# Learning Sidewalk Autopilot from Multi-Scale Imitation with Corrective Behavior Expansion

Honglin He[1], Yukai Ma[1], Brad Squicciarini[2], Wayne Wu[1], Bolei Zhou[1]

https://vail-ucla.github.io/MIMIC

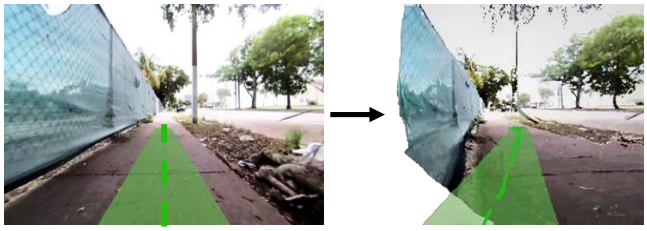
***Abstract*— Sidewalk micromobility is a promising solution for last-mile transportation, but current learning-based control methods struggle in complex urban environments. Imitation learning (IL) learns policies from human demonstrations, yet its reliance on fixed offline data often leads to compounding errors, limited robustness, and poor generalization. To address these challenges, we propose a framework that advances IL through corrective behavior expansion and multi-scale imitation learning. On the data side, we augment teleoperation datasets with diverse corrective behaviors and sensor augmentations to enable the policy to learn to recover from its own mistakes. On the model side, we introduce a multi-scale IL architecture that captures both short-horizon interactive behaviors and long-horizon goal-directed intentions via horizon-based trajectory clustering and hierarchical supervision. Real-world experiments show that our approach significantly improves robustness and generalization in diverse sidewalk scenarios. Demo video and additional information are available on the project page.**

## I. INTRODUCTION

Sidewalk micromobility has gained increasing attention as a solution for last-mile transportation in urban environments. Many applications have emerged in recent years, from robotic food delivery [1] to assistive power wheelchair [2, 3]. Figure 1 shows a food delivery robot navigating a crowded sidewalk with pedestrians, street vendors, and other obstacles. With the rapid development of learning-based approaches, control and decision-making in these robot systems have moved beyond purely rule-based methods and increasingly relied on data-driven paradigms. A promising approach to sidewalk navigation is imitation learning (IL) [4], which learns an end-to-end control policy directly from real-world human demonstrations. However, IL faces obvious limitations. Most notably, IL relies solely on learning from fixed and offline expert demonstrations, thus it often fails under closed-loop deployment where small errors are compounded over time and eventually lead to failure [5]. Meanwhile, collecting demonstration data for deviated scenarios and critical corner cases is particularly difficult, further limiting the policy's robustness and generalizability. Beyond these, a practical challenge in sidewalk scenarios is that input observations are egocentric RGB videos, from which all information, including scene geometry and object semantics, must be inferred. It increases the difficulty of training a generalist sidewalk autopilot. In summary, policies trained purely with IL often work poorly in complex sidewalk environments.

[1]Department of Computer Science, University of California, Los Angeles
[2]Coco Robotics

Sidewalk micromobility



Corrective behavior expansion

Multi-scale prediction

Fig. 1: This work aims to utilize corrective behavior expansion and multi-scale prediction to learn an autopilot model for sidewalk micromobility.

Prior work has focused on scaling data volume [6–10] to address these challenges. However, much of the existing data has been collected in relatively simple or structured environments [6, 7], which lack the complexity and diversity of real-world sidewalk scenarios. Meanwhile, these approaches are costly and still struggle to capture long-tail cases in specific domains, limiting generalization and robustness in real-world deployments. Other works utilize reinforcement learning (RL) [11] to go beyond demonstrations. However, RL requires costly reward engineering and high-fidelity simulators, and often produces non-human-like behaviors. An alternative path has emerged from recent work [12, 13], where the data seen by the policy during IL can be extended by augmenting the training data distribution. This motivates our work: can we push IL further by generating diverse and

plausible behaviors from a fixed offline dataset and fully exploiting each demonstration trajectory?

In this work, we study new ways to fully utilize real-world teleoperation data from both the *data side* and the *model side*, using data expansion with corrective behavior and multi-scale imitation learning. On the data side, we design a more effective way of augmenting teleoperation videos and demonstrations with corrective behaviors. Specifically, we synthesize novel data either from the observation side or by perturbing the action–observation–action loop, thereby exposing the policy to a broader distribution of plausible and diverse correction scenarios. Thus, the policy being trained can learn to recover from drifting off course. As illustrated in Fig. 1, our approach generates novel trajectories while preserving the underlying physical constraints in the original scenario. On the model side, we propose a multi-scale imitation learning and prediction framework to improve the policy's capacity to generalize across temporally and semantically diverse driving patterns. This framework first clusters trajectories based on temporal horizons and behavior patterns and then applies layer-wise supervision at different horizon levels, enabling the policy to learn both low-level interactions and high-level intentions in a unified framework. We summarize our contributions as:

- We propose a corrective behavior data expansion pipeline that synthesizes novel training data from existing teleoperation datasets by perturbing the action–observation–action loop, effectively increasing the coverage and diversity of training data.
- We propose a novel model architecture designed for tasks that require both short-horizon interactive behaviors and long-horizon goal-directed intentions.
- We establish real-world deployment and validation, demonstrating that our approach improves policy robustness and generalization in diverse, complex sidewalk environments using only offline teleoperation data.

## II. Related Work

**Sidewalk navigation.** Visual navigation has a long history. Early works focused on leveraging constructed 3D maps for localization and planning [14, 15]. In contrast, recent advances increasingly favor end-to-end learning models that map raw sensory observations directly to actions [6–10], known as mapless navigation. While these approaches span a wide range of navigation tasks, sidewalk navigation presents unique challenges, including narrow passages, frequent dynamic interactions with diverse pedestrians and other moving objects such as scooters and bikers, complex structures such as curbs and crosswalks, and complex urban layouts. Given these challenges, traditional map-based approaches, which rely on offline map construction, are often brittle in such environments. In this work, we focus on data-driven urban navigation foundation models that generalize across diverse sidewalk scenarios under varying environmental conditions. Prior work has collected large-scale data from real-world settings for policy learning [8, 9]. However, most of these approaches and datasets are limited to either indoor environments, outdoor but sparsely populated scenarios, or driving scenarios. While some prior studies have claimed that point-goal navigation is largely solved [16], the inherent complexity of real-world sidewalk navigation in a mapless, monocular RGB-camera setting remains a significant challenge that this work aims to address.

