HairFree: Compositional 2D Head Prior for Text-Driven 360° Bald Texture Synthesis

Mirela Ostrek[1,2] **Michael J. Black**[1] **Justus Thies**[1,2]

[1]Max Planck Institute for Intelligent Systems [2]Technical University of Darmstadt

Abstract

Synthesizing high-quality 3D head textures is crucial for gaming, virtual reality, and digital humans. Achieving seamless 360° textures typically requires expensive multi-view datasets with precise tracking. However, traditional methods struggle without back-view data or precise geometry, especially for human heads, where even minor inconsistencies disrupt realism. We introduce *HairFree*, an unsupervised texturing framework guided by textual descriptions and 2D diffusion priors, producing high-consistency 360° bald head textures—including non-human skin with fine details—***without any texture, back-view, bald, non-human, or synthetic training data***. We fine-tune a diffusion prior on a dataset of mostly frontal faces, conditioned on predicted 3D head geometry and face parsing. During inference, *HairFree* uses precise skin masks and 3D FLAME geometry as input conditioning, ensuring high 3D consistency and alignment. We synthesize the full 360° texture by first generating a frontal RGB image aligned to the 3D FLAME pose and mapping it to UV space. As the virtual camera moves, we inpaint and merge missing regions. A built-in semantic prior enables precise region separation—particularly for isolating and removing hair—allowing seamless integration with various assets like customizable 3D hair, eyeglasses, jewelry, etc. We evaluate *HairFree* quantitatively and qualitatively, demonstrating its superiority over state-of-the-art 3D head avatar generation methods. https://hairfree.is.tue.mpg.de/

1 Introduction

Generating realistic and consistent textured 3D head avatars is essential for applications in gaming, virtual reality, and digital human modeling. Achieving 360° appearance consistency in head texturing is a persistent challenge, especially, because scalp visibility varies dramatically with dynamic hair animations and different hairline shapes (e.g., straight, rounded, widow's peak, M- or V-shaped, receding), see Figure 1. Existing methods [15, 16, 17, 14, 28, 30, 40, 31, 34, 64, 2, 70] either bake hair (and its specific hairline) directly into the texture—leading to visible artifacts when swapping to any other hairstyle—or limit textures to the facial region, neglecting the full scalp. Moreover, other approaches rely on large-scale multi-view datasets of human heads with precise tracking, making them resource-intensive and less flexible.

This lack of flexibility undermines true compositionality, where any 3D hairstyle or other assets (hats, helmets, eyeglasses, jewelry) can be seamlessly added, swapped, or animated without visible seams or mismatches. For practical applications, a fully disentangled, hair-free scalp and face texture is crucial, serving as a neutral base for compositional 3D layering. In this paper, we introduce *HairFree*, an unsupervised generative texturing framework that generates high-quality, 360° head textures, providing a fully bald, neutral base for 3D asset integration. Unlike existing methods, *HairFree* uses a diffusion-based inpainting approach guided by textual descriptions and produces consistent, detailed head textures without relying on any texture, back-view, bald, non-human, or synthetic training

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Figure 1: ***HairFree*** is a hybrid 2D/3D neural rendering method that synthesizes diverse, high-fidelity 360° bald human head textures given *a 3D head mesh* and *a textual description*. The generated bald textures (columns 1, 4, 7) seamlessly integrate into classical graphics pipelines, allowing compatibility with any 3D hairstyle—regardless of hairline type (in green)—in a fully compositional manner.

data. Our method is based on a compositional 2D human head prior, trained on 100K mostly frontal images of faces, conditioned on RGB background, 3D head geometry, and partial facial semantics. The 3D head geometry and facial semantics are estimated using off-the-shelf methods (Spectre [12], FPM [68]).

At inference time, *HairFree* begins by generating a high-quality frontal view image of a face, conditioned on accurate 3D FLAME geometry [35] and precise skin masks. This initial image is projected onto the 3D FLAME texture space, creating a base texture. As the virtual camera moves around the head, a progressive inpainting process is applied—rendering the visible regions and filling in missing areas using the diffusion model in image space. These completed regions are then re-mapped to texture space, maintaining texture consistency. The entire process is guided by the same 3D-consistent FLAME geometry and skin masks, ensuring precise alignment across all views. The semantic conditioning allows for skin-hair separation, effectively isolating and removing hair, producing a clean bald texture. The final result is a complete 360° bald head texture, fully compatible with various 3D assets, such as strand-based hairstyles (Figure 1).

Our results demonstrate that *HairFree* generates high-quality, consistent, and detailed 360° textures for fully bald heads, including both human skin (Figure 4) and non-human skin textures (Figure 5). This opens up new possibilities for creating customizable 3D head avatars using textual descriptions, without the need for extensive multi-view datasets or supervised training. Finally, we evaluate *HairFree* both quantitatively and qualitatively, showing its superiority over state-of-the-art methods.

In summary, we make the following contributions:

- *Compositional 2D head prior:* a diffusion model that generates high-quality, photorealistic face images, conditioned on 3D geometry, facial semantics, and text prompts, enabling precise separation of hair and skin regions.
- *360° bald head texturing pipeline:* a robust, unsupervised texturing method that synthesizes consistent, high-quality 360° bald head textures from text prompts. The pipeline leverages our compositional 2D head prior in combination with a generic inpainting prior. This method integrates image-space inpainting with texture mapping onto a FLAME-based head mesh in UV space. It generates full head textures without relying on any texture, back-view, bald, non-human, or synthetic data, and generalizes to both human and non-human skin types.
- *A dataset* of 1,000 generated, high-quality, photorealistic human head textures, providing a diverse and scalable resource for realistic avatar and face model development.

2 Related Work

Related work spans three areas: 2D image synthesis, mesh texturing, and 3D character generation. While GAN- and diffusion-based models excel at face image creation and 3DMM or reconstruction methods yield plausible head textures, none offer a compositional 2D prior that cleanly separates hair and skin for full-head texturing on arbitrary meshes. Below, we review key advances in each area and show how our approach fills this gap.

2D Image Synthesis: Generative Adversarial Networks (GANs) [19] and their StyleGAN successors [25, 23, 24, 3] have set the standard for photorealistic image generation across objects and human faces [61, 62, 32, 56], with StyleAvatar extending StyleGAN to texture maps for 3DMM-based avatars [65, 63]. More recently, diffusion models [48, 52, 51, 1, 55, 45, 44] trained on LAION-5B [57] surpass GANs in quality and diversity, powering robust text-to-image synthesis [54] and supporting fine-grained control through arbitrary image-conditioning—landmarks, segmentation masks, depth maps, or rendered geometry—via ControlNet [71]. We fine-tune Stable Diffusion with ControlNet conditioning to provide precise, identity-preserving guidance for our avatar texturing pipeline.

Texturing and Face Textures: 3D morphable models (3DMMs) [10], such as the Basel Face Model [47], use PCA on textured scans to represent facial geometry and texture, becoming standard for face tracking [72] and neural rendering techniques like NeRF [43] and Gaussian splatting [27]. However, their texture spaces lack diversity due to limited 3D data used to create it. Methods like FlameTex [11], Slossberg et al. [59], Gecer et al.[16, 17], and CLIPFace [2] expand texture variety using in-the-wild images. CLIPFace uses the FLAME model [35] with a StyleGAN-like architecture for high-quality textures. DreamFace [70] uses CLIP-based selection for coarse geometry, then refines details with Score Distillation Sampling (SDS), combining generic and texture latent diffusion models to generate diverse, high-quality frontal textures of the facial region. FitMe [29] (GAN inversion), Relightify [46] (diffusion), and Luo et al. [40] (StyleGAN) reconstruct photorealistic facial textures directly from images but also cover only faces, not full heads, same as UV-IDM [33]. Most of these methods bake at least some hair into the texture, preventing compositional layering with separate hair assets.

3D Character Generation: Several recent works generate 3D textures and geometry via "generation by reconstruction," synthesizing multiple 2D views and then lifting them into 3D [37, 49, 4, 42, 53, 13, 7]. General-purpose methods like TEXTure [53], Text2Tex [6], and SceneTex [5] use diffusion-based priors in a generate-then-refine pipeline to texture arbitrary objects and scenes. Human-specific approaches include TADA [36] and HumanNorm [21], which recover full body assets via DMTet [58] and SDS optimization [49], and FaceLift [41], which directly predicts multi-view images with a latent diffusion model before reconstructing with Gaussian splatting. TECA [69] further introduces mesh-volumetric disentanglement of skin and hair under an SDS loss. Arc2Avatar [18] produces high-quality 3D heads from single images by leveraging a human face foundation model and full 3DMM integration for superior realism and identity preservation. None of these methods are designed to generate specifically bald textures or fully disentangle hair. In contrast, our compositional 2D diffusion prior disentangles hair from skin in a single-stage process, yielding fully editable, high-fidelity 360° head textures.

