MAP-GROUNDED WORLD MODEL FOR URBAN STREET GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

World models have made remarkable progress in generating realistic and interactive environments, yet long-horizon urban generation remains difficult because both scene structure and travel direction must remain consistent as the viewpoint moves. Maps naturally provide these two forms of naturally: a map describes the spatial structure of an environment, while a route provides vectors that define how the viewpoint should move through it. We introduce Vision–Map–Language (VML), a map-grounded framework for urban street generation. Given a map and a route, VML uses the map's structure to determine what should appear around the viewpoint and route vectors to determine where the viewpoint should move next. These signals are translated into coordinated camera motion, language, and visual conditions for a pretrained video generator. VML further projects map structure into the current view to guide generation, allowing the map to remain active throughout long-horizon exploration. We evaluate VML through 3 capabilities: exploring a selected map location, following a specified route, and generating surrounding scenes that reflect the mapped structure along that route. Our results demonstrate that maps can serve as a simple and reusable representation for organizing and navigating generative urban worlds.

1 INTRODUCTION

World models have made rapid progress in generating realistic and interactive environments (Reed et al., 2022; Yu et al., 2025). However, generating a coherent urban world over a prolonged navigation remains challenging. As the viewpoint moves, the model must not only produce realistic images, but also preserve the layout of the environment: roads should lead in the intended direction, turns should occur at the intended locations, and buildings, vegetation, and other scene elements should remain consistent with where the viewer is traveling.

Maps offer a natural way to provide this information (Raychaudhuri & Chang, 2025). They describe the structure of an environment—roads, regions, connectivity, and the spatial relationships between them–while a route provides the direction of travel. In this regard, the map and route provide two complementary forms of information: *structure* context for the scene and *vectors* for movement. In addition, both paper and digital maps can be reprojected to a global coordinate system and remains consistently available no matter how far the viewpoint travels.

This leads to a simple question: how can we leverage maps to generate the world? Rather than asking a generative model to infer the urban streetscape entirely from a large amount of visual observations (Xiong et al., 2024), we exploit the structure already encoded in the map. user can select a location and a route, and the map tells Diffusion Transformer (DiT) Models both where the viewpoint should move and what should appear around it as generation progresses along the selected route..

We introduce Vision-Map-Language (VML), a map-grounded framework for urban streetscape generation. VML uses the map in two complementarysteps. First, a map compiler translates route and scene information into conditioning signals already supported by the pretrained DiT video model (Lian et al., 2024), including camera motion, language, and visual initialization. Second, VML introduces an inference-time map-guided diffusion mechanism that directly corrects the generator's denoising trajectory (Berner et al., 2024; Fishman et al., 2023).

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.

The key idea is to guide the diffusion process with map structure e at inference time, without fine-tuning the video backbone. (Wang et al., 2024a; Singh et al., 2025). At each denoising step, DiT predicts a velocity v_i. VML uses this prediction to form an estimated clean latent, reads its scene structure with a fixed differentiable structural predictor, and compares the predicted structure with a view-aligned target projected from the map. The resulting layout loss is differentiated with respect to the predicted velocity itself, producing a corrected velocity $\tilde{v}_i$ that is used by the sampler. Thus, the generator parameters remain fixed while the sampling dynamics are modified according to the map.

This design separates the roles of the pretrained model and the map. The video world model provides the generative prior for appearance, texture, lighting, architectural detail, and other local visual properties, while the map provides an external structural reference dictating where the viewpoint should move and what coarse scene structure should be visible. Because the map can be queried continuously as the viewpoint advances, VML avoid relying on a single condition supplied at the beginning of the world generation. Instead, camera, language, visual context, and structural targets are refreshed along the route, allowing map guidance to remain active and resilient during autoregressive long-horizon generation (Duan et al., 2025).

We evaluate VML across three capabilities: whether a selected route can be rendered and traversed, whether changes to the map produce corresponding changes in the generated world, and whether map grounding persists as generation proceeds over long autoregressive rollouts. Our experiments include generations of up to 150 seconds and evaluate both route response and scene-structure alignment (Mitra et al., 2025). Our main contributions are:

- A map-based video diffusion framework that transforms any street map into an explorable street-level world.
- Map-projected diffusion guidance, which projects the local map structure into the current camera view and uses it to correct the DiT's denoising velocity during sampling. Route changes control viewpoint motion, while changes to the surrounding map structure can alter the generated scene.
- VML based map guidance throughout long autoregressive journeys. We demonstrate route and scene responses over 150-second generations.

2 Related Work

Spatial Control in Image and Video Generation. A growing body of work has explored how to control the spatial content and motion of generative models. ControlNet introduces image-based spatial conditioning for image generation (Zhang et al., 2023), while GLIGEN enables generation from specified object locations and layouts (Li et al., 2023). Lian et al. (2024) instead ground text-to-image prompts by composing scene layouts with a large language model before diffusion sampling, complementing architectural conditioning with a language-level layout interface. Subject-level spatial control within a single image has also been studied by Wang et al. (2024b), who reposition a depicted subject while preserving its identity. For video generation, CameraCtrl and MotionCtrl provide explicit control over camera and scene motion (He et al., 2025; Wang et al., 2024c), and the Latte latent diffusion transformer (Ma et al., 2025) establishes a DiT backbone for video generation that our frozen generator builds upon. Related work in driving and street-scene generation also uses road layouts, camera trajectories, and other geometric cues to guide synthesis (Hu et al., 2023; Gao et al., 2024), and Shen et al. (2025) couple motion and semantic learning within an end-to-end driving model.

