LayoutBridge: Anisotropic Brownian Bridges for Public Indoor Floorplan Generation

Tongyang Dai[1] daitongyang@bjtu.edu.cn
Ruipeng Gao[*1] rpgao@bjtu.edu.cn
Feng Liu[1] liu.feng@bjtu.edu.cn
Wenqiu Bai[1] wenqiubai@bjtu.edu.cn
Lingkun Li[1] lkli@bjtu.edu.cn
Qing Ni[2] q.ni@lancaster.ac.uk
[1]Beijing Jiaotong University, Beijing, China
[2]Lancaster University, Lancaster, United Kingdom
* Corresponding author

Abstract

Compared with residential layouts, public indoor floorplans present more substantial challenges with open corridors, complex room relations, and non-Manhattan geometries, which do not conform to conventional residential-layout assumptions. Consequently, existing methods are prone to boundary discontinuities, leading to fragmented corridors and irregular wall structures. To address the above issues, we propose *LayoutBridge*, a structure-conditioned generative framework for floorplan production. By moving beyond the Manhattan-geometry assumptions commonly adopted in residential scenarios, LayoutBridge reformulates the task as a latent-space Brownian-bridge diffusion process from structural constraints to target layouts. We have constructed our PISF dataset with 3,517 representative floorplans in public indoor spaces, and LayoutBridge reduces FID by 132.35 and improves BIoU by 15.52% over the latest image-to-image baselines. Regarding the residential benchmarks MSD and RPLAN, it also reduces FID by at least 2.53 compared with state-of-the-art. These results demonstrate its potential for scalable automated architectural design. Code implementing the proposed method is publicly available at http://github.com/lalalalaxxx/LayoutBridge.

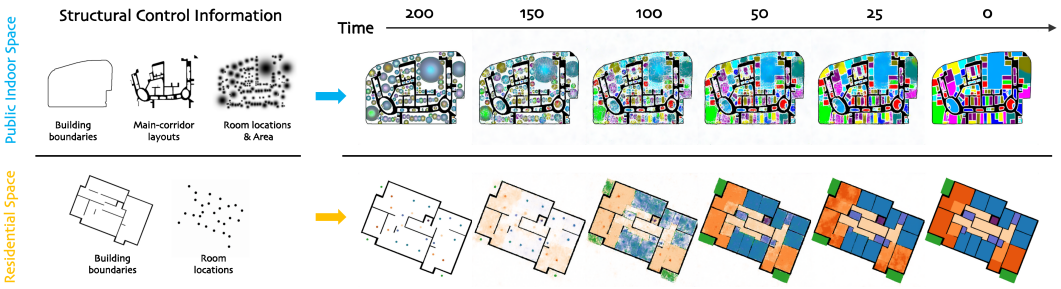


Figure 1: Given the structural control information as the input condition, LayoutBridge addresses public indoor layout generation while retaining applicability to residential spaces, moving beyond the common assumptions of Manhattan geometries.

1 Introduction

Automatic floorplan generation is a fundamental task in computer-aided architectural design [39, 10, 2], with broad applications in early-stage spatial planning, design exploration, spatial modeling, and intelligent environment perception. Recent graph-based layout modeling methods [10, 24, 25, 34]

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and diffusion-based layout synthesis techniques [9, 32, 46, 20] have achieved notable progress in relatively regular residential scenarios. However, public indoor spaces, such as shopping malls and parking facilities, remain underexplored despite their substantial practical demand [6, 44, 50].

Unlike residential layouts, public buildings are typically characterized by diverse functional spaces [38, 44] and corridor-dominated spatial organizations [45], placing higher demands on spatial coherence and geometric regularity. Consequently, methods developed under housing-scale assumptions exhibit limited transferability to such large-scale indoor scenarios. This limitation mainly arises from two aspects. (i) **The basic assumptions underlying residential layout generation become unreliable in public indoor settings.** Residential spaces are generally compact in scale [39] and exhibit relatively simple room organizations [36], allowing existing methods to neglect corridor constraints [4, 12] and non-Manhattan geometries [19]. By contrast, large-scale public interiors are associated with complex room configurations, non-Manhattan regions, intricate spatial topologies, and open corridor systems, under which these assumptions are difficult to maintain. (ii) **The generation process requires substantially higher structural preservation.** Floorplans differ from natural images in that they encode structured spatial layouts composed of walls, corridors, room boundaries, and adjacency relations. High-quality synthesis therefore depends on the preservation of local contour clarity, boundary consistency, and geometric regularity. However, existing diffusion-based methods [18, 7, 48, 52, 41] typically adopt isotropic reverse denoising, under which features are refined uniformly across spatial directions. Consequently, effective refinement along structural directions and undesirable cross-boundary smoothing are difficult to disentangle. Moreover, the absence of explicit constraints on room-shape regularity [43] may lead to blurred corners, distorted walls, and fragmented boundaries. As room count and topological complexity increase, these local artifacts accumulate and ultimately compromise structural readability and suitability for vectorization [27, 23, 21].

To address the challenges, we reformulate public indoor floorplan generation as a structure-conditioned image-to-image translation problem and propose LayoutBridge. It encodes structural control information with image-and-mask representations (Figure 1), and constructs a latent bridge from the structural domain to the target floorplan domain. LayoutBridge enforces wall regularity through two complementary mechanisms. On the one hand, we propose a regularization-prior guidance branch to construct layout priors. This branch employs an optional-input-driven regularization encoding module to encode geometric regularity. We model structural constraints and regularity priors separately through Decoupled Dual-stream Injection (DDI). We integrate the two streams through conflict-aware fusion to obtain the conditional representation. On the other hand, we formulate a structure-aware anisotropic reverse process to predict a spatially varying diffusion tensor from the conditional representation. It promotes direction-aware propagation and suppresses undesired diffusion during reverse denoising. Our contributions are summarized as follows:

- We propose *LayoutBridge*, a novel diffusion bridge framework that generalizes automatic floorplan production from residential layouts to challenging public indoor spaces. It directly maps structural conditions to target layouts, thereby moving beyond the Manhattan-geometry assumptions of existing methods and treating residential layouts as a simpler special case.
- We develop a structure-aware generation mechanism that couples anisotropic reverse process with regularization-prior guidance branch, enabling direction-aware propagation, boundary-preserving refinement, and improved geometric regularity in non-Manhattan public indoor layouts.
- We have constructed a dedicated dataset PISF for public indoor spaces with 3,517 representative floorplans, and LayoutBridge reduces FID by 132.35 and improves BIoU by 15.52% over mainstream image-to-image baselines. On the residential benchmarks MSD and RPLAN, LayoutBridge improves BIoU by 79.38% and reduces FID by 2.53, respectively. These results suggest a promising direction for scalable, structure-aware automated architectural design.

2 LayoutBridge

LayoutBridge addresses floorplan generation in public indoor spaces (Figure 2), aiming to produce structurally controllable layouts with regular room geometries.

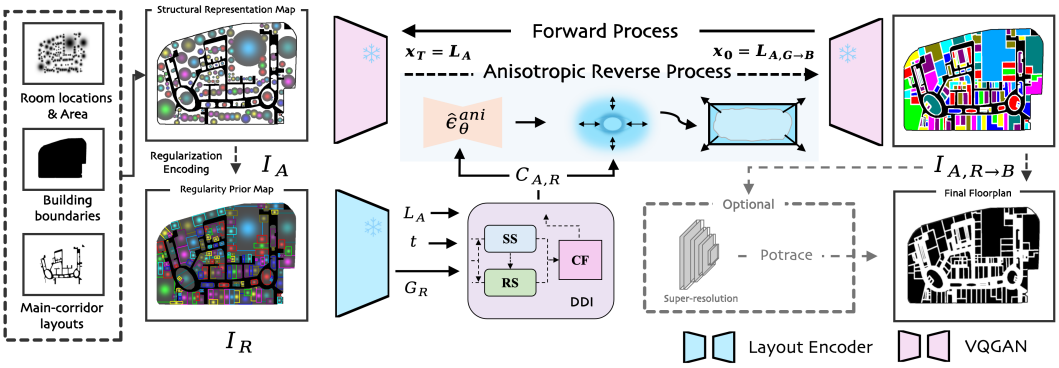


Figure 2: Overview of LayoutBridge. The structural representation map I_A defines the latent endpoint $x_T = L_A$, while the regularity prior map I_R provides the global regularity feature G_R. DDI fuses both conditions to guide anisotropic reverse denoising and generate the final floorplan. SS, RS, and CF denote the Structure Stream, Regularity Stream, and Conflict-aware Fusion, respectively.

2.1 Floorplan Representation

The structural control information I consists of room locations, building boundaries, and optional auxiliary cues such as room areas and main-corridor layouts. We encode these controls as an image-and-mask representation: room centers are rendered as RGB maps, room areas are represented by area-aware heatmaps when available, and boundaries or corridors are encoded as masks. The target floorplan is rendered with the same color-coding scheme to maintain semantic alignment in the latent space.

2.2 LayoutBridge Architecture

LayoutBridge is a latent-space Brownian bridge diffusion framework [18] that learns a direct domain-to-domain mapping from structural control information I to target floorplans. Given I, we construct the structural representation map I_A and encode it into the latent representation $L_A = E(I_A)$ using the frozen VQ encoder [5] E, with $x_T = L_A$ serving as the conditional endpoint from which the reverse generation process starts. In parallel, the Regularization Encoding Module derives a regularity prior map I_R from I_A, providing geometric regularity guidance for reverse denoising. In the regularization-prior guidance branch, a frozen DINOv3 [33]-based layout encoder extracts the global regularity feature G_R from I_R, and the Decoupled Dual-stream Injection (DDI) module fuses G_R with L_A to produce the conditional representation $C_{A,R}$ for reverse denoising (Figure 3). Starting from x_T, the denoising network $\hat{\epsilon}_\theta^{ani}$ predicts the bridge trajectory toward x_0 under the guidance of $C_{A,R}$, while an anisotropic process is injected into the reverse dynamics to enforce direction-aware propagation and boundary-preserving refinement. The final latent $L_{A,R\to B}$ is decoded by the VQ decoder D into the generated floorplan $I_{A,R\to B}$. A super-resolution module [37] and a vectorization step are further provided as optional post-processing.

