REDI-TTA: REFINEMENT-GUIDED DISTILLATION FOR MULTI-MODAL TEST-TIME ADAPTATION IN 3D SEMANTIC SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-modal test-time adaptation (MM-TTA) enables 3D semantic segmentation models to adapt to deployment shifts using unlabeled target observations without access to source data. Existing methods exploit predictive reliability to construct adaptation supervision. However, reducing fused predictions to hard pseudo-labels can still preserve an incorrect class decision, even when the full distributions of base models and teachers retain complementary support for alternative classes. To address this shortcoming, we propose Refinement-Guided Distillation for Test-Time Adaptation (ReDi-TTA), which exploits class-wise evidence through probability refinement and role-decoupled student learning while retaining the base adaptation procedure. First, Cross-modal Class Probability Refinement (CCPR) reweights each base prediction using class-aligned probabilities from both exponential moving average teachers, producing branch-specific soft targets that preserve information beyond the highest-scoring class. Second, Role-Decoupled Bimodal Classification (RBC) introduces dedicated student heads operating on aligned image and point-cloud features. These heads are excluded from target construction and EMA teacher updates, preventing student optimization from altering the guidance generation pathway. Third, Proximal Soft-Target Distillation (PSD) addresses the evolving feature representations during base adaptation by fitting pre-update soft targets on detached features recomputed after the base update, while regularizing student-head changes with a proximal constraint. Together, these designs transform refined class-wise evidence into a persistent student learning signal, allowing future predictions to benefit from refinement rather than replacing current-batch outputs. Extensive experiments demonstrate that ReDi-TTA achieves state-of-the-art performance across multiple MM-TTA benchmarks.

1 INTRODUCTION

Image appearance and point-cloud geometry provide complementary information for 3D semantic segmentation, while models based on different modalities may respond differently to changes in illumination, geography, and sensor configuration (Jaritz et al., 2020; Shin et al., 2022). Multi-modal test-time adaptation (MM-TTA) addresses these deployment shifts by adapting source-pretrained models using unlabeled target observations without access to source data(Liang et al., 2020). Therefore, effective adaptation requires exploiting information from both modalities while avoiding the accumulation of erroneous supervision during model updates (Shin et al., 2022; Cao et al., 2023).

Existing MM-TTA methods use different reliability estimates to guide cross-modal supervision. MM-TTA measures slow-fast prediction consistency to select or weight modality predictions for pseudo-label generation (Shin et al., 2022). Latte uses spatial-temporal entropy to weight cross-modal consistency learning and prediction fusion (Cao et al., 2024). PLATO-TTA derives modality fusion weights from prototype-based domain differences and combines pseudo-labelling with adaptive teacher-student updates (Xie et al., 2025). Although these methods use probability distributions differently, their pseudo-labeling branches reduce selected or fused predictions to a single class. An incorrect class preference can therefore become a hard training target(Arazo et al., 2020), even when the underlying distributions retain evidence for competing classes. This motivates examining class-wise support across prediction sources rather than relying on resulting hard label alone.

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

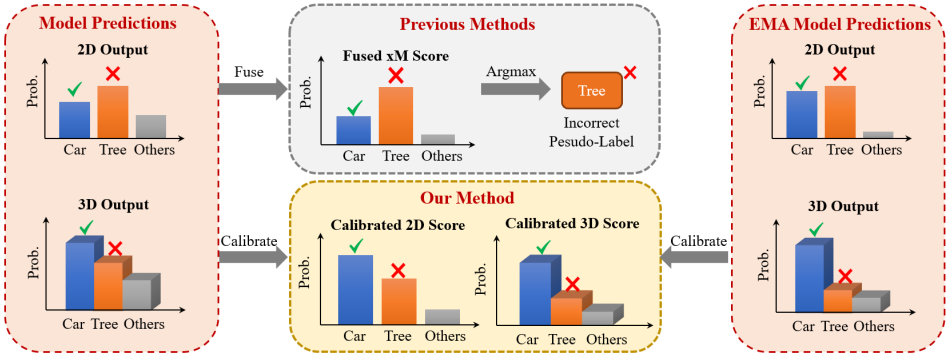


Figure 1: **Motivation for class-wise probability refinement.** Base predictions (left) disagree, and the illustrative fusion–argmax route (upper middle) produces an incorrect *Tree* pseudo-label. EMA predictions (right) provide additional class-wise support; the lower-middle panel depicts the resulting branch-specific refinement. The bars are schematic, not measured probabilities or benchmark results. "Calibrated" in the diagram refers to probability refinement, not an established statistical calibration property. Student learning is shown separately in Fig. 2.

Figure 1 illustrates a potential failure case of reliability-based prediction fusion used in existing MM-TTA methods. The image model favours *Tree*, whereas the point-cloud model favours *Car*. However, the fused prediction still selects *Tree*, producing an incorrect hard pseudo-label. Nevertheless, the base models and their exponential moving average (EMA) teachers(Tarvainen & Valpola, 2017) retain class-wise probability support for *Car*, even when it is not the highest-scoring class. Aggregating the support assigned to each class can therefore change the relative ordering before a hard decision is made. This motivates retaining refined probability distributions rather than immediately reducing them to single labels.

Refining probability distributions alone does not determine how the refined information affects future predictions. Soft-target distillation provides a natural way to transfer such information into model parameters (Hinton et al., 2015; Döbler et al., 2023), but applying it within an adaptive multi-modal framework introduces two challenges. First, the classifiers responsible for generating refinement targets should not simultaneously absorb these targets, as their updates would alter the guidance process itself. Second, once the learning role is separated, the student classifiers must adapt to feature representations that continue to change during base adaptation. Therefore, the student should be optimised on the feature representations produced after the corresponding base update.

Motivated by these observations, we introduce a Refinement-guided Distillation Test-Time Adaptation (ReDi-TTA) framework with three components (Fig. 2). Specifically, Cross-modal Class Probability Refinement (CCPR) first exploits class-wise evidence from base predictions and EMA teachers to construct complete soft targets while preserving probability structures beyond hard labels. Based on these refined targets, Role-Decoupled Bimodal Classification (RBC) separates guidance providers from dedicated learners, allowing the student heads to receive aligned image and point-cloud features without affecting target construction or teacher updates. To further address the evolving feature representations during base adaptation, Proximal Soft-Target Distillation (PSD) optimizes the student heads using pre-update targets, and stop-gradient features are recomputed after the base update, while regularizing deviations from their previous parameters. The original adaptation procedure is retained, and the learned heads are carried forward to subsequent batches. This design establishes a one-way learning pathway from refined class-wise evidence to future predictions without replacing current-batch outputs. Extensive experiments on multiple MM-TTA benchmarks demonstrate the effectiveness of ReDi-TTA under diverse cross-modal distribution shifts.

Our contributions are as follow:

- We propose ReDi-TTA, a refinement-guided distillation framework for online MM-TTA that converts class-wise evidence from base models and EMA teachers into full-distribution soft targets beyond hard pseudo-labels.

- We introduce CCPR and RBC to establish a refinement-to-learning pathway, where CCPR constructs class-aligned soft targets and RBC separates guidance generation from dedicated bimodal student learning.
- We develop PSD to fit refined targets on post-update encoder features with proximal regularization, enabling stable student adaptation while preserving the original base adaptation process.

2 Related Work

Online and continual test-time adaptation. Test-time adaptation aims to improve model robustness under distribution shifts without accessing target annotations(Sun et al., 2020; Zhang et al., 2022). Tent (Wang et al., 2021) adapts normalization parameters by minimizing prediction entropy(Grandvalet & Bengio, 2004). EATA (Niu et al., 2022) introduces selective updates and parameter regularization to reduce unreliable adaptation, while SAR (Niu et al., 2023) improves stability through reliability-aware filtering and sharpness-aware optimization. Beyond single-step adaptation, continual TTA methods consider evolving target streams(Yuan et al., 2023). CoTTA (Wang et al., 2022) leverages averaged teacher predictions and stochastic restoration to alleviate error accumulation, while RMT (Döbler et al., 2023) studies mean-teacher consistency under continual and gradual distribution shifts. Recent approaches further investigate uncertainty modeling (Zhang et al., 2025a), label enhancement (Hu et al., 2025), and adaptive parameter restoration (Lim et al., 2026) for long-term adaptation. These methods mainly address reliable optimization and model stability under distribution shifts. Building on these efforts, we study how refined adaptation signals can be transferred into future predictions through a separate learning pathway while keeping the shared encoders adaptive.

Multi-modal adaptation for 3D segmentation. Multi-modal adaptation exploits complementary information from images and point clouds for 3D semantic segmentation under domain shifts. xMUDA (Jaritz et al., 2020) introduces segmentation and cross-modal mimicry heads for image–point-cloud adaptation, while SUMMIT (Simons et al., 2023) transfers knowledge from independently trained uni-modal models to multi-modal targets in a source-free setting. Recent works extend these ideas to online MM-TTA. MM-TTA (Shin et al., 2022) leverages intra- and inter-modal consistency for pseudo-label refinement, and CoMAC (Cao et al., 2023) combines class-wise modality selection with feature restoration for continual adaptation. Latte (Cao et al., 2024) incorporates spatial-temporal evidence under sequential shifts, while PLATO-TTA (Xie et al., 2025) uses prototype-guided pseudo-labeling and adaptive teacher-student tuning. These approaches mainly improve cross-modal supervision reliability while preserving the adaptation process. However, transferring refined class-wise distributions from multiple prediction sources into a dedicated learning pathway remains less explored.

