HairGPT: Strand-as-Language Autoregressive Modeling for Realistic 3D Hairstyle Synthesis

HAIMIN LUO, ShanghaiTech University, China
MIN OUYANG, ShanghaiTech University and Deemos Technology Co., Ltd., China
LAN XU, ShanghaiTech University, China
JINGYI YU, ShanghaiTech University, China

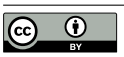
Fig. 1. We introduce "HairGPT", a unified autoregressive framework for realistic 3D hairstyle synthesis. HairGPT uses individual strands as the fundamental generative units and formulates hair generation as structured sequence modeling over strands. By integrating image, text, and hair modalities, HairGPT supports robust cross-modal conditioning. This unified formulation enables the generation of high-frequency hair details and complex topological structures, translating linguistic instructions and visual cues into high-fidelity 3D strands.

Hair is a rich medium of visual and cultural expression, yet its digital modeling remains challenging due to the duality of fluidity and structure. Many existing generative approaches rely primarily on continuous diffusion fields, which entangle global topology with local texture and obscure the semantic and structural organization of hairstyles. To address this, we propose HairGPT, a strand-centric framework that treats strands as generative primitives and formulates realistic 3D hairstyle synthesis as a dual-decoupled autoregressive sequence modeling problem. Our method applies spatial decoupling across semantic scalp regions and structural decoupling along a hierarchical strand representation, progressing from global layout to fine-grained style. We further introduce a geometric tokenizer and region-aware semantic annotations to guide strand-level generation, enabling compositional editing, synthesis of rare and complex hairstyles, and adaptation to stylized domains. By aligning generative modeling with the workflow of digital grooming, HairGPT turns hair generation from opaque texture synthesis into a structured and semantically controllable authoring process, supporting robust semantic conditioning and high-fidelity results across realistic and stylized domains.

CCS Concepts: • **Computing methodologies** → **Computer graphics**; **Computer vision**; **Shape representations**; **Shape modeling**.

Additional Key Words and Phrases: 3D hair synthesis, hairstyle generation, strand-based representation, autoregressive modeling, geometric tokenization, multimodal generation, neural graphics

Authors' Contact Information: Haimin Luo, ShanghaiTech University, Shanghai, China, luohm@shanghaitech.edu.cn; Min Ouyang, ShanghaiTech University and Deemos Technology Co., Ltd., Shanghai, China, ouyangmin2022@shanghaitech.edu.cn; Lan Xu, ShanghaiTech University, Shanghai, China, xulan1@shanghaitech.edu.cn; Jingyi Yu, ShanghaiTech University, Shanghai, China, yujingyi@shanghaitech.edu.cn.



1 Introduction

In the physical world, hair is far more than a biological filament; it is the "silken alphabet" of human identity. From the rebellious punk mohawk to the ceremonial braided bun, hair serves as a dynamic manifesto of culture, gender, and personality. This significance is not merely visual: hair is one of the most broadly shared media of human expression, woven into everyday life through acts of styling, care, and self-presentation. Its richness lies not only in appearance, but also in deliberate authorship and subtle structural design, through which strands are arranged into meaningful and culturally legible forms. Digital hair creation, therefore, should make this authorship equally accessible, translating personal intent into structured, controllable 3D geometry.

Realizing this goal, however, is difficult precisely because hair is not generic geometry to be reproduced, but an intentionally organized structure to be modeled and controlled. Its complexity lies in a paradox: hair is fluid yet structured, chaotic yet groomed. Capturing this duality digitally is not merely a geometric challenge; it is equally an artistic one. Hairstyles are not perceived as arbitrary collections of fibers, but as organized forms governed by anatomical regions and aesthetic intent. Accordingly, generative hair creation should not be framed as holistic synthesis over an undifferentiated signal, but as structured authoring over hierarchically controlled primitives.

Most existing generative approaches [Rosu et al. 2025; Sklyarova et al. 2023b], however, cast hair generation as a 2D texture synthesis problem. By projecting 3D geometry into latent texture maps parameterized by the scalp's UV coordinates, these methods leverage 2D diffusion over continuous geometric fields. While visually compelling, this paradigm fundamentally misaligns with the structured nature of hair. Flattening a 3D composition into a single monolithic latent field inherently entangles global topology with local texture. Consequently, compositional editing—such as modifying curl tightness without destroying the global silhouette, or adjusting bangs without disturbing the crown—becomes brittle or infeasible. This limitation prevents predictable, region-specific control for high-fidelity editing. More broadly, it leaves a profound semantic gap between how human creators intuitively describe hairstyles and how the model internally represents them.

We revisit the strand as the fundamental generative unit, casting hair generation as a structured sequence modeling problem. Rather than denoising an entire hairstyle holistically, we generate hair autoregressively as an ordered assembly of strand groups. To make this tractable for high-fidelity assets, we first condense dense raw hair models into sparse sets of representative guide strands, providing a compact yet expressive scaffold. By mapping these spatial strands into a discrete geometric vocabulary, autoregressive transformers can model hair as a sequence in which each structural decision is conditioned on previously generated geometry. In this view, realistic hair is no longer a monolithic field to be synthesized all at once, but a structured authoring process that progressively organizes strands into coherent forms.

Building on this strand-centric philosophy, we introduce HairGPT, a framework driven by Dual Decoupling along two orthogonal axes. First, we apply spatial decoupling by partitioning the scalp into semantically distinct regions, enabling localized synthesis and targeted editing. Second, we apply structural decoupling by decomposing strand geometry into a hierarchy of generative stages, progressing from global density and layout, to coarse strand shape, and finally to fine-grained style residuals.

Crucially, generation is not performed over a single monolithic sequence. Instead, we introduce a mode-specific autoregressive strategy. The model is trained and sampled under distinct layout, coarse, and style modes. Each mode constructs its own sequence over strand units, avoiding unnecessary coupling across stages and mitigating error accumulation. By decomposing geometry into this hierarchy, HairGPT internalizes the structured logic of professional grooming: it establishes global organization before refining local detail, allowing the model to master one level of abstraction at a time without requiring users to manually author these complex topological steps.

This structural foundation supports more accessible hair creation by exposing controllable abstractions that align with high-level human intent. To connect this structural representation with human-readable control, we introduce a region-aware text annotation pipeline. By fine-tuning a pretrained vision-language model, we extract both global hairstyle descriptions and region-specific local attributes aligned with our scalp partition. Injecting these linguistic tokens into the autoregressive sequence provides semantic guidance for the generative process, allowing intuitive text-driven control without sacrificing structural consistency.

Together, these design choices transform hair generation from an opaque texture synthesis problem into a transparent, compositional modeling process. By aligning representation, training, and inference with the organizational logic of strands and regions, HairGPT enables controllable synthesis, editing, and robust generation of complex hairstyles.

In summary, our contributions are:

- We introduce a **strand-as-language** paradigm for realistic 3D hair generation, reformulating the problem as a dual-decoupled autoregressive process in which strands serve as the fundamental generative units.
- We propose a novel geometric tokenization scheme for guide strands, based on multi-head product quantization, which efficiently encodes complex topology and high-frequency style details into a compact discrete vocabulary.
- We develop a hierarchical strand-language construction together with a multi-stage training strategy, organizing generation into region-aware and stage-specific sequences for improved structural coherence and stable optimization.
- We show that HairGPT supports robust semantic conditioning and compositional editing, enabling high-fidelity generation of rare and complex hairstyles, as well as effective adaptation to stylized domains.

2 Related Work

2.1 3D Hair Representation

Traditional Geometric Modeling. Early attempts to model hair focused primarily on explicit parametric geometric representations, such as parametric surfaces [Koh and Huang 2000; Liang and Huang 2003; Noble and Tang 2004], wisps and generalized cylinders [Chen et al. 1999; Choe and Ko 2005; Kim and Neumann 2002; Patrick et al. 2004; Wang and Yang 2004; Xu and Yang 2001; Yang et al. 2000], and hair meshes [Yuksel et al. 2009]. However, these models typically require significant manual labor to create realistic hairstyles. Subsequent image-based hair modeling efforts [Grabli et al. 2002; Kong and Nakajima 1998; Paris et al. 2004] sought to synthesize strands according to a 3D hair flow volume in a heuristic manner. Other works [Herrera et al. 2012; Luo et al. 2012, 2013; Paris et al. 2008; Wei et al. 2005] employed high-end acquisition systems to capture accurate 3D hair orientation fields, with hair primitives such as wisps and strands derived through subsequent growing procedures.

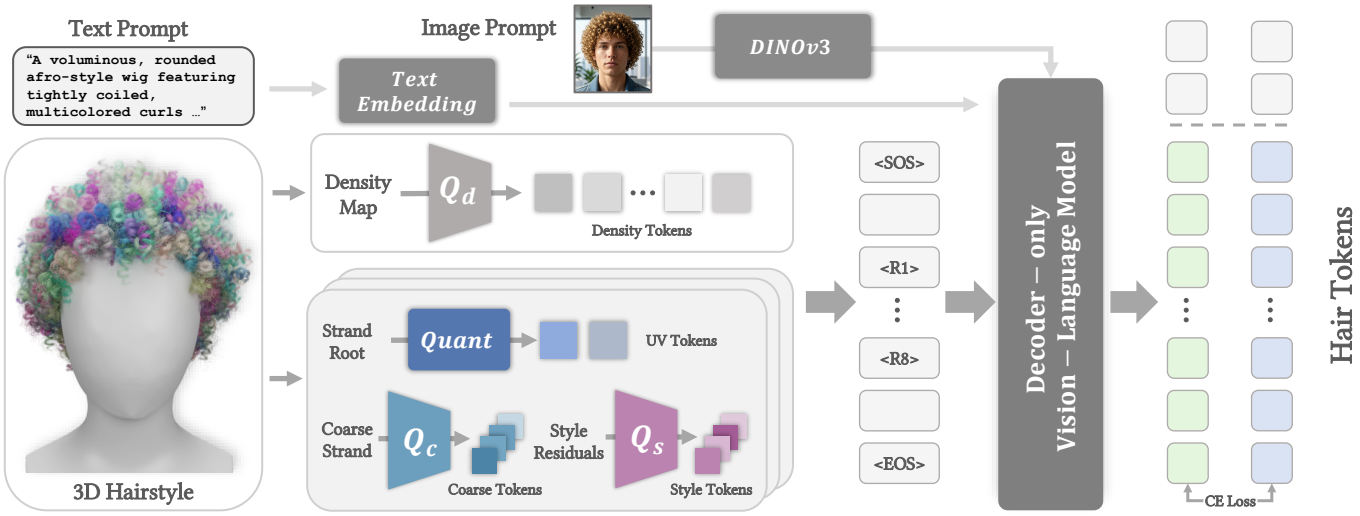


Fig. 2. **HairGPT Overview.** The 3D hairstyle geometry is decomposed into a global density map (quantized by tokenizer Q_d) and local strand features. Specifically, strand roots are encoded into two UV tokens. The strand geometry is decoupled into coarse shape and style residuals, which are further discretized into four tokens by tokenizers Q_c and Q_s. These geometric codes are assembled into a hierarchical sequence, which is processed by a decoder-only Transformer. After concatenation with text and image embeddings, the model autoregressively predicts the target hair tokens and is supervised via cross-entropy loss.

Such sophisticated capture systems prevent these methods from being universally accessible.

