# STEM: SELECTIVE TEMPORAL EVIDENCE MEMORY FOR ONLINE VECTORIZED HD MAPPING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Online vectorized HD mapping incrementally constructs structured map elements from streaming observations. Beyond per-frame accuracy, a practical system should maintain stable predictions across frames. Existing temporal mappers propagate or aggregate historical representations, but memories within a persistent track can have heterogeneous relations to the current query. They do not explicitly separate relation strength, which determines whether a memory is selected, from relation polarity, which determines how selected evidence enters the update. We propose Selective Temporal Evidence Memory (STEM), centered on an inference-time memory operator that makes this distinction. Triple Correlation Aggregation (TCA) accumulates complementary content, task-specific correspondence, and geometric-temporal context evidence into a signed query-memory score. Its magnitude drives sparse memory selection, while its polarity determines branch-specific integration into a query-conditioned residual update. Trackwise Contrastive Regularization (TCR) is a training-only contrastive regularizer on TCA's shared projected query-memory embeddings: it promotes consistency among score-positive same-track pairs and separates confusable cross-track embeddings. Experiments on nuScenes and Argoverse 2 demonstrate gains in mapping accuracy and temporal stability with comparable inference efficiency.

## 1 INTRODUCTION

Vectorized high definition (HD) maps provide structured, centimeter level representations of the driving environment. They support localization, motion planning, scene understanding, and trajectory prediction (Dong et al., 2025). Traditional HD mapping pipelines (Shan et al., 2020; Shan & Englot, 2018; Leitenstern et al., 2024; Han et al., 2025) rely on offline databases constructed through large scale surveying and manual maintenance. They are costly to build and difficult to keep current, especially in dynamic or unseen areas. Recent online vectorized HD mapping methods (Ma et al., 2024) instead infer structured map elements directly from onboard sensor data, enabling real time map construction without prebuilt maps.

Existing online HD mapping methods have made substantial progress in both representation and efficiency. Early approaches (Li et al., 2022; Zhang et al., 2024b) formulate online map generation as a BEV segmentation problem and predict dense rasterized representations from onboard observations. Although effective, these methods require raster to vector conversion to recover structured map elements, increasing computational cost and complicating inference. Later works (Liao et al., 2024; Hao et al., 2024; Gao et al., 2025) directly predict sparse point sequences with query based architectures. More recent methods incorporate temporal cues through memory propagation or cross frame association. For example, StreamMapNet (Yuan et al., 2024) adopts recurrent temporal aggregation, while MapTracker (Chen et al., 2025) introduces a tracking based formulation with temporal BEV and vector memories. However, temporal memory may still accumulate distracting or outdated cues, causing flickering predictions, localization jitter, and shape drift (Shan et al., 2026).

Recent benchmarking work StableHDMap (Shan et al., 2026) provides a clearer view of this issue. It shows that temporal stability is distinct from mapping accuracy: a strong mAP does not necessarily imply stable predictions over time. It also shows that even when temporal information is incorporated, the improvement in stability metrics remains limited. These observations motivate a closer examination of persistent instance memories. Within an associated track, historical entries can ex-



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.

hibit heterogeneous query-memory relations, differing in both strength and polarity. Prior methods already propagate, fuse, select, or store historical information through varied mechanisms (Yuan et al., 2024; Chen et al., 2025; Liao et al., 2024); the gap is not the use or selection of history itself. Rather than asking only what historical information should be propagated or retained, we focus on how individual stored entries within a persistent track relate to the current query. Representations from different tracks can also remain confusable.

To address this gap, we propose **Selective Temporal Evidence Memory (STEM)**, which separates relation strength from relation polarity in persistent memory use while leaving the BEV representation unchanged. Its inference-time operator, **Triple Correlation Aggregation (TCA)**, accumulates complementary content, task-specific correspondence, and geometric-temporal context evidence into a signed compatibility score. The score magnitude selects sparse same-track evidence, while its polarity determines how selected entries are integrated into a query-conditioned residual update. **Trackwise Contrastive Regularization (TCR)** is an auxiliary, training-only contrastive regularizer on TCA's shared projected query-memory embeddings, promoting score-positive same-track consistency and cross-track discrimination; TCR is removed at inference.

Our contributions are summarized as follows:

- We formulate persistent temporal-memory use around heterogeneous query-memory relations among entries within an associated track, explicitly separating relation strength (magnitude for retrieval) from relation polarity (sign for integration).
- We realize this formulation with TCA, which estimates a signed query-memory compatibility from complementary static, matching, and geometric-temporal evidence; its magnitude drives sparse Top-$K$ retrieval, while its polarity controls branch-specific integration.
- We use TCR only as auxiliary training-time support and validate STEM on nuScenes and Argoverse 2. STEM surpasses MapTracker, the prior state-of-the-art method included in our comparisons, with mAP/mAS gains of 2.07/3.84 points on nuScenes and 0.70/2.98 points on Argoverse 2, at comparable inference efficiency.

## 2 Related Work

### 2.1 Vectorized HD Map Construction

Conventional HD map construction relies on SLAM based pipelines and costly offline annotation Shan & Englot (2018), limiting large scale deployment. Early online approaches use BEV segmentation Zhang et al. (2024b); Jia et al. (2024); Li et al. (2022); Gao et al. (2025) followed by postprocessing, but this decoupling causes error propagation and limited geometric fidelity. Later methods adopt end to end vectorized representations. VectorMapNet Liu et al. (2023) predicts sparse polylines, while MapTR Liao et al. (2023) models ordered point sequences. Bézier curves in BeMapNet Qiao et al. (2023) and pivot point parameterizations in PivotNet Ding et al. (2023) improve compactness and smoothness. InstaGraM Shin et al. (2025) uses instance level graphs, GeMap Zhang et al. (2025) learns geometric relations, and Stream Query Denoising Wang et al. (2024) improves query based decoding. Most methods still process frames independently and do not exploit persistent temporal evidence.

### 2.2 Temporal Modeling and Stability

Temporal online mapping exploits history through several distinct mechanisms. StreamMapNet (Yuan et al., 2024) recurrently propagates BEV features and map queries, whereas MapTracker (Chen et al., 2025) associates map instances and fuses a subset of raster and vector memories selected by temporal distance strides. MemFusionMap (Song et al., 2025) fuses working-memory BEV features, while PredMapNet (Lang et al., 2026) stores per-instance rasterized map histories, injects them into track queries, and adds short-term future guidance. Thus, prior methods do retain and, in some cases, select historical information; their mechanisms organize it through recurrent state, temporal strides, feature fusion, or rasterized map priors. STEM addresses a different granularity: for each current query and stored same-track vector entry, TCA computes a query-conditioned signed compatibility score. Its magnitude selects the top-$K$ entries regardless of sign, while its polarity routes selected entries through separate aggregation branches.

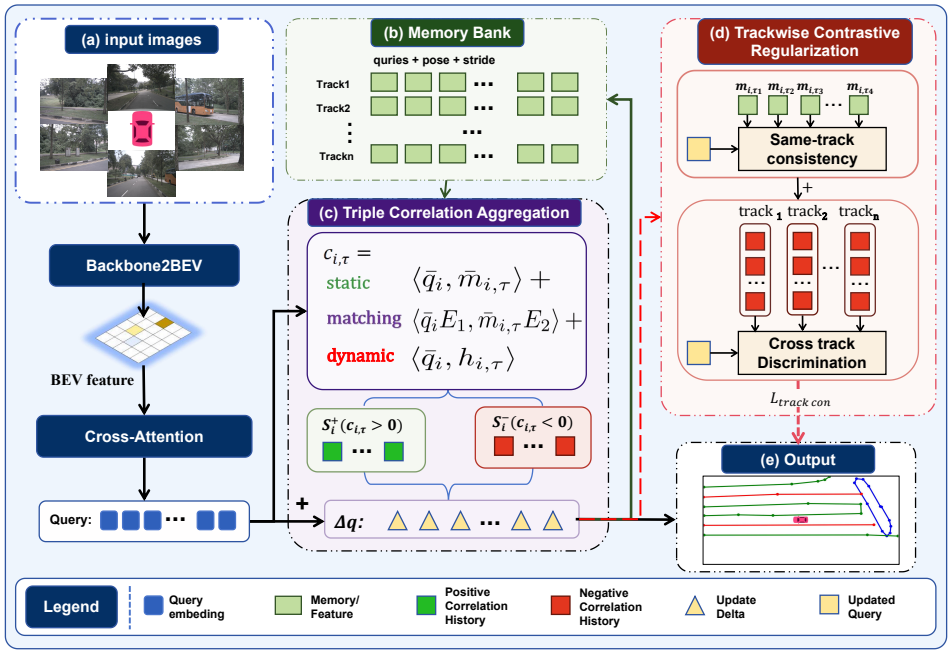


Figure 1: Overview of the proposed STEM framework. (a) Multiple input views are encoded into BEV features and decoded into query embeddings. (b) A memory bank stores historical instance features together with pose and stride information. (c) TCA estimates signed query-memory relations, retrieves memories by score magnitude, and integrates them according to polarity. (d) TCR, used only during training, improves intra-track consistency and separates confusable representations from other tracks. (e) Refined queries yield temporally stable map predictions.