Regularization Encoding Module. We design a regularization encoding module to enhance geometric regularity and boundary straightness in the target layout. Guided by optional cues, this module exploits the geometric properties of the structural representation map to construct four types of prior encodings, corresponding to four input settings: both predefined room areas and main-corridor layouts, only predefined room areas, only main-corridor layouts, and neither cue.

Decoupled Dual-stream Injection. DDI explicitly decomposes condition injection into two complementary streams, namely, a structure stream (SS) and a regularity stream (RS).

The SS enforces geometric feasibility through spatially aligned structural modulation. Given the structural latent L_A, a multi-scale conditional pyramid s_l is constructed at layer l, from which a spatial gating map is predicted as $M_l^s = \sigma(\psi_l(s_l))$, where $\psi_l(\cdot)$ and $\sigma(\cdot)$ denote a learnable projection and the sigmoid function, respectively. The gate highlights structure-sensitive regions and modulates the

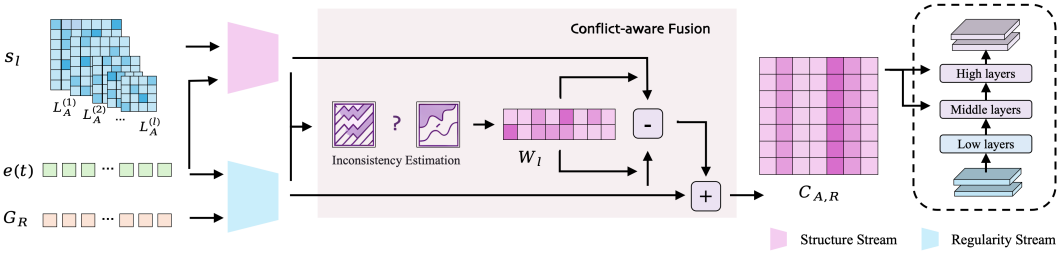
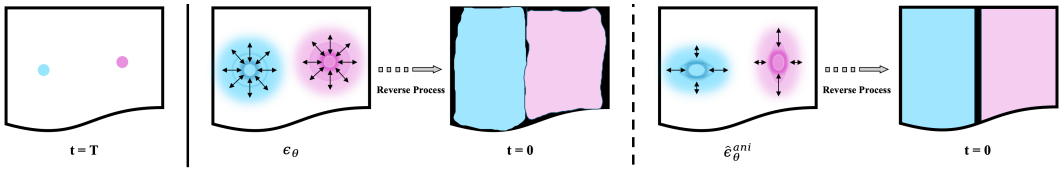


Figure 3: DDI architecture. It decomposes the regularized prior and global regularity embedding into structure and regularity streams, respectively, and adaptively balances their spatial contributions via conflict-aware fusion. DDI is restricted to the middle and high layers of the U-Net.



(a) Isotropic Operator (b) Anisotropic Operator (Ours)

Figure 4: Conceptual illustration of isotropic and anisotropic reverse processes. Given the same local structure at $t = T$, isotropic propagation uniformly diffuses information across spatial directions, which may lead to cross-boundary over-smoothing at $t = 0$ (Isotropic Operator). In contrast, anisotropic propagation aligns information flow with the local principal direction and suppresses cross-boundary diffusion, improving corridor continuity and boundary regularity at $t = 0$ (our Anisotropic Operator).

current latent feature map L_l as

$$\tilde{L}_l = L_l \odot (1 + \lambda_l^s M_l^s) + M_l^s \odot P_l^s, \tag{1}$$

where λ_l^s and P_l^s denote a learnable scaling factor and the projected structural representation map, respectively. The resulting spatial modulation imposes structure-aligned constraints on denoising and promotes geometrically feasible layout generation.

In parallel, the RS provides soft global guidance by modulating feature responses with global layout priors. Specifically, the global regularity embedding G_R is concatenated with the time-step embedding $e(t)$ and mapped by a learnable transformation $\phi_l(\cdot)$ to adaptive modulation parameters $(\gamma_l, \beta_l) = \phi_l([G_R, e(t)])$. The resulting feature L_l^c is injected through adaptive layer normalization $\mathrm{LN}(\cdot)$ as

$$L_l^c = \mathrm{LN}(\tilde{L}_l) \odot (1 + \gamma_l) + \beta_l. \tag{2}$$

Unlike the structure stream, which enforces local geometric plausibility, the regularity stream modulates feature responses in the channel dimension, allowing the model to capture higher-level layout regularities.

To mitigate interference between the two streams, we propose a conflict-aware fusion (CF) mechanism that adaptively balances their spatial contributions. The fusion weight is computed as $W_l = \sigma(\eta_l([\tilde{L}_l, L_l^c, s_l, e(t)]))$, where $\eta_l(\cdot)$ and $\sigma(\cdot)$ denote a learnable transformation and the sigmoid function, respectively. The final output feature $C_{A,R}^l$ is given by

$$C_{A,R}^l = W_l \odot L_l^c + (1 - W_l) \odot \tilde{L}_l. \tag{3}$$

A key design choice of DDI is to restrict conditional injection to the middle and high layers, rather than applying it uniformly throughout the entire U-Net hierarchy.

Anisotropic Reverse Process. L-BBDM [18] ensures structural controllability through endpoint constraints. However, its isotropic reverse diffusion cannot distinguish between propagation along structural directions and propagation across structures. This limitation makes it suboptimal for public indoor scenes, where corridor continuity, boundary alignment, and non-Manhattan geometric regularity are essential (Figure 4). To address this issue, we formulate the reverse denoising process

Table 1: Main quantitative comparison on PISF and MSD. On PISF, all methods take complete structural control information as conditions. On MSD, optional auxiliary constraints are removed for fair comparison with MHD, with all other processing settings kept consistent.

Dataset	Method	Type	LPIPS (↓)	FID (↓)	CW-SSIM (↑)	BIoU (↑)
PISF	CycleGAN	Raster	$14.41_{\pm 0.40}$	177.63	$73.04_{\pm 0.72}$	$68.51_{\pm 0.65}$
	Pix2pix		$12.78_{\pm 0.44}$	182.96	$80.07_{\pm 0.40}$	$81.31_{\pm 0.54}$
	BDBM		$24.52_{\pm 0.40}$	267.49	$50.99_{\pm 0.41}$	$55.22_{\pm 0.18}$
	CBBDM		$14.49_{\pm 0.32}$	275.13	$74.30_{\pm 0.25}$	$68.04_{\pm 0.47}$
	DocDiff		$13.98_{\pm 0.48}$	138.76	$71.01_{\pm 0.40}$	$68.83_{\pm 0.25}$
	Uni-DB		$24.86_{\pm 0.11}$	353.32	$48.24_{\pm 0.32}$	$63.03_{\pm 0.24}$
	LayoutBridge (Ours)		$\mathbf{10.07_{\pm 0.36}}$	**6.41**	$\mathbf{84.57_{\pm 0.45}}$	$\mathbf{96.83_{\pm 0.26}}$
MSD	MHD (MRR)	Vector	$58.41_{\pm 0.52}$	93.59	$24.27_{\pm 0.35}$	$10.92_{\pm 0.24}$
	MHD (MC)		$73.49_{\pm 0.44}$	164.71	$12.26_{\pm 0.65}$	$11.94_{\pm 0.68}$
	LayoutBridge (Ours)	Raster	$\mathbf{9.21_{\pm 0.25}}$	**86.72**	$\mathbf{85.70_{\pm 0.20}}$	$\mathbf{91.32_{\pm 0.12}}$

with anisotropic dynamics. Specifically, a spatially varying diffusion tensor field $\boldsymbol{A}_t$ is predicted from the conditional prior, local features, and temporal embedding $(C_{A,R}, e(t))$. At each spatial location p, the tensor is parameterized by the orientation angle $\theta_t(p)$, the principal diffusion strength $\lambda_{\parallel,t}(p)$, and the orthogonal diffusion strength $\lambda_{\perp,t}(p)$, thereby enabling direction-aware local information flow rather than isotropic propagation. Based on $\boldsymbol{A}_t$, an anisotropic operator is defined as $\boldsymbol{D}_{\boldsymbol{A}_t}(\boldsymbol{x}_t) = \nabla \cdot (\boldsymbol{A}_t \nabla \boldsymbol{x}_t)$. This operator is used to augment the original bridge residual prediction, yielding

$$\hat{\epsilon}_\theta^{\text{ani}}(\boldsymbol{x}_t, t, C_{A,R}) = \epsilon_\theta(\boldsymbol{x}_t, t, C_{A,R}) + \alpha_t \boldsymbol{D}_{\boldsymbol{A}_t}(\boldsymbol{x}_t), \tag{4}$$

where α_t controls the strength of the anisotropic correction. This design encourages the denoising trajectory to propagate preferentially along the dominant layout directions while suppressing excessive smoothing across boundaries, thereby improving wall continuity, preserving room adjacency, and enhancing overall geometric regularity under the same endpoint constraints.

Loss Functions. The training objective consists of the Brownian bridge diffusion loss and a frequency-aware reconstruction loss. The anisotropic denoising network $\hat{\epsilon}_\theta^{ani}(\cdot)$ is trained to predict the bridge residual from the intermediate latent state x_t, timestep t, and conditional representation $C_{A,R}$. The diffusion loss is defined as:

$$\mathcal{L}_{diff}^{ani} = c_t \left\| m_t(x_T - x_0) + \sqrt{\delta_t}\epsilon - \hat{\epsilon}_\theta^{ani}(x_t, t, C_{A,R}) \right\|_2^2, \tag{5}$$

where $c_t = 1/t$, $\epsilon \sim \mathcal{N}(0, I)$, m_t is the time scheduling function, and $\delta_t = 2(m_t - m_t^2)$.