Volumetric/Neural Representations. More recent works have introduced neural rendering techniques based on volumetric representations, such as neural orientation fields [Kuang et al. 2022; Sklyarova et al. 2023a; Wu et al. 2022], volumetric points [Wang et al. 2020], and neural radiance fields [Luo et al. 2021, 2022; Mildenhall et al. 2021; Wang et al. 2023, 2022; Wu et al. 2024; Zhou et al. 2024]. These approaches primarily prioritize appearance and rendering, though the resulting geometry is often constrained by the low resolution of the underlying representations. GaussianHair [Luo et al. 2024] and subsequent works [Pan et al. 2025; Zakharov et al. 2024; Zheng et al. 2025; Zhou et al. 2024] reformulate individual hair strands as sequences of cylindrical Gaussians, leveraging the popular 3DGS [Kerbl et al. 2023] framework to model hair from images. These methods achieve high-fidelity strand reconstruction and facilitate efficient real-time rendering. However, they remain fundamentally limited by time-consuming per-scene optimization processes, which lack the flexibility and speed required for interactive, prompt-based hairstyle synthesis.

2.2 Generative and Reconstructive Hair Modeling

Sparse-view Reconstruction. Reconstructing 3D hair from sparse-view images is an ill-posed task that requires strong generative priors. Early data-driven methods [Hu et al. 2014, 2015; Yu et al. 2014; Zhang et al. 2017a] relied on synthetic hairstyle databases containing complete strand geometries. These methods search the database to identify hairstyles that best match the observed sparse views. More recent deep learning-based works [Chai et al. 2016; Kuang et al. 2022; Rosu et al. 2022; Saito et al. 2018; Shen et al. 2023; Sklyarova et al. 2023a; Takimoto et al. 2024; Wu et al. 2024, 2022; Yang et al. 2019; Zhang et al. 2017b; Zheng et al. 2023; Zhou et al. 2018] instead learn shape priors from synthetic hair datasets. In these frameworks, an intermediate 3D hair orientation volume is inferred from sparse image inputs to guide strand-growing algorithms. However, such works tend to be constrained by the limited diversity of hair datasets.

Generative Synthesis. Driven by rapid advancements in 3D geometry synthesis, realistic hairstyle generation has evolved from VAE-based latent exploration [Zhou et al. 2023] to diffusion-based [Rosu et al. 2025] models. *HAAR* [Sklyarova et al. 2023b] established a text-to-strand generation framework, while TANGLED [Long et al. 2025] leverages a multi-view line-art conditional diffusion approach. DiffLocks [Rosu et al. 2025] proposed a scalp-texture diffusion model capable of handling diverse textures, including highly curled and afro-style hair. However, these works often treat hair as an entangled texture, overlooking the anatomical partitioning and structural hierarchy that a human stylist intuitively employs to manage geometric complexity. Alternatively, autoregressive (AR) models treat hair as a sequential "hair language." CHARM [He et al. 2025b] pioneered this by modeling anime hair cards as sequences of control points. Our HairGPT pushes this paradigm further by treating realistic 3D hair geometry as a native linguistic modality, unifying reconstruction and creative synthesis within a single vision-language transformer.

2.3 Vision-Language Models for 3D Generation

The integration of Large Multimodal Models (LMMs) has popularized the "Geometry-as-Language" paradigm. Pioneering works such as PolyGen [Nash et al. 2020] and MeshGPT [Siddiqui et al. 2024] established the foundation for mesh synthesis via discrete token prediction. Building upon these, recent frameworks like Argus [Anonymous 2025] and MeshAnythingV2 [Chen et al. 2024b] demonstrate the potential of transformers to synthesize general meshes with high

topological fidelity. In the hair domain, Hairmony [Meishvili et al. 2024] established a rigorous multidisciplinary taxonomy grounded in anthropology, hair science, and professional grooming, providing a comprehensive classification system for complex hairstyle attributes. Our HairGPT bridges the gap between this taxonomy and geometric synthesis by being the first to operationalize these principles within a native vision-language model.

3 Overview

At the core of HairGPT lies a dual-decoupled hairstyle representation, arising naturally from our view that hair is not generic geometry to be reproduced, but human-authored structure to be organized and controlled. This representation makes it possible to cast hairstyle synthesis as autoregressive generation over explicit strand units, while translating high-level human intent into structured 3D geometry. As illustrated in Fig. 2, we first derive a compact structural abstraction from dense 3D hairstyle geometry, consisting of a global scalp density map and a sparse set of guide strands. Each guide strand is then represented in a dual-decoupled form, where its spatial root is separated from its strand-level geometry, and the latter is further decomposed into a low-frequency coarse backbone and a high-frequency style residual. This structured representation, corresponding to the left and middle parts of Fig. 2, forms the focus of Sec. 4.

Built upon this representation, HairGPT further discretizes the density map and strand components into tokens, assembles them into region-aware hierarchical sequences, and autoregressively predicts them under joint image-text conditioning using a decoder-only transformer. This generative framework, corresponding to the right part of Fig. 2, is presented in Sec. 5. In this way, Sec. 4 establishes the core hairstyle parameterization, while Sec. 5 shows how this parameterization is operationalized within a multimodal autoregressive framework for structured hairstyle synthesis.

4 Dual-Decoupled Hairstyle Parameterization

We represent hairstyles as collections of individual strands, reflecting a grooming-inspired construction process in which strands are progressively placed and shaped over the scalp. Based on this view, our goal is to derive a compact strand-based representation from dense raw hair geometry that makes explicit both the global organization of hair over the scalp and the local geometric structure of individual strands. To this end, we decouple hairstyle modeling along two orthogonal axes: scalp-level spatial organization, which determines where strands are placed, and strand-level geometry, which determines how each strand is shaped from coarse flow to fine detail. This section introduces the resulting dual-decoupled parameterization, together with the guide-strand abstraction and strand decomposition that serve as the structural foundation of HairGPT.

4.1 The Dual-Decoupled Representation

We introduce a dual-decoupled representation that separates scalp-level spatial organization from strand-level geometric structure. It provides a structured interface for independently modeling strand placement and geometry within a sequential generation process.

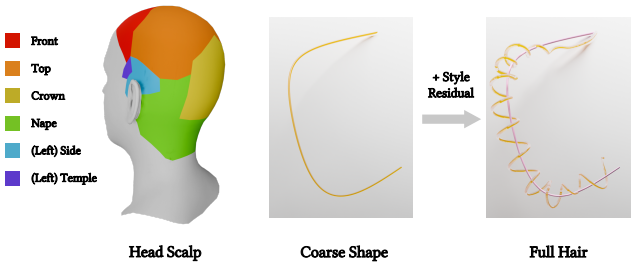


Fig. 3. **Dual-Decoupled Representation.** The scalp is semantically partitioned into eight regions, and each strand is decoupled into a low-frequency coarse backbone and a high-frequency style residual.

Spatial Decoupling via Scalp Partitioning. At the macroscopic level, hairstyle generation is formulated as a spatial planning problem on the 2D scalp manifold $\mathcal{S}$. Following the semantic taxonomy of *Hairmony* [Meishvili et al. 2024], we partition $\mathcal{S}$ into $M = 8$ regions $\mathcal{R} = \{\text{Front, Top, Crown, Nape, Right/Left Temple, Right/Left Side}\}$ (Fig. 3). This partitioning provides an explicit spatial prior that enforces locality during strand generation. In addition, we define a scalp density map $\mathcal{D} : \mathcal{S} \to \mathbb{R}^+$ to guide root placement independently of strand geometry.

Structural Decoupling via Strand Hierarchy. At the strand level, we model geometry generation itself as a hierarchical process. The geometry of a strand $\mathbf{s}$ is decomposed into a structured tuple $\mathbf{s} = (\mathbf{u}, \mathbf{c}, \mathbf{r})$, reflecting a coarse-to-fine synthesis within each autoregressive step. The root location $\mathbf{u} \in \mathcal{S}$ anchors the strand to the spatial plan. The coarse shape $\mathbf{c}$ encodes the low-frequency backbone of the strand, capturing global flow and topology. The style residual $\mathbf{r}$ adds high-frequency geometric detail (Fig. 3), yielding the final strand geometry $\mathbf{s} = \mathbf{c}+\mathbf{r}$. This explicit separation allows the model to reuse fine-scale style patterns across different global strand shapes, which is essential for learning strand-level stylistic regularities within a sequential generative process.

4.2 Sparse Guide Strand Extraction

Directly processing a raw hair model $\mathcal{H}$ is impractical due to its scale ($\sim 10^5$ strands) and the lack of an inherent strand-level ordering. We therefore condense the dense geometry into a sparse set of representative *Guide Strands* via clustering, yielding a strand-level sequence suitable for autoregressive modeling.

Frequency-Aware Strand Clustering. To reduce sensitivity to high-frequency noise, we perform clustering in a frequency-based feature space. Since hair strands are open, non-periodic 3D curves, we use the Discrete Cosine Transform (DCT) rather than the Discrete Fourier Transform (DFT): the periodic assumption of the DFT would introduce an artificial root-tip discontinuity, and low-frequency truncation may produce ringing near strand boundaries. The DCT avoids this issue through its implicit symmetric extension, yielding smoother low-frequency descriptors for strand shape analysis [Chen et al. 2024a].

Each raw strand $\mathbf{s}_i \in \mathcal{H}_{\text{raw}}$ is mapped to a compact shape descriptor $\mathbf{z}_i$ using the Discrete Cosine Transform (DCT), after subtracting

Fig. 4. **Continuous density maps in our dataset.** We show several representative hairstyles together with their corresponding scalp-space density maps.

its root position to remove global translation:

$$\mathbf{z}_i = \mathcal{T}_{K_{\text{feat}}}\left(\text{DCT}(\mathbf{s}_i - \mathbf{s}_{i,\text{root}})\right) \in \mathbb{R}^{K_{\text{feat}}\times 3}. \tag{1}$$

We retain the first $K_{\text{feat}} = 8$ coefficients, which act as a low-pass representation capturing the dominant strand shape. The strands are then grouped using k-means clustering into $N_{\text{guide}} = 512$ clusters by minimizing the standard intra-cluster variance $\sum \|\mathbf{z}_i - \boldsymbol{\mu}_j\|^2$.

For each cluster, the centroid strand is selected as the guide strand, providing a compact geometric proxy for the dense hair volume. In parallel, we project the root positions of all raw strands onto the scalp UV domain to compute a density map $\mathcal{D}$, which represents the spatial distribution of hair roots over the scalp, similar to DiffLocks [Rosu et al. 2025]. Each continuous value in $\mathcal{D}$ reflects the relative likelihood of a strand root occurring at the corresponding UV location, thereby providing a global prior for strand placement during generation. We visualize representative hairstyles together with their corresponding scalp-space density maps in Fig. 4.

Together, the extracted guide strands provide a sparse, strand-level abstraction suitable for autoregressive modeling.

4.3 Spectral-Spatial Decomposition

Based on the extracted guide strands, we further decompose each strand into a coarse geometric backbone and a fine-scale style residual, instantiating the strand-level hierarchy introduced in Sec. 4.1.

