Semantics-Aware Scene Encoder for Interpretable Active Learning in E2E Autonomous Driving

Masaaki Inoue, Shintaro Fukushima

Toyota Motor Corporation
masaaki_inoue_aa@mail.toyota.co.jp

Abstract

End-to-end autonomous driving models require massive labeled data, yet not all scenes contribute equally to learning. Robust autonomous driving further demands broad spatio-temporal 3D understanding, including reconstruction and prediction, making it crucial to identify scenes that most enhance learning. We propose a self-supervised, annotation-free framework for selecting high-value scenes to efficiently improve driving models. Our method learns a unified scene-level latent space in which safety criticality, rarity, and uncertainty emerge geometrically through proximity to hazardous exemplars, density, and dispersion of projected planned states. Within this space, we define a value score that enables principled selection of informative and diverse training scenes, without additional labeling or closed-loop evaluation. Experiments on the nuScenes dataset demonstrate that the learned embedding space preserves safety-related topology even with limited supervision, where geometric proximity correlates positively with safety-criticality. Ablation studies show complementary effects among the metrics, improving accuracy and reducing collisions, and providing guidelines for future active learning design.

Introduction

End-to-end autonomous driving (E2E-AD) learns driving policies directly from raw sensor inputs, eliminating the need for manually composed modular pipelines trained in isolation. By unifying perception, prediction, and planning in a single differentiable model and enabling coherent spatio-temporal scene understanding, E2E-AD better coordinates intermediate representations and handles diverse scenarios. They match or surpass modular baselines (Hu et al. 2023), although challenges remain in eliminating cross-module error propagation (Yurtsever et al. 2020).

However, there still remain issues to be solved in E2E-AD. The principal bottleneck is the necessity of richly labeled time-series data. Achieving reliable, safety-critical performance typically requires massive volumes of human-annotated supervision (Caesar et al. 2020). Such labels include 3D bounding boxes or voxel-level semantic annotations, which is often referred to as semantic occupancy (e.g., pedestrian and vehicle classes, lane and road-marking types). Procuring these annotations demands a substantial amount of both human/computational resources and time. Even after the acquisition, training itself is still expensive, making it inefficient to indiscriminately include low-value samples that contribute little to model performance (Sener and Savarese 2018). Compounding the problem, driving data are long-tailed—routine scenes dominate, whereas rare but safety-critical events are scarce and naive annotation both wastes effort and still fails to cover corner cases (Liu and Feng 2024). These challenges motivate approaches that improve learning efficiency by prioritizing informative, safety-relevant samples and by reducing overall labeling cost.

A substantial body of research spanning E2E architectures and beyond seeks to address this issue. *Active Learning* addresses the inefficiency and bias in training data utilization by focusing on samples with high predictive uncertainty and potential utility for downstream planning, rather than expending effort on routine or well-understood cases (Ren et al. 2021). In E2E-AD, this means prioritizing scenarios for additional model training that require supplementary annotations such as 3D bounding boxes, which are indispensable for the model training. This prioritization ensures that a limited annotation budget is devoted to cases that most effectively reduce policy risk (Gal, Islam, and Ghahramani 2017). Although few in number, recent E2E-specific studies substantiate this claim. *ActiveAD* adopts a planning-centric objective and, with carefully selected data, recovers performance comparable to training on the full dataset (Rao et al. 2024). *SEAD* guides selection using bird's-eye-view (BEV) embeddings without additional supervision (Jiang et al. 2025). Complementary approaches such as semi/self-supervised learning, simulation-based augmentation (Chen et al. 2020a), and dataset distillation/coresets (Yu, Liu, and Wang 2024) reduce labeling needs but depend on strong priors or high-fidelity simulators, suffer from sim-to-real shift, and can still miss rare, safety-critical events (Kalra and Paddock 2016). Active learning is a task to directly allocate labels to valuable scenes and can be combined with these techniques. Beyond initial curation, it also improves a trained model. It uses the current E2E policy as a critic to identify failure modes and influential scenarios from unlabeled data, and then fine-tunes on this focused subset to bridge generalization gaps without increasing annotation cost.

Limitations of Existing Work Existing active learning methods aim at selecting useful data absent from the current training set. Such methods can be grouped into two categories, each with drawbacks:

- **Heuristic-based.** The policy to broaden diversity is specified by human knowledge (e.g., categories of driving commands or surrounding environment). Consequently, important features of training data that affect E2E-AD performance possibly lie outside this prior knowledge and may be overlooked.
- **Latent Representation-based.** Naive latent spaces directly derived from BEV intermediates are weakly interpretable for active learning policy, and selection tends to reward apparent novelty. This prevents a principled balance among data rarity, safety relevance, and predictive uncertainty.

Our approach. To address these gaps, we develop a scene-level encoder that remains interpretable while retaining the ability to capture out-of-prior, safety-relevant cues in E2E-AD. We learn a scene-level latent representation $\mathbf{z}_i \in \mathbb{R}^d$ from time-series planning error patterns and score unlabeled scenes geometrically. This unified, annotation-free criterion replaces ad-hoc heuristics and requires no dense manual labels beyond standard training sensor signals. The policy surfaces high-value samples (e.g., atypical traffic or safety-boundary cases), improving data efficiency, yielding interpretable rationales grounded in the learned embedding.

Our approach dispenses with additional manual annotation or simulator-in-the-loop scoring. Beyond empirical gains, the formulation offers a unified recipe to bridge representation learning and data valuation for autonomous driving, turning scene selection into a geometric optimization problem in a learned space.

