SPATIO-TEMPORALLY CONSISTENT PANORAMIC GENERATION FOR VISION-LANGUAGE NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

The generalization of Vision-and-Language Navigation (VLN) agents remains constrained by the limited diversity of high-quality training environments. Generative data augmentation offers a scalable alternative, yet existing methods largely prioritize semantic diversity while imposing limited constraints on local geometry, panoramic wraparound, and cross-node continuity. To address this limitation, we propose a *spatio-temporally consistent panoramic generation (SCPGen)* framework that incorporates depth priors, cyclic boundary modeling, and view-adjacency cues into a pretrained diffusion model without additional fine-tuning. Specifically, the *Spatial Guidance Mechanism (SGM)* uses depth-weighted projection to suppress geometrically implausible content; the *Panoramic Cyclic Consistency Mechanism (PCCM)* jointly denoises cyclic seam regions to preserve left-right continuity; and the *Trajectory Consistency Mechanism (TCM)* reprojects geometry-supported content from neighboring viewpoints to maintain consistency along navigation trajectories. Together, these mechanisms generate visually diverse and spatially coherent panoramic trajectories for VLN training. Experiments across discrete and continuous VLN benchmarks demonstrate consistent improvements in unseen-environment navigation and competitive performance against augmentation methods that rely on large-scale external simulator data. Our code will be released.

1 INTRODUCTION

Vision-and-Language Navigation (VLN) requires an embodied agent to understand natural language instructions and execute a sequence of navigation actions in a 3D environment. As a fundamental task in embodied artificial intelligence, VLN has achieved substantial progress driven by advances in model architectures Anderson et al. (2018); Hao et al. (2020); Zheng et al. (2024) and indoor datasets Anderson et al. (2018); Qi et al. (2020); Ku et al. (2020); Thomason et al. (2020). Nevertheless, a long-standing bottleneck continues to hinder the practical deployment of robust VLN agents: *the severe scarcity of high-quality real-world training environments* Chen et al. (2022a); Yadav et al. (2023). Acquiring photorealistic 3D scans with dense navigability annotations is both prohibitively expensive and highly time-consuming. This limitation further restricts the generalization capability of learned navigation policies when they are exposed to unseen spatial layouts and visual appearances Li & Bansal (2023b); Wei et al. (2025).

Recent studies have attempted to alleviate this data bottleneck through a promising direction, namely augmenting training corpora with generated panoramic observations Li & Bansal (2023b); Wei et al. (2025); Zhong et al. (2024); Wang et al. (2025). However, existing data augmentation methods have yet to fully account for an important consideration: a panoramic navigation observation is not an isolated pixel plane, but rather a projection that encodes spatial information such as occlusion and geometric structure Bar-Tal et al. (2023); Rombach et al. (2022). Moreover, current approaches often build upon off-the-shelf generative models and prioritize semantic diversity, while giving comparatively less attention to spatial and local geometric constraints within the same physical world. By analyzing existing data augmentation methods, we identify three aspects that remain to be further explored: (i) *Geometric violations*: existing generation pipelines mainly synthesize RGB observations in the 2D appearance space, lacking explicit geometric priors Huo et al. (2023). However, VLN agents need to reason about traversable space, occlusion relationships, and candidate directions in 3D-structured environments Liu et al. (2024a); (ii) *Panoramic boundary discontinuity*: general text-

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

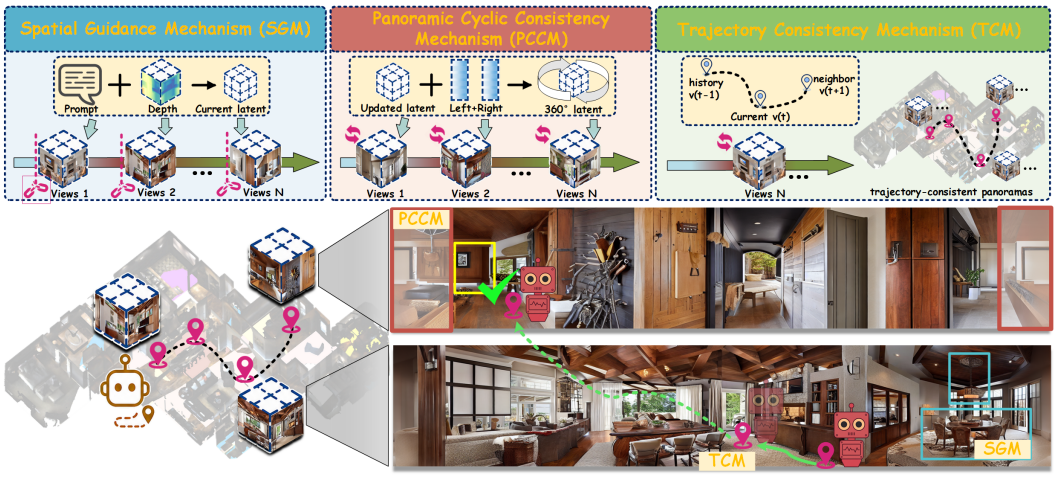


Figure 1: **SCPGen for panoramic VLN augmentation** (§1). *SGM*, *PCCM*, and *TCM* encourage consistency between overlapping crops, across the panoramic boundary, and among geometry-supported regions of trajectory observations, respectively.

to-image models Li & Bansal (2023b) typically treat panoramas as 2D rectangular canvases, and discretely generated panoramic views often exhibit fragmentation and cross-view inconsistency. As shown in Fig. 1, directly stitched images often exhibit discontinuities at the panorama seam Zhu et al. (2026); (iii) *Missing cross-node spatial connections*: existing VLN generation methods rarely consider visual associations among generated navigation nodes Bar et al. (2025); Wang et al. (2024b), leading to unstable visual and spatial relationships across viewpoints. Nevertheless, VLN is not a visual recognition task over independent static images, but a sequential decision-making process in which the current observation is tightly coupled with historical observations, executed actions, and candidate nodes Liu et al. (2024a); Ren et al. (2025).

To address the above issues, we argue that text-to-image (T2I) models must be specifically designed for indoor VLN tasks. To this end, we propose SCPGen, a spatio-temporally consistent panoramic generation framework for vision-language navigation. Our method addresses the limitations of existing approaches through three key mechanisms. First, to explicitly constrain the 3D layout and mitigate common geometric violations, we introduce the Spatial Guidance Mechanism (*SGM*). This mechanism uses estimated depth as an explicit structural prior. By conditioning generation on depth maps and optimizing a depth-weighted projection loss, it promotes locally coherent geometry and plausible depth and occlusion relationships. In addition, the Panoramic Cyclic Consistency Mechanism (*PCCM*) couples the left and right boundary regions during denoising to promote panoramic continuity. Specifically, through a unified denoising equation, we fuse the outputs obtained after multiple pre-denoising steps in the cyclic seam block into the global panorama generation process, thereby preserving semantic continuity at the cyclic seam. Finally, we introduce the Trajectory Consistency Mechanism (*TCM*). It directly constrains panoramic observations across different navigation nodes during data generation, enabling the generated results to preserve stable visual relationships between historical nodes and the current node.

We evaluate SCPGen on R2R Anderson et al. (2018), R4R Jain et al. (2019), REVERIE Qi et al. (2020), and R2R-CE Krantz et al. (2020). On R2R Val Unseen, SCPGen achieves 77.62% SR and 67.20% SPL, improving over the reported RAM results by 1.33 and 0.81 percentage points, respectively. It also improves R4R Val Unseen SR, REVERIE Val Unseen RGSPL, and R2R-CE Val Unseen SR over RAM in the corresponding comparisons. These results suggest that SCPGen-generated trajectories improve downstream performance across different navigation objectives and support transfer from discrete training trajectories to continuous navigation.

2 Related Work

Vision-and-Language Navigation (VLN). Existing VLN methods can be broadly grouped into three paradigms. Early sequence-to-sequence approaches Anderson et al. (2018); Fried et al. (2018);

Wang et al. (2019); Gu et al. (2022) established the standard VLN paradigm by grounding instruction fragments in local visual observations. However, their reliance on predominantly local vision-language alignment and recurrent state propagation limits long-horizon reasoning and global spatial modeling. More recent methods employ Transformer architectures and pretrained vision-language representations to aggregate trajectory history and construct map-enhanced spatial representations Hao et al. (2020); Hong et al. (2021); Chen et al. (2021); Wang et al. (2023); An et al. (2023); Qi et al. (2021). Although these designs improve long-range dependency modeling, their generalization remains sensitive to changes in visual appearance and spatial layout, while map-based variants introduce additional construction and maintenance costs. Another line of research leverages Large Language Models (LLMs) for zero-shot reasoning and high-level navigation planning Zheng et al. (2024); Chen et al. (2024); Long et al. (2024); Zhan et al. (2024); Deitke et al. (2022). Nevertheless, these advances remain primarily model-centric. Across all three paradigms, agents still rely on spatially faithful embodied observations to translate language-level decisions into executable navigation actions. However, geometrically consistent and spatially realizable panoramic trajectories remain scarce, limiting robust generalization to unseen environments. To address this data bottleneck, we investigate geometry-aware data augmentation from a generative perspective, synthesizing spatially realizable panoramic trajectories that provide visually diverse yet geometrically coherent supervision for VLN agents.