**Learning from teleoperation data.** Teleoperation provides a practical way to collect large-scale demonstrations for policy learning across diverse tasks and embodiments [8, 9]. Early efforts focused on modular learning, *i.e.* training different models for each sub-task like learning object detectors [17], planners [18] and controllers [19] separately. Recently, increasing attention has been paid to end-to-end approaches [8, 9]. These end-to-end approaches eliminate the need for handcrafted modules and offer the potential to capture complex correlations within the data. These offline end-to-end learning approaches require large volumes of data for training. However, in some real-world scenarios, data cannot be effectively collected or fully exploited due to limitations like coverage or annotation quality. At the same time, prior work has shown that imitation-only policies degrade rapidly when facing covariate shift or compounding errors [12, 20]. These challenges have led researchers to explore alternative strategies. In particular, many approaches have been developed to learn from a mixture of offline demonstrations and online interactions, combining the strengths of imitation learning and reinforcement learning to improve policy robustness and adaptability, including DAgger [20], residual reinforcement learning [21], and RLHF [22, 23]. Our work focuses on end-to-end learning without relying on reinforcement learning. Instead, we synthesize training data containing deviation-recovery trajectories, enabling the model to learn a robust policy that mitigates compounding errors commonly encountered in imitation learning. We also introduce a novel architecture tailored for tasks that require both short-horizon interactive behaviors and long-horizon goal-directed intentions, and demonstrate its effectiveness in learning from synthesized deviation-recovery trajectories.

## III. Method

In this section, we introduce the proposed learning framework **MIMIC** (**M**ulti-scale **IMI**tation with **C**orrective expansions), which leverages pretrained models to generate out-of-domain scenarios by training on both expert demonstrations and near-failure experiences, using multi-scale imitation.

### A. Problem Formulation

We aim to train a policy for mapless point-goal visual navigation, in which the agent receives only egocentric RGB images and GPS signals as input, both readily available on real-world robots. This setting eliminates the need for pre-built maps or localization modules and can be viewed as a sequential decision-making problem under partial observability. At each timestep $t$, the agent is provided with a history of the past $T_h$ RGB observations $i_{t-T_h:t}$, its past

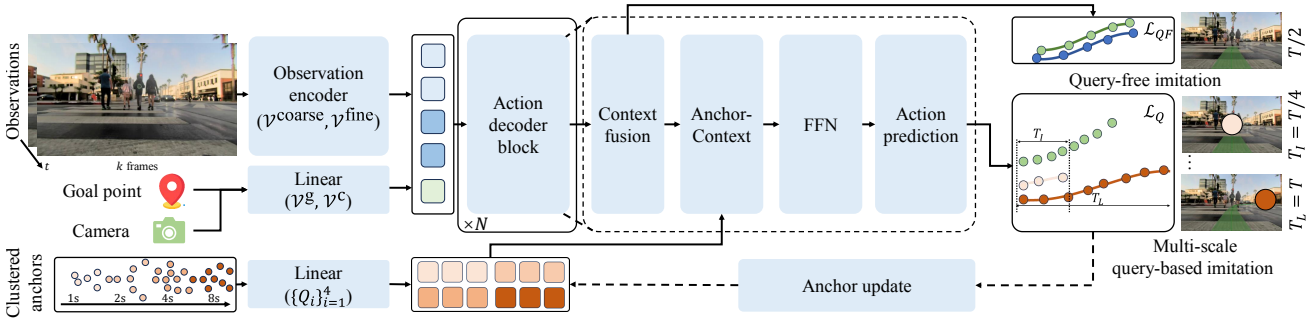



Fig. 2: **Illustration of the MIMIC framework**. The model adopts an encoder–decoder architecture that combines coarse historical embeddings with fine-grained current visual observations as context. The context encoder converts the observation sequence by combining the coarse flattened features of historical frames with the fine patch-level features of the current frame, together with the goal point and camera features. The action decoder leverages time-horizon-specific anchors to produce actions parameterized by GMMs across multiple horizons, thereby enhancing the output's diversity and robustness.

$T_h$ ego-states $e_{t-T_h:t}$ (e.g., GPS locations, velocities, orientations), and a sub-goal or route $g_t$ expressed in ego-centric coordinates. The policy $\pi_\theta$ takes observation $o_t = (i_{t-T_h:t}, e_{t-T_h:t}, g_t)$ as input and gives the action $a_t$ to control the robot. In the paradigm of imitation learning, the goal is to train a policy $\pi_\theta$ by minimizing the discrepancy between the agent's actions and the expert demonstrations. Formally, given expert trajectories $\mathscr{D} = (o_t, a_t)_{t=0}^{T}$, the objective is $\min_\theta \mathbb{E}_{(o_t,a_t)\sim\mathscr{D}}[\mathscr{L}(\pi_\theta(o_t), a_t)]$. In our formulation, $\pi_\theta$ outputs a probability distribution over candidate actions, and we adopt the negative log-likelihood (NLL) loss for supervision, *i.e.*,

$$\mathscr{L}(\pi_\theta(o_t), a_t) = -\log \pi_\theta(a_t|o_t). \tag{1}$$

For the action space $\mathscr{A}$, we define it as a sequence of waypoints sampled at a fixed frame rate. Each action $a_t \in \mathscr{A} \subset \mathbb{R}^{T\times 3}$ corresponds to a trajectory segment represented in bird's-eye view (BEV), where each waypoint encodes a 2D location and an orientation in ego-centric coordinates. To parametrize the model for the action distribution, we use a Gaussian Mixture Model (GMM) [24]. Specifically, at each timestep $t$, the policy $\pi_\theta$ outputs the parameters of a mixture distribution.

$$\pi_\theta(a_t|o_t) = \sum_{m=1}^{M} p_{\theta,m}(o_t)\mathscr{N}(\mu_{\theta,m}(o_t), \sigma_{\theta,m}(o_t)), \tag{2}$$

where $\mu_{\theta,m}(o_t) = \{(\hat{x}_{t+\tau}, \hat{y}_{t+\tau}, \hat{\psi}_{t+\tau})\}_{\tau=1}^{T}$ denotes the predicted waypoint sequence (2D position and heading) over horizon $T$ for the $m$-th Gaussian component conditioned on observation $o_t$ and $\sigma_{\theta,m}(o_t)$ denotes the corresponding variance capturing the uncertainty of the predicted waypoints.