Training-time temporal consistency is addressed by DTCLMapper (Li et al., 2024), which combines instance-level contrastive and map-level geometric consistency supervision, and MapTCL (Kim et al., 2026), which introduces bidirectional vector alignment and raster-map consistency objectives. StableHDMap (Shan et al., 2026) further evaluates temporal stability separately from mapping accuracy. Complementing these objectives, TCR applies training-only contrastive regularization directly to the normalized query-memory embeddings shared with TCA: track identity identifies same-track pairs with positive TCA scores and cross-track candidates, from which confusable hard negatives are mined. Recent methods thus span temporal propagation, historical-memory fusion, tracking, rasterized history, and consistency regularization. STEM instead focuses on how individual entries within an associated instance track are utilized, using a signed query-memory relation whose magnitude determines retrieval and whose polarity determines integration.

# 3 METHOD

## 3.1 OVERVIEW OF STEM

STEM centers on TCA, its inference-time memory operator, and uses TCR only to regularize the shared projected query-memory embeddings during training. Figure 1 shows their roles in the temporal mapping pipeline. After multiview images are encoded into BEV features and decoded into vector queries, the temporal mapper establishes instance associations. For each associated query-memory pair, TCA estimates a signed compatibility score from complementary relation evidence. Its magnitude selects sparse historical evidence, and its polarity controls how selected entries are integrated into a query-conditioned residual update. During training, TCR promotes consistency among score-positive same-track embeddings and separates confusable cross-track embeddings. TCR is removed at inference and does not introduce another aggregation pathway.

STEM maintains an instance wise memory bank for tracked map elements. For each active instance $i$ at frame $t$, the bank stores a fixed capacity history $\mathcal{M}_i^t = \{m_{i,\tau}\}_{\tau=1}^{T_i}$, where $T_i \leq 20$ follows

the memory capacity of MapTracker. Each entry also stores its frame index and ego pose metadata. Given the current frame, we derive a relative pose cue $p_{i,\tau}$ and a temporal stride index $s_{i,\tau}$ for every memory. The bank appends newly propagated instance features and evicts the oldest entries when it reaches capacity, forming the candidate history processed by TCA.

### 3.2 Triple Correlation Aggregation (TCA)

Figure 2 summarizes TCA. For each same-track query-memory pair, TCA estimates a signed compatibility score $c_{i,\tau}$. Its magnitude $|c_{i,\tau}|$ encodes relation strength and determines sparse retrieval, while $\mathrm{sign}(c_{i,\tau})$ encodes relation polarity and determines how the selected memory participates in integration. To estimate this signed relation, TCA combines three complementary sources of evidence: static, matching, and geometric-temporal correlation.

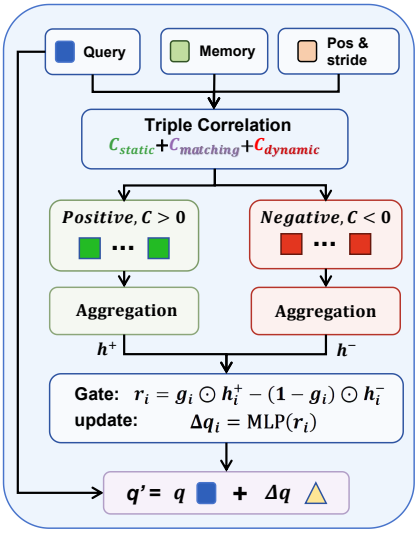


Figure 2: Structure of TCA. It estimates a signed query-memory relation from complementary evidence; score magnitude governs sparse retrieval, while score polarity governs branch-specific integration into residual update $\Delta q_i^t$.

**Relation estimation from complementary evidence.** The *static correlation* measures direct content compatibility between the current query and historical memory features. The *matching correlation* measures task-specific correspondence in a learned compact space. The *dynamic correlation* evaluates compatibility between the current query and the geometric-temporal context in which the memory was observed.

Let $d$ denote the correlation dimension. We first project and normalize the current query and historical memory as $\bar{q}_i = \mathrm{normalize}(W_q q_i^t) \in \mathbb{R}^d$ and $\bar{m}_{i,\tau} = \mathrm{normalize}(W_m m_{i,\tau}) \in \mathbb{R}^d$. For the matching branch, learnable projections $E_1, E_2 \in \mathbb{R}^{d \times r}$ with $r < d$ map the two representations into a compact matching space:

$$\tilde{q}_i = \mathrm{normalize}(\bar{q}_i E_1),$$
$$\tilde{m}_{i,\tau} = \mathrm{normalize}(\bar{m}_{i,\tau} E_2).$$

The relative pose $p_{i,\tau}$ and temporal stride $s_{i,\tau}$ are encoded as

$$h_{i,\tau} = \mathrm{normalize}\left(\mathrm{MLP}_d([p_{i,\tau}; s_{i,\tau}])\right) \in \mathbb{R}^d.$$

TCA then combines the three correlations as

$$c_{i,\tau} = \underbrace{\langle \bar{q}_i, \bar{m}_{i,\tau} \rangle}_{\text{static}} + \underbrace{\langle \tilde{q}_i, \tilde{m}_{i,\tau} \rangle}_{\text{matching}} + \underbrace{\langle \bar{q}_i, h_{i,\tau} \rangle}_{\text{dynamic}}. \tag{1}$$

Each term is an inner product between L2-normalized representations and lies in $[-1, 1]$, placing the three sources on comparable scales. We adopt additive composition to accumulate complementary relation evidence in a shared signed score space. This preserves an explicit signed compatibility score whose magnitude and polarity serve distinct roles in memory utilization. The matching branch uses two $d$-to-$r$ projections to compare representations in a compact task-specific space. For the dynamic term, $p_{i,\tau}$ contains the relative quaternion in $(x, y, z, w)$ order and the three-dimensional translation, while $s_{i,\tau}$ denotes the temporal stride. Together, they encode the geometric-temporal observation context of the historical memory.

TCA evaluates only same-instance query-memory pairs. The sign of the score is a learned relation polarity, not a ground-truth semantic label.

**Strength-based sparse selection.** TCA uses relation strength to select evidence: it retains the historical memories with the largest score magnitudes, independently of their polarity, and then partitions them by score sign.

Specifically, the selected indices are

$$\mathcal{S}_i = \mathrm{TopK}\left(\{|c_{i,\tau}|\}_{\tau=1}^{T_i}\right), \quad \mathcal{S}_i^+ = \{\tau \in \mathcal{S}_i : c_{i,\tau} \geq 0\}, \quad \mathcal{S}_i^- = \mathcal{S}_i \setminus \mathcal{S}_i^+. \tag{2}$$

where TopK returns the $K$ largest magnitudes. Writing $c_{i,\tau} = \text{sign}(c_{i,\tau})|c_{i,\tau}|$ makes the separation explicit: $|c_{i,\tau}|$ ranks memories for sparse selection, while $\text{sign}(c_{i,\tau})$ assigns selected memories to a polarity branch. We use $K = 3$ during training and inference (Table 8); empty sets yield zero features.

**Polarity-aware integration.** After selection, TCA aggregates the two polarity subsets through separate branches:

$$h_i^{\pm} = \sum_{\tau \in \mathcal{S}_i^{\pm}} \text{softmax}(\pm c_{i,\tau}) W^{\pm} m_{i,\tau}. \tag{3}$$

where each softmax is normalized within its subset, and $W^+$ and $W^-$ are branch-specific projections. Both branches use non-negative normalized weights, but their outputs enter the query-conditioned fusion with opposite signs. Thus, polarity is not merely used to reweight memories: it changes how selected historical evidence enters the residual update. The simplified fixed-value analysis in Appendix A shows how this operation can produce directions outside the convex hull of the original values; this operator-level result does not establish a universal expressivity ordering between complete architectures.