Probability refinement and classifier-level learning. Probability refinement and soft-target learning provide alternatives to hard pseudo-label supervision(Goyal et al., 2022). Normalized probability products (Hinton, 2002; Kittler et al., 1998) combine multiple predictive distributions, while knowledge distillation (Hinton et al., 2015) transfers information through soft targets beyond maximum-probability classes. In Multi-Modal TTA, SuMi (Guo & Jin, 2025) focuses on aligning predictive distributions across modalities. MDAA (Zhang et al., 2025b) further studies classifier-level adaptation by optimizing analytic classifiers with soft pseudo-labels while keeping encoders and projection layers fixed. Unlike these approaches, ReDi-TTA focuses on transferring refined class-wise distributions into a dedicated student learning pathway while retaining adaptive encoders. It decouples guidance generation from student learning and performs refinement-guided optimization on dynamically updated feature representations.

3 Method

3.1 Overview

ReDi-TTA exploits complementary class-wise information contained in base-model and EMA-teacher predictions and transfers the refined distributions into a dedicated student learning pathway. It separates target construction from student learning while retaining the original adaptation procedure.

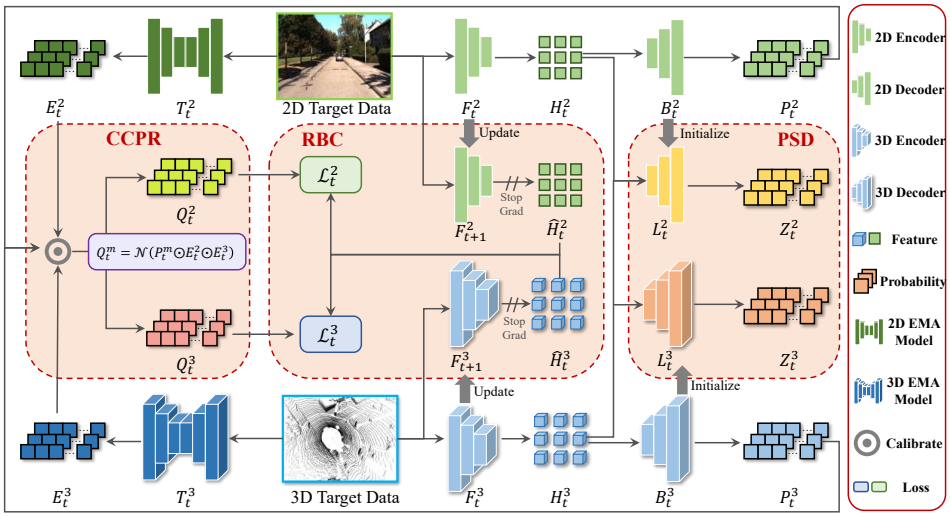


Figure 2: Overview of ReDi-TTA. CCPR refines base predictions using class-aligned evidence from both EMA teachers to construct branch-specific soft targets Q_t^2 and Q_t^3. RBC separates guidance-providing base classifiers from bimodal student heads, whose outputs S_t^2 and S_t^3 are combined by normalized element-wise multiplication. PSD optimizes student heads on post-update detached features with proximal regularization. The predict-then-update protocol preserves the original base adaptation and EMA teacher updates.

As illustrated in Fig. 2, the framework contains three key components. Cross-modal Class Probability Refinement (CCPR) refines each base prediction using class-aligned evidence from exponential moving average (EMA) teachers to construct branch-specific soft targets. Role-Decoupled Bimodal Classification (RBC) introduces separate student classifiers that receive aligned image and point-cloud features. In contrast, the base classifiers retain their role in guidance generation, and the student classifiers do not participate in target construction or EMA teacher updates. Proximal Soft-Target Distillation (PSD) fits the student classifiers to these refined targets using stop-gradient features recomputed after base adaptation and regularizes parameter changes through proximal optimization. Distillation updates only the student heads, while the original base-model adaptation objective and EMA teacher update strategy are preserved.

3.2 Problem Setting

We consider the online multi-modal test-time adaptation setting, where source-domain data and target-domain annotations are unavailable during adaptation. At step t, the model receives an RGB image batch x_t^2 and its corresponding point-cloud batch x_t^3. Point-pixel correspondences are used to align the two modalities on the same visible 3D points, ensuring that corresponding feature rows refer to the same observation. The aligned features are defined as:

$$H_t^m = F_t^m(x_t^m) \in \mathbb{R}^{N_t \times D_m}, \qquad m \in \{2, 3\}, \tag{1}$$

where F_t^m is the modality-specific encoder before adaptation on batch t, N_t is the number of aligned points, and D_m is the feature dimension. The superscripts 2 and 3 indicate the image and point-cloud modalities; for the image branch, Equation (1) includes gathering pixel features at the corresponding point projections. All predictions share the same C semantic classes and aligned point ordering. We follow a predict-then-update protocol: batch t is evaluated before adaptation on that batch, and updated parameters are used only for subsequent predictions.

3.3 Cross-modal Class Probability Refinement (CCPR)

The base adaptation process provides predictions from both modalities and their EMA teachers. Reducing these distributions to hard pseudo-labels retains the selected class but discards the relative

support for alternatives. CCPR instead combines their class-aligned probabilities to construct a full soft target for each branch. For modality m, the base classifier B_t^m and EMA teacher T_t^m produce

$$P_t^m = \sigma(B_t^m(H_t^m)), \qquad E_t^m = T_t^m(x_t^m), \tag{2}$$

where σ applies softmax across classes and T_t^m returns aligned teacher probabilities. Both matrices belong to $\mathbb{R}^{N_t \times C}$ and are obtained before the base update. The base prediction provides a branch-specific reference, whereas the teachers supply shared class-wise guidance. Rather than summarizing this guidance with one confidence score per point or modality, we retain each teacher's probability for every class and combine them in log space:

$$U_t(i,c) = \log E_t^2(i,c) + \log E_t^3(i,c). \tag{3}$$

For aligned point i and class c, $U_t(i,c)$ represents the teachers' combined log-support. The matrix $U_t \in \mathbb{R}^{N_t \times C}$ is shared by both refinement branches, but retains class-specific preferences that a scalar confidence cannot express. Support for non-maximal classes remains available to refinement. CCPR incorporates this support while anchoring each refined distribution to its corresponding base prediction. Over the probability simplex Δ_C, the refined target is defined by

$$Q_t^m(i,\cdot) = \arg\min_{q \in \Delta_C} \left[D_{\mathrm{KL}}(q \,\|\, P_t^m(i,\cdot)) - \sum_{c=1}^{C} q(c) U_t(i,c) \right]. \tag{4}$$

The KL term penalizes deviations from the base distribution (Bissiri et al., 2016), while the support term favors classes with stronger teacher support. The refined distribution need not preserve the base model's class ranking, as sufficiently strong teacher support can change the most probable class. Assuming strictly positive input probabilities, the first-order optimality condition under the simplex constraint gives $q(c) \propto P_t^m(i,c)\exp(U_t(i,c))$. Substituting Equation (3) and normalizing across classes yields the following:

$$Q_t^m(i,c) = \frac{P_t^m(i,c) E_t^2(i,c) E_t^3(i,c)}{\sum_{k=1}^{C} P_t^m(i,k) E_t^2(i,k) E_t^3(i,k)}. \tag{5}$$

The numerator reweights each base probability by both teachers' support for that same class, and the denominator restores a valid distribution. Equivalently, $Q_t^m = \sigma(\log P_t^m + U_t)$, with element-wise logarithms and row-wise softmax. The targets remain branch-specific through P_t^m and preserve supervision beyond a single selected class. CCPR requires no iterative optimization or additional trainable parameters; its outputs Q_t^2 and Q_t^3 are held fixed during student fitting. Shared errors can still affect the targets, so refinement alone does not guarantee a correct class assignment.

3.4 Role-Decoupled Bimodal Classification (RBC)

Directly fitting the original base classifiers to CCPR targets raises two coupled issues. Each target incorporates both modalities, whereas a base classifier observes only its own modality's features. Those classifiers also supply predictions for future target construction, so an additional distillation objective would feed student learning back into guidance generation. RBC separates these roles while sharing the modality encoders: base classifiers remain guidance providers, and dedicated bimodal student heads learn the refined distributions for subsequent inference.

Each student branch combines its own features with the aligned features from the complementary modality, denoted by $\bar{m}$. Its within-modality term retains the branch-specific classification pathway, while its cross-modal term provides access to information used in constructing the target. With weights $W_t^m \in \mathbb{R}^{C \times D_m}$, $A_t^m \in \mathbb{R}^{C \times D_{\bar{m}}}$, and bias $b_t^m \in \mathbb{R}^C$, the student prediction is

$$S_t^m = \sigma\big(H_t^m (W_t^m)^\top + H_t^{\bar{m}} (A_t^m)^\top + \mathbf{1}(b_t^m)^\top\big), \tag{6}$$

where $\mathbf{1} \in \mathbb{R}^{N_t \times 1}$ is an all-ones column. Within-modality weights and biases are copied from the corresponding base classifiers, and cross-modal weights are initialized to zero. Each branch therefore initially reproduces its base classifier's output, while subsequent fitting can use both feature sets. The students remain distinct through their initialization and branch-specific targets, and neither participates in CCPR target construction or EMA teacher updates.