Maps and Spatial Representations. Explicit spatial representations have long been used to describe and navigate physical environments. SLAM builds maps from observations for localization and navigation (Mur-Artal et al., 2015); procedural modeling constructs urban environments from road networks and geometric rules (Parish & Müller, 2001); and NeRF represents scenes in 3D for novel-view synthesis (Mildenhall et al., 2020). Raychaudhuri & Chang (2025) survey how semantic mapping organizes embodied agents' spatial understanding, and Lyu et al. (2024) reconstruct and segment open-world 3D scenes by grouping Gaussians through a 3D-aware memory bank, extending scene representation toward explicit semantic structure. Geospatially guided diffusion has also

begun to connect maps and generation: Xiong et al. (2024) synthesize mixed-view panoramas from geospatial conditioning, showing that geographic priors can steer a diffusion model.

3 METHOD

We introduce an inference-time map guidance mechanism for video diffusion transformer (as shown in Figure 1). We modify the velocity predicted by the pretrained DiT at each denoising step using gradients from a map-projected structural objective. VML combines two complementary mechanisms for controlling the video diffusion model. First, a map compiler converts route and scene information into conditioning signals already understood by the pretrained generator, including camera motion, language, and visual initialization. Second, map-projected velocity guidance modifies the DiT's predicted denoising velocity at inference time according to a differentiable map-alignment objective. VML generates a street-level journey:

$$(\mathcal{M}, \mathcal{R}, \mathcal{S}) \longrightarrow V, \tag{1}$$

where M denotes the semantic map encoding roads, buildings, vegetation, and water bodies; R denotes the selected route, whose local tangent directions specify the direction of viewpoint motion; the appearance description $\mathcal{S}$ specifies visual properties that are not determined by the map, while V denotes the generated street-level world (in video format).

Let z_i denote the noisy video latent at denoising step i, and let c_k denote the standard conditioning for the current video segment k. After classifier-free guidance, the video DiT predicts a velocity $v_i = G_\theta(z_i, c_k)$.

In standard sampling, v_i is passed directly to the sampler to obtain the next latent. VML instead inserts a map-guided correction between the DiT prediction and the sampler:

$$z_i \xrightarrow{G_\theta} v_i \xrightarrow{\text{map guidance}} \tilde{v}_i \xrightarrow{\text{sampler}} z_{i-1}. \tag{2}$$

where z_i denote the noisy video latent at denoising step i, z_{i-1} denotes the latent after the sampler update, and c_k denote the standard conditioning for video segment k. The frozen video DiT, denoted by G_θ with parameters θ, predicts a velocity $v_i = G_\theta(z_i, c_k)$.

This distinction is central to VML. The pretrained video model continues to provide the generative prior, while the map supplies an external spatial objective that steers the sampling trajectory. The native conditioning signals are described in Section 3.2; we next introduce the map-projected structural objective used to harness the sampling trajectory.

A semantic map describes the environment in a global coordinate system, whereas the video generator operates in the current camera view. VML therefore converts the relevant map structure into a

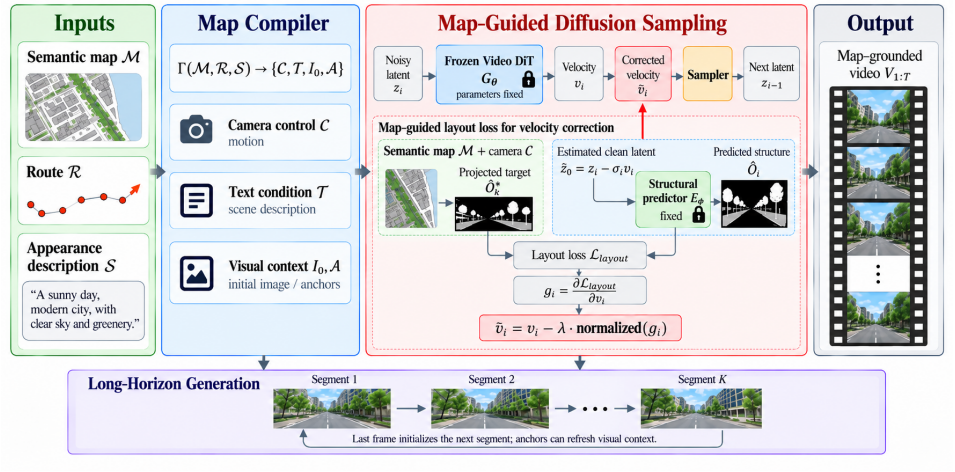

Figure 1: Overview of the VML framework

view-aligned structural target for each segment. Given the current camera trajectory C_k for segment k, we project the semantic map into the corresponding sequence of camera views:

$$O_k^* = \Pi(\mathcal{M}, C_k), \tag{3}$$

where Π denotes the map-to-view projection defined by the camera model, the presumed building height, and spatial smoothing. The resulting tensor O_k^* specifies where mapped scene structures should appear from the current viewpoint.

