This MICCAI paper is the Open Access version, provided by the MICCAI Society. It is identical to the accepted version, except for the format and this watermark; the final published version is available on SpringerLink.

Anatomically Accurate 3D Vessel Generation via 3-Phase Chebyshev Curve Diffusion

Jihwan Mo[1], Sangbaek Yoo[1], Jaesoon Choi[2], and Dong Eui Chang[1]

[1] School of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea
dechang@kaist.ac.kr
[2] Asan Medical Center, Seoul, Republic of Korea

Abstract. Large-scale collection of 3D vascular data is inherently difficult, resulting in scarce publicly available datasets. Existing methods represent vessel edges as discrete point sequences, which cannot guarantee smoothness and lose millimeter-scale size information through normalization. We propose 3-Phase Chebyshev Curve Diffusion, a framework that generates vessel curves directly in a continuous polynomial coefficient space, an approach not previously explored for vascular generation. Chebyshev coefficients yield a fixed-dimensional, length-independent representation that inherently ensures C^∞ smoothness without post-processing. We further design auxiliary modules that recover mm-scale size lost due to normalization and a mechanism that preserves overall curve trajectories. On the ImageCAS coronary artery dataset, we achieve superiority over two existing methods across key anatomical metrics, including up to a 5.2× reduction in curvature Wasserstein distance, while confirming mm-scale accuracy.

Keywords: Vascular 3D model · Generative modeling · Neural Networks

1 Introduction

Acquiring 3D vascular data requires direct patient imaging, and publicly available datasets remain scarce [1]. As coronary artery disease is among the leading causes of death worldwide [2], large quantities of anatomically accurate synthetic vessels are required for diverse medical applications such as vessel segmentation training [1] and endovascular robot simulation [3]. This has driven growing interest in generative modeling of vascular structures. Coronary arteries in particular exhibit substantial inter-patient variability in the number of bifurcations and edge structures, with each edge following a tortuous path of varying curvature. Moreover, vessel radii progressively decrease from proximal to distal segments [4], a tapering pattern that reflects hemodynamic optimality [5]. A generative model must therefore simultaneously ensure accurate mm-scale dimensions, smooth curves, and anatomically valid radius profiles.

Skeleton-based representations naturally separate branching topology from per-edge geometric properties and have become the mainstream approach for vessel generation [6–9]. Recent approaches, PartVessel [10] and VesselGPT [8], still represent centerlines as discrete point sequences, failing to guarantee smooth curves and producing anatomically implausible curvature profiles. Furthermore, normalization loses mm-scale size information. A fundamental shift to representing curves as continuous functions, together with a mechanism to recover physical dimensions, is therefore required.

In this paper, we generate vessel curves via diffusion in Chebyshev polynomial coefficient space, simultaneously achieving minimax-optimal approximation [11] and C^∞ smoothness. Magnitude, direction, and radius are then generated via independent diffusion models through polar decomposition. Our contributions are: (1) We propose 3-Phase Curve Diffusion based on polar decomposition of Chebyshev coefficients, generating C^∞-smooth curves in a fixed-dimensional space. (2) We design auxiliary mechanisms for mm-scale recovery, curve trajectory preservation, and per-edge diversity control via temperature calibration. (3) On ImageCAS coronary arteries, we achieve superiority over two existing methods across key anatomical metrics and quantitatively evaluate Murray's law compliance.

2 Related Work

Vessel Generation. Rule-based methods (VascuSynth [12], Schneider et al. [13]) generate vascular trees from physical laws but lack morphological diversity. Among data-driven methods, Wolterink et al. [6] and VesselVAE [7] generate coronary arteries using generative adversarial networks and recursive variational autoencoders (RVAEs), respectively, though with limited topological complexity. TrIND [14] diffuses implicit neural fields for representation but not generation, and Prabhakar et al. [9] use graph diffusion [15, 16] but connect nodes with straight lines. PartVessel [10] proposes a hierarchical part-based framework, yet smoothness limitations of point-sequence–based generation and scale loss persist. VesselGPT [8] autoregressively generates cerebral vasculature with B-spline [17] cross-sections but retains discrete centerline sequences.