Frequency-Based Coarse Geometry. A guide strand is represented as an ordered point sequence $\mathbf{P} = \{\mathbf{p}_0, \ldots, \mathbf{p}_{L-1}\}$. We operate on segment direction vectors $\mathbf{v}_j = \mathbf{p}_{j+1} - \mathbf{p}_j$ and apply the Discrete Cosine Transform (DCT) to the sequence $\mathbf{V} = \{\mathbf{v}_j\}$, which more directly captures the intrinsic flow of the strand. Retaining the first $K_{\text{geo}} = 4$ coefficients yields filtered directions $\hat{\mathbf{V}}$, which are recovered via inverse DCT:

$$\hat{\mathbf{V}} = \text{IDCT}\left(\mathcal{T}_{K_{\text{geo}}}\left(\text{DCT}(\mathbf{V})\right)\right). \tag{2}$$

The coarse geometry $\hat{\mathbf{P}} = \{\hat{\mathbf{p}}_j\}$ is reconstructed by cumulative integration: $\hat{\mathbf{p}}_j = \mathbf{p}_0 + \sum_{m=0}^{j-1} \hat{\mathbf{v}}_m$, yielding a smooth backbone curve that captures global strand flow.

Scaled Style Residuals. To encode fine-scale strand details independently of global orientation and physical scale, we define style residuals as normalized deviations from the coarse backbone, expressed in a local coordinate system along the strand. Notably, although the coarse backbone is extracted via DCT, the residuals are not simply the discarded high-frequency signal; instead, they are defined in a scale-normalized local frame to capture reusable strand-level style patterns.

Specifically, we first introduce a local scale factor σ_j to remove the influence of strand-sampling density and strand length:

$$\sigma_j = \frac{1}{2}\left(\|\hat{\mathbf{p}}_j - \hat{\mathbf{p}}_{j-1}\| + \|\hat{\mathbf{p}}_{j+1} - \hat{\mathbf{p}}_j\|\right), \tag{3}$$

To ensure invariance to global orientation, we represent these normalized deviations in a local orthonormal frame $\mathbf{F}_j$ defined along the coarse backbone. At each point $\hat{\mathbf{p}}_j$, the frame is constructed using Parallel Transport [Bishop 1975]. Together with the scale factor, the style residual at each point is computed as

$$\mathbf{r}_j = \frac{1}{\sigma_j}\mathbf{F}_j^\top(\mathbf{p}_j - \hat{\mathbf{p}}_j). \tag{4}$$

Since $\hat{\mathbf{p}}_j$, $\mathbf{F}_j$, and σ_j are deterministically defined from the strand, the original strand can be exactly reconstructed from the residuals and coarse backbone, making the parameterization bijective. This formulation therefore yields a scale- and rotation-invariant representation of local strand texture, allowing residual patterns to be compared and reused across strands as meaningful strand-level "style words," and to be naturally organized as a sequential token stream for autoregressive hairstyle synthesis.

5 HairGPT: Unified Autoregressive Hairstyle Synthesis

Building upon the dual-decoupled parameterization in Sec. 4, we next introduce the generative framework of HairGPT. The key idea is to convert the structured hairstyle representation into a discrete, language-like modality that can be modeled autoregressively under multimodal conditions. To this end, we first construct aligned image-text-hair training data; then we discretize the guide strands, including spatial roots $\mathbf{u}$, coarse geometry $\mathbf{c}$, and style residuals $\mathbf{r}$, as well as the density map, into tokens, organize them into region-aware hierarchical sequences, and model their generation with a decoder-only vision-language model. This section describes how the representation defined in Sec. 4 is operationalized into a unified multimodal framework for structured hairstyle synthesis.

5.1 Scalable Multimodal Data Engine

To address the scarcity of paired 3D hair data, we construct a scalable pipeline that converts heterogeneous sources into aligned triplets $(\mathcal{I}, \mathcal{T}, \mathcal{H})$ of image, text, and hairstyle geometry.

Generative Visual Synthesis ($\mathcal{I}$). To bridge the gap between synthetic training data and real-world inference, we leverage a pre-trained generative model (e.g., Qwen-Image) to synthesize diverse visual identities, as shown in Fig. 5. We prompt the VLM with canonical hair renders and specific attribute instructions, systematically varying skin tones, clothing styles, lighting conditions, and background scenes. This generative augmentation reduces domain bias and improves robustness to real-world visual conditions.

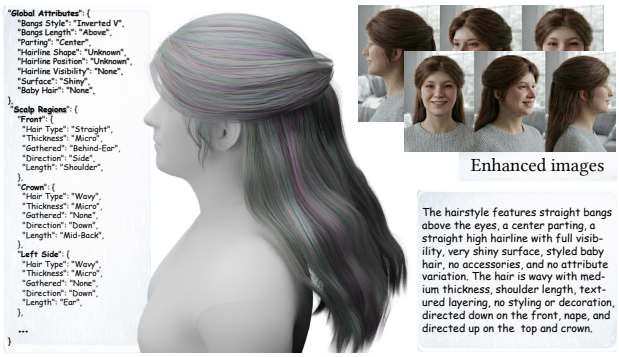


Fig. 5. **Data Example.** For a 3D hair-strand model, we annotate global and local text attributes for distinct scalp regions and provide an overall natural-language hairstyle description. We also utilize a generative model to render diverse photorealistic identities consistent with the underlying hair topology.

Region-Aware Semantic Annotation ($\mathcal{T}$). We observe that large pretrained VLMs already encode rich semantic priors for hairstyle understanding. Beyond recognizing coarse global attributes, they are often capable of describing how a hairstyle is deliberately authored—including how volume, flow, and local styling cues are arranged across different parts of the head to produce a coherent overall design. This makes them a natural source of semantic supervision for our problem, since they already capture much of the human prior needed to verbalize hairstyle structure from images. However, their outputs are typically unstructured and global, and do not explicitly align with localized scalp regions (e.g., distinguishing a fringe from temple hair).

To leverage such pretrained priors while enforcing semantic structure, we further fine-tune Qwen2.5-VL-7B-Instruct using Low-Rank Adaptation (LoRA) on a curated set of expert-labeled hairstyle images from Hairmony [Meishvili et al. 2024]. Concretely, we perform supervised fine-tuning on the Hairmony dataset, where each image is paired with manually annotated structural hair-part attributes, enabling the model to recognize and categorize distinct hair parts from input portraits. We use LoRA adapters with intrinsic rank $r = 8$ on all linear modules in the transformer blocks, covering both attention and MLP layers.

The resulting domain-adapted VLM is prompted to generate hierarchical textual annotations that follow the Hairmony taxonomy, producing both a *global hairstyle description* and *region-specific labels* for each of the $M = 8$ scalp regions (Sec. 4.1). These region-aware captions provide fine-grained semantic supervision that is explicitly aligned with strand geometry, enabling consistent cross-modal learning between text and 3D hair structure, as shown in Fig. 5.

Data Aggregation and Geometric Lifting. Our dataset combines synthetic priors (DiffLocks [Rosu et al. 2025], Perm [He et al. 2025a]), hundreds of artist-authored hairstyles, and reconstructed 3D hair from the Hairmony dataset. For image-only data, we lift geometry using a diffusion-based reconstruction model and apply VLM-based rejection sampling for quality control. The final corpus contains approximately **120k aligned triplets**.

5.2 Disentangled Geometric Tokenization

To enable discrete autoregressive generation, we discretize the dual-decoupled representation established in Sec. 4 through three specialized tokenization processes targeting spatial roots, strand geometry, and the global density map.

Strand Root Quantization ($\mathbf{u} \to \{u, v\}$). We discretize the continuous UV coordinates of the root $\mathbf{u} \in \mathcal{S}$ into integer indices. Specifically, we map the 2D scalp manifold $\mathcal{S}$ to a 256×256 grid. Each guide strand root $\mathbf{u}$ is quantized into a spatial token pair (u, v), where $u, v \in [0, 255]$. These tokens act as the spatial anchors for all subsequent geometric attributes in the generative sequence.

Multi-Head Geometric VQ-VAE ($\mathbf{c}, \mathbf{r} \to T_{coa}, T_{sty}$). We train two independent encoders for the coarse backbone $\mathbf{c}$ and the style residual $\mathbf{r}$, respectively. To enhance codebook expressivity and mitigate codebook collapse, we employ a product quantization strategy with four heads. The latent representations of $\mathbf{c}$ and $\mathbf{r}$ are partitioned into four sub-vectors, each discretized by an independent sub-codebook. This process yields a sequence of four coarse tokens $T_{\text{coa}} = \{c_1, \ldots, c_4\}$ and four style tokens $T_{\text{sty}} = \{r_1, \ldots, r_4\}$ per strand. This compact design uses only eight discrete tokens per strand, which is critical for keeping autoregressive sequence lengths manageable at the scale of hundreds of guide strands; by contrast, more direct parameterizations based on dense curve coefficients would require substantially longer token sequences, making training impractical for hairstyles with 512 guide strands.

The geometric tokenizers are pretrained on high-quality synthetic datasets (DiffLocks and Perm) to learn robust geometric priors, and remain frozen during the LLM training stage to ensure latent stability.

Global Density ($\mathcal{D} \to \mathbf{D}$). Beyond individual strands, we encode the density map $\mathcal{D}$ (defined in Sec. 4.1) using a standard 2D VQ-VAE. This autoencoder compresses the dense scalp-space density grid into a 32×32 latent feature map, which is subsequently quantized into a sequence of 1024 density tokens $\mathbf{D} = \{d_1, \ldots, d_{1024}\}$. These tokens $\mathbf{D}$ serve as a global structural condition, empowering the model to reason about the overall hair-root distribution before synthesizing specific guide strands.

5.3 Autoregressive Hairstyle VLM

5.3.1 Model Architecture and Multimodal Alignment. We adopt the pretrained Qwen model as our backbone transformer Φ. To bridge the structural hair representation with the linguistic space, we unify all discrete hair components—including UV coordinates (from $\mathbf{u}$), coarse backbones $\mathbf{c}$, style residuals $\mathbf{r}$, and density maps $\mathcal{D}$—into a unified vocabulary $\mathcal{V}$. Each component's local index is mapped to a global token ID $T \in \mathcal{V}$ via component-specific offsets. For any geometric token at step t, its input representation $\mathbf{x}_t$ is derived from the learnable embedding table: $\mathbf{x}_t = \mathbf{e}_{\text{geo}}(T_t)$, where $\mathbf{e}_{\text{geo}}$ denotes the embedding lookup function corresponding to the global token ID T_t. The model processes multimodal conditions $\mathcal{C} = \{\mathcal{I}, \mathcal{T}\}$, where $\mathcal{I}$ is a reference image and $\mathcal{T}$ represents linguistic instructions. To incorporate visual guidance, we extract patch-level features from a frozen DINOv3 [Siméoni et al. 2025] encoder and map them to the transformer's hidden dimension via a lightweight projector

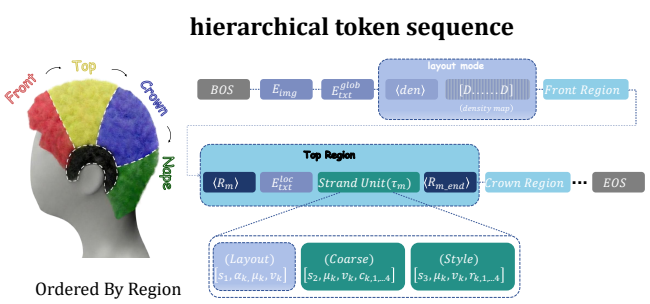


Fig. 6. **Sequence Illustration.** The sequence begins with global text and image embeddings and density map tokens (layout mode). Task-specific strand tokens are generated and constructed region by region, bounded by region markers. Each strand unit is constructed according to the task mode and conditioned on specific separators.

