# SCAFFOLDTREE: SCAFFOLD-TO-SUBTREE DIFFUSION FOR 3D TREE GENERATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

We present ScaffoldTree, a species-conditioned diffusion framework for structured 3D tree generation. ScaffoldTree decomposes a tree according to the structural roles of its branches: the trunk and major branches form a global scaffold, while fine-scale branches are represented as locally attached residual subtrees. Global Scaffold Diffusion first generates the scaffold by jointly modeling branch curves, attachment locations, and parent relationships. The Structure-Conditioned Residual Allocator then predicts the total number of residual branches from the generated scaffold and distributes them across scaffold anchors. Conditioned on the scaffold and allocation, Local Residual Diffusion generates the corresponding residual subtrees, which are hierarchically transformed according to their predicted parent–child relations and merged with the scaffold to form the complete tree. This decomposition separates global tree organization from local branching detail while maintaining their structural dependence. Experiments demonstrate improved structural and local-curve distribution matching and reference coverage compared with the evaluated baselines.

## 1 INTRODUCTION

Three-dimensional tree structures are important for ecological analysis (Mo et al., 2023; Nunes et al., 2023), digital forestry (Tang et al., 2024), and realistic virtual environments (Lee et al., 2023; Wang et al., 2025). Generating such structures remains challenging because tree architecture is inherently multi-scale. The trunk and major branches establish the global crown organization, whereas numerous higher-order branches contribute fine-scale geometric and topological detail. Modeling these two scales jointly requires long-range coordination of major branches while preserving dense local variation. Capturing both within a coherent generative representation remains challenging.

Procedural and biologically inspired methods generate tree structures through rewriting rules (Prusinkiewicz & Lindenmayer, 1990), parametric models (Weber & Penn, 1995), or simulated growth (Runions, 2007), but often require manual design and species-specific tuning. Learning-based approaches model local growth decisions (Zhou et al., 2023), hierarchical rules (Lee et al., 2023), or autoregressive branch sequences (Wang et al., 2025). Despite these advances, global and fine-scale branches are typically generated within a common process, although they impose different modeling requirements. Major branches require long-range coordination to establish the overall crown structure, whereas dense higher-order branching requires sufficient representational capacity to capture local variation. When both scales are modeled within a shared fixed-capacity representation, fine-scale branches can compete with major branches for representational resources, potentially weakening global organization or limiting local detail. This motivates an explicit separation of global structure and local branching.

To address this challenge, we introduce ScaffoldTree, a species-conditioned diffusion framework for 3D tree generation. Its key design is a *scaffold-to-subtree decomposition* that separates the trunk and major branches defining global organization from higher-order branches providing local detail, as illustrated in Figure 1. Global Scaffold Diffusion generates the global scaffold, after which the Structure-Conditioned Residual Allocator predicts the residual branch count and distributes it across scaffold anchors. Conditioned on the scaffold and allocated counts, Local Residual Diffusion generates the corresponding residual subtrees, which are hierarchically assembled into the complete tree. Global Scaffold Diffusion and Local Residual Diffusion use a hybrid continuous–discrete



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.




(a) Scaffold (b) Residual Subtrees (c) Complete Tree (d) Alternative View

Figure 1: ScaffoldTree first generates the trunk and major branches as a global scaffold (a), then attaches generated residual subtrees to recover local geometric and topological details (b). Hierarchical assembly produces the complete tree and its rendered foliage (c,d).

formulation that jointly models continuous branch geometry and discrete parent topology, preserving explicit branching relationships throughout generation.

Experiments show that ScaffoldTree achieves improved structural-distribution matching, local-curve distribution matching, and reference coverage over the evaluated baselines while remaining competitive in point-set distance. Our contributions are:

- We introduce a scaffold-to-subtree decomposition that separates global tree organization from fine-scale branching according to their structural roles.
- We develop a hybrid continuous–discrete diffusion formulation that jointly models continuous branch geometry and explicit discrete parent topology at both global and local scales.
- We introduce a Structure-Conditioned Residual Allocator that predicts local branching complexity and distributes residual branches across the generated scaffold.

## 2 Related Work

**Tree Modeling.** Early geometric plant modeling relies on procedural methods, including L-systems, parametric models, and space colonization (Prusinkiewicz & Lindenmayer, 1990; Weber & Penn, 1995; Runions, 2007), which generate branching through recursive rewriting, geometric parameters, and distributed attraction points, respectively. Another line reconstructs tree geometry from images (Tan et al., 2007; Guo et al., 2018; Li et al., 2021; Liu et al., 2025b) or 3D point clouds (Hu et al., 2017; Liu et al., 2021; Zhou et al., 2025; Marks et al., 2025). Learning-based approaches enable data-driven generation: DeepTree (Zhou et al., 2023) predicts local branch growth from situated latents; Latent L-systems (Lee et al., 2023) combine hierarchical L-string representations with Transformers to model geometry and topology; and Wang et al. (Wang et al., 2025) generate trees autoregressively using a multi-resolution hourglass Transformer, supporting complex structures, conditional generation, and temporal growth. Although these approaches reduce handcrafted rules, they generally model global organization and fine-scale branching within a common generative process.

**Structured 3D Generation.** Structured 3D generation models part geometry and relationships using hierarchical graphs, disentangled representations, and executable programs (Mo et al., 2019; Jones et al., 2020; Yang et al., 2022). Part-aware methods decompose objects into meaningful components and model their spatial organization (Hertz et al., 2022; Nakayama et al., 2023; Koo et al., 2023), while diffusion approaches generate part geometry and spatial configurations (Nakayama et al., 2023; Koo et al., 2023; Ding et al., 2026). Recent work further explores multi-view diffusion, contextual part latents, and coarse-to-fine generation with part connectivity (Chen et al., 2025; Dong et al., 2025; Liu et al., 2025a; Yang et al., 2025; Li et al., 2026). However, these methods primarily target semantic parts or articulated assemblies, whereas trees exhibit recursive branching topology with explicit attachment locations and parent–child relations. ScaffoldTree therefore treats branches as structural primitives and organizes generation through a scaffold-to-subtree hierarchy, jointly modeling continuous branch geometry and discrete topology.

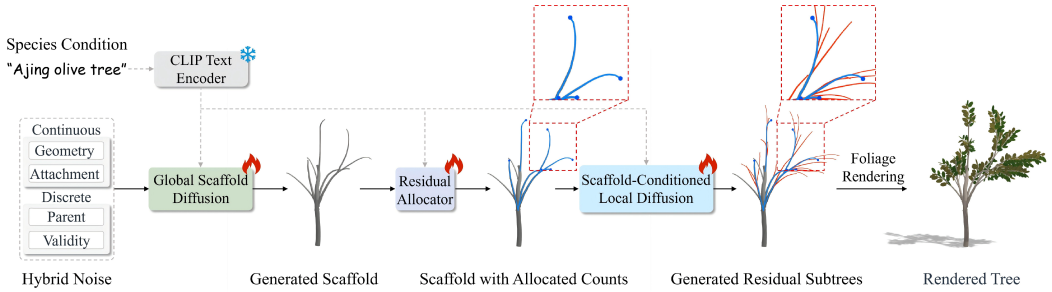



Figure 2: Overview of ScaffoldTree. Global Scaffold Diffusion first generates the scaffold. The allocator predicts branch-wise residual counts under a global count constraint, after which local residual diffusion generates a subtree for each selected scaffold anchor. All residual subtrees are merged with the scaffold to form the complete 3D tree. Gray denotes the scaffold, while red and blue distinguish residual subtrees attached to different anchors.

## 3 METHOD

Our goal is to generate a complete 3D tree with coherent global structure and detailed local branching. ScaffoldTree decomposes generation into three stages, as illustrated in Figure 2: global scaffold generation, residual branch allocation, and local residual subtree generation. This decomposition separates global organization from fine-scale branching while preserving their hierarchical dependence.

### 3.1 TREE REPRESENTATION AND SCAFFOLD DECOMPOSITION

We represent a 3D tree as a rooted collection of branches:

$$\mathcal{T} = \{(\mathbf{X}_i, a_i, p_i, \ell_i)\}_{i=1}^{N}, \tag{1}$$

where $N$ is the number of branches and $\mathbf{X}_i = [(x_{i,j}, y_{i,j}, z_{i,j}, \rho_{i,j})]_{j=0}^{P-1} \in \mathbb{R}^{P\times 4}$ contains $P$ sampled 3D points and their radii. For each non-root branch, $p_i$ is its parent and $a_i \in [0,1]$ is the normalized arc-length attachment position on the parent curve. The root has no parent or attachment. Topological levels satisfy $\ell_{\text{root}} = 0$ and $\ell_i = \ell_{p_i} + 1$; branches at larger levels are referred to as higher-order branches. Branches follow an attachment-ordered depth-first traversal, so every parent precedes its descendants and sibling order follows attachment position. We convert TreeNet3D (Tang et al., 2024) trees into this representation.

Branches play different structural roles: the trunk and major branches determine global shape, while higher-order branches mainly contribute local detail. We therefore decompose each tree into a global scaffold $\mathcal{S} = \{S_i\}_{i=1}^{N_G}$ and residual subtrees $\{\mathcal{R}_i\}_{i=1}^{N_G}$, where $\mathcal{R}_i$ is assigned to scaffold anchor $S_i$. For branch $i$, its structural saliency $v_i$ is defined from length $L_i$, mean radius $R_i$, descendant count $D_i$, and subtree extent $C_i$:

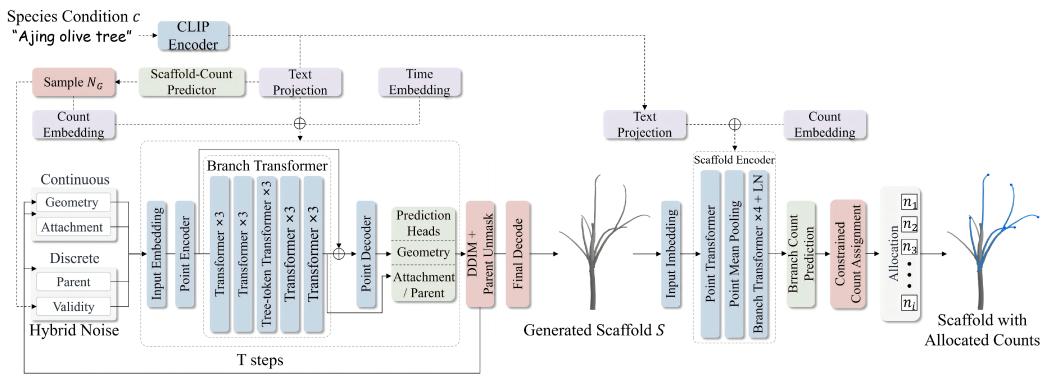
$$v_i = \lambda_L \widehat{L}_i + \lambda_R \widehat{R}_i + \lambda_D \widehat{\log(1+D_i)} + \lambda_C \widehat{C}_i. \tag{2}$$

Salient branches are retained under an ancestry constraint to form the scaffold, while each remaining branch is assigned to its nearest retained ancestor. The resulting residual group $\mathcal{R}_i$ may contain multiple roots directly attached to scaffold branch $i$; together with its scaffold anchor, it forms a connected local subtree.