Diffusion Models and Curve Representations. Denoising diffusion probabilistic models (DDPMs) [18–21] have demonstrated strong performance in image [22] and 3D point cloud [23, 24] generation. However, point-cloud–based approaches lack inter-point continuity and are unsuitable for smooth curve generation. For curve representation, B-splines [17] depend on knot placement and Fourier bases suffer from Gibbs phenomena at open endpoints due to periodicity [11]. In contrast, the Chebyshev polynomials T_k on $[-1, 1]$, defined via $T_k(\cos\theta) = \cos(k\theta)$, form a cosine-like orthogonal basis whose truncated expansion achieves near-minimax approximation [11]: a fixed-degree representation attains rapid convergence for smooth functions, is independent of curve length, and is free of the Gibbs artifacts that affect periodic bases on open intervals. We leverage these properties to perform diffusion directly in the Chebyshev coefficient space.

3 Method

3.1 Overview

Our approach builds on the framework of PartVessel [10]. A vascular tree is represented as a key graph, whose nodes correspond to bifurcations and endpoints and whose edges represent vessel segments connecting them. The attribute vector of each node, $\mathbf{v} = [\mathbf{p};\, r, \ell, d, \kappa;\, \mathbf{n}] \in \mathbb{R}^{10}$, consists of the position $\mathbf{p} \in \mathbb{R}^3$, geometric descriptors of the associated edge (mean radius r, arc length ℓ, chord length d, curvature κ), and a direction vector $\mathbf{n} \in \mathbb{R}^3$; all attributes are computed after normalizing each tree.

We briefly describe the PartVessel [10] framework. In Stage 1, an RVAE generates the key graph. The encoder aggregates node features $\mathbf{v}$ from leaves to root, producing hidden states $\mathbf{h} \in \mathbb{R}^{d_z}$ at each node via MLPs; a latent $\mathbf{z} \in \mathbb{R}^{d_z}$ is sampled from the root state $\mathbf{h}_{\text{root}}$ (at inference, $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$). The decoder reconstructs $\hat{\mathbf{h}}$ top-down from $\mathbf{z}$; at each node, $\hat{\mathbf{h}}_{\text{parent}}$ predicts child attributes $\hat{\mathbf{v}}$ and hidden states $\hat{\mathbf{h}}_{\text{child}}$. In Stage 2, a Transformer VAE generates the point sequence of each edge conditioned on geometric descriptors. Stage 3 assembles the tree via depth-first-search traversal. For further details, we refer the reader to [10]. We augment the Stage 1 decoder with auxiliary modules (Sect. 3.2) and replace Stage 2 with Chebyshev coefficient–based 3-Phase Curve Diffusion conditioned on a per-edge vector $\mathbf{C} \in \mathbb{R}^9$ (Sect. 3.4), as shown in Fig. 1.

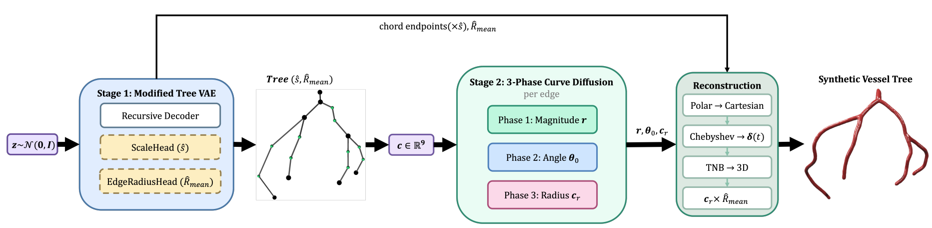

Fig. 1. Pipeline overview. Stage 1 generates a key graph via a modified RVAE (Sect. 3.2). A per-edge condition $\mathbf{C} \in \mathbb{R}^9$, extracted from the key graph, drives the Stage 2 3-Phase Curve Diffusion (Sect. 3.4) to generate curve and radius coefficients, which are reconstructed into the mm-scale vascular tree.