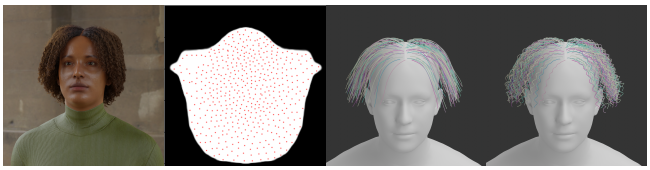


Fig. 7. **Phased Autoregressive Generation.** HairGPT progressively generates a hairstyle through multiple phases. The density phase first predicts density tokens, which then condition the following layout phase. Strand-root positions are generated sequentially and visualized as red points. Coarse-strand geometry tokens are then generated, and the style phase finally produces fine-grained residual details. Additional implementation details are provided in the supplementary material.

$\mathcal{P}_{\text{img}}$: $\mathbf{E}_{\text{img}} = \mathcal{P}_{\text{img}}(\text{DINOv3}(\mathcal{I}))$. In contrast, textual prompts $\mathcal{T}$ are processed by retaining Qwen's original word embeddings, which preserves the backbone's pretrained linguistic knowledge. These textual features are dimensionally aligned as:

$$\mathbf{E}_{\text{txt}} = \mathcal{P}_{\text{txt}}(\mathbf{e}_{\text{word}}(\mathcal{T})) \tag{5}$$

where $\mathcal{P}_{\text{txt}}$ ensures dimensional consistency. This hybrid embedding strategy allows the transformer Φ to interpret visual cues through the specialized projector while maintaining linguistic consistency for textual instructions.

5.3.2 Hierarchical Sequence Construction. We linearize a hairstyle into a structured sequence S that progresses from global layout to strand-level geometric details, enabling autoregressive generation with explicit structural control, as illustrated in Fig. 6.

Multi-stage Strand Units (τ_k). To preserve structural disentanglement, each hair strand k is decomposed into a multi-stage subsequence τ_k, organized by task-specific separator tokens $\{s_1, s_2, s_3\}$. Conditioned on global context C, the transformer treats spatial anchors (α_k, u_k, v_k) as coordinate queries and predicts the corresponding strand geometry. This formulation allows the transformer to be interpreted as a discrete, autoregressive implicit mapping that predicts strand geometry conditioned on spatial queries and global context.

Specifically, the strand representation is generated through three progressive modes, as illustrated in Fig. 7:

- **Layout Mode:** $\tau_k^{\text{lay}} = [s_1, \alpha_k, u_k, v_k]$, where α_k is a density-aware anchor derived from the density map $\mathcal{D}$. This stage instantiates strand root locations by mapping the global density prior to precise coordinates on the scalp manifold.
- **Coarse Mode:** $\tau_k^{\text{coa}} = [s_2, u_k, v_k, c_{k,1}, \ldots, c_{k,4}]$, which predicts the low-frequency backbone geometry $\mathbf{c}_k$ conditioned on the spatial query (u_k, v_k).
- **Style Mode:** $\tau_k^{\text{sty}} = [s_3, u_k, v_k, r_{k,1}, \ldots, r_{k,4}]$, which generates high-frequency style residuals $\mathbf{r}_k$ with spatial tokens.

By reusing the same spatial coordinates (u_k, v_k) across all stages, the layout stage strictly determines strand placement, while the coarse and style stages are constrained to only generate geometry conditioned on this fixed spatial anchor.

Region-Aware Assembly. Strands are grouped by scalp regions $\mathcal{R}_m$ and assembled into a mode-conditioned sequence:

$$\begin{aligned} \mathbf{S}^{(\cdot)} = \Big[&\text{BOS},\ \mathbf{E}_{\text{img}},\ \mathbf{E}_{\text{txt}}^{\text{glob}},\ \langle\text{den}\rangle,\ \mathbf{D}, \\ &\ldots,\ \langle\mathcal{R}_m\rangle,\ \mathbf{E}_{\text{txt},m}^{\text{loc}},\ \tau_{m,1\ldots N_m}^{(\cdot)},\ \langle\mathcal{R}_m\text{\_end}\rangle, \\ &\ldots,\ \text{EOS}\Big],\ \tau_k^{(\cdot)} \in \{\tau_k^{\text{lay}}, \tau_k^{\text{coa}}, \tau_k^{\text{sty}}\} \end{aligned} \tag{6}$$

where $\mathbf{S}^{(\cdot)}$ denotes one of the layout, coarse, or style sequences. BOS and EOS denote sequence boundary tokens. The global and region-specific text embeddings are aligned via Eq. 5. The markers $\langle\mathcal{R}_m\rangle$ and $\langle\mathcal{R}_m\text{\_end}\rangle$ delimit semantic regions. The fixed region order is `Front`, `Top`, `Crown`, `Nape`, followed by the left and right sides and temples, as shown in Fig. 6.

5.3.3 Multi-Stage Training Strategy.

Mode-Specific Sequence Sampling. During training, each sequence $\mathbf{S}^{(\cdot)}$ is constructed from a *single generation mode*—layout, coarse, or style—rather than concatenating all strand units. This ensures the model learns each stage independently while conditioning on the relevant global and regional context.

Training Loss. The model is trained via next-token prediction with a redundancy mask $\mathcal{M}$ to exclude repeated spatial tokens (u_k, v_k) from the loss in the coarse and style stages. Let T_t denote the token at step t and $\omega(T_t)$ its category-aware weight:

$$\begin{aligned} \mathcal{L} = -\frac{1}{N_{\text{valid}}} \sum_t \Big[&\omega(T_t) \cdot \mathbb{1}(T_t \notin \mathcal{M}) \\ &\cdot \log P(T_t | T_{<t}, C)\Big]. \end{aligned} \tag{7}$$

Repeated spatial tokens serve purely as conditioning anchors and are excluded from gradient computation. Multimodal dropout is applied concurrently to further improve robustness to variations in visual and textual prompts.

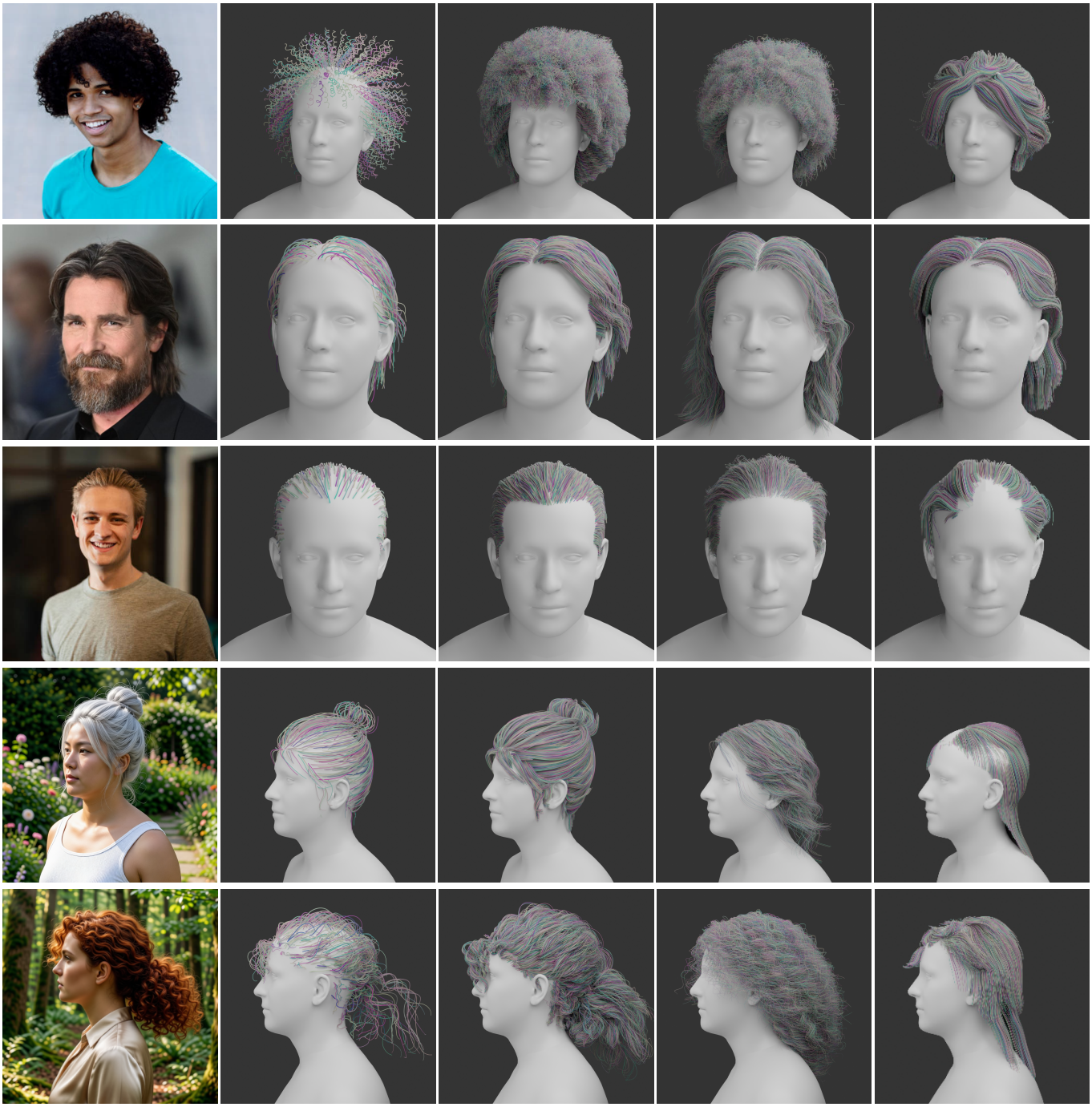


Fig. 8. **Image-guided hairstyle synthesis comparison.** HairGPT effectively generates tightly coiled hairstyles and complex hair topology conditioned on the input image, especially for buns and ponytails. We visualize both the raw guide strands directly output by our model and the dense strands produced via a simple interpolation algorithm; note that this upsampling process is employed solely for visualization and is not the primary focus of this work.

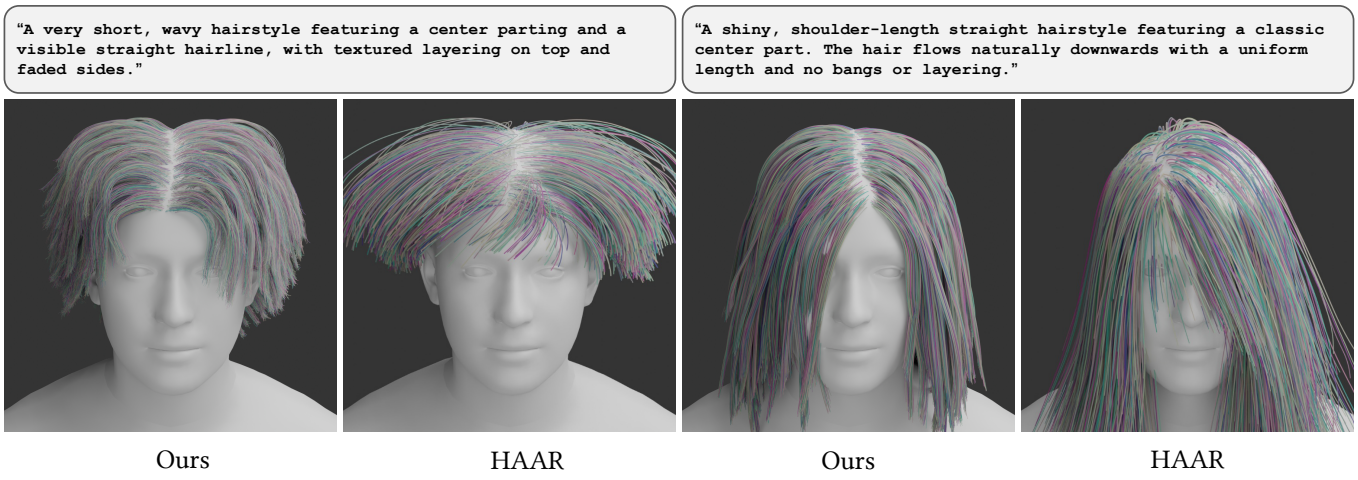


Fig. 9. **Text-guided hairstyle synthesis comparison.** Our HairGPT produces 3D hairstyles that adhere to fine-grained semantic instructions.