The map target alone is not sufficient because the current diffusion latent does not directly represent semantic occupancy. We therefore use a lightweight structural predictor E_ϕ that acts as a differentiable readout from the video latent to a semantic occupancy space. The predictor contains approximately 0.375M parameters, is trained separately using SegFormer-derived labels, and remains fixed during video generation. At denoising step i, the DiT produces velocity v_i. We first construct an estimated clean latent

$$\hat{z}_0 = z_i - \sigma_i v_i, \tag{4}$$

where σ_i is the noise scale associated with step i.

The structural predictor then estimates the scene structure implied by the current denoising direction:

$$\hat{O}_i = E_\phi(\hat{z}_0). \tag{5}$$

where E_ϕ denotes a lightweight structural predictor, ϕ denotes its fixed parameters.

The predictor does not generate the scene or modify the video. Its role is to provide a differentiable mapping from the current diffusion state to a semantic occupancy representation. This allows VML to compare the scene structure implied by the current denoising direction with the structure specified by the map. VML compares these quantities using a weighted occupancy loss:

$$\mathcal{L}_{\text{layout}} = \frac{\left\| \mathbf{W} \odot \left(\hat{O}_i - O_k^* \right) \right\|_2^2}{4B(F-1)HW}, \tag{6}$$

where $\mathbf{W} = (1, 1, 1, 0)$ assigns nonzero weight to road, building, and vegetation and zero weight to sky.

The denominator normalizes the loss over the 4 semantic classes, the batch size B, the spatial height H, the spatial width W, and the $F-1$ generated frames. Frame 0 is excluded from the frame count because it is fixed by the supplied initial image.

Conceptually, $\mathcal{L}_{\text{layout}}$ asks whether the scene structure implied by the current DiT prediction agrees with the structure that should be visible from the map at the current camera pose. If the two structures disagree, the loss provides a gradient for correcting the denoising direction.

3.1 Map-Guided Velocity Correction

The key operation in VML is to convert the map-alignment objective into a correction of the DiT's denoising velocity. At each guided denoising step, the raw velocity prediction v_i is detached from the video-transformer computation graph and treated as the variable with respect to which the structural objective is differentiated. The gradient is propagated through the clean-latent estimate and the fixed structural predictor, but not through the video DiT:

$$g_i = \frac{\partial \mathcal{L}_{\text{layout}}}{\partial v_i}. \tag{7}$$

The gradient g_i indicates how the current DiT velocity should change in order to reduce disagreement between the predicted scene structure and the map-projected target.

A direct gradient step would depend strongly on the relative magnitudes of v_i and g_i. We therefore normalize the correction by their root-mean-square magnitudes:

$$\tilde{v}_i = v_i - \lambda \frac{\max\left(\text{rms}(v_i), \epsilon\right)}{\max\left(\text{rms}(g_i), \epsilon\right)} g_i, \tag{8}$$

$$\mathrm{rms}(x) = \sqrt{\frac{1}{|x|}\sum_j x_j^2} \tag{9}$$

where x_j denotes the j-th element of tensor x and $|x|$ denotes its total number of elements. where $\mathrm{rms}(\cdot)$ denotes the root-mean-square over all tensor elements, λ is the guidance strength, and $\epsilon = 10^{-8}$ is a numerical-stability constant.

The corrected velocity $\tilde{v}_i$, rather than the original DiT prediction v_i, is then passed to the sampler:

$$z_{i-1} = \mathrm{Sampler}\left(z_i, \tilde{v}_i\right). \tag{10}$$

where $\mathrm{Sampler}$ denotes the diffusion sampler and z_{i-1} is the updated latent at the next denoising step.

The RMS normalization allows λ to intuitively control the scale of the structural correction relative to the model's own velocity prediction. If the structural loss produces no gradient, no correction is applied. Guidance is applied only within a configurable denoising interval

$$i \in [i_{\mathrm{start}}, i_{\mathrm{end}}). \tag{11}$$

where i_{start} and i_{end} denote the start and end indices of the guidance window.

3.2 Map Compiler

The map-projected guidance described above provides an explicit structural correction during diffusion sampling. In parallel, VML translates map and route information into conditioning signals that the pretrained video generator already knows how to interpret. We refer to this translation as the *map compiler*. Given an urban map $\mathcal{M}$, a route $\mathcal{R}$, and an appearance description $\mathcal{S}$, the compiler constructs

$$(\mathcal{M}, \mathcal{R}, \mathcal{S}) \xrightarrow{\Gamma_\xi} \{C, T, I_0, \mathcal{A}\}, \tag{12}$$

where C denotes camera control, Γ denotes the map compiler T denotes segment-level text conditions, I_0 is the initial visual condition, and $\mathcal{A}$ denotes optional visual anchors. These signals are delivered through conditioning interfaces already supported by the pretrained generator. The compiler does not update the video-generator parameters.

The native conditioning path and the sampling-time guidance path serve different but complementary roles. Camera, language, and visual conditions communicate map information through pretrained interfaces, whereas the structural objective introduced above directly corrects the denoising velocity. VML uses both mechanisms to specify how the viewpoint should move through the mapped environment.

Route → Camera Control. Let

$$R = \{(x_j, y_j)\}_{j=1}^{N},$$

where (x_j, y_j) denotes the j-th route waypoint and N is the number of waypoints.