For evaluation, the two student distributions are combined by a normalized element-wise product. The multiplication pairs probabilities for the same semantic class at each aligned point. The cross-modal output is

$$S_t^{\mathrm{xM}}(i,c) = \frac{S_t^2(i,c)S_t^3(i,c)}{\sum_{k=1}^{C} S_t^2(i,k)S_t^3(i,k)}. \tag{7}$$

Normalization is performed independently for each point, with final label $\widehat{y}_t(i) = \arg\max_c S_t^{\mathrm{xM}}(i,c)$. Equivalently, $S_t^{\mathrm{xM}} = \sigma(\log S_t^2 + \log S_t^3)$. This rule fuses the learned student outputs S_t^2 and S_t^3, not the refinement targets Q_t^2 and Q_t^3; teacher probabilities are not multiplied directly into the evaluated output. A low probability in one branch can suppress a class favored by the other, so multiplicative fusion does not imply that the fused prediction must outperform both branches.

3.5 Proximal Soft-Target Distillation (PSD)

The shared encoders continue to evolve under base adaptation, so fitting pre-update features would optimize the student heads for an outdated representation. PSD instead fits fixed CCPR targets on recomputed post-update features, with a proximal penalty constraining changes from the preceding student parameters. After base adaptation changes F_t^m to F_{t+1}^m, the same batch is forwarded through the updated encoders. The recomputed features are detached so that distillation gradients cannot update the encoders or the base classifiers. Using sg for stop-gradient, we form the augmented student input as

$$\widehat{H}_t^m = \mathrm{sg}\big(F_{t+1}^m(x_t^m)\big), \qquad X_t^m = \big[\widehat{H}_t^m, \widehat{H}_t^{\bar{m}}, \mathbf{1}\big]. \tag{8}$$

Here $X_t^m \in \mathbb{R}^{N_t \times d}$ with $d = D_2 + D_3 + 1$, including the bias coordinate. The targets remain those computed before base adaptation, whereas the features now correspond to the encoder state carried into the next batch. To express both feature contributions and the bias within one linear classifier, we stack their parameters in the same order:

$$\Theta_t^m = \big[(W_t^m)^\top; (A_t^m)^\top; (b_t^m)^\top\big] \in \mathbb{R}^{d \times C}, \tag{9}$$

where semicolons denote vertical concatenation. A candidate parameter matrix Θ produces $\sigma(X_t^m\Theta)$, which is the classifier in Equation (6) evaluated on the detached, post-update features. Detachment fixes the inputs during student fitting; it does not freeze the encoders in the original adaptation process. PSD matches the full target distribution rather than replacing it with a hard pseudo-label. With Q_t^m and X_t^m fixed, every class contributes to fitting according to its target probability. The branch-specific soft-target cross-entropy is

$$\mathcal{L}_t^m(\Theta) = -\frac{1}{N_t}\sum_{i=1}^{N_t}\sum_{c=1}^{C} Q_t^m(i,c)\log[\sigma(X_t^m\Theta)]_{ic}. \tag{10}$$

Since target entropy is constant with respect to Θ, minimizing this loss is equivalent to minimizing the average KL divergence from the CCPR targets to the student predictions. The loss thus transfers relative support for competing classes rather than supervising only the target's maximum-probability class. To balance fitting the current batch against changing the previously learned classifier, PSD anchors the update at Θ_t^m:

$$\Theta_{t+1}^m = \arg\min_{\Theta}\left[\mathcal{L}_t^m(\Theta) + \frac{\lambda_t^m}{2}\|\Theta - \Theta_t^m\|_F^2\right]. \tag{11}$$

Unlike a penalty centered at zero, this term discourages departure from the preceding student state without requiring the classifier to remain unchanged(Rockafellar, 1976; Parikh & Boyd, 2014). The coefficient λ_t^m controls this constraint, with larger values penalizing parameter changes more strongly. To relate its scale to the inputs used for fitting, we set it using the mean squared magnitude of the augmented features:

$$\lambda_t^m = \rho\max\left(\frac{\|X_t^m\|_F^2}{N_t(D_2+D_3+1)}, \varepsilon\right), \tag{12}$$

where $\rho > 0$ controls regularization strength, $\varepsilon > 0$ provides a positive floor for the feature-scale statistic, and $\|\cdot\|_F$ denotes the Frobenius norm. During PSD, only the student parameters are optimized; features, targets, base classifiers, and teachers are held fixed. The encoders and base classifiers continue to follow the original adaptation objective, and the dedicated student parameters do not enter the EMA update.

Table 1: Full-stream mIoU (%). Red/blue mark the best/second-best results in the main Cross comparison. All baseline results are local reproductions and ours is PLATO-based. Student branches use product xM fusion; baselines retain native fusion. Avg averages the three xM scores. Gains are computed from unrounded results.

Group	Method	U→S			A→S			S→S			Avg
		2D	3D	xM	2D	3D	xM	2D	3D	xM	
Lowerbound	Source only	31.40	41.10	43.90	47.40	17.90	44.30	23.40	36.40	38.20	42.13
Uni	TENT	36.80	36.10	41.10	43.30	44.40	49.10	22.40	35.50	37.50	42.57
	ETA	36.70	34.70	43.70	43.20	42.50	49.70	22.40	30.60	32.90	42.10
	SAR	36.60	37.20	43.90	43.00	42.70	50.30	20.70	31.30	32.90	42.37
	SAR-rs	36.60	35.50	43.90	43.10	43.80	50.10	24.20	25.10	28.80	40.93
Cross	xMUDA	19.40	22.90	24.20	13.10	33.00	32.20	12.20	14.80	14.10	23.50
	xMUDA+PL	36.20	38.20	43.00	42.80	48.30	50.90	30.90	34.10	36.30	43.40
	MM-TTA	37.19	41.51	45.47	44.37	51.40	52.64	27.29	33.57	34.43	44.18
	PsLabel	37.70	35.80	41.80	46.70	44.30	50.00	29.20	33.30	35.30	42.37
	Latte	37.46	40.99	46.19	46.71	53.68	55.20	31.53	36.72	38.66	46.68
	PLATO-TTA	43.62	45.38	49.87	51.61	55.10	55.43	39.52	44.98	46.92	50.74
	REDI-TTA (ours)	52.06	51.98	53.65	54.33	56.80	56.49	47.22	46.28	48.03	52.72
	Latte + Ours	46.88	47.69	49.25	53.59	57.34	56.80	37.50	39.77	39.89	48.65
	Δ Gain	↑9.42	↑6.70	↑3.06	↑6.88	↑3.66	↑1.60	↑5.97	↑3.04	↑1.22	↑1.96
	MM-TTA + Ours	45.58	45.20	46.81	51.26	53.97	53.98	33.05	35.70	35.29	45.36
	Δ Gain	↑8.39	↑3.68	↑1.34	↑6.89	↑2.57	↑1.34	↑5.76	↑2.13	↑0.86	↑1.18
Upperbound	Oracle TTA	38.70	45.50	50.30	49.10	61.30	62.50	36.00	54.50	54.60	55.80

3.6 ONLINE EXECUTION

For each target batch t, ReDi-TTA uses the pre-update model state to compute S_t^2, S_t^3, and the product-fused output (Equation (7)) for evaluation and caches Q_t^2 and Q_t^3 from the base classifiers and EMA teachers as fixed targets. After the original adaptation step updates the base model, PSD recomputes and detaches the aligned features using the updated encoders, then fits the student heads via Equations (10)-(12). EMA teachers are then updated as in the original framework, excluding the dedicated student heads. The updated encoders and student parameters are carried forward to the next batch, so batch-t guidance affects subsequent predictions through student learning without altering the recorded batch t outputs.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Benchmarks and protocol. We follow Latte and PLATO-TTA (Cao et al., 2024; Xie et al., 2025) on USA→Singapore (U→S, 16 classes) from nuScenes (Caesar et al., 2020), A2D2→SemanticKITTI (A→S, 10 classes) (Geyer et al., 2020; Behley et al., 2019), and SYNTHIA→SemanticKITTI (S→S, nine classes) (Ros et al., 2016; Behley et al., 2019). These cover geographic, cross-sensor, and synthetic-to-real shifts. Each target stream is processed once without source data or target annotations, with predictions recorded before current-batch adaptation. We compute full-stream mIoU from accumulated confusion matrices; Avg averages the three xM scores. Both student heads receive bimodal features, and student xM uses product fusion.

Baselines and implementation. PLATO-TTA is the primary base; Latte and MM-TTA assess transfer of the same student design. These baselines are reproduced locally, while other baseline results follow PLATO-TTA (Xie et al., 2025). Paired runs share checkpoints, streams, batch sizes, and seed 1. Each base retains its native fusion. Student heads use L-BFGS (Liu & Nocedal, 1989) with $\rho = 10^{-3}$, retaining the base adaptation procedure. Appendix A specifies the stream sizes, fusion rules, and fitting budget.

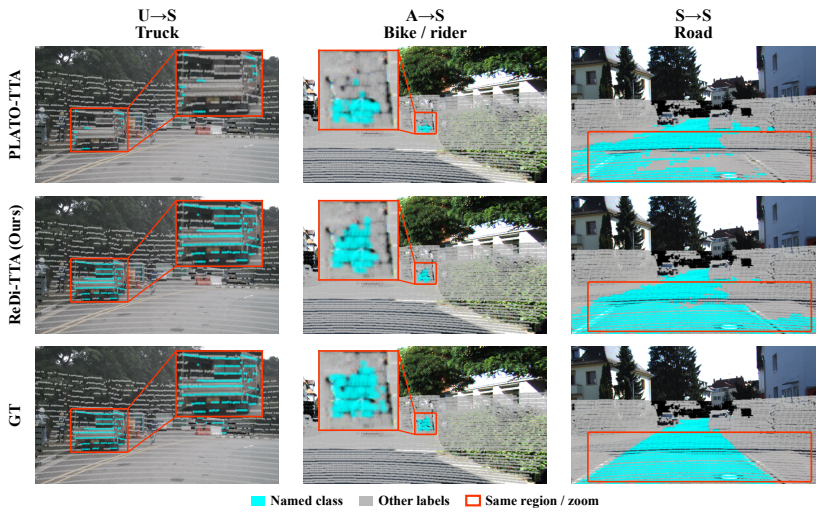


Figure 3: Selected pre-update xM predictions. Cyan highlights the named class; gray denotes other classes. Matched regions and insets use identical point locations. PLATO-TTA uses arithmetic fusion and ReDi-TTA uses product fusion. GT denotes ground truth.