Let $\mathbf{n} = (n_1, \ldots, n_{N_G})$ denote the residual allocation, where $n_i$ is the residual branch count associated with scaffold branch $i$. The total residual count is $N_R = \sum_{i=1}^{N_G} n_i$, and therefore $N = N_G + N_R$. Let $c$ denote the species condition. We factorize the conditional tree distribution as

$$p(\mathcal{T} \mid c) = p_G(\mathcal{S} \mid c)\, p_A(\mathbf{n} \mid \mathcal{S}, c) \prod_{i=1}^{N_G} p_L(\mathcal{R}_i \mid \mathcal{S}, i, n_i, c), \tag{3}$$

where $p_G$, $p_A$, and $p_L$ model scaffold generation, residual allocation, and local residual generation.




(a) Global Scaffold Diffusion (b) Residual Allocation

Figure 3: Global scaffold generation and structure-conditioned residual allocation in ScaffoldTree. (a) Global Scaffold Diffusion jointly models branch geometry and topology through a species-conditioned hybrid continuous–discrete diffusion process. (b) The allocator encodes the generated scaffold to predict the total residual branch count and its distribution across scaffold anchors under a global count constraint.

### 3.2 Global Scaffold Diffusion

The Global Scaffold Diffusion model $p_G(\mathcal{S} \mid c)$ generates the global scaffold conditioned on species identity. The species condition $c$ is encoded by a frozen text encoder (Radford et al., 2021) and injected into the denoising network. A scaffold-count predictor additionally estimates species-dependent structural complexity. Figure 3 (a) illustrates the architecture of the proposed Global Scaffold Diffusion model.

**Hybrid continuous–discrete diffusion.** Continuous branch geometry and attachment positions follow the Gaussian forward process of DDPMs (Ho et al., 2020):

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{4}$$

where $\mathbf{x}_0$ is the clean continuous variables, $\mathbf{x}_t$ their noisy state at timestep $t$, and $\bar{\alpha}_t$ is the cumulative noise schedule. In the final fixed-grid configuration, the scaffold count is sampled from the species-conditioned count predictor, after which the branch-existence and point-validity masks are fixed. Parent assignments constitute the discrete diffusion state and follow an absorbing-state process (Austin et al., 2021):

$$q(z_t \mid z_0) = \bar{\alpha}_t\, \delta(z_t = z_0) + (1-\bar{\alpha}_t)\, \delta(z_t = \mathtt{M}), \tag{5}$$

where $z_0$ and $z_t$ are the clean and corrupted parent labels, $\delta$ denotes a categorical point mass, and $\mathtt{M}$ is the absorbing mask state. Branch levels are predicted by a separate auxiliary head rather than treated as a diffusion state.

At each step, point features are aggregated into branch representations $\mathbf{h}_i$ and processed by global self-attention, while a learnable tree token summarizes scaffold-level information. Parent compatibility is computed as

$$s_{ij} = \frac{(W_q\mathbf{h}_i)^\top (W_k\mathbf{h}_j)}{\sqrt{d}}, \tag{6}$$

where $W_q$ and $W_k$ are learned projections, $d$ is the feature dimension, and $s_{ij}$ scores branch $j < i$ as parent of branch $i$. Restricting candidates to preceding branches ensures acyclic topology. Continuous geometry and discrete parent variables share the same timestep and are jointly denoised.

**Branch geometry representation.** To separate global branch extent from local shape variation, we use a chord-residual parameterization in branch-local coordinates. Preprocessing places the branch start at the local origin, $\mathbf{q}_{i,0} = \mathbf{0}$. Let $\mathbf{q}_{i,j}$ be the $j$-th point, $\mathbf{e}_i = \mathbf{q}_{i,P-1}$ the endpoint, and $u_j = j/(P-1)$ the within-branch sampling coordinate, which is distinct from the parent-curve

attachment variable $a_i$. The straight chord and residual are

$$\begin{aligned} \mathbf{b}_{i,j} &= (1-u_j)\mathbf{q}_{i,0} + u_j\mathbf{q}_{i,P-1} = u_j\mathbf{e}_i, \\ \mathbf{r}_{i,j} &= \mathbf{q}_{i,j} - \mathbf{b}_{i,j}, \qquad \mathbf{q}_{i,j} = \mathbf{b}_{i,j} + \mathbf{r}_{i,j}. \end{aligned} \tag{7}$$

Thus, $\mathbf{b}_{i,j} = u_j\mathbf{e}_i$ follows from the local zero-origin convention rather than an assumption about world coordinates. These equations apply to the XYZ channels; radius retains its normalized pointwise representation. The first token stores $\mathbf{e}_i$ in its XYZ channels, while the interior XYZ channels encode chord residuals; decoding fixes the first and last positions to $\mathbf{0}$ and $\mathbf{e}_i$, respectively. The scaffold model is optimized jointly for geometry and topology with additional geometric and whole-tree structural regularization; principal objectives and optimization details are provided in Appendix A.4.

### 3.3 Structure-Conditioned Branch Allocation

The allocator produces a residual-total distribution on the fixed label set $\{0, \ldots, B_{\max}\}$, where the full-tree capacity is $B_{\max} = 192$ and the per-anchor residual capacity is $K = 32$. For a generated scaffold containing $N_G$ branches, decoding restricts this distribution to the feasible residual capacity

$$C_R = \min(B_{\max} - N_G,\, N_G K), \qquad \hat{N}_R = \arg\max_{0 \le r \le C_R} p_A(N_R = r \mid \mathcal{S}, c). \tag{8}$$

Values above $C_R$ are excluded rather than added to the scaffold count; hence $N_G + \hat{N}_R \le B_{\max}$. For scaffold branch $i$, let $h_i$ be the occupancy logit and $\mathbf{z}_i$ the logits over positive counts:

$$p(n_i = 0 \mid \mathcal{S}, c) = \sigma(-h_i), \tag{9}$$

$$p(n_i = k \mid \mathcal{S}, c) = \sigma(h_i)\,\mathrm{Softmax}(\mathbf{z}_i)_k, \qquad k = 1, \ldots, K. \tag{10}$$

Here, $\sigma(\cdot)$ denotes the logistic sigmoid.

Let $s_i(k) = \log p(n_i = k \mid \mathcal{S}, c)$. To discourage an unrealistically large number of residual branches from concentrating on a single scaffold anchor, we apply a deterministic species-aware soft-cap cost:

$$\widetilde{s}_i(k) = s_i(k) - \lambda_{\text{over}} \frac{\max(k - \kappa_c, 0)^2}{\kappa_c}, \tag{11}$$

where $\kappa_c$ is the species-specific anchor soft cap estimated as the 90th percentile of the maximum per-anchor load in the training set, and we use $\lambda_{\text{over}} = 2.0$. This decoder-only adjustment does not modify the learned allocator parameters; the per-species values of $\kappa_c$ are reported in Appendix A.3.

To enforce consistency with the predicted global count, all branch counts are inferred jointly by

$$\mathbf{n}^* = \arg\max_{\mathbf{n}} \sum_{i=1}^{N_G} \widetilde{s}_i(n_i), \quad \text{s.t.} \quad \sum_{i=1}^{N_G} n_i = \hat{N}_R,\ 0 \le n_i \le K. \tag{12}$$

We solve this constrained MAP problem exactly using dynamic programming. Let $D(i, r)$ denote the best score for assigning $r$ residual branches to the first $i$ scaffold anchors:

$$D(i, r) = \max_{0 \le k \le \min(K, r)} \left[ D(i-1, r-k) + \widetilde{s}_i(k) \right]. \tag{13}$$

The boundary conditions are $D(0, 0) = 0$ and $D(0, r) = -\infty$ for $r > 0$. Backtracking from $D(N_G, \hat{N}_R)$ yields the deterministic allocation $\mathbf{n}^*$ while exactly preserving the predicted global residual count. Figure 3(b) illustrates this allocation stage.

### 3.4 Local Residual Diffusion

Local Residual Diffusion models $p_L(\mathcal{R}_i \mid \mathcal{S}, i, n_i, c)$ for each scaffold branch with $n_i > 0$.

**Fixed-count local generation.** Each residual branch $j$ is represented in a local frame defined by its parent $\pi(j)$. For a root of the residual group, $\pi(j)$ is scaffold anchor $i$; otherwise, it is a preceding residual branch. Let $\mathbf{o}_j$ be the point interpolated at attachment arc $a_j$ on the parent curve and let $\mathbf{R}_j \in \mathbb{R}^{3\times3}$ be the frame constructed from the parent tangent and global up direction:

$$\mathbf{q}_j^L = \mathbf{R}_j^\top(\mathbf{q}_j^W - \mathbf{o}_j), \qquad \mathbf{q}_j^W = \mathbf{R}_j\mathbf{q}_j^L + \mathbf{o}_j. \tag{14}$$

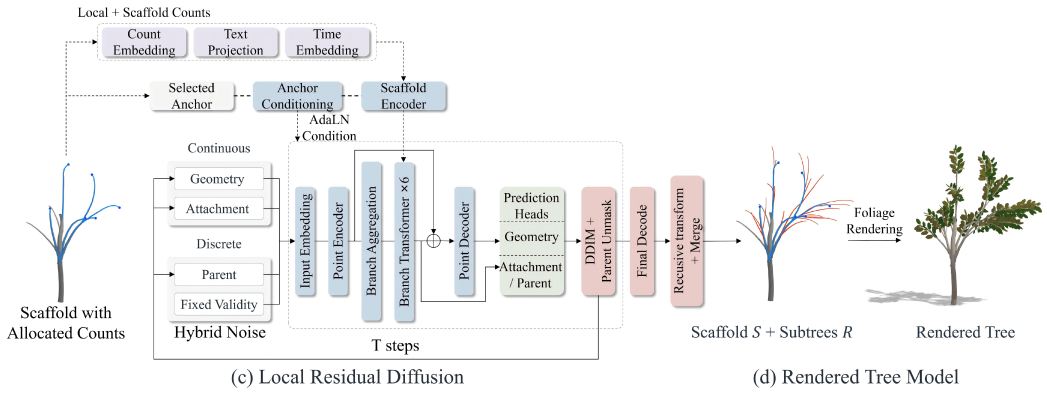



Figure 4: Architecture of Local Residual Diffusion and hierarchical assembly. Given the generated scaffold and branch allocation, hybrid diffusion generates local residual subtrees conditioned on the selected anchor and full scaffold context. The generated subtrees are hierarchically transformed into the complete tree, followed by foliage rendering for visualization.

This parent-relative canonicalization removes absolute translation and orientation while preserving hierarchical attachment geometry. Since $n_i$ is fixed by the allocator and all active branches use the fixed $P = 16$ point grid, Local Residual Diffusion applies Gaussian diffusion to curves and attachment arcs and absorbing-state diffusion only to parent labels. Branch allocation is therefore decoupled from local structure generation. Local branch curves use the same chord-residual parameterization as Global Scaffold Diffusion, so the model predicts deviations from a straight parent-relative chord rather than unconstrained point coordinates.

**Scaffold-conditioned local structure.** The complete scaffold is encoded as $\mathbf{H}^S = [\mathbf{h}_1^S, \ldots, \mathbf{h}_{N_G}^S] \in \mathbb{R}^{N_G \times d}$, where $\mathbf{h}_i^S$ is the feature token of scaffold branch $S_i$. Residual branch tokens first model local dependencies through self-attention and then attend to $\mathbf{H}^S$ through cross-attention. Local topology is defined relative to the scaffold anchor: parent class 0 denotes the anchor, while the remaining classes denote preceding residual branches. Restricting parents to this candidate set enables hierarchical local branching while preserving acyclicity.