Contributions

The contributions of this work are threefold:

1. **Active learning for E2E-AD with reduced designer bias.** We propose a label-free acquisition policy that is interpretable yet accounts for out of distribution scenes without hand crafted scenario knowledge. At its core, we train a scene-level encoder from both the current E2E-AD model's behavior and its learned training distribution, using supervised contrastive learning to separate safe vs. unsafe error patterns treated as time-series. This reduces designer bias while preserving auditability.
2. **Semantics-aware geometry for decision-making.** We introduce model-agnostic similarity metrics for scenes to train BEV path planning that jointly capture error-pattern concordance and safety margin. The encoder's latent space is geometrically interpretable along indices of safety criticality, scene rarity, and model uncertainty, with tunable value score Q_i for each scene i that provides a principled balance of these factors for selection and downstream decisions.
3. **Empirical validation on nuScenes.** On the *nuScenes* dataset, we evaluate our approach with *Vectorized Autonomous Driving (VAD)*, a widely studied E2E-AD model. We formalize the selection objective and instantiate an $\mathrm{embed} \rightarrow \mathrm{score} \rightarrow \mathrm{select}$ loop. Our methods exceed a simple uniform-random baseline in data selection efficiency compared with uncertainty-only and rarity-only selection baselines. The investigation with our tunable model suggests the potential to reveal a complementary structure underlying dataset-independent performance improvements.

Related Work

End-to-End Autonomous Driving

Recent progress in E2E-AD has been driven by the success of transformer-based BEV perception modules (Liu et al. 2023; Li et al. 2025), which enable scalable integration of multi-sensor inputs. BEV architectures have become dominant in modern E2E systems such as *UniAD* (Hu et al. 2023), *VAD* (Jiang et al. 2023), and *PARA-Drive* (Weng et al. 2024), which project multi-view camera and LiDAR features into a unified top-down coordinate frame aligned with the ego-vehicle. This shared spatial topology unifies perception, prediction, and planning, enabling coherent interaction reasoning and decision-making.

Despite these advances, E2E driving remains sensitive to the composition and coverage of training data (Philion, Kar, and Fidler 2020; Chen et al. 2024). In large-scale datasets such as *nuScenes* (Caesar et al. 2020) and *Argoverse 2* (Wilson et al. 2021), retraining on all collected scenes is infeasible, making the identification of informative scenes for model updates a key challenge for data-efficient autonomy.

Data Selection and Active Learning

Data selection and active learning have long aimed to improve sample efficiency in supervised learning (Killamsetty et al. 2021; Saha and Roy 2023). Classical active learning methods based on uncertainty sampling, diversity criteria (Gal, Islam, and Ghahramani 2017; Beluch et al. 2018; Sener and Savarese 2018; Ash et al. 2020), or influence estimation (Koh and Liang 2017; Pruthi et al. 2020) often require task-specific supervision, frequent validation, or repeated retraining, which is impractical for large-scale autonomous driving. Recent coreset selection approaches (Mirzasoleiman, Bilmes, and Leskovec 2020; Mindermann et al. 2022) approximate gradients or model sensitivities to identify informative samples but still depend on supervised signals or domain heuristics.

Research on active learning for E2E-AD has only recently emerged. *ActiveAD* (Rao et al. 2024) adopts hand-designed scenario policies (e.g., command and environment categories) to select scenes via planning degradation/recovery. However, these heuristics risk missing salient factors beyond these priors and are ill-suited to the continuously expanding datasets typical of real-world applications. Jiang et al. (Jiang et al. 2025) leverage BEV scene embeddings to capture non-explicit features, and report near full-dataset performance using only 30% of *nuScenes*. However, these pipelines face two key limitations in real-world use: (1) they rely on human-specified scenario heuristics (e.g., driving command categories, weather, or lighting), which do

not adapt well to distribution shifts in newly collected data; and (2) they perform naive selection in weakly interpretable BEV latent spaces, which tends to reward apparent novelty rather than a principled balance of driving-specific factors.

Latent Representations and Self-Supervised Learning

Representation learning has become a cornerstone for constructing generalizable embeddings that capture the intrinsic structure of complex data (Bengio, Courville, and Vincent 2013). Self-supervised contrastive frameworks such as SimCLR (Chen et al. 2020b), MoCo (He et al. 2020), and BYOL (Grill et al. 2020) enable feature learning without manual labels by training models to discriminate different views or augmentations of unlabeled data. In parallel, supervised contrastive learning (Khosla et al. 2020) leverages label information to impose class-aware structure in the embedding space, complementing self-supervised objectives when weak or task-specific signals are available.

In autonomous driving, scene-level representation learning has been explored for improving downstream tasks: map-conditioned trajectory planning (Xu et al. 2022; Dauner et al. 2023), risk assessment (Peng et al. 2024) and uncertainty estimation (Djuric et al. 2020), and motion prediction (Ngiam et al. 2022). However, these methods leverage learned embeddings primarily to benefit downstream predictions, not for evaluating data itself. Consequently, systematic investigation into representation learning strategies that acquire embeddings enabling effective scene retrieval for E2E-AD remains largely unexplored.

Notation and Problem Formulation

We aim to improve the performance of an E2E-AD model by selecting an optimal subset of scenes to add to the current training set. We provide the notations and formalize the problem setting as follows.

Active Learning for E2E-AD

Given an existing labeled training set $\mathcal{T}$ and a pool of additional unlabeled scenes $\mathcal{C}$, our goal is to select m scenes $A \subset \mathcal{C}$ to be labeled that maximize the expected model improvement:

$$\mathcal{A}^* = \arg \max_{\mathcal{A} \subset \mathcal{C},\, |\mathcal{A}|=m} \Delta\mathcal{L}_{\text{target}}(\mathcal{T}, A),$$

where $\Delta\mathcal{L}_{\text{target}}$ denotes the reduction in target loss (e.g., trajectory or control prediction loss) obtained when augmenting $\mathcal{T}$ with $\mathcal{A}$.

Data for Autonomous Driving

We consider a dataset of N labeled driving scenes, where the i-th scene is a multimodal time series of length T_i:

$$\mathbf{x}_i = \{(I_t, L_t, M_t, e_t)\}_{t=1}^{T_i},$$

where I_t denotes synchronized multi-view camera images, L_t denotes LiDAR or BEV voxel features, M_t denotes HD or SD map context (e.g., lane graphs and semantic layers), and e_t denotes the ego-vehicle state (e.g., ego trajectory, velocity, acceleration, steering angle). Each scene is paired with metadata $\mathbf{y}_i \in \mathcal{Y}$ obtained from the annotations, primarily comprising 3D bounding boxes and labels. These annotations are also used to compute surrogate safety indicators.