We simplify the route at changes in heading and compute the signed change between consecutive route directions. These changes are mapped to the camera actions supported by the pretrained video generator, including forward, forward-left, and forward-right motion. The resulting action sequence is converted into per-frame camera poses through the generator's pretrained camera-control pathway. Thus, the route controls the motion of the viewpoint, while the map-projected structural objective controls whether the generated scene remains compatible with the spatial structure surrounding that viewpoint.

Map Structure → Language. The semantic map determines which scene elements should appear along different portions of the route. For each selected structure, VML records its semantic class, its position relative to the route, and the route interval over which it should be visible. These properties are converted into segment-level descriptions. For example, a building parcel located to the right of the route over a particular interval produces a description specifying buildings on the right side of the view during that portion of the journey. The resulting text conditions provide coarse semantic and view-relative information through the pretrained language-conditioning pathway.

Map Structure → Visual Context. The initial image I_0 establishes the visual appearance at the beginning of the journey and is supplied, generated, or selected to be compatible with the mapped starting location. For long autoregressive sequences, language alone may not reliably introduce or preserve a required scene element. VML therefore allows visual anchors to be inserted at selected route positions. Candidate clips are generated from the corresponding map-derived description and screened for the required spatial arrangement. When a candidate satisfies the selection criterion, a selected frame is used as the initial visual condition of the corresponding segment.

In sum, these components provide four pathways from map information to generation:$R \rightarrow C$ for viewpoint motion; map structure is translated into segment-level text conditions for scene semantics; map-compatible initializations and visual anchors provide visual context; and the map is projected under the current camera trajectory $(M, C_k) \rightarrow O_k^*$ to provide the structural target used during map-guided sampling. The first three pathways operate through native conditioning interfaces of the pretrained video generator, whereas the fourth directly constrains the denoising trajectory.

3.3 Long-Horizon Map-Grounded Generation

A semantic map describes a global environment, whereas the video generator operates on short local segments. VML connects these two scales by generating the journey autoregressively while refreshing the map-derived conditions as the viewpoint advances along the route. Let k index the current video segment. Before generating segment k, VML retrieves the conditions associated with the corresponding route interval. For each segment k, VML refreshes the route-specific camera control, segment-level text condition, visual initialization, and projected structural target.

The camera, language, and visual conditions are obtained from the map compiler in Section 3.2. The structural target is obtained by projecting the map into the current camera view, and the segment is generated using the map-guided velocity correction described above. Note that the guidance mechanism is not redefined at the segment level: each segment simply receives the structural target associated with its current route position. After segment k is generated, its final frame normally provides the visual condition for the next segment:

$$I_{k+1} = V_{last}^{\mathrm{k}}. \tag{13}$$

where I_k denote the visual initialization of segment k, and $V^{(k)}$ denote the generated video for that segment.

If a visual anchor is scheduled at the next route interval, the anchor replaces the carried frame:

$$I_{k+1} = A_{k+1}. \tag{14}$$

where $V_{\mathrm{last}}^{(k)}$ denotes the final frame of $V^{(k)}$, A_{k+1} denotes the visual anchor scheduled for segment $k+1$.

Generation therefore proceeds as

$$V^{(1)} \rightarrow V^{(2)} \rightarrow \cdots \rightarrow V^{(K)}, \tag{15}$$

where K denotes the total number of generated video segments.Throughout this process, the map-derived camera, language, visual, and structural conditions are continuously updated according to the current route position. The generated history provides local visual continuity, while the map remains an external spatial reference throughout the journey rather than acting only as an initial condition.

4 Experiments and Results

Video generation backbone. We use Wan2.2-TI2V-5B with DreamX-World-5B-Cam as the frozen video backbone. Videos are generated at 704×1280, 121 frames per segment, 24 fps, and 20 denoising steps. Long rollouts autoregressively chain the final frame of each segment into the next.

Evaluation protocol. We evaluate three complementary properties of the generated videos. *Route response* measures whether the visual motion follows the route-derived camera trajectory. *Scene realization* evaluates whether selected map-specified structures appear at the intended side or route interval. *Structural alignment* measures agreement between the generated scene structure and the projected map target. For an intended structural target G_i, we report the own-target error e_{own} and the own-minus-cross margin m

$$e_{\text{own}} = \mathcal{L}(V, G_i), \qquad m = e_{\text{own}} - \frac{1}{|\mathcal{D}_i|} \sum_{j \in \mathcal{D}_i} \mathcal{L}(V, G_j), \tag{16}$$

where $\mathcal{D}_i$ denotes the fixed distractor set for the corresponding experiment. Lower e_{own} indicates stronger agreement with the intended projected structure, while a more negative margin indicates stronger preference for the intended target relative to the stated alternatives.

We use two complementary evaluation protocols. In mechanism-level controlled experiments, the camera trajectory, text condition, initial image, random seed, and sampler are matched so that the contribution of map-guided sampling can be isolated. In system-level map-edit experiments, the complete VML compiler is rerun from the edited map, producing the corresponding camera, language, visual, and structural conditions.