After denoising, residual branches are transformed to world coordinates in parent-before-child order using their predicted attachments and parent relations. Local parent indices are then mapped to global branch indices and merged with the scaffold, as illustrated in Figure 4.

# 4 Experiments

## 4.1 Experimental Setup

**Dataset and Implementation.** We evaluate nine TreeNet3D species (Tang et al., 2024): flamboyant, camphor, Ajing olive, fir, small-leaved olive tree, flooded gum, Tibetan cherry, Lombardy poplar, and lemon. We exclude agarwood because many instances contain multiple level-0 trunks, which violate the single-root assumption of our representation. All learned methods use the split of 11,847 training, 270 validation, and 270 test trees, with 30 trees per species in validation and test sets. Each branch is represented by 16 sampled skeleton points, with scaffold and residual-subtree capacities set to 64 and 32 based on the training distribution. Species names are encoded by a frozen CLIP ViT-B/32 text encoder. The three modules are trained with AdamW and batch size 8 on NVIDIA RTX 5090 GPU. Global scaffold diffusion is trained for 40 epochs at $5 \times 10^{-6}$, while the Allocator and scaffold-conditioned local diffusion are trained for 30 epochs at $2 \times 10^{-5}$.

**Baselines.** As public implementations of species-conditioned 3D tree generation methods are limited, we compare ScaffoldTree with two representative methods. *Latent L-Systems* (Lee et al., 2023) uses the released three-layer encoder–decoder Transformer and converts our branch representation into quantized hierarchical L-string rules; following its species-specific grammar, we train

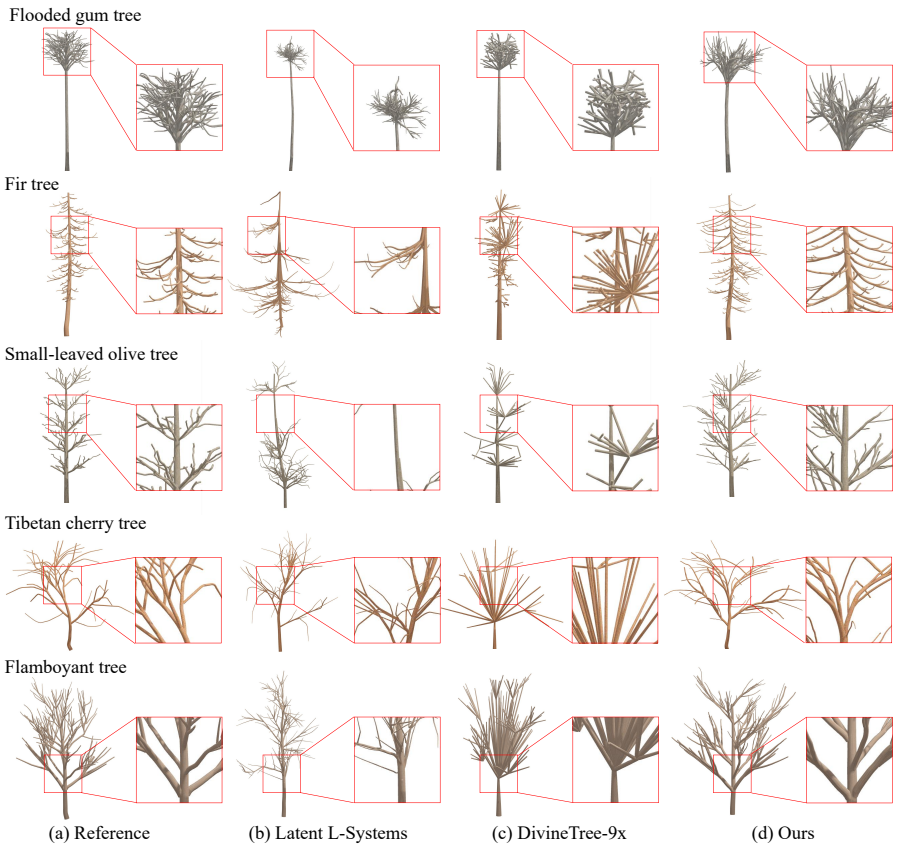



Figure 5: Qualitative comparison on five selected TreeNet3D species. Enlarged regions highlight local branching structures around major junctions. Complete qualitative results for all nine species are provided in Appendix A.1.

one model per species. *DivineTree-9x* (Xu et al., 2025) uses the released 16-layer, width-768 DivineTree diffusion architecture. Since it lacks species conditioning, we train nine independent unconditional models, one per species, with branch counts sampled from each species' empirical training distribution. Both baselines use the same train/validation/test split as ScaffoldTree.

**Evaluation protocol.** For each method, we generate 200 trees per species and evaluate them on the test split. Outputs are represented as upright 3D tree skeletons, with their roots translated to the origin without rotation or scale alignment. We sample 128 branch points proportional to arc length and exclude radius for fair comparison. Fidelity averages the nearest-reference symmetric squared Chamfer distance (Fan et al., 2017) over generated samples, whereas minimum matching distance (MMD) (Achlioptas et al., 2018) averages the nearest-generated distance over references. Coverage (Achlioptas et al., 2018) is the fraction of reference trees selected as the nearest neighbor of at least one generated sample. We also report structural Wasserstein-1 distance without depth (Struct. W1-ND), curve Wasserstein-1 distance (Curve W1) (Vallender, 1974), branch collision rate, and topology validity. The depth-inclusive Struct. W1 and its depth component are reported in Appendix A.7. Metrics are computed per species and macro-averaged over nine species. Detailed metric definitions and point-sampling sensitivity results are provided in Appendices A.5 and A.7.

### 4.2 Qualitative Analysis

Figure 5 presents results for five species: Flooded gum, Fir, Small-leaved olive, Tibetan cherry, and Flamboyant tree. All methods capture coarse species-specific shapes but differ in local branching. Latent L-Systems often produces sparse or uneven higher-order branches, particularly for Fir and Small-leaved olive. DivineTree-9x preserves tree extent but forms dense radial structures near

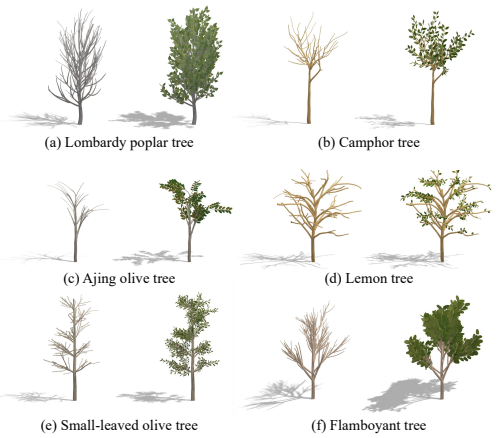



Figure 6: Generated branches (left) and corresponding foliage renderings (right).

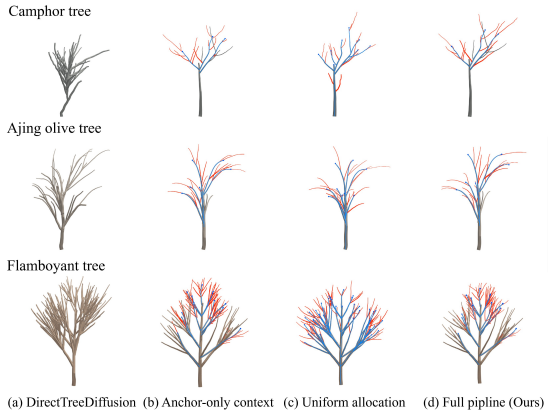



Figure 7: Qualitative ablation comparison across three selected TreeNet3D species.

major junctions, weakening parent–child hierarchy. In contrast, ScaffoldTree preserves coherent branching hierarchies, with major branches organizing the crown and higher-order branches developing along them. Enlarged regions reveal clearer branch relationships and better spatial separation. ScaffoldTree also retains Fir's vertical hierarchy and Tibetan cherry's lateral expansion. These observations align with improvements in structural matching, local curve statistics, and collision rate. Figure 6 presents corresponding foliage renderings.

### 4.3 Quantitative Analysis

Table 1: Comparison on the TreeNet3D test set. Results are macro-averaged over nine species. Struct. denotes Struct. W1-ND, Curve denotes Curve W1, and Coll. denotes the non-adjacent cross-branch collision rate.

| Method | MMD ↓ | Cov. ↑ | Struct. ↓ | Curve ↓ | Coll. (%) ↓ |
|---|---|---|---|---|---|
| Latent L-Systems | 0.0161 | 0.6519 | 1.0301 | 0.4136 | 1.711 |
| DivineTree-9x | **0.0073** | 0.7481 | 1.0204 | 1.0652 | 3.281 |
| **ScaffoldTree (ours)** | 0.0101 | **0.8148** | **0.7095** | **0.3960** | **0.820** |

Table 1 compares the distribution matching of the evaluated methods. Relative to Latent L-Systems, ScaffoldTree improves MMD, coverage, structural agreement, and local-curve matching. Although DivineTree-9x achieves the lowest MMD, ScaffoldTree yields better coverage, structural and curve distribution matching, and fewer branch collisions. DivineTree-9x also exhibits a higher average maximum depth than the held-out data (6.12 vs. 2.00). Since depth is excluded from Struct. W1-ND, its weaker structural agreement cannot be attributed to depth alone. These results highlight the distinction between point-set proximity and plausible branching organization. Both baselines require nine species-specific parameter sets, whereas ScaffoldTree uses a single shared species-conditioned model. Additional fidelity, diversity, validity, and component-wise results are provided in Appendix A.7.

### 4.4 Ablation Studies

We ablate hierarchical decomposition, structure-conditioned allocation, and full-scaffold conditioning. *DirectTreeDiffusion* generates the entire tree in one diffusion stage; *Uniform allocation* distributes a fixed predicted residual total uniformly across scaffold anchors; and *Anchor-only context* conditions local subtrees only on their anchor, allocated count, and species. Table 2 and Figure 7 present quantitative and qualitative comparisons, respectively. Full, Uniform, and Anchor-only share

Table 2: End-to-end ablation. Full gives the best fidelity, MMD, structural agreement, and diversity ratio, while Full and Uniform tie in coverage. Struct. denotes Struct. W1-ND; diversity is optimal when close to one.

| Variant | Fidelity ↓ | MMD ↓ | Cov. ↑ | Struct. ↓ | Diversity → 1 |
|---|---|---|---|---|---|
| DirectTreeDiffusion | 0.0476 | 0.0163 | 0.8000 | 1.8496 | 5.7382 |
| Uniform allocation | 0.0197 | 0.0110 | **0.8259** | 0.9357 | 2.9207 |
| Anchor-only context | 0.0194 | 0.0104 | 0.8222 | 0.7598 | 2.8411 |
| **Full pipeline (Ours)** | **0.0185** | **0.0100** | **0.8259** | **0.7255** | **2.7788** |

Table 3: Allocator evaluation on 270 ground-truth scaffolds with the residual total fixed to its oracle value. The ground-truth active-anchor ratio is 63.21%.

| Allocation | Occ. F1 ↑ | Count MAE ↓ | Height EMD ↓ | Active (%) |
|---|---|---|---|---|
| Uniform | 0.7548 | 1.9813 | 0.0733 | 99.58 |
| **Learned** | **0.9864** | **0.3031** | **0.0089** | 63.74 |

Table 4: Local-context ablation. Full-scaffold context improves junction orientation and reduces sibling collisions while preserving curve quality.

| Local conditioning | Curve W1 ↓ | Junction W1 ↓ | Sibling coll. (%) ↓ |
|---|---|---|---|
| Anchor only | 0.3942 | 0.5830 | 6.122 |
| **Full scaffold** | **0.3930** | **0.4723** | **5.068** |

the same 1,800 species–seed instances, generated scaffold, and predicted residual total per instance. Further ablation details and bootstrap analyses are provided in Appendices A.6 and A.7.