Generative Data Augmentation in VLN. The acquisition of high-quality real-world environments entails prohibitive human labor and time costs, making generative data augmentation a crucial focus in VLN. Existing generative methods can be broadly categorized into two complementary directions. The first line of research focuses on observation-level expansion Li et al. (2022); Zhong et al. (2024). By editing or synthesizing visual observations, these methods effectively increase environmental diversity and broaden the training distribution. However, they are inherently constrained by existing scene layouts and often struggle to maintain cross-view consistency, limiting their potential for cross-scene generalization. A second direction emphasizes semantic and supervisory generation Li & Bansal (2023b); Wei et al. (2025); Li & Bansal (2023a). Instead of directly modifying pixels, these approaches generate pseudo-labels, language-vision pairs, or future semantics. While highly efficient and adaptable, their focus remains primarily on the semantic level, leaving complex spatial structural constraints and panoramic geometric continuity largely unexploited.

Although both paradigms alleviate data scarcity, a shared bottleneck persists: they rarely treat spatial consistency as a central, unified constraint during generation. For panoramic navigation, synthesized data must guarantee not only visual realism and semantic diversity but also strict cross-view geometric continuity and spatial reachability. Motivated by this, our work explicitly incorporates spatial guidance signals – specifically depth, view adjacency, and path geometry – into the generation process, ensuring the synthesized environments are both diverse and geometrically navigable.

Spatial Guidance. In generative models, spatial guidance signals encode structural priors (*e.g.*, geometric correspondences and layouts) to improve the controllability and consistency of synthesized results Zhang et al. (2023); Mou et al. (2024); Li et al. (2023); Bar-Tal et al. (2023); Ye et al. (2024); Kalischek et al. (2025). In VLN, these signals bridge language semantics and 3D reasoning. Early works relied on waypoint supervision Raychaudhuri et al. (2021); Krantz et al. (2021), while later methods adopted topological maps, grid memory, and Bird's-Eye-View (BEV) projections to model global connectivity and trajectory history Chen et al. (2022b); Wang et al. (2023); An et al. (2023); Liu et al. (2023a; 2024a). Depth has subsequently emerged as a fundamental cue for local 3D structure and occlusion, motivating the use of RGB-derived depth for navigation learning Huo et al. (2023); Han et al. (2025); Zhao et al. (2025). Despite these advances, three limitations remain: (i) depth, topology, and path geometry are typically modeled separately; (ii) VLN augmentation emphasizes semantic diversity while overlooking cross-view geometric continuity in panorama generation; and (iii) panoramas at different navigation nodes are generated independently, failing to preserve visual dependencies induced by navigation history, actions, and shared regions.

We unify depth priors, panoramic cyclic constraints, and trajectory visual continuity within a single generation framework. Specifically, RGB-derived depth guides the local structure of each synthesized observation, while explicit modeling of the left-right panoramic boundary preserves wraparound continuity. Finally, depth-guided reprojection is combined with navigation history and neighboring views to preserve shared visual content across nodes. The resulting VLN data jointly satisfy local geometric plausibility, panoramic continuity, and trajectory consistency, thereby improving generalization to open-world environments.

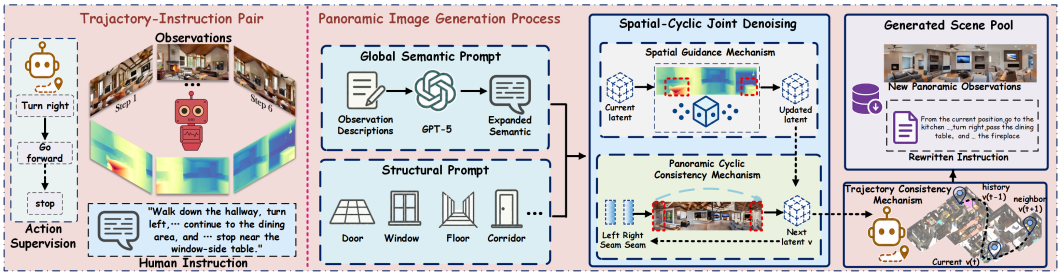


Figure 2: **Overview of SCPGen** (§3). Given scene descriptions, depth maps, camera poses, and topological adjacency, SCPGen synthesizes spatially realizable panoramic trajectories through three complementary mechanisms.

3 METHODOLOGY

3.1 METHOD OVERVIEW AND DIFFUSION PRELIMINARIES

Overview. As illustrated in Fig. 2, SCPGen augments VLN training with spatially realizable panoramic trajectories synthesized by a pretrained diffusion model. It operates at three complementary scales: (i) Spatial Guidance Mechanism (*SGM*) (§3.2) constrains local geometry via depth-weighted overlap guidance; (ii) Panoramic Cyclic Consistency Mechanism (*PCCM*) (§3.3) enforces seamless 360° wraparound through a cyclic seam block; and (iii) Trajectory Consistency Mechanism (*TCM*) (§3.4) preserves cross-node continuity by reprojecting geometry-supported content from neighboring viewpoints. Together, these modules yield visually diverse and geometrically coherent training trajectories.

Preliminaries. MultiDiffusion Bar-Tal et al. (2023) extends a pretrained diffusion model Γ to panoramic generation through a composite denoising operator Ω. Let $\mathcal{I} = \mathbb{R}^{H\times W\times C}$ denote the latent space of a local patch, and let $\mathcal{J} = \mathbb{R}^{H_p\times W'\times C}$ denote the panoramic latent space, where H_p and W' are the panoramic height and width, respectively. In the horizontal full-height setting considered in this work, $H_p = H$ and $W' \geq W$. Throughout the method, r indexes a diffusion step, t indexes a trajectory viewpoint, and k indexes a synthesized trajectory variant. In the aggregation below, i and j index crops within one panorama. Unless otherwise stated, H, W, W', and C denote latent-space dimensions; RGB image dimensions are denoted by H_{rgb} and W_{rgb}.

Let Γ_r denote the pretrained one-step denoising operator at diffusion step r. At this step, Ω updates the panoramic latent as $\boldsymbol{J}_{r-1} = \Omega(\boldsymbol{J}_r, \boldsymbol{\delta})$, where $\boldsymbol{J}_r \in \mathcal{J}$ and $\boldsymbol{\delta}$ denotes the text condition. The dependence of Ω on r is left implicit. The connection between Ω and Γ_r is established through N patch-extraction mappings $F_i : \mathcal{J} \to \mathcal{I}$ and their corresponding zero-padded placement operators $E_i : \mathcal{I} \to \mathcal{J}$, where $i \in \{1, \ldots, N\}$. For horizontal crops with stride ω, the regular crop grid contains $N = \lfloor (W' - W)/\omega \rfloor + 1$ patches. In our implementation, the integer stride satisfies $0 < \omega \leq W$ and $\omega \mid (W' - W)$, so the crops cover the panoramic canvas and the final crop is aligned with its right boundary. At each denoising step, Ω computes a global latent that best matches the locally denoised patches under a location-wise least-squares criterion. Its closed-form aggregation is

$$\Omega(\boldsymbol{J}_r, \boldsymbol{\delta}) = \sum_{i=1}^{N} \big[E_i(\mathbf{1}) \oslash \big(\sum_{j=1}^{N} E_j(\mathbf{1})\big)\big] \odot E_i\big(\Gamma_r\big(F_i(\boldsymbol{J}_r), \boldsymbol{\delta}\big)\big), \tag{1}$$

where $\mathbf{1}$ denotes an all-ones tensor in $\mathcal{I}$, and $\oslash$ and $\odot$ denote element-wise division and multiplication, respectively. The placement operator E_i embeds the i-th denoised patch into its corresponding location in the panoramic latent space; it is not the algebraic inverse of F_i. The crop arrangement covers every position in the panoramic latent, ensuring that the denominator is nonzero at each location. This aggregation extends the pretrained text-to-image diffusion model to text-conditioned panoramic generation without updating its parameters.

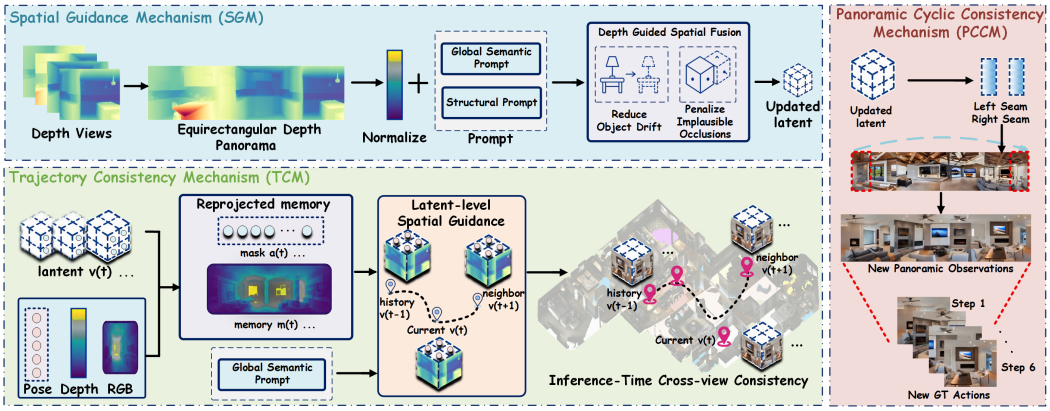


Figure 3: **Illustration of the three complementary components of SCPGen**: *SGM* (§3.2), *PCCM* (§3.3), and *TCM* (§3.4). *SGM* uses depth priors to improve the local geometric plausibility of each viewpoint, *PCCM* couples the left and right panoramic boundaries to preserve cyclic continuity, and *TCM* reprojects geometry-supported content from historical and neighboring viewpoints to maintain cross-node visual consistency.

