05$ ($\pm5\%$ of the timeline, ample for boundary corrections) is critical: a looser $\kappa = 0.1$ occasionally over-corrects the top-1 prediction and regresses recall, while a tighter $\kappa = 0.03$ is too constrained (Section 4.3). This design differs from a span-pooled boundary-regression head in that it reads *per-frame* audio signal at the boundary with fresh positional encoding.

Table 1: Results on the development-testing split (CASTELLA test). Baseline is the official QD-DETR with MS-CLAP features (CASTELLA+Clotho-Moment); all our systems use frozen M2D-CLAP features. Best per column in **bold**.

System	R1@.5	**R1@.7**	mAP	m@.5	m@.75
Baseline [1]	25.61	13.59	12.06	23.60	10.72
Slot 1: M2D base	43.06	28.36	23.36	38.86	22.56
Slot 2: +Mamba+BC+DN	47.07	29.55	24.30	41.27	23.45
Slot 3: +cascade (κ=.05)	48.40	31.03	25.82	43.33	24.94
Slot 4: +ens.+saliency	**50.41**	**33.70**	**29.16**	**44.77**	**28.47**

3.6. Inference-time aggregation (slot 4)

Slot 4 applies two orthogonal aggregation schemes on top of slot 3, with no additional training beyond a second seed.

Two-seed score-level ensemble. Two slot-3 models (seeds 2023 and 42) each emit N_q (span, p_{fg}) tuples; the $2N_q$ tuples are pooled and re-sorted by p_{fg}. Spans are *not* averaged: decoder slots are not semantically aligned across independently trained models, so averaging would produce midpoints matching no real event. This is pure variance reduction.

Saliency-as-proposals. The saliency head (trained by the saliency loss in Eq. (1) and normally discarded at inference) is repurposed as a parallel decoding stream. Per frame, we take $\sigma(r_t)$ from the saliency logit r_t, rescale per sample as $\rho_t = \sigma(r_t)/\max_{t'} \sigma(r_{t'})$ so that scores are comparable with p_{fg}, threshold at $\rho_t \geq 0.5$, and turn each contiguous above-threshold run into a proposal $(\text{start}, \text{end}, \overline{\rho})$. The top-10 saliency proposals from each model are merged with the decoder proposals and re-sorted. The threshold 0.5 in rescaled space is the operating point; 0.3 is too permissive and 0.7 too selective (Section 4.3).

4. EXPERIMENTS

4.1. Setup

Development data are Clotho-Moment ($\sim$51k one-minute synthetic clips, used for pre-training) and CASTELLA (manually annotated, used for fine-tuning, validation, and testing) [1, 16]. We report on the development-testing split (CASTELLA test) using the official metrics: Recall1@$\{0.5, 0.7\}$, mAP averaged over IoU thresholds, mAP@0.5, and mAP@0.75. The primary metric is Recall1@0.7. All M2D-CLAP features, including those for the evaluation set, are taken in the organizer-provided sliding-window form.

4.2. Main results

Table 1 reports the four submitted systems against the official baseline. Each slot improves monotonically on the primary metric. Relative to the baseline, slot 1 (M2D-CLAP features + coarse-aux + rerank) already raises Recall1@0.7 from 13.59 to 28.36, dominated by the feature-extractor upgrade; slot 2 (Mamba + boundary contrast + DN) adds +1.19; slot 3 (cascade) adds +1.48 and improves *every* metric over slot 2; and slot 4 (ensemble + saliency) reaches Recall1@0.7 = 33.70 and mAP = 29.16. Because the baseline uses MS-CLAP features while our systems use M2D-CLAP, the slot-1-vs-baseline gap should be read primarily as a feature-extractor effect; the slot-to-slot deltas isolate the architectural and inference-time contributions.

Table 2: Ablations on the development-testing split, grouped by component in slot order. The selected/used configuration in each group is shown in **bold**.

Variant	R1@.5	R1@.7	mAP	m@.5	m@.75
Coarse-aux γ_c × span-rerank (slot 1)					
γ_c=0, no rerank	42.98	25.76	21.69	38.61	20.40
γ_c=0.25, no rerank	43.65	27.10	21.56	37.96	20.04
γ_c=0.50, no rerank	41.72	23.61	19.78	36.55	18.08
γ_c=0.25**, +rerank**	**43.06**	**28.36**	**23.36**	**38.86**	**22.56**
Mamba depth N / conv width (slot 2)					
N=2, conv 4	42.61	26.13	21.18	38.21	19.90
N=2, conv 2	44.32	28.29	22.25	37.92	21.65
N=1**, conv 4**	**44.10**	**28.29**	**23.87**	**40.08**	**23.14**
N=1, conv 2	43.88	27.69	22.77	39.92	21.66
DN noise scale σ (slot 2)					
σ=0.03	45.29	26.50	23.61	41.68	22.09
σ=0.05	**47.07**	**29.55**	**24.30**	**41.27**	**23.45**
σ=0.10	44.17	27.62	22.21	38.93	21.06
Cascade bound κ (slot 3)					
κ=0.10	46.33	28.73	24.81	41.10	23.86
κ=0.03	45.36	29.40	24.35	40.53	23.47
κ=0.05	**48.40**	**31.03**	**25.82**	**43.33**	**24.94**
Saliency threshold τ_s (slot 3 ckpt, no ensemble)					
τ_s=0.3	49.07	32.07	27.92	44.74	27.29
τ_s=0.5	**49.67**	**32.74**	**28.31**	**44.71**	**27.91**
τ_s=0.7	46.62	29.40	26.51	43.01	26.07
Slot-4 decomposition					
slot 3 (no fusion)	48.40	31.03	25.82	43.33	24.94
+2-seed ensemble	49.59	32.59	27.68	43.91	26.92
+saliency only	49.67	32.74	28.31	44.71	27.91
+both (slot 4)	**50.41**	**33.70**	**29.16**	**44.77**	**28.47**

4.3. Ablations

Each ablation is a fresh pre-train+fine-tune at a single seed on the development-testing split.