Table 1. CLIP score analysis based on the similarity between the input image condition and rendered 3D hairstyles.

Method	DiffLocks	HairStep	Ours
CLIP Score ↑	0.577	0.537	**0.592**

Table 2. Quantitative comparison on the DiffLocks evaluation set.

Method	CD ↓ ($\times 10^{-3}$)	DCT-FID ↓ ($\times 10^{-3}$)	Precision ↑	Recall ↑	F-score ↑
Ours	**6.31**	**8.80**	0.824	**0.833**	**0.828**
DiffLocks	7.83	9.97	**0.839**	0.809	0.824

6 Results

We evaluate our model by comparing it with hair generation baselines, key components, and training strategies. Implementation and inference details are provided in the supplementary material.

6.1 Comparisons

We compare HairGPT against state-of-the-art hair synthesis baselines, including diffusion-based methods (*DiffLocks* [Rosu et al. 2025]) and reconstruction methods (*HairStep* [Zheng et al. 2023]) for image-guided generation. We also compare our approach with *HAAR* [Sklyarova et al. 2023b] in terms of text-guided synthesis capability.

Image-guided Generation. We provide only a single reference image for both HairGPT and the baseline methods. While HairStep overlooks fine-grained local hair details and DiffLocks struggles to synthesize hairstyles with complex topological structures, such as ponytails, as illustrated in Fig. 8, our framework successfully synthesizes a diverse spectrum of hairstyles—ranging from high-frequency afro hairstyles to complex topological structures, such as buns. Leveraging our dual-decoupled representation, our model maintains superior structural consistency and detail preservation.

Text-guided Generation. We provide only text prompts for our HairGPT and HAAR. As illustrated in Fig. 9, our method produces 3D hairstyles that adhere to fine-grained semantic instructions—such as precise parting locations and specific regional styles (e.g., diagonal bangs)—and exhibits superior structural coherence compared to HAAR, benefiting from the language priors inherited from the backbone model.

Quantitative analysis. We also evaluate the generation quality of our model quantitatively. First, we report CLIP scores in Tab. 1 and compare against DiffLocks and HairStep. Without ground-truth (GT) geometry, we compute the CLIP scores based on the feature similarity between the input condition and the rendered 3D hairstyles, using the same lighting, camera views, and shaders for all methods. We note that the CLIP score may overemphasize local texture cues while overlooking the global topology of hair geometry, which can slightly underestimate the performance of our method. We further evaluate geometric reconstruction quality on the DiffLocks evaluation set. In Tab. 2, we report point-cloud Chamfer distance and Precision/Recall/F-score metrics between the generated hairstyles and GT strands. We also compute the mean DCT-FID by representing each strand with DCT coefficients and measuring the Fréchet distance between the generated and GT strand distributions of all hairstyles.

6.2 Ablation Study

Multi-head Strand Tokenizer. We evaluate the effectiveness of the tokenizer design with the same parameter budget (0.26M). As shown in Tab. 3, a baseline *Single-head* exhibits limited capability and lower codebook usage, while the results of our multi-head design (Rows 2–3) indicate that product quantization improves the capacity of the codebook.

Coarse-Style Decoupling. We further evaluate the necessity of the decoupled representation (Sec. 4.1). As illustrated in Fig. 10, a tokenizer trained on raw strands directly (*w/o c/r*) struggles to recover high-frequency curling patterns, and fails to capture the characteristic tight coils of the reference afro-texture. Quantitatively, Tab. 3

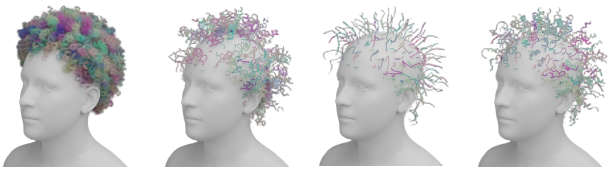


Fig. 10. Ablation of strand-level Coarse-Style Decoupling. The decoupling effectively enables the tokenizer to encode high-frequency details.

Table 3. Ablations on tokenizer and strand-level Coarse-Style Decoupling.

Method	Pos ↓	Dir ↓	Curv ↓	Usage (%) ↑	Heads ↑	#Params
Single-head	4.5×10^{-3}	1.2×10^{-2}	6.3×10^{-4}	63.9	1	0.26M
W/o **c**/**r**	2.5×10^{-3}	2.0×10^{-2}	6.5×10^{-4}	**99.1**	4	0.26M
Ours-full	$\mathbf{1.2 \times 10^{-3}}$	$\mathbf{3.4 \times 10^{-3}}$	$\mathbf{6.3 \times 10^{-4}}$	98.9	4	0.26M

(Row 2) shows that without coarse-style decomposition, the model suffers from significantly higher directional error and positional drift. In contrast, our full model can disentangle high-frequency residuals (**r**) from the low-frequency topological backbone (**c**).

Impact of Sequence Construction. We evaluate the role of task-specific separators by evaluating an *Interleaving* baseline, where the root (u), coarse (c), and style (r) tokens of all strands are directly concatenated into a continuous stream. As shown in Fig. 11, this configuration leads to complete geometric collapse. Without the anchoring effect of separators to clarify the task, the model suffers from severe autoregressive drift; the error accumulates through the subsequent geometry tokens, resulting in strands that are scattered chaotically and fail to form a coherent hairstyle.

Multi-stage Training. Furthermore, we evaluate the importance of our *Layout*/*Coarse*/*Style* staged training strategy. In the *w/o stage* variant, the model is trained on a single, flattened sequence containing all geometric layers at once. While this baseline can capture basic silhouettes for low-frequency straight hair, it struggles significantly with high-frequency afro-textures (bottom row), due to the long-range dependency bottleneck in extremely long sequences ($N = 512$ strands, about 10k tokens). Without staged supervision, the high-frequency style details at the end of the sequence suffer from information decay and attention attenuation relative to the global conditions. This results in the sparse, fragmented strands observed in Fig. 11.

6.3 Applications

Cross-Domain Adaptation. Although HairGPT is pretrained on realistic data, the structural priors it acquires facilitate efficient domain adaptation via fine-tuning. Fig. 12 illustrates the results on 2D anime characters: the model infers plausible hair topology from flat-shaded images, preserving the distinct aesthetic characteristics of the input styles. This capability highlights the potential of our representation to serve as a unified geometric backbone that can be specialized for diverse artistic domains with limited additional data.

Fig. 11. Ablations on sequence construction and multi-stage training.

Realistic Avatar Creation. Conditioned on compatible semantic attributes (e.g., specific eras or cultural styles), HairGPT can work in conjunction with the 3D face synthesis model [Zhang et al. 2023] to produce photorealistic avatars with unified visual aesthetics (Fig. 13). Crucially, this approach maintains geometric disentanglement, providing high-quality, separable hair and face assets that are ready for independent animation and editing.

Flexible Multimodal Editing. Our dual-decoupled representation and vision-language model naturally facilitate diverse editing applications. As shown in Fig. 14, users can explicitly control hair volume and distribution by modifying the density map (a). Thanks to the disentanglement of topology and texture, the coarse geometry can be guided by a reference image to transfer global styles (c), while local details—such as specific curl patterns (b) or regional shapes like bangs (d)—can be precisely altered via regional text prompts, without disrupting the overall hair structure.

6.4 Limitations and Discussion

Limitations. Despite the strong controllability and semantic expressiveness enabled by our strand-centric formulation, the current framework still has several limitations. First, our data engine relies in part on a fine-tuned vision-language model for scalable semantic annotation and augmented supervision. While this greatly improves data construction efficiency, the resulting labels still do not fully match expert human annotation, especially for subtle regional attributes, ambiguous boundaries, and rare hairstyle patterns. Incorporating more human-curated annotations and stronger human-in-the-loop verification would further improve the pipeline's reliability.

Second, strand tokenization remains inherently challenging. Our compact design represents each strand using only eight discrete tokens, which is critical for making autoregressive modeling feasible at the scale of hundreds of guide strands. However, this compactness inevitably sacrifices some local geometric fidelity. Although reconstruction quality is sufficient in our setting, strands with extremely high curvature or irregular local textures may be smoothed. Richer local parameterizations, adaptive tokenization, or hybrid detail representations may help alleviate this limitation.

Third, our framework focuses on generating structurally meaningful guide strands rather than dense hair directly. Dense hairstyles

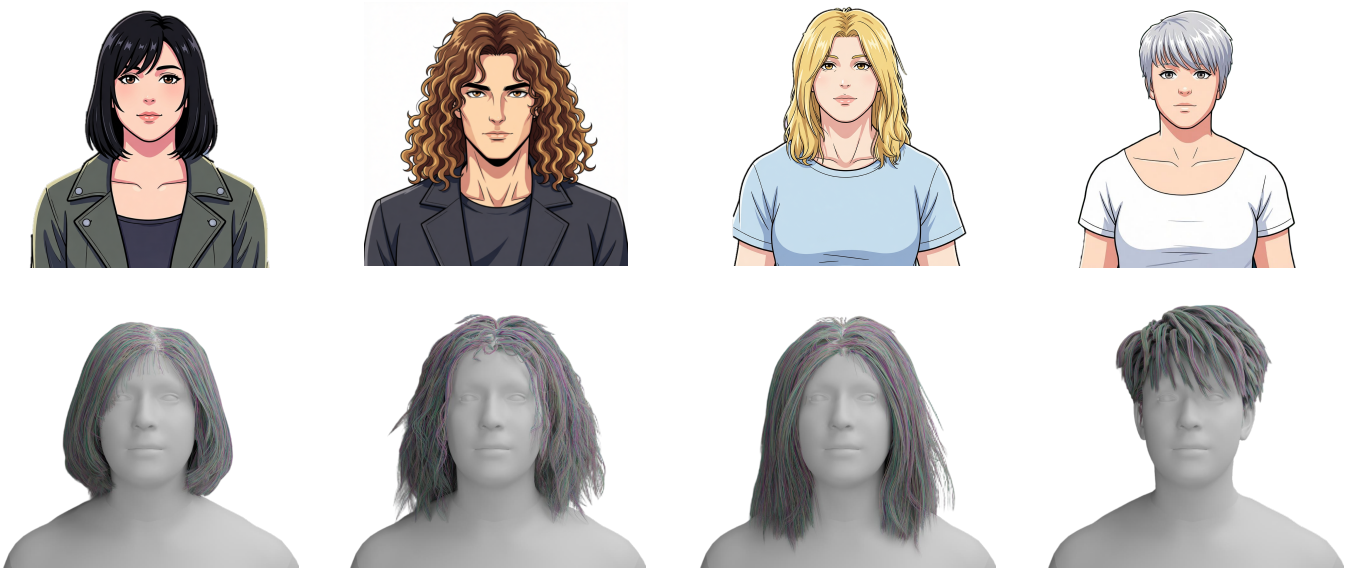


Fig. 12. **Cross-domain adaptation to stylized characters.** Our framework adapts to 2D cartoon inputs via fine-tuning. It generates plausible 3D strand arrangements that faithfully respect the volume and flow of the original anime portraits.

