TOSCA FOR SCENE-GRAPH-TO-3D GENERATION WITH SET-VALUED CERTIFIED ASSEMBLY

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating a 3D scene from a scene graph requires selecting plausible arrangements that satisfy its relations. Graph-conditioned generators can produce plausible layouts that leave requested relations unsatisfied, and high average relation accuracy can coexist with many scenes that fail to satisfy the complete graph. To address this problem, we propose Transport Operators with Set-valued Certified Assembly (TOSCA), which integrates geometric checks and corrections into generation through explicit constraint sets called transport sets. These sets retain the range of configurations permitted by each relation, such as the region of relative object positions allowed by close-by. Projection residuals measure deviations from the allowed sets and guide graph attention and geometric corrections in a flow-matching generator, followed by final refinement. Because individually valid relations may conflict when combined, the same formulation also provides infeasibility certificates for difference constraints and objective bounds for fixed affine convex subproblems. Across SG-FRONT room types, TOSCA increases the fraction of requested relations satisfied by generated geometry and lowers rendered-image FID relative to the evaluated full-split baselines. Ablations support retaining transport sets and final refinement, while conflict and editing experiments assess infeasibility detection and preservation of surrounding objects.

1 INTRODUCTION

A 3D scene graph represents objects as nodes and their relationships as edges (Wald et al., 2020). For scene generation, it specifies the objects and relations that the generated 3D arrangement should satisfy (Dhamo et al., 2021). Existing methods such as EchoScene (Zhai et al., 2024) use *learned relation embeddings* together with object and graph context to guide scene generation. In Fig. 1(a), three chairs must be close to one table, allowing many seating arrangements. The learned embeddings provide conditioning signals for generation, but do not guarantee that the geometric requirements of each relation are satisfied. Thus, such methods can place a chair too far from the table to satisfy close-by, breaking this requested grouping even when the rest of the room appears plausible (Fig. 1(b)). Therefore, faithful generation depends on realizing the requested relations in each scene, alongside producing plausible layouts.

Graph-to-3D (Dhamo et al., 2021), CommonScenes (Zhai et al., 2023), EchoScene (Zhai et al., 2024), and MMGDreamer (Yang et al., 2025) combine learned relation embeddings with object features, graph context, and sampled latent variables or noise to generate a distribution of layouts. This learned conditioning also appears in the graph Transformer of InstructScene (Lin & Mu, 2024) and the rectified flow of FlowScene (Yang et al., 2026). However, our controlled evaluation in Table 2 shows that methods relying on learned relation conditioning can achieve high edge-level satisfaction while still generating many scenes that violate at least one requested relation. Reliable generation must therefore prevent relation violations while retaining the multiple arrangements specified by the input graph.

To this end, we propose Transport Operators with Set-valued Certified Assembly (TOSCA). TOSCA represents each relation by its allowed geometric values in a constraint set called a *transport set*. For example, our close-by set admits a region of nearby placements. Our symmetrical-to formulation allows alternative mirror-symmetric center positions across either planar coordinate axis through the scene origin. Each axis defines a separate set of allowed configurations, which we call a *reflec-*

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

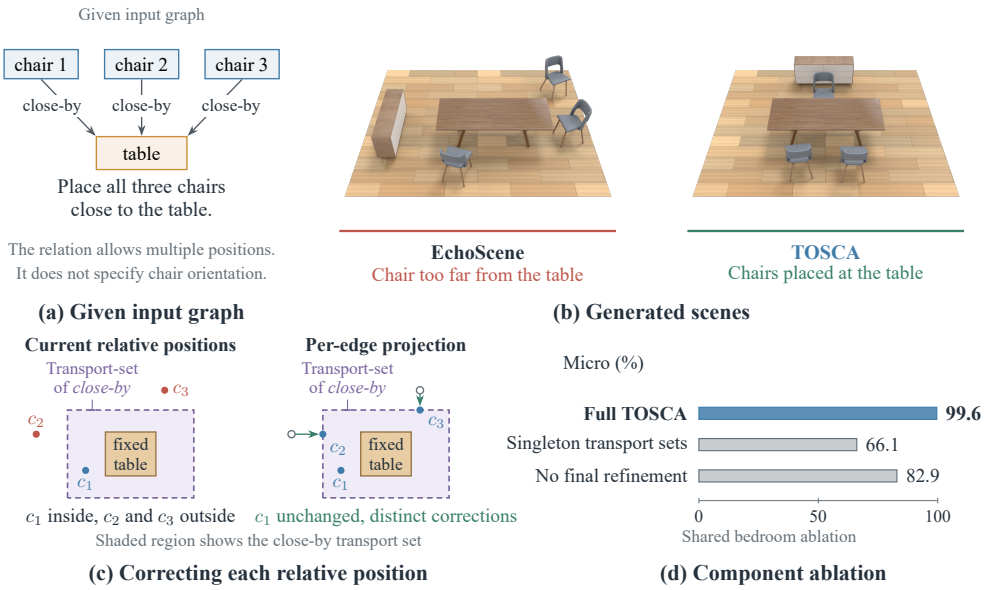


Figure 1: **Transport sets and geometric corrections.** (a,b) A shared input graph and scenes generated by EchoScene (Zhai et al., 2024) and TOSCA. (c) Schematic per-edge projection onto a close-by region with the table fixed. (d) Transport sets and final refinement improve relation satisfaction in the shared bedroom ablation (Table 18).

tion branch (Fig. 2(a)). Both examples retain multiple valid placements, and the transport operator projects the current geometry onto a closest allowed value in the transport set when the request is violated. Figure 1(c) illustrates this operation for three chairs around a fixed table. The shared relation supplies a common admissibility rule, while each edge's current geometry determines its correction. Projection preserves positions inside the proximity region and maps each violating position to its closest allowed value.

The admissibility rule must also remain consistent under edge reversal. A chair left of a table and the same table right of that chair express the same geometric requirement, so these inverse relations must admit identical layouts. We enforce this equivalence by sharing their constraint parameters.

Building on this set-valued representation, TOSCA performs certified assembly by connecting the representation to generation through *projection residuals* and applying final refinement, with infeasibility certificates and objective bounds for the stated constraint subproblems. Specifcally, for each edge, the residual is the difference between its current feature and the closest allowed value. Our Transport, Compare, Aggregate (TCA) network uses this residual in graph attention over neighboring objects and in geometric corrections. The transport sets specify admissible values, while flow matching learns how to select plausible layouts among them. Since learned updates can leave violations, final refinement applies further geometric corrections.

Transport sets preserve individual alternatives, but their joint requirements can conflict. A cycle of strict left requests, for example, cannot hold in any layout. A remaining violation alone does not distinguish incompatible requests from incomplete refinement. To distinguish these cases, we derive *infeasibility certificates*, verifiable evidence of such conflicts, for *difference constraints*. These inequalities bound differences between scalar object coordinates and represent directional relations and size comparisons. For fixed affine convex subproblems, bounds on the minimum violation also assess how close a returned layout is to the best achievable compromise (Sec. 3.5).

Across the room types in SG-FRONT (Zhai et al., 2023), TOSCA satisfies a larger fraction of requested benchmark relations and achieves substantially lower FID and KID than the evaluated full-split baselines (Secs. 4.2 and 4.3). Within TOSCA, replacing transport sets with *singleton transport sets* (i.e., each containing only one allowed value) or removing final refinement causes large drops in relation satisfaction (Fig. 1(d)). Conflict and editing experiments further examine whether the representation can identify incompatible requests and preserve surrounding objects after a relation changes. These findings support the following contributions.

- A set-valued geometric relation representation that supplies per-edge admissibility checks and corrections within learned generation, retaining multiple valid configurations through allowed regions and reflection branches while ensuring that inverse relations admit the same layouts.
- A generation architecture that uses projection residuals to guide graph attention and geometric correction based on current relation violations, and applies final refinement to reduce violations remaining after learned updates.
- Infeasibility certificates for incompatible difference constraints to identify requests that no layout can satisfy, objective bounds for fixed affine convex subproblems to assess how close a returned layout's violation is to the minimum achievable value, and edit-sensitivity bounds for optimal layouts in anchored and strongly convex problems to characterize how edit effects decay with graph distance under the stated regularity conditions.
- Controlled evaluations showing higher benchmark relation satisfaction and lower FID and KID than the recent baselines, with additional tests showing that the method detects most injected contradictions and realizes relation edits while limiting displacement of neighboring objects.

2 Related Work

Scene graphs encode objects and their relationships for visual reasoning and generation (Johnson et al., 2018; Wald et al., 2020). Graph-conditioned generators use learned relation embeddings in message passing or attention to guide a distribution of layouts. SceneLinker (Kim et al., 2026) further studies scene generation from graphs extracted from RGB sequences. Related approaches condition generation on instructions, unordered object sets, or plans produced by language models (Bai et al., 2025; Tang et al., 2024; Paschalidou et al., 2021; Feng et al., 2023; Yang et al., 2024b; Sun et al., 2025; Gu et al., 2026). Across these settings, plausible layouts can retain incorrect object relations. Correcting such errors requires accounting for the other relations incident to a moved object, since improving one relation can disturb another.

Constraint-guided generation addresses geometric errors through checks and corrections. Diffusion-CCSP (Yang et al., 2023) combines learned constraint energies, while projected diffusion (Christopher et al., 2024) restricts samples to supplied feasible regions. For scene graphs, individually reasonable requests can also conflict, so a remaining violation does not establish whether further correction can succeed. We therefore study both relation correction during generation and the diagnosis of incompatible requests. TOSCA uses relation-specific constraint sets to guide corrections and provides infeasibility certificates and objective bounds for fixed affine convex subproblems. For extended related work, see Appendix A.

3 Generation with Transport Sets

TOSCA uses transport sets to connect relation features to TCA updates, final refinement, and conflict checking (Fig. 2). We first define the object states and the features that each relation constrains, then derive the residuals used by the network. After describing generation and final refinement, we analyze feasibility, objective error, and edit sensitivity under their stated assumptions.

3.1 Object States and Relation Features

A scene graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \rho)$ has object nodes $\mathcal{V} = \{1, \dots, n\}$, directed edges $\mathcal{E}$, and a map ρ assigning a relation label to each edge. We write $e = (i, r, j)$ when $r = \rho(e)$ relates object i to object j, and denote their categories by c_i, c_j. Each object state combines planar position and yaw with size and elevation in the group

$$g_i \in \mathsf{G} = (\mathrm{SO}(2) \ltimes \mathbb{R}^2) \times \mathbb{R}^4. \tag{1}$$

Here $\mathrm{SO}(2) \ltimes \mathbb{R}^2$ is the planar rigid-motion group, in which rotations act on translations. The remaining coordinates $(\log \ell_i, \log h_i, \log w_i, z_i)$ give log-length, log-height, log-width, and elevation.

We call the quantities checked for an edge its *relation feature* η_e. Table 4 gives all fifteen SG-FRONT labels (Zhai et al., 2023) and their transport sets. The twelve spatial labels comprise five directional relations (left, right, front, behind, above), four size comparisons (bigger-than, smaller-than, taller-than, shorter-than), standing-on (floor height), close-by (proximity), and symmetrical-to

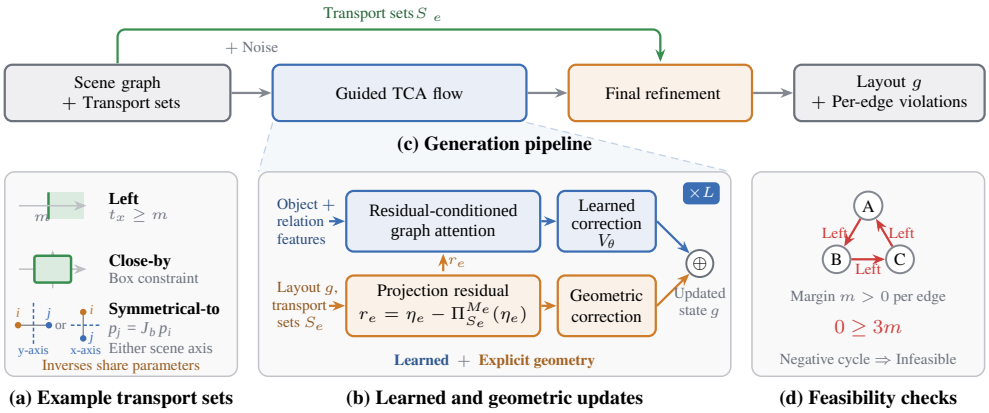


Figure 2: **Generation and checking with transport sets.** (a) Example transport sets. (b) Residual-conditioned graph attention and geometric TCA updates. (c) Generation followed by refinement and per-edge checks. (d) A negative cycle certifies incompatible difference constraints.

(symmetry). They constrain world-axis differences, log-size differences, bottom height, planar displacement, and reflected positions, respectively. Three semantic labels, same style as, same material as, and same super-category as, constrain corresponding latent shape features.

The reference frame determines which transformations preserve a relation. In object i's local frame, $\xi_{ij} = \log(g_i^{-1} g_j)$ expresses relative coordinates in a fixed local logarithm chart and is invariant to shared rigid transformations. SG-FRONT instead uses world axes, the floor, or the benchmark origin. Changing these references can alter satisfaction while preserving relative arrangements.

3.2 Allowed Configurations and Inverse Consistency

A transport set specifies the allowed values of an edge's relation feature, so different feature values can satisfy the same relation.

Definition 1 (Transport set)**.** For edge $e = (i, r, j)$, the transport set $S_e = S_r(c_i, c_j)$ contains the allowed values of η_e. We call membership $\eta_e \in S_e$ *fitted-set satisfaction*. A transport set containing exactly one feature value is a *singleton transport set*.

The criteria in Table 4 require signed directional or log-size comparisons of at least a fitted margin, bottom height within a tolerance of the floor for standing-on, and planar center separations bounded by summed half-extents plus a fitted surface-gap allowance for close-by, as derived from axis-wise surface gaps in Appendix B.2. These criteria define half-spaces, a slab, and a box, respectively. Semantic relations equate selected latent coordinates. These fitted constraints guide generation, while benchmark predicates score the output. For close-by, we use the box to keep the position constraint convex at fixed object sizes and retain the released minimum-corner-distance rule for scoring (Zhai et al., 2024).

The requests i left of j and j right of i must allow the same arrangement. For catalog inverse pairs whose reversed feature is its negation, we impose

$$S_{r^{-1}}(c_j, c_i) = -S_r(c_i, c_j), \qquad S_r(c_j, c_i) = -S_r(c_i, c_j) \quad \text{if } r = r^{-1}, \tag{2}$$

where r^{-1} is the inverse relation. A relation such as close-by is its own inverse, so exchanging its two objects preserves the label. Category-dependent bounds are tied accordingly, and each reversed pair uses the same distance metric.

A union of branches can represent alternative relation configurations, as illustrated by symmetrical-to in Fig. 2(a) (Appendix B.2). Fixed frames, object sizes, and branches give affine convex subproblems. Changing sizes and proximity bounds jointly or switching branches can break convexity.

3.3 Residuals and TCA Updates

Projection gives each violating edge a geometry-dependent correction while preserving every allowed feature value. Distance to the set measures violation, and the difference from a closest al-

lowed value specifies the correction. Let M_e be a symmetric positive-definite (SPD) distance metric. Write $\|u\|_{M_e}^2 = u^\top M_e u$, let $\Pi_{S_e}^{M_e}(\eta)$ minimize $\|\eta - s\|_{M_e}$ over $s \in S_e$, and denote this minimum by $d_{M_e}(\eta, S_e)$. With edge weights $q_e \in [0, 1]$, the projection residual and violation energy are

$$r_e = \eta_e - \Pi_{S_e}^{M_e}(\eta_e), \qquad E(g) = \tfrac{1}{2} \sum_{e \in \mathcal{E}} q_e d_{M_e}^2(\eta_e, S_e). \tag{3}$$

We assign unit weight to the relations specified by the input graph. For a fixed closed convex set, the residual vanishes exactly when its fitted constraint holds. Thus, every allowed configuration has zero penalty. More generally, zero energy establishes joint fitted-set satisfaction when every checked edge has positive weight. Zero-weight edges require separate checks.

For the left relation, projection preserves valid coordinate differences and moves a violating value to the margin. For close-by, the interval grows with object size because nearby surfaces can have distant centers. With a compatible diagonal metric, projection clamps only coordinates outside the allowed intervals. Each edge's current displacement therefore determines its projected value, while an already admissible displacement is unchanged by this projection (Appendix D.1).

Individual projection residuals can request different movements of a shared object. TCA uses residual-conditioned graph attention to aggregate messages from neighboring objects. The aggregated messages provide scene context for a learned correction, which is combined with the geometric correction in the state update. Node features $\mathbf{f}_i$ encode category and graph context. Let $N(i)$ be the graph neighbors of node i, e_r the learned embedding of relation type r, and t the flow time. Following neighborhood attention (Veličković et al., 2018), learned maps ϕ_m and ϕ_α produce messages and attention scores conditioned on node features, relation embeddings, and projection residuals. A softmax over $N(i)$ gives the weights for messages carried by scene-graph edges. At layer ℓ, a positive diagonal preconditioner P_θ and step size $\tau_\ell > 0$ scale the geometric correction, while V_θ maps the node features and aggregated messages to the learned correction. The map ψ updates node features. Suppressing layer indices on message inputs, we obtain

$$m_{j \to i} = \phi_m(\mathbf{f}_i, \mathbf{f}_j, r_e, \Pi_{S_e}^{M_e}(\eta_e), e_r, t), \quad \alpha_{ij} = \underset{j \in N(i)}{\operatorname{softmax}} \phi_\alpha(\mathbf{f}_i, \mathbf{f}_j, r_e, e_r), \tag{4}$$

$$g_i^{(\ell+1)} = g_i^{(\ell)} \exp\Big(-\tau_\ell P_\theta(\mathbf{f}_i) \widetilde{\nabla}_i E + V_\theta(\mathbf{f}_i, \textstyle\sum_j \alpha_{ij} m_{j \to i})\Big), \tag{5}$$

$$\mathbf{f}_i^{(\ell+1)} = \psi(\mathbf{f}_i^{(\ell)}, \textstyle\sum_j \alpha_{ij} m_{j \to i}). \tag{6}$$

Here θ collects learned parameters, $\widetilde{\nabla}_i$ expresses the energy gradient in tangent coordinates at the current state, and $\exp$ maps a tangent update back to the group. The learned correction V_θ can increase energy. With $P_\theta = I$ and $V_\theta = 0$, the update becomes residual-gradient descent for fixed affine constraints (Sec. 3.5).

3.4 Flow Matching and Final Refinement

Transport sets constrain the output geometry but do not define a probability distribution over plausible layouts. Flow matching learns this distribution by fitting a velocity field v_θ that transports noise g_0 toward a training layout g_1 (Chen & Lipman, 2024). For a sampled time $t \in [0, 1]$, let g_t be their interpolation and $\dot{g}_t$ its target velocity in the prediction's tangent coordinates. The training loss is

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{t, g_0, g_1} \|v_\theta(g_t, t, \mathcal{G}) - \dot{g}_t\|^2. \tag{7}$$

During sampling, the learned field and negative energy gradient jointly update the layout. Initial samples are centered to fix global translation. Because symmetrical-to refers to a specific origin and axes, we align layouts with this benchmark reference frame. We call this operation *reference-frame alignment*. It fixes the reference used to evaluate symmetrical-to, while a separate orientation channel supplies relative-angle features and predicts yaw quadrants. Yaw affects rendering but is absent from the original spatial scoring rules, so the orientation channel learns it separately (Appendix D.2).

Final refinement uses residual-gradient steps with the learned correction V_θ disabled to reduce violations, while separate certificates identify incompatible requests (Appendix D.4). For inequality relations, sampling and final refinement use smaller *sampling sets* with a *sampling margin* specifying the required distance inside the original boundaries. Final residuals and conflict checks use the original transport sets, since shrinking sets can introduce conflicts absent from the original request. A remaining violation alone does not establish infeasibility. Negative-cycle checks certify incompatible difference constraints, while primal-dual gaps bound objective error for fixed affine convex subproblems (Sec. 3.5). Training also fits transport-set boundaries to annotated relation features (Appendix D.3). The experiments render layouts with fixed or retrieved shapes.