A query-conditioned gate balances the two polarity branches before their signed fusion is mapped to a residual query update:

$$\begin{aligned} g_i &= \sigma(W_g q_i^t), \\ \Delta q_i^t &= \text{MLP}\left(g_i \odot h_i^+ - (1 - g_i) \odot h_i^-\right), \\ q_i^{t\prime} &= q_i^t + \Delta q_i^t. \end{aligned} \tag{4}$$

where $\odot$ denotes elementwise multiplication and $g_i$ is conditioned on the query. The MLP maps the gated, polarity-aware fusion to residual update $\Delta q_i^t$, which is added to the current query.

### 3.3 Trackwise Contrastive Regularization (TCR)

As illustrated in Figure 3, TCR is an auxiliary, training-only contrastive regularizer on the query-memory embeddings shared with TCA, not an inference-time aggregation mechanism. It retrieves same-track and cross-track memories according to track identity, applies consistency regularization to same-track pairs with positive TCA scores, and mines highly similar cross-track memories as hard negatives. The regularizer acts directly on the shared normalized projection space used by TCA's static correlation and as input to its matching and dynamic terms.

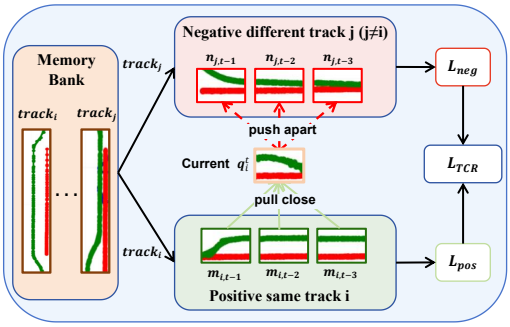


Figure 3: Illustration of TCR. TCR improves consistency among positively correlated pairs within a track and separates hard negatives from other tracks with a margin loss.

For a cross-track memory candidate $n$, we use the same memory projection as TCA and write $\bar{n} = \text{normalize}(W_m n)$. Because $\bar{q}_i$, $\bar{m}_{i,\tau}$, and $\bar{n}$ are L2-normalized, their inner products equal cosine similarities. The intra-track term increases consistency among qualifying same-track pairs, while the cross-track term separates hard negatives from other tracks:

$$\begin{aligned} \mathcal{L}_{\text{pos}} &= \sum_i \sum_{\tau \in \mathcal{S}_i^+} \alpha_{i,\tau} \left(1 - \langle \bar{q}_i, \bar{m}_{i,\tau} \rangle\right), \\ \mathcal{L}_{\text{neg}} &= \sum_i \sum_{n \in \mathcal{N}_i} \max\left(0, \langle \bar{q}_i, \bar{n} \rangle - \gamma\right). \end{aligned} \tag{5}$$

where $\alpha_{i,\tau} = \exp(c_{i,\tau}) / \sum_{\kappa \in \mathcal{S}_i^+} \exp(c_{i,\kappa})$. The set $\mathcal{N}_i$ contains at most $K_{\text{neg}}$ cross-track candidates with the highest projected-space similarities satisfying $\langle \bar{q}_i, \bar{n} \rangle \geq \tau_{\text{neg}}$. The mining threshold $\tau_{\text{neg}}$ determines which cross-track candidates are regarded as sufficiently similar hard negatives, whereas $\gamma$ is the margin offset in the hinge formulation. Under the adopted setting $\tau_{\text{neg}} = 0.5 > \gamma = 0.2$, all mined negatives fall within the active region of the hinge. If $\mathcal{S}_i^+$ is empty, $\mathcal{L}_{\text{pos}}$ is

| Method | Temp. | Mapping Accuracy | | | | Temporal Stability | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | $\mathbf{AP}_p \uparrow$ | $\mathbf{AP}_d \uparrow$ | $\mathbf{AP}_b \uparrow$ | **mAP** $\uparrow$ | **Presence** $\uparrow$ | **Loc** $\uparrow$ | **Shape** $\uparrow$ | **mAS** $\uparrow$ |
| MapTR (Liao et al., 2023) | No | 51.71 | 54.88 | 53.29 | 53.29 | 91.20 | 65.40 | 90.60 | 71.60 |
| GeMap (Zhang et al., 2025) | No | 45.62 | 53.95 | 54.20 | 51.26 | 92.30 | 69.70 | 92.60 | 75.50 |
| MGMap (Liu et al., 2024a) | No | 64.42 | 67.63 | 67.65 | 66.50 | 92.20 | 75.00 | 92.30 | 78.00 |
| MapTRv2 (Liao et al., 2024) | No | 68.10 | 68.33 | 69.70 | 68.70 | 91.50 | 68.60 | 90.90 | 73.90 |
| MapQR (Liu et al., 2024b) | No | 63.36 | 68.03 | 67.67 | 66.35 | 91.80 | 75.60 | 91.60 | 77.80 |
| AMap (Li et al., 2025) | No | 69.97 | 70.13 | 67.67 | 69.26 | N/A | N/A | N/A | N/A |
| MapTR (Liao et al., 2023) | Yes | 56.21 | 59.84 | 60.05 | 58.70 | 94.26 | 71.47 | 93.10 | 72.00 |
| StreamMapNet (Yuan et al., 2024) | Yes | 61.70 | 66.30 | 62.11 | 63.40 | 90.60 | 70.53 | 93.99 | 75.05 |
| HRMapNet (Zhang et al., 2024a) | Yes | 65.61 | 67.36 | 68.53 | 67.21 | 90.80 | 78.14 | 92.52 | 77.86 |
| MapTracker (Chen et al., 2025) | Yes | 73.61 | **73.77** | 72.43 | 73.26 | 92.11 | 83.97 | 97.83 | 84.57 |
| STEM (Ours) | Yes | **80.23** | 70.71 | **74.06** | **75.00** | **94.81** | **86.95** | **98.33** | **88.41** |

Table 1: Quantitative comparison on the nuScenes validation set. We report both mapping accuracy and temporal stability under the reported training settings of each method. *Temp.* denotes whether temporal information is used. $\mathrm{AP}_p$, $\mathrm{AP}_d$, and $\mathrm{AP}_b$ denote the AP of pedestrian crossings, lane dividers, and road boundaries, respectively, and mAP denotes their mean value. Presence, Loc, Shape, and mAS are temporal stability metrics following the StableHDMap protocol. STEM uses memory capacity 20 and $K = 3$; dashes denote unavailable stability metrics.

| Method | Mapping Accuracy | | | | Temporal Stability | | | |
|---|---|---|---|---|---|---|---|---|
| | $\mathbf{AP}_p \uparrow$ | $\mathbf{AP}_d \uparrow$ | $\mathbf{AP}_b \uparrow$ | **mAP** $\uparrow$ | **Presence** $\uparrow$ | **Loc** $\uparrow$ | **Shape** $\uparrow$ | **mAS** $\uparrow$ |
| MapTRv2 (Liao et al., 2024) | 62.9 | 72.1 | 67.1 | 67.4 | 94.60 | 86.39 | 96.23 | 86.86 |
| HRMapNet (Zhang et al., 2024a) | 65.1 | 71.4 | 68.6 | 68.3 | 92.60 | 81.80 | 96.90 | 83.33 |
| MapTracker (Chen et al., 2025) | 76.8 | 78.3 | 74.2 | 76.4 | 96.34 | 88.12 | **98.54** | 90.27 |
| STEM (Ours) | **77.4** | **79.0** | **74.9** | **77.1** | **99.47** | **88.48** | 98.01 | **93.25** |

Table 2: Quantitative comparison on the Argoverse 2 validation set. We report mapping accuracy and temporal stability for pedestrian crossings, lane dividers, and road boundaries. STEM achieves the best mAP and mAS among compared methods on these high-frequency urban sequences.

zero for that query. Minimizing $\mathcal{L}_{\mathrm{pos}}$ promotes consistency among same-track projected embeddings with positive TCA scores, while $\mathcal{L}_{\mathrm{neg}}$ separates confusable cross-track embeddings in the same space. Thus, TCR directly regularizes the shared projected query-memory embeddings used by TCA.

After defining TCR, the overall training objective combines it with the conventional mapping losses:

$$\begin{aligned} \mathcal{L}_{\mathrm{TCR}} &= \lambda_{\mathrm{pos}}\mathcal{L}_{\mathrm{pos}} + \lambda_{\mathrm{neg}}\mathcal{L}_{\mathrm{neg}}, \\ \mathcal{L} &= \mathcal{L}_{\mathrm{det}} + \mathcal{L}_{\mathrm{seg}} + \mathcal{L}_{\mathrm{trans}} + \mathcal{L}_{\mathrm{TCR}}. \end{aligned} \tag{6}$$

The first three terms are conventional baseline losses for map prediction, auxiliary BEV semantics, and temporal propagation, respectively; the two contrastive terms form $\mathcal{L}_{\mathrm{TCR}}$. At inference, TCA remains as the memory utilization operator and TCR is removed. Thus, TCA defines how persistent memories are selected and integrated, while TCR regularizes its shared projected query-memory embeddings during training.