### *B. Multi-scale Imitation Learning with Anchors*

**Multi-scale supervision.** Before introducing the model architecture, we first present the key modeling of the action space in our framework. While many existing imitation learning methods supervise the policy via the difference between the ground truth and model outputs at a single temporal scale, typically focusing on short-term predictions to ensure immediate responsiveness. However, this paradigm often leads to shortcut learning [25], where the model relies on spurious correlations rather than learning the intended underlying meaningful representations. Such behavior is particularly problematic in navigation tasks, which require both fine-grained interaction and global consistency to handle complex urban environments with many pedestrians, vehicles, road structures, etc. Therefore, we argue for introducing a multi-scale action space from short-horizon to long-horizon, where the policy is explicitly supervised across multiple temporal scales, enabling it to learn both immediate behaviors and long-term goal-aligned behaviors within a unified framework.

Concretely, we enrich the action space $\mathscr{A}$ by incorporating a multi-level supervision across different temporal horizons. Instead of supervising the policy at a single scale, we provide guidance simultaneously at the immediate, short, medium, and long horizons, denoted as $\{\mathscr{A}_1, \mathscr{A}_2, \mathscr{A}_3, \mathscr{A}_4\}$. This hierarchical supervision mitigates the shortcut behavior observed with single-horizon training [9] — where the model tends to optimize only for immediate success. As a result, the policy is encouraged to align fine-grained reactivity with long-term planning, yielding a more expressive and stable navigation model. Specifically, $a_{t,i} = \{(x_{t+\tau}, y_{t+\tau}, \psi_{t+\tau})\}_{\tau=1}^{T_i} \in \mathscr{A}_i$ and $\left\{T_1 = \frac{T}{8}, T_2 = \frac{T}{4}, T_3 = \frac{T}{2}, T_4 = T\right\}$ in our setting.

**Model architecture.** As shown in Fig. 2, we adopt an encoder–decoder architecture to model the policy. The encoder processes multimodal inputs—RGB observations, ego-states, and goal signals—into a compact spatiotemporal representation. Specifically, we encode the history of image observations $i_{t-k:t-1}$ using a visual backbone initialized from DINOv3 [26]. Each historical image within the $T_h$ input frames is first encoded into a high-dimensional embedding, forming a coarse temporal feature sequence $\mathscr{V}^{\text{coarse}} \in \mathbb{R}^{T_h\times C}$. For the current image observation $i_t$, we extract patch-level features from the backbone initialized from DINOv3 [26], and image patches are then downsampled via grid pooling and flattened into a sequence of tokens $\mathscr{V}^{\text{fine}} \in \mathbb{R}^{64\times C}$, which preserve fine-grained spatial details such as obstacles and scene geometry. The navigation goal is modeled as a compact

3D vector $(d, \cos\phi_{\text{goal}}, \sin\phi_{\text{goal}})$, encoding the distance and relative orientation to the target. Camera intrinsic parameters, together with the camera's 3D location relative to the robot center, are denoted by $c \in \mathbb{R}^{16}$. Both goal and camera parameters are projected into the embedding space $\mathscr{V}^g, \mathscr{V}^c \in \mathbb{R}^{1\times C}$ using an MLP. Each coarse visual token $\mathscr{V}_i^{coarse}$ is first modulated via a FiLM layer [27] to incorporate conditioning temporal information $\mathscr{V}_i^{coarse} \leftarrow \mathscr{V}_i^{coarse} \odot \gamma_i + \beta_i$, where $(\gamma_i, \beta_i) \in \mathbb{R}^C$ are scaling and shifting parameters generated from time-step $t-i$ relative to the current frame.

The action decoder comprises a stack of context-fusion and trajectory-refinement layers. At each layer, we first fuse the context features $\mathscr{V} = [\mathscr{V}^{coarse}, \mathscr{V}^{fine}, \mathscr{V}^g, \mathscr{V}^c]$ via multi-head attention [28] $\mathscr{V}' = \text{MHA}(Q = \mathscr{V}; K, V = \mathscr{V})$. Subsequently, the context features are used as keys and values for action decoding, allowing the decoder to attend to relevant spatial-temporal cues during trajectory prediction. The decoder generates actions by referencing a set of anchor trajectories, which serve as structured priors for plausible motion patterns. These anchors are pre-generated from data statistics based on K-means [29]. More specifically, instead of relying on a single query set, we generate four scale-specific anchor sets $\{\mathbb{A}_i\}_{i=1}^4$; $\mathbb{A}_i \subset \mathbb{R}^{64\times 3}$, $\forall i$ that correspond to the immediate, short, medium, and long horizons. They would be mapped to query tokens $\{\mathscr{Q}_i\}_{i=1}^4$ via a linear layer. Each query set interacts with the encoder representation $\mathscr{Q}_i' = \text{MHA}(Q = \mathscr{Q}_i; K, V = \mathscr{V}')$, enabling the model to jointly capture local reactivity and global consistency across different temporal scales. Given the multi-scale queries $\{\mathscr{Q}_i\}_{i=1}^4$ and the context condition $\mathscr{V}$ as input, $k$-th decoding layer $\mathscr{F}_{k,\theta}$ generates five trajectory predictions: four query-based heads, each conditioned on a specific query and the context, and one query-free head that relies solely on the contextual information $\mathscr{V}'$, *i.e.*,

$$\{\hat{\mathscr{T}}_{\text{QF}}, \hat{\mathscr{T}}_{\text{Q}}\} = \mathscr{F}_{k,\theta}(\mathscr{Q}_{1:4}, \mathscr{V}), \tag{3}$$

$$\hat{\mathscr{T}}_{\text{Q}} = \{\hat{p}_{i,m}, \hat{\mathscr{T}}_{i,m}\}_{i=1:4, m=1:M}, \tag{4}$$

where $\hat{\mathscr{T}}i,m$ denotes the predicted trajectories and $\hat{p}i,m$ the corresponding confidence scores of mode $m$ at horizon $i$, and $\hat{\mathscr{T}}_{QF}$ is the query-free trajectory prediction.