3.5 THEORETICAL ANALYSIS

The analysis characterizes the choices retained by transport sets, certifies incompatible requests, bounds minimum violation, and examines edit sensitivity. For the convex results, a *fixed affine convex subproblem* stacks layout coordinates into x and uses features $\eta_e = A_e x + c_e$, where A_e is a fixed linear map and c_e a fixed offset. Transport sets S_e are nonempty, closed, and convex, weights q_e are positive, and metrics M_e are SPD. Frames, size-dependent bounds, and branches remain fixed. The stacked maps and offsets are A and c. The guarantees follow from separation, projection, and sparse-matrix decay. Appendix C groups the proofs with related results on feasible alternatives, relation completion, fixed points, and coordinate equivariance.

Geometric choices in transport sets. Set-valued proximity constraints preserve different valid placements for objects that share the same relation label. Projection leaves these allowed values unchanged and gives each violating edge a correction determined by its current relative position. The following result explains why replacing the allowed region with one fixed relative pose can force distinct objects to coincide.

Theorem 1 (Limitation of equality to a single relative pose)**.** *Suppose relation r requires $\xi_{ij} = \mu_r$ for one fixed relative pose μ_r in the local logarithm coordinates of Sec. 3.1. This rule cannot represent a proximity region with positive volume because a one-point set has measure zero. Moreover, edges (u, r, w) and (v, r, w) force $g_u = g_v$. Replacing the single value by a set with nonempty interior removes this forced equality.*

The singleton ablation exhibits lower edge-weighted and relation-balanced benchmark satisfaction within TOSCA, supporting the practical value of retaining allowed alternatives (Table 18).

Infeasibility certificates and objective bounds. A large violation energy can reflect either incompatible constraints or incomplete refinement. Infeasibility certificates establish that no layout can satisfy all requests, while a *primal-dual gap*, the difference between a returned objective value and a certified lower bound, bounds the remaining objective error for a fixed affine convex subproblem. Joint feasibility requires every relation feature to lie in its transport set. A separating vector can certify that no such layout exists, reducing to a contradictory cycle for difference constraints.

Theorem 2 (Certificates for fixed affine convex subproblems)**.** *Let $S = \prod_e S_e$ and define its support function by $\sigma_S(y) = \sup_{s \in S} \langle y, s \rangle$. If $A^\top y = 0$ and $\sigma_S(y) < \langle y, c \rangle$, then no x satisfies $Ax + c \in S$. Every infeasible polyhedral system admits such a certificate. For conic systems, this guarantee requires strong separation and excludes weak infeasibility, where the affine image and S are disjoint but have zero distance.*

For difference constraints, the incidence operator B maps node values to edge differences. When $A = B$ and $c = 0$, a certificate lies in the cycle space $\ker B^\top$, consisting of edge vectors with zero net flow at every node. A system of difference constraints is feasible exactly when it contains no negative cycle, meaning a directed cycle whose inequality bounds sum to a negative value. Bellman-Ford (Cormen et al., 2009) decides this in $O(nm)$ for n nodes and m edges. Rational polyhedral feasibility is also decidable in polynomial time. With alternative reflection branches, infeasibility of the full problem requires ruling out every allowed branch assignment.

For example, summing three left constraints around a cycle with positive margin m gives $0 \leq -3m$. Once exact satisfaction is ruled out, a lower bound on minimum violation assesses how close refinement comes to the best compromise.

Theorem 3 (Objective bounds and convergence)**.** *For a fixed affine convex subproblem, let the total squared violation be $V(x) = 2E(x) = \sum_e q_e d^2_{M_e}(A_e x + c_e, S_e)$ and assume its minimum $V^\star$ is attained. With $W = \mathrm{blkdiag}(q_e M_e)$, the block-diagonal matrix of weighted metrics, every y satisfying $A^\top y = 0$ gives*

$$D(y) = \langle y, c \rangle - \sigma_S(y) - \tfrac{1}{4} y^\top W^{-1} y \;\leq\; V^\star \;\leq\; V(x). \tag{8}$$

The primal-dual gap $V(x) - D(y)$ therefore bounds the error in the returned objective value. A gap at most ϵ is an objective certificate that the returned value is within ϵ of the minimum. For infeasible constraints, attainment implies $V^\star > 0$.

The energy E is convex with a Lipschitz gradient. If L_E is a gradient Lipschitz constant, gradient descent with $0 < \tau < 2/L_E$ converges to a minimizer. For feasible constraints this minimizer

satisfies every relation. A linear rate additionally requires an error bound such as the Polyak-Łojasiewicz inequality in Appendix C.3.

At a finite iterate, the certificate compares primal and dual values for the same fixed sets, weights, and selected branches.

Sensitivity to relation edits. For an anchored convex optimum with uniform curvature and graph-sparse interactions, the effect of a relation edit decays exponentially with graph distance from the edited edge. The anchoring penalty discourages unnecessary movement because a small violation alone can correspond to many layouts that satisfy the edited graph. We state the conditions for this decay, then distinguish optimum sensitivity from information propagation in the learned network.

Theorem 4 (Local sensitivity of an anchored convex optimum)**.** *Let $F(x) = E(x) + \frac{\mu}{2}\|x - x^{\mathrm{ref}}\|^2$ with reference layout x^{ref} and anchor strength $\mu > 0$. Let F' change only edge e_0, keeping the same reference, and let $\hat{x}, \hat{x}'$ minimize the two objectives. Assume F' has a Lipschitz gradient and a graph-sparse Hessian, with nonzero blocks only between interacting nodes and $mI \preceq \nabla^2 F' \preceq MI$ almost everywhere between the minimizers, for $0 < m \leq M$. Then*

$$\|\hat{x}_i - \hat{x}'_i\| \leq C\beta^{d_{\mathcal{G}}(i,e_0)}, \qquad \beta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}, \quad \kappa = M/m, \tag{9}$$

where $d_{\mathcal{G}}(i, e_0)$ is the shortest graph distance from node i to either endpoint of the edited edge, measured on the union of the interaction graphs before and after the edit. The constant C depends on curvature bounds and the gradient change at the edge. The bound follows from decay in sparse SPD matrix inverses (Demko et al., 1984; Benzi & Golub, 1999; Rebeschini & Tatikonda, 2019).

A complementary propagation bound applies to an L-layer local network run for T explicit integration steps. With identical initial noise and only local auxiliary operations, an edit cannot affect objects beyond LT hops. Global consensus, recentering, branch selection, or further optimization can extend that range. The experiments measure movement after edits in the complete pipeline, while the theorem identifies conditions for exponential decay at an anchored convex optimum.

4 Experiments

We evaluated TOSCA in three complementary settings. The main comparison used SG-FRONT, the 3D-FRONT-based scene-graph benchmark used by CommonScenes and EchoScene (Zhai et al., 2023; 2024), to assess rendered-image fidelity and relation satisfaction (Tables 1 and 2). Appendix F tests replication after retraining on Inf3DLayout, the procedural component of the multi-source M3DLayout dataset (Zhang et al., 2026). Appendix G evaluates fidelity with the released InstructScene (Lin & Mu, 2024) layout-generation pipeline and includes published FreeScene (Bai et al., 2025) results as context (Table 9). These settings test graph-conditioned generation, replication across layout sources, and adaptation to an external evaluation pipeline, respectively. Appendix H collects published SG-FRONT reference results and diagnostics of annotations, scoring rules, and training overlap. Ablation, conflict, and editing experiments assess the method's design choices (Appendix I). Shared evaluation procedures appear in Appendix E, with experiment-specific settings beside their results.

4.1 Shared Evaluation Setup

We used SG-FRONT (Zhai et al., 2023) bedroom, living, and dining scenes. To compare graph-conditioned layouts with a common mesh library and renderer, we used EchoLayout (Zhai et al., 2024), the authors' layout-only variant of EchoScene. EchoLayout predicts object boxes for mesh retrieval, whereas full EchoScene also generates object shapes. We also included the officially released full EchoScene checkpoint as a reference in the relation results and qualitative comparisons.[1]

We retrained EchoLayout, category-only MMGDreamer (Yang et al., 2025), and the InstructScene (Lin & Mu, 2024) layout decoder on the same per-room splits with a common graph input. MMGDreamer was conditioned on the supplied graph, encoding object category labels through the authors' text-conditioning path with image conditioning disabled. For InstructScene, we supplied the

[1] Its training overlaps the living and dining test splits, so we evaluated its relations on overlap-free subsets and excluded its FID from the full-split ranking (Appendix H.4).

Table 1: **Rendered-image fidelity on SG-FRONT.** Generators were retrained as specified in Sec. 4.1. KID is reported in units of 10^{-3}. Bold marks the best reported generator for each metric.

	Bedroom		Living		Dining	
Source	FID↓	KID (10^{-3})↓	FID↓	KID (10^{-3})↓	FID↓	KID (10^{-3})↓
EchoLayout (Zhai et al., 2024)	38.4	0.15	44.0	3.31	47.4	4.12
MMGDreamer (Yang et al., 2025)	38.6	0.16	44.3	3.54	48.4	4.21
InstructScene (Lin & Mu, 2024)	31.3	0.12	37.5	2.78	36.2	3.31
TOSCA (ours)	**24.8**	**0.08**	**33.5**	**1.95**	**33.1**	**2.24**

Table 2: **Relation satisfaction on SG-FRONT.** Relation groups and aggregates are defined in Sec. 4.3. Annotated layouts measure agreement between dataset relations and the geometric evaluator. Released EchoScene was evaluated on overlap-free subsets, while other generators used full splits. Bold marks the best generator, and a hyphen marks an unreported Joint score.

	Relation-specific satisfaction						Aggregate satisfaction		
Layout source	LR↑	FB↑	BS↑	TS↑	Close↑	Sym.↑	Micro↑	Macro↑	Joint↑
Bedroom									
EchoLayout (Zhai et al., 2024)	.981	.986	.979	.975	.850	.550	.973	.931	.519
EchoScene (Zhai et al., 2024)	.986	.982	.976	.975	.823	.500	.971	.923	.685
MMGDreamer (Yang et al., 2025)	.984	.989	.982	.975	.850	.500	.975	.928	-
InstructScene (Lin & Mu, 2024)	.984	.993	.978	.969	.854	.725	.975	.947	-
TOSCA (ours)	**.999**	**1.00**	**1.00**	**.992**	**.986**	**.975**	**.997**	**.994**	**.801**
Annotated layouts	*.99*	*1.0*	*1.0*	*1.0*	*.89*	*1.0*	*.993*	*.988*	*.82*
Living									
EchoLayout (Zhai et al., 2024)	.984	.989	.939	.949	.640	.168	.953	.863	.047
EchoScene (Zhai et al., 2024)	.988	.996	.957	.959	.660	.400	.965	.896	.401
MMGDreamer (Yang et al., 2025)	.988	.992	.955	.960	.648	.216	.962	.877	-
InstructScene (Lin & Mu, 2024)	.986	.999	.947	.932	.730	.408	.960	.897	-
TOSCA (ours)	**.997**	**1.00**	**1.00**	**.990**	**.950**	**.883**	**.995**	**.982**	**.449**
Annotated layouts	*.99*	*1.0*	*1.0*	*1.0*	*.76*	*1.0*	*.990*	*.978*	*.35*
Dining									
EchoLayout (Zhai et al., 2024)	.981	.990	.934	.942	.700	.291	.950	.879	.051
EchoScene (Zhai et al., 2024)	.984	.992	.961	.966	.775	.531	.963	.919	.316
MMGDreamer (Yang et al., 2025)	.982	.993	.937	.929	.691	.313	.950	.879	-
InstructScene (Lin & Mu, 2024)	.985	.998	.950	.932	.779	.676	.961	.926	-
TOSCA (ours)	**.996**	**1.00**	**1.00**	**.991**	**.950**	**.945**	**.995**	**.988**	**.414**
Annotated layouts	*.98*	*1.0*	*1.0*	*1.0*	*.79*	*1.0*	*.990*	*.980*	*.41*

annotated test graph directly to the layout decoder and sampled appearance tokens from its prior, bypassing text-conditioned graph generation. All methods used the same released mesh library and renderer. Appendix E.2 distinguishes the baseline configurations, and Appendix E.4 specifies sampling and aggregation for the main comparisons. Sections 4.2 and 4.3 separately examine rendered appearance and satisfaction of the input graph.

4.2 Rendered-Image Fidelity

This evaluation tests whether generated layouts reproduce the distribution of rendered scenes. Fréchet Inception Distance (FID, Heusel et al., 2017) and Kernel Inception Distance (KID, Bińkowski et al., 2018) compare generated and held-out ground-truth (GT) image-feature distributions, with lower values indicating closer distributional agreement. We fixed meshes, rendering, and test references across generators. We averaged FID and KID for each retrained generator over three training seeds and three sampling runs per seed, as shown in Table 1. TOSCA has the lowest mean FID and KID among the evaluated full-split generators in every room, indicating closer agreement with the held-out image-feature distribution under the shared evaluation protocol.

4.3 Relation Satisfaction

This evaluation tests whether generated geometry realizes the requested relations, a property that neither FID nor KID measures directly. We used the SG-FRONT geometric predicates and relation-group presentation of CommonScenes (Zhai et al., 2023) and EchoScene (Zhai et al., 2024). Specifically, LR, FB, BS, and TS group the inverse pairs left/right, front/behind, bigger/smaller, and taller/shorter. Close and Sym. denote close-by and symmetrical-to. The aggregate scores also in-

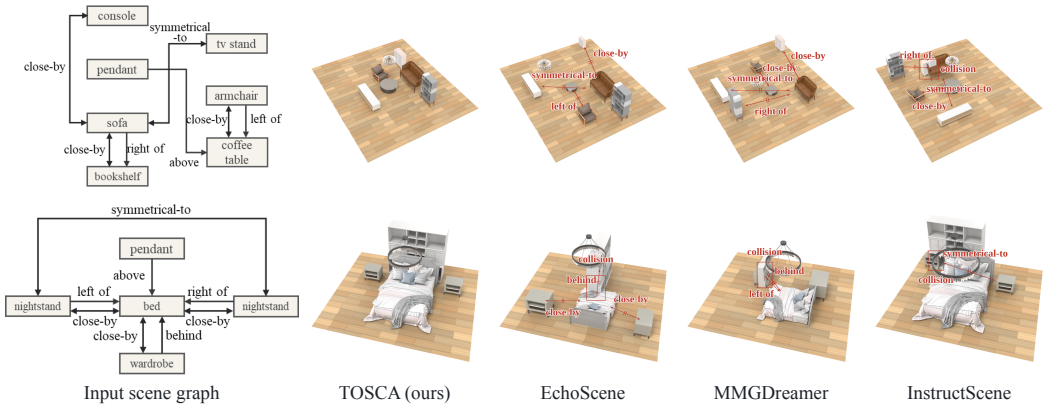


Figure 3: **Layout comparison for a fixed graph per row.** Living room (top) and bedroom (bottom). In the rendered layouts, slashed red arrows mark selected relation violations, and red boxes highlight visible collisions between objects.

clude standing-on, giving eleven scored labels. Above and the three semantic labels lie outside this benchmark score.

To summarize the binary predicate outcomes, *Micro* pools scored edges and *Macro* equally averages the eleven label-specific satisfaction rates, applying established micro- and macro-averaging principles (Koyejo et al., 2015). *Joint* reports the fraction of scenes satisfying every scored request, adapting the exact-match principle (Dembczyński et al., 2010). Macro exposes rare-label failures that Micro can obscure, while Joint tests simultaneous satisfaction. All three are higher-is-better, with definitions and aggregation details in Appendix E.3. The *Annotated layouts* rows apply these predicates to the dataset boxes and relation labels. They measure annotation agreement with the evaluator rather than agreement of a prediction with itself. Stored labels and geometric tests can disagree, so these rows need not reach one. A generator exceeding them satisfies the chosen predicates more often, which is distinct from producing more realistic scenes.

TOSCA achieves the highest Micro and Macro among full-split generators in every room (Table 2). Gains persist when labels receive equal weight and are largest for close-by and symmetrical-to, while directional predicates are already well satisfied by the baselines. Joint improves over EchoLayout, the retrained baseline reporting this measure.

Figure 3 visualizes these relation-specific failure modes in individual scenes. In the living-room example, the marked baseline violations involve close-by and symmetrical-to relations. In the bedroom example, directional or symmetry violations coexist with visible collisions between objects. The nightstands participate in proximity and directional requests relative to the bed and a symmetry request with each other, illustrating the need to assess these relations jointly. TOSCA shows consistent relative arrangements without visible collisions in the displayed views. These examples complement the joint relation-satisfaction results in Table 2. A user study also gives TOSCA the highest Top-1 and Top-2 preference shares for layout plausibility and relation adherence (Appendix I.7, Table 24). The component ablation in Table 18 tests the contributions of allowed regions and final refinement under a shared protocol.

5 Conclusion

We introduced TOSCA, which uses transport sets, projection residuals, and final refinement to retain valid alternatives and reduce relation violations. Across SG-FRONT room types, TOSCA achieves lower FID and KID and higher benchmark relation satisfaction than the evaluated full-split baselines. It also achieves the lowest FID among evaluated pipelines on the InstructScene layout benchmark. Its certificates identify incompatible difference constraints. The current formulation makes geometric requirements explicit but is limited to a predefined relation vocabulary. Future work will extend transport sets to richer functional and physical constraints while preserving interpretable checks and corrections.

AI USE STATEMENT

In the preparation of this manuscript, Generative AI tools were employed solely as an advanced typing assistant to refine sentence structures and correct grammar. No AI tools were involved in the generation of theoretical concepts, experimental results, or the manipulation of data. The entire underlying research and its novel contributions are entirely the authors' original work

REPRODUCIBILITY STATEMENT

The guarantees specify fixed-frame convexity and anchoring assumptions (Sec. 3.5), with proofs in Appendix C. Reproducing the method requires the relation features and fitted boundaries (Appendix B), network and sampling settings (Appendix D), and rendering and scoring protocols (Appendix E). Each additional benchmark specifies its conversion and evaluation settings with its results (Appendices F and G). The result tables identify the compared splits, model variants, and aggregation.

ETHICS STATEMENT

The method generates indoor scenes for design, simulation, and embodied-AI training, with the societal risks associated with generative modeling. Reporting conflicting constraints helps users understand why a scene cannot be generated as specified.

REFERENCES

Ralph Abboud, İsmail İlkan Ceylan, Thomas Lukasiewicz, and Tommaso Salvatori. BoxE: A box embedding model for knowledge base completion. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Tongyuan Bai, Wangyuanfan Bai, Dong Chen, Tieru Wu, Manyi Li, and Rui Ma. FreeScene: Mixed graph diffusion for 3D scene synthesis from free prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jacob Bamberger, Federico Barbero, Xiaowen Dong, and Michael M. Bronstein. Bundle neural network for message diffusion on graphs. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

Afonso S. Bandeira, Amit Singer, and Daniel A. Spielman. A Cheeger inequality for the graph connection Laplacian. *SIAM Journal on Matrix Analysis and Applications*, 34(4):1611–1630, 2013. doi: 10.1137/120875338.

Heinz H. Bauschke and Patrick L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. CMS Books in Mathematics. Springer, 2nd edition, 2017. doi: 10.1007/978-3-319-48311-5.

Michele Benzi and Gene H. Golub. Bounds for the entries of matrix functions with applications to preconditioning. *BIT Numerical Mathematics*, 39(3):417–438, 1999. doi: 10.1023/A:1022362401426.

Dmitry Berenson, Siddhartha Srinivasa, and James Kuffner. Task space regions: A framework for pose-constrained manipulation planning. *The International Journal of Robotics Research*, 30(12):1435–1460, 2011. doi: 10.1177/0278364910396389.

Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. URL https://openreview.net/forum?id=r1lUOzWCW.

Cristian Bodnar, Francesco Di Giovanni, Benjamin Paul Chamberlain, Pietro Liò, and Michael M. Bronstein. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in GNNs. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Christer Borell. The Brunn-Minkowski inequality in Gauss space. *Inventiones Mathematicae*, 30(2):207–216, 1975. doi: 10.1007/BF01425510.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D data in indoor environments. In *Proceedings of the International Conference on 3D Vision (3DV)*, 2017.

Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Jeongjun Choi, Yeonsoo Park, and H. Jin Kim. SceneNAT: Masked generative modeling for language-guided indoor scene synthesis. *arXiv preprint arXiv:2601.07218v2*, 2026a.