4.2 Segmentation Results

Full-stream performance. Table 1 reports xM scores of 53.65, 56.49, and 48.03 for ReDi-TTA on U→S, A→S, and S→S, respectively. The average xM achieves 52.72, improving over PLATO-TTA by 1.98 points. The largest improvement is observed on U→S, with gains of 3.78, 1.06, and 1.12 points across the three transfers. Both student heads outperform the corresponding base branches on all tasks, although the improvements are not fully reflected in the fused predictions. For example, on S→S, the 2D student improves by 7.71 points, while the fused xM increases by 1.12 points. ReDi-TTA achieves the highest average xM among the compared unsupervised methods. Under the predict-then-update protocol, adaptation updates cannot affect predictions already recorded for the current batch.

Transfer across adaptation frameworks. The same student design improves mean xM by 1.96 points with Latte and 1.18 points with MM-TTA. All nine framework–task combinations outperform their matched native baselines. The PLATO-based variant achieves the best xM on U→S and S→S, while the Latte-based variant obtains the highest A→S xM of 56.80. These consistent gains across different frameworks demonstrate that ReDi-TTA is not limited to a single adaptation framework, although the improvement varies with the base method and transfer task.

Class-wise and qualitative results. The ReDi-TTA improves xM on 12 of 16 classes for U→S, five of 10 classes for A→S, and four of nine classes for S→S (Appendix B.2). On U→S, barrier, pedestrian, and traffic cone show clear improvements, while motorcycle, sidewalk, and terrain decrease. On S→S, improvements in bike and pole coexist with declines in car and building, limiting the overall gain. Figure 3 presents representative corrections on truck, bike/rider, and road. Their class IoUs improve from 11.43 to 85.37, from 30.00 to 68.18, and from 53.67 to 70.98, respectively, evaluated over all valid points in each frame. Appendix B.3 provides the selection protocol, complete-frame metrics, and uncropped views.

4.3 Ablation Studies and Analysis

Refinement and student roles. On ReDi-TTA, removing CCPR reduces the 2D, 3D, and xM scores by 8.18, 3.35, and 3.23 points, respectively (Table 2). Jointly removing cross-modal inputs and guidance isolation reduces xM to 50.42. Separate controls further disentangle these effects: using only own-modality inputs decreases xM by 0.41 points, while allowing student predictions to participate in target construction decreases xM by 1.55 points (Appendix C). Both controls keep the base adaptation

states and EMA teacher updates unchanged. With the fitting budget fixed, these results show that both class-wise refinement targets and the role separation in bimodal student learning contribute to the final performance. The larger degradation caused by student-fed targets suggests that access to complementary modalities alone is insufficient without decoupling target generation from student learning.

Table 2: ReDi-TTA component controls. Row 1: native base; row 2: direct refinement; rows 3–6: learned students with product fusion. Disabling RBC removes cross-modal input and guidance isolation. Disabling PSD removes feature refresh and proximal regularization, retaining soft-target fitting. Rows are not additive.

#	CCPR	RBC	PSD	Output	2D	3D	xM
1	–	–	–	Base	43.62	45.38	49.87
2	✓	–	–	Refinement	52.47	52.49	53.19
3	–	✓	✓	Student	42.71	44.41	50.48
4	✓	–	✓	Student	43.88	48.63	50.42
5	✓	✓	–	Student	52.11	52.27	53.07
6	✓	✓	✓	Student	52.06	51.98	53.65

PSD choices. Hard targets and removal of the proximal penalty reduce xM by 0.14 and 0.06 points. Fitting on pre-update features reaches 53.23, exceeding the complete configuration at 52.91 by 0.32 points. Removing both feature refresh and the proximal term also improves xM to 53.07. Soft targets and proximal regularization therefore have small positive margins in this setting, but feature refresh has no demonstrated advantage. These single-seed differences do not establish statistical significance or benefits on the other transfers.

Learning and fusion. On ReDi-TTA, product fusion raises base xM from 49.87 to 51.21; the student reaches 52.90 under the same rule, retaining a 1.69-point gain. This fusion-matched comparison shows that the improvement is not explained by the fusion change alone. Applying both rules to identical student predictions further isolates the output effect: product fusion improves all three PLATO scores but reduces five of the six Latte and MM-TTA scores (Appendix C.2). On A→S, the ReDi-TTA 3D student reaches 56.80, above its fused score of 56.49. Improved branch accuracy therefore does not ensure that the fixed product rule fully exploits the available predictions.

Direct refinement and learned predictions. CCPR-only evaluates direct probability refinement, not an intermediate learned model, and serves as a guidance reference rather than an upper bound on student performance. The US200 diagnostic supports the role of class correspondence beyond scalar confidence in direct refinement but does not isolate its effect on student learning. Detailed comparisons and diagnostics are provided in Appendices B.1 and D, respectively.

5 Discussion and Limitations

CCPR isolates the effect of class-aligned probability refinement without modifying the underlying adaptation process. Its deterministic formulation introduces no additional trainable components or point-level memory. However, the product-based refinement may amplify correlated errors among prediction sources, and the quality of refined targets depends on the reliability of the underlying models. The current evaluation focuses on cross-modal adaptation performance, while further investigation of calibration, missing-modality robustness, and efficiency remains open.

6 Conclusion

We presented ReDi-TTA, a refinement-guided distillation framework for online multi-modal test-time adaptation. ReDi-TTA transfers class-wise cross-modal evidence into a dedicated student learning pathway while preserving the original adaptation process. CCPR constructs full-distribution soft targets, RBC decouples guidance generation from student learning, and PSD fits these targets on evolving feature representations with proximal regularization. Experiments on multiple MM-TTA benchmarks demonstrate the effectiveness of ReDi-TTA under diverse cross-modal distribution shifts.

REFERENCES

Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2020. doi: 10.1109/IJCNN48605.2020.9207304. URL https://doi.org/10.1109/IJCNN48605.2020.9207304.

Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Juergen Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9297–9307, 2019. URL https://openaccess.thecvf.com/content_ICCV_2019/html/Behley_SemanticKITTI_A_Dataset_for_Semantic_Scene_Understanding_of_LiDAR_Sequences_ICCV_2019_paper.html.

P. G. Bissiri, C. C. Holmes, and S. G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):1103–1130, 2016. doi: 10.1111/rssb.12158. URL https://doi.org/10.1111/rssb.12158.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11621–11631, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Caesar_nuScenes_A_Multimodal_Dataset_for_Autonomous_Driving_CVPR_2020_paper.html.

Haozhi Cao, Yuecong Xu, Jianfei Yang, Pengyu Yin, Shenghai Yuan, and Lihua Xie. Multi-modal continual test-time adaptation for 3D semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18809–18819, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Cao_Multi-Modal_Continual_Test-Time_Adaptation_for_3D_Semantic_Segmentation_ICCV_2023_paper.html.

Haozhi Cao, Yuecong Xu, Jianfei Yang, Pengyu Yin, Xingyu Ji, Shenghai Yuan, and Lihua Xie. Reliable spatial-temporal voxels for multi-modal test-time adaptation. In *Computer Vision – ECCV 2024*, pp. 232–249. Springer, 2024. doi: 10.1007/978-3-031-73390-1_14. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/4076_ECCV_2024_paper.php.

Mario Döbler, Robert A. Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7704–7714, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Dobler_Robust_Mean_Teacher_for_Continual_and_Gradual_Test-Time_Adaptation_CVPR_2023_paper.html.

Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S. Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, Tiffany Fernandez, Martin Jänicke, Sudesh Mirashi, Chiragkumar Savani, Martin Sturm, Oleksandr Vorobiov, Martin Oelker, Sebastian Garreis, and Peter Schuberth. A2D2: Audi Autonomous Driving Dataset. arXiv preprint arXiv:2004.06320, 2020. URL https://arxiv.org/abs/2004.06320.

Sachin Goyal, Mingjie Sun, Aditi Raghunathan, and J. Zico Kolter. Test Time Adaptation via Conjugate Pseudo-Labels. In *Advances in Neural Information Processing Systems*, volume 35, pp. 6204–6218. Curran Associates, Inc., 2022. doi: 10.52202/068431-0449. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/28e9eff897f98372409b40ae1ed3ea4c-Abstract-Conference.html.

Yves Grandvalet and Yoshua Bengio. Semi-supervised Learning by Entropy Minimization. In *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004. URL https://proceedings.neurips.cc/paper/2004/hash/96f2b50b5d3613adf9c27049b2a888c7-Abstract.html.

Zirun Guo and Tao Jin. Smoothing the shift: Towards stable test-time adaptation under complex multimodal noises. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/065e259a1d2d955e63b99aac6a3a3081-Abstract-Conference.html.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. doi: 10.1162/089976602760128018.

Yihao Hu, Congyu Qiao, Xin Geng, and Ning Xu. Selective label enhancement learning for test-time adaptation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/ac662d74829e4407ce1d126477f4a03a-Abstract-Conference.html.

Maximilian Jaritz, Tuan-Hung Vu, Raoul de Charette, Émilie Wirbel, and Patrick Pérez. xMUDA: Cross-modal unsupervised domain adaptation for 3D semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12605–12614, 2020. URL https://arxiv.org/abs/1911.12676.