3.2 Stage 1: Modified RVAE

The original RVAE loses mm-scale information through normalization and does not provide per-edge reference radii needed for our ratio-based radius representation (Sect. 3.3). We add two auxiliary modules to the decoder to address both.

ScaleHead and EdgeRadiusHead. Both heads are MLPs applied to detached decoder hidden states $\hat{\mathbf{h}}$, ensuring the auxiliary losses do not alter the learned latent distribution. ScaleHead maps $\hat{\mathbf{h}}_{\text{root}} \mapsto \hat{s}$, predicting the normalization scale factor so that coordinates are recovered to mm scale via $1/\hat{s}$; EdgeRadiusHead maps $[\hat{\mathbf{h}}_{\text{parent}}; \hat{\mathbf{h}}_{\text{child}}] \mapsto \hat{R}_{\text{mean}}$, the per-edge reference for radius generation. The extended loss is $\mathcal{L}_{\text{Stage1}} = \mathcal{L}_{\text{recon}} + \beta\mathcal{L}_{\text{KL}} + \lambda_s(\hat{s} - s)^2 + \lambda_e(\hat{R}_{\text{mean}} - R_{\text{mean}})^2$, where $\mathcal{L}_{\text{recon}}$ and $\mathcal{L}_{\text{KL}}$ are the VAE losses of [25].

3.3 Chebyshev Curve Representation

To model complex vessel curves as continuous functions, we represent each curve in Chebyshev polynomial coefficient space. We construct a tangent-normal-binormal (TNB) frame from the chord connecting the endpoints $\mathbf{P}_s$, $\mathbf{P}_e$ of each edge, as illustrated in Fig. 2 (left): $\mathbf{T} = (\mathbf{P}_e - \mathbf{P}_s) / \|\mathbf{P}_e - \mathbf{P}_s\|$, $\mathbf{N} = (\mathbf{T} \times \mathbf{e}_z) / \|\mathbf{T} \times \mathbf{e}_z\|$, and $\mathbf{B} = \mathbf{T} \times \mathbf{N}$. For parameter $x \in [0, 1]$, we define the curve as follows:

$$\mathbf{P}(x) = (1-x)\,\mathbf{P}_s + x\,\mathbf{P}_e + x(1-x)\big[\delta_n(x)\,\mathbf{N} + \delta_b(x)\,\mathbf{B}\big]. \tag{1}$$

The deflections $\delta_n(x)$ and $\delta_b(x)$ are each expanded in K terms as

$$\delta_n(x) = \sum_{k=0}^{K-1} c_{n,k}\, T_k(2x-1), \quad \text{likewise for } \delta_b,$$

where $T_k(2x-1) = \cos(k \arccos(2x-1))$ are shifted Chebyshev polynomials [11]. The boundary factor $x(1-x)$ in Eq. (1) ensures the curve passes through both endpoints regardless of deflection. Coefficient vectors $\mathbf{c}_n, \mathbf{c}_b \in \mathbb{R}^K$ represent each curve as a fixed $2K$-dimensional vector independent of curve length or point count. We set K=48; root-mean-square reconstruction error averages 0.003–0.020 mm across the dataset, compared to a voxel resolution of approximately 0.3 mm.