For each data sample, we assign a positive label to the mode $h_i$ within the candidate trajectory set $\{\hat{\mathscr{T}}i,m\}_{m=1}^M$ at horizon $i$, where the selected anchor trajectory $\hat{\mathscr{T}}i,h_i$ has the closest end-point to the ground-truth trajectory $\mathscr{T}i^{\text{gt}}$. That is, $p_{i,h_i} = 1$ and $p_{i,m} = 0$ for all $m \neq h_i$. For simplicity, we assume a fixed covariance $\Sigma = 0$, such that each trajectory mode degenerates into a deterministic prediction. In parallel, we introduce an auxiliary query-free (QF) reconstruction task that directly predicts future actions from the encoded visual patches, without relying on decoder queries. The QF head generates a single trajectory at a fixed short-term horizon (e.g., $\frac{T}{4}$), promoting fine-grained short-horizon supervision.

$$\mathscr{L}_k = \mathscr{L}_{k,Q} + \mathscr{L}_{k,QF}, \tag{5}$$

$$\mathscr{L}_{k,Q} = \sum_{i=1}^{4}\sum_{m=1}^{M}[\mathscr{L}_{k,i;reg} + \lambda \cdot \mathscr{L}_{k,i;cls}], \tag{6}$$

where $\mathscr{L}_{k,QF}$ and $\mathscr{L}_{k,i;reg}$ are regression loss terms between the prediction and ground truth, and $\mathscr{L}_{k,i;cls}$ is the BCE loss between $p_{i,h_i}$ and $\hat{p}_{i,h_i}$. Finally, the overall training objective averages the supervision over all $K$ decoder layers.

### C. Teleoperation Data Expansions

**Corrective behavior expansions.** Since the recorded logs are dominated by normal and straightforward observations and actions, they rarely include demonstrations that show how to recover from failure or near-failure cases [12, 13], for instance, the corrective actions to take when a vehicle starts drifting off its intended path. As a result, a policy trained purely by imitating demonstration data cannot learn to recover from its own mistakes. To simulate such failure-correction scenarios, we deliberately generate trajectories in which the model would take incorrect actions (e.g., deviating from the intended route, stepping onto the grass, colliding with obstacles, or stopping prematurely), and then provide corrective actions as supervision. As illustrated in Fig. 3, we begin by estimating a continuous metric depth sequence $I_D \in \mathbb{R}^{(T_h+T)\times H\times W}$ from ViPE [30] to annotate the surrounding scene geometry. After that, we leverage depth and RGB observations to construct a colored point cloud sequence $\mathscr{P} \in \mathbb{R}^{(T_h+T)\times(H\times W)\times 6}$ in the ego-centric frame, providing a 3D geometric representation of the scene, which we use to perturb trajectories. To induce deviations, we define a shifting sequence $\Delta\mathscr{T}(\tau)$ that smoothly varies from 0 back to 0 over the prediction horizon, following a sine-like profile $\Delta\mathscr{T}(\tau) = \alpha \cdot \sin(\frac{\pi\cdot\tau}{T_h+T})$, where $\alpha$ controls the maximum displacement. The novel RGB observations $i'_{t-T_h:t}$ are synthesized under the perturbation $\{\Delta\mathscr{T}(\tau)|\tau = t-T_h, ..., t-1\}$ by reprojecting the colored point cloud sequence $\mathscr{P}$ into the ego-centric camera frame, conditioned on the perturbed trajectories. Given the perturbed observation sequence, the supervision trajectory is the recovery trajectory generated from the original one and shifted by $\{\Delta\mathscr{T}(\tau)|\tau = t, ..., t+T-1\}$. This perturbation scheme introduces temporary lateral or longitudinal drifts into the original expert trajectory, mimicking realistic failure cases such as veering off-road or hesitating at obstacles. By pairing each perturbed trajectory with a corrective failure-to-recovery maneuver, we obtain failure–correction pairs that enable the policy to learn robust recovery behaviors.

**Sensor augmentation.** Besides the lack of corrective behaviors in the collected teleoperation dataset, the visual appearance of recorded videos is often overly simple, with fixed lighting, limited weather conditions, and low diversity of backgrounds. More importantly, teleoperation logs from the real world often over-represent normal behaviors (e.g., straight-line movement on clear sidewalks) while under-representing rare but safety-critical events, such as erroneous operations where the robot steps onto the grass, or pauses at crowded intersections. To address data imbalance, we introduce generative augmentation to enrich both the sensory inputs and the state–action pairs.

The key principle is to preserve scene geometry and structure while altering visual appearance. Prior work com-

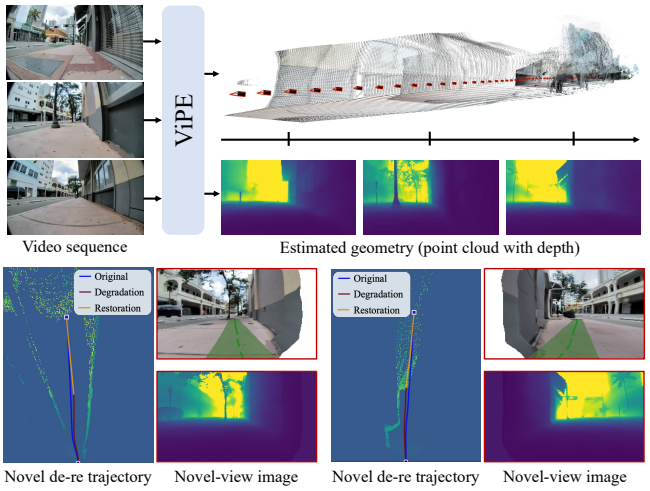



Fig. 3: Illustration of the corrective behavior expansion. We first estimate the depth sequence and reconstruct a point cloud. Given the 3D point cloud, we perturb the trajectory using a deviation–recovery noise sequence. Then we synthesize corresponding observation-action pairs.

monly employs depth- or semantic-based re-rendering [31, 32] to diversify illumination and textures. However, these approaches often introduce artifacts, such as inconsistent blending, where nearby objects inherit background lighting conditions. To alleviate this issue, we adopt a relighting model, Light-A-Video [33], that preserves scene geometry while modifying global appearance. Specifically, the model disentangles foreground objects $I_f$ from the background $I_b$ using depth, applies prompt-based relighting with different strength coefficients to the foreground and background, *i.e.*,

$$I' = f_{\text{relight}}(I_f; \alpha_f, p) \oplus f_{\text{relight}}(I_b; \alpha_b, p), \quad \alpha_f < \alpha_b, \tag{7}$$

where $f_{\text{relight}}(\cdot)$ denotes the prompt-based relighting model, $p$ is the textual prompt controlling illumination style, and $\alpha_f = 0.1, \alpha_b = 0.5$ are the respective relighting strengths applied to the foreground and background. As shown in Fig. 4, this asymmetric design preserves foreground consistency while enhancing background diversity.