3 Preliminaries: Diffusion Models

Our 2D human head prior is a latent diffusion model (LDM) [54], fine-tuned to transform rendered head meshes, partial face parsing masks, and RGB backgrounds into photorealistic human heads.

Denoising Diffusion Probabilistic Model (DDPM): In the DDPM framework [20], noise-corrupted samples are progressively denoised over T timesteps. The forward process adds Gaussian noise $\mathcal{N}(\mathbf{0}, \mathbf{I})$ in T steps using a variance schedule $\{\beta_t\}$:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}). \tag{1}$$

The reverse denoising process, parameterized by ϵ_θ, approximates $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)), \tag{2}$$

with an objective function:

$$\mathcal{L}_{\text{DDPM}} = \mathbb{E}_{\mathbf{x}_0, \epsilon\sim\mathcal{N}(0,1), t}\Big[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|_2^2\Big]. \tag{3}$$

Latent Diffusion Model (LDM): Operating in the latent space, LDMs [54] use a pre-trained VAE to map images to latent codes, reformulating the objective as:

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{\mathcal{E}(\mathbf{x}), \epsilon\sim\mathcal{N}(0,1), t}\Big[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, t)\|_2^2\Big], \tag{4}$$

where $\mathbf{z}_t$ is the latent code at timestep t.

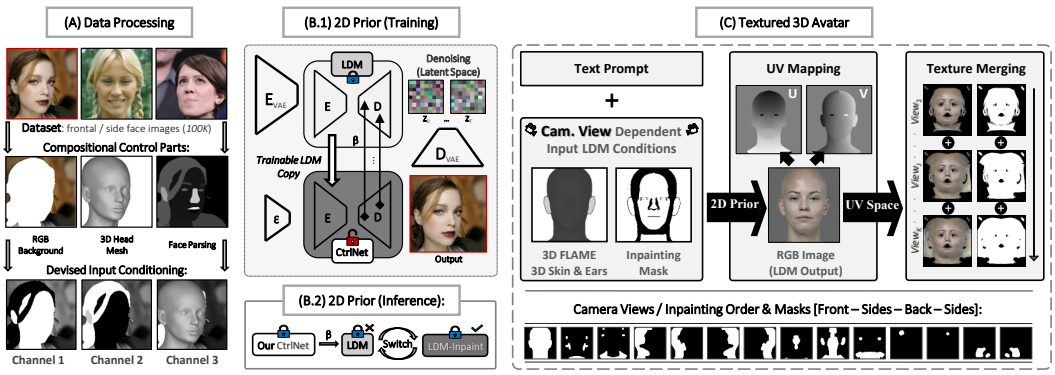

Figure 2: **System Overview:** (A) *Data Processing:* Estimate face parsing, 3D head mesh, and remove the foreground to form compositional inputs. (B.1) *Training Prior:* Fine-tune an LDM via ControlNet using these inputs and a generic "face" prompt. (B.2) *Inference Prior:* Swap in a generic LDM-Inpainting prior. (C) *Texture Generation:* Generate a frontal view, map to UV space, iteratively render "seen" and inpaint "unseen" regions while moving the camera, building a full 360° texture.

4 Method

Figure 2 outlines our system. We first train a compositional 2D head prior via latent diffusion on predominantly frontal face images, conditioned on rendered FLAME meshes, face parsing masks, and backgrounds. At inference, we use the FLAME model to condition the generation of a frontal view, project it into UV space, and then iteratively fill in unseen texture regions. The following sections describe (1) our diffusion prior and its training and (2) the progressive texturing process.

4.1 Compositional 2D Human Head Prior

We fine-tune a Latent Diffusion Model (LDM) [54] using ControlNet [71] to generate high-quality images conditioned on specific input features. ControlNet operates as a *trainable copy* of the base diffusion model (Stable Diffusion 2.1 [54]), which is kept frozen in *locked mode* during training. This ensures stable training while ControlNet learns to map conditions from a dataset of 100K images (FFHQ [26] and CELEB-A-HQ [22]).

We condition our model using a set of inputs, collectively denoted as $\mathbf{C}$, which include skin masks ($\mathbf{c}_{\text{skin}}$), hair masks ($\mathbf{c}_{\text{hair}}$), ears and accessory masks ($\mathbf{c}_{\text{ears}}$), a 3D head mesh ($\mathbf{c}_{\text{mesh}}$) rendered in 2D, and background information ($\mathbf{c}_{\text{background}}$). These conditions are combined as follows:

$$\mathbf{C} = \{\mathbf{c}_{\text{skin}}, \mathbf{c}_{\text{hair}}, \mathbf{c}_{\text{ears}}, \mathbf{c}_{\text{mesh}}, \mathbf{c}_{\text{background}}\}. \tag{5}$$

The training objective is defined as:

$$\mathcal{L}_{\text{COND}} = \mathbb{E}_{\mathbf{z}_0,\mathbf{t},\mathbf{C},\epsilon\sim\mathcal{N}(0,1)}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, \mathbf{t}, \mathbf{C})\|_2^2\right]. \tag{6}$$

To further enable text-based conditioning, we employ a CLIP text encoder, transforming prompts into embeddings that guide image generation via cross-attention layers. Classifier-free guidance (CFG) is applied to balance text adherence and image quality, adjusting the model's prediction as:

$$\epsilon_\theta(\mathbf{z}_t, \mathbf{C}) = (1+\omega)\cdot\epsilon_\theta(\mathbf{z}_t, \mathbf{C}) - \omega\cdot\epsilon_\theta(\mathbf{z}_t), \tag{7}$$

where ω controls the guidance strength.

Guided Inpainting: At inference, we replace the locked LDM with an inpainting variant, allowing for guided completion of missing texture regions. We introduce an inpainting mask ($\mathbf{c}_{\text{mask}}$) specifying areas to be filled, while maintaining coherence with the existing texture using the same conditioning set $\mathbf{C}$. The inpainting objective is defined as:

$$\mathcal{L}_{\text{INP}} = \mathbb{E}_{\mathbf{z}_0,\mathbf{t},\mathbf{C},\mathbf{c}_{\text{mask}},\epsilon\sim\mathcal{N}(0,1)}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, \mathbf{t}, \mathbf{C}, \mathbf{c}_{\text{mask}})\|_2^2\right]. \tag{8}$$

Training Details: Our prior is trained for $\sim$1500 GPU hours on an NVIDIA H100 (see [54, 71]).

4.2 3D Texturing Pipeline

First, we render skin and ear masks aligned to the 3D FLAME model [35] from multiple viewpoints and use them to condition our 2D diffusion prior, producing consistent pixel outputs. Next, these pixel colors are projected onto the mesh's UV atlas in an iterative process: visibility checks ensure only previously untextured regions are filled, while morphological erosion and bilinear interpolation refine boundaries and smooth transitions. Together, these steps yield a high-quality 360° head texture.

Rendering 3D-Consistent Input Controls: At test time, we generate 3D-consistent conditioning signals using the FLAME model. Specifically, we extract skin and ear maps aligned with the FLAME mesh, while omitting features like hair, earrings, and eyeglasses to ensure clean, bald head generation. The background is set to a uniform color to minimize artifacts, keeping the model's focus on the head region. The FLAME model, a 3D Morphable Model (3DMM), parameterizes human head shapes and expressions through a low-dimensional latent space. It outputs a 3D head mesh $\mathbf{M} = f_{\text{FLAME}}(\alpha, \delta, \theta)$, where $\mathbf{M}$ is represented by vertices $\{\mathbf{v}_i \in \mathbb{R}^3\}_{i=1}^{N}$ and a fixed topology of faces $\mathbf{F}$. These vertices are controlled by shape parameters α, expression parameters δ, and pose parameters θ, enabling precise manipulation of head structure and expressions.

To capture a complete 360° representation, we render the mesh from 14 viewpoints by rotating a virtual camera around the head. Each vertex $\mathbf{v}_i = (x_i, y_i, z_i)^\top$ is projected onto the 2D image plane using a perspective transformation, where the camera intrinsics $\mathbf{K}$ and extrinsics $\mathbf{R}, \mathbf{t}$ define the projection as $\mathbf{p}_i = \pi(\mathbf{K}(\mathbf{R}\mathbf{v}_i + \mathbf{t}))$. The perspective division is given by $\pi(\mathbf{x}) = \left(\frac{x}{z}, \frac{y}{z}\right)$ for a 3D point $\mathbf{x} = (x, y, z)^\top$.

These rendered meshes and accurate skin/ear segmentation masks serve as the conditioning signals for our compositional 2D image prior.