# 4 Experiments

## 4.1 Experimental Setup

**Datasets.** We evaluate STEM on two online HD mapping benchmarks: nuScenes (Caesar et al., 2020) and Argoverse 2 (Wilson et al., 2021). nuScenes serves as our main benchmark. It contains 1,000 driving scenes with synchronized observations from multiple cameras and vectorized map annotations, and is widely used for online HD map construction. To evaluate the applicability of STEM across different mapping benchmarks, we additionally conduct experiments on Argoverse 2, which provides driving sequences with higher sampling frequency and observations from multiple cameras under diverse scenarios. Following common practice, we evaluate three map element classes: pedestrian crossings, dividers, and boundaries.

| TCA | TCR | $\text{AP}_p \uparrow$ | $\text{AP}_d \uparrow$ | $\text{AP}_b \uparrow$ | mAP ↑ | mAS ↑ |
|---|---|---|---|---|---|---|
| × | × | 72.73 | **75.77** | 70.31 | 72.93 | 84.57 |
| ✓ | × | 78.14 | 69.45 | 72.98 | 73.52 | 86.16 |
| ✓ | ✓ | **80.23** | 70.71 | **74.06** | **75.00** | **88.41** |

Table 3: Ablation study of the components on the nuScenes validation set. The first row reports the MapTracker baseline; subsequent rows add TCA and TCR.

| Variant | $\text{AP}_p \uparrow$ | $\text{AP}_d \uparrow$ | $\text{AP}_b \uparrow$ | mAP ↑ | mAS ↑ |
|---|---|---|---|---|---|
| *Single factor variants* | | | | | |
| Static only | **81.23** | 70.58 | 72.08 | 74.63 | 86.72 |
| Matching only | 79.96 | 70.88 | 72.56 | 74.46 | 86.11 |
| Dynamic only | 80.70 | 69.18 | 73.11 | 74.33 | 86.48 |
| *Leave one out variants* | | | | | |
| w/o Static | 80.39 | 70.48 | 72.78 | 74.55 | 87.36 |
| w/o Matching | 80.74 | **71.12** | 72.55 | 74.80 | 87.74 |
| w/o Dynamic | 80.29 | 71.02 | 72.92 | 74.74 | 87.21 |
| Full TCA | 80.23 | 70.71 | **74.06** | **75.00** | **88.41** |

Table 4: TCA factor ablation on nuScenes (final checkpoint).

| Variant | $\text{AP}_p \uparrow$ | $\text{AP}_d \uparrow$ | $\text{AP}_b \uparrow$ | mAP ↑ | mAS ↑ |
|---|---|---|---|---|---|
| Full TCR | **80.23** | 70.71 | **74.06** | **75.00** | **88.41** |
| w/o $\mathcal{L}_{\text{pos}}$ | 77.53 | 70.74 | 71.75 | 73.34 | 84.33 |
| w/o $\mathcal{L}_{\text{neg}}$ | 78.23 | **72.31** | 71.88 | 74.14 | 82.73 |

Table 5: TCR loss ablation on nuScenes at $K = 3$.

| Method | FPS ↑ | Latency ↓ | mAP ↑ | mAS ↑ |
|---|---|---|---|---|
| MapTracker | 14.79 | 67.62 | 72.93 | 84.57 |
| STEM | **15.13** | **66.11** | **75.00** | **88.41** |

Table 6: Inference efficiency on nuScenes ($K = 3$, RTX 4090).

**Evaluation metrics.** We evaluate both mapping accuracy and temporal stability. For mapping accuracy, we report AP for each category, including pedestrian crossings ($\text{AP}_p$), lane dividers ($\text{AP}_d$), and road boundaries ($\text{AP}_b$), together with their mean (mAP).

For temporal stability, we follow StableHDMap (Shan et al., 2026) and report Presence, Localization (Loc), Shape, and mean Average Stability (mAS). Presence evaluates detection consistency across frames, Loc measures positional jitter after alignment across frames, and Shape measures geometric consistency. For each matched instance, the stability score is $\text{Presence} \cdot (\omega\text{Loc} + (1-\omega)\text{Shape})$, where $\omega = 0.7$. Stability for each class is averaged over matched instances, and mAS is the mean score across map categories.

**Implementation details.** We follow MapTracker (Chen et al., 2025) for preprocessing, vector matching, decoding, and optimization. Training uses its three-stage schedule (BEV pretraining, decoder warmup, joint temporal training) on four NVIDIA GeForce RTX 4090 GPUs (24/30 joint epochs on nuScenes/Argoverse 2). Each track stores at most 20 memories; each clip uses the current frame plus four of the previous ten. AdamW ($5 \times 10^{-4}$ learning rate, 0.01 weight decay) trains the final decoder layer with $K = 3$ for training and inference mixing (Table 8). TCR mines up to $K_{\text{neg}} = 4$ hard cross-track negatives per query, ranked and thresholded by cosine similarity in the normalized shared projection space ($\lambda_{\text{pos}} = 0.12$, $\lambda_{\text{neg}} = 0.005$, $\tau_{\text{neg}} = 0.5$, $\gamma = 0.2$). TCA uses $d = 32$, $r = 16$, and a pose/stride MLP of $8 \to 64 \to d$.

### 4.2 Quantitative Results

**nuScenes.** Table 1 compares STEM with temporal and non-temporal methods on nuScenes.

For mapping accuracy, STEM achieves the highest mAP (75.00) at $K = 3$. Gains on pedestrian crossings and road boundaries outweigh its lower AP on lane dividers, improving overall accuracy.

For temporal stability, STEM achieves the best Presence, Loc, Shape, and mAS scores at $K = 3$, indicating stronger consistency across frames. Together, the results indicate that correlation-guided memory integration improves reconstruction and temporal stability on nuScenes.

**Argoverse 2.** Table 2 reports results on the Argoverse 2 validation set. STEM achieves the best APs for all categories and raises mAP to 77.1. Gains on pedestrian crossings, lane dividers, and

| $\tau_{\text{neg}}$ | 0.0 | 0.2 | **0.5** | 0.8 | 1.0 |
|---|---|---|---|---|---|
| **mAP** | 74.77 | 74.81 | **75.00** | 74.93 | 74.14 |
| **Presence** | 92.35 | 94.39 | **94.81** | 94.02 | 92.55 |
| **Loc** | 82.81 | 83.81 | **86.95** | 86.07 | 86.00 |
| **Shape** | 94.41 | 98.08 | **98.33** | 98.51 | 97.30 |
| **mAS** | 83.12 | 84.93 | **88.41** | 87.16 | 82.73 |

Table 7: TCR threshold sensitivity on nuScenes ($K = 3$).

| $K$ | $\text{AP}_p \uparrow$ | $\text{AP}_d \uparrow$ | $\text{AP}_b \uparrow$ | mAP $\uparrow$ | mAS $\uparrow$ |
|---|---|---|---|---|---|
| 1 | 80.17 | 70.29 | 74.08 | 74.85 | 87.31 |
| 2 | 80.19 | 70.58 | 74.04 | 74.93 | 87.92 |
| 3 | **80.23** | **70.71** | **74.06** | **75.00** | **88.41** |
| 4 | 80.19 | 70.74 | 73.94 | 74.96 | 88.35 |
| 5 | 80.12 | 70.66 | 73.91 | 74.90 | 88.02 |
| 6 | 80.04 | 70.55 | 73.83 | 74.81 | 87.66 |
| 7 | 79.92 | 70.41 | 73.72 | 74.68 | 87.21 |

Table 8: Inference sensitivity to $K$ on nuScenes.