Yoonhyuk Choi, Jiho Choi, and Jiwoo Kang. Gauge-equivariant graph networks via self-interference cancellation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026b.

Yoonhyuk Choi, Jiho Choi, Taewook Ko, JongWook Kim, and Chong-Kwon Kim. Sheaf graph neural networks via PAC-Bayes spectral optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026c.

Jacob K. Christopher, Stephen Baek, and Ferdinando Fioretto. Constrained synthesis with projected diffusion models. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 3rd edition, 2009.

Rob Cornish, Anthony Caterini, George Deligiannidis, and Arnaud Doucet. Relaxing bijectivity constraints with continuously indexed normalising flows. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

Krzysztof Dembczyński, Weiwei Cheng, and Eyke Hüllermeier. Bayes optimal multilabel classification via probabilistic classifier chains. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2010. URL `https://icml.cc/Conferences/2010/papers/589.pdf`.

Stephen Demko, William F. Moss, and Philip W. Smith. Decay rates for inverses of band matrices. *Mathematics of Computation*, 43(168):491–499, 1984. doi: 10.1090/S0025-5718-1984-0758197-9.

Helisa Dhamo, Fabian Manhardt, Nassir Navab, and Federico Tombari. Graph-to-3D: End-to-end generation and manipulation of 3D scenes using scene graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006. doi: 10.1016/j.patrec.2005.10.010.

Weixi Feng, Wanrong Zhu, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. LayoutGPT: Compositional visual planning and generation with large language models. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, and Hao Zhang. 3D-FRONT: 3D furnished rooms with layOuts and semaNTics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Tingran Gao, Jacek Brodzki, and Sayan Mukherjee. The geometry of synchronization problems and learning group actions. *Discrete & Computational Geometry*, 65(1):150–211, 2021. doi: 10.1007/s00454-019-00100-2.

Thomas Gebhart, Jakob Hansen, and Paul Schrater. Knowledge sheaves: A sheaf-theoretic framework for knowledge graph embedding. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2023.

Zhifeng Gu, Yuqi Wang, and Bing Wang. R^3L: Reasoning 3D layouts from relative spatial relations. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3D scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-Softmax. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. URL https://arxiv.org/abs/1611.01144.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Seok-Young Kim, Dooyoung Kim, Woojin Cho, Hail Song, Suji Kang, and Woontack Woo. SceneLinker: Compositional 3D scene generation via semantic scene graph from RGB sequences. *IEEE Transactions on Visualization and Computer Graphics*, 32(5):3125–3135, 2026. doi: 10.1109/TVCG.2026.3679893.

Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: Exact likelihood generative learning for symmetric densities. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

Oluwasanmi Koyejo, Nagarajan Natarajan, Pradeep Ravikumar, and Inderjit S. Dhillon. Consistent multilabel classification. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2015. URL https://papers.nips.cc/paper_files/paper/2015/hash/85f007f8c50dd25f5a45fca73cad64bd-Abstract.html.

Matt Kraning, Eric Chu, Javad Lavaei, and Stephen Boyd. Dynamic network energy management via proximal message passing. *Foundations and Trends in Optimization*, 1(2):73–126, 2014. doi: 10.1561/2400000002.

Chenguo Lin and Yadong Mu. InstructScene: Instruction-driven 3D indoor scene synthesis with semantic graph prior. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Guan-Horng Liu, Tianrong Chen, Evangelos Theodorou, and Molei Tao. Mirror diffusion models for constrained and watermarked generation. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019. URL https://arxiv.org/abs/1711.05101.

Aaron Lou and Stefano Ermon. Reflected diffusion models. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

Léopold Maillard, Nicolas Sereyjol-Garros, Tom Durand, and Maks Ovsjanikov. DeBaRA: Denoising-based 3D room arrangement generation. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Seongrae Noh, SeungWon Seo, Gyeong-Moon Park, and HyeongYeop Kang. Edit-As-Act: Goal-regressive planning for open-vocabulary 3D indoor scene editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Art B. Owen and Dean Eckles. Bootstrapping data arrays of arbitrary order. *The Annals of Applied Statistics*, 6(3):895–927, 2012. doi: 10.1214/12-AOAS547.

Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. ATISS: Autoregressive transformers for indoor scene synthesis. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Alexander Raistrick, Lingjie Mei, Karhan Kayan, David Yan, Yiming Zuo, Beining Han, Hongyu Wen, Meenal Parakh, Stamatis Alexandropoulos, Lahav Lipson, Zeyu Ma, and Jia Deng. Infinigen Indoors: Photorealistic indoor scenes using procedural generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Patrick Rebeschini and Sekhar Tatikonda. Locality in network optimization. *IEEE Transactions on Control of Network Systems*, 6(2):487–500, 2019. doi: 10.1109/TCNS.2018.2843164.

Jochen Renz and Bernhard Nebel. On the complexity of qualitative spatial reasoning: A maximal tractable fragment of the Region Connection Calculus. *Artificial Intelligence*, 108(1-2):69–123, 1999. doi: 10.1016/S0004-3702(99)00002-8.

David M. Rosen, Luca Carlone, Afonso S. Bandeira, and John J. Leonard. SE-Sync: A certifiably correct algorithm for synchronization over the special Euclidean group. *The International Journal of Robotics Research*, 38(2-3):95–125, 2019. doi: 10.1177/0278364918784361.

Antonio Ruiz, Tao Wu, Andrew Melnik, Qing Cheng, Xuqin Wang, Lu Liu, Yongliang Wang, Yanfeng Zhang, and Helge Ritter. GeoSceneGraph: Geometric scene graph diffusion model for text-guided 3D indoor scene synthesis. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XI-4-2026:293–301, 2026. doi: 10.5194/isprs-annals-XI-4-2026-293-2026.

Antoine Salmona, Valentin De Bortoli, Julie Delon, and Agnès Desolneux. Can push-forward generative models fit multimodal distributions? In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. doi: 10.1002/j.1538-7305.1948.tb01338.x. Part II appeared in 27(4), 623-656.

Amit Singer. Angular synchronization by eigenvectors and semidefinite programming. *Applied and Computational Harmonic Analysis*, 30(1):20–36, 2011. doi: 10.1016/j.acha.2010.02.001.

Spiros Skiadopoulos and Manolis Koubarakis. On the consistency of cardinal direction constraints. *Artificial Intelligence*, 163(1):91–135, 2005. doi: 10.1016/j.artint.2004.10.010.

Fan-Yun Sun, Weiyu Liu, Siyi Gu, Dylan Lim, Goutam Bhat, Federico Tombari, Manling Li, Nick Haber, and Jiajun Wu. LayoutVLM: Differentiable optimization of 3D layout via vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE: Knowledge graph embedding by relational rotation in complex space. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. DiffuScene: Denoising diffusion models for generative indoor scene synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017. URL https://papers.nips.cc/paper/2017/hash/7a98af17e63a0ac09ce2e96d03992fbc-Abstract.html.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. URL https://arxiv.org/abs/1710.10903.

Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3D semantic scene graphs from 3D indoor reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Ruijie Xu, Xinnan Zhu, Jiayu Ying, Daoguo Dong, Yuzhou Ji, and Xin Tan. MUSE: Agentic 3D scene authoring via memory-grounded incremental requirement satisfaction. *arXiv preprint arXiv:2606.14168*, 2026.

Yandan Yang, Baoxiong Jia, Peiyuan Zhi, and Siyuan Huang. PhyScene: Physically interactable 3D scene synthesis for embodied AI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Yongyi Yang, Tang Liu, Yangkun Wang, Jinjing Zhou, Quan Gan, Zhewei Wei, Zheng Zhang, Zengfeng Huang, and David Wipf. Graph neural networks inspired by classical iterative algorithms. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, Chris Callison-Burch, Mark Yatskar, Aniruddha Kembhavi, and Christopher Clark. Holodeck: Language guided generation of 3D embodied AI environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Zhifei Yang, Keyang Lu, Chao Zhang, Jiaxing Qi, Hanqi Jiang, Ruifei Ma, Shenglin Yin, Yifan Xu, Mingzhe Xing, Zhen Xiao, Jieyi Long, Xiangde Liu, and Guangyao Zhai. MMGDreamer: Mixed-modality graph for geometry-controllable 3D indoor scene generation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2025.

Zhifei Yang, Guangyao Zhai, Keyang Lu, YuYang Yin, Chao Zhang, Zhen Xiao, Jieyi Long, Nassir Navab, and Yikai Wang. FlowScene: Style-consistent indoor scene generation with multimodal graph rectified flow. *arXiv preprint arXiv:2603.19598*, 2026.

Zhutian Yang, Jiayuan Mao, Yilun Du, Jiajun Wu, Joshua B. Tenenbaum, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Compositional diffusion-based continuous constraint solvers. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2023.

Jason Yim, Brian L. Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. SE(3) diffusion model with application to protein backbone generation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

Guangyao Zhai, Evin Pınar Örnek, Shun-Cheng Wu, Yan Di, Federico Tombari, Nassir Navab, and Benjamin Busam. CommonScenes: Generating commonsense 3D indoor scenes with scene graph diffusion. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Guangyao Zhai, Evin Pınar Örnek, Dave Zhenyu Chen, Ruotong Liao, Yan Di, Nassir Navab, Federico Tombari, and Benjamin Busam. EchoScene: Indoor scene generation via information echo over scene graph diffusion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

Xitong Zhang, Yixuan He, Nathan Brugnone, Michael Perlmutter, and Matthew Hirn. MagNet: A neural network for directed graphs. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2021a.

Yiheng Zhang, Zhuojiang Cai, Mingdao Wang, Meitong Guo, Tianxiao Li, Li Lin, and Yuwang Wang. M3DLayout: A multi-source dataset of 3D indoor layouts and structured descriptions for 3D generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Zhanqiu Zhang, Jie Wang, Jiajun Chen, Shuiwang Ji, and Feng Wu. ConE: Cone embeddings for multi-hop reasoning over knowledge graphs. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2021b.

Appendix

CONTENTS OF THE APPENDIX

A Extended Related Work **17**

A.1 Scene-Graph-to-3D Generation 17

A.2 Object-Set Generation, Planning, and Geometric Guidance 17

A.3 Geometric Relations, Structured Message Passing, and Equivariance 18

A.4 Unrolled Optimization and Scene Editing 18

B Relation Definitions and Fitted Transport Sets **20**

B.1 Coordinates and Relation Catalog 20

B.2 Constraint Geometry and Benchmark Scoring 21

B.3 Parameter Initialization and Regularization 21

C Theoretical Results and Proofs **23**

C.1 Single-Pose Constraints and Feasible Alternatives 23

C.2 Feasibility Certificates and Relation Completion 24

C.3 Objective Bounds, Convergence, and Fixed Points 25

C.4 Sensitivity to Relation Edits 26

C.5 Coordinate Equivariance 26

D Model, Training, and Sampling Details **27**

D.1 State Coordinates and Projection Operators 27

D.2 Network Architecture and Orientation Prediction 27

D.3 Training Objective and Optimization 28

D.4 Sampling, Refinement, and Constraint Checks 29

E Evaluation Setup and Metrics **30**

E.1 Datasets and Data Preparation 30

E.2 Baseline Configurations 30

E.3 Rendering and Evaluation Metrics 31

E.4 Sampling and Aggregation 32

F Replication on Inf3DLayout **33**

F.1 Dataset Conversion and Evaluation Setup 33

F.2 Generation Quality and Constraint Satisfaction 33

G Evaluation on the InstructScene Layout Benchmark **35**

G.1 Benchmark Setup and Adaptation 35

G.2 Relation Conversion Checks 35

G.3 Generation Results 36

H SG-FRONT Reference Results and Evaluation Diagnostics **37**

H.1 Published SG-FRONT Results 37

H.2 Annotation Agreement and Benchmark Reference Frames 37

H.3 Fitted Catalog Coverage and Annotation Feasibility 38

H.4 Training Overlap in the Released EchoScene Checkpoint 39

H.5 Position Matching by Layout-Twin Status 40

I Ablations and Additional Evaluations **42**

I.1 Component and Sampling Ablations 42

I.2 Orientation and Size Controls 43

I.3 Conflict Detection on SG-FRONT-Conflict 43

I.4 Relation Editing and Locality 44

I.5 Footprint Overlap 46

I.6 Qualitative Comparisons 46

I.7 User Study 49

A Extended Related Work

The related work addresses several operational problems in scene generation. A relation can admit multiple valid placements, equivalent descriptions should retain the same meaning, a failed request can reflect either incompatible constraints or incomplete optimization, and a local edit should preserve unrelated objects. Table 3 connects these problems to existing mechanisms and the scope of our analysis. Architectural descriptions identify how a method addresses a problem, while the controlled comparisons in Sec. 4 assess generated-layout behavior.

A.1 Scene-Graph-to-3D Generation

Graph-to-3D (Dhamo et al., 2021), CommonScenes (Zhai et al., 2023), EchoScene (Zhai et al., 2024), and MMGDreamer (Yang et al., 2025) encode relation labels as learned relation embeddings and aggregate messages computed from subject-relation-object features through a triplet graph convolutional network (triplet-GCN). Graph-to-3D explicitly models the one-to-many mapping from relations to layouts and evaluates diversity across repeated samples from the same graph. EchoScene and MMGDreamer update graph conditions using the current denoising states, and FlowScene (Yang et al., 2026) carries this information-exchange design into rectified flow. These methods use learned relation embeddings to shape a conditional distribution of scenes. Stochastic diversity and geometric satisfaction are separate requirements. A layout that differs from other samples can still fail to realize its requested relations, and our controlled comparisons reveal relation-specific failures that aggregate satisfaction can obscure (Table 2). To correct relation violations while retaining multiple valid placements, TOSCA supplements learned conditioning with geometric checks and corrections based on explicit admissible sets and projection residuals. Our component ablations assess the contribution of the transport-set design within TOSCA.

EchoScene's authors also report EchoLayout (Zhai et al., 2024), a layout-only variant that predicts object boxes and retrieves meshes, while their full model jointly learns layout and shape generation. Our controlled layout comparisons used retrained EchoLayout and a shared rendering pipeline (Sec. 4.1).

FreeScene (Bai et al., 2025) uses a graph-aware diffusion transformer with a symmetric matrix of edge types and inverse relations following dataset conventions. The operational question is whether exchanging the subject and object while reversing the relation leaves the accepted geometry unchanged. If the two descriptions accept different geometries, an arbitrary choice of edge direction changes the request. TOSCA ties inverse geometric constraints so that equivalent edge descriptions admit the same object arrangements, while layout evaluation checks whether the generated object geometry satisfies the requested relations. InstructScene (Lin & Mu, 2024) generates a semantic graph through discrete diffusion and then uses a graph Transformer to denoise layouts conditioned on node and edge features. Its stochastic layout decoder also generates different arrangements for the same graph. Evaluating the final layout is essential because Ruiz et al. (2026) report lower instruction recall there than on the intermediate graph. We therefore distinguish graph accuracy from layout accuracy. For the baseline comparison, we retrained InstructScene's second-stage sg2sc decoder with the ground-truth graph.

A.2 Object-Set Generation, Planning, and Geometric Guidance

ATISS (Paschalidou et al., 2021) and DiffuScene (Tang et al., 2024) generate unordered object sets, while DeBaRA (Maillard et al., 2024) focuses on spatially aware layouts. Language and vision-language planners, including LayoutGPT (Feng et al., 2023), Holodeck (Yang et al., 2024b), LayoutVLM (Sun et al., 2025), and R3L (Gu et al., 2026), improve scene validity through repeated revision or explicit optimization. R3L identifies two errors that can arise when relations are combined, namely *semantic drift*, where a directional relation changes meaning across coordinate frames, and *metric drift*, where displacement errors accumulate along a chain. These errors motivate checking combinations of relations. In these planners, constraint correction takes place outside the generative network.

A remaining violation after sampling does not identify whether the requests are incompatible or the optimizer needs further refinement. Physics-aware generation addresses geometric correction through methods such as collision and reachability guidance in PhyScene (Yang et al., 2024a)

and diffusion combined with optimization in SceneDiffuser (Huang et al., 2023). Projected diffusion treats sampling as constrained optimization and evaluates convex and nonconvex constraints (Christopher et al., 2024). Mirror (Liu et al., 2023) and reflected (Lou & Ermon, 2023) diffusion provide other ways to keep samples within a domain. Diffusion-CCSP (Yang et al., 2023) addresses constraint solving through learned energies on factor graphs. These approaches provide mechanisms for correction, with guarantees tied to their formulations. For explicit scene-relation sets, our infeasibility certificates identify incompatible requests, while objective bounds quantify the improvement still possible under the fixed affine convex problem in Sec. 3.5. The EchoScene guidance control separately measures the effect of correcting baseline outputs under our scene protocol.

A.3 Geometric Relations, Structured Message Passing, and Equivariance

Geometric knowledge-graph embeddings, including RotatE (Sun et al., 2019), BoxE (Abboud et al., 2020), and ConE (Zhang et al., 2021b), represent relations through geometric objects and algebraic rules. However, their scores do not directly specify valid object layouts or generate scenes. Task Space Regions (Berenson et al., 2011) specify allowed pose sets manually for planning. Group synchronization (Singer, 2011; Bandeira et al., 2013; Rosen et al., 2019; Gao et al., 2021) estimates consistent poses from noisy relative measurements and can certify optimality.

Neural sheaf diffusion (Bodnar et al., 2022), Bundle Neural Networks (Bamberger et al., 2025), and Knowledge Sheaves (Gebhart et al., 2023) learn linear maps between node features for discriminative tasks. Related approaches use magnetic Laplacians (Zhang et al., 2021a) or complex phase rotations with interference cancellation (Choi et al., 2026b) to transform messages using commuting, or abelian, operations. Sheaf graph neural networks also combine optimal-transport maps with PAC-Bayes calibration, which uses probabilistic generalization bounds, for node classification (Choi et al., 2026c). These message-passing maps operate on learned features, whereas TOSCA transport sets constrain geometric relation features. Theorem 1 distinguishes exact equality to one relative pose from an admissible region. Synchronization and simultaneous localization and mapping (SLAM) methods can incorporate inequalities, and a linear map with a nontrivial kernel leaves some directions unconstrained. The theorem therefore applies only when a model imposes an exact single-pose constraint.

Equivariant flows and diffusion models (Köhler et al., 2020; Yim et al., 2023) transform their outputs consistently with transformations of the input. GeoSceneGraph (Ruiz et al., 2026) uses diffusion equivariant to three-dimensional Euclidean transformations, but removes some relation classes and reports lower instruction recall in its comparison. Our formulation preserves equivariance when the features, transport sets, metrics, and reference frames transform together (Proposition 9). The benchmark's directional and symmetrical-to scoring rules instead refer to specified world axes and the scene origin (Appendix B).

A.4 Unrolled Optimization and Scene Editing

Editing requires both realizing the changed request and preserving the surrounding scene. EchoScene (Zhai et al., 2024) supports relation changes and object additions, Edit-As-Act (Noh et al., 2026) studies edit success and preservation, and MUSE (Xu et al., 2026) uses an agent for local changes. An optimization view relates this practical preservation question to the propagation of a changed constraint. TWIRLS (Yang et al., 2021) and proximal message passing (Kraning et al., 2014) view graph aggregation as energy minimization. TOSCA applies this idea to projection residuals, and Theorem 4 bounds the response of anchored optima under its curvature and sparsity assumptions. The editing experiment evaluates finite generated outputs separately from that conditional bound.

Table 3: **Operational problems and related approaches.** Each row begins with a scene-generation requirement and the consequence that makes it useful. The remaining columns describe established mechanisms and the scope of TOSCA's treatment. This qualitative comparison concerns formulations. Generated-layout quality is evaluated in Sec. 4.