Progressive UV Mapping: Guided by our 3D-consistent input controls, the generated images of the 2D prior align with the FLAME mesh. Each vertex $\mathbf{v}_i$ on the 3D mesh has a corresponding UV coordinate (u_i, v_i), allowing us to map surface points to a 2D texture space. For each view, visible pixels on the 2D render are mapped to UV coordinates (u, v) based on the surface-to-UV correspondence. Each pixel (x, y) in the rendered frame has an RGB color value $\mathbf{col}(x, y) = [r, g, b]$, which we splat to the texture space at the corresponding (u, v) locations.

Instead of computing the images of all views at once, we iteratively render the images using already existing texture parts. To improve texture quality and avoid blending artifacts near boundaries, we apply a morphological erosion operation to the existing UV texture mask before accumulating new color information. Specifically, for a pixel (x, y), we compute its corresponding UV coordinates (u, v) using precomputed channels:

$$\begin{aligned} u &= \lfloor u_{\text{channel}}(y, x) \cdot R \rfloor \,, \\ v &= \lfloor v_{\text{channel}}(y, x) \cdot R \rfloor \,, \end{aligned} \tag{9}$$

where u_{channel} and v_{channel} are UV maps providing (u, v) coordinates for each pixel and R denotes the image resolution. These (u, v) values are rounded down to the nearest integer for indexing into the texture map.

We conditionally update the UV texture based on visibility checks, ensuring that only new visible regions accumulate:

$$\mathbf{T}(u, v) = \begin{cases} \mathbf{col}(x, y) & \text{if visible at } (u, v), \\ \mathbf{T}(u, v) & \text{otherwise.} \end{cases} \tag{10}$$

For each pixel (x, y) with color $\mathbf{col}(x, y) = [r, g, b]$ in the 512×512 image space, we compute its UV coordinates on the 1024×1024 atlas and define $u_0 = \lfloor u \rfloor$, $u_1 = \lceil u \rceil$, $v_0 = \lfloor v \rfloor$, and $v_1 = \lceil v \rceil$. We then "splat" $\mathbf{col}(x, y)$ into each of the four texels $(u_k, v_\ell) \in \{(u_0, v_0), (u_1, v_0), (u_0, v_1), (u_1, v_1)\}$ by setting:

$$\mathbf{T}(u_k, v_\ell) = \mathbf{col}(x, y) \quad \text{if that texel is not yet filled.} \tag{11}$$

This effectively closes small gaps and holes in the accumulated texture. Applying this process across all pixels and viewpoints yields a full 360° UV texture.

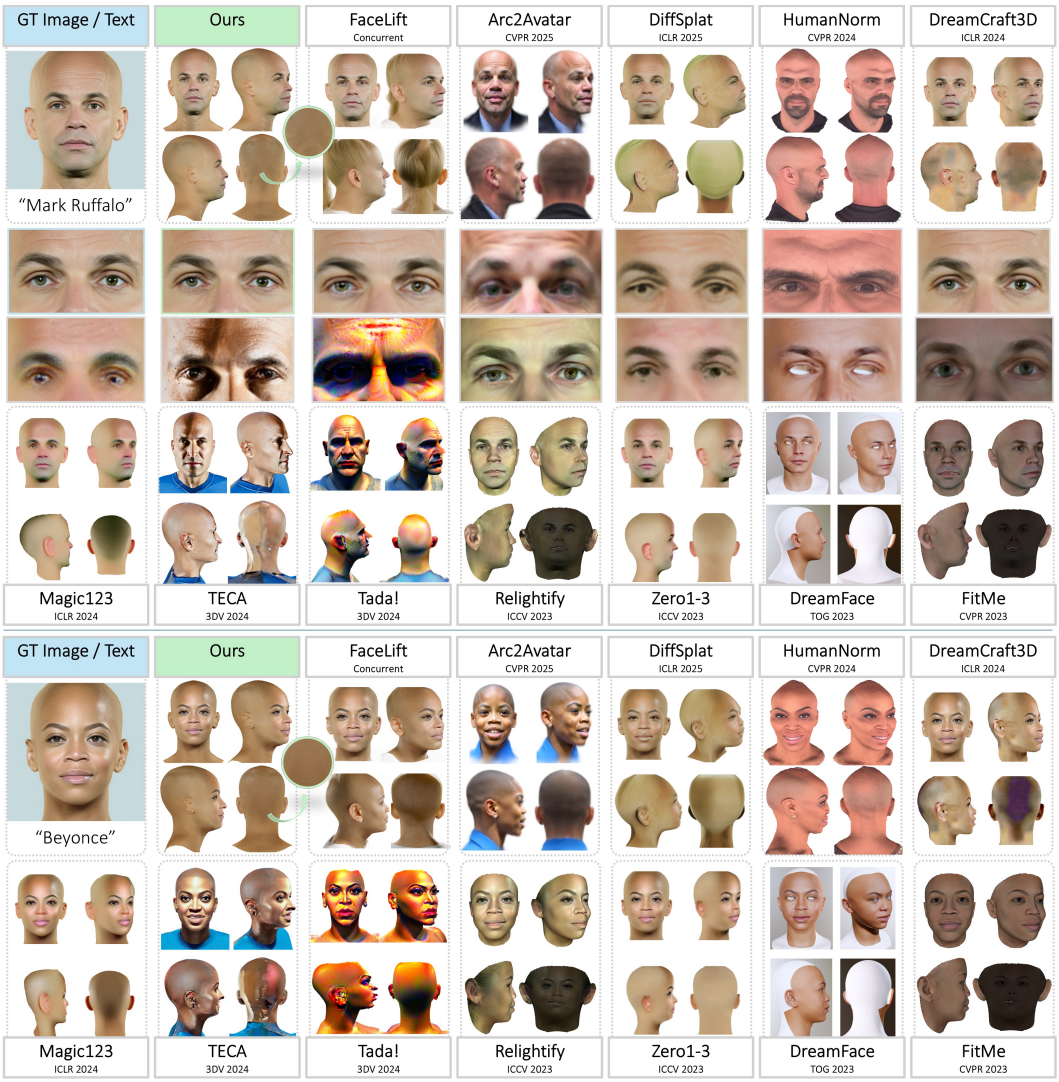

Figure 3: **Qualitative Comparison with State-of-the-Art:** We compare *HairFree* against recent 3D avatar techniques—FaceLift [41], Arc2Avatar [18], DiffSplat [38], HumanNorm [21], DreamCraft3D [60], Magic123 [50], TECA [69], TADA! [36], Relightify! [46], Zero123-XL [39], DreamFace [70], and FitMe [29]—on "Mark Ruffalo" and "Beyonce" with explicit bald-head constraints. *HairFree* delivers the most accurate 360° head shapes, realistic textures, and uniform lighting.

5 Experiments

We evaluate our proposed method in two critical aspects: (1) the compositional 2D human head prior and (2) the texture generation pipeline. Our 2D head prior offers precise, region-specific control over facial components, including skin, hair, and ears. The texture generation pipeline is assessed based on its ability to produce photorealistic, high-quality human head textures, maintaining diversity in ethnicity, age, and style, ensuring accurate texture synthesis throughout the full 360° range.

5.1 2D Human Head Prior

Comparison with State-of-the-Art 2D Bald Proxy Methods: To qualitatively evaluate the 2D human head prior, we compare it as a bald proxy method with the following approaches: (i) diffusion-based generic inpainting (LDM [54]), (ii) GAN-based 2D bald proxy estimation (HairMapper [67]), and (iii) the state-of-the-art hair editing method HairCLIPv2 [66]. The results highlight the ability of

Figure 4: **Photorealistic Rendering Results.** Without using any captioned text prompts during training, our method accurately follows text prompts at test time. *(A) Demographics Attributes:* Rendered textures demonstrating variations in age and demographics across head avatars. *(B) Celebrity Renderings:* Multi-view renderings and textures of famous celebrities. *(C) Varied Meshes and Nationalities:* Showing additional variety in nationality-based text prompts applied to different meshes. Corresponding text prompts, additional results and camera views are available in Appendix A.

our approach to preserve more accurate head shapes and poses while effectively removing hair. The results are shown in Appendix A.

Quantitative Evaluation - ControlNet Strength (CS): We evaluate our model using FID, KID, LPIPS, and PSNR, across different ControlNet strengths (CS). By reducing CS, we can generate more diverse examples, enabling the synthesis of novel head views that do not exist in the training data, see Table 2 and Figure 6 (C).

Ablation - Inpainting (RGB): We compare the performance of our approach with and without inpainting. As shown in Figure 6 (A), the model without inpainting is limited to generating accurate frontal views, while using gradual inpainting from front to back enables the synthesis of a broader range of views, including extreme side, back, and top perspectives (not present in the training data).