Let $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$ be the full dataset, decomposed into the current training scenes $\mathcal{T}$, a pool of additional scenes $\mathcal{C}$ used for selection, and evaluation scenes $\mathcal{V}$:

$$\mathcal{D} \;=\; \mathcal{T} \cup \mathcal{C} \cup \mathcal{V},$$

where $\mathcal{T} \cap \mathcal{C} = \emptyset$, $\mathcal{T} \cap \mathcal{V} = \emptyset$, and $\mathcal{C} \cap \mathcal{V} = \emptyset$. Providing annotations $\mathbf{y}_i$ for scenes in $\mathcal{C}$ incurs substantial 3D annotation cost. Unless otherwise stated, scenes in $\mathcal{C}$ are unlabeled (before labeling):

$$i \in \mathcal{C} \;\Rightarrow\; (\mathbf{x}_i, \mathbf{y}_i) = (\mathbf{x}_i, \varnothing),$$

while $\mathbf{y}_i$ are available for $i \in \mathcal{T} \cup \mathcal{V}$.

Training of Model

The E2E driving model f_θ integrates perception, motion prediction, and planning within a unified architecture. Let f_θ denote the E2E driving model. Given a scene history $\mathbf{x}_{i,1:t'} = \{(I_t, L_t, M_t, e_t)\}_{t=1}^{t'}$ up to time $t' \le t$ for scene i, the model f_θ predicts a sequence of future ego positions at planning reference time t:

$$f_\theta(\mathbf{x}_{i,0:t'}) = \{\hat{\mathbf{p}}_i^{(t)}(t+\Delta)\}_{\Delta>0}.$$

For each scene $(\mathbf{x}_i, \mathbf{y}_i)$ in the labeled training set $\mathcal{T}$, $\mathbf{x}_i$ denotes the multimodal inputs, while $\mathbf{y}_i$ denotes annotation-derived targets used only during training. When such annotations are present, we also derive scene-level safety indicators for the safety-augmented objective. At inference time, f_θ consumes inputs $\mathbf{x}$; neither $\mathbf{y}$ nor the safety indicators are available.

In typical open-loop training, the objective minimizes a supervised prediction loss:

$$\theta^* = \arg\min_\theta \frac{1}{|\mathcal{T}|} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{T}} \ell\big(f_\theta(\mathbf{x}_i),\, \mathbf{y}_i\big), \qquad (1)$$

where ℓ is a per-sample supervision loss (e.g., trajectory deviation, control MSE, detection/tracking losses). Once additional data $\mathcal{A}$ is obtained, we update the training set as $\mathcal{T} \leftarrow \mathcal{T} \cup \mathcal{A}$ and re-solve (1). Unless otherwise specified, performance for reporting and for computing $\Delta\mathcal{L}_{\text{target}}$ is evaluated on $\mathcal{V}$.

In general, various strategies exist for incorporating data increments, such as continual or incremental learning (Parisi et al. 2019; Van de Ven, Tuytelaars, and Tolias 2022). In our experiments, we retrained the model for a few epochs after each data addition to isolate the effect of data selection itself, without involving incremental optimization or additional parameter adaptation.

Method

Overview

Figure 1 provides a high-level view of our active learning pipeline. To select additional scenes that most effectively