4.1 Does Map-Guided Diffusion Improve Structural Alignment?

We first evaluate the central mechanism of VML: whether correcting the DiT velocity during denoising improves agreement between the generated scene structure and the structure projected from the map. We use four procedural urban layouts—S, L, Z, and Meander—and three generation seeds per layout. Within each map–seed group, the initial image, camera trajectory, text condition, diffusion sampler, and random seed are matched. The baseline condition (**C**) uses the map-derived camera and language conditions but performs standard diffusion sampling. The guided condition (**G**) uses exactly the same inputs and additionally applies the map-projected velocity correction introduced in Section 3. This comparison therefore isolates the effect of modifying the DiT velocity during sampling while keeping the pretrained video generator and its native conditioning unchanged. We measure the generated video's agreement with its own projected structural target using occupancy MSE, excluding the supplied first frame of each segment.

Results. Map-guided velocity correction reduces structural error on all four tested layouts. Averaged equally across maps, the occupancy MSE decreases from 0.26723 to 0.14950, corresponding to a **44.1% reduction**. The consistent reduction across S, L, Z, and Meander indicates that the effect is not confined to a single route geometry.

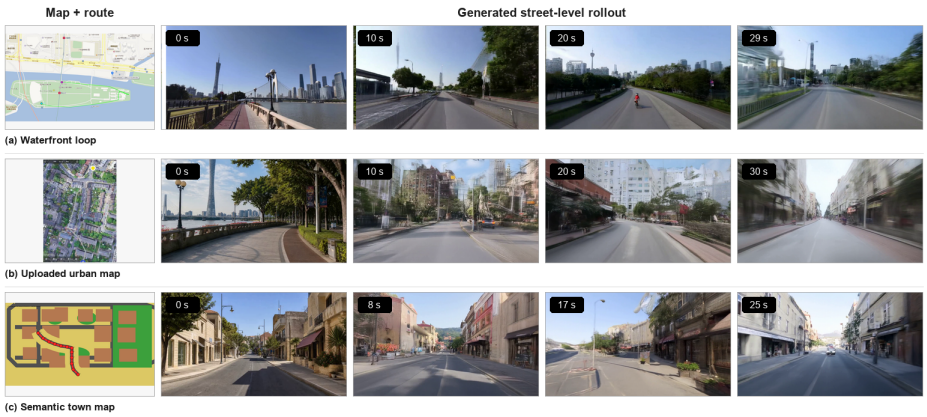

Figure 2: From maps to explorable street-level worlds.Each row shows an input map with a selected route and representative views from the corresponding street-level rollout.

4.2 MAPS TO WORLDS

We first evaluate the end-to-end map-to-world capability of VML. Given an input map and a selected route, VML generates the journey autoregressively while updating the map-derived conditions as the viewpoint advances. Figure 2 shows representative rollouts from two different map-defined environments. The generated views evolve with progress along the route, illustrating how VML turns a map-defined journey into a sequence of street-level views.

For route traversal, the route compiler converts local heading changes into camera controls. Across four seeds in the 30-second winding-route study, all 12 prespecified primary turn contrasts exhibit the expected relative visual response, with a pooled mean displacement of $+68.3$ pixels. The same control remains active over longer generation: in a 150-second rollout, lateral scene flow during planned turns is $2.3\times$ and $3.3\times$ larger than during straight intervals for the two tested route conditions.

VML also uses the surrounding map structure to control scene content along the journey. On route 0002, the fixed full system is evaluated on five selected map-derived scene requirements over fully chained 150-second rollouts. After fixing the configuration, five additional seeds satisfy 17/25 requirements, including 10/10 water-body and 7/15 building cases. Figure 5 shows how different conditioning mechanisms contribute to map-consistent scene generation. On the development seed, text conditioning alone satisfies only 1/5 selected scene requirements. Automatically compiled visual anchors recover both water-related requirements, increasing the score to 3/5. Adding map-projected co-location guidance resolves an additional building requirement, raising the score to 4/5. A map-compatible initialization then resolves the remaining first-segment mismatch, resulting in 5/5. These results indicate that the components play distinct but complementary roles: visual anchors improve the realization of map-specified scene content, while projected guidance and initialization improve spatial consistency.

4.3 LONG HORIZONS

A map-grounded world should remain responsive to the map as generation proceeds, rather than only near initialization. We therefore evaluate VML under autoregressive generation, where each segment starts from the preceding segment's own generated output. Over 30-second rollouts spanning eight routes and four seeds, map-projected guidance reduces mean own-target error from **0.201 to 0.124** and increases own-guide preference from **83/192 to 171/192** matched route–segment positions. Figure 4(a) shows representative examples, where the guided rollout remains more consistent with the projected map targets despite accumulated generated history.

We further extend generation to 30 autoregressive segments ($150\,\mathrm{s}$) on four geometries using a previously unused seed. Guidance reduces mean occupancy MSE from **0.242 to 0.150**, a **37.9%** reduction. As shown in Figure 4(b), the improvement remains visible throughout the rollout, including the final 100–$150\,\mathrm{s}$ interval. These results show that map-projected guidance remains effective under accumulated generated history and continues to influence generation over long autoregressive rollouts.

4.4 ABLATION AND GUIDANCE SENSITIVITY

We evaluate the contributions of visual anchors and map-projected structural guidance using four procedural geometries, three seeds, and four fixed configurations: **C** uses map-derived camera and text conditions, **A** additionally uses visual anchors, **G** adds structural guidance at $\lambda = 0.2$, and **AG** combines both. All configurations use matched start images, camera plans, text conditions, and samplers within each map–seed group. Component analysis further shows that structural guidance is the primary source of occupancy improvement: the mean MSE decreases from 0.267 for the base configuration to 0.150 with guidance, while adding anchors on top of guidance yields 0.149. We use $\lambda = 0.2$, which achieves the lowest own-guide error among the tested settings. Full component and quality diagnostics are reported in Appendix B, and the full sensitivity analysis in Appendix C.