5.2 3D Textured Avatar Synthesis

Comparison with State-of-the-Art 3D Methods: We conduct a qualitative comparison against recent state-of-the-art 3D avatar generation methods. Zero123-XL [39] and Magic123 [50] rely on 2D diffusion-based priors using Score Distillation Sampling [49], which often leads to oversaturation, low detail preservation, and inconsistent geometry across views. Tada![36], which is specialized for humans, is also SDS-based and struggles with oversaturation, while HumanNorm[21] introduces unnatural reddish skin tones. DreamCraft3D [60] and DiffSplat [38] leverage 3D Gaussian splatting, which improves efficiency but tends to produce low-frequency details and blurry textures. TECA [69] generates clothing in addition to the head. FaceLift [41] produces relatively accurate head shapes but may hallucinate hair instead of adhering to the bald constraint, and like many other methods, it suffers from extreme view-specific illumination artifacts (e.g., shadows). Arc2Avatar [18], Relightify [46],

Figure 5: **Generalization Properties.** Our method enables blending between photorealistic and stylized renderings through the classifier-free guidance (CSG) and control strength (CS) parameters. Despite being fine-tuned only on realistic human faces, it generalizes strongly to *diverse, non-uniform* bald textures, enabling appearances beyond natural human skin. **CSG:** Impacts the alignment with the text. **CS:** Lower values relax the model's adherence to our prior, allowing for greater style diversity (see row 3). Additional results are available in Appendix A.

Method	Runtime	Stages	Quality
FitMe	minutes	2	Mid
DreamFace	minutes	2	Mid
Zero1-3	minutes	**1**	Low
Relightify	minutes	2	Mid
Tada!	1 hour	**1**	Low
TECA	minutes	**1**	Low
Magic123	1 hour	2	Low
DreamCraft3D	3 hours	4	High
HumanNorm	1 hour	3	Mid
DiffSplat	**seconds**	**1**	Mid
Arc2Avatar	1-2 hours	2-3	Mid
FaceLift	**seconds**	**1**	High
HairFree (Our)	minutes	**1**	**High**

Table 1: **Quantitative Comparison with State-of-the-Art:** Runtime, stages, quality.

CS	PSNR ↑	MSE ↓	LPIPS ↓	FID ↓	KID ↓
1.0	**18.38**	**0.018**	**0.24**	**6.6**	**0.0027**
0.75	17.14	0.023	0.29	11.5	0.0056
0.5	15.05	0.035	0.39	27.9	0.0167
0.25	12.53	0.060	0.53	78.9	0.0518
0	10.27	0.097	0.7268	384	0.4706

Table 2: **Quantitative Analysis/Ablation:** We vary ControlNet Strength (CS) from 0 to 1 and measure image similarity over 10K samples. At CS = 1.0, outputs closely match the input; as CS decreases, diversity increases, enabling the synthesis of bald, back-view, and extreme side-views.

DreamFace [70], and FitMe [29] all struggle to generate a complete, bald scalp. Arc2Avatar produces a full 360° avatar but suffers from entanglement issues, with visible clothing and a non-bald scalp showing a hairline instead of clear skin. Relightify and FitMe both reconstruct texture and geometry together rather than focusing on texturing, limiting their ability to produce consistent high-quality results. Neither method effectively addresses hair removal, and their outputs are of medium quality. DreamFace is further limited to generating only the face region without the scalp, and has limited skin diversity. Most of the baselines fail to enforce baldness, introducing unwanted hair despite explicit constraints. Additional quantitative analysis with respect to runtime, method complexity, and image quality is given in Table 1. Our approach achieves the highest fidelity 360° textures, setting a new benchmark for textured 3D avatar quality (Figure 3).

Qualitative Evaluation: We demonstrate the diversity and realism of our textures across various text prompts, including photorealistic, celebrity, and stylized outputs (Figures 4 and 5). Additional results covering fantasy, artistic styles, animal faces, and material synthesis are in Appendix A.

Ablation - Classifier-Free Guidance (CSG): In Figure 6 (B), we demonstrate how varying CSG affects the alignment of the output with text. Higher CSG values lead to outputs more aligned with the text prompt, while lower values relax this alignment and result in a photorealistic appearance.

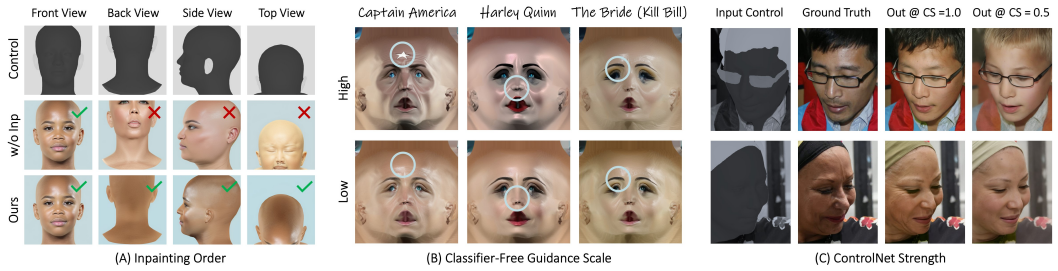

Figure 6: **Ablation Studies:** (A) Without our progressive inpainting, only frontal views are accurate. (B) Higher CSG values improve adherence to text prompts; lower values produce a more natural look. (C) Reducing ControlNet strength gradually weakens alignment with the input conditioning signal.

User Study: We conducted a user study for Person 1 (87 users) and Person 2 (94 users), asking to select the top 1 method in terms of avatar plausibility. Each user was shown one frontal, two side, and one back view of avatars generated by 13 different methods (Figure 3, Appendix A). For Person 1, *HairFree* led with 40.23% of participants, followed by *DreamCraft3D* at 18.39%, *FaceLift* and *Arc2Avatar* tied at 8.05% each, and *DiffSplat* with 6.90%. For Person 2, *HairFree* led with 39.36%, followed by *DreamCraft3D* at 14.89%, *FaceLift* at 9.57%, and *Arc2Avatar* and *DiffSplat* tied at 7.45% each. These results confirm the superiority of our method (Figure 7).

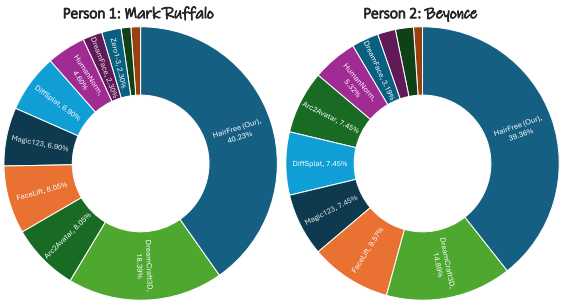

Figure 7: **User Study:** Distribution of user preferences among the selected methods for two individuals, highlighting the preference for our method.

Comparison with Inpainting Methods: Figure 8 compares three approaches for inpainting in the UV space: (1) a naive method where all non-frontal views are filled with a uniform color, resulting in visible seams, (2) Content-Aware Fill (CAF)-based inpainting in UV space, which also produces seams, particularly at the back, and (3) our method, which performs inpainting directly in image space. Unlike the other methods, our approach seamlessly preserves intricate patterns without any visible seams in the UV space, even for intricate non-human skin textures.

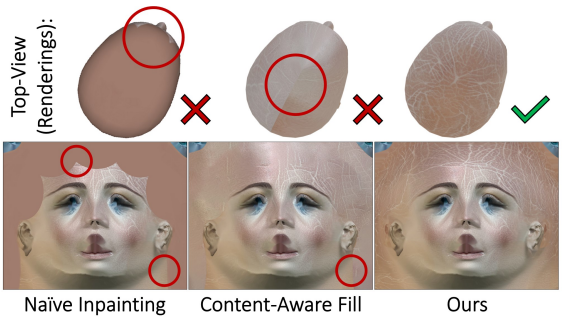

Figure 8: **Comparisons:** Inpainting in UV Space.

Limitations & Societal Impact: Our method is sensitive to lighting inconsistencies—addressable via intrinsic decomposition (e.g., IntrinsicAnything [8]) for relightable textures. Text prompts inherit any ethnicity or appearance biases from the pre-trained diffusion prior. High-fidelity textures could enable identity spoofing or deepfakes, so responsible, careful use is crucial.

6 Conclusion

We introduced *HairFree*, a diffusion-based framework that generates realistic, 3D-consistent bald head textures by conditioning a large latent diffusion model on face parsing maps, 3D meshes, and background cues. During inference, guided inpainting fills unseen regions as the camera moves, yielding seamless 360° textures. Our compositional 2D prior cleanly separates skin from hair, enabling flexible 3D layering of external assets—such as strand-based hairstyles—independent of hairline. Our evaluations demonstrate that *HairFree* delivers state-of-the-art fidelity and compositionality compared to existing 3D head avatar methods.