**Hierarchical decomposition.** Compared with DirectTreeDiffusion, ScaffoldTree reduces Fidelity by 61.2%, MMD by 38.4%, and Struct. W1-ND by 60.8% (Table 2). Figure 7(a,d) shows clearer hierarchical organization, with major branches forming a global scaffold and fine-scale lateral branches emerging as attached residual subtrees. These gains support modeling global organization and local branching through separate but hierarchically dependent stages.

**Structure-conditioned allocation.** Replacing learned allocation with Uniform allocation increases MMD from 0.0100 to 0.0110 and Struct. W1-ND from 0.7255 to 0.9357, despite identical residual totals and coverage (Table 2). Figure 7(c,d) shows that learned allocation places residual branches according to scaffold structure rather than distributing them uniformly. It also reduces Count MAE by 84.7% and Height EMD by 87.9% (Table 3). Its active-anchor ratio closely matches the reference (63.74% vs. 63.21%), indicating that it captures both the amount and spatial organization of residual branching.

**Full-scaffold conditioning.** Full-scaffold conditioning reduces Struct. W1-ND from 0.7598 to 0.7255 (Table 2). Although overall morphology remains similar to Anchor-only context (Figure 7(b,d)), full-scaffold conditioning improves local branching organization, reducing Junction W1 from 0.5830 to 0.4723 and sibling collisions from 6.122% to 5.068%, while Curve W1 remains nearly unchanged (Table 4). These results indicate better coordination between locally generated branches and the surrounding scaffold.

## 5 Conclusion and Limitations

We presented ScaffoldTree, a species-conditioned diffusion framework that separates global scaffold generation, residual branch allocation, and local subtree synthesis. Experiments demonstrate improved reference coverage and structural and local-curve distribution matching over the evaluated baselines, with competitive point-set fidelity. Ablations validate the contributions of hierarchical decomposition, learned allocation, and full-scaffold conditioning. However, the fixed branch-saliency criterion and predefined generation capacities may limit representation of complex trees. Evaluation is restricted to nine species with species-identity conditioning, leaving generalization to unseen species and richer conditions unexplored. Future work will explore learned scaffold decomposition, adaptive capacities, and image- or point-cloud-based conditioning.

### AI USE STATEMENT

In this work, we used generative AI tools to assist with language polishing, code review, and debugging. All AI-assisted code modifications were reviewed and tested by the authors. The authors take full responsibility for the final content of this work.

### REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the proposed framework and its components in the main paper. Dataset preparation, implementation details, training and inference configurations, and evaluation protocols are documented in Appendices. To further support reproducibility, we provide code as supplementary materials.

## REFERENCES

Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In *International conference on machine learning*, pp. 40–49. PMLR, 2018.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

Minghao Chen, Roman Shapovalov, Iro Laina, Tom Monnier, Jianyuan Wang, David Novotny, and Andrea Vedaldi. Partgen: Part-level 3d generation and reconstruction with multi-view diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5881–5892. IEEE, 2025.

Lihe Ding, Shaocong Dong, Yaokun Li, Chenjian Gao, Xiao Chen, Rui Han, Yihao Kuang, Hong Zhang, Bo Huang, Zhanpeng Huang, et al. Fullpart: Generating each 3d part at full resolution. In *International Conference on Learning Representations*, volume 2026, pp. 45856–45876, 2026.

Shaocong Dong, Lihe Ding, Xiao Chen, Yaokun Li, Yuxin Wang, Yucheng Wang, Qi Wang, Jaehyeok Kim, Chenjian Gao, Zhanpeng Huang, et al. From one to more: Contextual part latents for 3d generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8230–8240. IEEE, 2025.

Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 605–613, 2017.

Jianwei Guo, Shibiao Xu, Dong-Ming Yan, Zhanglin Cheng, Marc Jaeger, and Xiaopeng Zhang. Realistic procedural plant modeling from multiple view images. *IEEE transactions on visualization and computer graphics*, 26(2):1372–1384, 2018.

Amir Hertz, Or Perel, Raja Giryes, Olga Sorkine-Hornung, and Daniel Cohen-Or. Spaghetti: Editing implicit shapes through part aware generation. *ACM Transactions on Graphics (TOG)*, 41(4):1–20, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Shaojun Hu, Zhengrong Li, Zhiyi Zhang, Dongjian He, and Michael Wimmer. Efficient tree modeling from airborne lidar point clouds. *Computers & Graphics*, 67:1–13, 2017.

R. Kenny Jones, Theresa Barton, Xianghao Xu, Kai Wang, Ellen Jiang, Paul Guerrero, Niloy Mitra, and Daniel Ritchie. Shapeassembly: Learning to generate programs for 3d shape structure synthesis. *ACM Transactions on Graphics (TOG), Siggraph Asia 2020*, 39(6):Article 234, 2020.

Juil Koo, Seungwoo Yoo, Minh Hieu Nguyen, and Minhyuk Sung. Salad: Part-level latent diffusion for 3d shape generation and manipulation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14395–14405. IEEE, 2023.

Jae Joong Lee, Bosheng Li, and Bedrich Benes. Latent l-systems: Transformer-based tree generator. *ACM Transactions on Graphics*, 43(1):1–16, 2023.

Bosheng Li, Jacek Kałużny, Jonathan Klein, Dominik L Michels, Wojtek Pałubicki, Bedrich Benes, and Sören Pirk. Learning to reconstruct botanical trees from single images. *ACM Transactions on Graphics (TOG)*, 40(6):1–15, 2021.

Songlin Li, Despoina Paschalidou, and Leonidas Guibas. Pasta: Controllable part-aware shape generation with autoregressive transformers. In *2026 International Conference on 3D Vision (3DV)*, pp. 1522–1533. IEEE, 2026.

Jiayi Liu, Denys Iliash, Angel Chang, Manolis Savva, and Ali Mahdavi Amiri. Singapo: Single image controlled generation of articulated parts in objects. In *International Conference on Learning Representations*, volume 2025, pp. 97511–97532, 2025a.

Yanchao Liu, Jianwei Guo, Bedrich Benes, Oliver Deussen, Xiaopeng Zhang, and Hui Huang. Treepartnet: Neural decomposition of point clouds for 3d tree reconstruction. *ACM Transactions on Graphics (Proceedings of SIGGRAPH ASIA)*, 40(6):232:1–232:16, 2021.

Zhihao Liu, Zhanglin Cheng, and Naoto Yokoya. Neural hierarchical decomposition for single image plant modeling. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 733–742. IEEE, 2025b.

Elias Ariel Marks, Lucas Nunes, Federico Magistri, Matteo Sodano, Rodrigo Marcuzzi, Lars Zimmermann, Jens Behley, and Cyrill Stachniss. Tree skeletonization from 3d point clouds by denoising diffusion. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27607–27608. IEEE, 2025.

Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas Guibas. Structurenet: Hierarchical graph networks for 3d shape generation. *ACM Transactions on Graphics (TOG), Siggraph Asia 2019*, 38(6):Article 242, 2019.

Lidong Mo, Constantin M Zohner, Peter B Reich, Jingjing Liang, Sergio De Miguel, Gert-Jan Nabuurs, Susanne S Renner, Johan Van Den Hoogen, Arnan Araza, Martin Herold, et al. Integrated global assessment of the natural forest carbon potential. *Nature*, 624(7990):92–101, 2023.

George Kiyohiro Nakayama, Mikaela Angelina Uy, Jiahui Huang, Shi-Min Hu, Ke Li, and Leonidas Guibas. Difffacto: Controllable part-based 3d point cloud generation with cross diffusion. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14211–14221. IEEE, 2023.

Matheus Henrique Nunes, Marcel Caritá Vaz, José Luís Campana Camargo, William F Laurance, Ana de Andrade, Alberto Vicentini, Susan Laurance, Pasi Raumonen, Toby Jackson, Gabriela Zuquim, et al. Edge effects on tree architecture exacerbate biomass loss of fragmented amazonian forests. *Nature Communications*, 14(1):8129, 2023.

Przemyslaw Prusinkiewicz and Aristid Lindenmayer. Graphical modeling using l-systems. In *The Algorithmic Beauty of Plants*, pp. 1–50. Springer, 1990.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Adam Runions. Modeling trees with a space colonization algorithm. *Eurographics Association*, 2007.

Ping Tan, Gang Zeng, Jingdong Wang, Sing Bing Kang, and Long Quan. Image-based tree modeling. In *ACM SIGGRAPH 2007 papers*, pp. 87–es. 2007.

Shengjun Tang, Zhuoyu Ao, Yaoyu Li, Hongsheng Huang, Linfu Xie, Ruisheng Wang, Weixi Wang, and Renzhong Guo. Treenet3d: A large scale tree benchmark for 3d tree modeling, carbon storage estimation and tree segmentation. *International Journal of Applied Earth Observation and Geoinformation*, 130:103903, 2024.

SS Vallender. Calculation of the wasserstein distance between probability distributions on the line. *Theory of Probability & Its Applications*, 18(4):784–786, 1974.

Hanxiao Wang, Biao Zhang, Jonathan Klein, Dominik L Michels, Dong-Ming Yan, and Peter Wonka. Autoregressive generation of static and growing trees. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–12, 2025.

Jason Weber and Joseph Penn. Creation and rendering of realistic trees. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 119–128, 1995.

Jiabo Xu, Bo Su, Jingbo Wei, Xiangyun Hu, Hengming Dai, and Tao Ke. Divinetree: All-in-one 3d tree modeling with diverse and fused visual guidance. *Information Fusion*, pp. 104013, 2025.

Jie Yang, Kaichun Mo, Yu-Kun Lai, Leonidas J Guibas, and Lin Gao. Dsg-net: Learning disentangled structure and geometry for 3d shape generation. *ACM Transactions on Graphics (TOG)*, 42 (1):1–17, 2022.

Qitong Yang, Mingtao Feng, Zijie Wu, Weisheng Dong, Fangfang Wu, Yaonan Wang, and Ajmal Mian. Hierarchical gaussian mixture model splatting for efficient and part controllable 3d generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11104–11114. IEEE, 2025.

Xiaochen Zhou, Bosheng Li, Bedrich Benes, Songlin Fei, and Sören Pirk. Deeptree: Modeling trees with situated latents. *IEEE Transactions on Visualization and Computer Graphics*, 30(8): 5795–5809, 2023.

Xiaochen Zhou, Bosheng Li, Bedrich Benes, Ayman Habib, Songlin Fei, Jinyuan Shao, and Sören Pirk. Treestructor: Forest reconstruction with neural ranking. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–19, 2025.