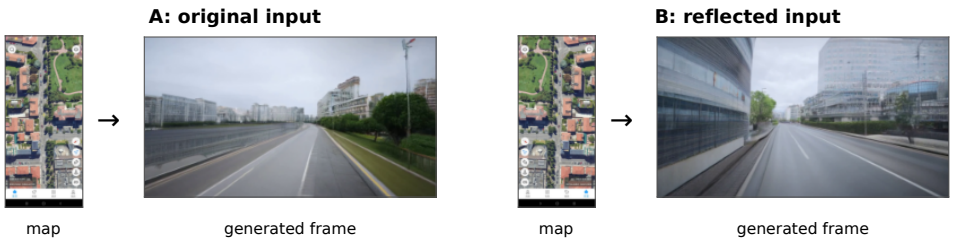

Figure 3: **Change the map, change the generated world.** The original map (World A) and its horizontal reflection (World B) are independently compiled into camera, text, and visual conditions. Matched-seed frames illustrate the induced changes. Across two seeds, route sweep reverses in both pairs and vegetation changes toward the edited target in both pairs, whereas building asymmetry does not reliably reverse. The result exposes a hierarchy of controllability rather than a binary notion of map following.

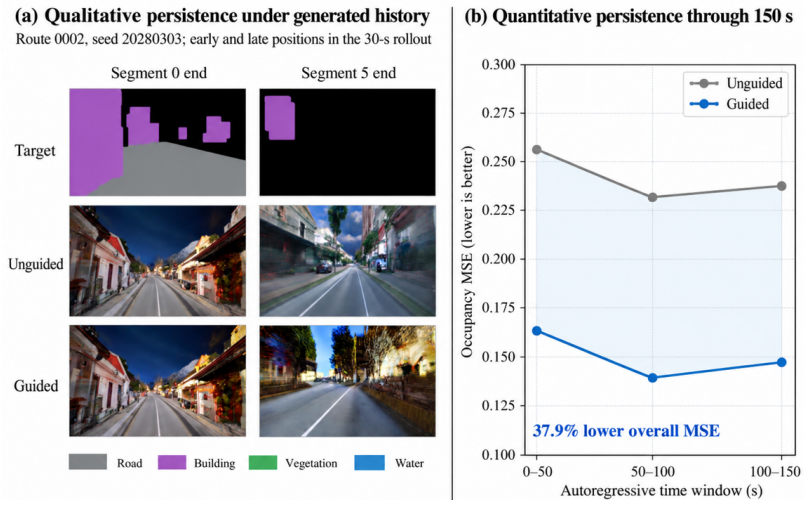

Figure 4: **Map grounding persists under generated history.** **(a)** Early and late examples from a 30-s autoregressive rollout. Guided outputs remain more consistent with the projected map targets despite accumulated generated history. **(b)** Over 150 s, guided rollouts maintain lower occupancy MSE than unguided rollouts across all three time windows, reducing overall MSE from 0.242 to 0.150 (37.9%).

5 Conclusion

We introduced Vision–Map–Language (VML), a map-grounded framework for turning urban maps into explorable street-level worlds. VML uses a map compiler to translate route and scene information into camera, language, and visual conditions, and introduces map-projected diffusion guidance to steer a frozen video generator toward the spatial structure specified by the map. Together, these mechanisms allow the map to serve not only as an initialization, but as a persistent spatial reference throughout generation. Our experiments demonstrate three capabilities. VML generates street-level journeys along map-defined routes while realizing selected surrounding scene content. Map-projected guidance reduces structural alignment error by 44.1% in controlled comparisons, and remains effective when generation proceeds autoregressively from its own outputs. Over 150-second rollouts, guidance continues to improve alignment in the final portion of the journey, with an overall 37.9% reduction in occupancy error. Our results establish that maps provide a compact and reusable interface for organizing both navigation and scene structure in generative urban worlds. Future work should explore exact geometric reconstruction to further bridge the gap between abstract 2D map representations and realistic 3D generation.

AI USE STATEMENT

In this work, we used generative AI tools only for correcting language mistakes (grammar and wording) in the manuscript text. We have not used generative AI tools for any other tasks with required disclosure. We have reviewed and edited all AI-assisted text and take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

A detailed description of the frozen video backbone, the generation protocol, the sampling-time intervention, and the map-derived structural targets is provided in Appendix A.

REFERENCES

Julius Berner, Lorenz Richter, and Karen Ullrich. An optimal control perspective on diffusion-based generative modeling. *Transactions on Machine Learning Research*, 2024.

Xintong Duan, Yutong He, Fahim Tajwar, Wen-Tse Chen, Ruslan Salakhutdinov, and Jeff Schneider. State combinatorial generalization in decision making with conditional diffusion models. *Transactions on Machine Learning Research*, 2025.

Nic Fishman, Leo Klarner, Valentin De Bortoli, Emile Mathieu, and Michael Hutchinson. Diffusion models for constrained domains. *Transactions on Machine Learning Research*, 2023.