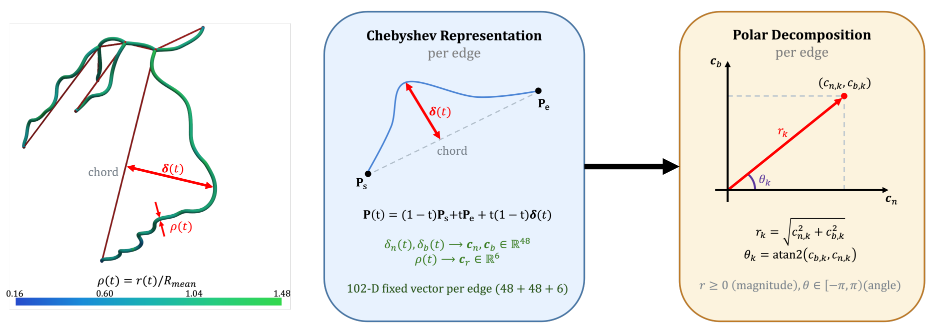

Fig. 2. (Left) A vessel edge with chord, deflection $\boldsymbol{\delta}(x)$, and radius ratio $\rho(x)$. (Center) Chebyshev expansion of $\boldsymbol{\delta}(x)$ and $\rho(x)$ into coefficient vectors, yielding a 102-D vector per edge. (Right) Polar decomposition separates each $(c_{n,k}, c_{b,k})$ pair into magnitude r_k and angle θ_k.

The per-edge radius $r(x) = R_{\text{mean}}\,\rho(x)$ is also represented using the same Chebyshev expansion, with the radius ratio $\rho(x)$ expanded using $K_r = 6$ coefficients as follows:

$$r(x) = R_{\text{mean}} \sum_{k=0}^{K_r-1} c_{r,k}\, T_k(2x-1). \tag{2}$$

During training, we use the ground-truth (GT) R_{mean}; at inference, it is replaced by the EdgeRadiusHead prediction $\hat{R}_{\text{mean}}$.

Polar Decomposition. Each Cartesian pair $(c_{n,k}, c_{b,k})$ couples bending magnitude and direction at degree k. We apply polar decomposition as illustrated in Fig. 2 (right): $r_k = \sqrt{c_{n,k}^2 + c_{b,k}^2}$, $\theta_k = \operatorname{atan2}(c_{b,k},\, c_{n,k})$. This separation empirically improves curve fidelity, as validated in the ablation study (Sect. 4.3).

Conditioning and Hint Nodes. The diffusion models in Sect. 3.4 share a per-edge condition vector $\mathbf{C} = [d,\, R_{\text{mean}},\, g_s,\, g_e;\, f_h,\, \mathbf{o}_1,\, \mathbf{o}_2] \in \mathbb{R}^9$, where g_s, g_e are the node degrees (number of connecting edges) at the source and target nodes. The remaining three entries carry hint-node information. For long and complex edges, the curve shape can be under-determined; hint nodes resolve this ambiguity by providing intermediate trajectory references. Specifically, f_h indicates the presence of hint nodes, while $\mathbf{o}_1, \mathbf{o}_2 \in \mathbb{R}^2$ represent their TNB-frame offsets. We set $f_h = 1$ and store the offsets in $\mathbf{o}_1, \mathbf{o}_2$ for edges with hint nodes; for hint-free edges, $f_h = 0$ and $\mathbf{o}_1 = \mathbf{o}_2 = \mathbf{0}$.

3.4 Stage 2: 3-Phase Curve Diffusion

The magnitude r_k, phase angle θ_k, and radius coefficients $\mathbf{c}_r$ have distinct domains and scales, necessitating separate treatment. We assign a dedicated DDPM to each. In each phase, the clean signal is forward-noised to timestep t via the DDPM schedule, and a noise predictor $\hat{\boldsymbol{\epsilon}}$ learns to recover the added noise.

Phase 1: MagnitudeDiffusion. The magnitude vector $\mathbf{r} = (r_0, \ldots, r_{K-1})$ from polar decomposition (Sect. 3.3) is generated via a DDPM [18] with T=500 steps. The noise predictor is a 4-layer MLP with timestep and $\mathbf{C}$ injected via FiLM [26] conditioning. Huber loss provides robustness to outlier coefficients: $\mathcal{L}_1 = \operatorname{Huber}(\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}(\mathbf{r}_t, t, \mathbf{C}))$.