To encourage both global structural consistency and local boundary fidelity, we further apply a frequency-aware reconstruction loss on the decoded raw output before any post-processing. Let $\hat{I}$ denote the decoded output of LayoutBridge and I_{GT} denote the ground-truth floorplan. We decompose the reconstruction error into low- and high-frequency components:

$$\mathcal{L}_{low} = \left\| \phi_L * (\hat{I} - I_{GT}) \right\|_2^2, \quad \mathcal{L}_{high} = \left\| \phi_H * (\hat{I} - I_{GT}) \right\|_2^2, \tag{6}$$

where ϕ_L and ϕ_H denote low-pass and high-pass filters, respectively. We use a Laplacian kernel for ϕ_H to emphasize boundary and edge discrepancies. The final objective is:

$$\mathcal{L}_{total}^{ani} = \mathcal{L}_{diff}^{ani} + \beta_0 \mathcal{L}_{high} + \beta_1 \mathcal{L}_{low}. \tag{7}$$

We set $\beta_0 = 0.5$ and $\beta_1 = 0.1$ in all experiments.

3 Experiments

3.1 Experimental Settings

Implementation Details. LayoutBridge is implemented in PyTorch based on the official L-BBDM [18] codebase. Unless otherwise specified, pretrained VQGAN and DINOv3 models are kept frozen. All models are trained for 150K iterations on four NVIDIA RTX 3090Ti GPUs with a batch size of 2, using Adam [3] with decoupled weight decay [13]. The initial learning rate is set to 1×10^{-3} and decayed by a factor of 10 every 100K iterations.

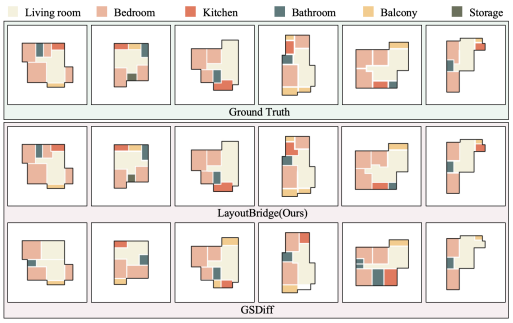


Figure 5: Qualitative comparison on RPLAN. LayoutBridge generates coherent residential layouts under the same protocol.

Table 2: Quantitative comparison on RPLAN. Results of existing methods are reported from GSDiff [11]. We follow its dataset split, metric computation, and evaluation protocol, using room locations and building boundaries as input conditions for fair comparison.

Method	**Type**	**FID** (↓)	**KID** (↓)
Graph2Plan	Raster	8.40	1.34
WallPlan		9.07	1.02
GSDiff	Vector	7.83	0.51
Cons2Plan		7.80	–
LayoutBridge (Ours)	Raster	**5.27**	**0.50**

Table 3: Quantitative comparison with floorplan-specific methods on PISF. All metrics are reported in percentages (%). The best results are highlighted in bold.

Method	Structural Metrics							Image-based Metrics		
	GED (↓)	Constraint (↑)	Validity (↑)	Connectivity (↑)	Reachability (↑)	Corridor (↑)	Accessibility (↑)	LPIPS (↓)	CW-SSIM (↑)	BIoU (↑)
HouseDiffusion	85.13	10.46	74.97	4.02	5.26	99.07	65.01	54.30	17.92	13.31
FloorDiffusion	**54.77**	40.35	**99.20**	13.02	**21.01**	98.41	98.04	36.88	76.41	87.03
LayoutBridge	60.58	**47.22**	98.68	**19.08**	12.77	**99.99**	**99.82**	**10.07**	**84.57**	**96.83**

Dataset Construction. We evaluate LayoutBridge on public indoor scenes (PISF) and residential scenes (MSD [36] and RPLAN [39]), according to the corresponding representation formats and application scenarios. **PISF** provides rasterized floorplans collected from two sources: (i) surveyed drawings of real commercial buildings and (ii) floorplan data legally obtained from public map and building-information platforms. Although PISF contains 11,482 rasterized floorplans in total, the main experiments use a building-level split with one fixed representative floorplan per building, resulting in 2,547/485/485 samples for training/validation/testing. This protocol avoids leakage among multiple floorplans from the same building. The remaining samples are retained for future extended evaluation and are not used in the main split. Both **MSD** and **RPLAN** provide raster and vector annotations. To ensure fair comparisons, we adopt the dataset splits used by MHD [36] and GSDiff [11], respectively. Detailed dataset comparisons are provided in Appendix D. Unless otherwise specified, the reported deviations are computed over test samples rather than independent training runs. All evaluation scripts and official split files will be released. For licensing-restricted samples, we will provide a reproducible evaluation protocol and a redistributable subset to allow independent verification.

Competing Approaches. On PISF, we conduct raster-based comparisons with advanced image-to-image methods, including DocDiff [42], Dual-approx Bridge [41], BDBM [15], CBBDM [16], Uni-DB [52], classical baselines such as CycleGAN [51] and Pix2pix [14], and floorplan-specific methods such as HouseDiffusion [30] and FloorDiffusion [32]. On MSD, we compare with the graph-based MHD [36], where room polygons are approximated by either More-Corners (MC) or Minimum Rotated Rectangles (MRR). On RPLAN, we compare with representative residential methods, including GSDiff [11], Cons2Plan [9], Graph2Plan [10], and WallPlan [34].

Evaluation Metrics. We adopt FID [8], LPIPS (%) [49], CW-SSIM (%) [28], and Boundary IoU (BIoU, %) [1] to evaluate the fidelity, perceptual quality, structural consistency, and boundary accuracy of generated floorplans. For BIoU, room boundaries, corridor boundaries, and building outer boundaries are extracted using morphological gradients with a fixed 5-pixel boundary width. we further add seven more architectural-validity metrics in vector-domain structural evaluations: (i)GED, (ii)Constraint Satisfaction,(iii)Polygon Validity, (iv)Connectivity, (v)Reachability, (vi)Corridor Continuity, (vii)Accessibility (see Appendix A for more details). For RPLAN, we further report KID [11] to enable a more comprehensive comparison with existing methods, with implementation details given in Appendix A. For LPIPS, CW-SSIM, and BIoU, we report mean "$\pm$" sample-wise standard deviation over the test set. FID and KID are reported as a single distribution-level value. Notably,

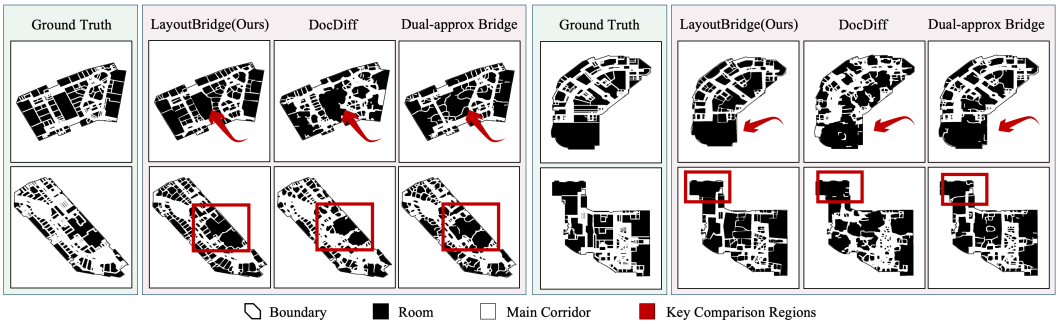


Figure 6: Comparison of floorplan samples generated by LayoutBridge, DocDiff and Dual-approx Bridge. Our results exhibit higher quality and regularity, whereas other methods commonly suffer from issues such as missing rooms, unreasonable layouts, and weak regularity.

the ESRGAN-based super-resolution module and Potrace vectorization are employed solely as optional post-processing steps for detail refinement and vector export. Unless otherwise specified, all quantitative results in the main comparative experiments are computed on the raw outputs before post-processing.

3.2 Quantitative Evaluations

The results in Table 1 show that LayoutBridge consistently outperforms representative baselines on PISF and MSD. On PISF, LayoutBridge achieves an LPIPS of 10.07 and a CW-SSIM of 84.57%. Across 485 test buildings, it reduces FID from 138.76, obtained by the second-best method DocDiff, to 6.41, indicating substantially improved distributional fidelity. It also obtains the highest BIoU, reflecting stronger boundary preservation and geometric alignment.The additional results in Table 3 demonstrate that LayoutBridge improves not only raster-level similarity but also vector-domain structural consistency. For HouseDiffusion, as the number of polygon vertices increases and their spatial configurations become complex, the model may suffer from unstable convergence, leading to degraded quality of floorplans. Although FloorDiffusion achieves competitive performance after vectorization, it still performs worse than our method under image-based evaluation metrics. On MSD, LayoutBridge consistently improves LPIPS, CW-SSIM, FID, and BIoU over MHD variants, validating its effectiveness under complex polygonal structures. Since RPLAN methods adopt different input specifications, the results are treated as a protocol-level benchmark rather than a strictly controlled comparison. As shown in Table 2, LayoutBridge achieves the lowest FID among the compared residential floorplan generators and a competitive KID, further supporting its transferability to residential layout generation.

3.3 Qualitative Evaluations

Figures 5, 6, and 7 provide qualitative comparisons on PISF, MSD, and RPLAN, respectively. LayoutBridge consistently produces high-quality floorplans with coherent room organization, continuous wall structures, and regular boundaries across different dataset settings. By contrast, image-to-image translation methods such as Dual-approx Bridge generate missing rooms, redundant rooms, and broken walls, reflecting insufficient structural constraint adherence. Vector-based methods such as MHD struggle to maintain plausible adjacency relations among rectangular regions, resulting in less realistic layout configurations. The RPLAN results further confirm the generalizability of LayoutBridge to residential layout generation.

3.4 Ablation Studies

Effectiveness of DDI. DDI consists of three key components. Table 6 quantitatively evaluates their respective contributions by removing each component in turn. The results show that the Structure Stream improves the model's responsiveness to input structural constraints, whereas the Regularity Stream is more effective in enhancing geometric regularity. On this basis, introducing Conflict-aware Fusion further yields the best performance across all metrics, indicating that it effectively alleviates the conflicts between different conditional cues and improves overall generation quality.