Ruiyuan Gao, Kai Chen, Bo Xiao, Guosheng Lin, Lanqing Hong, Zhenguo Li, Wei Zhang, Xihui Liu, and Qiang Xu. MagicDrive: Street view generation with diverse 3d geometry control. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.02601.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. CameraCtrl: Enabling camera control for text-to-video generation. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2404.02101.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. GAIA-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023. URL https://arxiv.org/abs/2309.17080.

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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2301.07093.

Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. LLM-grounded Diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Transactions on Machine Learning Research*, 2024.

Weijie Lyu, Xueting Li, Abhijit Kundu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Gaga: Group any gaussians via 3d-aware memory bank. *Transactions on Machine Learning Research*, 2024.

Xin Ma, Yaohui Wang, Xinyuan Chen, Gengyun Jia, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *Transactions on Machine Learning Research*, 2025.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pp. 405–421, 2020.

Shankhanil Mitra, Diptanu De, Shika Rao, and Rajiv Soundararajan. Image and video quality assessment using prompt-guided latent diffusion models for cross-dataset generalization. *Transactions on Machine Learning Research*, 2025.

Raúl Mur-Artal, José M. M. Montiel, and Juan D. Tardós. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. URL https://arxiv.org/abs/1502.00956.

Yoav I. H. Parish and Pascal Müller. Procedural modeling of cities. In *Proceedings of SIGGRAPH 2001*, 2001. URL https://cgl.ethz.ch/Downloads/Publications/Papers/2001/p_Par01.pdf.

Sonia Raychaudhuri and Angel X. Chang. Semantic mapping in indoor embodied AI – a survey on advances, challenges, and future directions. *Transactions on Machine Learning Research*, 2025.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A generalist agent. *Transactions on Machine Learning Research*, 2022.

Yinzhe Shen, Omer Sahin Tas, Kaiwen Wang, Royden Wagner, and Christoph Stiller. Divide and merge: Motion and semantic learning in end-to-end autonomous driving. *Transactions on Machine Learning Research*, 2025.

Anuj Singh, Sayak Mukherjee, Ahmad Beirami, and Hadi Jamali-Rad. CoDe: Blockwise control for denoising diffusion models. *Transactions on Machine Learning Research*, 2025.

Xi Wang, Nicolas Dufour, Nefeli Andreou, Marie-Paule Cani, Victoria Fernandez Abrevaya, David Picard, and Vicky Kalogeiton. Analysis of classifier-free guidance weight schedulers. *Transactions on Machine Learning Research*, 2024a.

Yikai Wang, Chenjie Cao, Ke Fan, Qiaole Dong, Yifan Li, Xiangyang Xue, and Yanwei Fu. Repositioning the subject within image. *Transactions on Machine Learning Research*, 2024b.

Zhouxia Wang, Ziyang Yuan, Xintao Wang, Tiansong Hu, Yaowei Zhang, Ting Zhang, Ying Shan, Wenwu Zhu, and Ping Luo. MotionCtrl: A unified and flexible motion controller for video generation. *ACM Transactions on Graphics*, 43(4), 2024c. URL https://arxiv.org/abs/2312.03641.

Zhexiao Xiong, Xin Xing, Scott Workman, Subash Khanal, and Nathan Jacobs. Mixed-view panorama synthesis using geospatially guided diffusion. *Transactions on Machine Learning Research*, 2024.

Zhouliang Yu, Yuhuan Yuan, Tim Z. Xiao, Fuxiang Frank Xia, Jie Fu, Ge Zhang, Ge Lin, and Weiyang Liu. Generating symbolic world models via test-time scaling of large language models. *Transactions on Machine Learning Research*, 2025.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE International Conference on Computer Vision*, 2023. URL https://arxiv.org/abs/2302.05543.

A Details of Backbone

VML is built on the Wan2.2-TI2V-5B text-image-to-video architecture, using the DreamX-World-5B-Cam checkpoint as the video transformer backbone. The pretrained generator remains frozen throughout all VML experiments. The video DiT contains 30 transformer blocks with hidden dimension 3072 and 24 attention heads, and operates on the 48-channel latent representation of the Wan2.2 VAE. The VAE compresses the video by a factor of four in time and sixteen in space, while a $(1, 2, 2)$ spatiotemporal patch embedding converts the latent volume into DiT tokens. Text conditions are encoded with UMT5-XXL.

Route-derived camera motion is delivered through the pretrained camera-control pathway. The route compiler converts local route geometry into camera actions and then into per-frame camera poses. Their PRoPE representations are injected through the camera-attention pathway of the DiT blocks. Thus, route motion is communicated through a control interface already learned by the pretrained generator, whereas VML's projected structural guidance modifies the diffusion sampling trajectory directly.

Generation protocol. The principal experiments generate videos at 704×1280 resolution in 121-frame segments at 24 fps, corresponding to approximately five seconds per segment. Each segment is generated with 20 denoising steps, classifier-free guidance scale 3, and bfloat16 inference. The supplied initial image fixes the first latent frame of each segment. After generation, the final decoded frame of segment k normally becomes the visual initialization of segment $k + 1$. Consequently, six successive segments form a nominal 30-second rollout, while thirty segments form a nominal 150-second rollout. Importantly, each experimental arm propagates its own generated history rather than sharing generated prefixes across conditions. Long-horizon results therefore measure control under accumulated model-generated context.