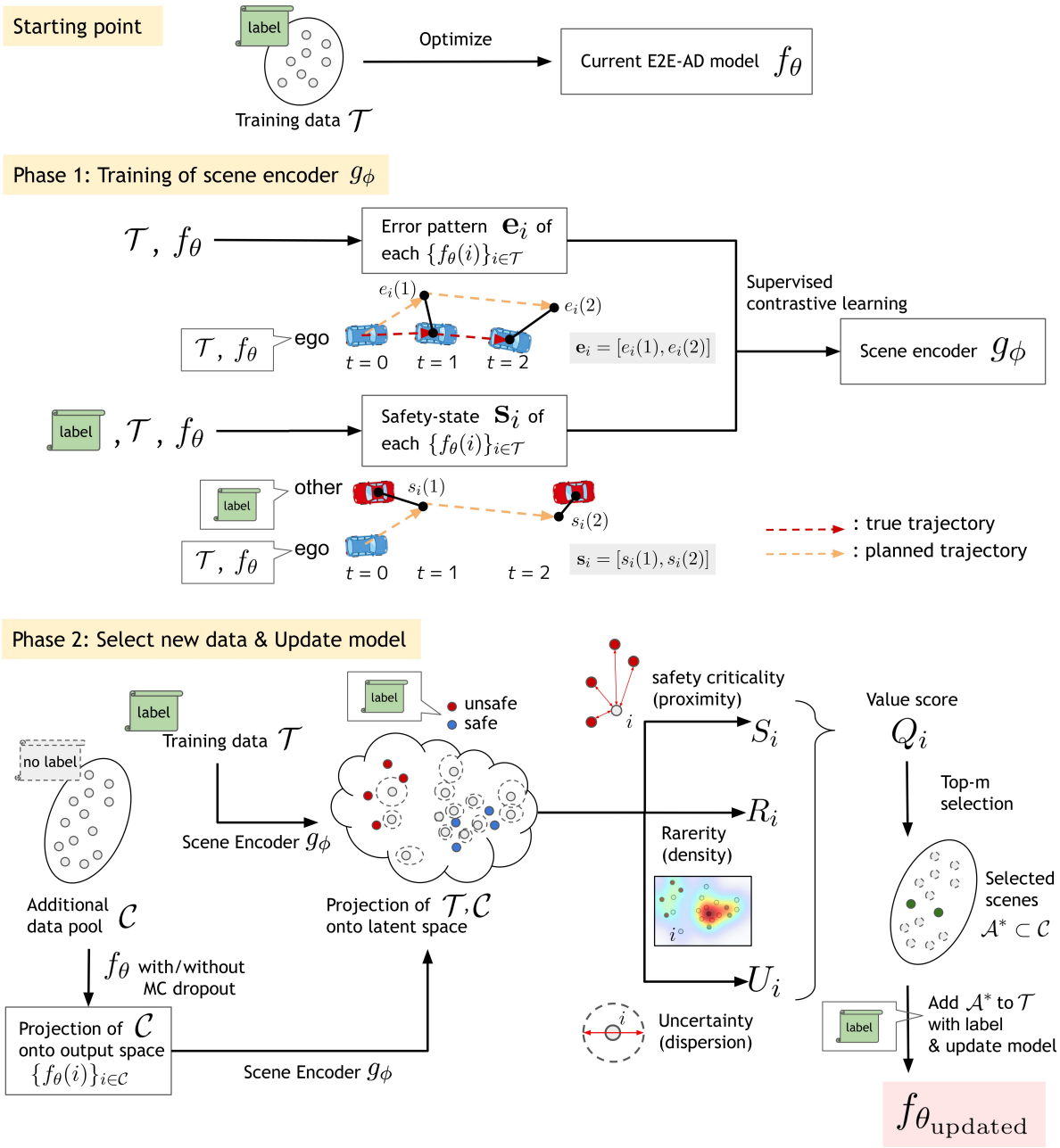

Figure 1: Overview of our active learning pipeline. Starting point: Training of an E2E-AD model f_θ on initial $\mathcal{T}$. Phase 1: From $\mathcal{T}$ and a frozen planner f_θ, a scene encoder g_ϕ is trained with supervised contrastive learning so that safety-critical scenes lie nearby in the latent space and scenes are organized by error-pattern similarity. Phase 2: A frozen scene encoder embeds each candidate scene into a latent space and computes three indices; safety-criticality S_i (e.g. average distance to nearby unsafe scenes), rarity R_i (local density), and epistemic uncertainty U_i (dispersion of predictions and its embeddings). These are combined into the value score $Q_i = \alpha_S S_i + \alpha_R R_i + \alpha_U U_i$ to rank scenes. A selector chooses the top-m scenes from the pool $\mathcal{C}$, augments the training set, and retrains the E2E-AD model.

improve the performance and safety of an E2E-AD model, we introduce a unified value score Q_i that quantifies the expected utility of including each labeled scene i in the training set. The score is designed to balance three complementary factors: safety-criticality, rarity, and uncertainty. The value score is denoted as:

$$Q_i = \alpha_S S_i + \alpha_R R_i + \alpha_U U_i,$$

where each α represents a non-negative weighting coefficient that controls the relative importance of the corresponding factor. Given the candidate pool $\mathcal{C}$, an ideal formulation would involve optimizing a subset of scenes that jointly maximize the overall value considering the preceding value score Q_i. The right part of Figure 1 also illustrates the rationales for $S_i, R_i,$ and U_i for intuitive understanding, represented geometrically in terms of proximity, density, and dispersion.

Scene Encoding via Error Patterns

(1) Scene-level Encoding Each scene i is encoded into a fixed-dimensional vector $\mathbf{z}_i \in \mathbb{R}^d$:

$$\mathbf{z}_i \;=\; g_\phi(\mathbf{e}_i),$$

where ϕ denotes the encoder parameters. Here, $\mathbf{e}_i$ is the representative error pattern for scene i, computed from a scene $\mathbf{x}_i$ and a trained model f_θ. We train g_ϕ so that distances between scenes reflect the similarity of their safety-critical error patterns between planning and ground-truth trajectories. Details of the encoding procedure are provided in Scene Encoder section.

(2) Error Pattern We define the representative error pattern $\mathbf{e}_i$ for each scene i. Let $\mathbf{p}_i(t+\Delta), \hat{\mathbf{p}}_i^{(t)}(t+\Delta) \in \mathbb{R}^2$ be the future ego BEV positions at timestep $t+\Delta$ for ground truth and planning, respectively, and define the deviation at planning reference time t:

$$\mathbf{e}_i^{(t)}(t+\Delta) = \hat{\mathbf{p}}_i^{(t)}(t+\Delta) - \mathbf{p}_i(t+\Delta).$$

To make each step consider a few steps ahead, fix a look-ahead horizon $H \geq 0$ and nonnegative look-ahead weights $\{\alpha_\Delta\}_{\Delta=1}^{H}$ (e.g., $\alpha_\Delta \equiv 1$ or $\alpha_\Delta \propto (1+\Delta)^{-1}$). For a step t, define the windowed deviation stack:

$$\tilde{\mathbf{e}}_i^{(t)} = \left[\mathbf{e}_i^{(t)}(t{+}1)^\top, \ldots, \mathbf{e}_i^{(t)}(t{+}H)^\top\right]^\top \in \mathbb{R}^{2H},$$

and score it by the weighted L_2 magnitude:

$$D_i(t) = \left(\sum_{\Delta=1}^{H} \alpha_\Delta \left\|\mathbf{e}_i^{(t)}(t{+}\Delta)\right\|_2^2\right)^{1/2}.$$

To ensure the window is well-defined, we restrict reference steps to $t \in \{1, \ldots, T{-}H\}$. We then select the reference step whose windowed deviation is maximal:

$$t_i^\star = \arg\max_{t \in \{1,\ldots,T-H\}} D_i(t). \tag{2}$$

Finally, we define the representative error pattern for scene i as the window around $t_i^\star$:

$$\mathbf{e}_i = \left[\alpha_1 \mathbf{e}_i(t_i^\star{+}1)^\top, \ldots, \alpha_H \mathbf{e}_i(t_i^\star{+}H)^\top\right]^\top \in \mathbb{R}^{2H}.$$

The encoder g_ϕ is trained so that scenes exhibiting similar representative error patterns $\mathbf{e}_i$ associated with safety-critical situations are mapped to nearby points in $\mathbb{R}^d$. We measure scene similarity in this embedding space with S_i and R_i, as explained in the following subsection.

Components of Value Score Q_i

(1) Safety-criticality Beyond performance metrics, E2E driving models must be reliable under safety-critical situations such as near-collisions or emergency braking. We therefore score each scene by its proximity in the latent space to individually labeled hazardous scenes. Let $\mathcal{H}$ be the index set of safety-critical scenes with embeddings $\{\mathbf{z}_h\}_{h\in\mathcal{H}}$. We define a nonparametric proximity score:

$$S_i = \max_{h\in\mathcal{H}} \exp\Big(-\frac{\|\mathbf{z}_i - \mathbf{z}_h\|_2^2}{2\tau_{\mathrm{s}}}\Big),$$

where $\tau_{\mathrm{s}} > 0$ controls sharpness. By construction, S_i increases monotonically as $\mathbf{z}_i$ approaches any hazardous embedding.