Josef Kittler, Mohamad Hatef, Robert P. W. Duin, and Jiří Matas. On Combining Classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998. doi: 10.1109/34.667881. URL https://doi.org/10.1109/34.667881.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do We Really Need to Access the Source Data? Source Hypothesis Transfer for Unsupervised Domain Adaptation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6028–6039. PMLR, 2020. URL https://proceedings.mlr.press/v119/liang20a.html.

Taejun Lim, Joongwon Hwang, and Kibok Lee. When and where to reset matters for long-term test-time adaptation. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/79bdd6fe3f012befcc459ad13de65d13-Abstract-Conference.html.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1–3):503–528, 1989. doi: 10.1007/BF01589116. URL https://doi.org/10.1007/BF01589116.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16888–16905. PMLR, 2022. URL https://proceedings.mlr.press/v162/niu22a.html.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2302.12400.

Neal Parikh and Stephen Boyd. Proximal Algorithms. *Foundations and Trends in Optimization*, 1(3):127–239, 2014. doi: 10.1561/2400000003. URL https://doi.org/10.1561/2400000003.

R. Tyrrell Rockafellar. Monotone Operators and the Proximal Point Algorithm. *SIAM Journal on Control and Optimization*, 14(5):877–898, 1976. doi: 10.1137/0314056. URL https://doi.org/10.1137/0314056.

German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3234–3243, 2016. URL https://openaccess.thecvf.com/content_cvpr_2016/html/Ros_The_SYNTHIA_Dataset_CVPR_2016_paper.html.

Inkyu Shin, Yi-Hsuan Tsai, Bingbing Zhuang, Samuel Schulter, Buyu Liu, Sparsh Garg, In So Kweon, and Kuk-Jin Yoon. MM-TTA: Multi-modal test-time adaptation for 3D semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16928–16937, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Shin_MM-TTA_Multi-Modal_Test-Time_Adaptation_for_3D_Semantic_Segmentation_CVPR_2022_paper.html.

Cody Simons, Dripta S. Raychaudhuri, Sk Miraj Ahmed, Suya You, Konstantinos Karydis, and Amit K. Roy-Chowdhury. SUMMIT: Source-free adaptation of uni-modal models to multi-modal targets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1239–1249, 2023. URL https://arxiv.org/abs/2308.11880.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-Time Training with Self-Supervision for Generalization under Distribution Shifts. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248. PMLR, 2020. URL https://proceedings.mlr.press/v119/sun20b.html.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/hash/68053af2923e00204c3ca7c6a3150cf7-Abstract.html.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7201–7211, 2022. URL https://arxiv.org/abs/2203.13591.

Jianxiang Xie, Yao Wu, Yachao Zhang, Xiaopei Zhang, Yuan Xie, and Yanyun Qu. PLATO-TTA: Prototype-guided pseudo-labeling and adaptive tuning for multi-modal test-time adaptation of 3D segmentation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 2226–2234. ACM, 2025. doi: 10.1145/3746027.3755793. URL https://dl.acm.org/doi/10.1145/3746027.3755793.

Longhui Yuan, Binhui Xie, and Shuang Li. Robust Test-Time Adaptation in Dynamic Scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15922–15932, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Yuan_Robust_Test-Time_Adaptation_in_Dynamic_Scenarios_CVPR_2023_paper.html.

Marvin Zhang, Sergey Levine, and Chelsea Finn. MEMO: Test Time Robustness via Adaptation and Augmentation. In *Advances in Neural Information Processing Systems*, volume 35, pp. 38629–38642. Curran Associates, Inc., 2022. doi: 10.52202/068431-2799. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/fc28053a08f59fccb48b11f2e31e81c7-Abstract-Conference.html.

Qingyang Zhang, Yatao Bian, Xinke Kong, Peilin Zhao, and Changqing Zhang. COME: Test-time adaption by conservatively minimizing entropy. In *International Conference on Learning Representations*, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/2ea18fdc667e0ef2ad82b2b4d65147ad-Abstract-Conference.html.

Yufei Zhang, Yicheng Xu, Hongxin Wei, Zhiping Lin, Xiaofeng Zou, Cen Chen, and Huiping Zhuang. Analytic continual test-time adaptation for multi-modality corruption. In *Proceedings of the 33rd ACM International Conference on Multimedia*. ACM, 2025b. doi: 10.1145/3746027.3755545. URL https://arxiv.org/abs/2410.22373v2.

A EXPERIMENTAL DETAILS

Appendix A specifies the evaluation and implementation; Appendix B provides additional segmentation results. Student controls and probability diagnostics appear in Appendices C and D. Appendix E collects reference checks and historical variants.

A.1 EVALUATION PROTOCOL AND OPTIMIZATION

We abbreviate the three transfers as US, AS, and SS. Paired experiments match pretrained checkpoint hashes, seed, batch size, frame count, and valid-point label support. PLATO-TTA uses batch sizes 4/4/3, giving 3,841/5,801/7,734 batches on US/AS/SS. Latte and MM-TTA use batch size 6, giving 2,561/3,867/3,867 batches. US contains 15,364 frames; each SemanticKITTI transfer contains 23,201. Equal batch counts across frameworks can therefore describe different frame prefixes.

PLATO-TTA, Latte, and MM-TTA are locally reproduced; other baseline results follow PLATO-TTA (Xie et al., 2025). Native xM fusion is arithmetic averaging for PLATO-TTA, spatial–temporal fusion for Latte, and EMA averaging for the MM-TTA implementation provided with Latte. Source only and Oracle TTA are non-adaptive and target-supervised references, respectively, not strict performance bounds. Predictions precede the current-batch update, and mIoU is computed from the pooled full-stream confusion matrix. Main-table differences and averages use unrounded records, which can cause last-decimal discrepancies when displayed entries are subtracted.

All final runs use seed 1. PSD uses $\rho = 10^{-3}$, $\varepsilon = 10^{-12}$, and target temperature 1. Each head is fitted in double precision with L-BFGS and strong-Wolfe line search, using at most 16 iterations and a requested limit of 48 function evaluations. Learning rate, history size, gradient tolerance, and objective-change tolerance are 1, 10, 10^{-7}, and 10^{-9}. The proximal anchor is the preceding head state; optimizer state is reset for each fit. Ablations retain this budget. Base objectives and teacher updates remain those of the selected framework.

A.2 REFINEMENT READOUTS AND IMPLEMENTATION

The direct CCPR reference evaluates refined current predictions without fitting the dedicated heads. Its cross-modal readout is

$$Q_t^{\mathrm{xM}} = \mathcal{N}(P_t^2 \odot P_t^3 \odot E_t^2 \odot E_t^3), \tag{13}$$

where $\mathcal{N}$ normalizes across classes. Each teacher appears once. Student xM in Equation (7) instead combines the learned head probabilities and contains no direct teacher factor.

For a positive vector v, define $\mathcal{N}(v) = v/\sum_c v_c$. With point and time indices suppressed, let $\widehat{p}^m = \max(p^m, \epsilon)$ and $R = \max(e^{2\mathrm{D}}, \epsilon)\max(e^{3\mathrm{D}}, \epsilon)$, using element-wise multiplication and a probability floor $\epsilon > 0$. Then

$$q^{2\mathrm{D}} = \mathcal{N}(\widehat{p}^{2\mathrm{D}} R), \quad q^{3\mathrm{D}} = \mathcal{N}(\widehat{p}^{3\mathrm{D}} R), \quad q^{\mathrm{xM}} = \mathcal{N}(\widehat{p}^{2\mathrm{D}}\widehat{p}^{3\mathrm{D}} R). \tag{14}$$

The branch normalizers are constant across classes, so

$$\mathcal{N}(q^{2\mathrm{D}} q^{3\mathrm{D}}) = \mathcal{N}(\widehat{p}^{2\mathrm{D}}\widehat{p}^{3\mathrm{D}} R^2) \neq q^{\mathrm{xM}} \tag{15}$$

in general. Multiplying the refined branch targets therefore counts shared teacher support twice; the direct reference in Equation (13) uses a single factor R.

Log-space evaluation is equivalent in exact arithmetic. Matching a floating-point result also requires matching the probability floor, precision, and operation order. The probability floor ϵ is distinct from the feature-scale floor ε in PSD. CCPR discards point-wise intermediate scores after each batch. Teacher parameters and native temporal processing remain part of the base framework, not a CCPR point-memory bank. Student heads persist through their parameters but do not enter teacher updates. Evaluation labels must not affect inference or adaptation state.

B ADDITIONAL SEGMENTATION RESULTS

B.1 DIRECT REFINEMENT

Table 3 reports the direct readouts defined in Appendix A.2. They quantify immediate probability reweighting and are not an upper bound on student performance. The PLATO, Latte, and MM-TTA

students have higher xM than their respective direct readouts on all three transfers. Main-text gains retain the fixed native baselines in Table 1.

Table 3: Full-stream direct CCPR refinement (mIoU, %). These readouts use base and teacher probabilities and do not evaluate the learned heads.