Sampling-time intervention. We apply the velocity correction of Section 3.1 after classifier-free guidance and before the sampler update. The DiT parameters and all generator weights are identical with and without structural guidance; only the velocity passed to the sampler changes.

Map-derived structural targets. At each route position, the map is projected into a view-aligned road, building, and vegetation target with 31 latent-time positions at 44×80 resolution. A fixed 0.375M-parameter convolutional predictor reads the corresponding occupancy from the estimated clean video latent and supplies the guidance signal; it is never updated during generation.

B Full Component Diagnostics

Table 1 reports the component analysis over four fixed configurations, and Table 2 reports the complete auxiliary diagnostics. Anchors alone yield a smaller occupancy reduction than guidance, consistent with their primary role in scene-content realization rather than structural alignment. The auxiliary measurements also reveal trade-offs: anchors increase segment-boundary differences, while guidance reduces sharpness, indicating that improved structural alignment does not necessarily imply improved visual quality or semantic placement; the water-side proxy indicates that anchors shift water content toward the mapped side, whereas the combined configuration slightly reduces it. Figure 5 shows the progressive system-construction study on the development seed referenced in Section 4: text conditioning alone satisfies 1/5 selected scene requirements, compiled visual anchors recover both water-related requirements (3/5), map-projected co-location guidance resolves an additional building requirement (4/5), and a map-compatible initialization resolves the remaining first-segment mismatch (5/5).

Table 1: Fixed-policy component analysis over four maps and three seeds per map.

Variant	Occupancy MSE ↓
C	0.26723
A	0.23957
G	**0.14950**
AG	**0.14902**

Table 2: Fixed-policy component analysis over four maps and three seeds per map (full diagnostics).

Variant	Occupancy MSE ↓	Sharpness vs. C	Boundary difference	Water-side proxy
C	0.26723	1.000×	0.0125	+0.0009
A	0.23957	—	0.1342	+0.0413
G	0.14950	0.750×	0.0112	+0.0043
AG	0.14902	—	0.1267	−0.0322

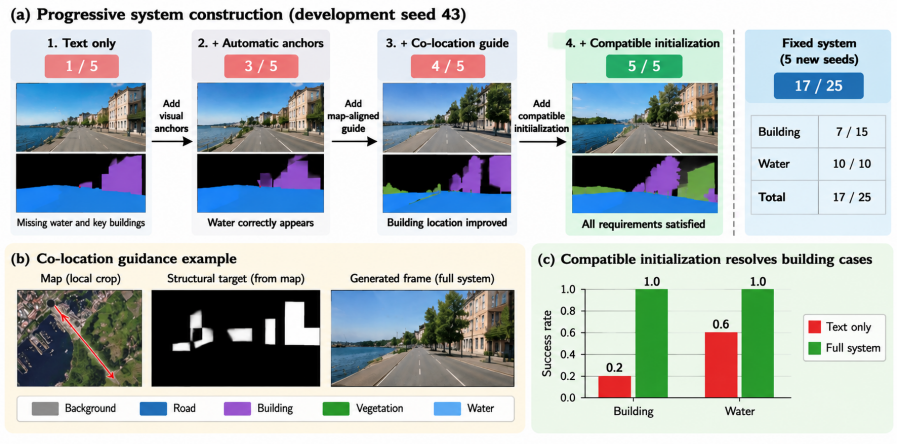

Figure 5: Translating map structure into controllable street-level generation. Visual anchors, map-projected guidance, and compatible initialization provide complementary control over scene content and spatial layout.

C Guidance-Strength Sensitivity

Guidance strength exhibits a non-monotonic response. As shown in Figure 6, own-guide error decreases from 0.250 at $\lambda = 0.02$ to 0.234 at $\lambda = 0.05$, 0.213 at $\lambda = 0.10$, and 0.208 at $\lambda = 0.2$, but rises to 0.247 at $\lambda = 0.4$, where sharpness also decreases. We therefore use $\lambda = 0.2$ as the tested operating point in the main experiments.

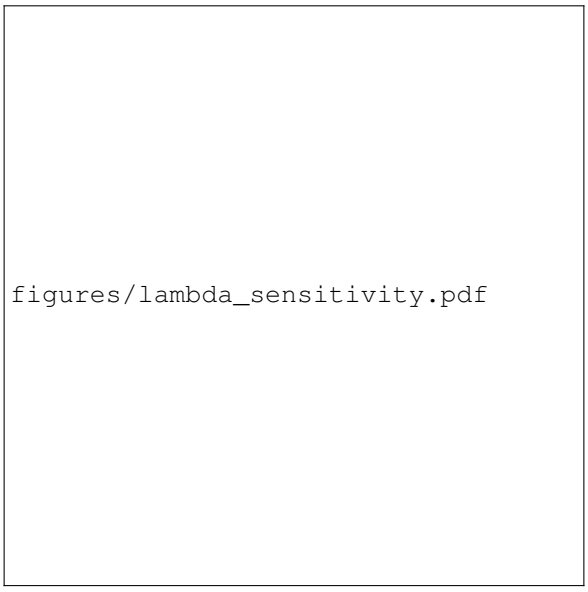

Figure 6: Own-guide error as a function of guidance strength λ. Performance degrades beyond the selected operating point $\lambda = 0.2$.