Phase 2: ThetaDiffusion and ThetaSampler. We observe that higher-order angles $\theta_{k\geq 1}$ show negligible correlation with $\mathbf{C}$ (mean absolute Pearson correlation below 0.04) and accordingly sample them from von Mises distributions [27] fitted to the training data. Only the zeroth angle θ_0, which captures the dominant bending direction of the entire curve, is learned via a DDPM with T=500 steps and $\mathbf{C}$ concatenated to the input. To handle periodicity, the diffusion operates on the unit-circle representation $\mathbf{u} = (\cos\theta_0, \sin\theta_0)$, with loss $\mathcal{L}_2 = \|\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}(\mathbf{u}_t, t, \mathbf{C})\|^2$. The Phase 2 DDPM is used for hint-free edges; for edges with hints, θ_0 is deterministically computed from the hint offset (Sect. 3.3).

Phase 3: RadiusDiffusion. The radius coefficient vector $\mathbf{c}_r = (c_{r,0}, \ldots, c_{r,K_r-1})$ from Eq. (2) is generated via a DDPM with T=200 steps and FiLM conditioning. The zeroth coefficient $c_{r,0}$ determines the overall radius scale, yet has the smallest training-set variance among the radius coefficients. To prevent standard MSE from underweighting it, we use an inverse-variance weighted loss $\mathcal{L}_3 = \sum_{k=0}^{K_r-1} w_k(\epsilon_k - \hat{\epsilon}_k)^2$, where w_k is inversely proportional to the training-set variance of $c_{r,k}$.

Temperature Calibration. We observe that the per-coefficient variance of the generated r_k or $c_{r,k}$ may not match that of the training data, causing over- or under-expressed curve variation. We group edges into bins by chord length and compute the variance ratio $\mathrm{VR} = \sigma^2_{\mathrm{gen}}/\sigma^2_{\mathrm{GT}}$, where σ^2_{gen} and σ^2_{GT} are the within-bin variances of generated and GT values, respectively. The sampling noise scale is scaled by $1/\sqrt{\mathrm{VR}}$ when VR falls outside [0.8, 1.2].
Reconstruction. From the outputs of the three phases, we recover Chebyshev coefficients via inverse polar transform. The 3D curve and radius of each edge are reconstructed using Eqs. (1)–(2). Edges are connected at bifurcation points following the Stage 1 key graph. The scale factor from ScaleHead is applied to assemble the complete vascular tree in mm units.

4 Experiments

Dataset and Setup. We use 855 left coronary artery (LCA) and 861 right coronary artery (RCA) samples from the ImageCAS [1] computed tomography angiography (CTA) dataset (IRB-approved), split 81%/19% for training/held-out validation. We implement on a single NVIDIA RTX 5090. Stage 2 diffusion models each train for 200 epochs with AdamW [28] (10^{-3}). Stage 2 training takes <1 h per artery.
Baselines. We use two baselines, PartVessel [10] and VesselGPT [8]. PartVessel reports superiority over VesselVAE [7] and TrIND [14], so outperforming it subsumes these earlier methods. VesselGPT [8] was originally evaluated on cerebral vasculature; we adapt it to ImageCAS coronary arteries by changing only the input dimension ($39 \to 4$) to match centerline-only data. As VesselGPT supports only unconditional generation, we evaluate 1,000 independently generated samples per artery.
Metrics. We use 163 held-out samples as GT samples and generate 1,000 samples per method. Chamfer Distance (CD) measures tree-level fidelity. Wasserstein distances of curvature, tortuosity, and radius capture edge-level geometric properties, while a tapering distance compares the per-tree proximal-to-distal endpoint-radius ratio. Bounds compliance measures the mean fraction of generated edges whose five geometric attributes (curvature, torsion, radius, length, tortuosity) each fall within the [2.5th, 97.5th] percentile range of the GT distribution. Murray's law compliance [5] quantifies physiological plausibility: it is the fraction of bifurcation points where $(\sum r^3_{\mathrm{child}})/r^3_{\mathrm{parent}}$ falls within $\pm 20\%$ of unity, where r_{parent} and r_{child} are the branch radii at the bifurcation. Since graph topology is determined by Stage 1, we focus on curve-level metrics.