Base	Task	2D	3D	xM
PLATO-TTA + CCPR	US	52.467	52.492	53.188
PLATO-TTA + CCPR	AS	54.288	56.648	55.766
PLATO-TTA + CCPR	SS	46.826	46.079	47.411
Latte + CCPR	US	47.188	47.733	48.614
Latte + CCPR	AS	53.427	57.110	55.928
Latte + CCPR	SS	37.305	39.559	38.983
MM-TTA + CCPR	US	45.877	45.445	46.298
MM-TTA + CCPR	AS	51.103	53.850	53.112
MM-TTA + CCPR	SS	32.890	35.557	34.422

B.2 Per-class performance

Tables 4–6 compare the complete PLATO-based model with its fixed baseline using the evaluation class order. Both student branches receive bimodal features. The xM score increases for 12 of 16 US classes, five of 10 AS classes, and four of nine SS classes. On SS, gains in bike and pole coexist with losses in car, road, and building. All classes, including decreases and near-zero scores, are retained. Archived per-class records also contain native-base, CCPR, and student readouts for the other frameworks.

Table 4: Class IoU on U→S (%). Base xM uses arithmetic fusion; student xM uses product fusion. Both student branches receive bimodal features.

Class	PLATO-TTA			ReDi-TTA		
	2D	3D	xM	2D	3D	xM
barrier	47.12	38.92	47.41	57.82	53.28	56.27
bicycle	1.98	0.10	0.59	1.65	0.23	0.88
bus	61.68	65.51	74.77	77.98	77.76	79.26
car	71.12	80.58	82.82	83.81	86.75	86.19
construction vehicle	33.95	28.91	40.43	44.87	43.43	45.50
motorcycle	25.41	30.84	37.58	34.46	37.01	36.99
pedestrian	41.18	51.01	56.08	59.61	63.68	63.63
traffic cone	25.16	19.75	29.81	38.94	33.55	38.07
trailer	0.05	1.64	0.94	7.99	4.63	7.32
truck	54.06	56.11	63.81	66.59	67.48	68.76
driveable surface	93.95	95.86	96.00	95.72	96.23	96.10
other flat	12.83	13.52	12.86	12.39	13.11	12.73
sidewalk	40.75	48.41	49.13	46.97	49.31	48.65
terrain	52.64	55.63	57.64	55.10	56.17	56.04
manmade	58.28	60.24	65.80	66.64	66.62	67.26
vegetation	77.70	79.13	82.30	82.38	82.40	82.84
mIoU	43.62	45.38	49.87	52.06	51.98	52.90

B.3 Qualitative evaluation

Selection and replay. Historical confusion-matrix comparisons over 100-batch windows motivate inspection of batches 100–200 on US and AS. The earlier US trace uses arithmetic student fusion only for interval selection; the replay evaluates the final product-fused student. Both methods are replayed through 200 batches from the original checkpoints with fixed recipes, seed 1, and batch size 4. Capturing all four frames at zero-based iterations 100–199 yields 400 paired frames per task.

Table 5: Class IoU on A→S (%), with the same branch and fusion conventions as Table 4.

Class	PLATO-TTA			ReDi-TTA		
	2D	3D	xM	2D	3D	xM
car	74.03	76.79	77.19	76.87	79.09	78.07
truck	43.02	51.18	51.54	48.24	52.53	50.59
bike	40.99	50.50	48.06	47.15	52.93	50.41
person	36.76	41.39	44.37	42.88	47.75	45.97
road	90.33	92.06	92.09	91.21	92.14	91.75
parking	6.92	5.79	5.84	6.16	5.52	5.76
sidewalk	63.33	65.71	66.50	65.19	67.02	66.26
building	53.91	58.33	57.70	55.95	58.52	57.32
nature	79.80	80.74	82.46	81.54	83.17	82.51
other-objects	27.00	28.56	28.55	28.07	29.32	28.74
mIoU	51.61	55.10	55.43	54.33	56.80	55.74

Table 6: Class IoU on S→S (%), with the same branch and fusion conventions as Table 4.

Class	PLATO-TTA			ReDi-TTA		
	2D	3D	xM	2D	3D	xM
car	67.28	82.59	84.09	81.63	82.42	82.83
bike	18.86	20.02	28.63	35.39	26.30	32.38
person	4.18	6.18	7.55	8.24	6.93	7.71
road	83.45	84.92	86.15	86.34	85.51	86.02
sidewalk	43.80	47.25	48.41	48.91	48.15	48.66
building	52.61	57.75	59.95	59.15	58.69	59.28
nature	67.12	67.68	69.71	69.87	68.74	69.53
pole	18.18	36.15	36.27	34.98	38.31	38.17
other-objects	0.17	2.31	1.49	0.51	1.46	0.96
mIoU	39.52	44.98	46.92	47.22	46.28	47.28

The earlier archive contains 112/80/84 US/AS/SS pairs, sampled every fourth batch after iteration 16. New US captures replace overlaps irrespective of score, producing 484/480/84 distinct frames and 1,048 pairs in total. Selection uses positive full-frame changes and interpretable class corrections. Ground truth is used only for evaluation and post-hoc illustration selection; no adaptation objective or hyperparameter is changed. The examples are not random samples. Frame rankings, source hashes, image paths, and predictions remain in the figure records.

Metrics and display. The US, AS, and SS examples are stream index 569 after 142 updates, SemanticKITTI 00/000773 after 193 updates, and 00/000146 after 48 updates. Their highlighted classes contain 245 truck, 84 bike/rider, and 4,300 road ground-truth points. The AS bike mapping combines bicycles, motorcycles, and riders. Table 7 computes frame and class IoUs over all valid points before cropping. Frame mIoU uses the fixed task classes with zero for zero-union classes; repairs and new errors cover all valid classes. These scores differ from pooled full-stream mIoU.

Table 7: Selected frames: PLATO-TTA / ReDi-TTA. IoUs use all valid points in each complete frame. Repairs and new errors cover all valid classes; indices are zero-based.

Task	Index	Class	Frame mIoU (%)	Class IoU (%)	Repairs / new errors
US	569	truck	27.84 / 37.24	11.43 / 85.37	241 / 28
AS	773	bike	45.85 / 51.47	30.00 / 68.18	198 / 90
SS	146	road	44.62 / 51.37	53.67 / 70.98	2,065 / 837

All panels use the original `draw_points_image_labels` renderer, point size 2.0, opacity 0.5, and matched ignore colors. The main figure uses cyan for the highlighted class and gray elsewhere; full views use the default NuScenesLidarSeg, A2D2, and Synthia palettes. Palette changes are temporary and leave labels, point locations, and renderer code unchanged. NuScenes RGB is resized

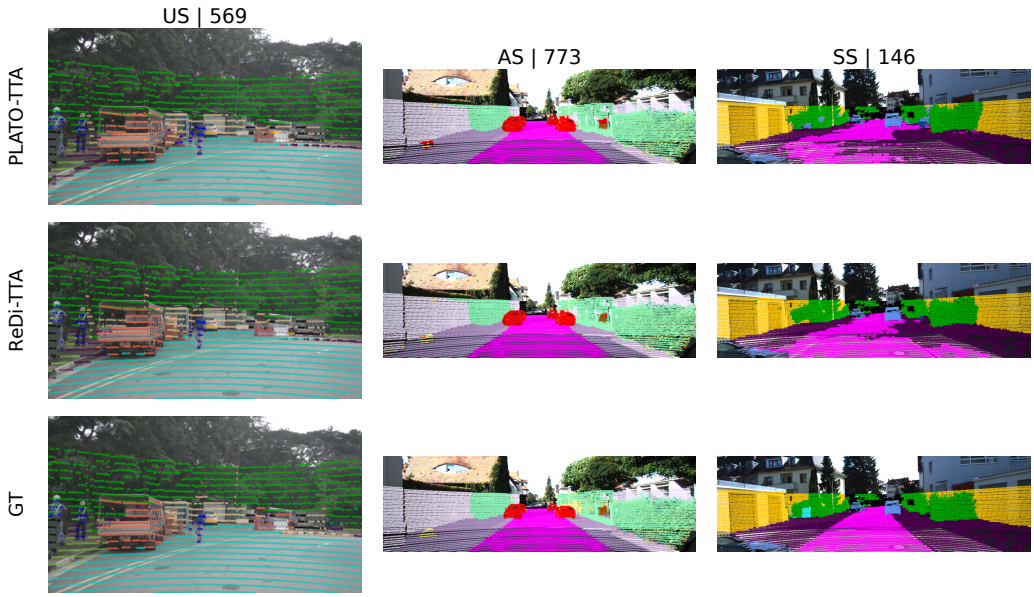


Figure 4: Uncropped views of the main-figure examples using the default dataset palettes. Rows show PLATO-TTA, ReDi-TTA, and ground truth at identical point locations. All classes and image regions are retained; column indices are zero-based stream positions.

to 800×450; SemanticKITTI retains its native display resolution. Corresponding crops and insets preserve aspect ratios. Insets enlarge existing pixels, with no added points, altered predictions, or contrast changes.

C Ablations and sensitivity analysis

C.1 Student input, guidance, and fitting

Table 8 separates the RBC and PSD choices on the ReDi-TTA stream. All students use the budget in Appendix A and product fusion. Own-modality inputs remove cross-modal features; student-fed targets allow head predictions into target construction. Both controls preserve the base states used for adaptation and EMA updates. Their xM reductions are 1.160 and 2.292 points, respectively.

For PSD, hard targets and removal of the proximal penalty reduce xM by 0.886 and 0.808 points. Pre-update features reduce it by 0.42 points. These single-seed controls do not establish uniformly positive effects or statistical significance. Disabling PSD in Table 2 removes feature refresh and proximal regularization while retaining soft-target fitting; it does not stop student learning.

Table 8: Individual student controls on ReDi-TTA (mIoU, %). All runs use the same fitting budget and product fusion. Δ is the ablated xM minus the complete-model xM, using the tabulated scores.