3.2 Spatial Guidance Mechanism

To reduce depth-related spatial inconsistencies, we explicitly condition panorama generation on structural depth priors. We use the original R2R depth maps aligned with the corresponding Matterport3D panoramic observations as geometric priors. The same depth representation is used by *SGM* and *TCM*. Given the depth patches and their crop offsets $\{s_i\}$, our goal is to generate geometrically consistent RGB patches $\{\boldsymbol{X}_i\}_{i=1}^{N}$. A straightforward solution independently samples each latent from a depth-conditioned diffusion model (*e.g.*, ControlNet Zhang et al. (2023)) and decodes it using a pretrained VAE Kingma & Welling (2014). However, independently initialized noises often produce inconsistent appearance and geometry in overlapping regions. For clarity, we describe *SGM* independently below; its integration with *PCCM* and *TCM* in the complete SCPGen generation pipeline is detailed in Appendix §C.

Multi-Patch Consistency. Rather than denoising the generated patches independently, we treat their latent representations as overlapping crops of a shared canonical latent canvas. At each denoising step, the provisional patch predictions are mapped into the canonical latent space and fused across their overlapping regions using the MultiDiffusion aggregation operator (§3.1). The resulting canonical latent representation is subsequently projected back to the individual patches, enabling information exchange across neighboring views throughout the generation process. This collective denoising procedure encourages the decoded RGB patches to form a coherent panoramic scene rather than a collection of independently synthesized observations.

Depth-Weighted Overlap Guidance. Although shared-latent aggregation couples overlapping crops, their separately decoded predictions may still differ in the overlap. We therefore introduce a depth-weighted overlap loss that places greater emphasis on nearer regions. Let $\boldsymbol{Z}_{r,i} = F_i(\boldsymbol{J}_r)$ denote the latent crop extracted from the shared panoramic canvas at diffusion step r. We estimate the clean latent as $\hat{\boldsymbol{Z}}_{0,i}^{(r)} = f_{\theta,r}(\boldsymbol{Z}_{r,i})$ and decode it as $\hat{\boldsymbol{X}}_{0,i}^{(r)} = g(\hat{\boldsymbol{Z}}_{0,i}^{(r)})$. For an overlapping patch pair (i,j), let s_i and s_j denote their horizontal offsets on the shared panorama. A pixel $\boldsymbol{p} = (u,v)$ in patch i corresponds to $\boldsymbol{p}' = (u + s_i - s_j, v)$ in patch j. We define $\mathcal{V}_{ij}$ as the set of pixels for which both $\boldsymbol{p}$ and $\boldsymbol{p}'$ lie within their respective patches.

$$\mathcal{L}_{\text{ov}}^{\text{depth}} = \sum_{(i,j)\in\mathcal{E}} \sum_{\boldsymbol{p}\in\mathcal{V}_{ij}} w_{ij}(\boldsymbol{p}) \left\| \hat{\boldsymbol{X}}_{0,i}(\boldsymbol{p}) - \hat{\boldsymbol{X}}_{0,j}(\boldsymbol{p}') \right\|_1, \tag{2}$$

where $\mathcal{E}$ denotes the set of overlapping patch pairs and $\mathcal{V}_{ij}$ contains their valid correspondences. For a pixel $\boldsymbol{p}$ in patch i and its corresponding pixel $\boldsymbol{p}'$ in patch j, the depth-aware weight is defined as

$$w_{ij}(\boldsymbol{p}) = \exp\left[-\frac{\alpha}{2}\left(\boldsymbol{D}_i(\boldsymbol{p}) + \boldsymbol{D}_j(\boldsymbol{p}')\right)\right], \tag{3}$$

where α controls the preference for closer surfaces. During inference, the patch latents are jointly refined by minimizing $\mathcal{L}_{\text{ov}}^{\text{depth}}$ after each denoising prediction. This depth-weighted overlap loss emphasizes the appearance consistency of near-field regions, while depth conditioning guides the local scene layout.

3.3 PANORAMIC CYCLIC CONSISTENCY MECHANISM

Existing diffusion-based VLN augmentation pipelines typically synthesize panoramas on a planar canvas. Their patch extraction and aggregation operations enforce consistency only within the finite image domain, while treating the first and last panorama columns as unrelated boundaries. Thus, the wraparound adjacency of equirectangular projections remains unconstrained, and directly generated or assembled views may lack visual and semantic continuity across the $0°/360°$ seam Li & Bansal (2023b); Bar-Tal et al. (2023); Wu et al. (2024); Zhang et al. (2024); Wang et al. (2024a).

We address this limitation with the training-free *Panoramic Cyclic Consistency Mechanism (PCCM)*. Let $\boldsymbol{U}_r \in \mathbb{R}^{H\times W\times C}$ denote the target-width latent panorama, and let $\boldsymbol{J}_r \in \mathcal{J} = \mathbb{R}^{H\times W'\times C}$ denote its extended-canvas representation. The target panorama is embedded into the central interval $[a, a+W)$, where $a = (W'-W)/2$, such that $C_{\mathrm{ctr}}(\boldsymbol{J}_r) = \boldsymbol{U}_r$. We set $W' = 2W$ and construct two cyclically shifted patches using $q_1 = W/4$ and $q_2 = 3W/4$. We assume that $4 \mid W$ so that q_1 and q_2 are integer latent coordinates. Specifically, for $j \in \{1, 2\}$, the cyclic extraction operator $G_j : \mathcal{J} \to \mathcal{I}$ satisfies $[G_j(\boldsymbol{J}_r)]_{v,u,c} = (\boldsymbol{J}_r)_{v,a+\langle u+q_j\rangle_W,c}$, where $\langle\cdot\rangle_W$ denotes modulo-W indexing. The corresponding placement operator $P_j : \mathcal{I} \to \mathcal{J}$ writes a denoised cyclic patch into the central interval: $[P_j(\boldsymbol{Y})]_{v,x,c} = Y_{v,\langle x-a-q_j\rangle_W,c}, \quad a \le x < a+W$, and returns zero elsewhere. At diffusion step r, the regular and cyclic predictions are jointly aggregated as

$$\boldsymbol{J}_{r-1} = \big[\sum_{i=1}^{N} E_i(\Gamma_r(F_i(\boldsymbol{J}_r), \boldsymbol{\delta})) + \sum_{j=1}^{2} P_j(\Gamma_r(G_j(\boldsymbol{J}_r), \boldsymbol{\delta}))\big] \oslash \boldsymbol{\Lambda}, \tag{4}$$

where $\boldsymbol{\Lambda} = \sum_{i=1}^{N} E_i(\mathbf{1}) + \sum_{j=1}^{2} P_j(\mathbf{1})$ normalizes the overlapping predictions element-wise. After reverse diffusion, the final target-width latent is $\boldsymbol{U}_{\mathrm{out}} = C_{\mathrm{ctr}}(\boldsymbol{J}_0)$, and the RGB panorama is decoded as $\boldsymbol{X}_{\mathrm{syn}} = g(\boldsymbol{U}_{\mathrm{out}})$.