(2) Rarity Autonomous driving operates under a heavy-tailed data distribution where rare conditions are under-represented. To mitigate this bias, we estimate the local sample density in the latent feature space using K-nearest neighbors. For each embedding $\mathbf{z}_i$, we compute the average Gaussian-kernel similarity to its K_{nn} nearest neighbors:

$$p(\mathbf{z}_i) = \frac{1}{K_{\mathrm{nn}}} \sum_{j\in\mathrm{kNN}(i)} \exp\Big(-\frac{\|\mathbf{z}_i - \mathbf{z}_j\|_2^2}{2\sigma_{\mathrm{r}}^2}\Big),$$

$$R_i = -\log p(\mathbf{z}_i),$$

where σ_{r} denotes the kernel bandwidth. Higher R_i values correspond to low-density regions, encouraging diversity and distributional coverage. This ensures that selected scenes span both common and rare operational domains.

(3) Uncertainty We implement stochastic predictions via Monte Carlo (MC) dropout (Gal and Ghahramani 2016). At inference we keep dropout layers active and run K_{mc} forward passes. Let $\hat{\mathbf{p}}_{i,k}$ be the k-th trajectory prediction and $\mathbf{z}_{i,k}$ the k-th embedding obtained under different dropout masks: $\{\hat{\mathbf{p}}_{i,k}\}_{k=1}^{K_{\mathrm{mc}}}$ and $\{\mathbf{z}_{i,k}\}_{k=1}^{K_{\mathrm{mc}}}$ dropout enabled at test-time. Dispersion is computed from the mean and covariance of embeddings as follows:

$$\mathbf{z}_i = \tfrac{1}{K_{\mathrm{mc}}} \sum_{m=1}^{K_{\mathrm{mc}}} \mathbf{z}_{i,k},\; \boldsymbol{\Sigma}_i = \frac{1}{K_{\mathrm{mc}}-1} \sum_{k=1}^{K_{\mathrm{mc}}} (\mathbf{z}_{i,k} - \mathbf{z}_i)\,(\mathbf{z}_{i,k} - \mathbf{z}_i)^\top$$

$$U_i = \mathrm{tr}\big(\boldsymbol{\Sigma}_i\big).$$

We use the same dropout rate as training and set K_{mc} to balance cost and stability.

Scene Encoder

Our objective of encoding is twofold: (i) scenes with a similar amount of safety-criticality should be embedded nearby, (ii) within a given safety level, scenes should be further organized by their error-pattern class. To this end, we adopt supervised contrastive learning (Khosla et al. 2020) with positives defined primarily by safety-criticality and refined by error-pattern proximity.

Safety-critical Contrastive Learning with Error Patterns

(1) Setup We use L_2-normalized embeddings and cosine similarities:

$$\bar{\mathbf{z}}_i = \frac{\mathbf{z}_i}{\|\mathbf{z}_i\|_2}, \qquad s_{ik} = \langle \bar{\mathbf{z}}_i, \bar{\mathbf{z}}_k \rangle, \qquad \ell_{ik} = \frac{s_{ik}}{\tau_{\mathrm{c}}},$$

where $\tau_{\mathrm{c}} > 0$ is a temperature parameter. Recall that the original-scale embedding is $\mathbf{z}_i = g_\phi(\mathbf{e}_i)$, where $\mathbf{e}_i$ denotes the representative error pattern for the scene i.

(2) Safety Proximity Using the reference step $t_i^\star$ selected by Eq.(2), we assess safety based on the planning trajectory around this representative window of scene i. Recall $\hat{\mathbf{p}}_i(t)$ is the planned ego positions at t, and let $\{\mathcal{O}_o(t)\}$ denote the footprints of relevant objects/boundaries attained from annotations $\mathbf{y}_i$.

We define a minimum distance $d_i^{\min}(t)$ between the plan of ego $\hat{\mathbf{p}}_i^{(t_i^\star)}(t)$ and its nearest object $\mathcal{O}_i^{\mathrm{nearest}}(t)$, and take the minimum across objects to form a per-step soft collision cost:

$$c_i(t) = \exp\left(\tfrac{-d_i^{\min}(t)}{\sigma_{\mathrm{sc}}}\right),$$

where $\sigma_{\mathrm{sc}} > 0$ is a scale parameter. Fix a look-ahead horizon $H \geq 1$ and nonnegative weights $\{\beta_\Delta\}_{\Delta=1}^{H}$. We form the windowed safety-state descriptor around $t_i^\star$:

$$\mathbf{s}_i = \left[\beta_1\, c_i(t_i^\star{+}1),\ \ldots,\ \beta_H\, c_i(t_i^\star{+}H)\right]^\top \in \mathbb{R}^H. \qquad (3)$$

To construct the positive set based on safety proximity, we measure pairwise distances between the descriptors $\mathbf{s}_i$ defined above:

$$d(i,k) = \|\mathbf{s}_i - \mathbf{s}_k\|_2.$$

A scene k is treated as a positive for a scene i if $d(i,k) \leq \delta$ and $k \neq i$:

$$\mathcal{P}(i) = \{\, k \neq i \mid d(i,k) \leq \delta \,\}.$$

(3) Supervised Contrastive Loss We employ a supervised contrastive objective that attracts positives and repels non-positive examples. For each anchor i, let $\mathcal{P}(i)$ denote the set of positives. The loss averages the log-softmax over all anchor-positive pairs:

$$\mathcal{L}_{\mathrm{CL}} = \frac{1}{\sum_i |\mathcal{P}(i)|} \sum_i \sum_{p \in \mathcal{P}(i)} \left[-\log \frac{\exp(\ell_{ip})}{\sum_{k \neq i} \exp(\ell_{ik})} \right].$$

Scenes with $|\mathcal{P}(i)| = 0$ are skipped. In practice, we set δ adaptively per mini-batch using a percentile of $\{d(i,k)\}_{k \neq i}$ to stabilize the number of positives. We optimize the parameters ϕ of the encoder g_ϕ by minimizing $\mathcal{L}_{\mathrm{CL}}$ over the training set $\mathcal{T}$.