Variant	2D	3D	xM	Δ xM
Complete model	52.06	51.98	53.65	0.000
RBC: own features only	45.631	49.782	52.490	-1.160
RBC: student-fed targets	51.263	51.393	51.358	-2.292
PSD: pre-update features	52.277	52.334	53.226	-0.424
PSD: no proximal term	52.000	51.945	52.842	-0.808
PSD: hard targets	51.828	51.704	52.764	-0.886

C.2 PREDICTION FUSION

Table 9 applies both fusion rules to identical student predictions without changing the learned state. Product fusion improves all three PLATO scores and all six Latte and MM-TTA scores. This output effect is distinct from the complete-model gains over native baselines.

Table 9: Fusion of identical student predictions (mIoU, %). Difference is product minus arithmetic, using the tabulated scores; neither readout directly incorporates teacher probabilities.

Base	Task	Arithmetic	Product	Difference
PLATO	US	52.7089	53.65	0.9411
PLATO	AS	55.7177	56.49	0.7723
PLATO	SS	47.1997	48.03	0.8303
Latte	US	48.4132	49.25	0.8368
Latte	AS	56.1212	56.80	0.6788
Latte	SS	39.2006	39.89	0.6894
MM-TTA	US	46.0838	46.81	0.7262
MM-TTA	AS	53.3421	53.98	0.6379
MM-TTA	SS	34.7425	35.29	0.5475

Table 10: PLATO-US fusion controls (mIoU, %). Difference is student product minus student arithmetic, using the tabulated scores. The 200-batch row is a development-prefix result.

Batches	Base avg.	Base prod.	Student avg.	Student prod.	Difference
200	41.4704	43.0574	43.0392	43.6593	0.6201
3841	49.8729	51.2128	52.7089	53.65	0.9411

Table 10 also tests a fusion-matched PLATO-US base. On the full stream, product fusion raises base xM from 49.8729 to 51.2128; the product-fused student reaches 53.65, retaining a 2.4372-point gain from the tabulated scores. The comparison reduces the fusion mismatch in the main table but does not replace its independent arithmetic-fusion reference. The 200-batch screening result is reported separately from the full-stream outcome.

C.3 TEACHER SUPPORT AND SCALAR WEIGHTING

The following controls evaluate direct refinement on fixed inputs within each base trajectory. US200 denotes 200 batches, not a shared number of frames across PLATO and Latte. These development prefixes were inspected repeatedly and do not provide independent validation of parameter choices.

Teacher-support strength. We vary the exponent in

$$q^m(\gamma) = \mathcal{N}\big(p^m \odot (e^2 \odot e^3)^\gamma\big), \qquad \gamma \in \{0, 0.25, 0.5, 1, 2, 4\}. \tag{16}$$

where $\gamma = 0$ recovers the raw branch and $\gamma = 1$ is the final no-memory CCPR rule. Figure 5 and Table 11 retain all six values. PLATO 2D peaks at 1, Latte 2D at 2, and both 3D branches improve up to 4. Thus, the fixed exponent is not uniformly optimal on these prefixes. This sweep does not vary the PSD coefficient ρ; no controlled ρ sweep is reported.

Scalar selection and averaging. Maximum-confidence selection compares the raw branch with both EMA vectors. Replacement requires a maximum-probability margin of at least τ over the raw prediction. Table 12 includes all seven thresholds: $\tau = 0$ selects maximum confidence, and $\tau = 1$ keeps the raw branch. Table 13 gives all 15 quarter-step nonnegative weight triples and equal weighting; weights apply to the raw branch, own EMA, and opposite EMA and sum to one. Both grids were specified before the US200 runs. Branch-only controls leave xM fixed. Archived diagnostics additionally contain inverse-entropy weighting and repair/damage aggregates.

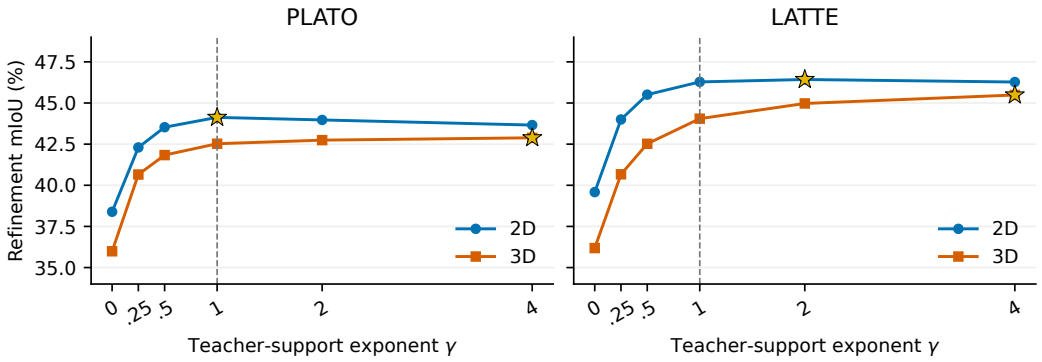


Figure 5: Teacher-support strength on US200. Each framework uses one trajectory. The dashed line denotes $\gamma = 1$; stars mark the best observed branch value. The frameworks process different frame counts per batch.

Table 11: Teacher-support exponent on US200 (refinement mIoU, %).

γ	PLATO		Latte	
	2D	3D	2D	3D
0	38.391	35.990	39.586	36.184
0.25	42.299	40.654	43.999	40.668
0.5	43.528	41.835	45.508	42.521
1	44.126	42.525	46.281	44.048
2	43.970	42.743	46.428	44.972
4	43.659	42.886	46.275	45.486

Table 12: Confidence-selection margin on US200 (refinement mIoU, %).

τ	PLATO		Latte	
	2D	3D	2D	3D
0	40.258	39.816	42.285	42.179
0.025	40.496	39.760	42.570	41.836
0.05	40.577	39.700	42.604	41.579
0.1	40.673	39.546	42.553	41.057
0.2	40.525	39.182	42.246	40.134
0.4	39.586	37.829	41.083	38.096
1	38.391	35.990	39.586	36.184

Table 13: US200 arithmetic weighting (refinement mIoU, %). Weights apply to the raw prediction, own EMA, and opposite EMA. All quarter-step triples and the equal-weight setting are retained.

w_p	w_{own}	w_{opp}	PLATO 2D	PLATO 3D	Latte 2D	Latte 3D
0.0	0.0	1.0	31.694	37.377	34.361	39.543
0.0	0.25	0.75	34.577	39.169	36.983	41.644
0.0	0.5	0.5	40.196	40.196	42.859	42.859
0.0	0.75	0.25	39.169	34.577	41.644	36.983
0.0	1.0	0.0	37.377	31.694	39.543	34.361
0.25	0.0	0.75	34.626	39.028	36.973	41.706
0.25	0.25	0.5	40.571	41.149	42.884	43.227
0.25	0.5	0.25	40.381	37.966	42.151	37.694
0.25	0.75	0.0	38.013	33.108	39.847	34.890
0.5	0.0	0.5	40.774	40.907	42.803	43.437
0.5	0.25	0.25	40.798	39.486	42.187	38.239
0.5	0.5	0.0	38.731	35.525	40.030	35.402
0.75	0.0	0.25	40.289	38.395	41.650	38.552
0.75	0.25	0.0	38.837	36.390	39.852	35.826
1.0	0.0	0.0	38.391	35.990	39.586	36.184
1/3	1/3	1/3	41.385	40.459	43.235	39.363

D CLASS-WISE PROBABILITY ANALYSIS

These experiments evaluate direct refinement on fixed base trajectories without retraining the final student. US200/400/600 denote diagnostic batch counts. Within-trajectory comparisons are controlled; equal batch indices across frameworks do not imply equal frame prefixes. The repeatedly inspected development streams are not independent validation sets.

D.1 CONDITIONAL CORRECTNESS DISCRIMINATION

The US200 diagnostic selects valid points where the raw branch and opposite EMA teacher predict different top-1 classes, without conditioning selection on correctness. The score is the opposite-teacher probability of the raw predicted class. Correct and incorrect predictions are compared within strata defined by that class, opposite-teacher maximum probability (five bins), opposite-teacher entropy (four bins), and own-branch maximum probability (64 bins). Ground truth determines correctness only.

Table 14 reports point-pair-weighted AUC, score-quantization bounds, early/late windows, and comparable-point coverage. Coverage measures eligible points in strata containing comparable predictions. All four AUCs exceed 0.5. Finer own-confidence stratification lowers some scores without removing this discrimination (Table 15). These are conditional associations, not evidence of calibration or causal effects on student learning. Quantization bounds are not statistical confidence intervals.

Table 14: US200 conditional correctness AUC. Bounds reflect score quantization, not confidence intervals. Coverage is the percentage of eligible points in comparable strata.

Base / branch	AUC	Bounds	First 100	Last 100	Coverage (%)
PLATO 2D	0.6341	0.6250–0.6432	0.6316	0.6412	97.32
PLATO 3D	0.6843	0.6752–0.6934	0.6900	0.6714	97.20
Latte 2D	0.6388	0.6304–0.6473	0.6006	0.6742	98.61
Latte 3D	0.6943	0.6851–0.7035	0.6728	0.7035	96.50

Table 15: US200 conditional AUC with different own-branch confidence bins. Other strata are fixed.

Base / branch	4 bins	16 bins	64 bins
PLATO 2D	0.6449	0.6329	0.6341
PLATO 3D	0.6929	0.6896	0.6843
Latte 2D	0.6742	0.6511	0.6388
Latte 3D	0.7101	0.7024	0.6943

D.2 PROBABILITY CORRESPONDENCE AND TEMPORAL CONTINUATION

Non-maximal class support. A tail permutation preserves the top-1 class, maximum probability, probability multiset, and entropy of each EMA vector, but changes non-maximal class assignments. Named-class probabilities and their class-conditional marginals are not preserved. On a separate US200 trajectory, both tail permutations reduce all four branch scores; removing the opposite EMA has larger effects (Table 16). Removing the own EMA improves 3D mIoU by 0.481 points for PLATO and 2.173 for Latte. An informative input can therefore be detrimental under fixed product weighting. These branch-only interventions hold xM unchanged.