3.4 TRAJECTORY CONSISTENCY MECHANISM

To encourage cross-node consistency, *TCM* conditions each panorama on preceding trajectory observations and their relative camera geometry Liu et al. (2023b); Yu et al. (2023); Liu et al. (2024b); Shi et al. (2024). Within the patch-based operators, i and j index crops inside one panorama, whereas s and t denote source and target trajectory viewpoints, respectively. Let $\tau = (v_0, \ldots, v_{L-1})$ denote an ordered trajectory, and let $\mathbf{T}_i = [\mathbf{R}_i \mid \boldsymbol{t}_i] \in SE(3)$ denote the camera-to-world pose at v_i. Given a scene description c_i^k, independent generation yields $\boldsymbol{X}_i^k = g(\boldsymbol{Z}_{0,i}^k)$, where $\boldsymbol{Z}_{0,i}^k \sim p_\theta(\boldsymbol{Z}_0 \mid c_i^k)$. For a 3D point $\boldsymbol{q}$ visible from source and target viewpoints v_s and v_t, respectively, we encourage

$$\boldsymbol{X}_s^k[\pi_s(\boldsymbol{q})] \approx \boldsymbol{X}_t^k[\pi_t(\boldsymbol{q})]. \tag{5}$$

Geometry-Supported Reprojected Memory. For target viewpoint v_t, we consider the preceding trajectory viewpoints $\mathcal{N}_t^{K_{\mathrm{traj}}} = \{s \mid \max(0, t-K_{\mathrm{traj}}) \le s < t\}$. Let $\Pi_{\mathrm{ERP},s}^{-1} : \mathbb{R}^2 \times \mathbb{R}_+ \to \mathbb{R}^3$ back-project an ERP pixel and its radial depth into the source camera coordinate system. For a source pixel $\boldsymbol{p}_s$ with valid depth, define $\boldsymbol{q}_s(\boldsymbol{p}_s) = \Pi_{\mathrm{ERP},s}^{-1}\big(\boldsymbol{p}_s, d_s(\boldsymbol{p}_s)\big), \quad \boldsymbol{q}_{s\to t}(\boldsymbol{p}_s) = \boldsymbol{R}_t^{\top}\left(\boldsymbol{R}_s\boldsymbol{q}_s(\boldsymbol{p}_s) + \boldsymbol{t}_s - \boldsymbol{t}_t\right)$, and $\boldsymbol{p} = \Pi_{\mathrm{ERP},t}\big(\boldsymbol{q}_{s\to t}(\boldsymbol{p}_s)\big)$. Horizontal ERP coordinates are wrapped modulo the panorama width. For a target pixel $\boldsymbol{p}$, if multiple source pixels project to the same target pixel, we retain the source pixel with the smallest target-camera radial distance. Denote the retained source pixel by $\boldsymbol{p}_s^\star(\boldsymbol{p})$: $\boldsymbol{p}_s^\star(\boldsymbol{p}) \in \arg\min_{\boldsymbol{p}_s:\Pi_{\mathrm{ERP},t}(\boldsymbol{q}_{s\to t}(\boldsymbol{p}_s))=\boldsymbol{p}} \|\boldsymbol{q}_{s\to t}(\boldsymbol{p}_s)\|_2$. Let $\mathcal{V}_{s\to t} \subseteq \mathcal{P}_t$ contain target pixels with a valid projected source point and valid target depth. For $\boldsymbol{p} \in \mathcal{V}_{s\to t}$, define $\hat{d}_{s\to t}(\boldsymbol{p}) = \big\|\boldsymbol{q}_{s\to t}\big(\boldsymbol{p}_s^\star(\boldsymbol{p})\big)\big\|_2$. The correspondence confidence is

$$w_{s\to t}(\boldsymbol{p}) = \begin{cases} \exp\left(-\dfrac{\delta_{s\to t}(\boldsymbol{p})}{\sigma_d} - \dfrac{b_{st}}{\sigma_b}\right), & \boldsymbol{p} \in \mathcal{V}_{s\to t} \text{ and } \delta_{s\to t}(\boldsymbol{p}) \le \epsilon_t(\boldsymbol{p}), \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

The valid coverage of source s is $\rho_{s\to t} = |\{\boldsymbol{p} \in \mathcal{P}_t : w_{s\to t}(\boldsymbol{p}) > w_{\min}\}|/|\mathcal{P}_t|$. After discarding sources below the coverage threshold, we select, at each target pixel, the remaining source with the highest correspondence confidence. Its reprojected color defines $\boldsymbol{M}_t^k(\boldsymbol{p})$, and its confidence defines $\boldsymbol{A}_t(\boldsymbol{p})$. Where no source is accepted, we set $\boldsymbol{A}_t(\boldsymbol{p}) = 0$.

Latent-Space Memory Guidance. We encode the reprojected memory with the panorama VAE and resize its confidence mask to the latent resolution. At denoising step r, we perturb the memory latent to the corresponding scheduler noise level and blend it with the current latent:



Figure 4: **Illustration of the original and SCPGen-generated R2R trajectory data** (§4.2). The left and right panels present the original sample and its SCPGen-generated counterpart. Red boxes localize visual landmarks referred to by the descriptions or instructions, while the yellow boxes connect corresponding regions across the starting panorama and trajectory observations. Matching text colors indicate cross-modal grounding.

$$\widetilde{\boldsymbol{Z}}_t^{k,(r)} = (1 - a_r \boldsymbol{A}_{t,\text{latent}}) \odot \boldsymbol{Z}_t^{k,(r)} + a_r \boldsymbol{A}_{t,\text{latent}} \odot \boldsymbol{Z}_{m,t}^{k,(r)}, \tag{7}$$

where $a_r \in [0, 1]$ controls the guidance strength. Thus, *SGM* constrains local geometry within each viewpoint, *PCCM* couples the left and right boundaries of each panorama, and *TCM* transfers geometry-supported content across successive trajectory viewpoints. Implementation details are provided in Appendix §C.

4 Experiments

4.1 Experimental Setup

Benchmarks and Metrics. We evaluate SCPGen on R2R Anderson et al. (2018), R4R Jain et al. (2019), REVERIE Qi et al. (2020), and R2R-CE Krantz et al. (2020), covering fine-grained, long-horizon, object-grounding, and transfer to continuous navigation, respectively. We report TL, NE, SR, and SPL on R2R; NE, SR, nDTW and SDTW on R4R; SR, OSR, SPL, RGS, and RGSPL on REVERIE; and NE, OSR, and SR on R2R-CE. All evaluations emphasize unseen-environment generalization. Further benchmark and metric descriptions are provided in Appendix §B.1.

Implementation Details. We adopt DUET Chen et al. (2022b) as the navigation backbone and generate SCPGen trajectories offline without modifying its architecture or inference procedure. *SGM*, *PCCM*, and *TCM* enforce local geometry, panoramic boundary continuity, and cross-node visual consistency, respectively. The pretrained diffusion and depth-estimation models remain frozen. Generated trajectories augment the pretraining and navigation training; for R2R-CE, augmented discrete trajectories are used for pretraining before fine-tuning in the continuous environment. The controlled GPT-5 Nano ablations in §4.3 share the same backbone, text-generation protocol, optimization schedule, and augmentation budget. Reproducibility materials, experimental settings, and mechanism integration details are provided in Appendices §A, §B, and §C, respectively. After acceptance, we will release the code, configurations, and evaluation scripts, together with generated data and checkpoints where permitted by the applicable licenses.

4.2 Comparison with Existing Approaches

Qualitative Results. Fig. 4 compares an original R2R trajectory with its SCPGen-generated counterpart. Objects introduced by the rewritten description are grounded in a coherent panoramic layout,

Table 1: **Quantitative evaluation** on R2R Anderson et al. (2018) and R4R Jain et al. (2019) (§4.2). * denotes methods using additional 3D environments; † denotes results obtained with different backbones or augmentation budgets.

Method	R2R Val Unseen				R4R Val Unseen			
	TL (m)	NE(m)↓	SR(%)↑	SPL(%)↑	NE(m)↓	SR(%)↑	nDTW(%)↑	SDTW(%)↑
ScaleVLN*† [ICCV23]	14.09	2.09	81.00	70.00	–	–	–	–
HAMT† [NeurIPS21]	11.46	**2.29**	66.00	61.00	6.09	44.60	50.30	31.80
REM† [ICCV21]	12.44	3.89	64.00	58.00	6.21	46.00	46.30	22.70
RecBERT† [CVPR21]	12.01	3.93	63.00	57.00	6.67	43.60	45.10	29.90
BSG† [ICCV23]	14.90	2.89	74.00	62.00	6.12	47.00	53.00	**34.00**
PanoGen† [NeurIPS23]	13.40	3.03	74.20	64.30	6.02	47.78	–	–
VER† [CVPR24]	14.83	2.80	76.00	65.00	6.10	47.00	**54.00**	33.00
WCGEN† [arXiv24]	12.57	2.89	75.00	65.80	5.89	48.50	–	–
DUET [CVPR22]	13.94	3.31	72.00	60.00	5.88	50.35	40.08	27.91
RAM [TNNLS25]	13.39	2.66	76.29	66.39	5.18	55.28	42.05	29.91
SCPGen (Ours)	14.36	2.54	**77.62**	**67.20**	**4.83**	**56.31**	42.60	30.15

Table 2: **Quantitative comparison** on the REVERIE Val Unseen split Qi et al. (2020) (§4.2). *: uses additional 3D environments.

Method	SR↑	OSR↑	SPL↑	RGS↑	RGSPL↑
AutoVLN*	55.89	62.14	40.85	36.58	26.76
ScaleVLN*	56.97	63.85	41.84	35.76	26.05
YouTube-VLN	48.11	53.71	34.43	32.15	23.43
PanoGen	51.18	–	**34.99**	33.26	22.99
VLN⟳BERT	30.67	35.02	24.90	18.77	15.27
WCGEN	–	52.23	34.33	31.82	22.33
DUET	48.71	53.62	34.26	32.18	22.64
RAM	51.89	58.47	35.00	34.31	23.20
SCPGen	**52.13**	**58.87**	34.37	**34.94**	**24.91**

Table 3: **Quantitative comparison** on the R2R-CE Val Unseen split Krantz et al. (2020) (§4.2). *: uses additional 3D environments.