Operational problem	Established approaches	TOSCA treatment and scope
Preserve valid alternatives so that an underspecified request allows distinct layouts.	Stochastic graph-conditioned generators model multiple layouts. Task Space Regions (Berenson et al., 2011) specify admissible poses, and knowledge-graph regions represent allowed feature regions.	Relation sets retain geometric freedom and reflection branches retain discrete alternatives. The single-pose result concerns exact pose equalities, while measured diversity requires empirical evaluation.
Keep equivalent edge descriptions consistent so that representation choices do not change the request.	Dataset inverse conventions and geometric relation algebra encode relationships between labels. Equivariant models address compatible coordinate transformations.	Tied inverse sets preserve edge-reversal semantics. Coordinate equivariance additionally requires transforming features, metrics, and reference frames together.
Distinguish incompatible requests from incomplete optimization so that failure has an actionable diagnosis.	Projected diffusion (Christopher et al., 2024) gives feasibility guarantees for convex constraints. Diffusion-CCSP (Yang et al., 2023) optimizes learned constraint energies. Synchronization certifies specified pose-estimation objectives.	Infeasibility certificates and objective bounds address fixed affine convex relation problems. Alternative branches require checking all assignments to certify infeasibility of their union.
Preserve surrounding objects during edits so that satisfying a new request does not unnecessarily move the scene.	EchoScene (Zhai et al., 2024) supports graph edits, and Edit-As-Act (Noh et al., 2026) evaluates edit success and preservation. Energy-based message passing relates updates to optimization.	Anchored-optimum sensitivity decays with graph distance under curvature and sparsity conditions. Finite-sample edit success and displacement are evaluated separately.

B Relation Definitions and Fitted Transport Sets

B.1 Coordinates and Relation Catalog

Table 4 lists all fifteen SG-FRONT relation labels (Zhai et al., 2023) and gives the fitted transport-set criterion for each. The rows cover twelve spatial labels and three semantic labels. The left, close-by, and symmetrical-to examples in Fig. 2(a) illustrate a half-space, a box, and a choice between reflections of planar object centers across the scene y- or x-axis. The remaining rows also constrain vertical position, volume, height, floor agreement, and latent attributes.

Let p_i be object i's planar center, $u_i = (\ell_i, h_i, w_i)$ its full extent vector, and z_i its elevation. Table 4 uses $t = p_j - p_i$, $s = \log u_j - \log u_i$, and $\Delta z = z_j - z_i$ for planar displacement, log-size difference, and elevation difference. The height component of s is s_h, and $\mathbf{1}^\top s$ is the log-volume difference. The catalog's planar x and y coordinates correspond to dataset world z and x, respectively, while elevation corresponds to dataset world y. Thus the planar full extents are $u_{i,x} = \ell_i$ and $u_{i,y} = w_i$. These coordinate differences generally differ from the translation component of the $\mathrm{SE}(2)$ logarithm when yaws differ.

Directional relations and size comparisons use a fitted margin $m_r > 0$, with inverse-pair parameters tied as in Equation 2. Standing-on uses the subject's bottom height b_i relative to the floor and tolerance $\tau \geq 0$. Close-by uses a surface-gap allowance $R \geq 0$, with $a \in \{x, y\}$ indexing planar axes. For symmetrical-to, $J_1 = \mathrm{diag}(-1, 1)$ reflects a planar position across the y-axis, while $J_2 = \mathrm{diag}(1, -1)$ reflects it across the x-axis. Both axes pass through the scene origin, and either branch satisfies the fitted constraint. The SG-FRONT evaluator provided with EchoScene accepts either axis reflection or a half-turn about the origin when the planar position error is below the benchmark tolerance specified in Appendix E.3 (Zhai et al., 2024). Table 4 uses the two exact axis-reflection constraints as the fitted sets for generation. The semantic rows use the shape-feature difference $\delta z^{\mathrm{sh}} = z_j^{\mathrm{sh}} - z_i^{\mathrm{sh}}$ and linear selectors P^{sty}, P^{mat}, and P^{cat} for style, material, and category coordinates. These latent equalities are separate from geometric layout scoring.

Table 4: **Complete SG-FRONT relation catalog and fitted transport sets.** All twelve spatial labels and three semantic labels have explicit membership criteria. The criteria define generation constraints, while benchmark predicates determine reported relation scores (Appendix E.3). For generation, symmetrical-to uses two exact reflections of planar object centers across the scene coordinate axes. Proximity bounds hold physical extents fixed for convex position solves.

Relation	Fitted-set membership criterion	Set geometry
left	$t_x \geq m_r$	half-space
right	$t_x \leq -m_r$	half-space
front	$t_y \geq m_r$	half-space
behind	$t_y \leq -m_r$	half-space
above	$\Delta z \leq -m_r$	half-space
bigger-than	$\mathbf{1}^\top s \leq -m_r$	half-space
smaller-than	$\mathbf{1}^\top s \geq m_r$	half-space
taller-than	$s_h \leq -m_r$	half-line
shorter-than	$s_h \geq m_r$	half-line
standing-on	$\lvert b_i \rvert \leq \tau$	slab
close-by	$\lvert t_a \rvert \leq \frac{1}{2}(u_{i,a} + u_{j,a}) + R$ for both axes	box
symmetrical-to	$p_j - J_b p_i = 0$ for some $b \in \{1, 2\}$	affine branches
same style as	$P^{\mathrm{sty}} \delta z^{\mathrm{sh}} = 0$	subspace
same material as	$P^{\mathrm{mat}} \delta z^{\mathrm{sh}} = 0$	subspace
same super-category as	$P^{\mathrm{cat}} \delta z^{\mathrm{sh}} = 0$	subspace

For Euclidean or compatible diagonal metrics, directional and height comparisons use scalar clamping, standing-on clamps bottom height to its slab, and close-by clamps the two planar coordinates. The log-volume half-space projects along $\mathbf{1}/\sqrt{3}$ in the Euclidean metric. Reflection branches and

latent equalities use affine projection, with branch selection taking the smaller reflection residual. Appendix D.1 gives the metric-dependent operators.

B.2 Constraint Geometry and Benchmark Scoring

The available guarantees depend on the relation type. World-axis directional relations and log-size comparisons are affine difference constraints, for which negative-cycle detection is complete. Floor height constrains one object relative to a fixed floor. Proximity boxes are convex in position when object sizes are fixed. Replacing a box with a rotation-invariant Euclidean ball would change the proximity rule. Each symmetrical-to constraint becomes affine once its frame and reflection branch are fixed. Collision avoidance combines alternative separating directions, so it does not form one convex cone. Latent equalities also define affine sets. The spatial transport sets constrain positions and sizes, while the orientation channel predicts object yaw.

Collision avoidance illustrates why fixing a discrete choice matters to convexity. A fixed separating direction gives an affine separation constraint, while choosing among directions produces a union of feasible sets. Tables 23 and 7 assess generated geometry through footprint overlap, which counts projected overlap independently of the input relation graph.

For symmetrical-to, the fitted model's two branches reflect planar positions across the axes through the benchmark origin. Branch $b \in \{1, 2\}$ uses feature $\eta_{e,b} = p_j - J_b p_i$ and transport set $S_{e,b} = \{0\}$. Its projection residual is $\eta_{e,b}$. Since $J_b^2 = I$, the reversed feature satisfies $p_i - J_b p_j = -J_b \eta_{e,b}$. Edge reversal therefore preserves branch membership and the Euclidean residual norm, while a general metric must transform with J_b. Object extents and elevation are constrained by their respective catalog relations.

Reflection about any fixed reference line and point symmetry about any fixed reference point also give affine constraints in the object positions. The fitted catalog uses line reflections across the coordinate axes through the scene origin.

Fitted-set membership and benchmark-rule satisfaction answer different geometric questions. For close-by, the fitted box $|t_a| \le \frac{1}{2}(u_{i,a} + u_{j,a}) + R$ bounds the gap between axis-aligned footprint intervals along each planar axis. Since $u_{i,a}$ and $u_{j,a}$ are full extents, their halves are the distances from each center to its interval endpoints. The nonnegative interval gap is $\max\{0, |t_a| - \frac{1}{2}(u_{i,a} + u_{j,a})\}$, so bounding it by R yields the fitted criterion. The benchmark instead uses EchoScene's minimum-corner-distance scoring rule specified in Appendix E.3 (Zhai et al., 2024). Diagonal offsets and overlapping footprints can therefore receive different labels. Table 14 measures membership in the fitted box, while Table 12 measures satisfaction of the benchmark scoring rule. We use this box to retain a convex constraint in position at fixed sizes. With a compatible diagonal metric, its projection clamps each coordinate to the fitted interval. The box depends on the scene axes, and jointly optimizing positions and exponentiated log-sizes may make the problem nonconvex.

The distinction also applies to other relation types. The released EchoScene evaluator (Zhai et al., 2024) compares top elevations for taller-than and shorter-than, whereas the fitted sets compare object heights through log-extents. Its standing-on predicate compares the two objects' bottom elevations, which becomes a floor-height test when the reference object is the floor. These comparisons explain why a small fitted-set residual and a high benchmark relation score describe different properties. Appendix E.3 specifies scoring subsets, predicates, and aggregation.

B.3 Parameter Initialization and Regularization

Transport-set fitting used annotated training edges in the catalog coordinates. Validation and test layouts entered only the coverage audit in Table 14. Inverse pairs share one parameter block after reversing the edge and negating the corresponding feature. Category-pair statistics determine category-dependent blocks, with relation-level statistics providing a fallback for category pairs absent from training. Bigger-than uses the sum $\mathbf{1}^\top s$ of log-extent differences because it compares volume. Taller-than uses only s_h, allowing an object to be taller but smaller in volume than another.

Let $Q_p(v)$ be the empirical quantile of a nonempty training sample v for $0 < p \le 1$, defined by its sorted entry at rank $\lceil p|v| \rceil$. We used $q = .98$ to initialize the finite boundary parameters. For a directional or size inequality written as $d_e \ge m_r$, d_e is the signed scalar comparison in Table 4. For

proximity, define the size-adjusted gap $v_e = \max_a\{|t_a| - \frac{1}{2}(u_{i,a} + u_{j,a})\}$. The reference parameters are

$$\widehat{m}_r = \max\{\epsilon_{\text{fit}}, Q_{1-q}(d_e)\}, \qquad \widehat{R} = \max\{0, Q_q(v_e)\}, \qquad \widehat{\tau} = Q_q(|b_i|), \tag{10}$$

where each quantile uses the annotated edges of its relation block and $\epsilon_{\text{fit}} = 10^{-4}$ in the comparison feature's units preserves a positive comparison margin. Lower quantiles initialize lower bounds and upper quantiles initialize upper bounds. The two reflection branches retain their fixed affine equations. A margin loss refines the trainable boundaries around these empirical references (Appendix D.3), while the final coverage is measured directly.

An unrestricted transport set would cover every training pair, so boundary fitting also penalizes departures from its empirical reference. Let $\zeta_r \in \mathbb{R}^{k_r}$ collect a block's trainable margins, radii, or slab widths, let $\widehat{\zeta}_r$ collect the corresponding values from Equation 10, and let s_{rj} be the interquartile range of the statistic used for coordinate j, floored at 10^{-3} in that statistic's units. We define

$$\Omega(S_r) = \frac{1}{k_r} \sum_{j=1}^{k_r} \left(\frac{\zeta_{rj} - \widehat{\zeta}_{rj}}{s_{rj}} \right)^2 . \tag{11}$$

Fixed sets with no trainable boundary receive $\Omega(S_r) = 0$. The sum counts each shared inverse block once. This dimensionless penalty is finite for unbounded half-spaces because it acts on boundary parameters rather than set volume. After each fitting update, margins are clamped to $[\epsilon_{\text{fit}}, \infty)$ and radii and slab widths to $[0, \infty)$. These domain constraints preserve the catalog semantics while the regularizer keeps the fitted boundaries close to the annotated geometry.

C Theoretical Results and Proofs

This appendix groups proofs and related results by topic. The affine convex results use the fixed edge maps, nonempty closed convex sets, positive metrics and weights, reference frames, size bounds, and branch choices specified in Sec. 3.5. Each result states its additional assumptions.

C.1 Single-Pose Constraints and Feasible Alternatives

Proof of Theorem 1. A one-point set $\{\mu_r\}$ has measure zero, so equality to μ_r cannot describe a proximity region with positive volume. For the second claim, $\log(g_u^{-1}g_w) = \mu_r = \log(g_v^{-1}g_w)$ gives $g_u^{-1}g_w = g_v^{-1}g_w$, hence $g_u = g_v$. If a transport set has nonempty interior in the fixed logarithm chart, choose distinct μ_u, μ_v in that set and let $g_u = g_w \exp(-\mu_u)$ and $g_v = g_w \exp(-\mu_v)$. Both edge constraints hold, and injectivity of the chart gives $g_u \neq g_v$. The set-valued constraints therefore preserve distinct valid placements relative to the shared object. □

A different issue arises when several reflection branches for symmetrical-to remain feasible. Their union can contain disconnected components, which a generator must represent if it is to preserve those alternatives. For a deterministic decoder driven by a single Gaussian latent variable, the separation between these regions limits how much probability can remain entirely feasible.

Lemma 5 (Multimodal obstruction adapted from Salmona et al., 2022)**.** *Let $F = \bigcup_{c=1}^{m} F_c$ be a union of closed feasible components with $m \geq 2$, each separated from the others by at least $2\delta > 0$. Let $H(\mathcal{G})$ be a graph embedding produced by a message-passing neural network and consider $x = D(z, H(\mathcal{G}))$ with $z \sim \mathcal{N}(0, I)$. Suppose $D(\cdot, h)$ is L-Lipschitz for $L > 0$. If two components each receive probability at least $p \in (0, 1/2]$, then*

$$\Pr[x \notin F] \geq c(p, 2\delta/L) > 0, \tag{12}$$

$$c(p,t) = \min\{\Phi(\Phi^{-1}(p) + t) - p,\ \Phi(t) - \tfrac{1}{2}\}, \tag{13}$$

where Φ is the standard normal cumulative distribution function.

Proof of Lemma 5. We adapt the argument of Salmona et al. (2022) (see also Cornish et al., 2020) to a decoder conditioned on a message-passing neural network (MPNN). Fix $h = H(\mathcal{G})$. The bound holds for every fixed h, so changing the graph embedding does not remove it. Let γ denote standard Gaussian measure on the latent space and set $t = 2\delta/L$. The sets $A_c = D(\cdot, h)^{-1}(F_c)$ contain the latent values that map to each feasible component. For the two components with masses $s_a = \gamma(A_a) \geq p$ and $s_b = \gamma(A_b) \geq p$, we have $s_a + s_b \leq 1$. We can therefore relabel them so that $p \leq s_a \leq \frac{1}{2}$.

First, for $z \in A_a$ and $z' \in A_b$, the outputs satisfy $\|D(z) - D(z')\| \geq \operatorname{dist}(F_a, F_b) \geq 2\delta$. Since D is L-Lipschitz, their inputs must satisfy $\|z - z'\| \geq 2\delta/L = t$. Thus, the enlarged set $A_a^{<t} = \{z \mid \operatorname{dist}(z, A_a) < t\}$ cannot intersect A_b. The same argument applies to every other A_c, since each F_c is at least 2δ from F_a.

Next, by the Gaussian isoperimetric inequality (Borell, 1975), $\gamma(A_a^{<t'}) \geq \Phi(\Phi^{-1}(s_a) + t')$ for every $t' < t$. Letting $t' \uparrow t$ and using continuity of Φ gives $\gamma(A_a^{<t}) \geq \Phi(\Phi^{-1}(s_a) + t)$.

To locate the probability mass added by this enlargement, let $z \in A_a^{<t} \setminus A_a$. There is $z_0 \in \overline{A_a}$ with $\|z - z_0\| < t$, so $\|D(z) - D(z_0)\| < Lt = 2\delta$ and, since F_a is closed and $D(z_0) \in F_a$, $\operatorname{dist}(D(z), F_a) < 2\delta$. By the 2δ-separation this excludes $D(z) \in F_c$ for every $c \neq a$, and $D(z) \notin F_a$ because $z \notin A_a$. Hence $D(z) \notin F$, and

$$\Pr[x \notin F] \ \geq\ \gamma\big(A_a^{<t} \setminus A_a\big) \ =\ \gamma(A_a^{<t}) - s_a \ \geq\ g(s_a), \qquad g(s) \coloneqq \Phi\big(\Phi^{-1}(s) + t\big) - s.$$

To obtain a bound uniform over the component mass, let φ be the standard normal density. Differentiating gives $g'(s) = \varphi(\Phi^{-1}(s) + t)/\varphi(\Phi^{-1}(s)) - 1 = e^{-t\Phi^{-1}(s) - t^2/2} - 1$. This derivative is positive exactly when $\Phi^{-1}(s) < -t/2$. Thus, g increases and then decreases, with its maximum at $s = \Phi(-t/2)$. Its minimum on $[p, \frac{1}{2}]$ must therefore occur at an endpoint, giving

$$g(s_a) \ \geq\ \min\big\{g(p),\, g(\tfrac{1}{2})\big\} \ =\ \min\big\{\Phi(\Phi^{-1}(p) + t) - p,\ \Phi(t) - \tfrac{1}{2}\big\} \ =\ c(p,t) \ >\ 0,$$

which proves the bound. For a probability-flow ODE with a deterministic globally Lipschitz final map, the same argument applies when its output assigns the stated mass to separated components. Explicit component selection represents alternatives through conditional maps. TOSCA's discrete reflection branches provide this type of selection for the symmetrical-to relation. □

The lemma concerns one deterministic Lipschitz map assigning positive mass to separated components. A proximity ball is connected, whereas proximity combined with further constraints can produce the required separation. Explicit discrete branch selection represents these alternatives through a family of conditional maps, each associated with a selected branch.

Disconnected feasible regions concern the geometry of valid outputs. A different restriction can come from indistinguishable input nodes. Here, graph symmetry means a node swap that preserves the graph and its conditioning, rather than the geometric symmetrical-to relation. The next lemma isolates how this node symmetry interacts with initialization.

Lemma 6 (Invariant coincidence under a swap symmetry)**.** *Suppose exchanging nodes u, v leaves the graph and all conditioning information unchanged. An equivariant encoder gives them identical deterministic features. If a swap-equivariant, locally Lipschitz ordinary differential equation (ODE) starts from $x_u = x_v$, this equality persists. Independent initial noise can instead separate the objects. If a positive-weight closed constraint excludes a neighborhood of zero relative displacement and the full graph is feasible, every certified feasible layout keeps the objects outside that neighborhood.*

Proof of Lemma 6. The swap-equivariant encoder gives the two interchangeable nodes equal deterministic features. For drift b and swap π, $b(\pi x) = \pi b(x)$. When $x_u = x_v$, we have $\pi x = x$ and therefore $b_u(x) = b_v(x)$. Uniqueness of the ODE solution preserves this equality. Near equality, a Lipschitz drift can separate or bring together the objects at a rate $O(\|x_u - x_v\|)$. A closed set excluding zero has positive distance from zero. By continuity of the edge feature, every feasible layout must then exclude a neighborhood in which the objects coincide. Joint feasibility ensures that such constraint-satisfying layouts exist. □

Under equal initialization and swap-equivariant dynamics, equal conditioning preserves equal states. Independent initial noise changes this premise, while a jointly feasible separation constraint rules out coincidence in a constraint-satisfying layout. For an infeasible graph, the objective in Theorem 3 measures the compromise among the selected constraints.

C.2 Feasibility Certificates and Relation Completion

Proof of Theorem 2. If $s = Ax + c \in S$ and $A^\top y = 0$, then $\langle y, s\rangle = \langle y, c\rangle \leq \sigma_S(y)$. A strict inequality in the opposite direction therefore proves infeasibility. Farkas' lemma guarantees such a certificate for polyhedral systems. For nonpolyhedral conic sets, completeness requires a strict separator because closed individual sets alone do not exclude weak infeasibility. When $A = B$ is the incidence matrix, $B^\top y = 0$ describes a circulation that can be decomposed into graph cycles. General affine reflection constraints and node bounds need not have this cycle interpretation. For difference constraints, each inequality has the form $x_v - x_u \leq b_e$. Summing around a negative cycle yields a contradiction. If there is no such cycle, shortest-path values give a feasible solution in $O(nm)$. With alternative branches, infeasibility requires ruling out every allowed assignment. □

The same feasibility question arises when completing a graph with missing relations. Each candidate completion changes the joint feasible set, so it can be checked before accepting the added relations.