## IV. Experiments

eWe evaluate our proposed approach, MIMIC, on both offline sidewalk videos and real-world deployments with a wheeled robot. We report the overall performance of our model in comparison with prior baselines, conduct ablation studies to analyze the contributions of all components, and provide qualitative results to illustrate the effectiveness of the proposed approach.

### A. Dataset

We have collected a large-scale video teleoperation dataset, **CoS** (short for Coco-on-SideWalks). In total, the dataset contains 3,040 trajectories collected by multiple wheeled robots from Coco Robotics[1] navigating diverse sidewalks across various US cities, each lasting 1 minute,

[1] https://www.cocodelivery.com/

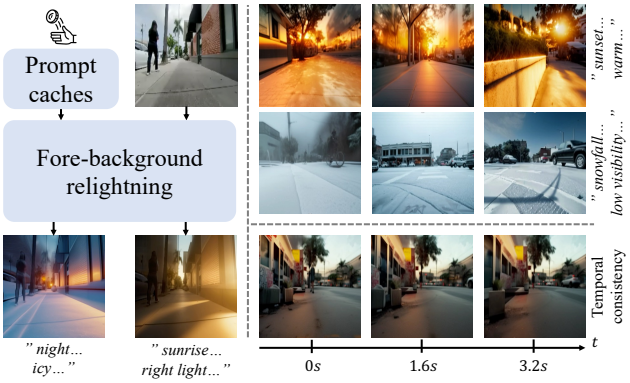



Fig. 4: Illustration of the sensor augmentation. A pretrained relighting model is used to modify the scene guided by different lighting prompts. The original scenario is segmented into foreground and background regions where different relighting parameters are applied. The outputs are then blended to synthesize novel relighted observations.

amounting to about 50 hours of data. For each trajectory segment, we record fisheye RGB videos at 20Hz, along with synchronized robot-state logs that include position, orientation, linear velocity, and angular velocity, derived from GPS and onboard odometry. We split the dataset into 2,740 trajectories for training, 200 for validation, and 100 for testing. As illustrated in Fig. 5, we present qualitative results of predicted trajectories alongside ground truth across several scenarios in the dataset.

**Dataset curation.** After collecting teleoperation logs, we perform a systematic curation process to ensure data quality and consistency. Specifically, the process involves:

(i) Behavior classification and balancing. We classify trajectories into basic behavioral categories (e.g., straight walking, turning, stopping). Since straightforward walking behaviors dominate the teleoperation logs, we downsample redundant segments while retaining a higher proportion of diverse behaviors, thereby alleviating class imbalance.

(ii) Filtering abnormal segments. We remove sequences in which the robot exhibits undesirable motions, such as sensor-induced rotations while staying still or backward behaviors. This filtering step prevents the model from overfitting to noisy or unrepresentative actions.

(iii) Goal point definition. For each trajectory, the goal point is defined in two ways: (1) randomly sampling the next 5–20 frames like [8, 9], or (2) splitting the trajectory into $N$ segments ($N \in [3, 7]$) and selecting the nearest segment endpoint. This strategy avoids shortcut learning by sampling not only the immediate few frames that are strongly correlated with the current state.

(iv) Trajectory smoothing. For each sub-trajectory of length $(T_h + T)$ used in training, we apply slerp to smooth the recorded poses, thereby reducing variations caused by differences among teleoperators and noise introduced by operation habits. Specifically, we first compute the total trajectory length and then regenerate the trajectory by interpolating poses at a constant velocity along the path.

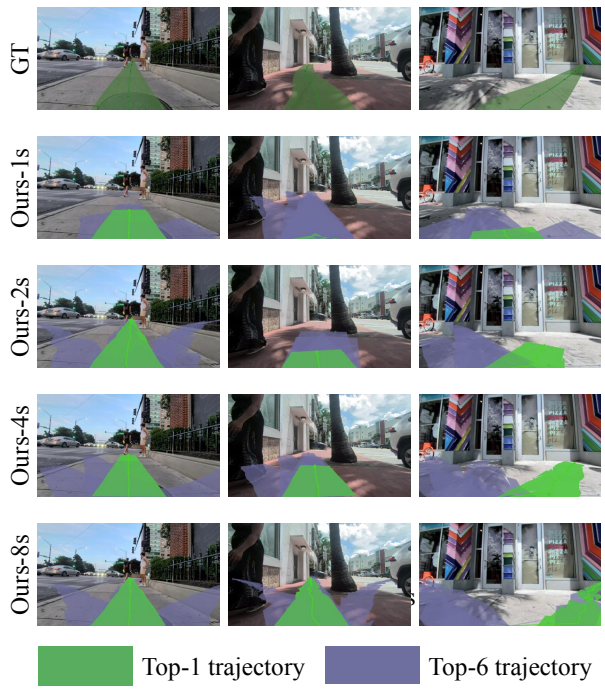



Fig. 5: Qualitative results of MIMIC on the **CoS** test set. The green trajectory denotes the one with the highest probability, while the others represent the top-6 trajectories filtered by non-maximum suppression (NMS).