Method	NE↓	OSR↑	SR↑
HNR	4.42	67.00	61.00
g3D-LF	4.53	68.00	61.00
ScaleVLN*	4.80	–	55.00
Sim2Sim	6.07	52.00	43.00
DUET	5.14	59.65	37.25
RAM	4.95	61.45	44.15
SCPGen	**4.83**	**61.91**	**46.17**

illustrating the geometric plausibility and cyclic continuity promoted by *SGM* and *PCCM*. Along the trajectory, the referenced landmarks – *wooden round table*, *wood-paneled corridor*, *grey seats*, and *framed photo* – appear sequentially at Steps 1–4. This alignment illustrates how *TCM* preserves cross-node visual correspondence, producing coherent, action-compatible trajectories rather than independent views.

Comparison on R2R and R4R. SCPGen improves navigation performance over DUET-based baselines through geometry-guided, cyclically coupled, and history-aware augmentation. On R2R Val Unseen, it increases SR and SPL by 5.62 and 7.20 percentage points over DUET, respectively, and retains gains of 1.33 and 0.81 points over RAM. The reduction in NE from 3.31 m to 2.54 m is consistent with *SGM*'s depth-guided structural constraints, while the SR and SPL gains align with *PCCM*'s boundary coupling and *TCM*'s landmark-preserving memory, which target seam ambiguity and cross-node appearance changes. On R4R Val Unseen, SCPGen improves SR by 5.96 points over DUET and, relative to RAM, reduces NE by 0.35 m and increases SR by 1.03 points. The smaller nDTW and SDTW gains of 0.55 and 0.24 points are consistent with more limited benefits to reference-path adherence from stable cross-node visual cues. Overall, SCPGen provides a more reliable balance between reaching the destination and following a spatially coherent trajectory.

Comparison on REVERIE. As shown in Table 2, SCPGen achieves 52.13% SR, 58.87% OSR, 34.37% SPL, 34.94% RGS, and 24.91% RGSPL on Val Unseen, improving all five metrics over DUET. Relative to RAM, it increases SR, OSR, RGS, and RGSPL by 0.24, 0.40, 0.63, and 1.71 percentage points, respectively, while SPL decreases by 0.63 points. This pattern supports the downstream utility of combining depth guidance, cyclic boundary coupling, and cross-node memory for navigation and remote grounding, without uniform gains in path efficiency. SCPGen also exceeds PanoGen on SR, RGS, and RGSPL, although methods using additional 3D environments, including AutoVLN and ScaleVLN, retain stronger overall performance.

Table 4: **Ablation of SCPGen data-generation configurations on R2R** Anderson et al. (2018) (§4.3). The GPT-3.5 baseline reproduces the results reported by RAM Wei et al. (2025) and is included for reference rather than as an independently rerun ablation. The LLM columns specify the text generator, while checkmarks indicate the spatial constraints applied during data generation.

LLM		Data Configuration			Val Unseen			Test Unseen		
GPT-3.5	GPT-5Nano	PCCM	SGM	TCM	NE(m)↓	SR(%)↑	SPL(%)↑	NE(m)↓	SR(%)↑	SPL(%)↑
✓	–	–	–	–	2.66	76.29	66.39	3.08	75.00	65.00
–	✓	–	–	–	3.18	71.73	63.05	3.75	70.43	60.36
–	✓	✓	–	–	3.05	73.82	63.41	3.64	70.35	60.93
–	✓	–	✓	–	2.87	74.16	63.74	2.99	72.86	62.88
–	✓	–	–	✓	2.84	73.91	63.77	3.05	72.94	61.31
–	✓	✓	✓	–	2.75	74.84	64.26	3.17	72.40	62.59
–	✓	✓	–	✓	2.83	74.06	63.89	2.94	73.48	63.56
–	✓	–	✓	✓	2.61	76.70	65.24	**2.87**	75.93	65.82
–	✓	✓	✓	✓	**2.54**	**77.62**	**67.20**	2.92	**76.61**	**66.13**

Transfer to Continuous R2R-CE. After pretraining DUET with SCPGen-augmented discrete trajectories and fine-tuning on R2R-CE, SCPGen achieves 4.83 m NE, 61.91% OSR, and 46.17% SR on Val Unseen. Relative to DUET, it reduces NE by 0.31 m and improves OSR and SR by 2.26 and 8.92 percentage points; relative to RAM, the corresponding improvements are 0.12 m, 0.46 points, and 2.02 points. These results support the transfer of geometry-guided panoramic augmentation to continuous navigation through pretraining and task-specific fine-tuning. Although HNR Wang et al. (2024b) and g3D-LF Wang & Lee (2025) report higher OSR and SR, SCPGen improves the DUET backbone through offline augmentation without modifying its architecture or adding generation overhead during navigation inference.

4.3 Ablation Study

All GPT-5 Nano variants share the same backbone, text-generation protocol, augmentation budget, and training settings; the GPT-3.5 RAM result serves only as an external reference. Both *SGM* and *TCM* improve the unconstrained baseline, and their combination achieves the best two-module results, including 2.61/2.87 m NE on Val Unseen/Test Unseen. The complete configuration corresponds to the main SCPGen results, reaching 77.62/76.61% SR and 67.20/66.13% SPL, respectively. Relative to the controlled baseline, it improves SR by 5.89/6.18 and SPL by 4.15/5.77 percentage points. Adding *PCCM* to *SGM+TCM* contributes 0.92/0.68 points in SR and 1.96/0.31 points in SPL, while reducing Val Unseen NE by 0.07 m but increasing Test Unseen NE by 0.05 m. These comparisons support complementary downstream benefits from depth-guided structure and trajectory memory, with additional SR and SPL gains from cyclic boundary coupling but nonuniform effects on NE. Additional ratio ablations are provided in Appendix D.

5 Conclusion

SCPGen generates spatially consistent panoramic observations along navigation trajectories for VLN augmentation. Built on a frozen diffusion model, it combines three complementary constraints: *SGM* uses depth conditioning and depth-weighted overlap guidance to strengthen local structure; *PCCM* couples the panoramic boundaries through cyclic denoising; and *TCM* reprojects geometry-supported content from earlier viewpoints to preserve visual correspondences along a route. These constraints act during data generation without modifying the navigation backbone. Results across four benchmarks show that SCPGen surpasses the reported RAM baseline on key navigation and object-grounding metrics. Controlled R2R ablations suggest that preserving geometry within panoramas and visual correspondence across viewpoints makes synthetic observations more useful for VLN training. The reliability of the generated trajectories nevertheless depends on accurate depth estimates, camera poses, and sufficient overlap between viewpoints. Future work will explore multiview geometric estimation and automatic quality verification to further improve the reliability of generated trajectories.

AI USE STATEMENT

For manuscript preparation, generative AI tools were used for linguistic refinement. For data construction, building upon the augmentation paradigm of RAM Wei et al. (2025), GPT-5 Nano was used to generate textual descriptions of synthetic navigation scenes and their corresponding navigation instructions. All AI-generated content was reviewed by the authors, who take full responsibility for the methodology, experimental results, and final manuscript.

REPRODUCIBILITY STATEMENT

Detailed implementation settings, hyperparameters, prompt templates, data-generation and filtering procedures, dataset splits, and evaluation protocols are provided in the Appendix §A. All experiments were conducted on an NVIDIA RTX PRO 6000 GPU. Our code will be released.

REFERENCES

Dong An, Yuankai Qi, Yangguang Li, Yan Huang, Liang Wang, Tieniu Tan, and Jing Shao. BEVBert: Multimodal Map Pre-training for Language-guided Navigation. In *ICCV*, pp. 2737–2748, 2023.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, pp. 3674–3683, 2018.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *CVPR*, pp. 15791–15801, 2025.

Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. In *ICML*, pp. 1737–1752, 2023.

Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee Wong. Mapgpt: Map-guided prompting with adaptive path planning for vision-and-language navigation. In *ACL*, pp. 9796–9810, 2024.

Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. History aware multimodal transformer for vision-and-language navigation. In *NeurIPS*, pp. 5834–5847, 2021.

Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Learning from unlabeled 3d environments for vision-and-language navigation. In *ECCV*, pp. 638–655, 2022a.

Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Think global, act local: Dual-scale graph transformer for vision-and-language navigation. In *CVPR*, pp. 16537–16547, 2022b.

Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Ali Farhadi, Aniruddha Kembhavi, and Roozbeh Mottaghi. ProcTHOR: Large-Scale Embodied AI Using Procedural Generation. In *NeurIPS*, volume 35, pp. 5982–5994, 2022.

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. In *NeurIPS*, 2018.

Jing Gu, Eliana Stefani, Qi Wu, Jesse Thomason, and Xin Eric Wang. Vision-and-Language Navigation: A Survey of Tasks, Methods, and Future Directions. In *ACL*, pp. 7606–7623, 2022.

Mingfei Han, Liang Ma, Kamila Zhumakhanova, Ekaterina Radionova, Jingyi Zhang, Xiaojun Chang, Xiaodan Liang, and Ivan Laptev. Roomtour3d: Geometry-aware video-instruction tuning for embodied navigation. In *CVPR*, pp. 27586–27596, 2025.

Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. Towards learning a generic agent for vision-and-language navigation via pre-training. In *CVPR*, pp. 13137–13146, 2020.

Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. VLN BERT: A recurrent vision-and-language BERT for navigation. In *CVPR*, pp. 1643–1653, 2021.

Jingyang Huo, Qiang Sun, Boyan Jiang, Haitao Lin, and Yanwei Fu. Geovln: Learning geometry-enhanced visual representation with slot attention for vision-and-language navigation. In *CVPR*, pp. 23212–23221, 2023.

Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. Stay on the Path: Instruction Fidelity in Vision-and-Language Navigation. In *ACL*, pp. 1862–1872, 2019.

Nikolai Kalischek, Michael Oechsle, Fabian Manhardt, Philipp Henzler, Konrad Schindler, and Federico Tombari. Cubediff: Repurposing diffusion-based image models for panorama generation. In *ICLR*, 2025.

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *ICLR*, 2014.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *ECCV*, pp. 104–120, 2020.

Jacob Krantz, Aaron Gokaslan, Dhruv Batra, Stefan Lee, and Oleksandr Maksymets. Waypoint models for instruction-guided navigation in continuous environments. In *ICCV*, pp. 15162–15171, 2021.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *EMNLP*, pp. 4392–4412, 2020.

Jialu Li and Mohit Bansal. Improving vision-and-language navigation by generating future-view image semantics. In *CVPR*, pp. 10803–10812, 2023a.

Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. In *NeurIPS*, pp. 21878–21894, 2023b.

Jialu Li, Hao Tan, and Mohit Bansal. Envedit: Environment editing for vision-and-language navigation. In *CVPR*, pp. 15407–15417, 2022.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *CVPR*, pp. 22511–22521, 2023.

Rui Liu, Xiaohan Wang, Wenguan Wang, and Yi Yang. Bird's-eye-view scene graph for vision-language navigation. In *ICCV*, pp. 10968–10980, 2023a.

Rui Liu, Wenguan Wang, and Yi Yang. Volumetric Environment Representation for Vision-Language Navigation. In *CVPR*, pp. 16317–16328, 2024a.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot One Image to 3D Object. In *ICCV*, pp. 9298–9309, 2023b.

Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. SyncDreamer: Generating Multiview-consistent Images from a Single-view Image. In *ICLR*, 2024b.

Yuxing Long, Xiaoqi Li, Wenzhe Cai, and Hao Dong. Discuss before moving: Visual language navigation via multi-expert discussions. In *ICRA*, pp. 17380–17387. IEEE, 2024.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *AAAI*, volume 38, pp. 4296–4304, 2024.

Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. REVERIE: Remote embodied visual referring expression in real indoor environments. In *CVPR*, pp. 9982–9991, 2020.

Yuankai Qi, Zizheng Pan, Yicong Hong, Ming-Hsuan Yang, Anton van den Hengel, and Qi Wu. The Road To Know-Where: An Object-and-Room Informed Sequential BERT for Indoor Vision-Language Navigation. In *ICCV*, pp. 1655–1664, 2021.

Sonia Raychaudhuri, Saim Wani, Shivansh Patel, Unnat Jain, and Angel Chang. Language-aligned waypoint (law) supervision for vision-and-language navigation in continuous environments. In *EMNLP*, pp. 4018–4028, 2021.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. GEN3C: 3D-informed world-consistent video generation with precise camera control. In *CVPR*, pp. 6121–6132, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pp. 10684–10695, 2022.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied AI research. In *ICCV*, pp. 9339–9347, 2019.

Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. MVDream: Multi-view Diffusion for 3D Generation. In *ICLR*, 2024.

Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. Vision-and-Dialog Navigation. In *CoRL*, pp. 394–406, 2020.

Hai Wang, Xiaoyu Xiang, Yuchen Fan, and Jing-Hao Xue. Customizing 360-Degree Panoramas Through Text-to-Image Diffusion Models. In *WACV*, pp. 4933–4943, 2024a.

Sen Wang, Dongliang Zhou, Liang Xie, Chao Xu, Ye Yan, and Erwei Yin. PanoGen++: Domain-adapted text-guided panoramic environment generation for vision-and-language navigation. *Neural Networks*, 187:107320, 2025. doi: 10.1016/j.neunet.2025.107320.

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *CVPR*, pp. 6629–6638, 2019.

Zihan Wang and Gim Hee Lee. g3D-LF: Generalizable 3D-Language Feature Fields for Embodied Tasks. In *CVPR*, pp. 14191–14202, 2025. doi: 10.1109/CVPR52734.2025.01324.

Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. GridMM: Grid memory map for vision-and-language navigation. In *ICCV*, pp. 15625–15636, 2023.

Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, Junjie Hu, Ming Jiang, and Shuqiang Jiang. Lookahead exploration with neural radiance representation for continuous vision-language navigation. In *CVPR*, pp. 13753–13762, 2024b.

Ziming Wei, Bingqian Lin, Yunshuang Nie, Jiaqi Chen, Shikui Ma, Hang Xu, and Xiaodan Liang. Unseen from Seen: Rewriting Observation-Instruction Using Foundation Models for Augmenting Vision-Language Navigation. *IEEE TNNLS*, 2025.

Tianhao Wu, Chuanxia Zheng, and Tat-Jen Cham. PanoDiffusion: 360-degree Panorama Outpainting via Diffusion. In *ICLR*, 2024.

Karmesh Yadav, Ram Ramrakhya, Santhosh Kumar Ramakrishnan, Theo Gervet, John Turner, Aaron Gokaslan, Noah Maestre, Angel Xuan Chang, Dhruv Batra, Manolis Savva, Alexander William Clegg, and Devendra Singh Chaplot. Habitat-matterport 3d semantics dataset. In *CVPR*, pp. 4927–4936, 2023.

Weicai Ye, Chenhao Ji, Zheng Chen, Junyao Gao, Xiaoshui Huang, Song-Hai Zhang, Wanli Ouyang, Tong He, Cairong Zhao, and Guofeng Zhang. Diffpano: Scalable and consistent text to panorama generation with spherical epipolar-aware diffusion. In *NeurIPS*, pp. 1304–1332, 2024.

Jason J. Yu, Fereshteh Forghani, Konstantinos G. Derpanis, and Marcus A. Brubaker. Long-Term Photometric Consistent Novel View Synthesis with Diffusion Models. In *ICCV*, pp. 7094–7104, 2023.

Zhaohuan Zhan, Lisha Yu, Sijie Yu, and Guang Tan. MC-GPT: Empowering Vision-and-Language Navigation with Memory Map and Reasoning Chains. *arXiv preprint arXiv:2405.10620*, 2024.

Cheng Zhang, Qianyi Wu, Camilo Cruz Gambardella, Xiaoshui Huang, Dinh Phung, Wanli Ouyang, and Jianfei Cai. Taming Stable Diffusion for Text to 360 Panorama Image Generation. In *CVPR*, pp. 6347–6357, 2024.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding Conditional Control to Text-to-Image Diffusion Models. In *ICCV*, pp. 3836–3847, 2023.

Xiaobei Zhao, Xingqi Lyu, and Xiang Li. MDE-AgriVLN: Agricultural Vision-and-Language Navigation with Monocular Depth Estimation. *arXiv preprint arXiv:2512.03958*, 2025.

Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. Towards learning a generalist model for embodied navigation. In *CVPR*, pp. 13624–13634, 2024.

Yu Zhong, Rui Zhang, Zihao Zhang, Shuo Wang, Chuan Fang, Xishan Zhang, Jiaming Guo, Shaohui Peng, Di Huang, Yanyang Yan, Xing Hu, Ping Tan, and Qi Guo. World-Consistent Data Generation for Vision-and-Language Navigation. *arXiv preprint arXiv:2412.06413*, 2024.

Zhengdong Zhu, Weiyi Xue, Zuyuan Yang, Wenlve Zhou, and Zhiheng Zhou. Pano360: Perspective to panoramic vision with geometric consistency. In *CVPR*, pp. 7600–7609, 2026.

Supplementary Material for SCPGen

Supplementary Overview

This supplementary material is organized as follows: §A describes the reproducibility scope, verified experimental artifacts, and planned release. §B reports the software and hardware environment, data-processing protocol, training configuration, and external assets. §C details the integration of *SGM*, *PCCM*, and *TCM*. Additional experiments and qualitative results are presented in §D. Finally, §E discusses the limitations and potential social impact of SCPGen.

A Reproducibility Details

Pipeline and Settings. SCPGen comprises some offline stages: panorama generation, multi-view conversion, visual feature extraction, and VLN training. Their intermediate outputs can be inspected and reused independently. The pretrained diffusion model, VAE, depth estimator, and CLIP encoder remain frozen; generated trajectories are stored before training. SCPGen leaves the DUET architecture unchanged and adds no computation at navigation inference time. The experimental environment, data processing, and training settings are described in §B.