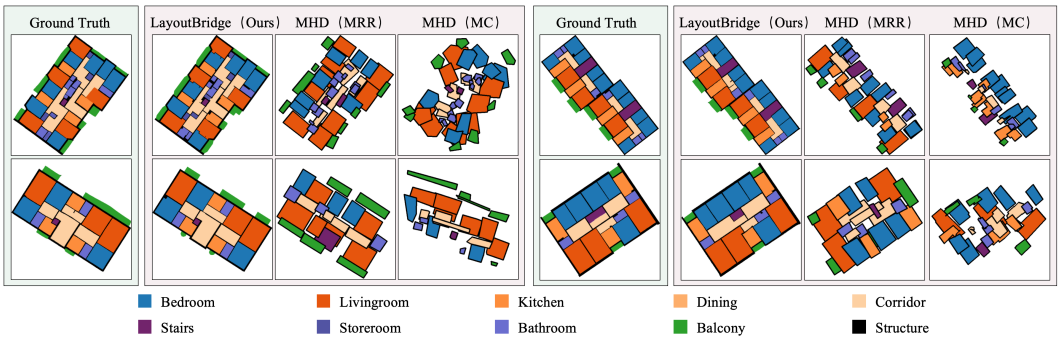


Figure 7: Comparison of floorplan samples generated by LayoutBridge and MHD on the MSD dataset. MRR denotes rooms modeled by the minimum rotated rectangle, while MC denotes rooms with more than four corners. LayoutBridge produces noticeably higher-quality results than MHD.

Table 4: Effectiveness of the proposed DDI scheme. ExemplarNet follows EBDM [17].

Method	LPIPS (↓)	CW-SSIM (↑)	BIoU (↑)
L-BBDM	$16.92_{\pm 0.20}$	$71.48_{\pm 0.32}$	$91.47_{\pm 0.36}$
+ Conv. [16]	$14.49_{\pm 0.15}$	$74.30_{\pm 0.12}$	$94.76_{\pm 0.17}$
+ ExemplarNet	$14.64_{\pm 0.19}$	$74.86_{\pm 0.31}$	$94.62_{\pm 0.24}$
+ DDI	$13.48_{\pm 0.24}$	$79.96_{\pm 0.32}$	$95.77_{\pm 0.32}$
LayoutBridge	$\mathbf{10.07_{\pm 0.36}}$	$\mathbf{84.57_{\pm 0.45}}$	$\mathbf{96.83_{\pm 0.26}}$
w/o DDI, w/ Conv.	$13.92_{\pm 0.38}$	$76.48_{\pm 0.41}$	$95.40_{\pm 0.45}$
w/o DDI, w/ ExemplarNet	$13.87_{\pm 0.42}$	$78.92_{\pm 0.34}$	$95.86_{\pm 0.35}$

Table 5: Ablation study of conditional injection at different U-Net levels, where Low, Mid, and High denote the low-, middle-, and high-level layers, respectively.

Low	Mid	High	LPIPS (↓)	CW-SSIM (↑)	BIoU (↑)
✓			$24.31_{\pm 0.53}$	$68.32_{\pm 0.42}$	$87.38_{\pm 0.44}$
	✓		$16.45_{\pm 0.44}$	$71.65_{\pm 0.40}$	$91.14_{\pm 0.41}$
		✓	$15.23_{\pm 0.25}$	$70.64_{\pm 0.27}$	$90.96_{\pm 0.31}$
✓	✓		$16.32_{\pm 1.10}$	$70.99_{\pm 0.82}$	$92.45_{\pm 0.67}$
✓	✓	✓	$13.12_{\pm 0.53}$	$78.44_{\pm 0.46}$	$95.37_{\pm 0.48}$
✓		✓	$11.24_{\pm 0.42}$	$80.94_{\pm 0.34}$	$95.56_{\pm 0.33}$
	✓	✓	$\mathbf{10.07_{\pm 0.36}}$	$\mathbf{84.57_{\pm 0.45}}$	$\mathbf{96.83_{\pm 0.26}}$

Table 6: Ablation study of the DDI module. SS denotes the Structure Stream, RS denotes the Regularity Stream, and CF denotes the Conflict-aware Fusion.

SS	RS	CF	LPIPS (↓)	CW-SSIM (↑)	BIoU (↑)
✓			$15.63_{\pm 0.13}$	$69.44_{\pm 0.19}$	$90.17_{\pm 0.24}$
✓	✓		$13.20_{\pm 0.15}$	$76.54_{\pm 0.40}$	$94.56_{\pm 0.06}$
	✓		$16.13_{\pm 0.04}$	$71.21_{\pm 0.05}$	$92.14_{\pm 0.08}$
✓	✓	✓	$\mathbf{10.07_{\pm 0.36}}$	$\mathbf{84.57_{\pm 0.45}}$	$\mathbf{96.83_{\pm 0.26}}$

Table 7: Effectiveness of different floorplan representations.

Representing Method		LPIPS (↓)	CW-SSIM (↑)	BIoU (↑)
Channels	Type			
3	RGB	$13.62_{\pm 0.24}$	$77.45_{\pm 0.23}$	$92.31_{\pm 0.24}$
1	Binary	$18.94_{\pm 0.05}$	$65.81_{\pm 0.05}$	$89.42_{\pm 0.08}$
3	Mask	$12.21_{\pm 0.35}$	$80.36_{\pm 0.31}$	$93.16_{\pm 0.46}$
5	Mask + RGB(Ours)	$\mathbf{10.07_{\pm 0.36}}$	$\mathbf{84.57_{\pm 0.45}}$	$\mathbf{96.83_{\pm 0.26}}$

Furthermore, as shown in Table 4, removing DDI or replacing it with an Exemplar Network leads to consistent performance degradation, further confirming the superiority of the proposed DDI in structural perception and generation quality enhancement.

Effect of DDI Injection Levels. Table 5 demonstrates that conditional injection is most effective at the middle and high U-Net levels. Low-level-only injection produces the weakest performance, suggesting that early conditioning disrupts fine-grained feature representation. Middle-level injection primarily benefits structural constraint modeling, whereas high-level injection better incorporates global semantics and regularity priors. The limited gain from combining low- and middle-level injection further indicates interference from low-level conditioning. The best performance is achieved by joint middle- and high-level injection, confirming that this configuration best aligns with the hierarchical feature roles in floorplan generation.

Analysis of Floorplan Representation. Table 7 shows that jointly modeling mask and visual information yields the best performance among different input representations. Binary representation relies mainly on contour information and is insufficient for capturing complex spatial structures. In contrast, combining mask and RGB representations achieves the strongest results, indicating that explicit spatial partition information and rich visual context are complementary for floorplan layout reconstruction.

Table 8: Ablation study on the effectiveness of the anisotropic reverse process. Here, "Isotropic Operator" denotes an additional reverse operator whose tensor is fixed as the identity matrix. "Fixed Prior" uses an explicit structure-derived tensor field, whereas "Learnable" predicts the tensor field entirely from the network. All settings are kept identical except for the variable under study.

Method	**Operator**	**Tensor Type**	**LPIPS** (↓)	**CW-SSIM** (↑)	**BIoU** (↑)
Reverse Process	–	–	$15.68_{\pm 0.12}$	$72.11_{\pm 0.26}$	$81.32_{\pm 0.11}$
+ Isotropic Operator	✓	$A_t = I$	$15.35_{\pm 0.10}$	$73.23_{\pm 0.08}$	$81.91_{\pm 0.16}$
+ Fixed Anisotropic Tensor	✓	Fixed prior	$12.44_{\pm 0.35}$	$79.36_{\pm 0.29}$	$85.14_{\pm 0.28}$
+ Learnable Tensor(Ours)	✓	Learnable	$\mathbf{10.07_{\pm 0.36}}$	$\mathbf{84.57_{\pm 0.45}}$	$\mathbf{96.83_{\pm 0.26}}$

Table 9: Additional ablation study on the anisotropic process.

Method	Extra Branch	**Learnable**	Directional Anisotropy	Tensor Operator	**BIoU** (↑)
Reverse Process					81.32
Fixed anisotropic tensor			✓	✓	85.14
Learnable isotropic tensor	✓	✓		✓	89.28
Parameter-matched residual branch	✓	✓			91.87
Isotropized at inference	✓	✓		✓	90.61
Learnable anisotropic tensor	✓	✓	✓	✓	**96.83**

Table 10: Effect of additional post-processing. "–" denotes raw model outputs.

Method	**Post-processing**	**LPIPS** (↓)	**CW-SSIM** (↑)	**BIoU** (↑)
LayoutBridge	–	$10.07_{\pm 0.36}$	$84.57_{\pm 0.45}$	$96.83_{\pm 0.26}$
LayoutBridge	ESRGAN	$9.62_{\pm 0.54}$	$86.11_{\pm 0.62}$	$96.94_{\pm 0.33}$
LayoutBridge	ESRGAN + Potrace	$10.21_{\pm 0.12}$	$84.12_{\pm 0.06}$	$96.62_{\pm 0.10}$

Effectiveness of the Anisotropic Reverse Process. Table 8 shows that anisotropic tensor-guided propagation yields more stable performance gains. The improvements are particularly evident in LPIPS and Boundary IoU [1], indicating that direction-aware propagation enhances perceptual quality, boundary alignment, and layout regularity. Moreover, the learnable tensor variant achieves the best performance, indicating that the network can infer spatially varying diffusion directions from the conditional representation. Table 9 shows that the fixed anisotropic tensor achieves 85.14% BIoU, while learnable isotropic and parameter-matched residual branches reach 89.28% and 91.87%, respectively. The full learnable anisotropic tensor achieves 96.83% but drops to 90.61% when isotropized at inference, indicating that the gain primarily stems from directional anisotropy rather than additional model capacity. Figure 8 shows that the learned anisotropic strength is concentrated around walls and structural boundaries and remains stable throughout the reverse trajectory. Meanwhile, the corresponding operator responses are also predominantly distributed in these regions. These observations demonstrate that the learned diffusion tensor field exhibits a clear correlation with the underlying geometric structures of the generated floorplans.