Corollary 7 (Certificate-aware relation completion)**.** *Suppose missing relations and branches are chosen from a finite catalog. For each choice, a fixed affine convex feasibility test checks whether the completed graph admits a layout. A feasible layout satisfies both observed and added constraints. If the observed constraints are certified inconsistent, every completion retaining them is infeasible. Exhaustive search may take exponential time, and multiple feasible completions can share the same observed graph.*

Proof of Corollary 7. For each candidate relation and branch assignment, a feasible point verifies all augmented constraints. A certificate that the observed subgraph is inconsistent rules out every

extension retaining it. The number of assignments is the product of the per-edge catalog sizes and can grow exponentially with the number of missing relations. Each accepted assignment is a feasible completion, and distinct assignments can satisfy the same observed constraints. □

This separates continuous feasibility from discrete search. Difference constraints encoding strict order have complete polynomial-time tests, and fixed-branch polyhedral systems reduce to linear feasibility. Choosing among relation alternatives introduces a separate combinatorial problem, with general directional relation systems admitting NP-complete cases (Skiadopoulos & Koubarakis, 2005; Renz & Nebel, 1999). Complete infeasibility certification for nonpolyhedral conic systems additionally requires strong separation (Theorem 2).

C.3 Objective Bounds, Convergence, and Fixed Points

Proof of Theorem 3. Using the weighted norm defined in Sec. 3.3, write $V(x) = \min_{s\in S} \|Ax + c - s\|_W^2$. The conjugate identity

$$\|z\|_W^2 = \sup_y \{\langle y, z\rangle - \tfrac{1}{4} y^\top W^{-1} y\}$$

implies, for $A^\top y = 0$,

$$V(x) \geq \langle y, c\rangle - \sigma_S(y) - \tfrac{1}{4} y^\top W^{-1} y = D(y).$$

Thus $D(y) \leq V^* \leq V(x)$, so the primal-dual gap bounds the error in the returned objective value. The quadratic term turns the separating direction into a numerical lower bound. If a minimizer exists, $V^* = 0$ holds exactly when the constraints are feasible, because all weights are positive and all sets are closed.

The objective bound evaluates a returned layout. To establish how residual-gradient refinement reaches a minimizer, observe that $E = V/2$ is a sum of convex differentiable squared-distance functions with

$$\nabla E(x) = \sum_e q_e A_e^\top M_e \big(A_e x + c_e - \Pi_{S_e}^{M_e}(A_e x + c_e)\big).$$

The gradient is Lipschitz, with a valid bound $L_E \leq \|A\|^2 \|W\|$. For $0 < \tau < 2/L_E$, the standard convergence theorem for gradient descent on a convex smooth function gives convergence in finite dimensions when a minimizer exists (Bauschke & Combettes, 2017). Each iteration uses projections to compute residuals and then follows the gradient of their summed squared-distance energy. □

Assumption 1 (An error bound for a linear objective rate)**.** Let $E^* = \min_x E(x)$. The Polyak-Łojasiewicz error bound assumes, on the iterates of the fixed convex problem, that $\frac{1}{2}\|\nabla E(x)\|^2 \geq \nu(E(x) - E^*)$ for some $\nu > 0$.

For $0 < \tau \leq 1/L_E$, the descent lemma gives $E(x^{k+1}) - E^* \leq (1 - \tau\nu)(E(x^k) - E^*)$. The rate is conditional on the stated error bound for the fixed objective E. For a different fixed branch assignment or transport-set choice, the same argument applies to its corresponding objective and error-bound constant.

Convergence to a feasible layout is determined by the fixed points of the residual-gradient update. The next result characterizes these fixed points and gives conditions that exclude a constant layout in which every object has the same state.

Theorem 8 (Fixed points and constant feasible layouts)**.** *Consider a fixed affine convex subproblem with an attained energy minimum and the step sizes in Theorem 3. Residual-gradient updates stop exactly at minimizers of E and converge to such a minimizer. If the graph is feasible and at least one positive-weight difference constraint excludes zero, a constant layout cannot minimize the energy. For an infeasible graph, opposing residuals can balance at a constant layout. Initializing the updates from a learned sample preserves convergence for the same fixed objective.*

Proof of Theorem 8. For differentiable convex E, $x = (I - \tau\nabla E)x$ holds exactly when $\nabla E(x) = 0$, or equivalently $x \in \arg\min E$. Theorem 3 gives convergence. For a feasible graph, $E^* = 0$, but assigning all objects the same state violates any positive-weight difference constraint that excludes

zero. For an infeasible graph, gradients may cancel at a positive-energy minimizer. The dimension of the feasible set is that of its affine hull. Adding the free dimensions of individual edges can overcount when constraints are dependent. When only independent affine equalities restrict a neighborhood inside the feasible set, its local dimension is the remaining ambient dimension minus the rank of those equalities. With active inequalities, feasible directions additionally satisfy the tangent-cone restrictions. Learned initialization preserves convergence for the fixed objective, and projection residuals measure the remaining violation at each finite iterate. □

Feasible sets may retain free coordinates after global pose is fixed, allowing distinct converged layouts. Their dimension depends on the combined constraints, rather than on the sum of the freedoms allowed by individual edges. At a finite stopping iterate, projection residuals quantify the remaining violation for the selected sets and branches.

For a chosen branch assignment and fixed size bounds, the residual and primal-dual gap quantify feasibility and objective accuracy at a finite iterate.

C.4 Sensitivity to Relation Edits

Proof of Theorem 4. Let $\delta = \nabla F'(\hat{x}) - \nabla F(\hat{x})$, supported on the endpoints of the edited edge, and let $\overline{H} = \int_0^1 \nabla^2 F'(\hat{x} + t(\hat{x}' - \hat{x}))\, dt$. Absolute continuity of the gradient along the segment yields $0 = \nabla F'(\hat{x}') = \delta + \overline{H}(\hat{x}' - \hat{x})$. The averaged Hessian preserves graph sparsity and has eigenvalues in $[m, M]$, giving $\hat{x}' - \hat{x} = -\overline{H}^{-1}\delta$. Approximating $1/\lambda$ by polynomials on $[m, M]$ approximates this inverse with error decreasing geometrically at rate $\beta = (\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1)$. A degree-$k$ polynomial of a graph-sparse matrix has zero blocks between nodes more than k hops apart. Since δ is nonzero only at the edited edge's endpoints, applying this approximation yields the exponential bound, with the constant C covering the endpoints. If $m = M$, the Hessian is a scalar multiple of the identity, so only the edited coordinates respond.

The anchored-optimum bound concerns movement after optimization. The separate network propagation bound concerns which nodes can be affected during a finite computation and follows by induction over local layers and integration steps. Each step can pass information only through its local operations, so every auxiliary operation must also be local. The resulting LT-hop bound identifies which objects can respond, without bounding their movement within that region. Global consensus or extra optimization can extend the range. □

C.5 Coordinate Equivariance

The preceding results keep the reference frame fixed. Coordinate equivariance instead asks how the update and its output change when the features, constraints, and reference frame transform together, following equivariant-flow methods (Köhler et al., 2020; Yim et al., 2023).

Proposition 9 (Equivariance under a compatible frame action)**.** *Assume edge features, sets, metrics, and auxiliary operations transform consistently under a group action. Express updates in left-trivialized coordinates, which describe motion relative to the current object state. If the learned update uses invariant inputs in these coordinates, both the field and its numerical sampler are equivariant. Thus, transforming the input before generation gives the same result as transforming the generated output. Transforming the initial distribution accordingly also transforms the output distribution. World-axis and origin-based rules require their reference frames to transform as well.*

Proof of Proposition 9. For intrinsic coordinates, $(ag_i)^{-1}(ag_j) = g_i^{-1}g_j$. Thus, when v_i is invariant, the update $g_i \exp(v_i)$ transforms to $ag_i \exp(v_i)$. In other coordinate systems, the frame, edge map, and metric transform together under the compatible action. Applying this property at each sampling step gives an equivariant output map and the corresponding transformation of the output distribution. For priors expressed in world coordinates, compatibility includes transforming the prior with the frame. We handle global translation by centering or by identifying layouts that differ only by translation because a finite distribution over absolute positions cannot be invariant to every translation. □

D Model, Training, and Sampling Details

D.1 State Coordinates and Projection Operators

Each object has 7 coordinates in $\mathsf{G} = (\mathrm{SO}(2) \ltimes \mathbb{R}^2) \times \mathbb{R}^4$. We interpolate states with the Lie exponential in a local coordinate chart. This path need not be a geodesic for every metric. Initial samples are centered to fix global translation, since no finite distribution of absolute positions is invariant to every translation. The logarithm and exponential use closed-form planar $\mathrm{SE}(2)$ formulas and act as the identity on the remaining coordinates. Taylor expansions stabilize the formulas near zero rotation.

Once the object states determine the relation features, computing a projection residual requires a projection onto the corresponding transport set. Projection formulas depend on both the allowed set and its distance metric. For a half-space $a^\top \eta \le b$ with nonzero normal a, scalar bound b, and SPD metric M, the projection is $\eta - M^{-1}a[a^\top \eta - b]_+/(a^\top M^{-1}a)$. Here $[u]_+ = \max(u, 0)$ denotes the positive part. A diagonal metric allows coordinate-wise clamping for an axis-aligned box. An isotropic metric allows radial projection for a Euclidean ball, and a compatible coordinate transformation gives the formula for a metric-aligned ellipsoid. These shortcuts need not work with an arbitrary SPD metric, and intersecting sets may require iterative optimization. The convex results assume that each projection is computed for the specified metric. The layout configuration used $M_e = I$ in the catalog feature coordinates, with positions and extents in meters and size comparisons in log-extents. The positive learned preconditioner P_θ changes network updates and is distinct from this fixed projection metric.

D.2 Network Architecture and Orientation Prediction

The reference network used $L = 4$ TCA layers of width $d_f = 128$, where d_f is the node-feature dimension. Each layer uses residual-conditioned graph attention over scene-graph neighbors as defined in Equation 4, and V_θ maps the node features and aggregated messages to the learned correction in Equation 5. Table 18 reports the depth and width controls. Two-layer multilayer perceptrons (MLPs) implement the learned maps in Equations 4-6. Softplus keeps the diagonal preconditioner positive. The layout network has $.84$ M parameters.

The object state includes yaw alongside position, size, and elevation, but both the fitted spatial relation features and the original relation scoring rules omit yaw. Changing yaw alone can therefore alter rendered-image fidelity while leaving those relation scores and violation energy unchanged. We provide the network with relative-angle features $(\cos \Delta\vartheta, \sin \Delta\vartheta)$, where $\Delta\vartheta$ is the yaw difference between two objects. A head that predicts one of four quadrants avoids averaging distinct axis-aligned directions into an intermediate angle.

The most informative orientation frame depends on object category (Table 5). We compared yaw ϑ in world coordinates with scene-relative yaw $\vartheta - \psi$, where ψ is the direction from the scene center to the object. Let p_k be the empirical fraction of orientations in yaw quadrant k. The Shannon entropy is $H = -\sum_{k=0}^{3} p_k \log_2 p_k$, with zero-probability terms taken as zero (Shannon, 1948). It ranges from zero bits for a single occupied quadrant to two bits for a uniform distribution. *Mode share* is $\max_k p_k$, the fraction in the most common quadrant. These statistics describe the annotated orientation distribution. Wardrobes and TV stands are more concentrated in the scene-relative frame, whereas ceiling lamps favor world axes. Table 5 shows selected categories and aggregates over all categories. Pooling categories obscures their differences, so selecting a frame per category gives a more informative reference than one shared frame.

These differences motivate selecting an orientation frame for each category. For categories modeled in a scene-relative frame, jointly rotating the scene and its reference direction preserves the relative yaw. Categories modeled with world-axis priors retain their alignment to the dataset axes. The geometric transport construction is equivariant under compatible coordinate transformations (Proposition 9).

The frame choice used the training split alone, whereas Table 5 describes the combined training and validation distribution. For each category, we chose the world or scene-relative frame with lower training-set quadrant entropy, using the world frame for ties. Let $\varphi_i = 0$ in the world frame and let φ_i be the direction from the scene center to object i in the scene-relative frame, with $\varphi_i = 0$ for an

Table 5: **Object orientation in world and scene coordinates** (combined training and validation bedroom split, 19,372 objects). Lower quadrant entropy H and higher Mode share indicate more concentrated orientations. The individual rows show selected categories, while the pooled and per-category aggregates include all categories. Bold marks the more concentrated frame.

		world frame ϑ		scene frame $\vartheta - \psi$	
Category	n	$H \downarrow$	Mode share↑	$H \downarrow$	Mode share↑
wardrobe	3115	1.96	0.35	**1.23**	**0.57**
tv_stand	579	1.91	0.30	**1.06**	**0.59**
nightstand	5278	1.92	0.31	**1.82**	**0.38**
chair	442	2.00	0.27	**1.96**	**0.35**
double_bed	3320	**1.93**	**0.32**	1.99	0.28
pendant_lamp	2459	**1.01**	**0.81**	1.99	0.29
ceiling_lamp	1222	**0.66**	**0.89**	2.00	0.27
all objects, pooled	19372	**1.913**	**0.389**	1.999	0.264
per category, best frame	19372	**1.56** bits, **0.42** at the mode			

object at the center. The target $k_i \in \{0, 1, 2, 3\}$ is the nearest quadrant center to the annotated yaw $\vartheta_i - \varphi_i$, with angular ties assigned to the smaller index. If $\pi_{\theta,i}$ is the head's four-class probability vector, its loss and decoded yaw are

$$\mathcal{L}_{\text{yaw}} = \mathbb{E}\left[-\frac{1}{n}\sum_i \log \pi_{\theta,i}(k_i)\right], \qquad \widehat{\vartheta}_i = \varphi_i + \frac{\pi}{2}\arg\max_k \pi_{\theta,i}(k) \pmod{2\pi}. \tag{14}$$

The expectation uses the same scenes and interpolation times as flow matching. Sampling decodes the head after flow integration, using the generated object positions to determine scene-relative reference directions.

D.3 Training Objective and Optimization

Flow matching fits the layout distribution, while transport-set fitting determines the geometric constraints used to guide it. The fitting objective encourages annotated relation features to lie inside their transport sets. For an inequality set, define the signed margin $\mathrm{mgn}_r(\eta)$ as the distance to the boundary inside the set and the negative distance to the set outside, using the same feature metric as its projection. Let $\mathcal{E}_{\text{data}}$ be annotated training edges, η_e^{data} their features, and $m_{0,e}$ the target interior margin. Nonnegative weights λ_{sec} and λ_{reg} balance margin fitting and the finite parameter penalty $\Omega(S_r)$ defined in Equation 11. The orientation head uses the quadrant classification loss $\mathcal{L}_{\text{yaw}}$ in Equation 14. For inequality edges, the combined objective is

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda_{\text{yaw}}\mathcal{L}_{\text{yaw}} + \lambda_{\text{sec}} \sum_{e \in \mathcal{E}_{\text{data}}} [m_{0,e} - \mathrm{mgn}_{\rho(e)}(\eta_e^{\text{data}})]_+^2 + \lambda_{\text{reg}} \sum_r \Omega(S_r). \tag{15}$$

Equality edges replace the squared hinge term with $d^2_{M_e}(\eta_e^{\text{data}}, S_e)$ because an equality set has no positive ambient interior margin. For inequalities, the loss encourages features to lie at least $m_{0,e}$ inside the set. The regularization sum ranges over the unique trainable relation/category-pair blocks, counting shared inverse parameters once. Each term penalizes normalized deviations of finite boundary parameters from their training-quantile references, balancing interior coverage against changes in the relation geometry. Flow matching and quadrant classification update the network. Their gradients with respect to catalog parameters are stopped, while gradients through relation features to the network are retained. The margin and regularization terms update catalog parameters, which are frozen for sampling and evaluation.

For each symmetrical-to edge, we compute the energy of each reflection branch from the residual between one object's planar position and the other's reflected position (Appendix B). During training, the negative branch energies served as logits for a Gumbel-softmax (Jang et al., 2017) relaxation, whose temperature decreased linearly from 1 to .1 over training. Sampling uses the discrete branch rule in Appendix D.4.

We used AdamW (Loshchilov & Hutter, 2019) with a learning rate of 10^{-4}, gradient clipping at 1, and gradient accumulation over 8 scenes of variable size. We set the loss weights to $\lambda_{\text{sec}} = \lambda_{\text{yaw}} = 1$ and $\lambda_{\text{reg}} = 10^{-2}$, with common target margin $m_{0,e} = m_0 = .05$ in each feature's metric units. The runs in Tables 1 and 2 used 160k training steps, with seeds and sampling draws specified in Appendix E.4. The orientation and size controls in Table 20 used the same training duration. Training used two numerical safeguards. Log-extents were clamped to $[-5, 3]$ before exponentiation when constraint bounds depend on object size, keeping the extents and resulting energy finite. Updates with non-finite losses or gradient norms were skipped.

D.4 Sampling, Refinement, and Constraint Checks

We sampled layouts with $T = 50$ explicit integration steps, then applied $K = 500$ residual-gradient steps for final refinement. We assigned unit weight to relations specified by the input graph. Zero-weight edges contribute no penalty. We set the nominal sampling margin to $m_s = .25$ in each feature's units and capped it by the available set width in Equation 16.

Sampling selects the lower-energy reflection branch for each symmetrical-to edge, with ties assigned to branch 1, and holds this choice fixed within each refinement step. Each branch defines an affine reflection constraint for the fixed convex subproblem. The convex guarantees apply after the relation sets and branches are fixed.

The sampling margin encourages inequality features to lie farther inside their allowed sets, reducing sensitivity to boundary-adjacent predictions (Table 19). For a full-dimensional convex inequality set, let ∂S_e denote its boundary and let $\iota_e = \sup_{\eta \in S_e} d_{M_e}(\eta, \partial S_e)$ be its metric inradius. The effective margin is $m_{s,e} = \min\{m_s, \iota_e/2\}$, with $\iota_e = \infty$ for a half-space. The resulting sampling set is

$$S_e^{\text{samp}} = \{\eta \in S_e \mid d_{M_e}(\eta, \partial S_e) \geq m_{s,e}\}. \tag{16}$$

Projection onto the smaller sampling set moves an outside point to its boundary and leaves an inside point unchanged. For a Euclidean box, ι_e is its smallest half-width, and for the standing-on slab it equals τ. The cap preserves a nonempty sampling set even when its width is smaller than the nominal margin. Equality subspaces and zero-width slabs use zero margin and their original residuals. Individually nonempty sampling sets can still impose incompatible requests when combined in a graph. Every point in S_e^{samp} also belongs to S_e, so satisfying the sampling sets satisfies the original constraints. The stronger sampling margins can introduce additional conflicts, making the choice of sets part of the objective. We therefore evaluated final residuals and verified infeasibility certificates on the original transport sets with their associated weights, bounds, and branches. The objective bounds in Theorem 3 apply to that fixed problem.

Edge projections act on separate relation features, but all edges must refer to one shared layout. The evaluated sampler enforces this coupling through residual-gradient steps, which sum the incident edges' corrections at each object. For a fixed affine convex subproblem, an equivalent consensus formulation gives each edge a local feature variable and constrains it to agree with the shared layout. Its objective retains the squared-distance penalties, allowing it to minimize violation even when the relation graph is infeasible. The corresponding dual formulation in Equation 8 supplies lower bounds on this fixed objective.

For difference constraints, a verified negative cycle certifies incompatibility. The primal-dual gap bounds objective error for the fixed convex problem (Equation 8). Refinement reduces the violation by passing each feature-space correction through the relation-feature map. With fixed affine features, $A_e^\top$ maps each metric-weighted projection residual back to layout coordinates. Let $\deg(i)$ count the edges incident to node i. For unweighted Euclidean difference constraints, $L_E \leq 2 \max_i \deg(i)$, so the step size $1/(4 \max_i \deg(i))$ is conservative. The corresponding smoothness bound depends on the metrics, feature selectors, and weights.

For the complete catalog, refinement starts each step at $\alpha_0 = 1/(4 \max\{1, \max_i \deg(i)\})$ and halves it until the sampling-set energy satisfies the Armijo decrease condition with coefficient 10^{-4}. Relation branches remain fixed during this line search, and size-dependent bounds are recomputed at each trial layout. We allowed at most 20 trial steps and retained the current layout if none was accepted or the gradient was zero. This finite safeguard also handles nondifferentiable changes in the active size-dependent bound. The rule accommodates the log-volume selector and changing proximity bounds in addition to simple coordinate differences.

E Evaluation Setup and Metrics

This appendix specifies the shared data preparation, baseline configurations, metrics, and averaging conventions. Controls specific to individual experiments are described with their results, and published reference values retain their source protocols.