(4) Information Retrieval After the training of an encoder g_ϕ, an unseen scene j can be embedded as $\mathbf{z}_j = g_\phi(\mathbf{e}_j)$ without annotations $\mathbf{y}_j$. Then, the score value $Q_i = \alpha_S S_i + \alpha_R R_i + \alpha_U U_i$ is computed to rank additional scenes.

Experiments

Dataset and Setup

We performed experiments on the nuScenes dataset (Caesar et al. 2020), which provides diverse urban driving scenes with multimodal sensor data. For trajectory planning, we extracted ego-centric temporal sequences consisting of 5 future timesteps (0.5 s intervals) for each scenario. The overall framework follows the VAD pipeline, with additional modules for representation learning and uncertainty estimation. We conducted experiments on 1,000 scenes from the nuScenes dataset (Caesar et al. 2020), excluding 150 hidden test scenes without public labels. Following the official devkit split, 700 scenes were used for active learning and 150 for validation. Within the training set, 10% (70 scenes) were used as the initial labeled subset $\mathcal{T}$, while the remaining 630 formed the unlabeled pool $\mathcal{C}$. To mitigate initialization bias, all experiments were repeated 10 times with different random splits.

Model Architecture and Scoring

Encoder. We adopt a lightweight Transformer-based trajectory encoder to obtain the scene embedding $\mathbf{z}$. The encoder processes per-timestep 2D inputs and produces a 128-dimensional ℓ_2-normalized embedding for contrastive learning. The details of architecture are provided in Appendix A.1.

Safety, Rarity, and Uncertainty metrics. For active selection we score each scene by three complementary criteria. Safety-criticality S is defined by similarity to hazardous scenes, where hazards are the top 10% of training samples ranked by the maximum risk pattern over 5 future steps. Rarity R is estimated from k-NN density on pool embeddings (k=20), yielding higher scores in sparse regions. Uncertainty U is computed as the trace of covariance among MC-Dropout embeddings, reflecting epistemic uncertainty. Each metric is standardized by its own standard deviation before combination:

$$Q \;=\; \frac{S}{\sigma_S} \;+\; \frac{R}{\sigma_R} \;+\; \frac{U}{\sigma_U},$$

and scenes with the highest Q are selected for annotation and subsequent fine-tuning/retraining. Full hyperparameter settings, selection thresholds, and MC-Dropout configurations are provided in Appendix A.2.

Evaluation of embedding

We obtained the following results of active learning on the learned embedding space. The training/pool split is N_{train}=70 and N_{pool}=630 with embedding dimension d=128.

Consistency of Embedding and Safety. We constructed the proposed scene embeddings and verified that geometric proximity correlates with safety criticality. As a scalar safety criticality for each scene i, we take the maximum component of a safety state $\mathbf{s}_i$ in Eq. (3). Specifically, we computed Pearson's correlation between the ℓ_2 embedding distance and the difference in safety-criticality surrogate over

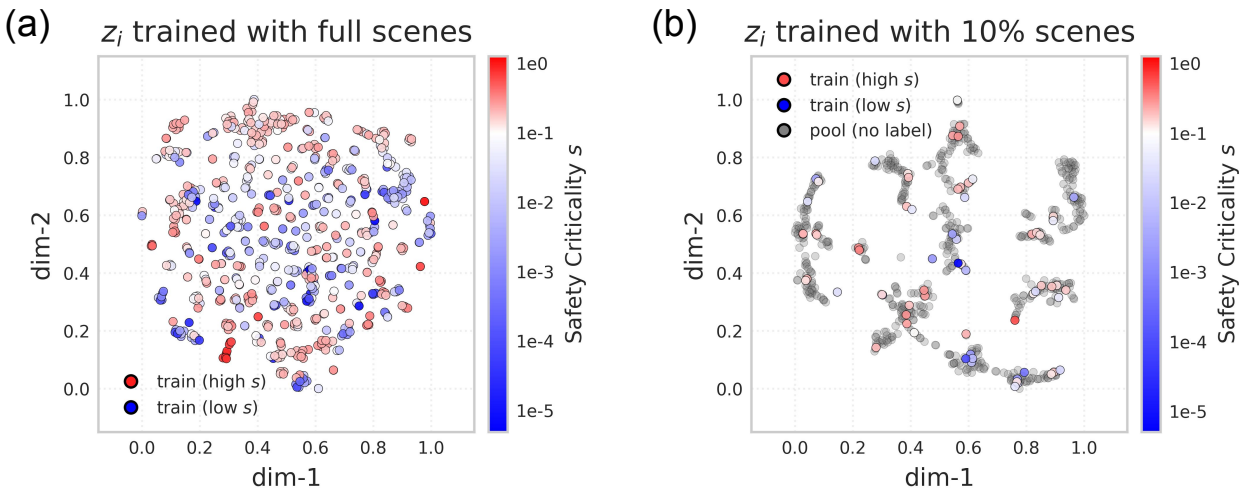

Figure 2: Relationship between scene embedding distance and safety criticality. For visualization, we project embeddings to 2D using t-SNE with perplexity 30, while all quantitative analyses use the original embedding space. The figures show embeddings of 700 scenes produced by encoders (a) trained on all 700 scenes and (b) trained on 70 scenes (≈10% of the full data). Positive Pearson correlations in the original latent space (d=128) indicate that the learned geometry preserves safety-related proximity.

each sample's 10 nearest neighbors in the embedding space. The correlation was positive in both panels of Fig. 2: (a) $\rho^{(a)} = 0.44$ and (b) $\rho^{(b)} = 0.36$. These results indicate that, even with only 10% of full data, the embedding preserves the neighborhood structure of safety-critical scenes, suggesting that new informative samples can be reliably discovered from the pool by nearest-neighbor search around hazardous exemplars.