Effectiveness of the optional Post-processing. Table 10 demonstrates that ESRGAN and Potrace serve as optional post-processing modules for detail refinement and vector export, rather than

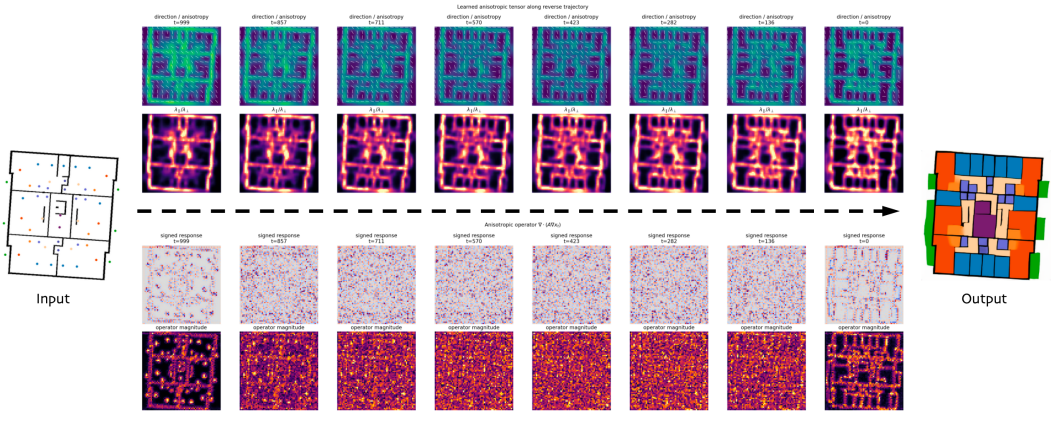


Figure 8: Visualization of the anisotropic reverse process. The first-row visualizations show the learned anisotropic tensors and their directional strength ratios during the reverse generation process. And the lower-row visualizations illustrate the responses produced by applying the anisotropic operator to the intermediate states x_t along the reverse diffusion trajectory.

constituting the primary source of performance improvement. LayoutBridge achieves competitive performance without post-processing, and ESRGAN further improves boundary delineation and perceptual quality. The use of Potrace yields a slight performance decrease due to contour simplification and geometric approximation during vectorization.

4 Related Work

Floorplan generation. Floorplan generation has been widely studied in computer-aided architectural design [39, 10, 29]. Existing methods cover data-driven and GAN-based residential layout generation [39, 24, 25], as well as graph- [31, 35], vector- [43, 45], and diffusion-based approaches for modeling spatial topology, wall structures, and polygonal geometry [10, 34, 22, 30, 11, 40, 9, 32]. Nevertheless, most methods target residential scenarios with compact and regular layouts [39, 36], making their assumptions less reliable for public indoor spaces with open corridors, dense room organizations, and non-Manhattan geometries.

Public indoor layout generation. Compared with residential floorplans, public indoor layouts remain less explored. Existing studies mainly focus on crowd-driven mid-scale layout design [6], procedural layout modeling for shopping centres [44], and probabilistic mall layout organization [50]. Related indoor mapping studies also highlight the practical importance of large-scale public indoor spatial modeling [38, 47]. However, these studies do not directly address structure-conditioned raster floorplan generation for public indoor spaces. LayoutBridge targets this problem by combining regularization-prior guidance, decoupled condition injection, and anisotropic reverse denoising.

Diffusion bridge and conditional generation. Brownian bridge diffusion models formulate image-to-image translation as an endpoint-constrained stochastic process between source and target domains [18, 26]. Subsequent bridge-based methods improve conditional translation through consistency modeling [7], bidirectional bridge learning [15], exemplar guidance [17], implicit sampling [48], stochastic optimal control [52], and dual-approximator denoising [41]. These methods provide a suitable foundation for structure-conditioned floorplan generation. However, most existing bridge [26, 23] models employ isotropic reverse denoising, where feature propagation is spatially uniform. Such a formulation is insufficient for floorplans, because layout synthesis requires preserving wall continuity, suppressing cross-boundary smoothing, and refining structures along dominant geometric directions.

5 Conclusion

This paper presents LayoutBridge, a latent-space Brownian bridge diffusion framework for structure-conditioned public indoor floorplan generation. By combining regularization-prior guidance with Decoupled Dual-stream Injection and a structure-aware anisotropic reverse process, LayoutBridge improves structural controllability, boundary alignment, and geometric regularity in complex public indoor layouts. Extensive evaluations on PISF, MSD, and RPLAN datasets have demonstrated our effectiveness in producing high-quality and complex floorplans in public indoor spaces and residential layouts. However, *LayoutBridge* still lacks a quantitative metric to measure aesthetic quality, which will be addressed in our future work.

Acknowledgments and Disclosure of Funding

We would like to thank the anonymous reviewers for their valuable comments and suggestions. This work was supported in part by the National Natural Science Foundation of China (Grant No. 62572041 and No. 62402028), the European Union CoGNETs project under the Horizon Europe programme (grant number: 101135930), the UKRI through the UK Government's Horizon Europe funding Guarantee (grant number: 10119564), the ByteDance Research Collaboration Plan, and the Xiaomi Research Collaboration Plan.

References

[1] Cheng, B., Girshick, R., Dollár, P., Berg, A.C. & Kirillov, A. (2021) Boundary IoU: Improving object-centric image segmentation evaluation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15329–15337.

[2] Chen, J., Deng, R. & Furukawa, Y. (2023) PolyDiffuse: Polygonal shape reconstruction via guided set diffusion models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NeurIPS)*.

[3] Diederik P. & Jimmy B. (2014) Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[4] Dupty, M.H., Dong, Y., Leng, S. et al. (2024) Constrained layout generation with factor graphs. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12851–12860.

[5] Esser, P., Rombach, R. & Ommer, B. (2020) Taming transformers for high-resolution image synthesis. *arXiv preprint arXiv:2012.09841*.

[6] Feng, T., Yu, L.-F., Yeung, S.-K., Yin, K. & Zhou, K. (2016) Crowd-driven mid-scale layout design. *ACM Transactions on Graphics* **35**(4):1–14.

[7] He, G., Zheng, K., Chen, J., Bao, F. & Zhu, J. (2024) Consistency diffusion bridge models. In *Advances in Neural Information Processing Systems*, Vol. 37.

[8] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. & Hochreiter, S. (2017) GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 6629–6640.

[9] Hong, S., Zhang, X., Du, T., Cheng, S., Wang, X. & Yin, J. (2024) Cons2Plan: Vector floorplan generation from various conditions via a learning framework based on conditional diffusion models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 3248–3256.

[10] Hu, R., Huang, Z., Tang, Y., van Kaick, O., Zhang, H. & Huang, H. (2020) Graph2Plan: Learning floorplan generation from layout graphs. *ACM Transactions on Graphics* **39**(4).

[11] Hu, S., Wu, W., Wang, Y., Xu, B. & Zheng, L. (2025) GSDiff: Synthesizing vector floorplans via geometry-enhanced structural graph generation. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence*.

[12] Ibrahim, M.R., Musil, J. & Gallou, I. (2024) Generating floorplans for various building functionalities via latent diffusion model. *arXiv preprint arXiv:2412.06859*.

[13] Ilya Loshchilov & Frank Hutter. (2018) Decoupled weight decay regularization. In International Conference on Learning Representations.

[14] Isola, P., Zhu, J.-Y., Zhou, T. & Efros, A.A. (2017) Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[15] Kieu, D., Do, K., Nguyen, T., Nguyen, D. & Nguyen, T. (2025) Bidirectional diffusion bridge models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1139–1148.

[16] Kim, S.-H. & Chung, D. (2025) Conditional Brownian bridge diffusion model for VHR SAR to optical image translation. *IEEE Geoscience and Remote Sensing Letters*.

[17] Lee, E., Jeong, S. & Sohn, K. (2025) EBDM: Exemplar-guided image translation with Brownian-bridge diffusion models. In *Computer Vision – ECCV 2024*, Vol. 15071 of *Lecture Notes in Computer Science*. Springer Nature Switzerland.

[18] Li, B., Xue, K., Liu, B. & Lai, Y.-K. (2023) BBDM: Image-to-image translation with Brownian bridge diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1952–1961.

[19] LIFULL Co., Ltd. (2015) LIFULL Home's: A large-scale database of residential floor plans. Available at *https://www.nii.ac.jp/dsc/idr/en/lifull/*.

[20] Liu, C., Fan, W., Liu, Y. et al. (2023) Generative diffusion models on graphs: Methods and applications. *arXiv preprint arXiv:2302.02591*.

[21] Long, Z., Zheng, M., Feng, K. et al. (2025) Follow-Your-Shape: Shape-aware image editing via trajectory-guided region control. *arXiv preprint arXiv:2508.08134*.

[22] Luo, Z. & Huang, W. (2022) FloorplanGAN: Vector residential floorplan adversarial generation. *Automation in Construction* **142**:104470.

[23] Mahesh, R.M., Chao, J.-J. & Isler, V. (2025) VisDiff: SDF-guided polygon generation for visibility reconstruction, characterization and recognition. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*.

[24] Nauata, N., Chang, K.-H., Cheng, C.-Y., Mori, G. & Furukawa, Y. (2020) House-GAN: Relational generative adversarial networks for graph-constrained house layout generation. In *Computer Vision – ECCV 2020*, Vol. 12346 of *Lecture Notes in Computer Science*. Springer International Publishing.

[25] Nauata, N., Hosseini, S., Chang, K.-H., Chu, H., Cheng, C.-Y. & Furukawa, Y. (2021) House-GAN++: Generative adversarial layout refinement network towards intelligent computational agent for professional architects. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13627–13636.

[26] Nguyen, D.T., Nguyen, T.T., Nguyen, H.T. et al. (2025) CT to PET translation: A large-scale dataset and domain-knowledge-guided diffusion approach. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1498–1507.