E.1 Datasets and Data Preparation

The main comparison used SG-FRONT bedroom, living, and dining scenes with the per-room splits used by CommonScenes and EchoScene (Zhai et al., 2023; 2024). We retrained the graph-conditioned layout models on the same splits. Three released annotation graphs contain certified contradictions, so we excluded these scenes from training for every method and evaluated their contradictions separately (Appendix H.3). Appendix F specifies the separate Inf3DLayout replication, and Appendix G specifies the external InstructScene benchmark.

The loaders read the relation type from `triple[2]`. This is a 1-based index into `relationships.txt`, with 0 reserved for an untyped relation. The exporter writes raw instance IDs and obtains category names directly from the dataset annotations. This preserves the object categories when room-specific training vocabularies merge fine-grained classes or change category indices.

E.2 Baseline Configurations

EchoLayout (Zhai et al., 2024) is the layout-only variant introduced by the EchoScene authors, who report it separately from their full layout-and-shape model. It generates object boxes through the layout diffusion branch and uses retrieved meshes to populate them. For our full-split comparisons, we retrained this variant with the common mesh library and renderer used by the other methods, evaluating object arrangement under controlled appearance. We used the released training configuration with 2000 epochs per room and batch size 64. EchoLayout's collision loss uses intersection-over-union (IoU) kernels from `mmdet3d`. We replaced them with an axis-aligned `torch` implementation that approximates overlap but does not generally bound IoU for rotated objects. We tested EchoLayout with and without this regularizer and report the stronger variant per room.

The released full EchoScene checkpoint is a separate reference model trained across room types. Some of its training scenes occur in the per-room living and dining test splits, so we excluded these scenes from relation evaluation and this checkpoint from the full-split FID ranking (Appendix H.4). In the controlled-result tables and qualitative comparisons, EchoLayout denotes the retrained layout variant and EchoScene denotes this released full-model checkpoint. Tables that reproduce published results identify their source protocol separately. Evaluation used the matching vocabulary and normalization statistics. For per-room evaluation of the full-model checkpoint, we provided annotation-file aliases and a 14-value bounds vector computed from the union of the room training and validation splits. We generated layouts with the released sampling code and exported them to the shared renderer. Box formats, normalization statistics, and vocabularies come from the released `bbox.zip` and `classes_all.txt`, with per-room normalization bounds matching the released values.

Category-only MMGDreamer (Yang et al., 2025) received the supplied graph and encoded object category labels through the authors' text-conditioning path. We retrained its layout branch without ground-truth image features from contrastive language-image pretraining (CLIP) (Radford et al., 2021). Image features and the signed distance field (SDF) branch were disabled through `--with_image False` and `--with_SDF False`, including their masks and loss terms. The loader used the image vector-quantized variational autoencoder (VQ-VAE) (van den Oord et al., 2017) checkpoint referenced by the CommonScenes release where required for model initialization. Visualization initialization was disabled, and data loading used a single process.

The retrained InstructScene (Lin & Mu, 2024) layout decoder received the annotated test relation graph directly and appearance tokens sampled from its own prior, bypassing text-conditioned graph generation. Together with the category-only MMGDreamer configuration, this removes additional ground-truth appearance information from the graph-conditioned comparison. The separately evaluated released InstructScene pipeline sampled its own graph, as specified in Appendix G.

E.3 Rendering and Evaluation Metrics

The main SG-FRONT comparisons used the same mesh library, renderer, and test-image reference for every method. Generated object states were rendered with fixed meshes, retrieved meshes, or box proxies according to the named experiment. The Inf3DLayout box-proxy protocol and the external InstructScene rendering pipeline are specified with their respective benchmarks (Appendices F and G).

Relation satisfaction and rendered-image fidelity answer different questions. The former tests requested relations in generated geometry, while the latter compares rendered-image distributions. Tables 1 and 2 separate these measurements. The metric definitions apply throughout the paper. Diagnostics specific to conflicts, edits, orientation, and layout similarity are defined with their experiments.

The geometric predicates follow CommonScenes (Zhai et al., 2023) and EchoScene (Zhai et al., 2024). Micro and Macro apply the established principles of pooling observations and averaging label-specific rates to the predicates' binary outcomes (Koyejo et al., 2015). Joint adapts the exact-match criterion by requiring every scored request in a scene to hold (Dembczyński et al., 2010). Frequent labels can dominate Micro, so Macro gives each label equal weight. Joint records whether any scored request remains unsatisfied. The observations here are requested geometric relations, and each scene can contain a different number of scored edges.

Let $\mathcal{S}$ be the evaluated test scenes, let $\mathcal{E}_s^{\text{rule}}$ contain their scored relation edges, and let $b_{s,e} \in \{0, 1\}$ indicate whether the generated layout passes the benchmark predicate for edge e. For relation label r, let n_r count its scored edges and let a_r be their satisfaction fraction. Write $\mathcal{R}_+$ for labels with $n_r > 0$. The per-label rates and aggregate scores are defined as follows.

$$a_r = \frac{1}{n_r} \sum_{s \in \mathcal{S}} \sum_{\substack{e \in \mathcal{E}_s^{\text{rule}} \\ \rho(e) = r}} b_{s,e}, \qquad n_r = \sum_{s \in \mathcal{S}} \big| \{ e \in \mathcal{E}_s^{\text{rule}} \mid \rho(e) = r \} \big|, \tag{17}$$

$$\text{Micro} = \frac{\sum_{s \in \mathcal{S}} \sum_{e \in \mathcal{E}_s^{\text{rule}}} b_{s,e}}{\sum_{s \in \mathcal{S}} |\mathcal{E}_s^{\text{rule}}|}, \qquad \text{Macro} = \frac{1}{|\mathcal{R}_+|} \sum_{r \in \mathcal{R}_+} a_r, \tag{18}$$

$$\text{Joint} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \prod_{e \in \mathcal{E}_s^{\text{rule}}} b_{s,e}. \tag{19}$$

Micro gives each scored edge equal weight, Macro gives each represented label equal weight, and Joint gives each scene one binary outcome. All three lie between zero and one. We evaluated them per sampling run before computing the run and seed averages specified in Appendix E.4. Figure 1(d) displays Micro as a percentage, while the tables report fractions.

The relation columns follow the SG-FRONT reporting convention used by CommonScenes (Zhai et al., 2023) and EchoScene (Zhai et al., 2024). LR, FB, BS, and TS group inverse directional or comparative labels, namely left/right, front/behind, bigger-than/smaller-than, and taller-than/shorter-than. Close and Sym. report close-by and symmetrical-to individually. The same six groups also appear in FlowScene (Yang et al., 2026) and SceneLinker (Kim et al., 2026). These columns organize ten scored labels into familiar geometric categories.

Standing-on is the eleventh scored label and is included in Micro, Macro, and Joint. Macro averages the eleven individual label accuracies rather than the displayed groups. We recomputed this label-balanced aggregate for every evaluated generator, including released EchoScene. The released EchoScene script also reports a mean over seven relation groups, which uses different weights and is separate from the Macro reported here. Above and the three semantic labels in Table 4 are outside this evaluator, so these scores assess the specified geometric subset of the full fifteen-label catalog. The six displayed columns alone do not cover that subset completely.

EchoScene's SG-FRONT evaluation predicates check axis displacement for directions, relative volume or top elevation for size comparisons, and object bottom heights for standing-on.[2] The close-by and symmetrical-to predicates both use a 0.45 m distance threshold. Close-by tests minimum corner

[2]The EchoScene predicate and aggregation implementations are available in `helpers/metrics_3dfront.py` and `scripts/eval_3dfront.py`.

distance, whereas symmetrical-to accepts a planar sign flip in either coordinate or both coordinates. The evaluator also provides a strict directional mode that rejects intersection volume greater than 0.3 times the smaller box's volume. These benchmark predicates differ from the fitted catalog's height comparison, floor-height slab, surface-gap box, and two reflection branches (Appendix B). Rows labeled Annotated layouts apply the same predicates to the recorded object geometry. They measure agreement between the dataset annotations and the evaluation rules before generation. A score below one indicates that an annotated relation fails its geometric predicate, for example when an annotated close-by pair exceeds the scoring threshold. A generated layout can pass more predicates by moving objects across these boundaries. Its higher score then means stronger rule satisfaction, while rendered-image fidelity and other geometric checks assess different properties.

Fitted-set satisfaction, abbreviated *Set sat.* in Table 6, instead pools membership indicators for $\eta_e \in S_e$ over the checked generated-layout edges. Its allowed sets and checked edge set can differ from those of the benchmark predicates. Inequality membership uses a nonnegative signed margin, while equality membership uses a projection residual tolerance. We retained the original fitted sets for evaluation even when generation used a positive sampling margin. *Catalog coverage* applies the same membership question to annotated relation instances and is reported as a percentage in Table 14. These two names distinguish generated-layout satisfaction from retention of annotated configurations.

For the main SG-FRONT comparison, FID (Heusel et al., 2017) and KID (Bińkowski et al., 2018) compare Inception features of generated test-scene renderings with those of held-out test-layout renderings. Lower values indicate closer feature-distribution agreement under the specified renderer, camera, meshes, and feature pipeline. Seed and sampling variation follow the aggregation stated with each table.

Footprint overlap is the fraction of objects whose oriented planar footprints interpenetrate another object's footprint by more than 2 cm. We excluded floors and lamps and counted each eligible object once, even if it overlapped multiple objects. The check includes object pairs without an input relation and applies the same rule to generated and annotated layouts.

E.4 Sampling and Aggregation

For Tables 1 and 2, we evaluated each retrained generator using three training seeds and three sampling runs per seed, giving nine runs per method and room. FID, KID, and relation metrics were computed separately for each run before averaging over sampling runs and training seeds.

The per-relation comparison evaluated released EchoScene on 83 living scenes and 100 dining scenes without training overlap (Table 2). Its bedroom test split is already overlap-free. For TOSCA, the spread of mean Micro across seeds is below 0.001, and most relation columns vary by no more than ± 0.002 across sampling runs. Symmetrical-to rule satisfaction can vary by ± 0.05 because a discrete coordinate choice can flip the entire scene, making a single run less representative. The external benchmark and controlled experiments state their own samples and averaging with the corresponding results.

F REPLICATION ON INF3DLAYOUT

Evaluation on SG-FRONT alone leaves open whether the same method remains effective when layouts come from a different construction process. We therefore repeated training and held-out evaluation on Inf3DLayout, the procedural component of M3DLayout (Zhang et al., 2026), to test whether the fidelity and constraint-satisfaction advantages persist beyond professionally designed layouts.

The full collection combines professional designs from 3D-FRONT (Fu et al., 2021), scanned environments from Matterport3D (Chang et al., 2017), and rooms generated with Infinigen (Raistrick et al., 2024). The M3DLayout paper reports $21{,}367$ layouts in total, including $13{,}929$ procedural layouts and $5{,}754$ layouts in its filtered 3D-FRONT component. Inf3DLayout broadens the source collection with small-object arrangements obtained by generating houses and partitioning them into rooms. These collection-level statistics motivate evaluating a complementary source, while the room subsets actually used in our experiment are specified below.

For this replication, we selected bedroom, living-room, and dining-room layouts from Inf3DLayout and constructed separate training and test splits. Using the procedural component makes this an evaluation on a different layout source from SG-FRONT, which already derives from 3D-FRONT. All results reported for this replication, including Table 6, refer to this Inf3DLayout subset. The protocol assesses reproducibility across data sources after retraining each method. Distinct scene IDs prevent exact overlap but do not exclude similar layouts across splits.

F.1 DATASET CONVERSION AND EVALUATION SETUP

After converting scene IDs and object categories, the bedroom split contains 856 training scenes and 84 test scenes, the living split contains 808 training scenes and 85 test scenes, and the dining split contains $2{,}411$ training scenes and 256 test scenes. Each method was trained on the corresponding Inf3DLayout training split and evaluated on its held-out rooms. Conversion used 1-based instance IDs, matching SG-FRONT and the condition `scene_instance_id > 0` in EchoScene-based loaders. It preserved every object and its box geometry.

Inf3DLayout provides object boxes rather than the SG-FRONT relation annotations. We assigned relation labels using the SG-FRONT vocabulary and fitted transport sets. Standing-on uses the floor-height test, close-by uses the surface gap, and directional relations and size comparisons retain their catalog definitions (Appendix B). The resulting annotations contain no realizable symmetrical-to instances, so training and scoring used the remaining relation types.

We retrained EchoLayout (Zhai et al., 2024), the layout-only EchoScene variant, per room, retaining intersection-over-union (IoU) regularization for living and dining and omitting it for bedroom. Category-only MMGDreamer (Yang et al., 2025) and the InstructScene (Lin & Mu, 2024) decoder used the same splits, with InstructScene receiving the GT graph. All methods shared a box-proxy renderer and excluded floor slabs. FID compares generated box renderings with held-out annotated-layout renderings, while fitted-set satisfaction measures the fraction of scored generated-layout edges whose features lie in their allowed sets (Appendix E.3). These settings compare generated layouts under the same object representation within Inf3DLayout, while our SG-FRONT experiments used their mesh-rendering protocol.

F.2 GENERATION QUALITY AND CONSTRAINT SATISFACTION

TOSCA achieves the highest fitted-set satisfaction and lowest mean FID among the evaluated methods in each room (Table 6). The simultaneous gains show that stronger fitted-constraint satisfaction is compatible with closer agreement to the procedural layout distribution under the shared box-proxy representation. Since all methods were retrained on Inf3DLayout, the comparison supports reproducibility across layout sources after dataset-specific training. The footprint overlap rate also lies closest to the Data overlap rate in each room (Table 7), indicating agreement with the annotated layouts' aggregate projected-overlap frequency. We scored SG-FRONT relations with its original benchmark rules, so its absolute relation-satisfaction values describe a different evaluation.

Table 6: **Fidelity and fitted-set satisfaction on Inf3DLayout.** This is the procedural subset of M3DLayout (Zhang et al., 2026), generated with Infinigen (Raistrick et al., 2024) and introduced in Appendix F. All methods used the same box-proxy renderer with floor slabs excluded. MMG-Dreamer used category-only inputs, and InstructScene used its layout decoder. Lower FID indicates closer generated and reference image-feature distributions. Higher fitted-set satisfaction (Set sat.) indicates a larger fraction of scored edges satisfying the fitted geometric constraints (Appendix E.3). TOSCA FID is a three-seed mean $\pm$ s.d. across seed means, with three draws per seed and six for bedroom. Baseline FID is a six-draw mean $\pm$ s.d. across draws. Hyphens denote unreported MMGDreamer dining results. Bold marks the best generator per column.

	Bedroom		Living		Dining	
Method	FID↓	Set sat.↑	FID↓	Set sat.↑	FID↓	Set sat.↑
EchoLayout (Zhai et al., 2024)	47.8 ± 1.7	.881	40.8 ± 1.1	.919	37.4 ± 0.4	.922
MMGDreamer (Yang et al., 2025)	44.4 ± 0.9	.906	38.3 ± 1.1	.939	-	-
InstructScene (Lin & Mu, 2024)	40.8 ± 0.6	.895	34.9 ± 0.9	.910	24.7 ± 0.4	.952
TOSCA (ours)	$\mathbf{37.1 \pm 0.4}$	**.991**	$\mathbf{31.5 \pm 0.5}$	**.994**	$\mathbf{22.9 \pm 0.3}$	**.995**

Table 7: **Footprint overlap relative to annotated data on Inf3DLayout.** The rates use the common footprint-overlap criterion defined in Appendix E.3. We measured the Data overlap rate on annotated layouts. Absolute deviations $|\Delta\mathrm{Data}|$ compare each generated rate with this empirical data rate, and smaller values indicate closer aggregate agreement. The data rate includes projected overlaps between objects at different heights. Bold marks the smallest deviation per room. A hyphen denotes an unreported result.

	bedroom	living	dining
Data overlap rate	0.681	0.606	0.644
EchoLayout (Zhai et al., 2024), $\lvert\Delta\mathrm{Data}\rvert$	0.23	0.11	0.11
MMGDreamer (Yang et al., 2025), $\lvert\Delta\mathrm{Data}\rvert$	0.46	0.42	-
InstructScene (Lin & Mu, 2024), $\lvert\Delta\mathrm{Data}\rvert$	0.51	0.41	0.25
TOSCA (ours), $\lvert\Delta\mathrm{Data}\rvert$	**0.06**	**0.01**	**0.01**

G EVALUATION ON THE INSTRUCTSCENE LAYOUT BENCHMARK

This evaluation tests whether the fidelity improvement persists under a different relation vocabulary and rendering pipeline. The InstructScene benchmark shares the room splits with the main comparison but changes relation labels, rendering, and FID references. We evaluated complete pipelines because the released InstructScene model sampled its own graph while TOSCA received the test graph. The main SG-FRONT comparison instead used retrained layout models with a common graph input.

G.1 BENCHMARK SETUP AND ADAPTATION

The converted bedroom split contains 3722 training scenes, 157 validation scenes, and 162 test scenes. The living split contains 543 training scenes, 78 validation scenes, and 192 test scenes. The dining split contains 631 training scenes, 92 validation scenes, and 177 test scenes. Bedroom scenes have medians of 5 objects and 9 typed edges, living scenes have 12 objects and 49 typed edges, and dining scenes have 11 objects and 46 typed edges.

The benchmark's ten relation types include planar directions, above and below, and directions qualified by closeness. We removed untyped edges labeled None and mapped the remaining relations to the transport-set catalog. Relation conversion followed geometry because SG-FRONT and InstructScene name horizontal directions differently. SG-FRONT left/right uses world z and front/behind uses x, while InstructScene left/right uses x and front/behind uses z. We therefore mapped InstructScene's left, right, front, and behind labels to SG-FRONT's behind, front, right, and left, respectively. InstructScene stores only relation rows with subject index s smaller than object index o, so we added reverse relations to preserve both edge descriptions. Omitting them would make below more frequent than above. We also converted its half-extents and center translations to the full extents and floor-level y coordinates used by SG-FRONT.

InstructScene uses a $1\,\mathrm{m}$ center-distance test for *closely*, while SG-FRONT uses the minimum-corner-distance *close-by* predicate defined in Appendix E.3. Generation used the mapped transport sets, and geometric checks used the external benchmark's predicates after conversion. These proximity definitions can disagree even after the coordinate mapping is correct. FID remains the primary comparison for this external pipeline.

The evaluation environment used `diffusers` 0.24 and `transformers` 4.35 with their compatible dependencies. Blender supplied the quantitative renders, and texture export preserved the filenames referenced by each material. We computed Fréchet distances using float64 arithmetic with a positive diagonal covariance regularization ε to stabilize the matrix square root for small sample sets. We evaluated the released InstructScene code, checkpoints, data, and default guidance with native retrieval and category-and-size retrieval. TOSCA used category-and-size retrieval and one sampling draw.

G.2 RELATION CONVERSION CHECKS

Agreement between converted annotations and geometric predicates checks the subject-object convention and coordinate conversion (Table 8). Mapping directional geometry improves agreement over literal label matching. This increase shows that matching relation names alone can introduce apparent geometric errors during conversion. Converted proximity edges retain lower agreement because the two benchmarks use different proximity tests. Consistent coordinate conversion therefore does not make the proximity scores interchangeable.

Table 8: **Relation conversion checks for the external pipeline.** Higher agreement means that more annotated pairs pass the mapped geometric predicate. Directional geometry matching improves agreement over literal label matching, while the different proximity predicates retain a discrepancy. Ranges summarize the conversion checks described above.

Implementation check	Result
Literal direction names, agreement↑	.41-.55
Geometric direction mapping, agreement↑	.93-.96
Converted proximity edges, agreement↑	.75-.80

G.3 Generation Results

TOSCA has lower FID than the locally evaluated InstructScene (Lin & Mu, 2024) release in every room, including under matched category-and-size retrieval (Table 9). Native and category-and-size retrieval give comparable bedroom FID over the full test split, and the matched retrieval control shows that the retrieval rule alone does not account for the fidelity difference. These results compare complete pipelines because the released model sampled its own graph while TOSCA received the test graph.

Published FreeScene (Bai et al., 2025) values provide additional context, while the local controls supply measurements under the shared renderer. Published references and local measurements occupy separate blocks in Table 9. The local InstructScene reproduction lies within one published seed standard deviation for bedroom and dining and slightly outside it for living.