Fig. 13. **Realistic Avatar Creation.** Our model can work in conjunction with the 3D face synthesis model [Zhang et al. 2023] to produce photorealistic avatars with unified visual aesthetics.

are currently obtained through a simple interpolation procedure, and the final visual quality may therefore be affected by the interpolation algorithm itself. A natural future direction is to combine our autoregressive guide-strand representation with a dedicated downstream refinement model for dense strand synthesis. In addition, autoregressive inference (30–60 seconds) remains slower than diffusion-based alternatives because strand-level generation requires long token sequences. Improving efficiency through more compact representations or more parallel generation strategies remains an important future direction.

Discussion. We believe the significance of HairGPT lies not only in introducing a new model for hairstyle synthesis, but more importantly in advancing a new generative paradigm for hair. Rather than treating hairstyle as a single entangled field to be synthesized holistically, our framework represents it as an explicit, structured, and semantically grounded composition of strands and formulates generation as an autoregressive construction over layout, coarse structure, and fine style. This makes the generative process more transparent, more aligned with the structured logic of hairstyle authoring, and better suited for translating high-level intent into coherent 3D geometry.

More importantly, this strand-centric formulation opens a broad space of new problems beyond static generation. Because strands remain explicit throughout the pipeline, the same representation naturally supports semantic editing, topology transfer, hairstyle completion, sparse-to-dense grooming, personalized avatar creation,

Fig. 14. **Editing.** Our dual-decoupled representation and vision-language model naturally facilitate diverse editing applications with either image or text prompts.

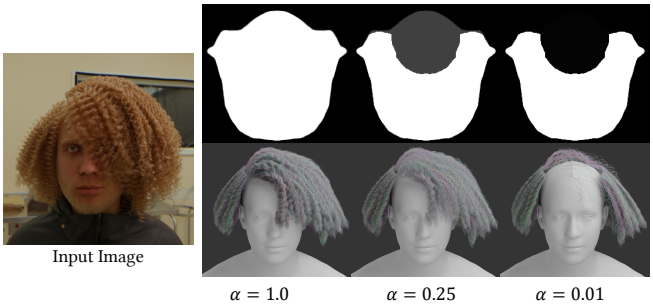


Fig. 15. By tuning the scaling factor α, we can continuously control the hair density in the top region.

and simulation-aware generation. We therefore view HairGPT not as an endpoint, but as an initial foundation for a broader family of structured hair generation systems.

Looking forward, we believe this formulation also provides a promising foundation for more agentic hairstyle generation. Rather than producing all strands in a single pass, future systems could iteratively plan, edit, and verify hairstyle structure across semantic scalp regions and generative stages, using high-level goals to guide global layout before refining local strand style. Such an agentic formulation would move beyond one-shot generation toward interactive and self-refining 3D hair authoring. We hope this work can motivate future research on compositional, artist-aligned, cognitively meaningful, and ultimately agentic generative models for complex 3D content.

7 Conclusion

We proposed HairGPT, a strand-centric framework that models realistic 3D hairstyles through dual-decoupled autoregressive generation. At its core is the dual-decoupled hairstyle representation, accompanied by a carefully designed geometric tokenizer that enables effective strand discretization. HairGPT transforms hair generation from opaque texture synthesis into a transparent, structured, and semantically meaningful process. Experiments demonstrate that this framework supports robust semantic conditioning and compositional editing, enabling the high-fidelity generation of rare, complex hairstyles and effective downstream adaptation to stylized domains.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant W2431046, Central Guided Local Science and Technology Foundation of China YDZX20253100001001, and by MoE Key Lab of Intelligent Perception and Human-Machine Collaboration (ShanghaiTech University), the Shanghai Frontiers Science Center of Human-centered Artificial Intelligence. This work was also supported by the HPC Platform of ShanghaiTech University.

The authors would also like to thank Heng'an Zhou from ShanghaiTech University for his assistance with the supplementary video, and Zijun Zhao from Deemos Technology Co., Ltd. for helping to process part of the raw hairstyle data.

References

Anonymous. 2025. Argus: Large Language Models as General 3D Mesh Generators. arXiv preprint.

Richard L. Bishop. 1975. There is More Than One Way to Frame a Curve. *The American Mathematical Monthly* 82, 3 (1975), 246–251.

Menglei Chai, Tianjia Shao, Hongzhi Wu, Yanlin Weng, and Kun Zhou. 2016. AutoHair: fully automatic hair modeling from a single image. *ACM Trans. Graph.* 35, 4, Article 116 (jul 2016), 12 pages. doi:10.1145/2897824.2925961

Lieu-Hen Chen, Santi Saeyor, Hiroshi Dohi, and Mitsuru Ishizuka. 1999. A system of 3d hair style synthesis based on the wisp model. *The Visual Computer* 15 (1999), 159–170.

Yunlu Chen, Francisco Vicente Carrasco, Christian Häne, Giljoo Nam, Jean-Charles Bazin, and Fernando De la Torre. 2024a. Doubly Hierarchical Geometric Representations for Strand-based Human Hairstyle Generation. Advances in Neural Information Processing Systems, 89728–89751 pages. doi:10.52202/079017-2849

Yiwen Chen, Tong He, Di Huang, Weicai Ye, Sijin Chen, Jiaxiang Tang, Xin Chen, Zhongang Cai, Lei Yang, Gang Yu, et al. 2024b. MeshAnything: Artist-Created Mesh Generation with Autoregressive Transformers. arXiv preprint arXiv:2406.10163.

Byoungwon Choe and Hyeong-Seok Ko. 2005. A statistical wisp model and pseudophysical approaches for interactive hairstyle generation. *IEEE Transactions on Visualization and Computer Graphics* 11, 2 (2005), 160–170.

Stéphane Grabli, François X Sillion, Stephen R Marschner, and Jerome E Lengyel. 2002. Image-based hair capture by inverse lighting. Proceedings of Graphics Interface (GI), 51–58 pages.

Chengan He, Xin Sun, Zhixin Shu, Fujun Luan, Sören Pirk, Jorge Alejandro Amador Herrera, Dominik L. Michels, Tuanfeng Y. Wang, Meng Zhang, Holly Rushmeier, and Yi Zhou. 2025a. Perm: A Parametric Representation for Multi-Style 3D Hair Modeling. International Conference on Learning Representations.

Yuze He, Yanning Zhou, Wang Zhao, Jingwen Ye, Yushi Bai, Kaiwen Xiao, Yong-Jin Liu, Zhongqian Sun, and Wei Yang. 2025b. CHARM: Control-point-based 3D Anime Hairstyle Auto-Regressive Modeling. Proceedings of the SIGGRAPH Asia 2025 Conference Papers.

Tomas Lay Herrera, Arno Zinke, and Andreas Weber. 2012. Lighting hair from the inside: a thermal approach to hair reconstruction. *ACM Trans. Graph.* 31, 6, Article 146 (nov 2012), 9 pages. doi:10.1145/2366145.2366165

Liwen Hu, Chongyang Ma, Linjie Luo, and Hao Li. 2014. Robust hair capture using simulated examples. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–10.

Liwen Hu, Chongyang Ma, Linjie Luo, and Hao Li. 2015. Single-view hair modeling using a hairstyle database. *ACM Transactions on Graphics (ToG)* 34, 4 (2015), 1–9.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. ACM Transactions on Graphics.

Tae-Yong Kim and Ulrich Neumann. 2002. Interactive multiresolution hair modeling and editing. *ACM Transactions on Graphics (TOG)* 21, 3 (2002), 620–629.

Chuan Koon Koh and Zhiyong Huang. 2000. Real-time animation of human hair modeled in strips. Computer Animation and Simulation 2000: Proceedings of the Eurographics Workshop in Interlaken, Switzerland, August 21–22, 2000, 101–110 pages.

Waiming Kong and Masayuki Nakajima. 1998. Generation of 3D Hair Model from Multiple Pictures. *The Journal of the Institute of Image Information and Television Engineers* 52, 9 (1998), 1351–1356. doi:10.3169/itej.52.1351

Zhiyi Kuang, Yiyang Chen, Hongbo Fu, Kun Zhou, and Youyi Zheng. 2022. Deepmvshair: Deep hair modeling from sparse views. SIGGRAPH Asia 2022 Conference Papers.

Wenqi Liang and Zhiyong Huang. 2003. An enhanced framework for real-time hair animation. 11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings., 467–471 pages.

Pengyu Long, Zijun Zhao, Min Ouyang, Qingcheng Zhao, Qixuan Zhang, Wei Yang, Lan Xu, and Jingyi Yu. 2025. TANGLED: Generating 3D Hair Strands from Images with Arbitrary Styles and Viewpoints. arXiv preprint arXiv:2502.06392.

H. Luo, A. Chen, Q. Zhang, B. Pang, M. Wu, L. Xu, and J. Yu. 2021. Convolutional Neural Opacity Radiance Fields. In *2021 IEEE International Conference on Computational Photography (ICCP)*. IEEE Computer Society, Los Alamitos, CA, USA, 1–12. doi:10.1109/ICCP51581.2021.9466273

Haimin Luo, Min Ouyang, Zijun Zhao, Suyi Jiang, Longwen Zhang, Qixuan Zhang, Wei Yang, Lan Xu, and Jingyi Yu. 2024. Gaussianhair: Hair modeling and rendering with light-aware gaussians. arXiv preprint arXiv:2402.10483.

Haimin Luo, Teng Xu, Yuheng Jiang, Chenglin Zhou, Qiwei Qiu, Yingliang Zhang, Wei Yang, Lan Xu, and Jingyi Yu. 2022. Artemis: Articulated Neural Pets with Appearance and Motion Synthesis. *ACM Trans. Graph.* 41, 4, Article 164 (jul 2022), 19 pages. doi:10.1145/3528223.3530086

Linjie Luo, Hao Li, Sylvain Paris, Thibaut Weise, Mark Pauly, and Szymon Rusinkiewicz. 2012. Multi-view hair capture using orientation fields. 2012 IEEE Conference on Computer Vision and Pattern Recognition, 1490–1497 pages.

Linjie Luo, Cha Zhang, Zhengyou Zhang, and Szymon Rusinkiewicz. 2013. Wide-baseline hair capture using strand-based refinement. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 265–272 pages.

Givi Meishvili, James Clemoes, Charlie Hewitt, Zafiirah Hosenie, Xiao Xian, Martin de La Gorce, Tibor Takacs, Tadas Baltrušaitis, Antonio Criminisi, Chyna McRae, Nina Jablonski, and Marta Wilczkowiak. 2024. Hairmony: Fairness-aware hairstyle classification. SIGGRAPH Asia 2024 Conference Papers (SA Conference Papers '24), December 3-6, Tokyo, Japan.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.

Charlie Nash, Yaroslav Ganin, SM Ali Eslami, and Peter Battaglia. 2020. Polygen: An autoregressive generative model of 3d meshes. International conference on machine learning, 7220–7229 pages.

Paul Noble and Wen Tang. 2004. Modelling and animating cartoon hair with nurbs surfaces. Proceedings Computer Graphics International, 2004., 60–67 pages.