Experimental Artifacts. For feature extraction, we record the input-view organization, CLIP checkpoint, feature schema, output size, and validation results. Each exported record identifies its scan, viewpoint, trajectory, navigation step, and description index, together with its multi-view feature. We validate the resulting TSV by checking record counts, parsing, and feature dimensions. The augmented TSV and original Matterport3D feature HDF5 remain separate training inputs. For each run, we record the initialization checkpoint, random seed, optimizer settings, training schedule, model configuration, and input mapping. The current archive validates feature extraction and records the configuration used to initiate stage-1 training. The release archive will also include training logs, checkpoint-selection records, and evaluation outputs for the reported results. Generation manifests will link each generated corpus to its model checkpoints, prompts, sampling settings, and image-processing configuration.

Release and Data Access. After acceptance, we will release the SCPGen implementation, configurations, feature-extraction utilities, training and evaluation commands, and dependency specifications. Checksums and generation manifests will identify the versions of the data, checkpoints, and code used in each experiment. Matterport3D observations will not be redistributed and must be obtained through the official access procedure. Released code, generated artifacts, and checkpoints will follow the applicable licenses and redistribution conditions of their source datasets and pretrained models.

B Experiment Setting

B.1 Benchmarks, Metrics, and Experimental Environment

Benchmarks and Evaluation Protocols. We evaluate SCPGen on four VLN benchmarks. R2R Anderson et al. (2018) evaluates fine-grained instruction following, R4R Jain et al. (2019) extends R2R with longer trajectories, REVERIE Qi et al. (2020) evaluates remote object grounding, and R2R-CE Krantz et al. (2020) transfers R2R instructions to continuous Habitat Savva et al. (2019) environments. We follow the official evaluation splits and primarily assess generalization to unseen environments. For R2R, we report Trajectory Length (TL), Navigation Error (NE), Success Rate (SR), and Success weighted by Path Length (SPL). For R2R-CE, we report NE, Oracle Success Rate (OSR), and SR. TL measures the executed path length, NE measures the final shortest-path distance to the goal, and SR evaluates whether the agent stops within the benchmark-specific success threshold. SPL further accounts for path efficiency. REVERIE additionally reports Oracle Success Rate (OSR), Remote Grounding Success (RGS), and RGS weighted by Path Length (RGSPL). For R4R, we reportwe report NE, SR, normalized Dynamic Time Warping (nDTW), and Success weighted by nDTW (SDTW) to evaluate fidelity to the reference trajectory.

Table B.1: **Software and hardware environment** used for VLN training (§B.1).

Item	Configuration
Operating system	Ubuntu 22.04.4 LTS
Python	3.10
PyTorch	2.7.1
CUDA	12.8
Training GPU	NVIDIA RTX PRO 6000 Blackwell, 96 GB
Execution mode	Single GPU/process (`world_size`=1)
Random seed	0

Table B.2: **Datasets and pretrained models** used in our experiments (§B.2).

Asset	Source / Version
Matterport3D observations	Matterport3D RGB-D scans
R2R / R4R annotations	Official R2R and R4R releases
REVERIE annotations	Official REVERIE release
R2R-CE environment	VLN-CE with `R2R_VLNCE` v1–3
Text-to-image model	Frozen diffusion model and ControlNet checkpoint
Depth estimator	Depth Anything V2
Visual encoder	CLIP ViT-L/14
Navigation backbone	Official VLN-DUET implementation
SCPGen implementation	To be released

Software and Hardware Environment. SCPGen performs panorama generation and visual feature extraction offline before navigation training. All augmented panoramas and their corresponding features are precomputed and cached, introducing no additional generation overhead during VLN training or inference. We adopt DUET Chen et al. (2022b) as the navigation backbone and retain its original architecture and action space. Unless otherwise specified, all experimental variants are executed under the same software environment and hardware configuration to ensure a controlled comparison. The complete configuration is summarized in Table B.1.

B.2 Data Processing and Training Protocol

Augmented Data Construction. For the recorded depth-controlled R2R experiment, generated panoramas are converted into viewpoint-aligned multi-view observations. CLIP ViT-L/14 is then used to extract 36-view visual features, with a dimensionality of 768 for each view. The resulting corpus contains 4,411 trajectory directories, 26,478 viewpoints, and 69,637 feature records. The extraction procedure reports no incomplete samples or parsing failures and produces a 9.6 GiB TSV file. The augmented TSV is extracted directly from the rendered views rather than converted from an HDF5 file. During training, the TSV supplies the generated observations through the augmentation input, whereas the original Matterport3D HDF5 provides the real R2R observations. These two feature sources are loaded independently, allowing the augmented corpus to be controlled without modifying the original visual features.

Navigation Training. Following DUET, the navigation agent is pretrained for 20k iterations using Masked Language Modeling (MLM), Instruction–Trajectory Matching (ITM), and Single Action Prediction (SAP). Fine-tuning follows DAgger Ross et al. (2011) with a mixing-then-focusing strategy. The recorded depth-controlled stage-1 run uses batch size 8, AdamW, a learning rate of 5×10^{-6}, weight decay of 0.005, and a scheduled length of $200{,}000$ iterations.

The model employs dynamic fusion, nine language layers, four cross-modal layers, two panoramic layers, and full-graph spatial-relation encoding. It is initialized from the specified pretrained DUET checkpoint. Input paths, feature dimensions, and TSV integrity are validated before optimization. All controlled ablations use the same backbone, initialization, optimization schedule, and augmentation budget so that performance differences can be attributed to the evaluated SCPGen components.

C Implementation Details

Algorithm C.1 Pseudo-code of the three-mechanism fusion in SCPGen.

```
# Inputs: trajectory tau, captions C, RGB observations I,
# camera poses T, depth skyboxes, camera poses, and the frozen diffusion model
# Output: generated panoramic trajectory X_gen

D = load_depth_skybox(I)
X_gen, history = [], []

for t in range(len(tau)):
    # Construct geometry-supported trajectory memory
    sources = select_sources(t, T, D, history)
    M_t, A_t = TCM_memory(
        sources, history, D, T
    )

    # Initialize the canonical panorama latent
    Z = initialize_latent(C[t])

    for r in diffusion_schedule:
        # Local geometry and cyclic-boundary constraints
        Q_SGM = SGM_denoise(Z, D[t], C[t])
        Q_pccm = PCCM_denoise(Z, C[t])

        # Aggregate predictions in the canonical space
        Z_bar = aggregate(P_SGM, P_pccm)

        # Preserve reliable cross-node content
        Z = TCM_fuse(Z_bar, M_t, A_t, r)

    X_t = center_crop(vae_decode(Z))
    X_gen.append(X_t)
    history.append(X_t)

return X_gen
```

Fusion of *SGM*, *PCCM*, and *TCM*. SCPGen integrates three complementary constraints into a sequential panoramic generation pipeline. *SGM* constrains local geometry within overlapping views, *PCCM* establishes cyclic continuity between the left and right panoramic boundaries, and *TCM* preserves geometry-supported content across neighboring navigation nodes. All three mechanisms operate on a shared canonical panorama latent and use the same pretrained diffusion model without generator fine-tuning.

Given a trajectory $\tau = (v_0, \ldots, v_{L-1})$, SCPGen generates panoramas in trajectory order. At a target node v_t, it first estimates the target depth prior and retrieves preceding viewpoints from the local topological neighborhood. During each denoising step r, *SGM* applies depth-supported projection to the regular overlapping patches, constraining their local structure and occlusion relationships. In parallel, *PCCM* constructs a cyclic seam block from the two panoramic ends and obtains two auxiliary predictions that couple the left and right boundaries. The regular and cyclic predictions are then mapped to the canonical panorama canvas and normalized over their overlapping regions.

Let $\mathcal{Q}^{k,(r)}_{t,\mathrm{SGM}}$ and $\mathcal{Q}^{k,(r)}_{t,\mathrm{PCCM}}$ denote the depth-guided regular-patch predictions and cyclic seam predictions, respectively. Their normalized aggregation is expressed as

$$\overline{\boldsymbol{Z}}^{k,(r-1)}_t = \mathcal{A}\big(\mathcal{Q}^{k,(r)}_{t,\mathrm{SGM}} \quad \text{and} \quad \mathcal{Q}^{k,(r)}_{t,\mathrm{PCCM}}\cdot\big), \tag{C.1}$$

where $\mathcal{A}$ denotes the extraction, placement, and overlap-normalization operator defined by the MultiDiffusion formulation in the main paper.

After canonical aggregation, *TCM* reprojects reliable content from the selected historical viewpoints to the target view, producing a memory image $\boldsymbol{M}^k_t$ and a soft geometric confidence mask $\boldsymbol{A}_t$. The memory is encoded by the same VAE and perturbed to the noise level of the current denoising step, yielding $\boldsymbol{Z}^{k,(r-1)}_{m,t}$. It is then incorporated into the aggregated panorama latent through masked fusion:

$$\widetilde{\boldsymbol{Z}}^{k,(r-1)}_t = (1 - a_r \boldsymbol{A}_{t,\mathrm{latent}}) \odot \overline{\boldsymbol{Z}}^{k,(r-1)}_t + a_r \boldsymbol{A}_{t,\mathrm{latent}} \odot \boldsymbol{Z}^{k,(r-1)}_{m,t}. \tag{C.2}$$

Here, $\boldsymbol{A}_{t,\mathrm{latent}}$ is the confidence mask resized to the latent resolution, and a_r controls the guidance strength. Consequently, historical observations guide only regions supported by reliable geometry, whereas uncovered regions remain governed by the current scene description. After the final

Table D.1: **Ratio ablation of *SGM* and *PCCM*** on R2R Anderson et al. (2018) (§D.1). The ratio denotes the proportion of augmented trajectories processed by the corresponding mechanism. A checkmark indicates that *SGM* or *PCCM* is activated, while the total augmentation budget is fixed across all variants. Results are reported on the Val Unseen and Test Unseen splits.