# A Appendix

## A.1 Qualitative Visualization

Figure 8 extends the qualitative comparison to all nine evaluated species. Figures 9–11 further present multiple generated samples for each species, showing their branch structures and corresponding foliage renderings. Together, these results illustrate the generation of diverse tree structures across and within species. Foliage is added only for visualization, but it is not predicted by ScaffoldTree and is excluded from all quantitative evaluations.

## A.2 Dataset Construction and Representation

**Species and data split.** Table 5 summarizes the fixed data split used for ScaffoldTree, all external baselines, and all ablation studies. The validation and test sets each contain 30 trees per species. Agarwood is excluded because many instances contain multiple level-0 stems, whereas our representation assumes a single root branch.

**Branch extraction and canonical ordering.** We use the branch representation and canonical traversal order defined in Section 3.1. Each branch polyline is resampled to $P = 16$ points at equal arc-length intervals. Coordinates and radii are stored in $\mathbf{X}_i \in \mathbb{R}^{P\times 4}$, while padding is represented by separate branch and point masks. All normalization statistics for curves and attachment arcs are estimated exclusively from the training split.

**Deterministic scaffold–residual decomposition.** For each branch $i$, we compute its length $L_i$, mean radius $R_i$, descendant count $D_i$, and the spatial extent $C_i$ of the complete subtree rooted at that branch. Each feature is min–max normalized within the current tree. The saliency score used in all experiments is

$$v_i = 0.15\,\widehat{L}_i + 0.30\,\widehat{R}_i + 0.30\,\widehat{\log(1+D_i)} + 0.25\,\widehat{C}_i. \tag{15}$$

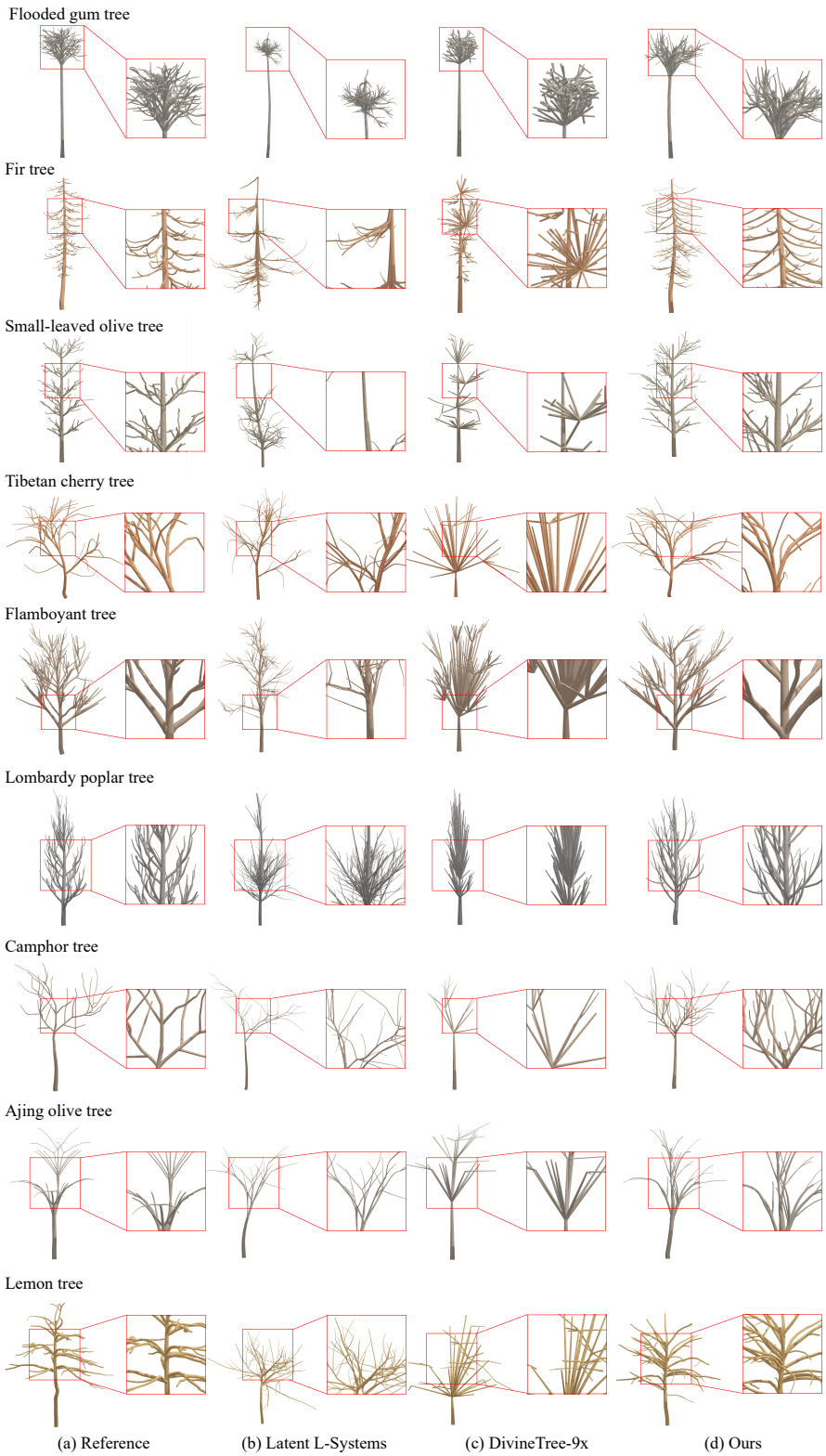



Figure 8: Qualitative comparison across all nine TreeNet3D species. Enlarged regions highlight local branching structures around major junctions.

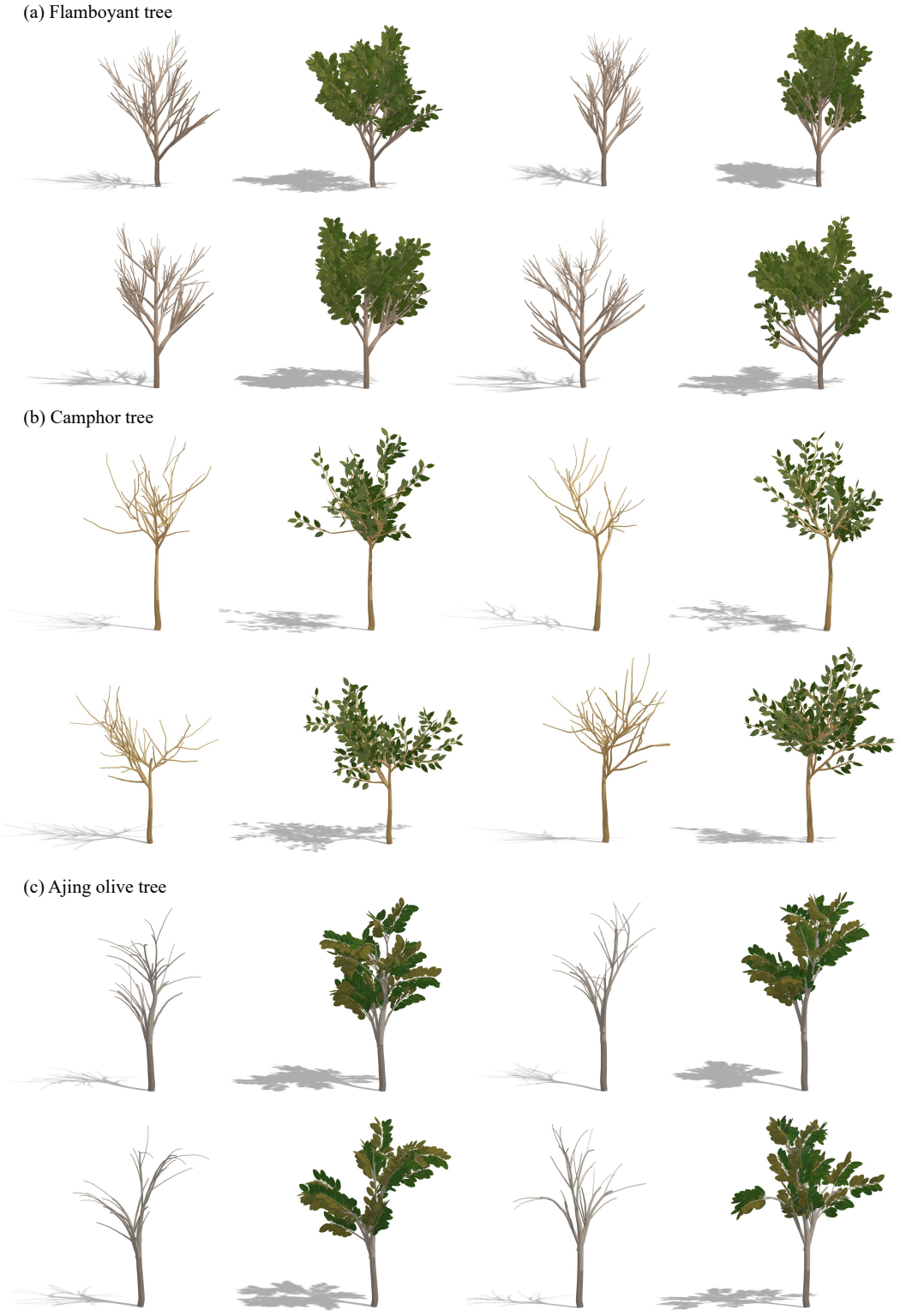



Figure 9: Generated branches and corresponding foliage renderings for Flamboyant, Camphor, and Ajing olive.