4.1 Quantitative Results

Table 1 reports quantitative results. Our method achieves the best results in 11 of 14 comparisons. Curvature and tortuosity Wasserstein distances are best across both arteries, and Murray's law compliance reaches 35–37%, substantially surpassing PartVessel (0%) and VesselGPT (15–17%). For physiological

Table 1. Held-out quantitative comparison (LCA/RCA, 163 GT). PV = PartVessel [10], VG = VesselGPT [8]. Ours: mean±std over 5 runs (CD: $\times 10^3$). **Bold**: best per artery.

Metric	LCA			RCA		
	PV	VG	Ours	PV	VG	Ours
CD ($\times 10^3$) ↓	**57.0**	99.0	59.1±0.7	72.8	119.9	**59.0±0.8**
Curv. Wass ↓	3.14	2.94	**0.60±0.06**	27.62	35.93	**19.65±0.23**
Tortu. Wass ↓	0.089	0.186	**0.042±0.002**	0.117	0.239	**0.074±0.007**
Radius Wass ↓	0.010	0.005	**0.004±0.000**	0.012	**0.006**	0.009±0.000
Tapering Wass ↓	0.415	**0.082**	0.129±0.003	0.226	0.042	**0.033±0.002**
Murray Comp. ↑	0.000	0.173	**0.367±0.007**	0.000	0.152	**0.346±0.010**
Bounds ↑	0.837	0.780	**0.928±0.001**	0.598	0.508	**0.680±0.003**

context, GT Murray's compliance on the same dataset is only 16.6%/15.9% (LCA/RCA) under the ±20% criterion: GT radii are skeleton-derived from standard centerline-extraction preprocessing rather than direct mm measurements, a property inherited equally by all methods. Under this preprocessing-bounded reference, our 35–37% represents 2.2× GT performance and is uniformly best across methods; closure to ideal Murray would require advances in centerline-radius estimation itself. VesselGPT outperforms ours in tapering (LCA) and radius (RCA), consistent with its explicit B-spline cross-section parameterization [8]; our 3-phase design trades local radius fidelity for global geometry. Tree-level CD is best for RCA and close to PartVessel for LCA, where the gap is attributable to Stage 1 under-generating bifurcations (mean bifurcation count 3.2 vs. GT 4.8) rather than inferior curve quality. ScaleHead and EdgeRadiusHead achieve mm-scale accuracy within 5% (length) and 3% (radius), whereas PartVessel and VesselGPT cannot be evaluated in mm units due to normalization.

4.2 Qualitative Results

We compare generation results in Fig. 3. VesselGPT, despite being the minimum-CD sample among ∼1,000 generated vessels, produces a barely recognizable structure with tokenization-induced piecewise-linear shapes. PartVessel edges deviate from the expected trajectory, appearing to penetrate the heart interior, and show high-frequency fluctuations (dataset-wide curvature variance: PV 107.2 vs. GT 23.9, Ours 21.5). In contrast, our C^∞ representation produces smooth skeletons similar to GT. Our two samples are visually distinct yet anatomically consistent, whereas the two PartVessel samples are nearly identical, suggesting limited diversity. Quantitatively, on LCA our manifold coverage (fraction of GT 6D-feature k-means clusters reached) is 0.70 vs. 0.50 for PartVessel, and the coefficient of variation of per-edge curvature/radius/length is +14%/+59% (LCA/RCA) higher than PartVessel.