TABLE I: Open-loop evaluation on **CoS-Regular**.

| | $\text{minADE}_{1s}\downarrow$ | $\text{minFDE}_{1s}\downarrow$ | mAP ↑ | $\text{L2}_{1s}\downarrow$ | $\text{L2}_{2s}\downarrow$ |
|---|---|---|---|---|---|
| GNM‡ | 0.594 | 0.988 | - | 0.988 | - |
| ViNT‡ | 0.638 | 1.056 | - | 1.056 | - |
| NoMaD‡ | 0.523 | 0.858 | 0.216 | 1.072 | 2.182 |
| MBRA | 0.617 | 1.019 | - | 1.019 | 2.034 |
| CityWalker | 0.648 | 1.125 | - | 1.125 | - |
| ViNT* | 0.247 | 0.425 | - | 0.425 | 0.925 |
| CityWalker* | 0.180 | 0.353 | - | 0.353 | 0.786 |
| MIMIC | **0.071** | **0.129** | **0.695** | **0.342** | **0.700** |

TABLE II: Open-loop evaluation on **CoS-Recovery**.

| | $\text{minADE}_{1s}\downarrow$ | $\text{minFDE}_{1s}\downarrow$ | mAP ↑ | $\text{L2}_{1s}\downarrow$ | $\text{L2}_{2s}\downarrow$ |
|---|---|---|---|---|---|
| MBRA | 2.586 | 3.297 | - | 3.297 | 7.000 |
| CityWalker | 0.929 | 1.740 | - | 1.740 | - |
| CityWalker* | 0.398 | 0.695 | - | 0.695 | 1.397 |
| MIMIC | **0.196** | **0.328** | **0.565** | **0.645** | **1.348** |

### B. Implementation Details

Our neural network consists of 4 encoder–decoder layers with a hidden dimension of 512. The observation encoder is initialized from DinoV3-S [26]. Each input sequence consists of 16 frames sampled at 5 Hz, with all images resized to a resolution of $256 \times 256$. For trajectory prediction, we define the longest horizon as 40 frames at 5Hz, and each horizon is associated with 64 anchors for multi-modal decoding. All parameters are trained jointly in an end-to-end manner.

We adopt a cosine learning rate schedule with an initial learning rate of $1 \times 10^{-4}$ and a total batch size of 192. To improve the model robustness, we apply random masking during training: the goal token is masked with a probability of 0.5 to force the model to exploit contextual features, while other tokens are masked with a probability of 0.2. The model is trained for 100 epochs, which takes approximately 1.5 days on 8 NVIDIA L40S GPUs.

### C. Open-Loop Evaluation

We first evaluate our approach in an open-loop setting, where predicted trajectories are compared against ground-truth future trajectories on the test set. We conduct experiments on two subsets of our dataset: **CoS-Regular** and **CoS-Recovery**. The SideWalks-Regular set contains normal teleoperation trajectories, while the SideWalks-Recovery set includes perturbed observations. This separation allows us to evaluate both the prediction accuracy under standard conditions and the robustness of the policy when confronted with deviation-induced observations. For evaluation, we adopt the standard open-loop metrics proposed in prior works [34, 35]. A trajectory is considered positive if its endpoint at 1s lies within 1m of the ground truth. It is worth noting that previous works generate only a single-mode trajectory. Therefore, when reporting mAP, we report them using only the Average Precision (AP).

**Baselines.**We compare against several state-of-the-art navigation foundation models: 1) image-goal approaches‡ including GNM [6], ViNT [7], NoMaD [8], and 2) point-based approaches CityWalker [9], MBRA [10], ViNT* and CityWalker* (*denotes model re-trained on our dataset).

Tab. I and Tab. II show that MIMIC consistently outperforms all baseline methods on both Regular and Recovery test sets. Specifically, MIMIC achieves a 60.6% lower $\text{minADE}_{1s}$ and 63.5% lower $\text{minFDE}_{1s}$ than the second-best method (CityWalker*) on SideWalks-Regular, along with a 19.5% improvement in $\text{L2}_{2s}$. On the SideWalks-Recovery set, MIMIC yields a 50.8% reduction in $\text{minADE}_{1s}$ and 52.8% in $\text{minFDE}_{1s}$ compared to CityWalker*, while also achieving a 3.5% lower $\text{L2}_{2s}$.

We provide qualitative results of our approach on *Sidewalks*. As illustrated in Fig. 5, the predictions remain accurate across all horizons. In the second column, our approach successfully finds a feasible path between the pedestrian and the obstacle. In the third column, when encountering a door in front, the policy attempts to avoid a collision.

### D. Ablation Study

We conduct ablation studies to evaluate the effectiveness of the model design and the data expansions.

**Effect of the model design.** We conduct ablation studies by comparing different model configurations. As illustrated in Tab. III, introducing anchor-based prediction *S* significantly improves short-term accuracy compared to relying solely on the context-based head (*QF*). The short-horizon head (*S*) achieves the lowest $\text{minADE}_{1s}$ and $\text{minFDE}_{1s}$, but its mAP is relatively low, indicating weaker overall accuracy on multi-modal prediction compared to multi-horizon settings

TABLE III: Ablation study on model design. $I, S, M, L$ denote the prediction head at immediate, short, medium, and long horizons, respectively, and $QF$ denotes the prediction head derived directly from the context features.

| $I$ | $S$ | $M$ | $L$ | $QF$ | minADE$_{1s}$ ↓ | minFDE$_{1s}$ ↓ | mAP ↑ | L2$_{2s}$ ↓ | L2$_{8s}$ ↓ |
|---|---|---|---|---|---|---|---|---|---|
| | | | | ✓ | 0.188 | 0.371 | - | 0.789 | 4.083 |
| | ✓ | | | | **0.067** | **0.113** | 0.310 | **0.596** | - |
| | | | ✓ | | 0.081 | 0.132 | 0.670 | 0.711 | 3.805 |
| ✓ | ✓ | ✓ | ✓ | | 0.074 | 0.135 | 0.680 | 0.637 | 4.068 |
| ✓ | ✓ | ✓ | ✓ | ✓ | 0.071 | 0.129 | **0.695** | 0.700 | **3.718** |

TABLE IV: Ablation study on data expansions. $\mathscr{D}_S$ denotes the set from sensor augmentation, and $\mathscr{D}_C$ denotes the set from corrective behavior expansion.

| $\mathscr{D}_S$ | $\mathscr{D}_C$ | mAP ↑ | L2$_{1s}$ ↓ | L2$_{2s}$ ↓ | L2$_{4s}$ ↓ | L2$_{8s}$ ↓ |
|---|---|---|---|---|---|---|
| | | 0.660 | 0.381 | 0.789 | 1.677 | 4.139 |
| ✓ | | 0.670 | 0.358 | 0.748 | 1.617 | 3.940 |
| | ✓ | 0.679 | 0.355 | 0.754 | 1.621 | 4.028 |
| ✓ | ✓ | **0.695** | **0.342** | **0.700** | **1.434** | **3.718** |
| ✓ | | 0.374 | 0.914 | 1.949 | 4.817 | 9.137 |
| ✓ | ✓ | **0.565** | **0.645** | **1.348** | **4.499** | **8.562** |

$\{I, S, M, L\}$. Combining all horizon-specific heads with the context head provides a balanced trade-off between short-term accuracy and long-term consistency, yielding more stable overall performance.