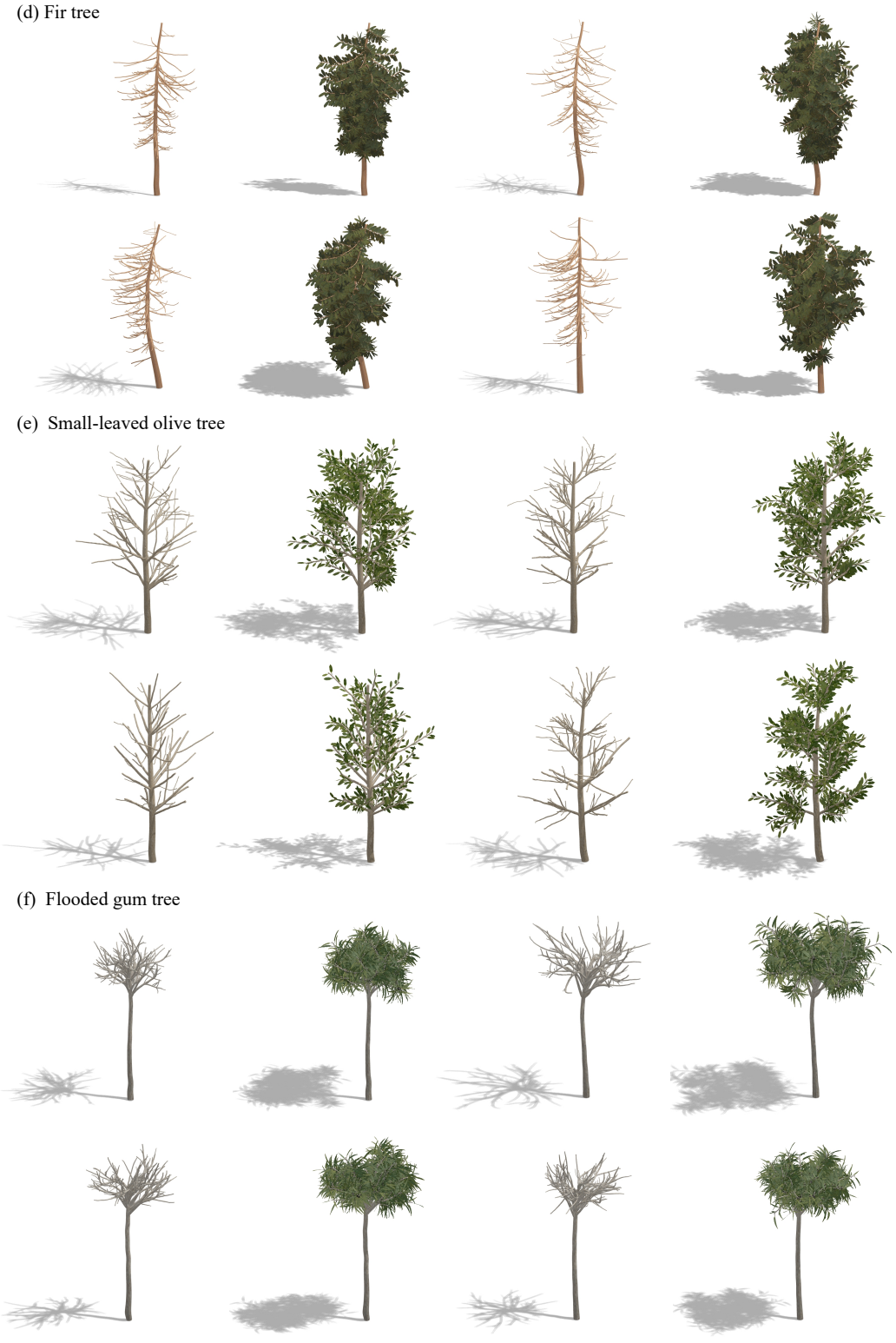
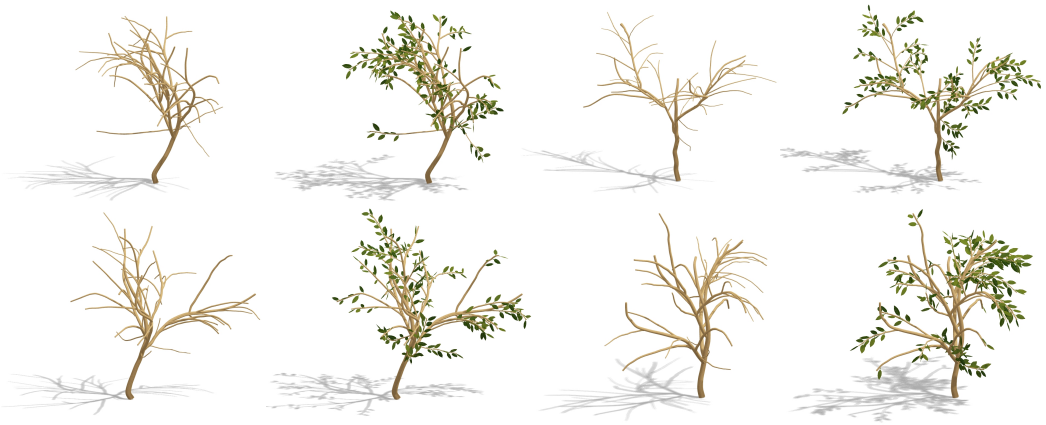
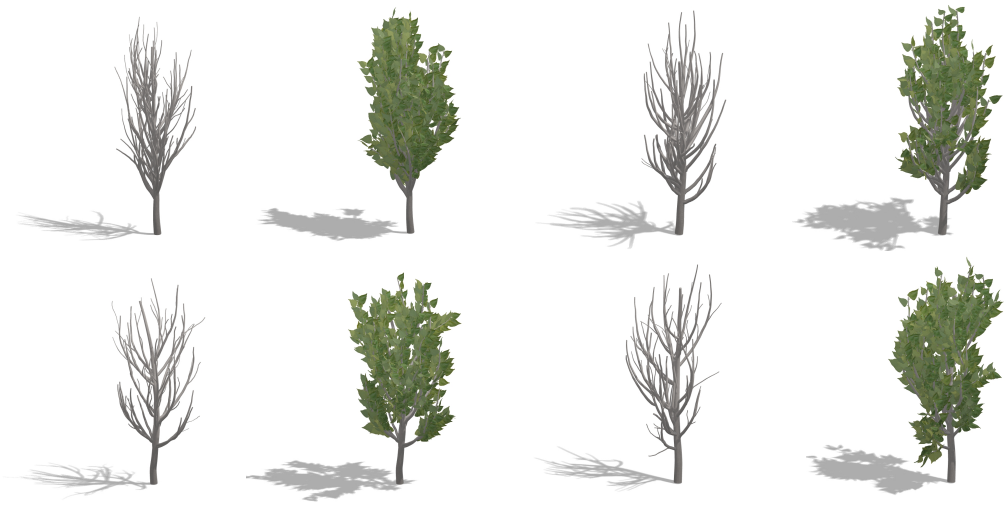



Figure 10: Generated branches and corresponding foliage renderings for Fir, Small-leaved olive, and Flooded gum.

(g) Tibetan cherry tree



(h) Lombardy poplar tree



(i) Lemon tree

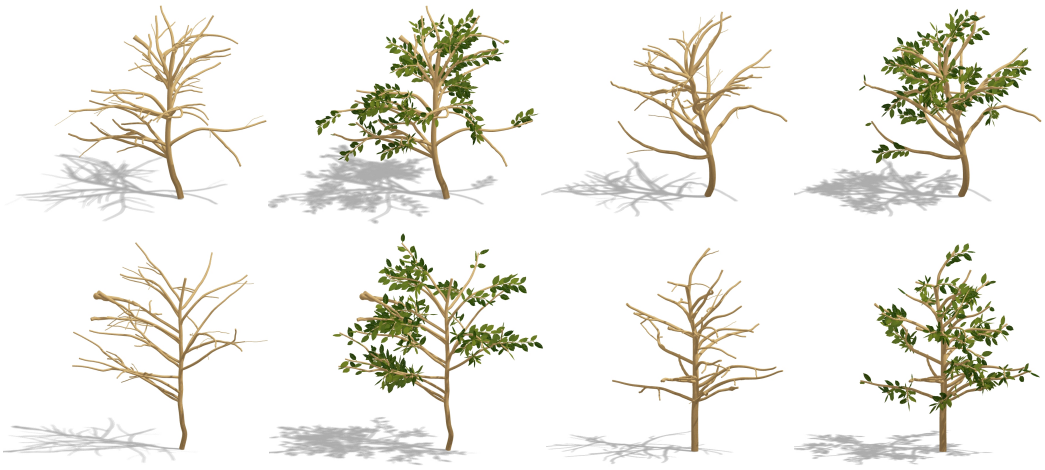


Figure 11: Generated branches and corresponding foliage renderings for Tibetan cherry, Lombardy poplar, and Lemon.

Table 5: TreeNet3D split used in all experiments.

| Species | Label | Train | Val. | Test |
|---|---|---|---|---|
| Flamboyant | 0 | 1,340 | 30 | 30 |
| Camphor | 1 | 1,241 | 30 | 30 |
| Ajing olive | 2 | 1,281 | 30 | 30 |
| Fir | 3 | 1,290 | 30 | 30 |
| Small-leaved olive | 4 | 1,289 | 30 | 30 |
| Flooded gum | 5 | 1,336 | 30 | 30 |
| Tibetan cherry | 6 | 1,340 | 30 | 30 |
| Lombardy poplar | 7 | 1,290 | 30 | 30 |
| Lemon | 8 | 1,440 | 30 | 30 |
| Total | – | 11,847 | 270 | 270 |

The root receives a score larger than every non-root branch and is therefore always retained. The initial semantic quota is

$$N_G^{(0)} = \min(N, G, 48, \max(8, \mathrm{round}(0.30N))) , \tag{16}$$

where $N$ is the full branch count and $G = 64$ is the scaffold tensor capacity. Candidate branches are processed in descending saliency order. A candidate is retained only if it can be added together with all missing ancestors without exceeding the quota, yielding an ancestor-closed scaffold.

Each omitted branch is assigned to its nearest retained ancestor. An anchor's omitted branches may have multiple roots directly attached to the scaffold; we call this group a residual subtree because it becomes connected through the anchor, although the residual branches alone may form a forest. When a group exceeds $K = 32$ branches, the highest-saliency residual path feasible under scaffold capacity $G = 64$ is promoted. Assignment and promotion repeat until every group satisfies $K$; samples that cannot meet these constraints are excluded during preprocessing rather than truncated. The shared full-tree capacity $B_{\max} = 192$ is a hard tensor limit for preprocessed trees, assembled outputs, and DirectTreeDiffusion, not an additional residual allowance beyond the scaffold.

### A.3 Implementation Details

**Species-name conditioning.** The conditioning signal is a fixed, unexpanded English name-only prompt. The exact prompts stored in the training cache are *flamboyant tree*, *camphor tree*, *Ajing olive tree*, *fir tree*, *small-leaved tree*, *flooded gum tree*, *Tibetan cherry tree*, *Lombardy poplar tree*, and *lemon tree*. Thus, the dataset display label *Small-leaved olive* is mapped to the fixed prompt *small-leaved tree*. We encode each prompt once with the frozen text tower of CLIP ViT-B/32 and $\ell_2$-normalize the resulting 512-dimensional feature. Each module applies its own layer normalization and linear projection before adding the condition to its tokens. CLIP parameters remain frozen. During training, the condition is dropped with probability 0.1. For the two diffusion models this learns the unconditional branch used for classifier-free guidance; for the Allocator it acts as conditioning regularization. The present experiments therefore consider a closed set of nine species and do not assume zero-shot generalization to unseen species.

**Global Scaffold Diffusion.** Global Scaffold Diffusion represents within-branch geometry with point tokens while maintaining attention over all $G = 64$ branch slots. Branch tokens encode geometry, attachment, the fixed active-branch mask, and the current parent state. A global tree token provides scaffold-level morphological context, while separate heads predict scaffold count and branch level. The final architecture is summarized in Table 6.

Global and Local Residual Diffusion use the chord-residual representation defined in Eq. 7. The parameterization is applied to the XYZ channels, while radius retains its normalized pointwise representation. All 16 points are used during training and inference, without curve fitting or control-point reduction.

To mitigate the train–test discrepancy induced by discrete parent prediction, the world-coordinate geometry objectives progressively replace a fraction of ground-truth parent transformations with differentiable soft-parent reconstructions. The maximum training fraction is 0.35, the softmax temperature is 0.75, and the fraction is ramped from epoch 6 over 15 epochs. Beginning at epoch 8, a

Table 6: Global Scaffold Diffusion architecture and diffusion configuration.

| Configuration | Value |
|---|---|
| Hidden dimension / attention heads | 512 / 8 |
| Point encoder depth | 2 |
| Fine / coarse / bottleneck depth | 3 / 3 / 3 |
| Geometry–topology coupling depth | 2 |
| Branch pooling | none |
| Global tree token | enabled |
| Maximum branches / points per branch | 64 / 16 |
| Diffusion training steps | 1,000 |
| Noise schedule | linear, $10^{-4} \rightarrow 0.02$ |
| Minimum-SNR parameter | 5.0 |
| EMA decay | 0.999 |

four-step deterministic diffusion rollout is sampled with probability 0.25; the final rollout state receives the world-space physical objective with weight 0.25. Rollout start times are sampled between diffusion timesteps 50 and 250.