All results are collected in Table 2, grouped by component in slot order.

Coarse-auxiliary coefficient. On a backbone without the rerank head, γ_c shows a U-shaped optimum at 0.25 (too small leaves the multi-resolution signal inactive; too large competes with the primary span loss). Adding the span-rerank BCE on top of $\gamma_c = 0.25$ composes roughly additively, and together the two auxiliary signals define slot 1.

Mamba depth. A single bidirectional scan ($N = 1$, conv 4) is preferable to deeper stacks on the boundary-precision metrics (mAP, mAP@0.75); a second layer cumulatively smooths position-specific token content and erodes boundary precision.

DN and cascade scales. Both the DN noise scale σ and the cascade bound κ show U-shaped optima at 0.05. At $\kappa = 0.05$ the cascade improves *both* recall and mAP over slot 2 with no metric trade-off, whereas the looser $\kappa = 0.1$ wins on tight-IoU mAP but regresses recall through over-correction of the top-1 span.

Slot-4 aggregation. The two-seed ensemble (variance reduction) and saliency-as-proposals (information from a structurally different supervision channel) target orthogonal failure modes and compose into the best system. The saliency operating threshold is best at $\tau_s = 0.5$ in rescaled space, with 0.3 too permissive and 0.7 too selective.

5. COMPLIANCE AND CONCLUSION

External resources. All resources are on the DCASE 2026 Task 6 approved list. The feature extractor is M2D-CLAP [7] with a

BERT-based text branch [11]; training uses Clotho-Moment (pre-training) and CASTELLA (fine-tuning) [1, 16], whose audio originates from AudioCaps. We use no LLM APIs, no visual information from the source videos, and no subjective inspection or annotation of the evaluation set. The saliency-as-proposals scheme is a purely inference-time re-use of an already-trained component and introduces no external data or supervision.

Conclusion. Treating AMR as DETR-style set prediction, we showed that stacking complementary training-only supervision channels (coarse-resolution, span re-ranking, boundary contrast, denoising) on a Mamba-augmented QD-DETR backbone, followed by an identity-initialized cascade refinement stage and two inference-time aggregation schemes, yields a monotone progression that reaches Recall1@0.7 = 33.70 on the development-testing split. The cascade and the saliency stream are notable in that they improve boundary precision and recall *simultaneously* once their correction magnitude and operating threshold are tuned.

6. REFERENCES

[1] H. Munakata, T. Nishimura, S. Nakada, and T. Komatsu, "Language-based audio moment retrieval," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, 2025, pp. 1–5.

[2] DCASE Community, "DCASE 2026 Challenge Task 6: Audio Moment Retrieval from Long Audio," 2026, https://dcase.community/challenge2026/task-audio-moment-retrieval-from-long-audio.

[3] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. European Conf. Computer Vision (ECCV)*, 2020, pp. 213–229.

[4] J. Lei, T. L. Berg, and M. Bansal, "Detecting moments and highlights in videos via natural language queries," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, pp. 11 846–11 858.

[5] W. Moon, S. Hyun, S. Park, D. Park, and J.-P. Heo, "Query-dependent video representation for moment retrieval and highlight detection," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 23 023–23 033.

[6] B. Elizalde, S. Deshmukh, and H. Wang, "Natural language supervision for general-purpose audio representations," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, 2024, pp. 336–340.

[7] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, M. Yasuda, S. Tsubaki, and K. Imoto, "M2D-CLAP: Masked modeling duo meets CLAP for learning general-purpose audio-language representation," in *Proc. Interspeech*, 2024, pp. 57–61.

[8] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," in *Proc. Conf. on Language Modeling (COLM)*, 2024.

[9] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[10] F. Li, H. Zhang, S. Liu, J. Guo, L. M. Ni, and L. Zhang, "DN-DETR: Accelerate DETR training by introducing query denoising," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 13 619–13 627.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North American Chapter of the Assoc. for Computational Linguistics (NAACL-HLT)*, 2019, pp. 4171–4186.

[12] D. Meng, X. Chen, Z. Fan, G. Zeng, H. Li, Y. Yuan, L. Sun, and J. Wang, "Conditional DETR for fast training convergence," in *Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2021, pp. 3651–3660.

[13] S. Liu, F. Li, H. Zhang, X. Yang, X. Qi, H. Su, J. Zhu, and L. Zhang, "DAB-DETR: Dynamic anchor boxes are better queries for DETR," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2022.

[14] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 658–666.

[15] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2019.

[16] H. Munakata, T. Imamura, T. Nishimura, and T. Komatsu, "CASTELLA: Long audio dataset with captions and temporal boundaries," 2026.