**Effect of data expansions.** We conduct ablation studies on the effectiveness of different data expansion strategies. As shown in Tab. IV, each expansion individually improves performance over the baseline on SideWalks-Regular, and combining both yields the best results across all metrics, demonstrating their complementary benefits. Furthermore, on Sidewalks-Recovery, incorporating $\mathscr{D}_C$ significantly reduces both short-horizon and long-horizon errors, indicating that corrective behavior expansion enables the policy to learn from near-failure cases and recover from deviations.

## V. Real-World Deployment

In this section, we present details of our real-world deployments with the wheeled robot[2].

### A. Experimental Setup

We validate the effectiveness of the proposed approach across four environments, evaluated in both daytime and nighttime settings. The routes span different lengths (20m, 20m, 50m and 400m) to validate both short-horizon and long-horizon navigation performance. In each environment, a pedestrian walks across the path of the robot twice along the route to evaluate its performance in real-world sidewalk scenarios. For short-horizon trials, goal points are defined relative to the robot, while in long-horizon trials, GPS-based waypoints are used for continuous navigation. We use the success rates for goal reaching and pedestrian avoidance, and the success weighted by path length (SPL), for evaluation across all scenarios. In long-horizon navigation, we do not terminate the task when the robot goes off-route or collides. Instead, a human operator intervenes to take control, and we report the number of interventions as an additional metric in the 400m navigation task.

[2]A demo video is available on the project page.

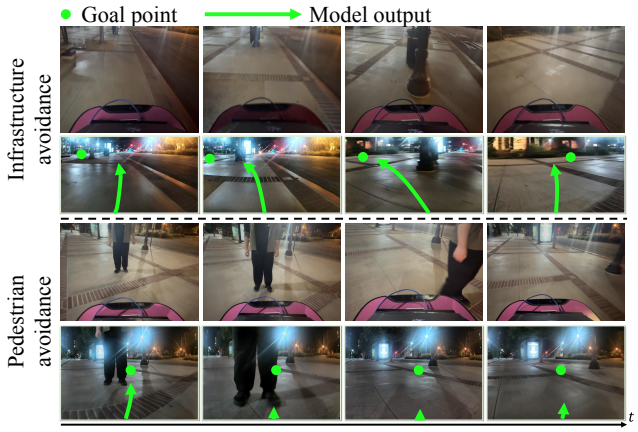



Fig. 6: Qualitative results of MIMIC in the real-world.

TABLE V: Closed-loop evaluation in the real world.

| | Goal Reaching ↑ | Pedestrian Avoidance ↑ | SPL↑ | Intervention Times ↓ |
|---|---|---|---|---|
| CityWalker | 0.55 | 0.18 | 0.46 | 19 |
| CityWalker* | 0.70 | 0.29 | 0.44 | 11 |
| MIMIC | **0.90** | **0.76** | **0.69** | **4** |

### B. Results

As illustrated in Tab. V, MIMIC outperforms CityWalker and its fine-tuned variant. MIMIC achieves the highest success rate in all navigation tasks. It requires far fewer intervention times, demonstrating the effectiveness of the proposed approach. We further provide qualitative results of two scenarios in Fig. 6. In the first scenario, the policy successfully navigates toward a goal point defined behind a tree: the robot turns to reach the target once sufficient space is available. In the second scenario, when a pedestrian is in front of the robot, the robot yields to avoid a collision.

## VI. Conclusions and Future Work

In this work, we present an imitation learning framework, MIMIC, for learning a sidewalk autopilot from the teleoperation dataset. First, we introduce corrective behavior expansion to extend the training distribution. Second, we propose using multi-scale, horizon-specific anchors for learning. We validate the proposed method on both the offline test set and real-world deployments, demonstrating its effectiveness.

**Limitations.** While MIMIC demonstrates its effectiveness, it also has limitations. Without explicit 3D or semantic supervision, the policy may degrade in highly cluttered or visually ambiguous environments. Introducing additional visual supervision, or distilling such knowledge from pre-trained models, would be a promising direction.

## VII. Acknowledgment

The project was supported by the NSF Grants CNS-2235012 and IIS-2339769. Honglin He is supported by the Amazon Trainium Fellowship. We thank Coco Robotics for the generous donation of data and equipment.

## References

[1] V. Engesser, E. Rombaut, L. Vanhaverbeke, and P. Lebeau. "Autonomous delivery solutions for last-mile logistics operations: A literature review and research agenda". In: *Sustainability* 15.3 (2023), p. 2774.

[2] A. Tuomi, I. P. Tussyadiah, and J. Stienmetz. "Applications and implications of service robots in hospitality". In: *Cornell Hospitality Quarterly* 62.2 (2021), pp. 232–247.

[3] X. Liu, L. Zhang, and T. Zhu. "Service robots in my workplace: Effects of employee-service robot co-work experiences on psychological empowerment". In: *Journal of Hospitality Marketing & Management* 34.2 (2025), pp. 175–203.

[4] D. A. Pomerleau. "Alvinn: An autonomous land vehicle in a neural network". In: *NeurIPS* 1 (1988).

[5] N. Lambert, K. Pister, and R. Calandra. "Investigating compounding prediction errors in learned dynamics models". In: *arXiv preprint:2203.09637* (2022).

[6] D. Shah, A. Sridhar, A. Bhorkar, N. Hirose, and S. Levine. "Gnm: A general navigation model to drive any robot". In: *arXiv preprint:2210.03370* (2022).