**Structure-Conditioned Allocator.** The Allocator uses a scaffold point encoder and four branch-transformer blocks with hidden dimension 512, eight attention heads, point depth one, and dropout 0.1. For each scaffold branch, it predicts an occupancy logit and a positive-count distribution over $\{1, \ldots, K\}$. A separate 193-way categorical head predicts the residual total. At inference, its support is restricted according to Eq. 8, and the branch-wise counts are decoded using the exact-sum objective in Eq. 12. For a real training tree with $N \leq B_{\max}$ branches, the target is $N_R = N - N_G$. For a generated scaffold used in context-mixed training, the paired target is restricted to $[0, B_{\max} - N_G]$. At inference, the same remaining-capacity bound is passed to the exact-sum decoder, and assembly additionally checks that the output never exceeds $B_{\max}$ branches. The soft overload term discourages an individual anchor from absorbing an implausibly large fraction of the residual structure without imposing hard clipping on its count. We use $\lambda_{\text{over}} = 2.0$ and per-species caps $(\kappa_0, \ldots, \kappa_8) = (10, 11, 4, 19, 6, 7, 6, 5, 13)$ in the label order of Table 5. These values correspond to the 90th percentile of the per-species maximum anchor load, estimated exclusively from the training targets. The network weights are those of the context-mixed Allocator; the overload term is a deterministic decoder-only policy.

**Local Residual Diffusion.** Local residual branches use the parent-relative coordinate frame defined in Eq. 14. Each local patch encodes the complete scaffold once; residual tokens first interact through self-attention and then attend to all scaffold tokens through cross-attention. Parent class 0 denotes the scaffold anchor, while the remaining classes denote preceding residual branches. Local curves use the chord-residual representation in Eq. 7. Table 7 lists the detailed configuration.

Table 7: Local Residual Diffusion architecture and diffusion configuration.

| Configuration | Value |
|---|---|
| Hidden dimension / attention heads | 512 / 8 |
| Scaffold point / context depth | 1 / 3 |
| Residual point / branch depth | 2 / 6 |
| Dropout | 0.1 |
| Maximum residual branches / points | 32 / 16 |
| Diffusion training steps | 1,000 |
| Noise schedule | linear, $10^{-4} \rightarrow 0.02$ |
| Fixed allocated count | enabled |
| Full-scaffold cross-attention | enabled |

After denoising, residual branches are transformed from their parent-relative frames into world coordinates in parent-before-child order. Local parent indices are then remapped to global indices before merging the patch with the scaffold, producing a final tree with a single explicit parent array and no implicit patch boundaries.

Table 8: Principal loss weights used by the two diffusion models. A dash denotes an inapplicable term.

| Term | Global | Local | Term | Global | Local |
|---|---|---|---|---|---|
| Curve | 4.00 | 4.00 | Attachment | 2.00 | 2.00 |
| Existence | 1.00 | – | Parent | 0.75 | 0.75 |
| Count | 1.00 | – | Level / depth | 0.50 / 0.25 | – |
| Tangent | 1.00 | 1.00 | Segment | 0.25 | 0.25 |
| Curvature | 1.00 | 1.25 | Excess curvature | 1.00 | 0.75 |
| Length | 0.25 | 0.25 | Connection | 0.50 | 0.50 |
| Tortuosity | 0.50 | – | Jerk | 0.50 | – |
| Root geometry | 4.00 | – | World curve | 0.25 | – |
| Root / branch reversal | 1.00 / 1.00 | – | Self / cross collision | 0.10 / 0.20 | – |
| Sibling dispersion | – | 0.25 | Assembled-tree collision | – | 0.25 |
| Below ground | 0.05 | – | Rollout objective | 0.25 | – |

### A.4 Objectives and Optimization

**Hybrid diffusion objective.** The continuous state contains normalized curve residuals and attachment arcs and follows the Gaussian forward process. In the final Global model, a species-conditioned head predicts the scaffold count; this count fixes the active-branch mask, and all active branches use the fixed $P = 16$ point grid. Parent labels are the discrete diffusion state and follow the absorbing-state corruption process. Branch existence and level are retained as auxiliary supervision, with level predicted by a separate head. In Local Residual Diffusion, the Allocator fixes the branch count and the same $P = 16$ grid fixes the point mask, so only parent labels undergo discrete corruption; local existence and count losses are disabled. Parent classes are masked to the root/anchor symbol and valid preceding branches.

Table 8 reports the principal nonzero loss weights used for the final checkpoints. Auxiliary geometric penalties are evaluated on denoised reconstructions in physical coordinates and act only as training regularizers; they are not used for sample rejection during inference.

The local sibling term penalizes sibling pairs that attach too closely and depart in nearly parallel directions. The assembled-tree collision term transforms a predicted residual patch into world coordinates and penalizes non-adjacent residual–residual and residual–scaffold pairs within distance 0.012. Together, these terms regularize local geometry and reduce spatial conflicts with nearby scaffold and residual branches.

**Allocator objective.** For real scaffold contexts, the Allocator objective is

$$\mathcal{L}_A = \mathcal{L}_{\text{total}} + \mathcal{L}_{\text{occ}} + \mathcal{L}_{\text{count}} + 0.25\mathcal{L}_{\text{sum}} + 0.05\mathcal{L}_{\text{ordinal}} + \mathcal{L}_{\text{CDF}}. \tag{17}$$

Here, $\mathcal{L}_{\text{sum}}$ aligns the sum of expected anchor counts with the target residual total. Generated scaffold contexts retain the total count of the paired real tree, but exact anchor correspondences are unavailable; consequently, anchor-level occupancy and count supervision are masked, while total-count and expected-sum supervision are retained.

**Context-mixed optimization.** Although the three modules are optimized with separate objectives, downstream components are not trained exclusively with oracle contexts. The Allocator samples ground-truth and generated scaffold contexts with equal probability. Local Residual Diffusion similarly mixes real contexts and cached contexts produced by the final Global Scaffold Diffusion and Allocator at a 0.5/0.5 ratio. For each active anchor in a generated context, its supervised residual target is drawn from a real patch of the same species with the exact allocated count. If that count is unavailable, we use the smallest available same-species patch with a larger count and retain its first $n_i$ branches in canonical depth-first order. The parent-relative representation makes these local targets transferable to generated anchors. This context-mixed strategy mitigates exposure bias by presenting downstream modules with the geometric, topological, and cardinality discrepancies encountered at inference time.

The reported Global checkpoint is the result of the 40-epoch fixed-$P = 16$ fine-tuning stage initialized from the EMA weights of the preceding structural checkpoint. The reported Allocator is

Table 9: Optimization settings of the final fine-tuning stages.

| Module | Epochs | Batch | LR | Weight decay |
|---|---|---|---|---|
| Global Scaffold Diffusion | 40 | 8 | $5\times10^{-6}$ | $10^{-4}$ |
| Allocator | 30 | 8 | $2\times10^{-5}$ | $10^{-3}$ |
| Local Residual Diffusion | 30 | 8 | $2\times10^{-5}$ | $10^{-3}$ |

a context-mixed fine-tuning of the real-context Allocator, and the reported Local model is a fixed-$P = 16$ context-mixed physical fine-tuning of the fixed-count Local checkpoint. Consequently, Table 9 reports the final fine-tuning stages rather than training from random initialization.

All modules use AdamW, bfloat16 precision, and cosine learning-rate decay. Global Scaffold Diffusion uses three warmup epochs and gradient clipping at 0.25. The Allocator and Local Residual Diffusion use two warmup epochs, a minimum learning rate of $10^{-6}$, and gradient clipping at 1.0. Their early-stopping patience values are 10 and 5 epochs, respectively. Model selection uses validation data only. Random seeds are 42 for training and 1234 for validation.

**Inference settings.** At inference, Global Scaffold Diffusion and Local Residual Diffusion use 100 and 50 denoising steps, respectively, with a classifier-free guidance scale of 2.0 for both models. We use the EMA checkpoint for Global Scaffold Diffusion and the deterministic exact-sum Allocator with the species-aware soft-cap cost.

### A.5 Unified Evaluation Protocol

**Alignment and centerline sampling.** All methods are converted to the same explicit branch-centerline and parent representation under a shared preprocessed coordinate convention. During evaluation we translate the start point of the unique root to the origin, without additional instance-specific rotation or scale alignment. Since Latent L-Systems does not predict radii, cross-method evaluation is restricted to centerline geometry and topology.

For point-set metrics, we concatenate all nondegenerate line segments and place sampling targets uniformly over their cumulative arc length. Thus, longer segments receive proportionally more samples. The main results use 128 points per tree.

**Point-set distribution metrics.** Let $P_a$ and $P_b$ be the sampled centerline point sets of trees $a$ and $b$. Their symmetric squared Chamfer distance is

$$d_{\mathrm{CD}}(a,b) = \frac{1}{|P_a|}\sum_{\mathbf{x}\in P_a}\min_{\mathbf{y}\in P_b}\|\mathbf{x}-\mathbf{y}\|_2^2 + \frac{1}{|P_b|}\sum_{\mathbf{y}\in P_b}\min_{\mathbf{x}\in P_a}\|\mathbf{y}-\mathbf{x}\|_2^2. \tag{18}$$

For generated set $\mathcal{G}$ and reference set $\mathcal{R}$,

$$\mathrm{Fidelity} = \frac{1}{|\mathcal{G}|}\sum_{g\in\mathcal{G}}\min_{r\in\mathcal{R}} d_{\mathrm{CD}}(g,r), \tag{19}$$

$$\mathrm{MMD} = \frac{1}{|\mathcal{R}|}\sum_{r\in\mathcal{R}}\min_{g\in\mathcal{G}} d_{\mathrm{CD}}(r,g), \tag{20}$$

$$\mathrm{Coverage} = \frac{|\{\arg\min_{r\in\mathcal{R}} d_{\mathrm{CD}}(g,r) : g\in\mathcal{G}\}|}{|\mathcal{R}|}. \tag{21}$$

Fidelity measures the proximity of generated samples to the reference distribution, whereas MMD measures how well reference samples are represented by the generated set. Coverage counts the number of distinct reference samples selected as nearest neighbors. Generated and reference diversity are the mean pairwise Chamfer distances within their respective sets; their ratio is preferred near one.

**Structural and curve distributions.** For a scalar statistic $z$, we compute the empirical one-dimensional Wasserstein distance and normalize it by

$$\sigma_z = \max\big(\mathrm{std}(z_{\mathcal{R}}), 0.05|\,\mathrm{mean}(z_{\mathcal{R}})|, 10^{-3}\big)\,. \tag{22}$$

The structural statistics comprise branch count, maximum depth, terminal ratio, tree height, crown width, total branch length, node degree, branch length, attachment arc, branch angle, tortuosity, and local turning angle. Struct. W1 is the mean of their 12 normalized distances. Because maximum depth has negligible reference variation, Struct. W1-ND averages the remaining 11 components; the complete score and depth component are reported in Appendix A.7. Curve W1 is the mean normalized W1 over branch length, branch angle, tortuosity, and local turning angle.