Acknowledgements: The authors thank Tsvetelina Alexiadis, Tomasz Niewiadomski, and Taylor Obersat for perceptual study; Yao Feng, Tingting Liao, Dimitrios Gerogiannis, Weijie Lyu, Alexandros Lattas, and Foivos Paraperas Papantoniou for help with the baselines; Peter Kulits, Yuliang Xiu, and all reviewers for their valuable feedback; and Benjamin Pellkofer for IT support. Justus Thies is supported by the DFG Excellence Strategy— EXC-3057 and the project is co-funded by the European Union (ERC, Lemo, 101162081). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] Midjourney. https://www.midjourney.com, 2023. Accessed: 2025-05-01.

[2] Shivangi Aneja, Justus Thies, Angela Dai, and Matthias Nießner. ClipFace: Text-guided Editing of Textured 3D Morphable Models. In *ArXiv preprint arXiv:2212.01406*, 2022.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations (ICLR)*, 2019.

[4] Yukang Cao, Yan-Pei Cao, Kai Han, Ying Shan, and Kwan-Yee K Wong. DreamAvatar: Text-and-Shape Guided 3D Human Avatar Generation via Diffusion Models. *arXiv preprint:2304.00916*, 2023.

[5] Dave Zhenyu Chen, Haoxuan Li, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Scenetex: High-quality texture synthesis for indoor scenes via diffusion priors, 2023.

[6] Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*, 2023.

[7] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3D: Disentangling Geometry and Appearance for High-quality Text-to-3D Content Creation. In *ICCV*, 2023.

[8] Xi Chen, Sida Peng, Dongchen Yang, Yuan Liu, Bowen Pan, Chengfei Lv, and Xiaowei Zhou. Intrinsicanything: Learning diffusion priors for inverse rendering under unknown illumination. In *European Conference on Computer Vision*, pages 450–467. Springer, 2024.

[9] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[10] Bernhard Egger, William A. P. Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, Christian Theobalt, Volker Blanz, and Thomas Vetter. 3d morphable face models—past, present, and future. *ACM Trans. Graph.*, 39(5), June 2020.

[11] Haven Feng. Photometric flame fitting. https://github.com/HavenFeng/photometric_optimization, 2019.

[12] Panagiotis P Filntisis, George Retsinas, Foivos Paraperas-Papantoniou, Athanasios Katsamanis, Anastasios Roussos, and Petros Maragos. Spectre: Visual speech-informed perceptual 3d facial expression reconstruction from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5744–5754, 2023.

[13] William Gao, Noam Aigerman, Groueix Thibault, Vladimir Kim, and Rana Hanocka. Textdeformer: Geometry manipulation using text guidance. In *ACM Transactions on Graphics (SIGGRAPH)*, 2023.

[14] Baris Gecer, Jiankang Deng, and Stefanos Zafeiriou. Ostec: One-shot texture completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7628–7638, June 2021.

[15] Baris Gecer, Alexander Lattas, Stylianos Ploumpis, Jiankang Deng, Athanasios Papaioannou, Stylianos Moschoglou, and Stefanos Zafeiriou. Synthesizing coupled 3d face modalities by trunk-branch generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV)*. Springer, 2020.

[16] Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[17] Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos P Zafeiriou. Fast-ganfit: Generative adversarial network for high fidelity 3d face reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[18] Dimitrios Gerogiannis, Foivos Paraperas Papantoniou, Rolandos Alexandros Potamias, Alexandros Lattas, and Stefanos Zafeiriou. Arc2avatar: Generating expressive 3d avatars from a single image via id guidance. *arXiv preprint arXiv:2501.05379*, 2025.

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[21] Xin Huang, Ruizhi Shao, Qi Zhang, Hongwen Zhang, Ying Feng, Yebin Liu, and Qing Wang. Humannorm: Learning normal diffusion model for high-quality and realistic 3d human generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4568–4577, 2024.

[22] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[23] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:12104–12114, 2020.

[24] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021.

[25] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[26] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12):4217–4228, 2021.

[27] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023.

[28] Alexandros Lattas, Stylianos Moschoglou, Baris Gecer, Stylianos Ploumpis, Vasileios Triantafyllou, Abhijeet Ghosh, and Stefanos Zafeiriou. Avatarme: Realistically renderable 3d facial reconstruction "in-the-wild". In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[29] Alexandros Lattas, Stylianos Moschoglou, Stylianos Ploumpis, Baris Gecer, Jiankang Deng, and Stefanos Zafeiriou. Fitme: Deep photorealistic 3d morphable model avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8629–8640, 2023.

[30] Alexandros Lattas, Stylianos Moschoglou, Stylianos Ploumpis, Baris Gecer, Abhijeet Ghosh, and Stefanos P Zafeiriou. Avatarme++: Facial shape and brdf inference with photorealistic rendering-aware gans. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[31] Myunggi Lee, Wonwoong Cho, Moonheum Kim, David I. Inouye, and Nojun Kwak. Styleuv: Diverse and high-fidelity uv map generative model. *ArXiv*, abs/2011.12893, 2020.

[32] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. Tryongan: Body-aware try-on via layered interpolation. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2021)*, 40(4), 2021.

[33] Hong Li, Yutang Feng, Song Xue, Xuhui Liu, Bohan Zeng, Shanglin Li, Boyu Liu, Jianzhuang Liu, Shumin Han, and Baochang Zhang. Uv-idm: identity-conditioned latent diffusion model for face uv-texture generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10585–10595, 2024.

[34] Ruilong Li, Karl Bladin, Yajie Zhao, Chinmay Chinara, Owen Ingraham, Pengda Xiang, Xinglei Ren, Pratusha Prasad, Bipin Kishore, Jun Xing, and Hao Li. Learning formation of physically-based face attributes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[35] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Trans. Graph.*, 36(6):194–1, 2017.

[36] Tingting Liao, Hongwei Yi, Yuliang Xiu, Jiaxiang Tang, Yangyi Huang, Justus Thies, and Michael J Black. Tada! text to animatable digital avatars. In *2024 International Conference on 3D Vision (3DV)*, pages 1508–1519. IEEE, 2024.

[37] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3D: High-Resolution Text-to-3D Content Creation. In *CVPR*, 2023.

[38] Chenguo Lin, Panwang Pan, Bangbang Yang, Zeming Li, and Yadong Mu. Diffsplat: Repurposing image diffusion models for scalable 3d gaussian splat generation. In *International Conference on Learning Representations (ICLR)*, 2025.

[39] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9298–9309, October 2023.

[40] Huiwen Luo, Koki Nagano, Han-Wei Kung, Qingguo Xu, Zejian Wang, Lingyu Wei, Liwen Hu, and Hao Li. Normalized avatar synthesis using stylegan and perceptual refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11662–11672, June 2021.

[41] Weijie Lyu, Yi Zhou, Ming-Hsuan Yang, and Zhixin Shu. Facelift: Single image to 3d head with view generation and gs-lrm, 2024.

[42] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2Mesh: Text-Driven Neural Stylization for Meshes. In *CVPR*, 2022.

[43] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, volume 12346, pages 405–421. Springer, 2020.

[44] Mirela Ostrek, Carol O'Sullivan, Michael J. Black, and Justus Thies. Synthesizing environment-specific people in photographs. In *European Conference on Computer Vision (ECCV)*, 2024.

[45] Mirela Ostrek and Justus Thies. Stable video portraits. In *European Conference on Computer Vision (ECCV)*, 2024.

[46] Foivos Paraperas Papantoniou, Alexandros Lattas, Stylianos Moschoglou, and Stefanos Zafeiriou. Relightify: Relightable 3d faces from a single image via diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8806–8817, 2023.

[47] Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3d face model for pose and illumination invariant face recognition. In *2009 sixth IEEE international conference on advanced video and signal based surveillance*, pages 296–301. Ieee, 2009.

[48] Ryan Po, Wang Yifan, Vladislav Golyanik, Kfir Aberman, Jonathan T Barron, Amit H Bermano, Eric Ryan Chan, Tali Dekel, Aleksander Holynski, Angjoo Kanazawa, et al. State of the art on diffusion models for visual computing. *arXiv preprint arXiv:2310.07204*, 2023.

[49] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. DreamFusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023.

[50] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, and Bernard Ghanem. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. In *International Conference on Learning Representations (ICLR)*, 2024.

[51] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[52] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[53] Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. Texture: Text-guided texturing of 3d shapes. In *ACM SIGGRAPH 2023 conference proceedings*, pages 1–11, 2023.

[54] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[55] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. 2022.

[56] Kripasindhu Sarkar, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. Humangan: A generative model of human images. In *2021 International Conference on 3D Vision (3DV)*, pages 258–267. IEEE, 2021.

[57] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

[58] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[59] Ron Slossberg, Ibrahim Jubran, and Ron Kimmel. Unsupervised high-fidelity facial texture generation and reconstruction. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIII*, page 212–229, Berlin, Heidelberg, 2022. Springer-Verlag.