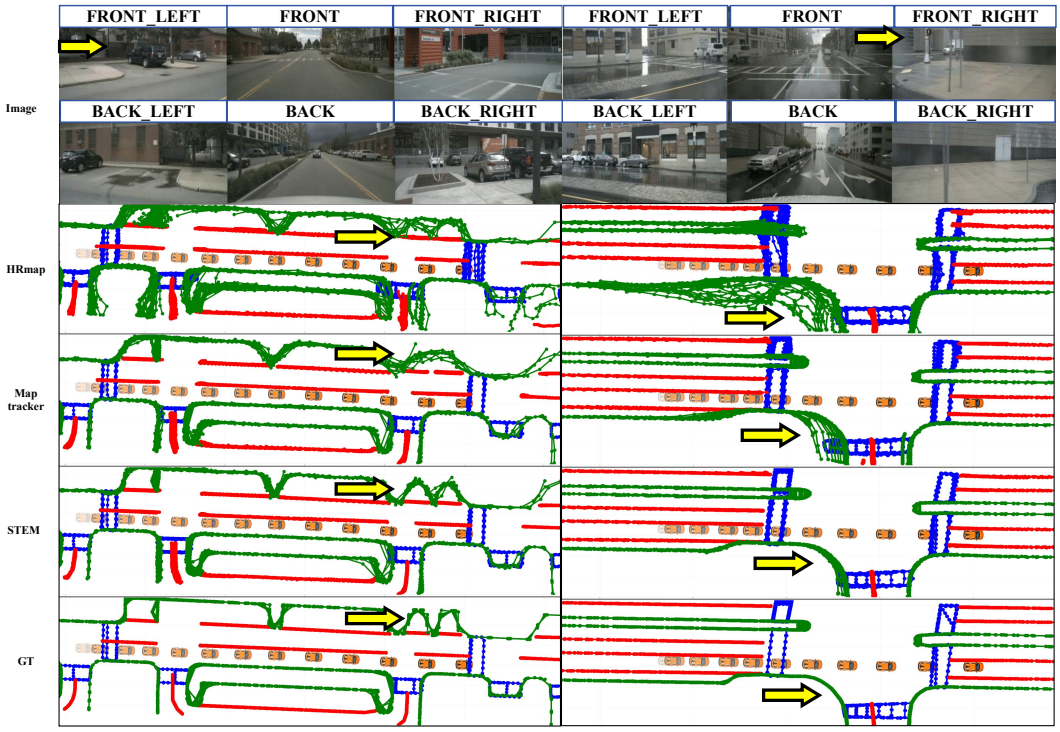


Figure 4: Qualitative comparison on two challenging scenarios. From top to bottom, the rows show HRMapNet, MapTracker, STEM, and GT. Each visualization aggregates 40 consecutive frames, and yellow arrows mark the principal differences.

road boundaries show that the memory formulation remains effective across all three categories of map elements. For temporal stability, STEM obtains the highest Presence, Loc, and mAS scores. Although MapTracker has a slightly higher Shape score, STEM raises overall mAS to 93.25 and provides a better balance.

**Ablation Study.** Table 3 shows the complementary roles of TCA and TCR on nuScenes. TCA alone raises mAP/mAS from $72.93/84.57$ to $73.52/86.16$, contributing $+0.59$ mAP and $+1.59$ mAS over MapTracker. This supports the benefit of selective memory utilization by the inference-time operator. Adding TCR reaches $75.00/88.41$, a further gain of $+1.48$ mAP and $+2.25$ mAS, consistent with training-time contrastive regularization of TCA's shared projected query-memory embeddings. TCR is removed at inference and introduces no additional aggregation pathway; the strongest result comes from their joint use. A controlled $2 \times 2$ ablation in Appendix A.2 independently varies retrieval and integration. With polarity-aware integration fixed, magnitude-based Top-$K$ improves over signed-score Top-$K$ by $0.43$ mAP and $0.80$ mAS; with magnitude-based retrieval fixed, polarity-aware integration improves over unified aggregation by $0.52$ mAP and $0.98$ mAS. These complementary gains support the proposed strength-polarity decomposition.

**TCA Factor Ablation.** Table 4 varies the three correlation factors with TCR enabled. Among single-factor variants, Static only performs best at $74.63$ mAP. Matching only and Dynamic only reach $74.46$ and $74.33$, respectively. Removing static, matching, or dynamic correlation reduces mAP by $0.45$, $0.20$, and $0.26$ points from $75.00$. Together, the three cues provide complementary correlation estimates.

**TCR Loss Ablation.** Table 5 keeps TCA fixed and removes each TCR term in turn. Removing $\mathcal{L}_{\text{pos}}$ reduces mAP/mAS from $75.00/88.41$ to $73.34/84.33$, corresponding to drops of $1.66$ mAP and $4.08$ mAS. This result suggests that promoting consistency among positively correlated same-track pairs benefits both geometric reconstruction and consistency across frames. Removing $\mathcal{L}_{\text{neg}}$ yields $74.14$ mAP and $82.73$ mAS. Its smaller $0.86$ mAP drop but larger $5.68$ mAS drop indicates the value of separating confusable cross-track representations during training. Both single-term variants also underperform the mAS of TCA alone, suggesting that the two auxiliary objectives work best jointly.

**Threshold Sensitivity.** Table 7 evaluates the hard-negative mining threshold $\tau_{\text{neg}}$ under the selected $K = 3$ configuration, with the hinge margin fixed at $\gamma = 0.2$. Thresholds of $0$ and $0.2$ admit less similar cross-track candidates and yield mAS scores of $83.12$ and $84.93$. Setting $\tau_{\text{neg}} = 0.5$ selects sufficiently similar hard negatives and achieves the best balance of $75.00$ mAP and $88.41$ mAS. Although $0.8$ slightly improves Shape, it lowers Presence, Loc, and overall mAS, which is consistent with reduced supervision coverage from a more restricted negative pool. At $\tau_{\text{neg}} = 1.0$, the negative term is disabled and reproduces the setting without $\mathcal{L}_{\text{neg}}$, with the same $74.14$ mAP and $82.73$ mAS. Because $\tau_{\text{neg}} = 0.5 > \gamma = 0.2$ in the adopted configuration, all selected negatives are already in the active region of the hinge; $\tau_{\text{neg}}$, rather than $\gamma$, determines which candidates enter the mined set. We therefore adopt $0.5$ as the default.

**Sparse Selection Robustness.** Table 8 varies the inference budget with memory capacity fixed at 20 and the checkpoint trained at $K = 3$. mAP peaks at $75.00$ for $K = 3$ and decreases from $74.96$ to $74.68$ as $K$ increases from 4 to 7, suggesting lower-ranked memories add no benefit.

**Inference Efficiency.** Table 6 compares STEM with MapTracker at $K = 3$ on one NVIDIA GeForce RTX 4090. STEM reaches 15.13 FPS and 66.11 ms, compared with MapTracker's 14.79 FPS and 67.62 ms. This small difference indicates comparable efficiency: TCA adds negligible overhead, while TCR is used only during training.

### 4.3 Qualitative Results

Figure 4 compares two representative temporal mapping challenges: incomplete structural observations and persistent interference from dynamic objects. Each visualization aggregates 40 consecutive frames, allowing the comparison to reflect how historical evidence is retained and integrated rather than the quality of an isolated prediction.

In the left scenario, the yellow arrows highlight two temporary parking areas that are only partially observed in individual frames. Recovering their complete geometry requires complementary evidence to be accumulated over time. HRMapNet and MapTracker produce incomplete or distorted structures around these areas, whereas STEM reconstructs more continuous road geometry and better preserves the surrounding map topology. This case is consistent with selective integration of correlated historical observations when evidence from the current frame is incomplete.

In the right scenario, a large building blocks part of the road corner, while heavy rain obscures the front-right camera. The baselines produce incomplete or distorted predictions; STEM reconstructs the corner more completely despite poor visibility.

## 5 Conclusion

STEM selects persistent memories by score magnitude and integrates them by polarity. TCA combines content, correspondence, and geometric-temporal cues at inference, while training-only TCR contrastively regularizes TCA's shared projected query-memory embeddings. On nuScenes and Argoverse 2, STEM improves mapping accuracy and temporal stability with comparable inference efficiency.

### AI USE STATEMENT

Generative AI supported language editing and LaTeX template conversion. The authors reviewed its output and take responsibility for the manuscript.

## REFERENCES

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 *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2020. doi: 10.1109/cvpr42600.2020.01164. URL http://dx.doi.org/10.1109/cvpr42600.2020.01164.

Jiacheng Chen, Yuefan Wu, Jiaqi Tan, Hang Ma, and Yasutaka Furukawa. Maptracker: Tracking with strided memory fusion for consistent vector hd mapping. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 90–107, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72658-3.

Wenjie Ding, Limeng Qiao, Xi Qiu, and Chi Zhang. Pivotnet: Vectorized pivot learning for end-to-end hd map construction. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3649–3659, 2023. doi: 10.1109/ICCV51070.2023.00340.