Table 16: US200 branch interventions minus same-trajectory no-memory refinement (mIoU points). Positive entries favor the intervention. Branch-only changes leave xM fixed.

Intervention	PLATO 2D	PLATO 3D	Latte 2D	Latte 3D
Remove opposite EMA	-5.294	-6.969	-6.263	-8.659
Permute opposite EMA tail	-1.833	-2.561	-2.091	-2.107
Permute own EMA tail	-3.730	-3.216	-3.147	-4.129
Remove own EMA	-1.365	+0.481	-0.455	+2.173

The US400 experiments either uniformize the non-top-1 mass or permute its class assignment (Table 17). Both reduce all six refinement scores. Uniformization changes entropy; permutation preserves it and more directly tests class correspondence. Counterexamples remain: maximum-confidence selection for Latte 3D has lower error in the archived high-disagreement group despite lower overall mIoU. Counting shared teacher support twice changes xM by $+0.036$ for PLATO and

-0.124 for Latte. These observations do not identify a universally preferred fusion rule or teacher multiplicity.

Table 17: US400 refinement mIoU (%). Tail permutation preserves top-1, maximum probability, and entropy; uniformization changes entropy. Comparisons use fixed base trajectories.

Base	Support	2D	3D	xM
PLATO	No-memory reference	48.782	47.715	48.787
PLATO	Uniform non-top-1 tail	46.122	45.195	47.520
PLATO	Permuted non-top-1 tail	44.708	43.550	46.647
Latte	No-memory reference	51.567	49.632	51.775
Latte	Uniform non-top-1 tail	48.915	45.494	50.841
Latte	Permuted non-top-1 tail	47.675	44.617	49.678

Later stream interval. The US600 continuation evaluates the predeclared batches 401–600 (Table 18). Compared with opposite-EMA tail permutation, no-memory refinement improves PLATO 2D/3D by 1.234/3.541 points and Latte 2D/3D by 2.643/2.385. Both EMA-tail permutations also reduce mIoU in the 401–500 and 501–600 subwindows. Removing own-EMA support improves Latte 3D by 2.719 points, while maximum-confidence and inverse-entropy selection obtain higher point accuracy despite lower mIoU. This extends the diagnostic within the same studied stream, not to an independent validation set or a final-student evaluation.

Table 18: US600 continuation: no-memory refinement minus each control in batches 401–600 (mIoU points). Each comparison uses the same base trajectory.

Branch	Opp. perm.	Own perm.	Mean	Max conf.	Inv. entropy
PLATO 2D	+1.234	+4.826	+2.687	+3.446	+2.503
PLATO 3D	+3.541	+3.209	+2.213	+3.082	+1.994
Latte 2D	+2.643	+3.756	+3.821	+5.333	+4.469
Latte 3D	+2.385	+4.586	+4.655	+2.372	+2.359

D.3 Repairs and newly introduced errors

A repair changes an incorrect prediction to the correct class; damage changes a correct prediction to an incorrect class. Net corrections equal repairs minus damage. Relative error reduction (RER) is the net count divided by the raw error count. Table 19 compares historical no-memory refinement with same-trajectory raw predictions over full US, covering 47,604,330 valid points per output. All six outputs have positive net corrections.

Table 20 separates agreement and disagreement of the two raw top-1 predictions. Net corrections concentrate in the disagreement group. All PLATO agreement-group outputs incur net damage; Latte has small mixed effects. These point counts describe direct refinement against historical raw references, not student gains against the native baselines in Table 1.

E Reproducibility and historical controls

E.1 Reference variability and earlier student runs

Table 21 subtracts the fixed independent baseline from the internal-base readout in each final student run. The largest xM drift is $+0.1861$ on PLATO-SS, compared with the student gain of $+1.11$ over the same reference, using the tabulated main-table scores. The fixed baseline remains the denominator for the main comparisons.

Table 22 reports earlier AS/SS students with arithmetic fusion. Both PLATO-SS executions are retained: the repeat improves on the first in 3D and xM. These are separate executions from the final product-fusion runs and cannot isolate the fusion effect; Table 9 uses identical predictions for

Table 19: Historical full-US refinement: repaired points, new errors, net corrections, and relative error reduction (RER, %). Each output covers 47,604,330 valid points.

Base	Branch	Repaired	Damaged	Net	RER (%)
PLATO	2D	2,192,228	838,582	1,353,646	20.094
PLATO	3D	1,698,951	954,499	744,452	12.478
PLATO	XM	831,510	680,058	151,452	2.876
Latte	2D	2,402,338	654,094	1,748,244	21.879
Latte	3D	1,805,761	660,850	1,144,911	15.854
Latte	XM	598,151	420,434	177,717	2.882

Table 20: Relative point-error reduction (%) by agreement of the two raw top-1 predictions. Negative values indicate net damage.

Base	Group	2D	3D	xM
PLATO	All	20.094	12.478	2.876
PLATO	Disagreement	33.279	23.436	6.045
PLATO	Agreement	-3.974	-3.983	-0.954
Latte	All	21.879	15.854	2.882
Latte	Disagreement	30.912	23.405	4.587
Latte	Agreement	-0.028	0.033	0.079

that comparison. Neither the internal-base differences nor these historical runs constitute multi-seed uncertainty estimates or replacements for the final results.

Table 21: Internal-base score minus the independent reference (mIoU points). These execution differences are not student gains or multi-seed uncertainty estimates.

Base	Task	2D	3D	xM
PLATO	US	-0.0310	-0.0398	-0.0577
PLATO	AS	0.0014	0.0034	0.0020
PLATO	SS	0.2843	0.0675	0.1861
Latte	US	-0.0011	-0.0069	-0.0103
Latte	AS	0.0003	-0.0014	-0.0014
Latte	SS	0.0003	-0.0012	0.0005
MM-TTA	US	-0.0010	-0.0032	-0.0039
MM-TTA	AS	-0.0000	-0.0025	-0.0000
MM-TTA	SS	-0.0000	-0.0040	-0.0001

Table 22: Historical AS/SS student runs with arithmetic fusion (mIoU, %). These are separate executions from the final product-fusion runs; both PLATO-SS runs are retained.

Base	Task	2D	3D	xM
PLATO-TTA	AS	54.326	56.785	55.710
PLATO-TTA	SS	45.459	44.109	45.020
PLATO-TTA (repeat)	SS	47.420	46.699	47.613
Latte	AS	53.599	57.338	56.126
Latte	SS	37.493	39.763	39.196
MM-TTA	AS	51.266	53.979	53.346
MM-TTA	SS	33.052	35.707	34.741

E.2 Historical output and memory controls

Table 23 retains ten US500 output interventions on one base trajectory per framework. They cover raw predictions, a memory-bearing predecessor, history and teacher-support removal, restricted support, top-2 output restriction, alternative refined-output fusion, and own-predictor–EMA averaging. The no-memory row implements the final CCPR rule. None changes base adaptation. PLATO uses 500 batches of four frames; Latte uses 500 batches of six.

Within the memory-bearing predecessor, removing current EMA support leaves historical factors intact, and same-modality current support can retain opposite-modality information through history. Neither control removes all corresponding teacher information. The top-2 row restricts output candidates and does not retrain a student.

The full-US memory comparison covers 47,604,330 valid points per framework (Table 24). Removing memory changes PLATO 2D, 3D, and xM by $+0.009$, -0.085, and $+0.064$ points, respectively. The changes for Latte are -0.160, -0.559, and -0.222 points. Scores are computed from stored confusion matrices with same-trajectory arithmetic-fused raw references; the historical Latte baseline

Table 23: Historical US500 output controls (mIoU, %). Each framework uses one fixed trajectory. The memory-bearing predecessor differs from the final no-memory CCPR rule.

Variant	PLATO			Latte		
	2D	3D	xM	2D	3D	xM
Raw predictor	42.970	42.254	47.548	43.079	40.317	47.919
Memory-bearing predecessor	49.788	49.227	50.089	50.894	49.025	50.893
No memory (final operator)	49.597	49.093	49.998	50.812	48.531	50.761
Without current EMA support	44.220	43.686	49.037	44.589	42.044	50.805
Same-modality current EMA	45.150	42.543	50.089	44.931	41.393	50.893
Support only	49.350	49.350	49.350	50.479	50.479	50.479
Top-2 output restriction	47.945	47.544	50.064	48.872	46.914	50.855
Product of refined branches	49.788	49.227	49.986	50.894	49.025	50.782
Average of refined branches	49.788	49.227	49.950	50.894	49.025	50.588
Own predictor–EMA average	44.045	40.731	48.399	43.452	39.808	47.849

Table 24: Historical full-US memory controls (refinement mIoU, %). PLATO and Latte process 3,841 and 2,561 batches, respectively, covering 15,364 frames each. Raw xM uses arithmetic fusion.

Variant	PLATO			Latte		
	2D	3D	xM	2D	3D	xM
Raw predictor	43.613	45.381	49.839	37.703	41.238	46.193
With historical memory	52.468	52.615	53.113	47.343	48.271	48.828
Without memory (final)	52.477	52.530	53.177	47.184	47.713	48.606

differs from its native fusion in the main table. Final CCPR excludes point memory, but these controls establish neither zero performance cost nor an efficiency advantage.