Table 9: **Generation on the InstructScene benchmark** (FID with InstructScene's reference images and evaluation pipeline). The upper block contains published reference values, including the graph-to-scene FreeScene variant, and the lower block contains local measurements under the InstructScene renderer (Appendix G.1). TOSCA has the lowest FID in every measured column, and bold is confined to that block. The two baseline rows compare appearance retrieval rules, while TOSCA used category-and-size retrieval. The released InstructScene pipeline sampled its own graph and TOSCA was conditioned on the test graph, so this is a pipeline comparison. The locally evaluated baseline is within one published seed s.d. for bedroom and dining, but slightly outside it for living.

Method	Bedroom FID↓	Living FID↓	Dining FID↓
InstructScene (Lin & Mu, 2024)	114.78	110.39	129.76
FreeScene (Bai et al., 2025)	98.31	107.62	122.57
InstructScene (Lin & Mu, 2024) native retrieval	113.9	111.3	129.1
InstructScene (Lin & Mu, 2024) category-and-size retrieval	114.8	110.8	132.0
TOSCA (ours)	**91.6**	**99.1**	**111.9**

H SG-FRONT Reference Results and Evaluation Diagnostics

H.1 Published SG-FRONT Results

Tables 10 and 11 collect published SG-FRONT results under their source-specific rendering, conditioning, scoring, and averaging protocols. The controlled comparisons under our shared evaluation setup are reported in Tables 1 and 2. Table 10 reports published rendered-image FID, while Table 11 places published relation scores alongside an aggregate of our room-specific results.

Table 10: **Published SG-FRONT fidelity results.** Values retain the cited papers' rendering and conditioning protocols, with older rows following the source compilation. Lower FID indicates closer rendered-image feature distributions under each source protocol. Tables 1 and 2 report the controlled local comparisons.

Method	Bedroom FID↓	Living FID↓	Dining FID↓
Graph-to-3D (Dhamo et al., 2021)	63.72	82.96	72.51
CommonScenes (Zhai et al., 2023)	57.68	80.99	65.71
EchoScene (Zhai et al., 2024)	48.85	75.95	62.85
MMGDreamer (Yang et al., 2025)	45.75	68.94	55.17
FlowScene (Yang et al., 2026)	38.87	67.83	54.38

Table 11 reports satisfaction for each relation group. Each entry is the fraction of requested relations passing that group's geometric predicates under the source protocol. Across the published methods, close-by and symmetrical-to have lower satisfaction than the directional groups, indicating persistent difficulty with proximity and symmetry.

Table 11: **Published per-relation satisfaction for context.** Each score measures satisfaction of the named relation group, with higher indicating that more requested relations hold. The relation abbreviations follow Table 2. The upper block contains the cited papers' generation-mode values. The lower block is the unweighted mean of the three room values in Table 2. Published and local evaluators may differ in predicates, conditioning, and aggregation, so each block retains its own protocol. The primary controlled comparison is against retrained methods in Table 2.

Method	LR↑	FB↑	BS↑	TS↑	**Close**↑	**Sym.**↑
Graph-to-3D (Dhamo et al., 2021)	0.98	0.99	0.97	0.95	0.74	0.57
CommonScenes (Zhai et al., 2023)	0.98	**1.00**	0.97	0.95	0.77	0.60
EchoScene (Zhai et al., 2024)	0.98	0.99	0.96	0.96	0.74	0.55
MMGDreamer (Yang et al., 2025)	0.98	0.99	0.97	0.96	0.76	0.62
FlowScene (Yang et al., 2026)	0.98	**1.00**	0.97	0.97	0.76	0.58
SceneLinker (Kim et al., 2026)	0.98	0.99	0.97	0.97	0.82	0.71
TOSCA (ours)	**.997**	**1.00**	**1.00**	**.991**	**.962**	**.934**

H.2 Annotation Agreement and Benchmark Reference Frames

Table 12 applies the SG-FRONT evaluation predicates from EchoScene (Appendix E.3) to the annotated layouts using their dataset relation labels. This comparison measures annotation-predicate agreement, rather than comparing each layout with itself. Agreement can be below one because a labeled pair need not satisfy the evaluator's geometric threshold or reference-frame convention. Reporting this agreement shows where a generated-layout score reflects the scoring rule's interpretation of the request. TOSCA's higher close-by scores indicate that more requested pairs satisfy the corner-distance rule, which a generator can achieve by moving objects across its scoring boundary. FID and KID assess rendered-image agreement, while footprint overlap measures projected overlap in the resulting scenes.

For symmetrical-to, the reference frame is part of the rule itself. The original predicate tests approximate planar agreement after an axis reflection or a half-turn about the fixed origin (Appendix E.3).

Table 12: **Annotated-layout agreement with benchmark predicates.** Each score applies the named geometric predicate to object pairs bearing that relation label in the dataset. Values below 1.0 identify annotation-predicate disagreement. The scores characterize this agreement in the annotated layouts and do not impose upper bounds on generated-layout satisfaction.

Room	left/right↑	front/behind↑	standing-on↑	**close-by**↑	symmetrical-to↑
Bedroom	0.994	1.000	0.991	**0.894**	1.000
Living	0.986	1.000	1.000	**0.755**	1.000
Dining	0.982	1.000	1.000	**0.794**	1.000

Translating annotated layouts can therefore sharply lower the fraction of symmetrical-to pairs passing this predicate while leaving relative object positions unchanged (Table 13). Reference-frame alignment keeps the layout aligned with the origin and axes used by this benchmark scoring rule (Sec. 3.4). The ablation removing this alignment reduces Macro much more than Micro (Table 18). Macro gives each scored relation type equal weight, whereas the edge-pooled Micro score gives a rare type such as symmetrical-to little total weight.

Table 13: **The *symmetrical-to* rule depends on the frame.** Translating annotated bedroom layouts preserves relative object positions but changes their relation to the fixed origin. The score is the fraction of 957 annotated pairs satisfying the benchmark's symmetrical-to predicate, and higher means more pairs pass that fixed-frame test. Bold highlights the complete loss of agreement after translation.

Translation of annotated layouts	0 m	0.1 m	0.5 m	1.0 m
symmetrical-to score↑	1.000	0.994	**0.000**	0.002

H.3 Fitted Catalog Coverage and Annotation Feasibility

The fitted transport sets retain most annotated spatial relation instances, whereas requiring an exact singleton value excludes every tested close-by instance (Table 14). This comparison measures *catalog coverage*, the fraction of annotated relation features inside their fitted allowed sets. Higher coverage means that more observed configurations are retained (Appendix E.3). The largest remaining discrepancies occur on standing-on and close-by. Their limited share of annotated edges keeps the overall score close to complete coverage despite these exclusions. Per-label coverage is therefore needed to assess how well each relation's observed configurations are retained. Because the surface-gap box and the original corner-distance rule differ, these discrepancies reflect representation as well as annotation. The singleton control illustrates why preserving a region of allowed values matters for representing varied annotated configurations (Theorem 1).

Catalog coverage tests individual annotated edges. Graph feasibility asks whether all requests can hold simultaneously, so high coverage does not rule out contradictions between edges.

Feasibility checking catches incompatible requests already present in the SG-FRONT relation annotations, before a generator attempts to satisfy them (Theorem 2, Table 15). Each detected case has a two-edge contradiction involving inverse relations on the same ordered pair, or the same directional relation after exchanging the subject and object. Summing their difference constraints gives a negative cycle, proving that no layout can satisfy them together. The cases involve ceiling lamps at nearly identical positions, where the axis-aligned annotation rule is numerically unstable.

We excluded these three scenes from training for every method and evaluated their contradictions separately. They provide naturally occurring positive cases for conflict detection and complement the synthetic cases in SG-FRONT-Conflict.

Table 14: **Coverage of the fitted transport catalog** on annotated SG-FRONT layouts across all rooms and splits (515,731 spatial edges). The fitted-catalog block reports per-label coverage followed by the edge-pooled overall value. Coverage counts membership in the transport sets and is not an exact-equivalence test against benchmark scoring rules. The separate singleton control reports close-by coverage when the allowed set contains one feature value (Theorem 1). A hyphen marks the unreported edge count for this control. We held the fitted catalog fixed while evaluating coverage across the SG-FRONT splits. Higher coverage retains more annotated instances. Bold highlights the aggregate coverage and the singleton contrast.

Relation	Edges	Coverage↑
Fitted transport catalog		
smaller-than	66,199	100.00%
bigger-than	64,096	100.00%
left	53,406	100.00%
right	53,001	100.00%
behind	49,487	100.00%
front	49,381	100.00%
above	47,542	100.00%
shorter-than	40,422	100.00%
taller-than	39,212	100.00%
standing-on	32,743	98.43%
close-by	17,614	92.77%
symmetrical-to	2,628	100.00%
Overall	**515,731**	**99.65%**
Singleton control		
close-by	-	**0.00%**

Table 15: **Certified contradictions in the released SG-FRONT annotations.** Testing all 5,754 graphs with Theorem 2 identifies three infeasible cases. Each contains two contradictory edges involving inverse relations on the same ordered pair or the same directional relation after exchanging the subject and object. Their difference constraints form a negative cycle, proving infeasibility. All three involve nearly coincident ceiling lamps, where the axis-aligned annotation rule is numerically unstable.

Scene	Room	Axis	Conflicting edges
MasterBedroom-1409	bedroom	t_x	$\text{lamp}_0 \xrightarrow{\text{right}} \text{lamp}_1$ **and** $\text{lamp}_1 \xrightarrow{\text{right}} \text{lamp}_0$
LivingRoom-3587	living	t_x	$\text{lamp}_5 \xrightarrow{\text{left}} \text{lamp}_6$ **and** $\text{lamp}_5 \xrightarrow{\text{right}} \text{lamp}_6$
DiningRoom-2131	dining	t_x	$\text{lamp}_6 \xrightarrow{\text{left}} \text{lamp}_7$ **and** $\text{lamp}_6 \xrightarrow{\text{right}} \text{lamp}_7$

H.4 Training Overlap in the Released EchoScene Checkpoint

Training overlap prevents using the released full EchoScene checkpoint for a clean full-split comparison in living and dining (Table 16). Its `relationships_all_trainval` split assigns LivingDiningRoom scans to training and validation, while per-room splits assign some of the same scans to testing. Bedroom has no scene-ID overlap and supports a full-split comparison. Table 16 separates full test splits from subsets without training overlap. Bedroom supports a direct full-split comparison, in which TOSCA improves FID over the release. Living and dining require overlap-free subsets. We computed their subset FIDs on different test samples from the full-split scores, so these values are interpreted under their respective subset settings.

We used retrained EchoLayout for the full-split ranking and evaluated the released EchoScene checkpoint as a separate control. We computed its relation scores in Table 2 on overlap-free subsets. Scene-ID overlap concerns whether a test scene occurred in training. The separate layout-similarity diagnostics in Table 17 compare spatial arrangements.

Table 16: **Released EchoScene checkpoint audit** (Zhai et al., 2024). The first two rows give overlapping and total test-scene counts by room. Lower FID is better under a matched protocol. Italic full-split fidelity entries involve overlap. We applied final refinement to the same released-model outputs for a paired comparison. Overlap-free subsets exclude overlapping scene IDs. The TOSCA row gives the full-split FID means from Table 1. The bedroom subset cell is n/a because its full test split is already overlap-free, so a separate reduced-subset comparison is unnecessary.

	bedroom	living	dining
Test scenes seen in training	0	109	77
Total test scenes	162	192	177
FID↓, full test split	39.4	*32.3*	*35.3*
+ our final refinement	39.6	*37.6*	*39.4*
FID↓, overlap-free subset	n/a	56.4	50.1
TOSCA FID↓, full test split	24.8	33.5	33.1

Applying final refinement to the same released EchoScene outputs changes bedroom FID little and worsens it on the overlap-affected living and dining splits (Table 16). This paired control complements the TOSCA refinement ablation in Table 18, which measures relation satisfaction under its own fixed configuration. The refinement objective targets relation residuals, so it does not directly optimize image-distribution agreement. The FID increases in this control illustrate the practical consequence of that difference.

H.5 Position Matching by Layout-Twin Status

We identified layout twins by comparing annotated test layouts with training or validation layouts having the same object categories and counts. After centering the layouts, we used greedy matching within categories and selected the smallest mean planar distance over four global rotations in 90° increments. A test scene is a *layout twin* if any candidate achieves a mean distance of at most the twin threshold ϵ_{twin}, and is *twin-free* otherwise. This spatial similarity is distinct from the scene-ID overlap audited in Table 16.

Table 17 measures recovery of the annotated test-scene positions separately for these two groups. After centering each scene, we compared generated objects with their ground-truth counterparts in graph-node order. The position match rate is the fraction whose planar distance is at most the object-level threshold ϵ_{pos}, pooled over objects across all evaluated scenes.[3] For Table 17, we set $\epsilon_{\text{twin}} = \epsilon_{\text{pos}} = 0.25\,\text{m}$. TOSCA has the lowest position match rate in both groups for every room, indicating less recovery of the particular test arrangements. In the full-split comparisons, this lower recovery accompanies its better FID, KID, and relation satisfaction (Tables 1 and 2). Similar arrangements can increase position matching without exact scene reproduction. A nonmatching position may be either a valid alternative or a placement error, so this rate does not rank layout quality.

[3]For twin matching, floor and scene nodes were removed, and layouts with fewer than two remaining objects were excluded. Position matching excluded floor nodes and scenes with mismatched object counts.

Table 17: **Position match rate by training-twin status.** Object-level position match rates, pooled across scenes and grouped by layout-twin status using the thresholds and matching procedures defined in Appendix H.5. Twin scenes account for 58% of bedroom, 17% of living, and 33% of dining test scenes. Twin-free scenes have no such match. Lower rates indicate less recovery of the particular annotated test arrangements, and bold marks the minimum in each column. This direction is interpreted alongside fidelity and relation satisfaction because position mismatch can also reflect placement error. The released EchoScene row is interpreted with its documented overlap (Appendix H.4). MMGDreamer and InstructScene were retrained using the category-only and layout-decoder adaptations, respectively, described in Sec. 4.1.

	bedroom		living		dining	
Method	twin	twin-free	twin	twin-free	twin	twin-free
EchoScene (Zhai et al., 2024)	.662	.481	.585	.617	.596	.618
EchoLayout (Zhai et al., 2024)	.704	.534	.249	.206	.368	.206
MMGDreamer (Yang et al., 2025)	.677	.439	.298	.209	.376	.212
InstructScene (Lin & Mu, 2024)	.820	.503	.835	.341	.850	.376
TOSCA (ours)	**.285**	**.222**	**.140**	**.104**	**.224**	**.121**

I Ablations and Additional Evaluations

I.1 Component and Sampling Ablations

Retaining allowed regions and applying final refinement both improve benchmark satisfaction in the shared bedroom ablation (Table 18). Restricting each transport set to one allowed feature value substantially lowers both Micro, the fraction of scored edges passing benchmark predicates, and Macro, the equally weighted mean of per-type satisfaction rates. Many individual proximity edges remain satisfied, but more requested relations fail across the full system. The much larger loss in Micro and Macro than in close-by shows that the singleton restriction affects satisfaction across the relation catalog. This supports retaining allowed regions when several types of relation must be realized together. Removing final refinement separately lowers both aggregate rates and especially the close-by rate, demonstrating the contribution of explicit geometric correction.

The architecture controls in Table 18 changed network width or depth. Their effects on benchmark satisfaction are smaller than those of removing allowed regions or final refinement, and increasing depth does not improve the reference score. Removing reference-frame alignment mainly reduces Macro because symmetrical-to is rare, so its failures receive little weight in edge-pooled Micro but equal type weight in Macro. Removing the sampling margin or shortening refinement also lowers both aggregate benchmark satisfaction rates. Replacing the learned yaw head with a fitted yaw rule achieves comparable relation scores (Appendix D.2). Because the scored predicates omit yaw, comparable relation scores can coexist with different rendered appearances. The sampling ablations reused the reference checkpoint, while the training ablations changed one geometric or architectural setting under a shared evaluation protocol.

Table 18: **Component ablations** on the SG-FRONT bedroom test split. *train* denotes retraining with the changed setting, while *sample* denotes rerunning the shared checkpoint. We evaluated all configurations under the same protocol using three draws each and the nominal sampling margin specified in Appendix D.4, unless changed by the ablation. Micro pools benchmark-rule satisfaction over scored edges, Macro weights relation types equally, and close-by reports its own satisfaction rate (Appendix E.3). Higher means that more requested relations hold under the corresponding aggregation. Bold marks the best Micro and Macro values. Close-by is not bolded because repeated draws can differ by up to 0.015, while the no-refinement result is substantially lower.

Ablation	type	Micro↑	Macro↑	close-by↑
TOSCA	*reference*	**0.996**	**0.990**	0.981
singleton transport sets	train	0.661	0.698	0.951
no orientation channel	train	0.992	0.985	0.964
depth $4 \to 2$	train	0.986	0.973	0.939
depth $4 \to 6$	train	0.989	0.984	0.967
width d_f $128 \to 64$	train	0.988	0.980	0.933
margin $m_0 = 0$	train	0.985	0.979	0.951
margin $m_0 = 0.15$	train	0.985	0.978	0.963
no reference-frame alignment	sample	0.992	0.940	0.985
no sampling margin ($m_s = 0$)	sample	0.970	0.968	0.980
fitted yaw rule instead of the learned head	sample	0.995	**0.990**	0.982
final refinement $500 \to 100$ steps	sample	0.985	0.981	0.967
no final refinement	sample	0.829	0.833	0.576

A separate paired control isolates the sampling margin on a shared bedroom checkpoint, using the same sampling seeds for both settings (Table 19). This checkpoint differs from the component-ablation reference in Table 18. A positive margin improves aggregate benchmark satisfaction under unchanged scoring predicates. The per-relation rates show more satisfied size comparisons and close-by requests, while standing-on declines slightly. The two controls therefore support interior placement under their respective fixed configurations.

Table 19: **Sampling margin (eq. 16) on a bedroom checkpoint, using two settings with the same seeds.** Sampling used fitted or shrunk sets, while all scoring predicates remained unchanged. Certificates and coverage refer to the original transport sets. The paired settings shared a checkpoint and sampling seeds, isolating the effect of the margin. The displayed relation columns are individual labels, while Micro includes all eleven scored labels.

Sampling sets	left↑	front↑	bigger↑	taller↑	standing-on↑	close-by↑	**Micro↑**
fitted ($m_s = 0$)	0.998	1.000	0.916	0.903	**1.000**	0.973	0.955
shrunk ($m_s = 0.25$)	**1.000**	1.000	**1.000**	**1.000**	0.991	**0.985**	**0.998**

I.2 Orientation and Size Controls

Table 20 evaluates sequential additions of orientation projection and category-specific size constraints on SG-FRONT bedrooms using final checkpoints from the training schedule specified in Appendix D.3. Orientation projection adjusts yaw to fitted orientation choices (Appendix D.2), and category-specific size sets constrain object extents by category. The remaining controls vary final refinement and integration length (Appendix D.4). These controls separate rendered appearance from satisfaction of the scored relations. Adding orientation projection lowers FID, and adding category-specific size sets reduces it further. The configuration with both additions also achieves higher Micro than the relation-only configuration. These results show that high relation satisfaction alone does not ensure close agreement with the rendered-image distribution.

Table 20: **Orientation and size contributions to fidelity** on SG-FRONT bedrooms under a shared evaluation pipeline. All configurations used final checkpoints with the training duration specified in Appendix D.3. The first three rows sequentially add orientation projection and category-specific size sets to the relation-only configuration. The bottom rows report controls for final refinement and integration length. FID measures rendered-image distribution agreement and Micro measures the fraction of scored edges satisfying benchmark rules (Appendix E.3). Yaw is absent from those relation rules, whereas changing extents can affect their satisfaction. A hyphen denotes an unreported Micro measurement.

Sampler configuration	FID↓	Micro↑
transport sets only	36.7	0.974
+ orientation projection	31.4	-
+ category-specific size sets	**25.1**	**0.994**
no final refinement	38.2	0.827

I.3 Conflict Detection on SG-FRONT-Conflict

SG-FRONT-Conflict evaluates the warning and localization outputs of the generation pipeline on incompatible order requests. A *cycle conflict* starts from a path $i_1 \to \cdots \to i_k$ of directional edges along one axis. Adding the same relation from the last node to the first closes a contradictory cycle, with $k \in \{3, 4, 5\}$. A *mutual-exclusion conflict* adds requests that cannot hold together, such as left and right on the same ordered pair or a bigger-than cycle in log-scale. These constructions produce difference-constraint contradictions, for which exact negative-cycle detection on the original constraints is complete. We injected conflicts into 40% of test graphs and retained each injected cycle as the reference for locating the conflicting edges (Table 21). Feasible controls had matched graph sizes and densities.