[60] Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, and Yebin Liu. Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. In *International Conference on Learning Representations (ICLR)*, 2024.

[61] Ayush Tewari, Ohad Fried, Justus Thies, Vincent Sitzmann, Stephen Lombardi, Kalyan Sunkavalli, Ricardo Martin-Brualla, Tomas Simon, Jason Saragih, Matthias Nießner, et al. State of the art on neural rendering. In *Computer Graphics Forum*, volume 39, pages 701–727. Wiley Online Library, 2020.

[62] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Wang Yifan, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, et al. Advances in neural rendering. In *Computer Graphics Forum*, volume 41, pages 703–735. Wiley Online Library, 2022.

[63] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *Acm Transactions on Graphics (TOG)*, 38(4):1–12, 2019.

[64] Lizhen Wang, Zhiyuan Chen, Tao Yu, Chenguang Ma, Liang Li, and Yebin Liu. Faceverse: A fine-grained and detail-controllable 3d face morphable model from a hybrid dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20333–20342, June 2022.

[65] Lizhen Wang, Xiaochen Zhao, Jingxiang Sun, Yuxiang Zhang, Hongwen Zhang, Tao Yu, and Yebin Liu. Styleavatar: Real-time photo-realistic portrait avatar from a single video. *arXiv preprint arXiv:2305.00942*, 2023.

[66] Tianyi Wei, Dongdong Chen, Wenbo Zhou, Jing Liao, Weiming Zhang, Gang Hua, and Nenghai Yu. Hairclipv2: Unifying hair editing via proxy feature blending. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[67] Yiqian Wu, Yong-Liang Yang, and Xiaogang Jin. Hairmapper: Removing hair from portraits using gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4227–4236, June 2022.

[68] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *International Journal of Computer Vision*, 129:3051–3068, 2021.

[69] Hao Zhang, Yao Feng, Peter Kulits, Yandong Wen, Justus Thies, and Michael J Black. Teca: Text-guided generation and editing of compositional 3d avatars. In *2024 International Conference on 3D Vision (3DV)*, pages 1520–1530. IEEE, 2024.

[70] Longwen Zhang, Qiwei Qiu, Hongyang Lin, Qixuan Zhang, Cheng Shi, Wei Yang, Ye Shi, Sibei Yang, Lan Xu, and Jingyi Yu. Dreamface: Progressive generation of animatable 3d faces under text guidance. *ACM Trans. Graph.*, 42(4), July 2023.

[71] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.

[72] Michael Zollhöfer, Justus Thies, Pablo Garrido, Derek Bradley, Thabo Beeler, Patrick Pérez, Marc Stamminger, Matthias Nießner, and Christian Theobalt. State of the art on monocular 3d face reconstruction, tracking, and applications. In *Computer graphics forum*, volume 37, pages 523–550. Wiley Online Library, 2018.

A Appendix

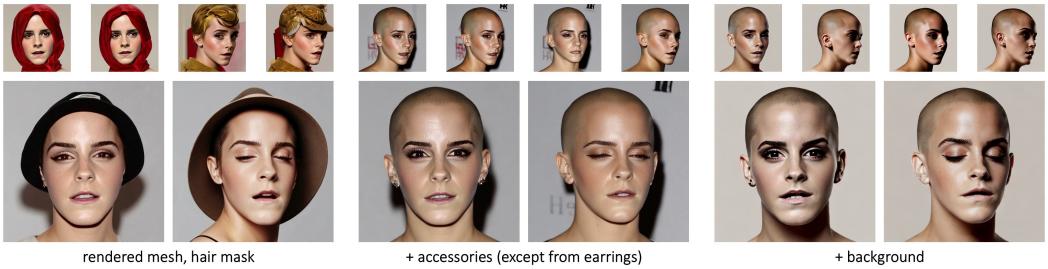

Figure 9: Outputs from three models trained with progressively richer conditioning inputs: (left) rendered mesh with hair mask only; (middle) + accessories (excluding earrings); (right) + background. All examples are from an early training epoch and highlight the effect of each conditioning signal. The model gradually learns to associate additional inputs with their corresponding visual semantics. Note that earrings are not included in the conditioning, which causes them to appear or disappear inconsistently across examples. The final model, trained for longer, achieves more stable results and can fully remove hair during inference.

A.1 Ablation Study on Semantic Conditioning Signals

The face and skin mask covers all facial regions including eyes, nose, lips, etc, teaching the model what skin and facial features look like (see "Devised Input Conditioning" in Figure 2). The hair mask is used during training only, helping the model learn hair locations so it can remove hair at inference (e.g., for bald heads). Ear masks are used during both training and inference; at inference, we use precise 3D ear masks from FLAME for accurate, mesh-aligned ear synthesis. Accessory masks allow the model to learn to recognize and remove the accessories by omitting the mask at test time. Note that during training time those semantic masks can not be derived from the 3D mesh. See Figure 9 and Figure 10 for an ablation study on semantic conditioning signals.

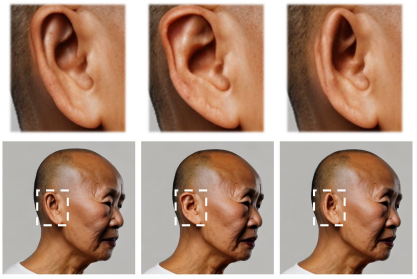

Figure 10: Examples showing inconsistent ear synthesis when ear masks are omitted. Each column shows a bald head generated from the same text prompt (e.g., "old Asian woman") with zoomed-in ear crops. Without explicit ear guidance, the model produces visible variations in ear shape, size, and placement across samples. This instability indicates that accurate, mesh-aligned ear-mask supervision is crucial for maintaining structural consistency during generation.

Janus effect: The Janus effect arises because the ControlNet is trained mostly on frontal and side views. When directly conditioned on a back-view mesh and semantic mask, especially with the default ControlNet strength of 1.0, it tends to hallucinate a face, having never seen such inputs during training (see Figure 6 (A), Figure 11). Progressive inpainting resolves this by gradually completing the texture from front to back, so later views only need to fill in small missing regions. Lowering ControlNet strength during these steps prevents over-conditioning on unfamiliar views. Additionally, we use a single face and skin mask that combines all facial attributes (eyes, lips, nose, etc.) into one region. We avoid using separate masks for each part, as many of these features are small and prone to inaccuracies in the off-the-shelf face parsing model. See "Devised Input Conditioning" in Figure 2 for an example of this mask.

A.2 Controlnet Strength Influence on Distribution Shift

The ControlNet strength has a major influence on shifting from one distribution to another, see Figure 12. Unfortunately, we don't have a large dataset of bald heads, where we could directly evaluate the generation of the backside of the head. However, we conducted an experiment on changing the distribution from human faces to cat faces (which is an extreme case of a distribution shift), where we could analyse the effects of the ControlNet Strength. Specifically, we applied our

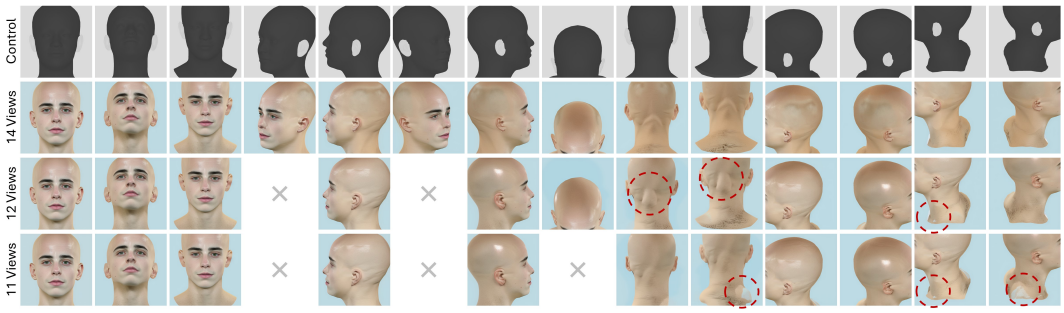

Figure 11: **Ablation on the number of conditioning views:** We evaluate the effect of reducing the number of input views used for texture generation. Row 1 shows the 14 control views. Row 2 shows our full setting with 14 views. Rows 3 and 4 show results with 12 and 11 views, respectively. Red highlights indicate artifacts that emerge as view coverage decreases.

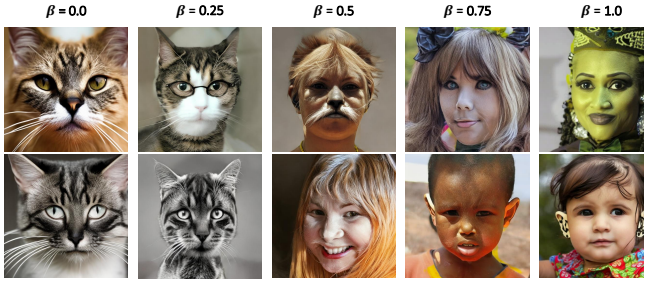

Figure 12: Visualization of the distribution shift between the cat and the human face distribution by controlling the ControlNet strength.