Zhiwei Dong, Ran Ding, Wei Li, Peng Zhang, Guobin Tang, and Jia Guo. Leveraging sd map to augment hd map-based trajectory prediction. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17219–17228, 2025. doi: 10.1109/CVPR52734.2025.01605.

Shuang Gao, Qiang Wang, and Yuxiang Sun. Boundary-aware semantic bird-eye-view map generation based on conditional diffusion models. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(10):10188–10198, 2025. doi: 10.1109/TCSVT.2025.3564002.

Xianjing Han, Wenmiao Hu, Xuemeng Song, Hannes Kruppa, See-Kiong Ng, and Roger Zimmermann. Traffic sign localization and orientation classification for automated map updating. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(10):9736–9747, 2025. doi: 10.1109/TCSVT.2025.3570488.

Xiaoshuai Hao, Hui Zhang, Yifan Yang, Yi Zhou, Sangil Jung, Seung-In Park, and ByungIn Yoo. Mbfusion: A new multi-modal bev feature fusion method for hd map construction. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15922–15928, 2024. doi: 10.1109/ICRA57147.2024.10609873.

Peijin Jia, Tuopu Wen, Ziang Luo, Mengmeng Yang, Kun Jiang, Ziyuan Liu, Xuewei Tang, Zhiquan Lei, Le Cui, Bo Zhang, Kehua Sheng, and Diange Yang. Diffmap: Enhancing map segmentation with map prior using diffusion model. *IEEE Robotics and Automation Letters*, 9(11):9836–9843, 2024. doi: 10.1109/LRA.2024.3455853.

Hyeonseo Kim, Juyeb Shin, Hyeonjun Jeong, Hiwon Shin, and Dongsuk Kum. MapTCL: Temporal consistency learning via bidirectional alignment for vectorized HD map construction, 2026. URL https://arxiv.org/abs/2608.05209. arXiv preprint.

Bo Lang, Nirav Savaliya, Zhihao Zheng, Jinglun Feng, Zheng-Hang Yeh, and Mooi Choo Chuah. PredMapNet: Future and historical reasoning for consistent online HD vectorized map construction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5477–5487, 2026. doi: 10.1109/WACV61042.2026.00531.

Maximilian Leitenstern, Florian Sauerbeck, Dominik Kulmer, and Johannes Betz. Flexmap fusion: Georeferencing and automated conflation of hd maps with openstreetmap. In *2024 IEEE Intelligent Vehicles Symposium (IV)*, pp. 272–278, 2024. doi: 10.1109/IV55156.2024.10588414.

Qi Li, Yue Wang, Yilun Wang, and Hang Zhao. Hdmapnet: An online hd map construction and evaluation framework. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 4628–4634, 2022. doi: 10.1109/ICRA46639.2022.9812383.

Ruikai Li, Xinrun Li, Mengwei Xie, Hao Shan, Shoumeng Qiu, Xinyuan Chang, Yizhe Fan, Feng Xiong, Han Jiang, Yilong Ren, Haiyang Yu, Mu Xu, Yang Long, Varun Ojha, and Zhiyong Cui. AMap: Distilling future priors for ahead-aware online hd map construction, 2025.

Siyu Li, Jiacheng Lin, Hao Shi, Jiaming Zhang, Song Wang, You Yao, Zhiyong Li, and Kailun Yang. Dtclmapper: Dual temporal consistent learning for vectorized hd map construction. *IEEE Transactions on Intelligent Transportation Systems*, 25(12):21672–21686, 2024. doi: 10.1109/TITS.2024.3450704.

Bencheng Liao, Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Chang Huang. Maptr: Structured modeling and learning for online vectorized hd map construction. In *International Conference on Learning Representations*, 2023.

Bencheng Liao, Shaoyu Chen, Yunchi Zhang, Bo Jiang, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. MapTRv2: An end-to-end framework for online vectorized hd map construction. *International Journal of Computer Vision (IJCV)*, pp. 1–23, 2024.

Xiaolu Liu, Song Wang, Wentong Li, Ruizi Yang, Junbo Chen, and Jianke Zhu. Mgmap: Mask-guided learning for online vectorized hd map construction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14812–14821, June 2024a.

Yicheng Liu, Tianyuan Yuan, Yue Wang, Yilun Wang, and Hang Zhao. VectorMapNet: End-to-end vectorized HD map learning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 22352–22369. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/liu23ax.html.

Zihao Liu, Xiaoyu Zhang, Guangwei Liu, Ji Zhao, and Ningyi Xu. Leveraging enhanced queries of point sets for vectorized map construction. In *European Conference on Computer Vision*, 2024b.

Yuexin Ma, Tai Wang, Xuyang Bai, Huitong Yang, Yuenan Hou, Yaming Wang, Yu Qiao, Ruigang Yang, and Xinge Zhu. Vision-centric bev perception: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10978–10997, 2024. doi: 10.1109/TPAMI.2024.3449912.

Limeng Qiao, Wenjie Ding, Xi Qiu, and Chi Zhang. End-to-end vectorized hd-map construction with piecewise bézier curve. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13218–13228, 2023.

Hao Shan, Ruikai Li, Han Jiang, Yizhe Fan, Ziyang Yan, Bohan Li, Xiaoshuai Hao, Hao Zhao, Zhiyong Cui, Yilong Ren, and Haiyang Yu. Stability under scrutiny: Benchmarking representation paradigms for online hd mapping. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=mxz5RqhCMe.

Tixiao Shan and Brendan Englot. Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4758–4765, 2018. doi: 10.1109/IROS.2018.8594299.

Tixiao Shan, Brendan Englot, Drew Meyers, Wei Wang, Carlo Ratti, and Daniela Rus. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5135–5142, 2020. doi: 10.1109/IROS45743.2020.9341176.

Juyeb Shin, Hyeonjun Jeong, Francois Rameau, and Dongsuk Kum. Instagram: Instance-level graph modeling for vectorized hd map learning. *IEEE Transactions on Intelligent Transportation Systems*, 26(2):1889–1899, 2025. doi: 10.1109/TITS.2024.3518537.

Jingyu Song, Xudong Chen, Liupei Lu, Jie Li, and Katherine A. Skinner. MemFusionMap: Working memory fusion for online vectorized HD map construction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. URL https://arxiv.org/abs/2409.18737.

Shuo Wang, Fan Jia, Yingfei Liu, Yucheng Zhao, Zehui Chen, Tiancai Wang, Chi Zhang, Xiangyu Zhang, and Feng Zhao. Stream query denoising for vectorized hd map construction. In *ECCV*, 2024.

Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021. URL https://openreview.net/forum?id=vKQGe36av4k.

Tianyuan Yuan, Yicheng Liu, Yue Wang, Yilun Wang, and Hang Zhao. Streammapnet: Streaming mapping network for vectorized online hd map construction. In *WACV*, pp. 7356–7365, 2024.

Xiaoyu Zhang, Guangwei Liu, Zihao Liu, Ningyi Xu, Yunhui Liu, and Ji Zhao. Enhancing vectorized map perception with historical rasterized maps. In *European Conference on Computer Vision*, 2024a.

Yiwei Zhang, Jin Gao, Fudong Ge, Guan Luo, Bing Li, Zhaoxiang Zhang, Haibin Ling, and Weiming Hu. VQ-Map: Bird's-eye-view map layout estimation in tokenized discrete space via vector quantization. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024b. Curran Associates Inc. ISBN 9798331314385.

Zhixin Zhang, Yiyuan Zhang, Xiaohan Ding, Fusheng Jin, and Xiangyu Yue. Online vectorized hd map construction using geometry. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 73–90, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72967-6.

# A Analysis of Polarity-Aware Temporal Aggregation

We analyze correlation polarity while holding the selected memory set $\mathcal{S}_i$ and its value features $v_{i,\tau}$ fixed. Let $c_{i,\tau}$ be the learned score, with $\mathcal{S}_i^+ = \{\tau \in \mathcal{S}_i \mid c_{i,\tau} \geq 0\}$ and $\mathcal{S}_i^- = \{\tau \in \mathcal{S}_i \mid c_{i,\tau} < 0\}$. Unified aggregation is

$$h_i^{\text{uni}} = \sum_{\tau \in \mathcal{S}_i} \alpha_{i,\tau} v_{i,\tau} \in \text{Conv}(\{v_{i,\tau}\}_{\tau \in \mathcal{S}_i}), \quad \alpha_{i,\tau} \geq 0, \quad \sum_{\tau \in \mathcal{S}_i} \alpha_{i,\tau} = 1. \tag{7}$$

For a simplified polarity-aware operator using the same values in both branches, define

$$\begin{aligned} h_i^+ &= \sum_{\tau \in \mathcal{S}_i^+} \alpha_{i,\tau}^+ v_{i,\tau}, \quad h_i^- = \sum_{\tau \in \mathcal{S}_i^-} \alpha_{i,\tau}^- v_{i,\tau}, \\ u_i^{\text{pol}} &= \lambda h_i^+ - (1-\lambda) h_i^-, \quad \lambda \in [0,1], \end{aligned} \tag{8}$$

with non-negative weights normalized within each non-empty branch. The scalar gate is for analysis only; TCA uses a query-conditioned element-wise gate. If a branch is empty, we retain the other.