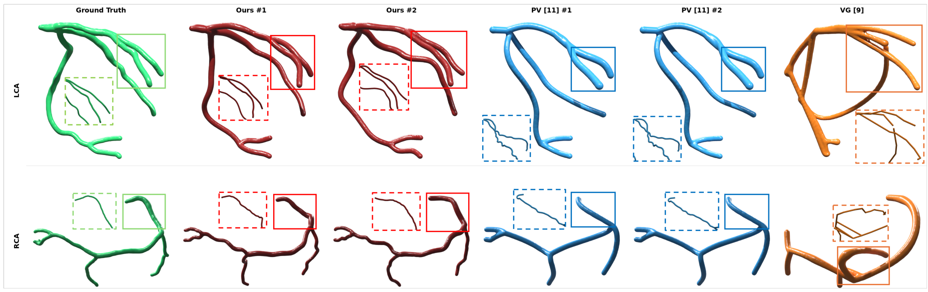

Fig. 3. Qualitative comparison. Given the same GT branching topology, Ours and PartVessel [10] each generate two curve samples. VesselGPT [8]: minimum-CD sample from ∼1,000 unconditional generations. Colors: GT=green, Ours=dark red, PV=blue, VG=orange. Dashed boxes show skeleton details.

4.3 Ablation Study

ScaleHead and EdgeRadiusHead are essential for mm-scale coordinate recovery and are therefore excluded from Table 2.

Table 2. Ablation Study (LCA, Held-out, single run).

Configuration	CD ($\times 10^3$) ↓	Curv. Wass ↓	Tortu. Wass ↓	Murray Comp. ↑	Bounds ↑
Full model (Ours)	58.6	**0.62**	0.042	0.370	0.927
w/o Hint nodes	**53.0**	6.95	0.079	0.290	0.526
w/o Polar decomp.	57.7	7.02	0.270	0.368	0.874
w/o Phase decomp.	60.9	3.04	**0.036**	**0.377**	0.833
w/o Temp. calib.	59.7	0.65	0.040	0.365	**0.928**

Removing hint nodes increases curvature Wasserstein by 11× and decreases bounds by 40 pp; the lower CD reflects curves collapsing to the conditional mean rather than genuine quality. Removing polar decomposition—replacing the three-phase pipeline with a single 102-D Cartesian DDPM—degrades curvature by 11× and tortuosity by 6.4×, confirming that disentangling magnitude and direction is essential. Merging the three DDPMs while retaining polar form (without phase decomposition) marginally improves tortuosity and Murray compliance, but incurs a 5× increase in curvature Wasserstein distance—a critical metric for anatomical plausibility—confirming that phase-wise generation is essential for faithful curvature reproduction. Temperature calibration is a cost-free post-hoc correction whose effect is diluted in aggregate metrics.

5 Conclusion

We have presented 3-Phase Chebyshev Curve Diffusion, which departs from discrete point-sequence representations by generating vessel curves in continuous Chebyshev coefficient space, inherently ensuring C^∞ smoothness. Polar decomposition and phase-wise diffusion faithfully reproduce anatomical curvature and radius profiles, while auxiliary modules recover mm-scale dimensions and preserve curve trajectories. On ImageCAS coronary arteries, the framework achieves superiority over two existing methods in curvature fidelity, Murray's law compliance, and mm-scale accuracy, and qualitative evaluation confirms that the generated vessels are anatomically plausible with smooth and diverse morphologies. Several directions remain for future work: improving Stage 1 to support general graph topologies (e.g., loops as in the circle of Willis) and to address the bifurcation undercount, validating cross-anatomy generalization to cerebral and pulmonary vasculature, and evaluating downstream utility in segmentation training and surgical simulation. Code: `https://github.com/mozzi-bro/Chebyshev_Diffusion`