β	FID	KID
1.0	231.32	0.2074
0.75	228.46	0.2035
0.5	212.10	0.1910
0.25	39.88	0.0237
0	81.93	0.0347

Table 3: Quantiative Controlnet strength influence (controlled with β) on distribution shift.

method to the AFHQ-Cat dataset from StarGAN v2 [9], a commonly used dataset from a significantly different domain. We measured FID and KID scores at various values, see Table 3. For higher values of ControlNet strength (e.g., $\beta >= 0.5$), the model remains biased toward synthesizing human faces. This prevents effective generation of samples from the cat distribution, resulting in high FID/KID scores and poor visual alignment with the AFHQ domain. At $\beta = 0.25$, the ControlNet guidance is reduced enough to allow the diffusion prior to generate realistic cat faces, while still retaining enough conditioning to preserve global head structure. This balance aligns well with the AFHQ-Cat distribution, as reflected in the significantly improved scores. At $\beta = 0$, although the model continues to produce cat faces, the samples are often zoomed-in facial crops rather than full cat heads, deviating from the AFHQ distribution. In contrast, at $\beta = 0.25$, structural guidance helps preserve head framing consistent with the dataset.

A.3 Fixed & Varied Meshes with & without Hair

Figures 13 and 14 illustrate texture maps rendered on head models from three different views, highlighting the consistency of our method in maintaining photorealistic details. The examples span a diverse range of ethnicities and age groups, emphasizing the versatility of our approach in capturing the nuanced characteristics of human faces.

To explore the expressive capabilities of our method, Figures 15 and 16 contain examples generated from text prompts with "fantasy" elements. The rendered textures demonstrate the ability of our method to produce imaginative and stylistic results, maintaining consistency and coherence across different views.

To analyze the ability of our method to replicate artistic styles, Figures 17, and 18 present generations inspired by famous paintings and painting techniques. These results highlight how our approach captures distinct brushwork, color schemes, and compositional elements, preserving the essence of each referenced style while maintaining structural coherence.

Figure 13: **Photo-Realistic Results (I):** Rendered texture maps of the head are shown from three different views. The textures are displayed across a range of ethnicities and ages, illustrating the versatility and effectiveness of our method in handling diverse facial features with high fidelity. These results correspond to the Figure: "Photorealistic Rendering Results," in the main paper, further demonstrating the robustness of our approach in capturing fine-grained details.

To further evaluate versatility, Figure 19 contains synthesized animal faces, illustrating the method's capability to generate lifelike textures and anatomical consistency. The results reflect detailed fur patterns, expressive facial structures, and species-specific characteristics, demonstrating both realism and artistic stylization.

Finally, Figures 20, 21, 22, 23 and 24 include generations resembling various materials and gemstones, emphasizing the model's ability to synthesize diverse surface qualities. From the translucency of crystals to the roughness of natural stone, the results capture essential visual properties such as light refraction, texture variation, and intricate reflections, reinforcing the adaptability of our technique across different material types.

A.4 Comparison with 2D Bald Proxy Baselines

In Figures 25 and 26, we compare our approach to several 2D baseline methods, including LDM Inpaint [54], HairMapper [67], and HairCLIPv2 [66]. Our results are shown alongside bald proxy and hairstyle editing methods, demonstrating superior preservation of head shapes and poses while addressing limitations such as quality degradation and inconsistent outputs. This highlights the robustness of our 2D diffusion prior.

A.5 Texture Maps Results

Finally, Figure 27 provides additional examples of celebrity textures generated from text prompts using fixed meshes. These results include a variety of skin tones, facial features, and expressions, showing the versatility of our method in creating high-quality, photorealistic faces. The generated textures capture the unique facial features that distinguish each celebrity, such as specific bone structures, eye shapes, and other signature traits. By handling a wide range of characteristics, these examples highlight how our approach maintains realism and consistency across different celebrity textures. This demonstrates the robustness of our method in accurately capturing detailed facial features and natural variations.

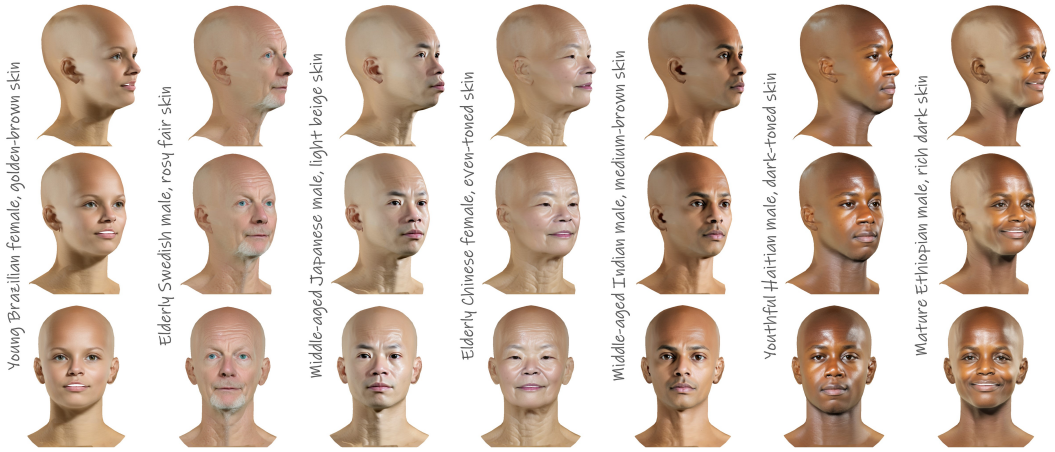

Figure 14: **Photo-Realistic Results (II):** Rendered texture maps of the head are shown from three different views. The textures cover a range of ethnicities, ages, and facial features, demonstrating the versatility and effectiveness of our method in capturing diverse characteristics. These results correspond to the figure "Photorealistic Rendering Results" in the main paper, illustrating the ability of our approach to produce high-quality, realistic faces with consistent detail across views.

A: *Elderly Mongolian female, light olive skin*	**H:** *Young Brazilian female, golden-brown skin*
B: *Middle-aged Chilean male, warm tan skin*	**I:** *Elderly Swedish male, rosy fair skin*
C: *Young Somali female, deep dark brown skin*	**J:** *Middle-aged Japanese male, light beige skin*
D: *Elderly Inuit male, pale beige skin*	**K:** *Elderly Chinese female, even-toned skin*
E: *Middle-aged Turkish male, olive-brown skin*	**L:** *Middle-aged Indian male, medium-brown skin*
F: *Youthful Thai male, light golden skin*	**M:** *Youthful Haitian male, dark-toned skin*
G: *Elderly Moroccan female, sandy-toned skin*	**N:** *Mature Ethiopian male, rich dark skin*

Table 4: Skin tones and demographic descriptions for the results shown in the main paper (“Photorealistic Rendering Results”).

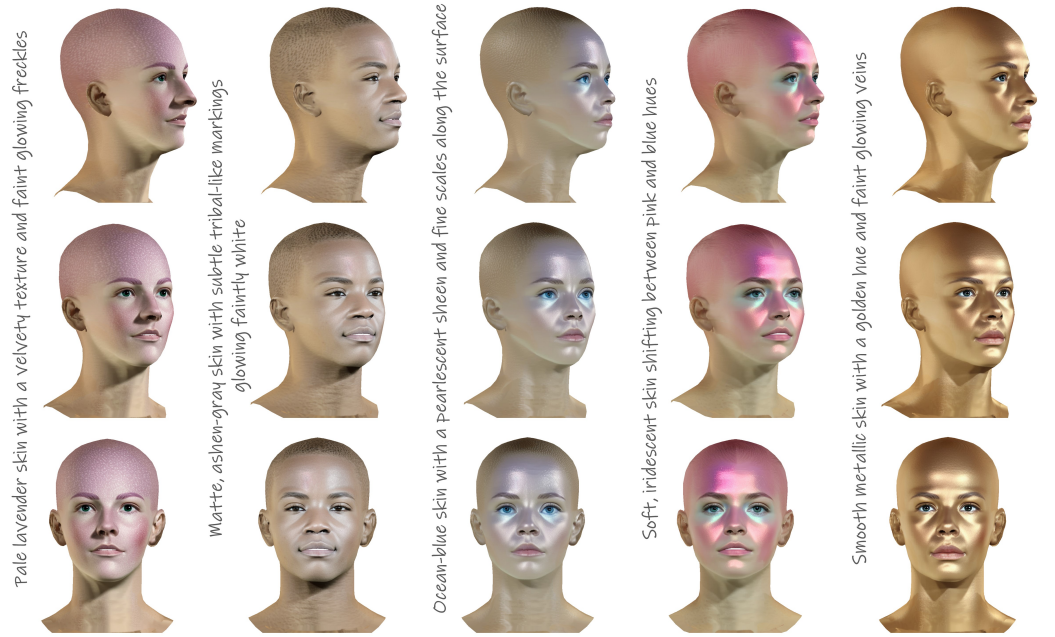

Figure 15: **Abstract (I):** Rendered textures generated from text prompts with "fantasy" elements, shown from three different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