[27] Ribeiro, L.F.R., Savarese, P.H.P. & Figueiredo, D.R. (2017) Struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 385–394.

[28] Sampat, M.P., Wang, Z., Gupta, S., Bovik, A.C. & Markey, M.K. (2009) Complex wavelet structural similarity: A new image similarity index. *IEEE Transactions on Image Processing* **18**(11):2385–2401.

[29] Sereshgi, E.H. & Wenk, C. (2024) Map Stitcher: Graph sampling-based map conflation. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Spatial Big Data and AI for Industrial Applications*, pp. 5–15.

[30] Shabani, M.A., Hosseini, S. & Furukawa, Y. (2023) HouseDiffusion: Vector floorplan generation via a diffusion model with discrete and continuous denoising. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5466–5475.

[31] Sharma, K., Kumar, S. & Trivedi, R.S. (2024) Diffuse, sample, project: Plug-and-play controllable graph generation. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

[32] Shim, J., Moon, J., Kim, H. & Hwang, E. (2024) FloorDiffusion: Diffusion model-based conditional floorplan image generation method using parameter-efficient fine-tuning and image inpainting. *Journal of Building Engineering* **95**:110320.

[33] Simeoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., Massa, F., Haziza, D., Wehrstedt, L., Wang, J., Darcet, T., Moutakanni, T., Sentana, L., Roberts, C., Vedaldi, A., Tolan, J., Brandt, J., Couprie, C., Mairal, J., Jégou, H., Labatut, P. & Bojanowski, P. (2025) DINOv3. *arXiv preprint arXiv:2508.10104*.

[34] Sun, J., Wu, W., Liu, L., Min, W., Zhang, G. & Zheng, L. (2022) WallPlan: Synthesizing floorplans by learning to generate wall graphs. *ACM Transactions on Graphics* **41**(4):1–14.

[35] Tenorio, V.M., Zilberstein, N., Segarra, S. & Marques, A.G. (2025) Graph guided diffusion: Unified guidance for conditional graph generation. *arXiv preprint arXiv:2505.19685*.

[36] Van Engelenburg, C., Mostafavi, F., Kuhn, E. et al. (2025) MSD: A benchmark dataset for floorplan generation of building complexes. In *Computer Vision – ECCV 2024*, Vol. 15116 of *Lecture Notes in Computer Science*. Springer Nature Switzerland.

[37] Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y. & Loy, C.C. (2018) ESRGAN: Enhanced super-resolution generative adversarial networks. In *The European Conference on Computer Vision Workshops (ECCVW)*.

[38] Wang, Y., Kuang, J., Liu, T., Niu, X. & Liu, J. (2024) CrowdMagMap: Crowdsourcing-based magnetic map construction for shopping mall. *IEEE Internet of Things Journal* **11**(3):5362–5373.

[39] Wu, W., Fu, X.-M., Tang, R., Wang, Y., Qi, Y.-H. & Liu, L. (2019) Data-driven interior plan generation for residential buildings. *ACM Transactions on Graphics* **38**(6):1–12.

[40] Wu, W., Sheng, T., Zhang, G. & Zheng, L. (2025) FloorplanSBS: Synthesizing vector floorplans by patch-based floorplan segmentation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10778–10786.

[41] Xiao, B., Wang, P., He, Q. & Dong, M. (2025) Deterministic image-to-image translation via denoising Brownian bridge models with dual approximators. *arXiv preprint arXiv:2512.23463*.

[42] Yang, Z., Liu, B., Xiong, Y., Yi, L., Wu, G., Tang, X., Liu, Z., Zhou, J. & Zhang, X. (2023) DocDiff: document enhancement via residual diffusion models. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 2795–2806.

[43] Yu, D., Hu, Y., Li, Y. & Zhao, L. (2024) PolygonGNN: Representation learning for polygonal geometries with heterogeneous visibility graph. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 4012–4022.

[44] Zhang, B. & Li, B. (2025) From knowledge encoding to procedural generation for early-stage layout design: A case of linear shopping centres. *Frontiers of Architectural Research* **14**(1):282–294.

[45] Zhang, H., Savov, A. & Dillenburger, B. (2024) MaskPLAN: Masked generative layout planning from partial input. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8964–8973.

[46] Zhang, H. & Zhang, R. (2025) Generating accessible multi-occupancy floorplans with fine-grained control using a diffusion model. *Automation in Construction* **177**:106332.

[47] Zhang, J., Wu, S., Ma, X. & Schwertfeger, S. (2025) Generation of indoor OpenStreetMaps for robot navigation from CAD files. *arXiv preprint arXiv:2507.00552*.

[48] Zheng, K., He, G., Chen, J., Bao, F. & Zhu, J. (2025) Diffusion bridge implicit models. In *The Thirteenth International Conference on Learning Representations (ICLR)*.

[49] Zhang, R., Isola, P., Efros, A.A., Shechtman, E. & Wang, O. (2018) The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[50] Zhou, Q. & Koh, I. (2025) Adaptive shopping malls: Probabilistic spatial layouts with graph neural networks. *Design and Artificial Intelligence* **1**(1):100003.

[51] Zhu, J.-Y., Park, T., Isola, P. & Efros, A.A. (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*.

[52] Zhu, K., Pan, M., Ma, Y. et al. (2025) UniDB: A unified diffusion bridge framework via stochastic optimal control. In *Proceedings of the Forty-Second International Conference on Machine Learning (ICML)*.

Technical appendices and supplementary material

A Detail of Evaluation Metrics

Boundary IoU For BIoU, we first convert each generated and ground-truth floorplan into semantic masks and extract boundary maps using a morphological gradient operator. Room boundaries, corridor boundaries, and building outer boundaries are included. The boundary width is fixed to 5 pixels for all methods. BIoU is then computed as the intersection-over-union between the generated and ground-truth boundary regions.

KID Following GSDiff [11], we additionally report Kernel Inception Distance (KID) to measure the distributional discrepancy between generated and real floorplans. KID compares the maximum mean discrepancy between the feature distributions of real and generated samples in the Inception feature space. Given real features $X = \{x_i\}_{i=1}^{m}$ and generated features $Y = \{y_j\}_{j=1}^{n}$, KID is computed with a polynomial kernel:

$$k(a, b) = \left(\frac{1}{d} a^\top b + 1\right)^3, \tag{8}$$

and its unbiased estimate is:

$$\mathrm{KID} = \frac{1}{m(m-1)} \sum_{i \neq j}^{m} k(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j}^{n} k(y_i, y_j) - \frac{2}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} k(x_i, y_j). \tag{9}$$

Lower KID values indicate that the generated floorplans are closer to the real data distribution.

We also employ the following seven metrics for quantitative evaluation:

1. **GED / Compatibility GED (↓).**
   We employ the Hungarian algorithm to establish spatial correspondences between predicted and ground-truth nodes based on polygon Intersection over Union (IoU) and centroid distance. The normalized node-edit cost and adjacency-edge-edit cost are then computed and averaged. This metric represents an approximate Compatibility Graph Edit Distance (GED) obtained through spatial matching.

2. **Constraint Satisfaction (↑).**
   Since the dataset does not provide explicit constraint annotations, we treat ground-truth adjacency relations as structural constraints. The metric is computed using dataset-level micro-averaging:

   $$\text{Constraint Satisfaction} = 1 - \mathrm{CVR} = \frac{\sum TP_{\mathrm{adj}}}{\sum |E_{\mathrm{GT}}|}. \tag{10}$$

   This metric serves as a proxy derived from adjacency constraints rather than a strict reproduction of the original Constraint Violation Rate (CVR).

3. **Polygon Validity (↑).**
   Following the OGC Simple Feature specifications, we measure the proportion of geometrically valid polygons among all vectorized polygons. Invalid geometries include self-intersections, invalid interior-ring configurations, and degenerate boundaries.

4. **Connectivity / Adjacency Precision (↑).**
   After establishing spatial node correspondences, we compute the proportion of correctly predicted adjacency edges among all predicted adjacency edges. The metric is calculated using dataset-level micro-averaging:

   $$\text{Connectivity} = \text{Adjacency Precision} = \frac{\sum TP_{\mathrm{adj}}}{\sum |E_{\mathrm{pred}}|}. \tag{11}$$

5. **Pairwise Reachability (↑).**
   Given the predicted adjacency graph, we measure the proportion of unordered node pairs that are connected by a finite-length path. The metric is defined as:

   $$\text{Reachability} = \frac{\sum_k \binom{|C_k|}{2}}{\binom{|V|}{2}}, \tag{12}$$

   where C_k denotes the k-th connected component of the predicted adjacency graph, and V denotes the set of predicted nodes.

6. **Corridor Continuity (↑).**
   We evaluate corridor connectivity using the largest connected corridor component ratio, defined as:

   $$\text{Corridor Continuity} = \frac{|C_{\max}^{\mathrm{corridor}}|}{\sum_k |C_k^{\mathrm{corridor}}|}, \tag{13}$$

   where C_k^{corridor} denotes the k-th connected component of the corridor adjacency subgraph, and $C_{\max}^{\mathrm{corridor}}$ denotes its largest connected component. Corridor regions are inferred from the free-space geometry within the ground-truth spatial extent.

7. **Accessibility (↑).**

   Since the original dataset does not provide explicit entrance or doorway annotations, we approximate the entrance region using the largest exterior free-space component. We then measure the proportion of predicted regions that are reachable from this component:

$$\text{Accessibility} = \frac{N_{\text{regions connected to exterior}}}{N_{\text{predicted regions}}}, \tag{14}$$

   Therefore, this metric serves as a geometry-based proxy for the entrance-to-room accessibility ratio.

B Evaluation on Manhattan and non-Manhattan layouts within PISF

We further construct two separate test subsets from PISF: one with Manhattan structures and the other with non-Manhattan structures. For each subset, we report LPIPS, CW-SSIM, BIoU. The results shown in Table 11 demonstrate that our LayoutBridge maintains stable reconstruction performance on both subsets. It achieves superior performance in LPIPS, CW-SSIM, and BIoU, indicating its robustness to diagonal walls, irregular boundaries, and complex non-orthogonal geometries. In contrast, most baseline methods show larger performance gaps in perceptual similarity and boundary consistency.