Evaluation of Scoring. We conducted an ablation study to evaluate the contribution of each component of our framework. As a baseline, we employed a naive selection that samples scenes uniformly at random, corresponding to setting $Q_i = C$, where C is a constant score assigned to all scenes. For the proposed variants, we examined the effect of including or excluding S_i, R_i, and U_i in the score Q_i to evaluate their impact. All methods trained the VAD-tiny model for 10 epochs and then retrained it for two additional epochs, adding 10 scenes selected by each strategy. Performance was measured by the ℓ_2 displacement error and the collision rate. Table 1 summarizes the results, showing that while certain combinations exhibit consistently high contributions, the effectiveness of each component still varies across metrics. These findings suggest that distinct spatial characteristics within the embedding space contribute differently to various aspects of autonomous driving model performance. As combinations were controlled only by variance-based normalization, exploring their synergistic interactions remains an important direction for future work. Further investigation of the interactions among these three factors could provide valuable insights for designing active learning methods that incorporate some or all of these characteristics, and may also help explain occasional deviations from baseline performance observed in certain combinations of S, R, and U.

Table 1: Evaluation of components S, R, and U in Q

Method	L2 (m)			Collision (%)		
	1s	2s	3s	1s	2s	3s
baseline	0.55	0.91	1.30	0.37	0.59	0.93
S	0.53	0.89	1.31	**0.25**	0.51	0.86
R	0.57	0.96	1.41	0.37	0.55	0.85
U	0.56	0.93	1.37	0.30	0.52	0.87
S, R	0.53	0.88	1.31	0.33	0.49	**0.76**
S, U	0.53	0.87	1.29	0.33	0.51	0.83
R, U	**0.51**	**0.85**	**1.26**	0.29	**0.47**	0.79
S, R, U	0.53	0.89	1.31	0.29	0.51	0.88

Conclusion

We introduced a semantics-aware active learning framework for end-to-end autonomous driving that jointly leverages safety-criticality S_i, rarity R_i, and uncertainty U_i for a scene i defined in a learned embedding space. Our encoder maps representative error patterns to a designed latent space, enabling geometric scoring via proximity, density, and dispersion. Each component was designed to provide a complementary signal. The experimental results demonstrate that this complementarity indeed emerges across different evaluation metrics. By combining policy scores into a single acquisition value, our method selects informative scenes that improve downstream performance while preserving label efficiency.

Future directions include extending the notion of safeness beyond proximity to safety-critical scenes defined by object distance, integrating map and interaction priors into the embedding space, and investigating budget-aware acquisition strategies under realistic deployment constraints.

References

Ash, J.; Zhang, C.; Krishnamurthy, A.; Langford, J.; and Agarwal, A. 2020. Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds. In *International Conference on Learning Representations*.

Beluch, W. H.; Genewein, T.; Nöll, A.; and Körner, J. 2018. The Power of Ensembles for Active Learning in Image Classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9368–9377. IEEE.

Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8): 1798–1828.

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.-H. Y.; Baldan, G.; and Beijbom, O. 2020. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11621–11631. IEEE.

Chen, D.; Zhou, B.; Koltun, V.; and Krähenbühl, P. 2020a. Learning by Cheating. In *Proceedings of the Conference on Robot Learning*, volume 100, 66–75. Proceedings of Machine Learning Research.

Chen, L.; Wu, P.; Chitta, K.; Jaeger, B.; Geiger, A.; and Li, H. 2024. End-to-End Autonomous Driving: Challenges and Frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 10164–10183.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020b. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the International Conference on Machine Learning*, volume 119, 1597–1607. Proceedings of Machine Learning Research.

Dauner, D.; Hallgarten, M.; Geiger, A.; and Chitta, K. 2023. Parting with Misconceptions about Learning-based Vehicle Motion Planning. In *Proceedings of the Conference on Robot Learning*, volume 229, 1268–1281. Proceedings of Machine Learning Research.

Djuric, N.; Radosavljevic, V.; Cui, H.; Nguyen, T.; Chou, F.-C.; Lin, T.-H.; Singh, N.; and Schneider, J. 2020. Uncertainty-aware Short-term Motion Prediction of Traffic Actors for Autonomous Driving. In *IEEE Winter Conference on Applications of Computer Vision*, 2084–2093. IEEE.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proceedings of the International Conference on Machine Learning*, volume 48, 1050–1059. Proceedings of Machine Learning Research.

Gal, Y.; Islam, R.; and Ghahramani, Z. 2017. Deep Bayesian active learning with image data. In *Proceedings of the International Conference on Machine Learning*, 1183–1192. Journal of Machine Learning Research.

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P. H.; Buchatskaya, E.; Doersch, C.; Pires, B. A.; Guo, Z. D.; Azar, M. G.; Piot, B.; Kavukcuoglu, K.; Munos, R.; and Valko, M. 2020. Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning. In *Advances in Neural Information Processing Systems*, volume 33, 21271–21284.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9729–9738. IEEE.

Hu, Y.; Yang, J.; Chen, L.; Li, K.; Sima, C.; Zhu, X.; Chai, S.; Du, S.; Lin, T.; Wang, W.; et al. 2023. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17853–17862. IEEE.

Jiang, B.; Chen, S.; Xu, Q.; Liao, B.; Chen, J.; Zhou, H.; Zhang, Q.; Liu, W.; Huang, C.; and Wang, X. 2023. VAD: Vectorized Scene Representation for Efficient Autonomous Driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8340–8350. IEEE.

Jiang, W.; Li, D.; Hu, M.; Ma, C.; Wang, K.; and Zhang, Z. 2025. Active Learning from Scene Embeddings for End-to-End Autonomous Driving. *arXiv preprint arXiv:2503.11062*.

Kalra, N.; and Paddock, S. M. 2016. Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? *Transportation Research Part A: Policy and Practice*, 94: 182–193.

Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised Contrastive Learning. In *Advances in Neural Information Processing Systems*, volume 33, 18661–18673.

Killamsetty, K.; Durga, S.; Ramakrishnan, G.; De, A.; and Iyer, R. 2021. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *Proceedings of the International Conference on Machine Learning*, 5464–5474. Journal of Machine Learning Research.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*, 1885–1894. Journal of Machine Learning Research.

Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Yu, Q.; and Dai, J. 2025. BEVFormer: Learning Bird's-Eye-View Representation From LiDAR-Camera via Spatiotemporal Transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3): 2020–2036.

Liu, H. X.; and Feng, S. 2024. Curse of rarity for autonomous vehicles. *Nature Communications*, 15(1): 4808.