[7] D. Shah, A. Sridhar, N. Dashora, K. Stachowicz, K. Black, N. Hirose, and S. Levine. "ViNT: A foundation model for visual navigation". In: *arXiv preprint:2306.14846* (2023).

[8] A. Sridhar, D. Shah, C. Glossop, and S. Levine. "Nomad: Goal masked diffusion policies for navigation and exploration". In: *2024 ICRA*. IEEE. 2024, pp. 63–70.

[9] X. Liu, J. Li, Y. Jiang, N. Sujay, Z. Yang, J. Zhang, J. Abanes, J. Zhang, and C. Feng. "Citywalker: Learning embodied urban navigation from web-scale videos". In: *CVPR*. 2025, pp. 6875–6885.

[10] N. Hirose, L. Ignatova, K. Stachowicz, C. Glossop, S. Levine, and D. Shah. "Learning to Drive Anywhere with Model-Based Reannotation". In: *arXiv preprint:2505.05592* (2025).

[11] H. He, Y. Ma, W. Wu, and B. Zhou. "From Seeing to Experiencing: Scaling Navigation Foundation Models with Reinforcement Learning". In: *arXiv preprint:2507.22028* (2025).

[12] M. Bansal, A. Krizhevsky, and A. Ogale. "Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst". In: *arXiv preprint:1812.03079* (2018).

[13] M. Goff, G. Hogan, G. Hotz, A. du Parc Locmaria, K. Raczy, H. Schäfer, A. Shihadeh, W. Zhang, and Y. Yousfi. "Learning to drive from a world model". In: *CVPR*. 2025, pp. 1964–1973.

[14] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. "g 2 o: A general framework for graph optimization". In: *2011 ICRA*. IEEE. 2011, pp. 3607–3613.

[15] X. Lei, M. Wang, W. Zhou, and H. Li. "Gaussnav: Gaussian splatting for visual navigation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025).

[16] X. Puig, E. Undersander, A. Szot, M. D. Cote, T.-Y. Yang, R. Partsey, R. Desai, A. W. Clegg, M. Hlavac, S. Y. Min, et al. "Habitat 3.0: A co-habitat for humans, avatars and robots". In: *arXiv preprint:2310.13724* (2023).

[17] S. Ren, K. He, R. Girshick, and J. Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks". In: *NeurIPS* 28 (2015).

[18] A. H. Qureshi, A. Simeonov, M. J. Bency, and M. C. Yip. "Motion planning networks". In: *2019 ICRA*. IEEE. 2019, pp. 2118–2124.

[19] M. Mittal, C. Yu, Q. Yu, J. Liu, N. Rudin, D. Hoeller, J. L. Yuan, R. Singh, Y. Guo, H. Mazhar, et al. "Orbit: A unified simulation framework for interactive robot learning environments". In: *RAL* 8.6 (2023), pp. 3740–3747.

[20] S. Ross, G. Gordon, and D. Bagnell. "A reduction of imitation learning and structured prediction to no-regret online learning". In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2011, pp. 627–635.

[21] H. Wang, A. H. Tan, A. Fung, and G. Nejat. "X-Nav: Learning End-to-End Cross-Embodiment Navigation for Mobile Robots". In: *arXiv preprint:2507.14731* (2025).

[22] Z. M. Peng, W. Mo, C. Duan, Q. Li, and B. Zhou. "Learning from active human involvement through proxy value propagation". In: *NeurIPS* 36 (2023), pp. 77969–77992.

[23] Z. Peng, Z. Liu, and B. Zhou. "Data-efficient learning from human interventions for mobile robots". In: *arXiv preprint:2503.04969* (2025).

[24] S. Shi, L. Jiang, D. Dai, and B. Schiele. "Motion transformer with global intention localization and local movement refinement". In: *NeurIPS* 35 (2022), pp. 6531–6543.

[25] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. "Shortcut learning in deep neural networks". In: *Nature Machine Intelligence* 2.11 (2020), pp. 665–673.

[26] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, et al. "Dinov3". In: *arXiv preprint:2508.10104* (2025).

[27] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. "Film: Visual reasoning with a general conditioning layer". In: *AAAI*. Vol. 32. 1. 2018.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need". In: *NeurIPS* 30 (2017).

[29] S. Lloyd. "Least squares quantization in PCM". In: *IEEE Transactions on Information Theory* 28.2 (1982), pp. 129–137.

[30] J. Huang, Q. Zhou, H. Rabeti, A. Korovko, H. Ling, X. Ren, T. Shen, J. Gao, D. Slepichev, C.-H. Lin, J. Ren, K. Xie, J. Biswas, L. Leal-Taixe, and S. Fidler. "ViPE: Video Pose Engine for 3D Geometric Perception". In: *NVIDIA Research Whitepapers*. 2025.

[31] N. Agarwal, A. Ali, M. Bala, Y. Balaji, E. Barker, T. Cai, P. Chattopadhyay, Y. Chen, Y. Cui, Y. Ding, et al. "Cosmos world foundation model platform for physical ai". In: *arXiv preprint:2501.03575* (2025).

[32] H. A. Alhaija, J. Alvarez, M. Bala, T. Cai, T. Cao, L. Cha, J. Chen, M. Chen, F. Ferroni, S. Fidler, et al. "Cosmos-transfer1: Conditional world generation with adaptive multimodal control". In: *arXiv preprint:2503.14492* (2025).

[33] Y. Zhou, J. Bu, P. Ling, P. Zhang, T. Wu, Q. Huang, J. Li, X. Dong, Y. Zang, Y. Cao, et al. "Light-a-video: Training-free video relighting via progressive light fusion". In: *arXiv preprint:2502.08590* (2025).

[34] B. Varadarajan, A. Hefny, A. Srivastava, K. S. Refaat, N. Nayakanti, A. Cornman, K. Chen, B. Douillard, C. P. Lam, D. Anguelov, et al. "Multipath++: Efficient information fusion and trajectory aggregation for behavior prediction". In: *2022 ICRA*. IEEE. 2022, pp. 7814–7821.

[35] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. R. Qi, Y. Zhou, et al. "Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset". In: *ICCV*. 2021, pp. 9710–9719.