Table 11: Quantitative comparison on the Manhattan and non-Manhattan subsets. The best results are highlighted in bold. All metrics are reported in percentages (%).

Method	Manhattan			Non-Manhattan		
	LPIPS ↓	CW-SSIM ↑	BIoU ↑	LPIPS ↓	CW-SSIM ↑	BIoU ↑
BDBM	33.73	59.13	33.86	38.14	55.86	52.16
CBDM	30.18	66.43	63.66	31.66	70.96	79.90
CycleGAN	32.09	58.06	55.02	32.76	68.78	74.07
DocDiff	32.67	65.71	70.25	34.34	65.90	71.83
Pix2Pix	30.94	68.97	74.56	32.31	79.75	75.00
UniDB	33.01	58.26	65.81	33.86	68.50	72.58
LayoutBridge	**10.05**	**93.89**	**89.64**	**9.02**	**94.78**	**91.31**

C Vector-domain Quantitative Evaluations

In addition, we further add seven more layout-native structure-level metrics in vector-domain evaluations: i. GED(Graph Edit Distance), ii. Constraint Satisfaction(Constraint Satisfaction Rate), iii. Polygon Validity(OGC polygon validity rate), iv. Connectivity(Adjacency Precision), v. Reachability(pairwise reachability), vi. Corridor Continuity(Largest Connected Corridor Component Ratio), vii. Accessibility(Entrance-to-room accessibility ratio).

The results in Table 12 show that on MSD and RPLAN, LayoutBridge achieves competitive performance in terms of GED, adjacency constraint satisfaction, adjacency precision, pairwise reachability, and accessibility. Notably, on RPLAN, LayoutBridge improves the constraint satisfaction rate from 76.03% (GSDiff) to 97.88%, while increasing adjacency precision from 78.52% to 97.48%.

On PISF, LayoutBridge achieves the highest adjacency constraint satisfaction, corridor continuity, and accessibility, although it does not attain the best results in GED and adjacency precision. These results suggest that LayoutBridge is more effective at preserving global spatial connectivity and free-space structures, but may still introduce spurious adjacency edges, thereby reducing adjacency precision.

Table 13: Comparison between our dataset and other datasets of similar type.

(a) Basic statistics and completeness.

Dataset	Region	Space Type	Scale
RPLAN [39]	Asia	Single apartment	81k+ floorplans, 161.8k+ rooms
MSD [36]	Europe	Single / multi-apartment	5,372 floorplans, 165.3k+ rooms
PISF(Ours)	Asia	Mall / parking lot	11,482 floorplans, 1,010k+ rooms

(b) Complexity and diversity.

Dataset	Complexity		Diversity	
	Non-Manhattan Room Ratio	Corridor Areas	Avg. #Corners	Graph Dist. Entropy
RPLAN	No	No	5.04	4.56
MSD	5%+	No	8.68	8.02
PISF(Ours)	91.86%	Yes	17.20	12.23

Table 12: Comparison of structural and topological metrics after vectorization.

Dataset	Method	GED (%, ↓)	Constraint (%, ↑)	Validity (%, ↑)	Connectivity (%, ↑)	Reachability (%, ↑)	Corridor (%, ↑)	Accessibility (%, ↑)
PISF	BDBM	92.21	4.09	57.21	9.07	**36.20**	77.99	67.26
	CBBDM	63.64	38.93	96.47	16.46	15.10	99.81	98.95
	CycleGAN	59.61	36.44	97.02	24.46	4.06	99.68	98.95
	DocDiff	67.35	20.53	97.42	**32.68**	2.83	99.02	98.39
	Pix2Pix	**57.55**	39.80	96.33	26.88	6.33	99.07	98.22
	Uni-DB	63.30	37.24	97.74	16.40	13.53	99.33	99.04
	Ours	60.58	**47.22**	**98.68**	19.08	12.77	**99.99**	**99.82**
MSD	MHD (MRR)	73.89	23.80	87.90	3.12	30.41	–	74.58
	MHD (MC)	67.19	26.45	94.76	32.83	59.84	–	86.97
	Ours	**36.36**	**49.83**	**98.54**	**66.45**	**80.23**	–	**99.54**
RPLAN	GSDiff	**2.74**	76.03	**99.87**	78.52	98.39	–	98.65
	Ours	4.77	**97.88**	99.64	**97.48**	**99.43**	–	**99.94**

D Detailed Dataset Comparison

To demonstrate the greater complexity of public indoor floorplans compared with residential layouts, we provide a detailed comparison between PISF and representative residential floorplan datasets, including RPLAN and MSD, as shown in Table 13. Unlike residential datasets that primarily cover single-apartment layouts, our dataset focuses on public indoor spaces such as shopping malls and parking facilities. Although PISF contains fewer floorplan images than large-scale residential datasets such as RPLAN, it contains over 1.01 million room units, indicating substantially higher room-level density and spatial complexity.In terms of geometric structure, residential datasets contain only a limited proportion of non-Manhattan rooms, e.g., approximately 5% in MSD, whereas non-Manhattan structures account for 91.86% of our dataset. Moreover, public-space layouts commonly include explicit corridor regions and open circulation spaces. The average number of corners increases from 4–8 in residential datasets to 17.20 in ours, suggesting considerably more intricate geometric configurations. From a topological perspective, the graph distribution entropy of our dataset reaches 12.23, markedly higher than that of residential datasets, which is typically below 8. This indicates more diverse connectivity patterns and richer spatial organizational forms.

A detailed dataset card, including data sources, preprocessing, quality control, license considerations, privacy issues, and release plan, is provided in Table 14. We will additionally release a redistributable PISF-mini subset with the same preprocessing and evaluation protocol to enable independent verification.