Yimin Pan, Matthias Nießner, and Tobias Kirschstein. 2025. Hair Strand Reconstruction based on 3D Gaussian Splatting. 36th British Machine Vision Conference 2025, BMVC 2025, Sheffield, UK, November 24-27, 2025. https://bmva-archive.org.uk/bmvc/2025/papers/Paper_1220/paper.pdf

Sylvain Paris, Hector M Briceno, and François X Sillion. 2004. Capture of hair geometry from multiple images. *ACM transactions on graphics (TOG)* 23, 3 (2004), 712–719.

Sylvain Paris, Will Chang, Oleg I Kozhushnyan, Wojciech Jarosz, Wojciech Matusik, Matthias Zwicker, and Frédo Durand. 2008. Hair photobooth: geometric and photometric acquisition of real hairstyles. *ACM Trans. Graph.* 27, 3 (2008), 30.

Deborah Patrick, Shaun Bangay, and Adele Lobb. 2004. Modelling and rendering techniques for african hairstyles. Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa, 115–124 pages.

Radu Alexandru Rosu, Shunsuke Saito, Ziyan Wang, Chenglei Wu, Sven Behnke, and Giljoo Nam. 2022. Neural strands: Learning hair geometry and appearance from multi-view images. European Conference on Computer Vision, 73–89 pages.

Radu Alexandru Rosu, Keyu Wu, Yao Feng, Youyi Zheng, and Michael J Black. 2025. DiffLocks: Generating 3D Hair from a Single Image using Diffusion Models. Proceedings of the Computer Vision and Pattern Recognition Conference, 10847–10857 pages.

Shunsuke Saito, Liwen Hu, Chongyang Ma, Hikaru Ibayashi, Linjie Luo, and Hao Li. 2018. 3D hair synthesis using volumetric variational autoencoders. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–12.

Yuefan Shen, Shunsuke Saito, Ziyan Wang, Olivier Maury, Chenglei Wu, Jessica Hodgins, Youyi Zheng, and Giljoo Nam. 2023. CT2Hair: High-Fidelity 3D Hair Modeling using Computed Tomography. *ACM Transactions on Graphics* 42, 4 (2023), 1–13.

Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. 2024. Meshgpt: Generating triangle meshes with decoder-only transformers. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 19615–19625 pages.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. 2025. DINOv3. arXiv:2508.10104 [cs.CV] https://arxiv.org/abs/2508.10104

Vanessa Sklyarova, Jenya Chelishev, Andreea Dogaru, Igor Medvedev, Victor Lempitsky, and Egor Zakharov. 2023a. Neural haircut: Prior-guided strand-based hair reconstruction. Proceedings of the IEEE/CVF International Conference on Computer Vision, 19762–19773 pages.

Vanessa Sklyarova, Egor Zakharov, Otmar Hilliges, Michael J Black, and Justus Thies. 2023b. Haar: Text-conditioned generative model of 3d strand-based human hairstyles. arXiv preprint arXiv:2312.11666.

Yusuke Takimoto, Hikari Takehara, Hiroyuki Sato, Zihao Zhu, and Bo Zheng. 2024. Dr.Hair: Reconstructing Scalp-Connected Hair Strands without Pre-training via Differentiable Rendering of Line Segments. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Cen Wang, Minye Wu, Ziyu Wang, Liao Wang, Hao Sheng, and Jingyi Yu. 2020. Neural Opacity Point Cloud. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 7 (2020), 1570–1581. doi:10.1109/TPAMI.2020.2986777

Tao Wang and Xue Dong Yang. 2004. Hair design based on the hierarchical cluster hair model. Geometric modeling: techniques, applications, systems and tools, 329–359 pages.

Ziyan Wang, Giljoo Nam, Tuur Stuyck, Stephen Lombardi, Chen Cao, Jason Saragih, Michael Zollhöfer, Jessica Hodgins, and Christoph Lassner. 2023. NeuWigs: A Neural Dynamic Model for Volumetric Hair Capture and Animation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8641–8651 pages.

Ziyan Wang, Giljoo Nam, Tuur Stuyck, Stephen Lombardi, Michael Zollhöfer, Jessica Hodgins, and Christoph Lassner. 2022. Hvh: Learning a hybrid neural volumetric representation for dynamic hair performance capture. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6143–6154 pages.

Yichen Wei, Eyal Ofek, Long Quan, and Heung-Yeung Shum. 2005. Modeling hair from multiple views. ACM SIGGRAPH 2005 Papers, 816–820 pages.

Keyu Wu, Lingchen Yang, Zhiyi Kuang, Yao Feng, Xutao Han, Yuefan Shen, Hongbo Fu, Kun Zhou, and Youyi Zheng. 2024. MonoHair: High-Fidelity Hair Modeling from a Monocular Video. arXiv preprint arXiv:2403.18356.

Keyu Wu, Yifan Ye, Lingchen Yang, Hongbo Fu, Kun Zhou, and Youyi Zheng. 2022. Neuralhdhair: Automatic high-fidelity hair modeling from a single image using implicit neural representations. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1526–1535 pages.

Zhan Xu and Xue Dong Yang. 2001. V-hairstudio: an interactive tool for hair design. *IEEE Computer Graphics and Applications* 21, 3 (2001), 36–43.

Lingchen Yang, Zefeng Shi, Youyi Zheng, and Kun Zhou. 2019. Dynamic hair modeling from monocular videos using deep neural networks. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–12.

Xue Dong Yang, Zhan Xu, Jun Yang, and Tao Wang. 2000. The cluster hair model. *Graphical Models* 62, 2 (2000), 85–103.

Xuan Yu, Zhan Yu, Xiaogang Chen, and Jingyi Yu. 2014. A hybrid image-cad based system for modeling realistic hairstyles. Proceedings of the 18th meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, 63–70 pages.

Cem Yuksel, Scott Schaefer, and John Keyser. 2009. Hair meshes. *ACM Transactions on Graphics (TOG)* 28, 5 (2009), 1–7.

Egor Zakharov, Vanessa Sklyarova, Michael Black, Giljoo Nam, Justus Thies, and Otmar Hilliges. 2024. Human hair reconstruction with strand-aligned 3d gaussians. European Conference on Computer Vision, 409–425 pages.

Longwen Zhang, Qiwei Qiu, Hongyang Lin, Qixuan Zhang, Cheng Shi, Wei Yang, Ye Shi, Sibei Yang, Lan Xu, and Jingyi Yu. 2023. DreamFace: Progressive Generation of Animatable 3D Faces under Text Guidance. *ACM Trans. Graph.* 42, 4, Article 138 (jul 2023), 16 pages. doi:10.1145/3592094

Meng Zhang, Menglei Chai, Hongzhi Wu, Hao Yang, and Kun Zhou. 2017a. A data-driven approach to four-view image-based hair modeling. *ACM Trans. Graph.* 36, 4 (2017), 156–1.

Meng Zhang, Menglei Chai, Hongzhi Wu, Hao Yang, and Kun Zhou. 2017b. A data-driven approach to four-view image-based hair modeling. *ACM Trans. Graph.* 36, 4, Article 156 (jul 2017), 11 pages. doi:10.1145/3072959.3073627

Yang Zheng, Menglei Chai, Delio Vicini, Yuxiao Zhou, Yinghao Xu, Leonidas Guibas, Gordon Wetzstein, and Thabo Beeler. 2025. GroomLight: Hybrid Inverse Rendering for Relightable Human Hair Appearance Modeling. Proceedings of the Computer Vision and Pattern Recognition Conference, 16040–16050 pages.

Yujian Zheng, Zirong Jin, Moran Li, Haibin Huang, Chongyang Ma, Shuguang Cui, and Xiaoguang Han. 2023. Hairstep: Transfer synthetic to real using strand and depth maps for single-view 3d hair modeling. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12726–12735 pages.

Yuxiao Zhou, Menglei Chai, Alessandro Pepe, Markus Gross, and Thabo Beeler. 2023. Groomgen: A high-quality generative hair model using hierarchical latent representations. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–16.

Yuxiao Zhou, Menglei Chai, Daoye Wang, Sebastian Winberg, Erroll Wood, Kripasindhu Sarkar, Markus Gross, and Thabo Beeler. 2024. Groomcap: High-fidelity prior-free hair capture. *ACM Transactions on Graphics (TOG)* 43, 6 (2024), 1–15.

Yi Zhou, Liwen Hu, Jun Xing, Weikai Chen, Han-Wei Kung, Xin Tong, and Hao Li. 2018. Hairnet: Single-view hair reconstruction using convolutional neural networks. Proceedings of the European Conference on Computer Vision (ECCV), 235–251 pages.

A Architecture Details of Strand Tokenizer

The strand tokenizer is implemented as a Vector Quantized Variational Autoencoder (VQ-VAE) tailored for sequential 1D geometric data. The detailed architecture is described below.

A.1 Network Structure

Given that hair strands are represented as ordered sequences of 3D coordinates, we leverage 1D Convolutional Neural Networks (1D-CNNs) as the backbone.

- **Encoder:** The encoder transforms the input strand geometry $\mathbf{x} \in \mathbb{R}^{N\times 3}$ into a latent representation. It consists of stacked 1D convolutional blocks. To ensure training stability, we apply Weight Normalization across layers, followed by non-linear activations (e.g., LeakyReLU). The network progressively reduces the temporal resolution, resulting in a latent tensor $\mathbf{z}_e \in \mathbb{R}^{4\times 8}$.
- **Decoder:** The decoder mirrors the encoder's architecture. It utilizes transposed 1D convolutions to upsample the quantized latent codes back to the original spatial resolution N, reconstructing the strand geometry $\hat{\mathbf{x}}$.

A.2 Multi-Head Vector Quantization

To balance codebook usage and reconstruction fidelity, we employ a Multi-Head Vector Quantizer (Product Quantization) mechanism.

- **Subspace Decomposition:** The latent dimension D is naturally set to 8, and the tokenizer contains 4 heads.
- **Independent Codebooks:** We maintain 4 separate codebooks $\{C_1, \ldots, C_4\}$. Each codebook C_m contains K entries. Quantization for the m-th head is performed by finding the nearest neighbor in C_m. We use cosine similarity as the distance metric, as it provides higher codebook utilization than standard Euclidean distance in our experiments. We use 8192 entries for the coarse-shape VQ-VAE and 2048 entries for the style-residual VQ-VAE.
- **Training Stability:** The model is trained using a combination of reconstruction loss and commitment loss. To further improve stability:
  (1) **EMA Updates:** For codebook updates, we use Exponential Moving Average (EMA), which smooths the learning dynamics.
  (2) **Noise Injection:** We inject scaled random noise into the latent codes during training: $\mathbf{z}_{noisy} = \mathbf{z}_e + \epsilon$, where $\epsilon \propto \text{std}(\mathbf{z}_e)$. This noise injection encourages the encoder to generate robust representations and prevents the quantizer from relying on precise floating-point artifacts.

A.3 Efficient Training via Cluster-based Sampling

To facilitate efficient training and handle the intractable scale of the original synthetic datasets (typically containing $\sim 10^5$ strands per model), we adopt a stratified sampling approach based on our clustering results.

Following the frequency-aware clustering described in the main paper, which partitions the hair volume into $N_{\text{guide}} = 512$ clusters, we do not train on the full dense geometry. Instead, for each cluster, we randomly sample 10 individual strands as training representatives. This sampling strategy allows the model to learn the local geometric variance within each cluster while reducing the total number of strands per hairstyle to a manageable size (5, 120 strands). This approach ensures a balanced representation of all semantic regions of the scalp and maintains the structural richness of the original hairstyle with significantly lower computational overhead.