All metrics are computed independently for each species and subsequently macro-averaged, assigning equal weight to each species.

**Collisions and validity.** Each 16-point branch is densified with three samples per segment. A self-collision is registered when two points on the same branch are within 0.012 while separated along the branch by at least the larger of 0.048 and 20% of its arc length. For cross-branch collisions, direct parent–child pairs are excluded and the first two original points of every non-root branch are ignored to avoid treating a valid junction as a collision. The reported cross-branch collision rate is the fraction of eligible non-adjacent branch pairs with minimum sampled distance below 0.012.

A generated tree is topology-valid if it has at least one branch, exactly one root, no invalid or self-parent index, and no directed cycle. Overall validity additionally requires finite coordinates and at least two valid points on every active branch.

### A.6 Baseline and Ablation Details

**Latent L-Systems.** We retrain the released three-layer encoder–decoder Transformer separately for every species. The branch hierarchy is converted to quantized hierarchical L-string rules. The model uses width 128, four attention heads, feed-forward width 128, dropout 0.1, and maximum sequence length 512. Each species model is trained for at most 100 epochs with batch size 512, learning rate $10^{-5}$, gradient clipping at 1.0, and early-stopping patience 20. The parameter count depends on the species vocabulary (3.23M for the Flamboyant model). Generated L-strings are converted back to explicit branch centerlines. Missing radius does not affect the common evaluation.

**DivineTree-9x.** The released DivineTree architecture has no nine-class condition in the configuration used here. We therefore train one unconditional model per species. Each contains 16 Transformer layers of width 768 with 16 heads and maximum length 192 (175.86M parameters per model). Training uses at most 300 epochs, batch size 16, learning rate $10^{-4}$ decayed to $10^{-5}$, bfloat16 precision, gradient clipping at 1.0, and early-stopping patience 30. Branch counts are sampled from the corresponding species' training distribution.

**DirectTreeDiffusion.** This controlled baseline uses the same 512-dimensional hybrid continuous–discrete diffusion backbone, species-name condition, chord-residual parameterization, and 16-point branch representation as Global Scaffold Diffusion, but generates the complete tree directly in a single 192-branch tensor. This design isolates the contribution of scaffold-to-subtree factorization from that of the representation and denoising backbone.

**Controlled hierarchical ablations.** Uniform allocation retains the Full model's generated scaffold and predicted residual total but distributes the total as evenly as possible over active scaffold anchors. Anchor-only context retains the species, allocated count, and anchor geometry but removes all other scaffold tokens from Local Residual Diffusion. Full, Uniform, and Anchor-only share the same seeds, generated scaffolds, and residual totals for all 1,800 samples.

### A.7 Additional Experimental Results

**Point-sampling sensitivity.** Table 10 recomputes the point-set metrics using 128, 256, and 512 arc-length samples while keeping the generated trees fixed. The relative comparison is stable across sampling densities: DivineTree-9x attains the lowest MMD, whereas ScaffoldTree achieves the highest coverage at all three resolutions.

**Depth-inclusive structural score.** Table 11 reports both the complete structural aggregate and the depth-excluded value used in the main paper. DivineTree-9x's complete score is dominated by

Table 10: Sensitivity to the number of sampled centerline points.

| Points | Method | Fidelity ↓ | MMD ↓ | Coverage ↑ |
|---|---|---|---|---|
| 128 | Latent L-Systems | 0.2743 | 0.0161 | 0.6519 |
| | DivineTree-9x | **0.0087** | **0.0073** | 0.7481 |
| | DirectTreeDiffusion | 0.0476 | 0.0163 | 0.8000 |
| | ScaffoldTree | 0.0162 | 0.0101 | **0.8148** |
| 256 | Latent L-Systems | 0.2613 | 0.0139 | 0.6704 |
| | DivineTree-9x | **0.0073** | **0.0059** | 0.7407 |
| | DirectTreeDiffusion | 0.0439 | 0.0142 | 0.7704 |
| | ScaffoldTree | 0.0143 | 0.0086 | **0.8185** |
| 512 | Latent L-Systems | 0.2544 | 0.0130 | 0.6630 |
| | DivineTree-9x | **0.0067** | **0.0053** | 0.7593 |
| | DirectTreeDiffusion | 0.0419 | 0.0131 | 0.7852 |
| | ScaffoldTree | 0.0133 | 0.0079 | **0.8000** |

Table 11: Depth-inclusive and depth-excluded structural scores. Struct. W1-ND is used in the main paper; the depth component is reported separately for transparency.

| Method | Struct. W1 | Struct. W1-ND | Depth W1 |
|---|---|---|---|
| Latent L-Systems | 0.9994 | 1.0301 | **0.6611** |
| DivineTree-9x | 4.3710 | 1.0204 | 41.2278 |
| ScaffoldTree | **0.8402** | **0.7095** | 2.2778 |

Table 12: Per-species test results at 128 centerline points. Species labels follow Table 5; Coll. is the non-adjacent branch-pair collision percentage.

| Species | Method | MMD ↓ | Cov. ↑ | Struct. ↓ | Curve ↓ | Coll. (%) ↓ |
|---|---|---|---|---|---|---|
| 0 | ScaffoldTree | 0.0117 | 0.9000 | 0.4663 | 0.4777 | 0.876 |
| | Latent L-Systems | 0.0140 | 0.8667 | 0.7590 | 0.5554 | 1.053 |
| | DivineTree-9x | 0.0083 | 0.9000 | 1.3686 | 1.4795 | 3.823 |
| 1 | ScaffoldTree | 0.0118 | 0.9000 | 0.5494 | 0.3989 | 0.930 |
| | Latent L-Systems | 0.0173 | 0.6000 | 1.1985 | 0.4841 | 2.252 |
| | DivineTree-9x | 0.0106 | 0.9000 | 0.7365 | 0.7276 | 2.705 |
| 2 | ScaffoldTree | 0.0102 | 0.7333 | 0.6509 | 0.3960 | 1.474 |
| | Latent L-Systems | 0.0197 | 0.8333 | 0.9760 | 0.2785 | 1.046 |
| | DivineTree-9x | 0.0062 | 0.9000 | 0.6051 | 0.4971 | 2.660 |
| 3 | ScaffoldTree | 0.0047 | 0.6667 | 0.9480 | 0.3017 | 0.589 |
| | Latent L-Systems | 0.0244 | 0.4333 | 1.6726 | 0.3251 | 2.258 |
| | DivineTree-9x | 0.0037 | 0.7667 | 0.9359 | 0.8873 | 2.012 |
| 4 | ScaffoldTree | 0.0056 | 0.9000 | 0.5627 | 0.3421 | 0.663 |
| | Latent L-Systems | 0.0077 | 0.6000 | 0.6690 | 0.3207 | 1.027 |
| | DivineTree-9x | 0.0045 | 0.7000 | 0.9692 | 1.0781 | 2.302 |
| 5 | ScaffoldTree | 0.0038 | 0.4000 | 0.9495 | 0.4026 | 1.024 |
| | Latent L-Systems | 0.0071 | 0.4333 | 1.3652 | 0.5804 | 4.205 |
| | DivineTree-9x | 0.0021 | 0.7667 | 1.1551 | 1.4150 | 3.441 |
| 6 | ScaffoldTree | 0.0222 | 0.9667 | 0.5351 | 0.5193 | 0.454 |
| | Latent L-Systems | 0.0296 | 0.8000 | 0.4677 | 0.4524 | 0.734 |
| | DivineTree-9x | 0.0162 | 0.7000 | 1.1681 | 1.3157 | 0.961 |
| 7 | ScaffoldTree | 0.0046 | 0.9333 | 0.8291 | 0.4432 | 0.910 |
| | Latent L-Systems | 0.0055 | 0.7667 | 1.3368 | 0.3631 | 1.994 |
| | DivineTree-9x | 0.0040 | 0.4667 | 1.4307 | 1.3013 | 10.200 |
| 8 | ScaffoldTree | 0.0162 | 0.9333 | 0.8941 | 0.2828 | 0.465 |
| | Latent L-Systems | 0.0193 | 0.5333 | 0.8264 | 0.3629 | 0.833 |
| | DivineTree-9x | 0.0098 | 0.6333 | 0.8144 | 0.8856 | 1.422 |

maximum depth because it generates an average depth of 6.12 while the reference average is 2.00. Excluding depth prevents this near-constant reference statistic from dominating the aggregate score,

Table 13: Additional macro-averaged diagnostics. Diversity ratio is optimal near one.

| **Method** | **Fidelity** $\downarrow$ | **Diversity** $\rightarrow 1$ | **Valid** $\uparrow$ | **Topology** $\uparrow$ |
|---|---|---|---|---|
| ScaffoldTree | 0.0162 | 2.1650 | 1.0000 | 1.0000 |
| Latent L-Systems | 0.2743 | 51.3252 | 0.9994 | 0.9994 |
| DivineTree-9x | **0.0087** | **0.9240** | 1.0000 | 1.0000 |

Table 14: Paired species-level bootstrap comparisons. Values are mean improvement of Full over the ablated method with 95% confidence intervals.

| **Comparison** | **Metric** | **Mean improvement** | **95% CI** |
|---|---|---|---|
| Full vs. DirectTreeDiffusion | MMD | 0.00626 | [0.00240, 0.01078] |
| | Struct. W1-ND | 1.12404 | [0.85138, 1.39630] |
| | Curve W1 | 0.36294 | [0.22952, 0.49880] |
| Full vs. Uniform allocation | MMD | 0.00098 | [0.00035, 0.00178] |
| | Struct. W1-ND | 0.21021 | [0.10374, 0.34049] |
| | Curve W1 | −0.01132 | [−0.03992, 0.01393] |
| Full vs. Anchor-only | MMD | 0.00034 | [0.00007, 0.00064] |
| | Struct. W1-ND | 0.03429 | [−0.01582, 0.08970] |
| | Curve W1 | 0.00127 | [−0.03583, 0.04240] |
| | Junction-angle W1 | 0.11079 | [0.03255, 0.20602] |

while preserving the relative comparison between DivineTree-9x and ScaffoldTree on the remaining structural statistics.

**Per-species results.** Table 12 reports the per-species results underlying the macro-averaged comparison in the main paper. Performance varies across species, and no single method dominates every metric. The per-species breakdown therefore complements the aggregate results by revealing cross-species variation that is not captured by the macro-average alone.

**Validity and diversity diagnostics.** All methods achieve near-perfect serialized-tree validity. The substantially inflated diversity ratio of Latent L-Systems coincides with its weaker Fidelity and should not be interpreted as improved coverage of the reference distribution. DivineTree-9x yields the diversity ratio closest to one, whereas ScaffoldTree attains the highest coverage in the main comparison.

**Paired ablation uncertainty.** We estimate uncertainty by bootstrapping the nine species-level paired differences 10,000 times; positive values indicate lower error for the Full model. As shown in Table 14, the confidence intervals exclude zero for all reported decomposition metrics and for the MMD and Struct. W1-ND improvements of learned allocation. Full-scaffold conditioning yields a reliable improvement in junction-angle W1, whereas the confidence intervals for aggregate Struct. W1-ND and Curve W1 include zero. The Full-model values in this analysis are computed from the paired ablation pool and therefore need not coincide with those obtained from the independently generated samples used in the external-method comparison.