We treat an incompatible graph as a positive case and a feasible control as a negative case. A returned infeasibility warning is a positive prediction. With true-positive, false-negative, false-positive, and true-negative graph counts denoted by TP, FN, FP, and TN, the standard diagnostic rates are (Fawcett, 2006)

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \qquad \mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \qquad \mathrm{FNR} = \frac{\mathrm{FN}}{\mathrm{TP} + \mathrm{FN}} = 1 - \mathrm{TPR}. \tag{20}$$

TPR is recall, also called sensitivity. FPR is the false-positive rate, and FNR is the false-negative rate. Higher TPR and lower FPR and FNR are desirable. A zero observed FPR refers to these evaluated feasible controls. Localization compares the returned conflict support with the stored injected conflicting edges and concerns where the contradiction occurs rather than whether a warning is returned.

The evaluated TOSCA pipeline detects and localizes most injected conflicts, with no observed false positives and complete detection on the tested living-room inputs (Table 21). Some bedroom and dining inputs receive no warning. Localization matches detection at the reported precision. These inputs contain directional and size contradictions, all of which admit a negative-cycle certificate when their original difference constraints are checked exactly (Theorem 2). Table 21 scores the warning and localization outputs returned by the complete pipeline. A missed warning therefore records a failure to report a known contradiction, and the absence of a warning does not establish feasibility. The reported TPR characterizes the pipeline's delivered diagnostics, while the exact test establishes the complete-detection criterion for these inputs.

Violation energy E is the weighted squared-distance objective in Equation 3, evaluated with the same fitted sets, feature metrics, and weights. The violation-energy ratio compares mean final energy on incompatible inputs with mean final energy on feasible controls. Compatible inputs admit near-zero residual energy, while incompatible inputs retain geometric violation after refinement. The denominator is the mean energy of feasible inputs, so a very large ratio reflects that near-zero reference. The ratio describes separation between these input groups and has no universal better direction. The detector supplies the explicit warning.

Table 21: **Contradiction handling on SG-FRONT-Conflict**. Rates score warnings and localized conflicts returned by the complete generation pipeline. TPR, FPR, and FNR use the graph-level denominators defined above. Localization assesses the returned conflict support. Exact negative-cycle checking of each original constraint system would detect every injected contradiction (Theorem 2). The last column divides mean final violation energy on impossible inputs by that on feasible inputs. This ratio describes residual geometric violation relative to the feasible-input reference and has no universal better direction.

Room	Method	TPR↑	FPR↓	Localization↑	FNR↓	E ratio
Bedroom	TOSCA	0.914	0.000	0.914	0.086	4.2×10^{10}
Living	TOSCA	**1.000**	0.000	**1.000**	0.000	1.9×10^{4}
Dining	TOSCA	0.985	0.000	0.985	0.015	3.5×10^{4}

I.4 Relation Editing and Locality

A static generation score does not show how a scene responds when one requested relation changes. Scene-graph manipulation, including relation changes and object additions, is studied by Graph-to-3D (Dhamo et al., 2021), CommonScenes (Zhai et al., 2023), and EchoScene (Zhai et al., 2024). Realizing a revised request while preserving surrounding objects requires evaluating both the requested change and its effect on the rest of the scene. We used a study-specific paired-generation diagnostic on 60 scenes per room. We generated two layouts from identical initial noise, using the original graph and a graph with one edge's relation type changed. Holding noise fixed isolates the response to that graph change from resampling variation. MMGDreamer and InstructScene used model-specific adapters to reuse noise under the same protocol. Table 22 measures the resulting movement together with edit success.

Single-relation edits concentrate movement at the edited object, with substantially smaller displacement at one-hop neighbors. For each object, displacement is the Euclidean distance between its planar positions before and after editing. We denote the median displacement at the designated edited object by D_0 and at its one-hop neighbors by D_1. The reported ratio is D_1/D_0, the ratio of these medians. Local preservation means that objects surrounding the designated edited object retain their planar positions, as measured by their median displacement in meters. We permuted the graph-distance labels to test whether the movement contrast depends on distance from the edit and repeated the same input and noise to check for movement without a graph change. The shuffled

contrast is much smaller, and the repeated-input control gives zero movement. A lower neighbor-to-edited displacement ratio indicates that movement is more concentrated near the edit. These finite-output measurements characterize the tested generation pipeline. Theorem 4 separately establishes distance-dependent sensitivity of anchored optima under its curvature and sparsity assumptions.

Small displacement could also arise if a method ignores the requested edit. We therefore assessed preservation alongside edit success, the fraction of edited requests that the generated layout realizes. Fitted-set edit success checks the changed edge's membership in its transport set, whereas benchmark-rule edit success checks its benchmark predicate. For most feasible edits, the returned layouts satisfy the changed fitted constraint, while success under the benchmark scoring rule is lower because its definition differs. TOSCA's success rates condition on feasible edited graphs, and EchoLayout, MMGDreamer, and InstructScene rates describe success over all requested edits. The feasible-input fractions show that some edited graphs admit no layout satisfying all requested relations. This makes feasibility part of interpreting edit success, since some requested changes cannot be realized while retaining all other relations. TOSCA has the smallest reported one-hop displacement in each room, indicating limited surrounding movement under the tested protocol.

Table 22: **Edit locality and success**. We changed one edge's relation type and reused the same noise. Displacements are median planar distances in meters at graph distance $d = 0$ and $d = 1$ from the edited object. Their ratio measures neighbor displacement relative to edited-object displacement and is rounded to two decimals. Shuffled controls permuted graph-distance labels, while repeating the input without an edit gives 0.000 m movement for all methods. Bold marks the smallest one-hop displacement and ratio. Edited-object displacement and the feasible-input fraction describe the edit and its constraints, with no universal better direction. Feasible is the fraction of edited graphs admitting a layout that satisfies their constraints. TOSCA's fitted-set and benchmark-rule success rates use feasible edits, while EchoLayout, MMGDreamer, and InstructScene rates use all edits, so the denominators differ. MMGDreamer and InstructScene used the adaptations in Sec. 4.1. Hyphens denote unreported baseline feasibility and fitted-set success. For shuffled controls, feasibility and success are marked n/a because permuting graph-distance labels defines no separate edit experiment.

		edit response			input	edit success	
Room	Method	D_0 (m)	D_1 (m)↓	D_1/D_0 ↓	Feasible	Fitted set↑	Benchmark↑
Bedroom	TOSCA (ours)	0.137	**0.016**	**0.12**	0.783	0.957	0.660
	shuffled control	0.177	0.175	0.99	n/a	n/a	n/a
	EchoLayout (Zhai et al., 2024)	0.282	0.035	**0.12**	-	-	0.467
	MMGDreamer (Yang et al., 2025)	0.308	0.060	0.19	-	-	0.488
	InstructScene (Lin & Mu, 2024)	0.080	0.019	0.24	-	-	0.314
Living	TOSCA (ours)	0.340	**0.019**	**0.06**	0.783	0.894	0.766
	shuffled control	0.122	0.118	0.97	n/a	n/a	n/a
	EchoLayout (Zhai et al., 2024)	0.281	0.035	0.12	-	-	0.417
	MMGDreamer (Yang et al., 2025)	0.319	0.054	0.17	-	-	0.385
	InstructScene (Lin & Mu, 2024)	0.294	0.022	0.07	-	-	0.419
Dining	TOSCA (ours)	0.309	**0.020**	**0.06**	0.717	0.930	0.791
	shuffled control	0.150	0.122	0.81	n/a	n/a	n/a
	EchoLayout (Zhai et al., 2024)	0.265	0.041	0.15	-	-	0.350
	MMGDreamer (Yang et al., 2025)	0.285	0.055	0.19	-	-	0.435
	InstructScene (Lin & Mu, 2024)	0.263	0.022	0.08	-	-	0.339

I.5 FOOTPRINT OVERLAP

TOSCA's footprint overlap rate is closest to the data overlap rate among the evaluated methods, while collision guidance moves EchoScene farther from it (Table 23). All methods used the common footprint-overlap metric defined in Appendix E.3. Applying the same test to annotated layouts gives the data overlap rate. This reference contextualizes the frequency of projected overlap in the dataset, including valid arrangements at different heights. Its absolute difference from a generated-layout rate measures agreement with that aggregate frequency. Matching the reference does not establish that the same object pairs overlap or that each overlap is physically valid. The baselines depart from the data rate in both directions, with EchoScene and EchoLayout below it and MMGDreamer and InstructScene above it in every room. Absolute deviation captures both types of departure, so TOSCA's advantage reflects closer agreement with the observed overlap frequency. The Inf3DLayout experiment likewise shows that TOSCA achieves the smallest deviation from the data overlap rate in every room (Appendix F). Physical validity additionally depends on support, reachability, and three-dimensional collision checks.

Table 23: **Footprint overlap and the data reference.** The rates use the common footprint-overlap criterion defined in Appendix E.3. The raw rate measures projected overlap, which can include valid arrangements at different heights and has no universal better direction. The Data overlap rate row applies the same test to annotated layouts. $|\Delta\mathrm{Data}|$ is the absolute difference from this rate, and bold marks the generator with the smallest difference per room. This difference measures agreement with the dataset's overlap frequency. TOSCA rates are three-seed means with seed s.d. of .011 for bedroom, .003 for living, and .008 for dining. Collision guidance reduces overlap but moves its aggregate rate farther from the data rate. MMGDreamer and InstructScene were retrained using the category-only and layout-decoder adaptations, respectively, described in Sec. 4.1.

	Bedroom		Living		Dining	
Method	rate	$\lvert\Delta\mathrm{Data}\rvert \downarrow$	rate	$\lvert\Delta\mathrm{Data}\rvert \downarrow$	rate	$\lvert\Delta\mathrm{Data}\rvert \downarrow$
Data overlap rate	0.549	0.000	0.550	0.000	0.583	0.000
EchoScene (Zhai et al., 2024)	0.465	0.084	0.289	0.261	0.270	0.313
EchoLayout (Zhai et al., 2024)	0.448	0.101	0.492	0.058	0.531	0.052
EchoScene (Zhai et al., 2024) + collision guidance	0.182	0.367	0.238	0.312	0.207	0.376
MMGDreamer (Yang et al., 2025)	0.662	0.113	0.598	0.048	0.621	0.038
InstructScene (Lin & Mu, 2024)	0.718	0.169	0.609	0.059	0.652	0.069
TOSCA (ours)	0.475	**0.074**	0.545	**0.005**	0.559	**0.024**

I.6 QUALITATIVE COMPARISONS

Figure 4 extends the graph-conditioned comparison in Fig. 3 with two dining rooms, a living room, and a bedroom. Each row used the same input scene graph across methods and the evaluation setup in Sec. 4.1. We used the same camera, ground plane, and object mesh identities within each row. EchoScene denotes the released full-model checkpoint. MMGDreamer used category-only graph conditioning, and the InstructScene layout decoder received the supplied graph directly (Appendix E.2).

In the first dining example, six chairs share close-by requests to one table and occupy specified sides of it. TOSCA distributes the chairs along both sides, keeping them near the table and visually distinct. EchoScene and MMGDreamer crowd several chairs into visibly overlapping groups. MMGDreamer also places the chair marked with a right-of mismatch over the table region rather than on the requested side. InstructScene concentrates several chairs at overlapping positions. The example illustrates why proximity to a shared table must be coordinated with correct side placement to form a coherent seating arrangement.

The second dining example additionally specifies two symmetrical-to chair pairs and a close-by relation between the table and wine cabinet. TOSCA places four chairs in two opposing pairs beside the table, with the cabinet nearby. EchoScene leaves a chair separated from the table in the foreground, and MMGDreamer places the chairs unevenly across the two sides. InstructScene gathers two chairs

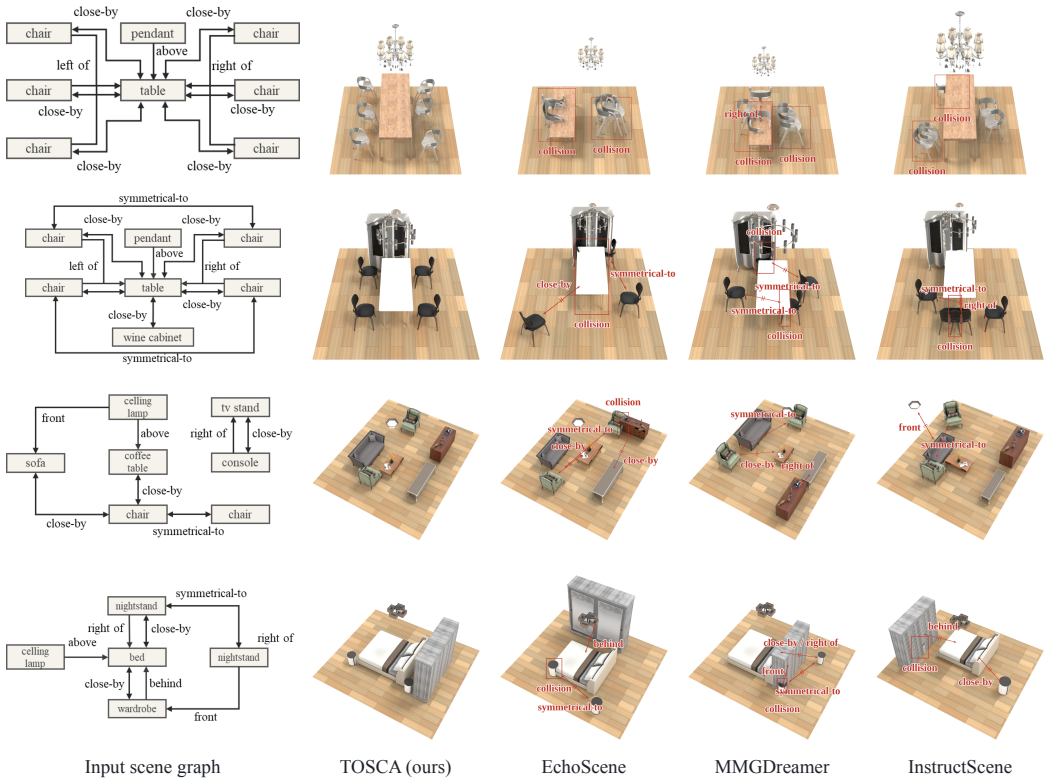


Figure 4: **Further graph-conditioned layout comparisons on SG-FRONT.** Rows show two dining rooms, a living room, and a bedroom. Columns show the input scene graph, TOSCA, EchoScene, MMGDreamer, and InstructScene. In the rendered layouts, slashed red arrows mark selected relation violations, and red boxes highlight visible collisions between objects.

at the near end of the table, including the chair marked with a right-of mismatch. Their concentration near the table centerline disrupts the requested side placement and paired arrangement. These differences show how proximity, direction, and symmetry jointly shape the requested layout.

In the living-room example, the graph couples chair proximity and symmetry with two directional requests. The console must be right of the TV stand, and the ceiling lamp must be in front of the sofa. TOSCA combines these relative placements with a chair beside both the sofa and coffee table and its counterpart across the room. EchoScene leaves wider gaps between the sofa-side chair and coffee table and between the console and TV stand. MMGDreamer displaces the console to the opposite end of the TV stand, producing the marked right-of mismatch. InstructScene displaces the ceiling lamp to the opposite side of the sofa, producing the marked front mismatch. Recognizable seating areas can therefore coexist with incorrect ordering of particular object pairs.

The last bedroom example requests symmetrical-to nightstands, with one right of and close to the bed, while the other is in front of the wardrobe. The wardrobe must also be behind and close to the bed. TOSCA combines these relative placements with a paired nightstand arrangement. EchoScene and InstructScene displace the wardrobe to different sides of the bed, as highlighted by the behind annotations. MMGDreamer places one nightstand away from the requested side of the bed, producing the marked right-of and close-by mismatches, while the other nightstand produces the marked front mismatch relative to the wardrobe. This example highlights the need to coordinate one object's relations to multiple neighbors, since moving a nightstand or wardrobe affects several requests simultaneously.

Across these examples, TOSCA maintains distinct object placements while coordinating proximity, symmetry, and directional ordering around shared objects. This behavior is consistent with transport sets retaining allowed regions and projection residuals supplying corrections for each edge, which

graph aggregation and final refinement combine across the scene. The marked directional mismatches illustrate how individual requests can fail even when aggregate directional scores are high (Table 2). The qualitative comparisons thus complement both edge-level and scene-level satisfaction, while Table 18 separately quantifies the contributions of allowed regions and final refinement.

I.7 User Study

We compared perceived layout plausibility and relation adherence by showing all four methods together, following the multi-method ranking evaluations in FreeScene and SceneNAT (Bai et al., 2025; Choi et al., 2026a). Our criteria follow perceptual studies of scene plausibility and adherence to graph or text requests (Zhai et al., 2023; Yang et al., 2026). We randomly selected 30 held-out SG-FRONT graphs with scored relations, with 10 per room type, before inspecting generated outputs. Each comparison set contained TOSCA, retrained EchoLayout, category-only MMGDreamer, and the InstructScene layout decoder under the input adaptations in Sec. 4.1.

Each method generated one layout per graph using a fixed sampling assignment. The four layouts shared object meshes and materials, lighting, and one top-down and two oblique views while retaining each method's predicted poses and scales. They were displayed simultaneously with matched views. Method names and automatic-score overlays were hidden, and display positions and trial order were randomized.

The analysis included 30 adult participants, with 15 assigned to each of two independent panels. The plausibility panel ranked the four furniture arrangements from most to least plausible using the scene views alone. The relation-adherence panel ranked them by fulfillment of up to three distinct requests sampled from the input graph after collapsing redundant inverse descriptions. The requests were fixed across methods and raters, with object IDs and coordinate guides connecting them to the views. Each participant evaluated 30 comparison sets, balanced across rooms, without seeing the same graph twice. Each graph received 15 judgments per panel. Participants could assign tied ranks or indicate that they could not judge the set.

Table 24 reports Top-1 and Top-2 preference shares for every method and each criterion. For each judgment, Top-1 assigns one unit of credit to the first rank, and Top-2 assigns one unit to each of the first two ranks. Tied methods share the credit for their occupied rank positions equally. We averaged each method's credits within graphs, then equally across graphs within rooms and across room types. We computed 95% confidence intervals using a crossed participant-and-graph cluster bootstrap (Owen & Eckles, 2012).

Table 24: **User-study preference across all methods.** Each entry reports a preference share p in percent with its 95% confidence interval $[L, U]$. Top-2 includes both first and second ranks, with tied methods sharing rank credit. Higher shares indicate greater preference within the same four-method comparison. Bold marks the highest preference share in each column. All four methods shared the same trials within a panel.

	Layout plausibility		Relation adherence	
Method	Top-1 (%)↑	Top-2 (%)↑	Top-1 (%)↑	Top-2 (%)↑
EchoLayout (Zhai et al., 2024)	16.0 [12.6, 19.4]	40.0 [35.5, 44.5]	10.0 [7.2, 12.8]	34.0 [29.6, 38.4]
MMGDreamer (Yang et al., 2025)	14.0 [10.8, 17.2]	38.0 [33.5, 42.5]	8.0 [5.5, 10.5]	32.0 [27.7, 36.3]
InstructScene (Lin & Mu, 2024)	24.0 [20.0, 28.0]	54.0 [49.3, 58.7]	26.0 [21.9, 30.1]	58.0 [53.4, 62.6]
TOSCA (ours)	**46.0** [41.4, 50.6]	**68.0** [63.7, 72.3]	**56.0** [51.4, 60.6]	**76.0** [72.0, 80.0]

TOSCA receives the highest Top-1 and Top-2 preference shares for both layout plausibility and relation adherence (Table 24). InstructScene is the strongest baseline under both criteria, and TOSCA's lead over it is larger for relation adherence. The lead in both ranking summaries indicates that the preference extends from first-place choices to inclusion among the two preferred layouts. Because the displays shared meshes, materials, lighting, and views, the comparison associates this preference with the predicted object poses and scales. These preferences complement the fidelity and geometric relation-satisfaction results in Tables 1 and 2 by directly assessing perceived arrangement quality and fulfillment of the displayed requests.