Define $\mathcal{V}_i^+ = \{v_{i,\tau} \mid \tau \in \mathcal{S}_i^+\}$ and $\mathcal{V}_i^- = \{v_{i,\tau} \mid \tau \in \mathcal{S}_i^-\}$. When both branches are non-empty, $u_i^{\text{pol}}$ is a convex combination of $\mathcal{V}_i^+ \cup (-\mathcal{V}_i^-)$: its coefficients are non-negative and sum to $\lambda + (1-\lambda) = 1$.

**Proposition 1.** On fixed selected memory-value features, unified aggregation with non-negative normalized weights lies in the convex hull of the original values, whereas polarity-aware aggregation can produce signed updates outside it. This operator-level result does not imply a universal expressivity ordering between end-to-end architectures.

*Proof.* Let the positive- and negative-correlation values be:

$$v^+ = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad v^- = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{9}$$

For $0 \leq \alpha \leq 1$, the unified update and the polarity-aware update at $\lambda = \frac{1}{2}$ are

$$\begin{aligned} h^{\text{uni}} &= \alpha v^+ + (1-\alpha) v^- = \begin{bmatrix} \alpha \\ 1-\alpha \end{bmatrix}, \\ u^{\text{pol}} &= \tfrac{1}{2} v^+ - \tfrac{1}{2} v^- = \begin{bmatrix} \frac{1}{2} \\ -\frac{1}{2} \end{bmatrix}. \end{aligned} \tag{10}$$

Every point in the hull has a non-negative second coordinate, whereas $u^{\text{pol}}$ has a negative one; hence it lies outside. □

In actual TCA, the branch values are $v_{i,\tau}^+ = W^+ m_{i,\tau}$ and $v_{i,\tau}^- = W^- m_{i,\tau}$, with the query-conditioned gate $g_i = \sigma(W_g q_i^t)$. Let $G_i = \text{Diag}(g_i)$. The pre-MLP update is

$$u_i = G_i h_i^+ - (I - G_i) h_i^-, \tag{11}$$

or, for feature dimension $j$, $u_{i,j} = g_{i,j} h_{i,j}^+ - (1 - g_{i,j}) h_{i,j}^-$. TCA generalizes the scalar analysis through per-feature gating, branch-specific projections, and a residual MLP.

## A.1 Polarity Usage Analysis

The preceding analysis characterizes the operator-level effect of signed aggregation; we next examine how the two polarity branches are exercised by the trained model in practice. On the nuScenes validation set, TCA processes 277,038 valid queries and 985,804 valid query-memory candidate pairs, retaining 531,270 pairs through magnitude-based Top-$K$ retrieval. Among the selected pairs, 342,787 (64.52%) have nonnegative scores and 188,483 (35.48%) have negative scores, so the negative branch participates in more than one third of selected memory interactions rather than appearing only as a rare edge case. The category-wise negative proportions are 32.79% for pedestrian crossings, 35.97% for dividers, and 35.73% for boundaries, indicating that signed memory relations occur across all three semantic categories rather than being dominated by a particular class.

Table 9: Polarity usage of Top-$K$ selected memories on the nuScenes validation set. Positive and negative denote the proportions of selected query-memory pairs with $c_{i,\tau} \geq 0$ and $c_{i,\tau} < 0$, respectively.

| Category | # Queries | Positive (%) | Negative (%) |
|---|---|---|---|
| Ped. crossing | 35,226 | 67.21 | 32.79 |
| Divider | 134,526 | 64.03 | 35.97 |
| Boundary | 107,286 | 64.27 | 35.73 |
| Overall | 277,038 | 64.52 | 35.48 |

Table 10: Mean absolute correlation magnitude before and after Top-$K$ memory selection.

| Memory Population | Static | Matching | Dynamic |
|---|---|---|---|
| All candidate pairs | 0.1340 | 0.5646 | 0.6956 |
| Top-$K$ selected pairs | 0.1488 | 0.6472 | 0.7621 |

Table 10 compares the mean absolute values of the three correlation components across all candidate pairs and the Top-$K$ selected pairs. Selection increases the static, matching, and dynamic magnitudes from 0.1340 to 0.1488, from 0.5646 to 0.6472, and from 0.6956 to 0.7621, respectively. This pattern is consistent with the intended role of score magnitude in sparse retrieval: memories retained by Top-$K$ tend to exhibit stronger component-wise relation evidence than the full candidate population.

### A.2 Controlled Ablation of Retrieval and Polarity

To isolate the contributions of retrieval strength and polarity-aware integration, we conduct a $2 \times 2$ ablation on nuScenes. Each variant is trained separately with all other configurations held fixed. Signed Top-$K$ ranks memories by their raw signed scores $c_{i,\tau}$, whereas Abs Top-$K$ ranks them by $|c_{i,\tau}|$. Unified aggregation processes the selected memories without separating them by polarity; Polarity uses TCA's separate positive- and negative-score branches.

Table 11: Controlled ablation of Top-$K$ retrieval and polarity-aware integration on the nuScenes validation set. All variants are separately trained with the remaining configurations fixed.

| Variant | mAP ↑ | mAS ↑ |
|---|---|---|
| Signed Top-$K$ + Unified | 74.12 | 86.91 |
| Abs Top-$K$ + Unified | 74.48 | 87.43 |
| Signed Top-$K$ + Polarity | 74.57 | 87.61 |
| **Abs Top-$K$ + Polarity** | **75.00** | **88.41** |

The results support both design choices. With polarity-aware integration fixed, Abs Top-$K$ improves over Signed Top-$K$ by 0.43 mAP and 0.80 mAS; with unified integration, it improves by 0.36 mAP and 0.52 mAS. Conversely, polarity-aware integration improves over unified aggregation by 0.52 mAP and 0.98 mAS with Abs Top-$K$, and by 0.45 mAP and 0.70 mAS with Signed Top-$K$. Their combination yields the strongest result, supporting the use of score magnitude for retrieval and score polarity for integration in this nuScenes setting.

## B Additional Qualitative Results

Figures 5–6 show nine additional nuScenes validation scenes with diverse road layouts and driving conditions, including long straight segments, turns, lane changes, and irregular intersections. Across all panels, rows show HRMapNet, MapTracker, STEM, and ground truth from top to bottom. These examples complement the quantitative results by illustrating temporal map consistency across varied scenes. The supplementary material also includes videos from diverse driving scenarios, providing sequence-level visualizations of STEM's temporal mapping behavior.

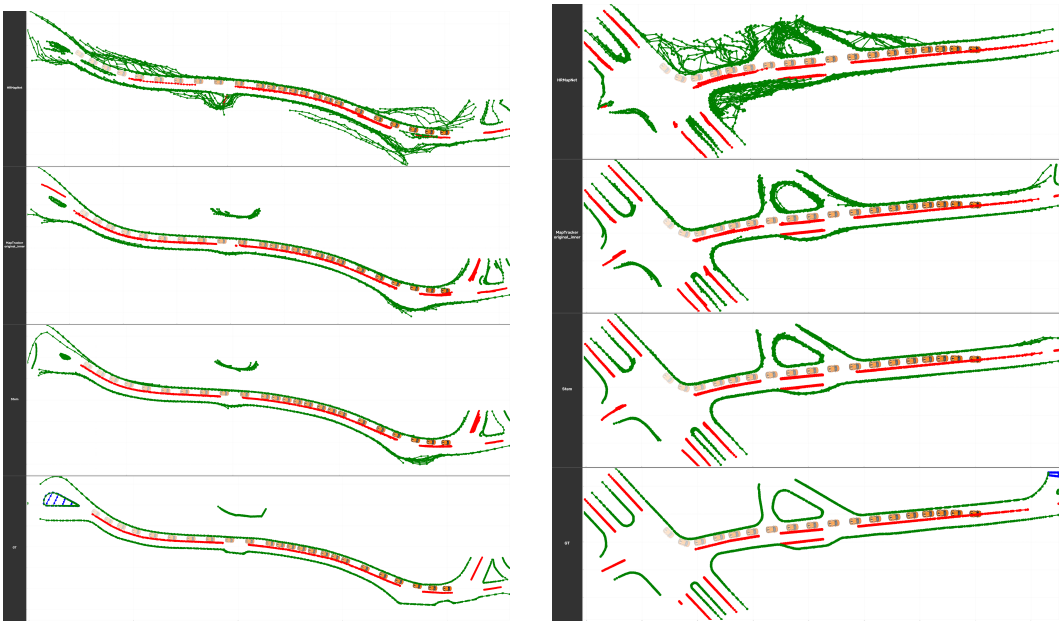

Figure 5: Additional qualitative results for nuScenes scenes 0012 and 0016. In each scene, rows show HRMapNet, MapTracker, STEM, and ground truth from top to bottom.