B Data Details

B.1 Text Annotation Example.

To illustrate the granularity of our data engine, we provide a representative example of the textual annotations associated with a 3D hairstyle. Our annotations consist of two components: structured hierarchical attributes and a synthesized natural language description.

Structured Attributes. The metadata is organized into global hairstyle characteristics and region-specific geometric properties for $M = 8$ scalp regions.

- **Global Attributes:**
  - *Parting:* Center; *Bangs:* None; *Surface:* Shiny.
  - *Hairline:* Straight shape, Medium position, Full visibility.
  - *Others:* No accessories, No baby hair, No attribute variation.
- **Scalp Region Details:**
  Front, Sides, Temples: Straight hair type, medium thickness, directed **downward**, chin length. Layering is textured (except at temples).
  Top, Crown: Straight hair type, medium thickness, directed **upward**, chin length, textured layering.
  Nape: Straight hair type, medium thickness, directed **downward**, **short** length, textured layering.

Synthesized Natural Language Description. The domain-adapted VLM converts the structured attributes above into a coherent, semantically rich paragraph used for cross-modal training:

> *"The hairstyle features no bangs, no accessories, a center parting, a straight medium-positioned hairline with full visibility, a shiny surface, no baby hair, and consistent attributes throughout. The hair is straight with medium thickness, directed downward in the front, right side, left side, nape, and right and left temples, and upward in the top and crown. Lengths reach the chin in most areas (front, top, crown, right side, right temple, left temple, left side) and are short at the nape. Layering is textured in the front, top, crown, right side, and left side, while the right and left temples have no layering."*

This hierarchical annotation ensures that HairGPT learns to associate specific linguistic tokens with localized geometric structures, such as the upward growth flow at the crown versus the downward flow at the nape.

B.2 Multimodal Annotation and Augmentation Pipeline

To construct the high-quality aligned triplets $(\mathcal{I}, \mathcal{T}, \mathcal{H})$ required for training *HairGPT*, we developed a multi-stage data engine. This pipeline performs hierarchical semantic labeling, natural language normalization, and generative visual augmentation to bridge the gap between synthetic geometry and real-world diversity.

Phase 1: Hierarchical Semantic Labeling. We first extract fine-grained attributes from 3D hair models using a domain-adapted Qwen2.5-VL-7B-Instruct model. As shown in Listing 1, we employ a strict **Hairstyle Taxonomy Prompt** that enforces a structured JSON output. This taxonomy covers 10 global attributes (e.g., parting type, hairline shape) and 8 localized geometric parameters for each of the $M = 8$ scalp regions. By restricting the VLM to a predefined candidate set, we ensure taxonomic consistency across the dataset.

Phase 2: Natural Language Normalization. To convert structured JSON metadata into natural-language descriptions suitable for transformer training, we use a **Normalization System Prompt**. The model is instructed to synthesize a coherent narrative starting with *"The hairstyle features..."*, prioritizing global style before detailing regional specifics. This process serializes spatial attributes into a semantic modality that is rich in contextual information.

Phase 3: Contextual and Stylized Augmentation. To improve robustness, we implement a generative augmentation pipeline. Using the extracted attributes, we systematically vary 13 hair colors, 12 subject identities, 11 clothing types, and 12 backgrounds to synthesize augmented images.

- **Photorealistic Branch:** Focused on creating diverse real-world contexts (e.g., *cyberpunk street*, *modern office*) to prevent overfitting to synthetic renders.
- **Stylized Branch (Anime):** Aimed at cross-domain generalization. We utilize specific prompts for *cel-shading* and *2D flat colors* to create anime counterparts while preserving the exact 3D hair topology (see Listing 2).

Listing 1. Fragment of the Hairstyle Taxonomy Prompt.

```
Analyze the hairstyle... Fill each field using only values from:
Global Attributes:
- Bangs Style: [None, Straight, V-shape, Inverted U, ...]
- Parting: [Center, Right, Left, Zig-Zag, ...]
Scalp Regions (Front, Top, Crown, Nape, ...):
- Hair Type: [Coily, Curly, Wavy, Straight]
- Direction: [Down, Side, Up, Out]
- Length: [Very Short, Ear, Chin, Shoulder, ...]
Return ONLY JSON:
{ "Global Attributes": {...}, "Scalp Regions": {"Front": {...}, ...} }
```

Listing 2. Image Augmentation Prompt Templates.

```
# Realistic Photo Template:
"A realistic photo of {subject} {clothing}, {background}. The person has {
      hair_color}. Keep the exact same hairstyle and structure as the input
      image. High quality, photorealistic."

# Anime Style Template:
"High quality anime style illustration of {subject} {clothing}, {background}. The
       character has {hair_color}. Keep the exact same hairstyle and structure
      as the input image. Masterpiece, cel shading, flat color."
```

C HairGPT Implementation Details

C.1 HairGPT Architecture Details

The architecture of *HairGPT* is designed to unify linguistic, visual, and geometric modalities into a single autoregressive framework. The following details describe the backbone, multimodal projectors, and the discrete strand tokenizer.

Transformer Backbone. We use the *Qwen3-1.7B* decoder-only Transformer as the core generative backbone of HairGPT. The model operates on a hidden dimension of $D = 2048$ and consists of standard self-attention layers and feed-forward networks. To accommodate hairstyle synthesis, the original linguistic vocabulary is extended into a unified vocabulary $\mathcal{V}$ that incorporates discrete geometric tokens via specific ID offsets.

Multimodal Condition Encoders. To bridge the gap between continuous visual/textual features and the discrete LLM space, we employ specialized encoders and projection layers:

- **Vision Encoder:** We leverage a frozen *DINOv3* (ViT-L/16) backbone to extract high-level visual features. For an input image $\mathcal{I} \in \mathbb{R}^{512\times512\times3}$, the encoder produces patch-level tokens. These are mapped to the Transformer's hidden dimension using a learnable lightweight MLP projector, $\mathcal{P}_{\text{img}}$, resulting in a sequence of visual embeddings $\mathbf{E}_{\text{img}}$.
- **Text Projector:** While we retain the pretrained word embeddings from Qwen, we apply a learnable linear projector $\mathcal{P}_{\text{txt}}$ to align the text features with the shared multimodal latent space, ensuring consistent conditioning across global and regional instructions.

C.2 Training Details

Cluster-based Token Augmentation. To enhance the model's generalization and prevent it from overfitting to specific geometric instances, we implement a cluster-based strand token sampling strategy as a primary data augmentation technique. During training, instead of utilizing a fixed representative centroid for each of the $N_{\text{guide}} = 512$ clusters, we dynamically sample a strand from the corresponding cluster pool (as described in Sec. A.3). This stochastic selection ensures that the autoregressive transformer encounters varied geometric realizations of the same topological structure across different iterations. By introducing this instance-level variability, we effectively regularize the latent space, forcing the model to learn robust structural relationships and growth priors rather than memorizing individual strand coordinates.

Mode Selection Probability. To ensure the Transformer backbone Φ learns each hierarchical level of the hairstyle representation with equal proficiency, we adopt a mode-specific training scheme. During each training iteration, for every aligned triplet $(\mathcal{I}, \mathcal{T}, \mathcal{H})$, we randomly sample a generation mode to construct the input sequence $\mathbf{S}^{(\cdot)}$. We apply a uniform probability distribution for mode selection: $P(\text{Layout}) = 1/3$, $P(\text{Coarse}) = 1/3$, and $P(\text{Style}) = 1/3$. This balanced sampling prevents the model from overfitting to any single stage of the geometric synthesis and ensures a stable gradient flow across the entire generative hierarchy.

Multimodal Condition Dropout. To enhance the model's robustness to missing or noisy inputs, we implement a multimodal dropout strategy during training. Conditioning signals are randomly replaced with a learnable null embedding $\emptyset$ according to the following probabilities:

- **Image Dropout:** The visual context $\mathbf{E}_{\text{img}}$ is dropped with a probability of $p_{\text{img}} = 0.3$.
- **Text Dropout:** The linguistic prompts $\mathbf{E}_{\text{txt}}$ (both global and regional) are dropped with a probability of $p_{\text{txt}} = 0.3$.
- **Joint Unconditional Training:** With a probability of $p_{\text{null}} = 0.1$, both image and text conditions are dropped simultaneously, forcing the model to perform unconditional generation based purely on the hairstyle distribution priors.

When a modality is dropped, its corresponding attention mask is set to zero to completely isolate the influence of the input. This strategy enables users to steer the generation process during inference by adjusting the guidance scale between conditional and unconditional logits.

Training Details and Hyperparameters. We train *HairGPT* using 32 NVIDIA H20 GPUs for approximately 24 hours. The Transformer backbone is initialized from the pretrained `Qwen3-1.7B` weights to leverage its linguistic priors. Such initialization preserves pre-existing knowledge, enabling zero-shot understanding of fine-grained text attributes (e.g., "curly" and "afro"). For optimization, we employ the AdamW optimizer with a weight decay of 0.1. The learning rate follows a cosine decay schedule, starting from a base learning rate of 5×10^{-5} and annealing to a final value of 1×10^{-6}. Notably, to facilitate rapid alignment between the geometric and multimodal latent spaces, we apply a 10× learning rate multiplier (5×10^{-4}) specifically to the vision and text projectors, while the Transformer backbone parameters are optimized using the base learning rate.

C.3 Inference Details

Hairstyle inference in *HairGPT* is modeled as a progressive hierarchical unfolding process, transforming global multimodal conditions $C = \{\mathcal{I}, \mathcal{T}\}$ into localized geometric details. Following the dual-decoupled parameterization, the synthesis is executed through a multi-pass generation flow to ensure rigorous structural control.

Phased Autoregressive Generation. The inference process is partitioned into three distinct functional phases:

- **Density Phase:** The model first generates a compressed sequence of density tokens **D**. This establishes the macroscopic hair distribution and occupancy plan on the 2D scalp manifold.
- **Layout Phase:** Guided by the density prior and task separator s_1, the model autoregressively predicts spatial anchors (u_k, v_k) for each of the $M = 8$ semantic regions. This phase defines the precise location of every guide strand.
- **Coarse and Style Phases:** Once strand placements are fixed, the model enters the geometric refinement stages. Here, the transformer acts as a coordinate-conditioned mapper. By injecting the previously generated (u_k, v_k) as a spatial prefix alongside mode separators s_2 or s_3, the model predicts the corresponding low-frequency backbones T_{coa} and high-frequency residuals T_{sty}.

This "position-first, geometry-after" strategy ensures that complex geometric details are strictly grounded to the established physical layout, preventing spatial drift in long-sequence generation.

Task Steering via Separators. To manage multiple synthesis objectives within a unified transformer backbone, we utilize explicit task separators $\{s_1, s_2, s_3\}$ as functional switches. In our implementation, these separators serve as tokens that steer the model's internal attention to focus on specific sub-tasks: s_1 for layout planning, s_2 for topological backbone synthesis, and s_3 for fine-grained textural detail.

Compositional Editing. The decoupled nature of our inference process naturally enables flexible downstream applications. By strategically re-injecting prefixes and task separators, users can perform **compositional editing** without regenerating the entire hairstyle. For instance, by keeping the spatial anchors and coarse backbones constant and only re-sampling tokens following the s_3 separator, *HairGPT* can perform style transfer or texture refinement (e.g., converting straight hair to coily hair) while preserving the original hair volume and global flow.