E Additional Inference Results of LayoutBridge

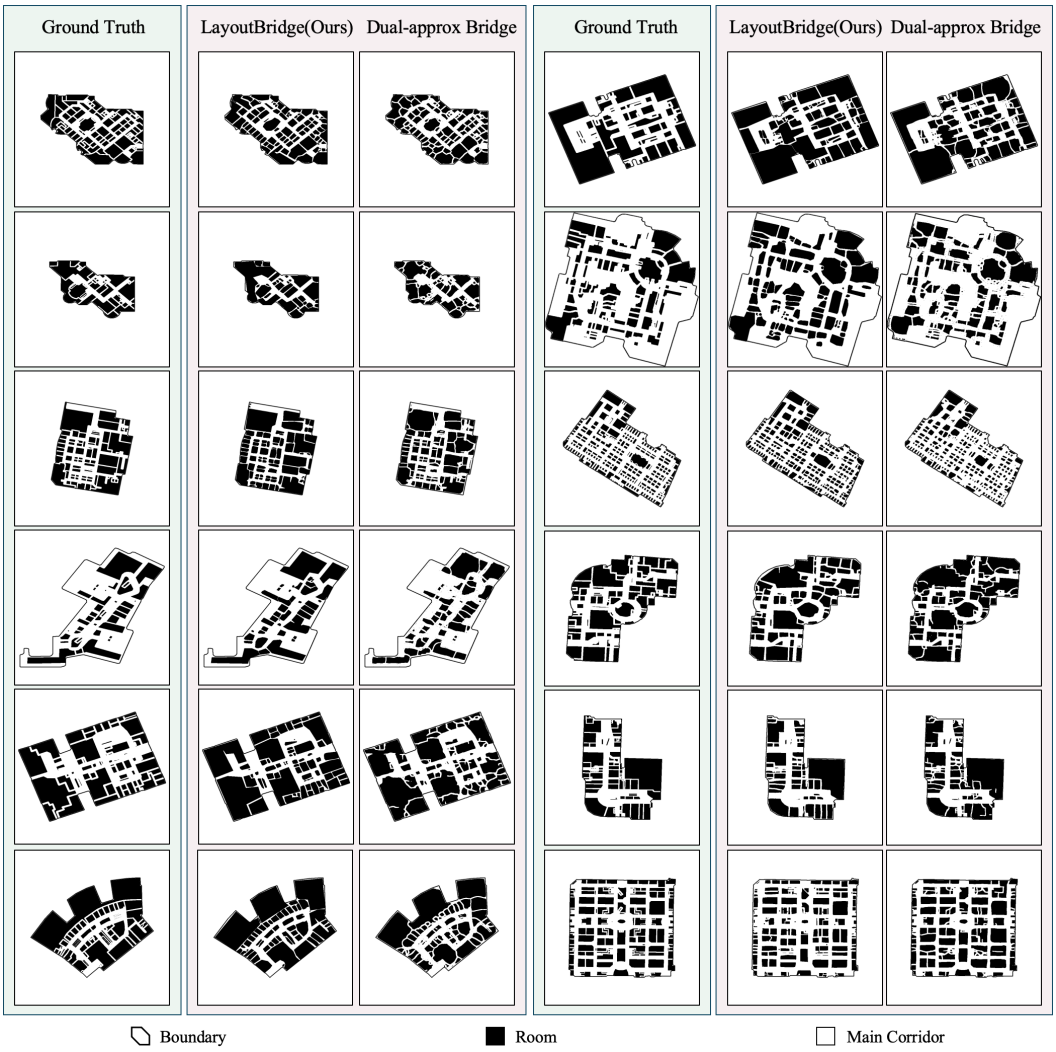


Figure 9: Inference Results on the PISF Dataset

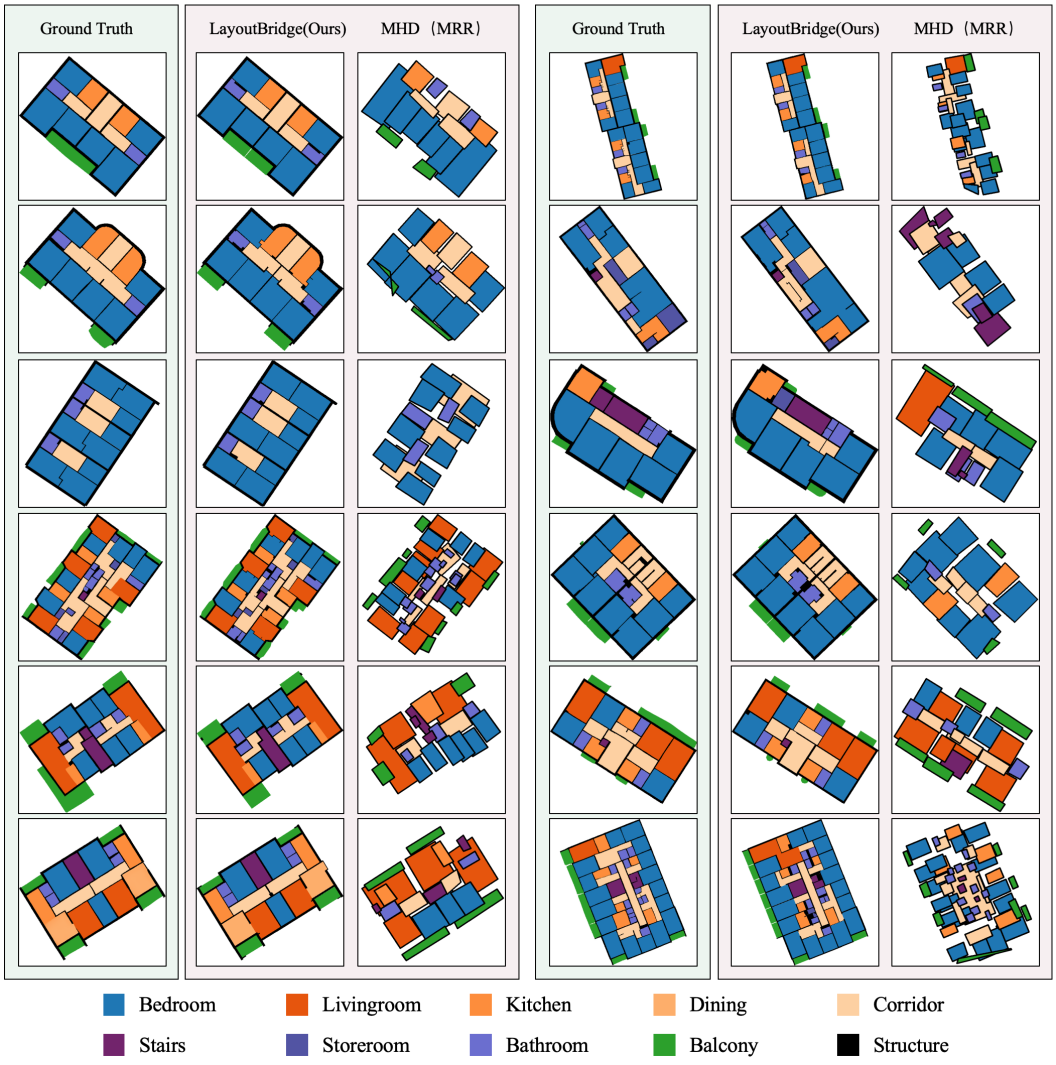


Figure 10: Inference Results on the MSD Dataset



Figure 11: Inference Results on the RPLAN Dataset

Table 14: Dataset card for PISF.

Item	Description
Dataset name	Public Indoor Space Floorplans (PISF).
Task	Structure-conditioned public indoor floorplan generation. Given structural controls such as room locations, building boundaries, optional room-area cues, and optional main-corridor layouts, the task is to generate a complete public indoor floorplan.
Data sources	PISF is constructed from two sources: surveyed drawings of real commercial buildings and floorplan data legally obtained from public map and building-information platforms.
Region and space type	The dataset mainly covers public indoor spaces in Asia, including shopping malls and parking facilities.
Scale	PISF contains 11,482 floorplans and over 1.01 million room units.
Structural characteristics	Compared with residential floorplan datasets, PISF contains more complex public-space layouts, explicit corridor/open circulation areas, a high proportion of non-Manhattan rooms, and more diverse topology.
Annotations	Each sample is represented as paired structural-control input and target floorplan. The structural-control input includes room locations, building boundary masks, optional room-area cues, and optional main-corridor masks. The target floorplan is rendered using the same color-coding scheme as the input image to maintain semantic alignment in the latent space.
Cleaning and preprocessing	We remove duplicated floorplans, incomplete drawings, severely corrupted images, unreadable low-resolution samples, and samples with inconsistent room-boundary topology. The remaining floorplans are rasterized to a unified resolution. Room regions, building boundaries, and corridor areas are normalized into the image-and-mask representation used by LayoutBridge.
Quality control	Samples are checked for valid building boundaries, non-empty room regions, consistent room-color encoding, and valid mask alignment between structural inputs and target floorplans. Samples failing these checks are excluded from training and evaluation.
Data split	We select 2547 representative buildings for training and use 485 buildings each for validation and testing. The split is performed at the building level to avoid leakage between train and test samples from the same building.
License and authorization	The dataset includes only floorplans for which the authors have surveying permission or legal access from public map/building-information platforms. Raw source files that are restricted by third-party licenses are not redistributed. When redistribution is allowed, only processed rasterized representations and annotations are released for non-commercial research use. To facilitate reproducibility, we will release the official train/validation/test split files, evaluation scripts, preprocessing code, and all redistributable processed annotations. For non-redistributable samples, we provide metadata descriptions and construction protocols so that eligible users can reconstruct the corresponding subset when they have legal access to the original sources.
Privacy and ethical considerations	The dataset contains architectural layout information and does not include human subjects, personal identities, trajectories, or private behavioral records. Sensitive metadata unrelated to floorplan generation is removed during preprocessing.
Release plan	An anonymized implementation and preprocessing scripts will be provided during review. The processed PISF annotations and metadata will be released under a research-only license where permitted by the original data terms. For samples that cannot be redistributed due to licensing restrictions, we provide data-construction scripts and metadata descriptions to support reproducibility. A redistributable PISF-mini subset with the same preprocessing and evaluation protocol will also be released to facilitate independent verification.
Intended use	Research on public indoor floorplan generation, layout synthesis, structure-conditioned image-to-image translation, and floorplan evaluation.
Out-of-scope use	The dataset should not be used for identifying private building information, security-sensitive indoor mapping, or commercial redistribution without permission from the original data owners.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Introduction state the problem scope, the proposed LayoutBridge framework, the regularization-prior guidance branch, and the anisotropic reverse update. The claimed improvements are supported by the quantitative, qualitative, and ablation results in the Experiments section.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses its current limitations in the Conclusion, noting that LayoutBridge still lacks a quantitative metric for evaluating aesthetic quality and leaves this issue for future work. It also bounds the experimental scope by clarifying the use of raw outputs for main evaluation and by specifying the release constraints of licensing-restricted PISF samples.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not claim formal theoretical theorems. It provides mathematical formulations of the Brownian bridge process, DDI module, loss functions, and anisotropic reverse dynamics, together with implementation and stability discussions in Appendix **??**.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper reports the model architecture, floorplan representation, training protocol, datasets, data splits, competing methods, evaluation metrics, and ablation settings. Additional implementation and methodological details are provided in the appendix to support reproduction of the main results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: An anonymized implementation, preprocessing scripts, and evaluation protocols will be provided during review. For PISF, all redistributable processed annotations and metadata will be released under a research-only license where permitted by the original data terms. For restricted samples, the paper provides data-construction scripts, metadata descriptions, and split information to support reproducibility.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The Experiments section specifies the optimizer, learning rate schedule, batch size, number of training iterations, diffusion steps, hardware configuration, dataset splits, baselines, and evaluation metrics. These details are sufficient to understand the reported training and testing setup.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: For the main comparison and ablation studies, the reported deviations are computed over test samples rather than independent training runs. The same data split, training schedule, and evaluation protocol are used across all compared methods. The source of variation and the number of runs are specified in the experimental setup. We acknowledge that multi-seed training would provide stronger evidence of statistical significance, but it is not performed due to computational cost. The reported deviations are computed over test samples rather than independent training runs. We acknowledge that multi-seed training would provide stronger statistical evidence and leave it as a limitation due to computational cost.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The Implementation Details section reports the main computational resources used for training, including four NVIDIA RTX 3090Ti GPUs. The training iterations, batch size, and diffusion steps are also specified to indicate the computational scale of the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

Justification: The study uses architectural floorplan data for layout generation and does not involve human subjects, personal identities, behavioral records, or interventions. Potential risks related to sensitive indoor layout information are considered in the dataset card, and restricted raw files are not redistributed.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses positive impacts by positioning LayoutBridge as a tool for early-stage spatial planning, design exploration, spatial modeling, and intelligent environment perception. It also addresses potential negative impacts related to sensitive indoor layout information, stating that restricted raw files are not redistributed and that privacy and ethical risks are considered in the dataset documentation.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The work does not release high-risk models such as large language models, face generators, surveillance systems, or scraped personal datasets. The generated content is restricted to indoor floorplan layouts, and therefore the high-risk release safeguards described in the question are not applicable.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites the existing datasets, codebases, models, and metrics used in the experiments, including L-BBDM, ESRGAN, DINOv3, MSD, RPLAN, and standard evaluation metrics. Their use is limited to research benchmarking and method implementation, with attribution provided through references.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces the PISF dataset for public indoor floorplan generation. A dataset card is provided in Appendix D, including the task definition, data sources, scale, annotation format, preprocessing pipeline, quality control, data split, license considerations, privacy issues, release plan, intended use, and out-of-scope use.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The study does not involve crowdsourcing experiments or research with human subjects. No participant instructions, screenshots, or compensation details are therefore applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The study does not involve crowdsourcing or human-subject experiments, and no personal data are collected from participants. Therefore, IRB approval or equivalent human-subject review is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method, experiments, and claims do not use LLMs as an important, original, or non-standard component. Any use of language tools, if any, is limited to writing or editing and does not affect the scientific methodology or results.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.