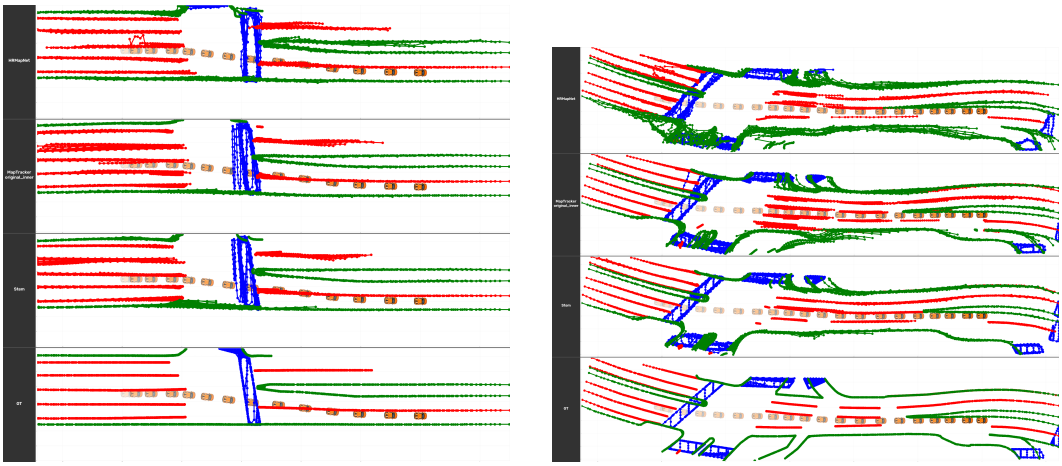

Figure 6: Additional qualitative results for nuScenes scenes 0101 and 0106. In each scene, rows show HRMapNet, MapTracker, STEM, and ground truth from top to bottom.

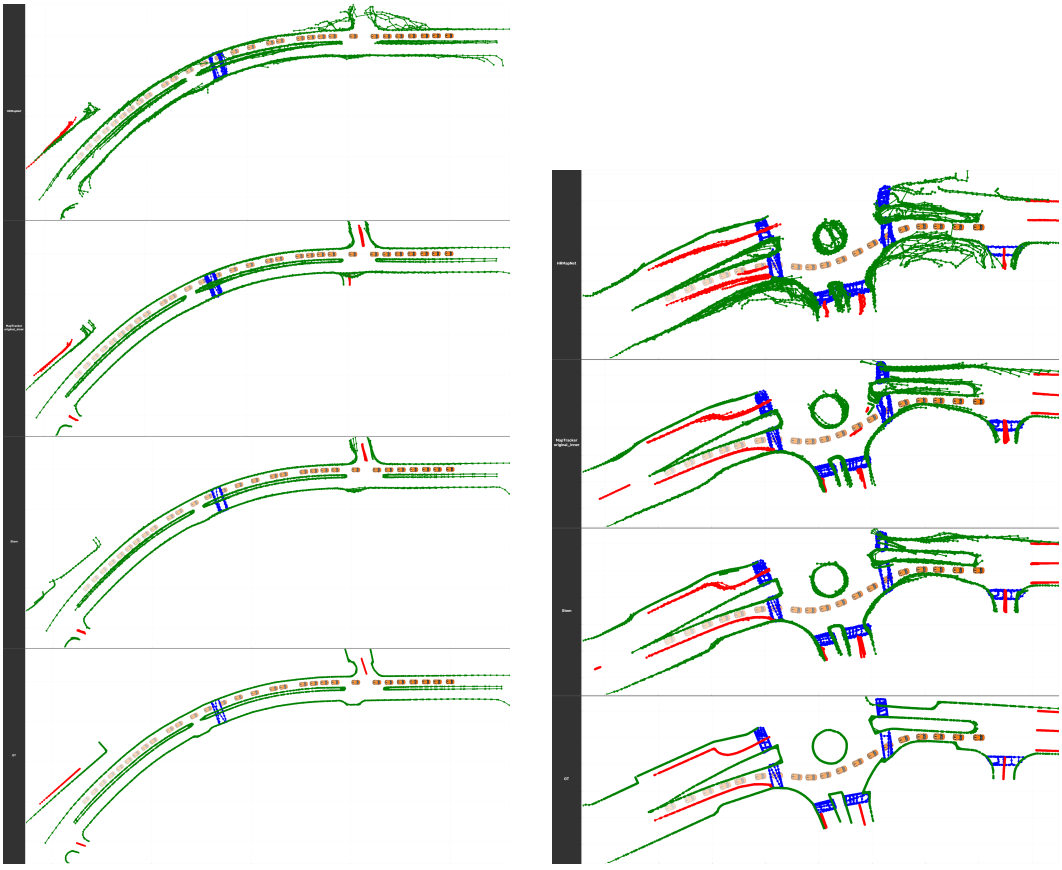

Figure 7: Additional qualitative results for nuScenes scenes 0035 and 0109. In each scene, rows show HRMapNet, MapTracker, STEM, and ground truth from top to bottom.

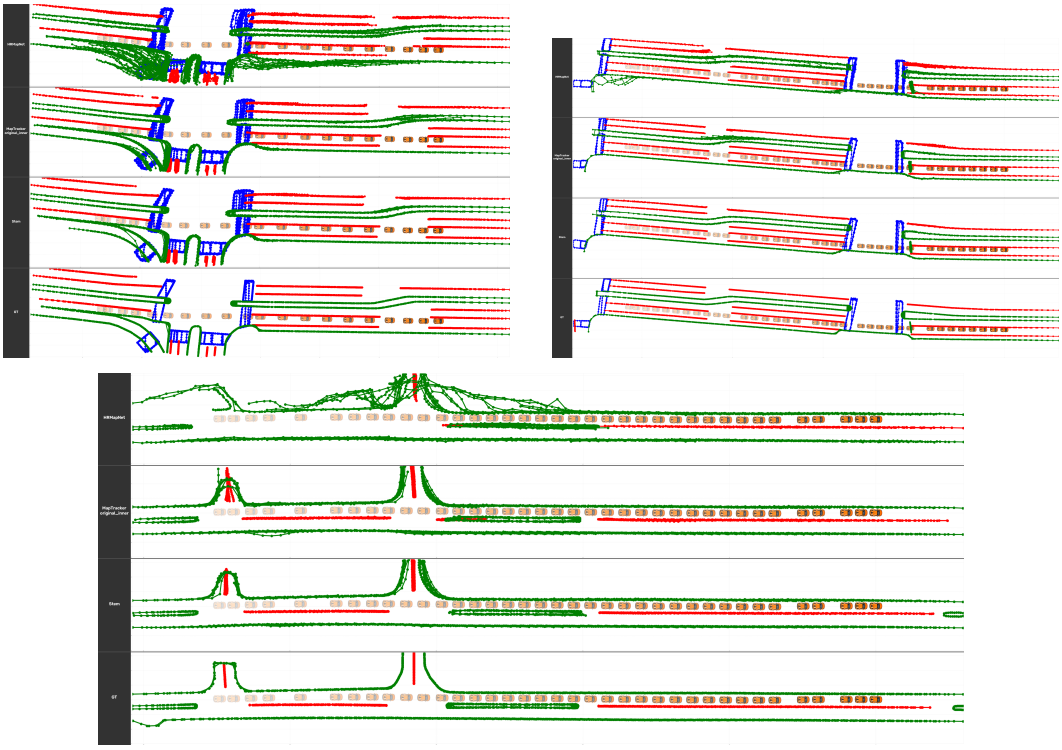

Figure 8: Additional qualitative results for nuScenes scenes 0096, 0098, and 1070. In each scene, rows show HRMapNet, MapTracker, STEM, and ground truth from top to bottom.