Liu, Z.; Tang, H.; Amini, A.; Yang, X.; Mao, H.; Rus, D.; and Han, S. 2023. BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation. In *IEEE International Conference on Robotics and Automation*, 2774–2781. IEEE.

Mindermann, S.; Whitney, W.; Gal, Y.; and Kurth-Nelson, Z. 2022. Prioritized Training on Points that are Learnable, Worth Learning, and Not Yet Learnt. In *Proceedings of the International Conference on Machine Learning*, 15630–15649. Journal of Machine Learning Research.

Mirzasoleiman, B.; Bilmes, J.; and Leskovec, J. 2020. Coresets for Data-efficient Training of Machine Learning Models. In *International Conference on Machine Learning*, 6952–6962. Journal of Machine Learning Research.

Ngiam, J.; Vasudevan, V.; Caine, B.; Zhang, Z.; Chiang, H.-T. L.; Ling, J.; Roelofs, R.; Bewley, A.; Liu, C.; Venugopal, A.; Weiss, D. J.; Sapp, B.; Chen, Z.; and Shlens, J. 2022. Scene Transformer: A unified architecture for predicting future trajectories of multiple agents. In *International Conference on Learning Representations*.

Parisi, G. I.; Kemker, R.; Part, J. L.; Kanan, C.; and Wermter, S. 2019. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113: 54–71.

Peng, L.; Li, B.; Yu, W.; Yang, K.; Shao, W.; and Wang, H. 2024. SOTIF Entropy: Online SOTIF Risk Quantification and Mitigation for Autonomous Driving. *IEEE Transactions on Intelligent Transportation Systems*, 25(2): 1530–1546.

Philion, J.; Kar, A.; and Fidler, S. 2020. Learning to Evaluate Perception Models Using Planner-Centric Metrics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE.

Pruthi, G.; Liu, F.; Kale, S.; and Sundararajan, M. 2020. Estimating Training Data Influence by Tracing Gradient Descent. In *Advances in Neural Information Processing Systems*, 19920–19930.

Rao, Z.; Song, Z.; Zhu, J.; Qiao, L.; Zhu, Y.; Chen, Y.; et al. 2024. ActiveAD: Autonomous Driving Data Selection and Evaluation via Closed-loop Simulation. *arXiv preprint arXiv:2403.02877*.

Ren, P.; Xiao, Y.; Chang, X.; Huang, P.-Y.; Li, Z.; Gupta, B. B.; Chen, X.; and Wang, X. 2021. A Survey of Deep Active Learning. *ACM Computing Surveys*, 54(9).

Saha, G.; and Roy, K. 2023. Continual learning with scaled gradient projection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37. AAAI Press.

Sener, O.; and Savarese, S. 2018. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *International Conference on Learning Representations*.

Van de Ven, G. M.; Tuytelaars, T.; and Tolias, A. S. 2022. Three types of incremental learning. *Nature Machine Intelligence*, 4(12): 1185–1197.

Weng, X.; Ivanovic, B.; Wang, Y.; Wang, Y.; and Pavone, M. 2024. PARA-Drive: Parallelized Architecture for Real-time Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15449–15458. IEEE.

Wilson, B.; Qi, W.; Agarwal, T.; Lambert, J.; Singh, J.; Khandelwal, S.; Pan, B.; Kumar, R.; Hartnett, A.; Pontes, J. K.; Ramanan, D.; Carr, P.; and Hays, J. 2021. Argoverse 2: Next Generation Datasets for Self-driving Perception and Forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*.

Xu, C.; Li, T.; Tang, C.; Sun, L.; Keutzer, K.; Tomizuka, M.; Fathi, A.; and Zhan, W. 2022. PreTraM: Self-supervised Pre-training via Connecting Trajectory and Map. In *Computer Vision – ECCV 2022*, 34–50. Springer.

Yu, R.; Liu, S.; and Wang, X. 2024. Dataset Distillation: A Comprehensive Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1): 150–170.

Yurtsever, E.; Lambert, J.; Carballo, A.; and Takeda, K. 2020. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access*, 8: 58443–58469.

Appendix

A.1 Encoder Architecture

The encoder g_ϕ used in the experiments is a compact Transformer network that maps a sequence of 2D coordinates into a latent representation. Each timestep input is first processed by two fully connected layers with ReLU activations to obtain a hidden feature of dimension 128, to which learnable positional embeddings are added to encode temporal order. The resulting sequence is fed into a Transformer encoder consisting of two layers of multi-head self-attention with four heads. The hidden state corresponding to the final timestep is used as the trajectory-level representation, which is then passed through a projection head to produce a 128-dimensional embedding $\mathbf{z}$ that is ℓ_2-normalized on the unit hypersphere for contrastive learning. The encoder is trained for 3000 epochs to ensure convergence and stable latent representations.

A.2 Hyperparameters and defaults

Encoder g_ϕ: The embedding dimension is d=128 with ℓ_2 normalization; the contrastive temperature is fixed to τ_{c}=0.07. Vector-weight parameters for $\mathbf{e}_i$ and $\mathbf{s}_i$ are set uniformly, while threshold parameters are chosen adaptively. We use a look-ahead horizon H=5 with uniform per-step weights α_Δ=1 and β_Δ=1. The safety proximity threshold δ is set per mini-batch to the 10th percentile of pairwise safety-descriptor distances, targeting $\approx 10\%$ positive rate. The soft-collision scale is set to $\sigma_{\mathrm{sc}} = 1$.

Components of Score Q_i: For Safety-criticality, the sharpness is set to $\tau_{\mathrm{s}} = 0.5$. The set of safety-critical scenes $\mathcal{H}$ is determined by ranking all trained scenes according to the maximum component of their safety state $\mathbf{s}_i$ and selecting the top 10% scenes. For rarity, we use K_{nn}=20 nearest neighbors with a Gaussian kernel, where the bandwidth σ_{r} is automatically set based on the median distance to the K_{nn}-th nearest neighbor across all samples. For uncertainty, inference runs in evaluation mode while reactivating only stochastic submodules (Dropout). Each unseen scene i is evaluated with K_{mc}=10 stochastic forward passes using distinct random seeds, yielding embeddings $\{\mathbf{z}_{i,k}\}_{k=1}^{10}$.