Ratio	Mechanism		Val Unseen			Test Unseen		
	PCCM	SGM	NE(m)↓	SR(%)↑	SPL(%)↑	NE(m)↓	SR(%)↑	SPL(%)↑
25%	✓		3.35	71.01	60.93	3.89	69.47	59.04
50%	✓		3.02	71.56	62.52	3.94	69.32	59.38
75%	✓		3.09	72.69	63.21	3.82	70.12	60.17
25%	✓	✓	3.07	72.86	62.81	3.37	71.59	60.50
50%	✓	✓	2.79	74.30	63.55	3.03	72.16	61.58
75%	✓	✓	2.68	75.78	64.91	2.97	73.11	63.13

denoising step, the latent is decoded and center-cropped to obtain the target panorama, which is subsequently added to the trajectory memory for generating the next node. This sequential fusion preserves the distinct role of each mechanism while allowing spatial information to propagate from local geometry to panoramic boundaries and across navigation nodes. Algorithm C.1 summarizes the integration of *SGM*, *PCCM*, and *TCM*. *SGM* and *PCCM* operate within each denoising step, whereas *TCM* provides history-dependent guidance across consecutive navigation nodes.

D More Experiments and Visualization

D.1 Spatial Constraint Coverage and Complementarity

***SGM* and *PCCM* Ratio Ablation.** Table D.1 varies the proportion of augmented trajectories processed by each mechanism while keeping the total augmentation budget and training protocol fixed. For the *PCCM*-only variants, increasing the ratio consistently improves SPL on both splits, while the SR trend is generally positive but not strictly monotonic on Test Unseen. In comparison, jointly applying *SGM* and *PCCM* yields lower NE and higher SR/SPL than the corresponding *PCCM*-only setting at every ratio. The joint variants also improve consistently as the ratio increases, with the 75% configuration achieving the best results: 2.68 m NE, 75.78% SR, and 64.91% SPL on Val Unseen, together with 2.97 m NE, 73.11% SR, and 63.13% SPL on Test Unseen. These results indicate that depth-based local geometry and panoramic cyclic continuity provide complementary supervision, and that applying both mechanisms to a larger portion of the fixed augmentation budget is beneficial.

D.2 Action-Conditioned Trajectory Visualization

Comparison Setup. We conduct a controlled qualitative comparison between RAM Wei et al. (2025) and SCPGen along the same five-step navigation trajectory. Both methods are conditioned on an identical route-level action description:

> *"Starting near the closet, turn right into the hallway and pass the zebra-striped artwork; after entering the living area, turn left and proceed towards the dining room/kitchen, finally stopping next to the refrigerator."*

This example jointly evaluates whether the generated observations preserve instruction-relevant landmarks and whether shared visual content remains consistent across consecutive navigation nodes.

Instruction-Relevant Landmark Preservation. As shown in Fig. D.1, the early observations generated by RAM provide weak correspondence with the action description. At Step 1, the closet that defines the initial turning location is not clearly represented. At Step 2, the zebra-striped artwork used to disambiguate movement through the hallway is also absent or visually ambiguous. Without these landmarks, the generated observations provide insufficient visual evidence for grounding the corresponding action phrases.The difference becomes more pronounced at the endpoint. The instruction requires the agent to stop next to the refrigerator at Step 5. However, the RAM-generated observation does not contain a recognizable refrigerator and exhibits noticeable object displacement

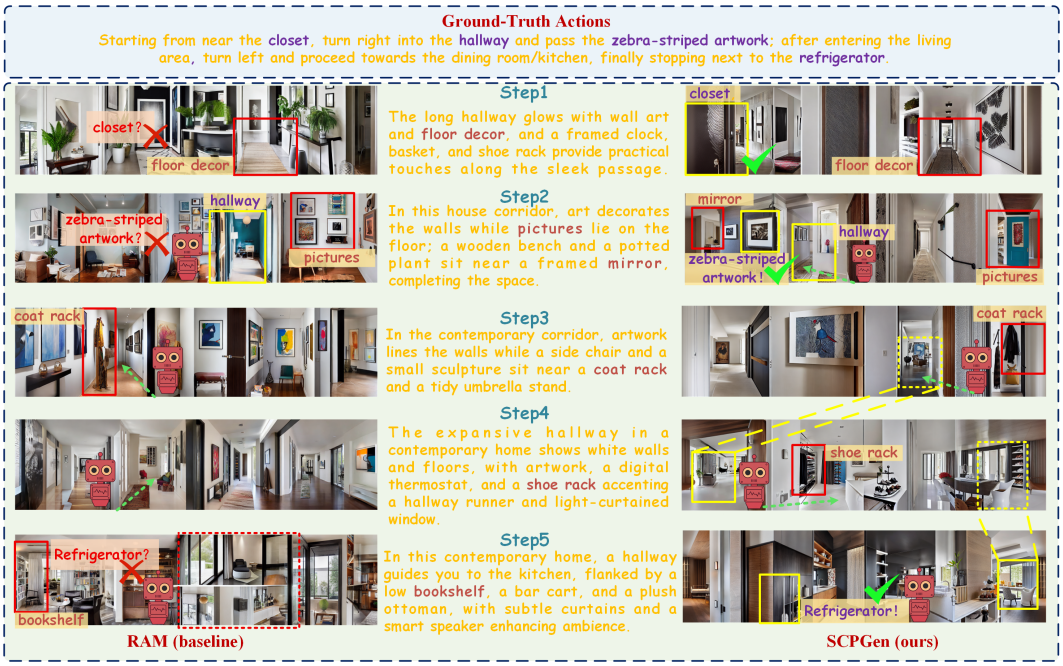


Figure D.1: **Action-conditioned trajectory comparison between RAM and SCPGen** (§D.2). Both methods are conditioned on identical action and scene descriptions, and each row corresponds to one navigation step. Yellow solid boxes highlight instruction-relevant landmarks, including the closet, zebra-striped artwork, and refrigerator. Yellow dashed boxes denote co-visible regions shared across adjacent viewpoints, while red boxes indicate regions associated with the scene descriptions. Green arrows show the agent's direction of movement along the trajectory.

and visual clutter in the highlighted region. Consequently, the final observation is poorly aligned with the instruction-defined destination. SCPGen, by comparison, retains the closet and artwork cues in the intermediate observations and generates a recognizable refrigerator at the final node. The resulting visual sequence therefore provides clearer correspondence between the route description and its navigation landmarks.

Cross-Node Visual Consistency. The yellow dashed annotations further illustrate the effect of *TCM*. The same scene object remains identifiable between Steps 3 and 4 under SCPGen despite the change in viewpoint. Similarly, the white chair and the surrounding window–curtain structure are preserved from Step 4 to Step 5. These correspondences indicate that SCPGen transfers geometry-supported visual content across neighboring nodes rather than synthesizing each panorama as an independent observation. In contrast, the RAM trajectory exhibits noticeable appearance and style variation across successive nodes. Objects may disappear, shift position, or change appearance without a corresponding viewpoint-induced explanation. Several panoramas also contain visible left–right boundary discontinuities. These effects weaken the interpretation of the generated sequence as observations acquired along a single coherent route.

Qualitative Findings. This example illustrates the complementary effects of *SGM*, *PCCM*, and *TCM* on local geometry, panoramic continuity, and cross-node correspondence, respectively. Although not a substitute for quantitative evaluation, it provides qualitative evidence of improved landmark grounding and trajectory coherence.

E Limitation and Social Impact

Limitations. SCPGen depends on reliable geometric priors. *SGM* uses estimated depth for local guidance, while *TCM* requires accurate depth and camera poses, together with sufficient overlap between viewpoints, to reproject historical observations. Errors in these inputs can weaken overlap alignment and cross-node correspondence. Future work will investigate uncertainty-aware depth

guidance, adaptive source-view selection, and multi-view geometric estimation, supported by automatic checks of projection reliability.

Social Impact. SCPGen generates additional panoramic observations at viewpoints in existing annotated environments, potentially reducing the need to collect new visual data for VLN augmentation. Its relevance to assistive and indoor service robotics remains prospective, as our experiments evaluate navigation benchmarks rather than physical robot deployment. Generated observations may reflect biases in pretrained models, while errors in depth estimates or camera poses can reduce their spatial reliability. We therefore recommend documenting the provenance of source assets, screening unreliable samples, and using synthetic data to supplement existing training data rather than replace real-world validation. Matterport3D observations will not be redistributed, and any released artifacts will follow the applicable licenses and redistribution conditions of their source datasets and pretrained models.