Acknowledgments. This research was supported by a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI) funded by the Ministry of Health & Welfare, Republic of Korea (grant number: RS-2022-KH129707), by the industry-sponsored research project (Project No. G01260194), and by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2022-0-00469/ RS-2022-II220469, Development of Core Technologies for Task-oriented Reinforcement Learning for Commercialization of Autonomous Drones).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Zeng, A., Wu, C., Lin, G., Xie, W., Hong, J., et al.: ImageCAS: A large-scale dataset and benchmark for coronary artery segmentation based on computed tomography angiography images. Comput. Med. Imaging Graph. **109**, 102287 (2023)
2. Virani, S.S., Alonso, A., Aparicio, H.J., Benjamin, E.J., Bittencourt, M.S., et al.: Heart disease and stroke statistics—2021 update: A report from the American Heart Association. Circulation **143**(8), e254–e743 (2021)
3. Karstensen, L., Robertshaw, H., Hatzl, J., Jackson, B., Langejürgen, J., et al.: Learning-based autonomous navigation, benchmark environments and simulation framework for endovascular interventions. Comput. Biol. Med. **196**, 110844 (2025)
4. Dodge, J.T., Brown, B.G., Bolson, E.L., Dodge, H.T.: Lumen diameter of normal human coronary arteries: Influence of age, sex, anatomic variation, and left ventricular hypertrophy or dilation. Circulation **86**(1), 232–246 (1992)
5. Murray, C.D.: The physiological principle of minimum work: I. The vascular system and the cost of blood volume. Proc. Natl. Acad. Sci. **12**(3), 207–214 (1926)
6. Wolterink, J.M., Leiner, T., Isgum, I.: Blood vessel geometry synthesis using generative adversarial networks. arXiv:1804.04381 (2018)

7. Feldman, P., Fainstein, M., Siless, V., Delrieux, C., Iarussi, E.: VesselVAE: Recursive variational autoencoders for 3D blood vessel synthesis. In: MICCAI, pp. 67–76 (2023)
8. Feldman, P., Sinnona, M., Delrieux, C., Siless, V., Iarussi, E.: VesselGPT: Autoregressive modeling of vascular geometry. In: MICCAI (2025)
9. Prabhakar, C., Shit, S., Musio, F., Yang, K., Amiranashvili, T., Paetzold, J.C., Li, H.B., Menze, B.: 3D vessel graph generation using denoising diffusion. In: MICCAI (2024)
10. Chen, S., Zhang, G., Lai, J., Shen, B., Zhang, S., et al.: Hierarchical part-based generative model for realistic 3D blood vessel generation. In: MICCAI (2025)
11. Trefethen, L.N.: Approximation Theory and Approximation Practice. SIAM (2013)
12. Hamarneh, G., Jassi, P.: VascuSynth: Simulating vascular trees for generating volumetric image data with ground-truth segmentation and tree analysis. Comput. Med. Imaging Graph. **34**(8), 605–616 (2010)
13. Schneider, M., Reichold, J., Weber, B., Szekely, G., Hirsch, S.: Tissue metabolism driven arterial tree generation. Med. Image Anal. **16**(7), 1397–1414 (2012)
14. Sinha, A., Hamarneh, G.: TrIND: Representing anatomical trees by denoising diffusion of implicit neural fields. In: MICCAI (2024)
15. Jo, J., Lee, S., Hwang, S.J.: Score-based generative modeling of graphs via the system of stochastic differential equations. In: ICML (2022)
16. Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., Frossard, P.: DiGress: Discrete denoising diffusion for graph generation. In: ICLR (2023)
17. Farin, G.: Curves and Surfaces for CAGD: A Practical Guide, 5th edn. Morgan Kaufmann (2002)
18. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS, pp. 6840–6851 (2020)
19. Nichol, A.Q., Dhariwal, P.: Improved denoising diffusion probabilistic models. In: ICML, pp. 8162–8171 (2021)
20. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: ICLR (2021)
21. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: ICLR (2021)
22. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
23. Luo, S., Hu, W.: Diffusion probabilistic models for 3D point cloud generation. In: CVPR (2021)
24. Yang, G., Huang, X., Hao, Z., Liu, M.Y., Belongie, S., Hariharan, B.: PointFlow: 3D point cloud generation with continuous normalizing flows. In: ICCV (2019)
25. Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. In: ICLR (2014)
26. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: FiLM: Visual reasoning with a general conditioning layer. In: AAAI (2018)
27. Mardia, K.V., Jupp, P.E.: Directional Statistics. John Wiley & Sons (2000)
28. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2019)