Figure 16: **Abstract (II):** Rendered textures generated from text prompts with "fantasy" elements, shown from three different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

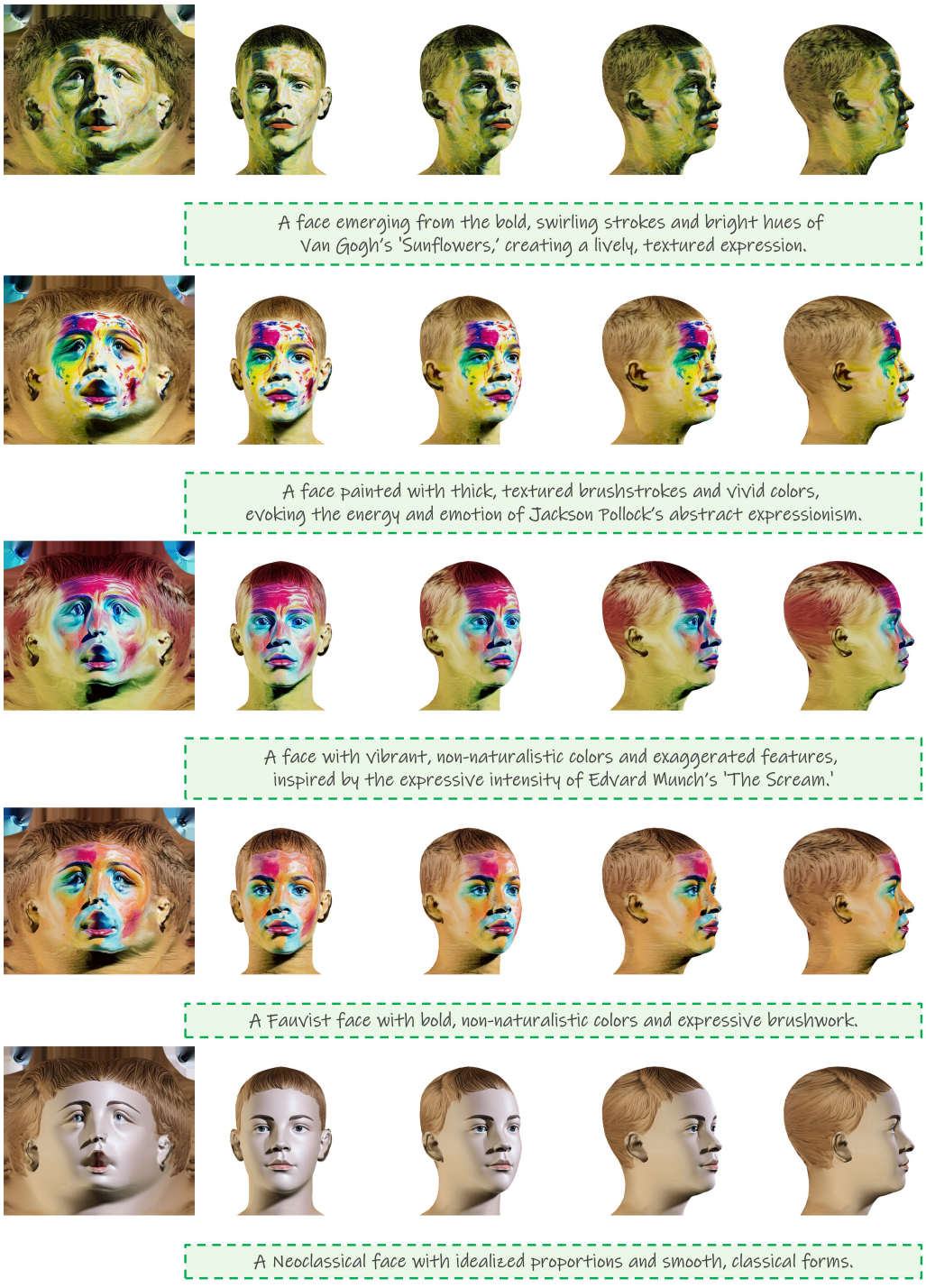

Figure 17: **Painting(s) Styles (I):** Rendered textures generated from text prompts with "fantasy" elements, shown from four different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

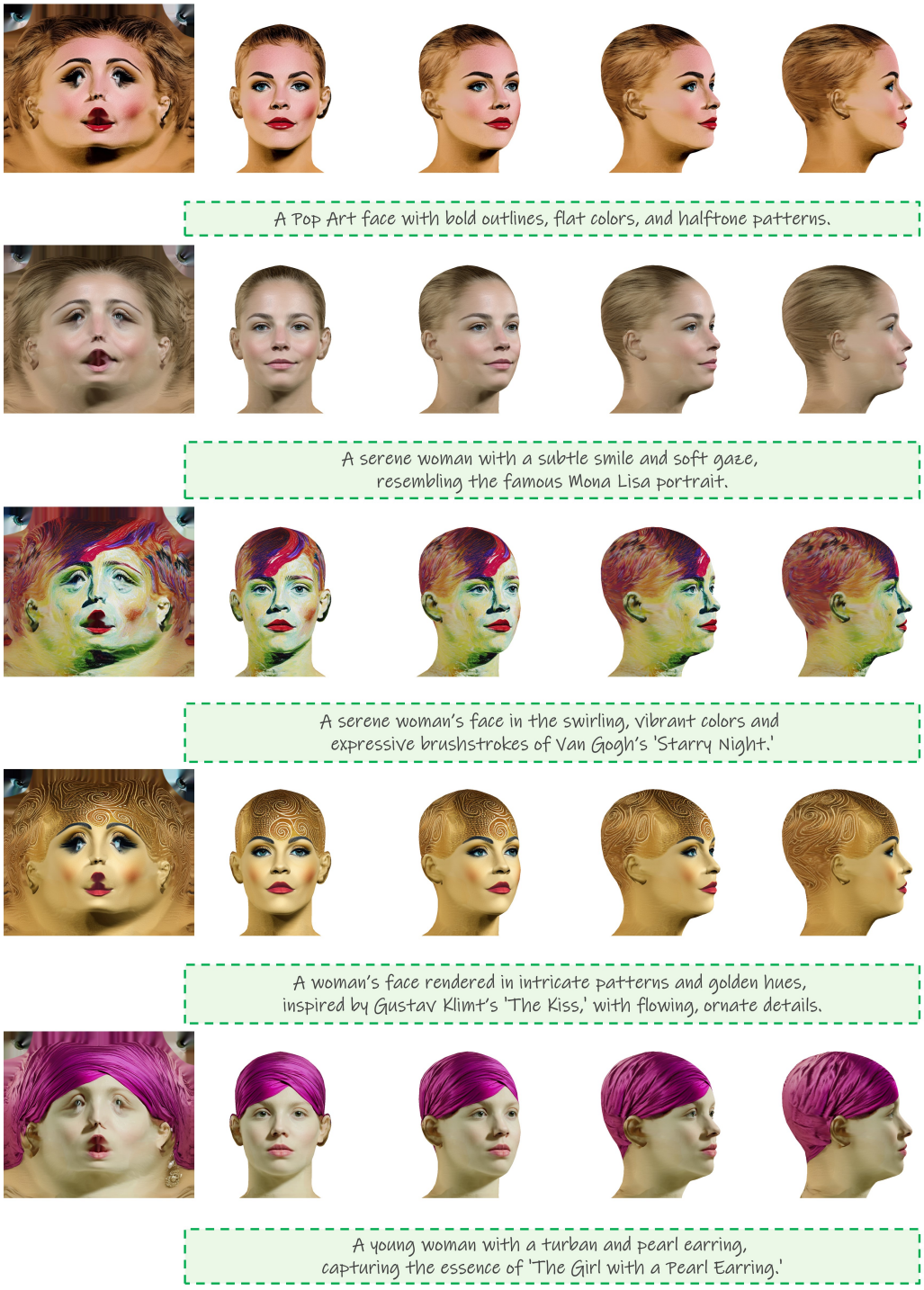

Figure 18: **Painting(s) Styles (II):** Rendered textures generated from text prompts with "fantasy" elements, shown from four different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

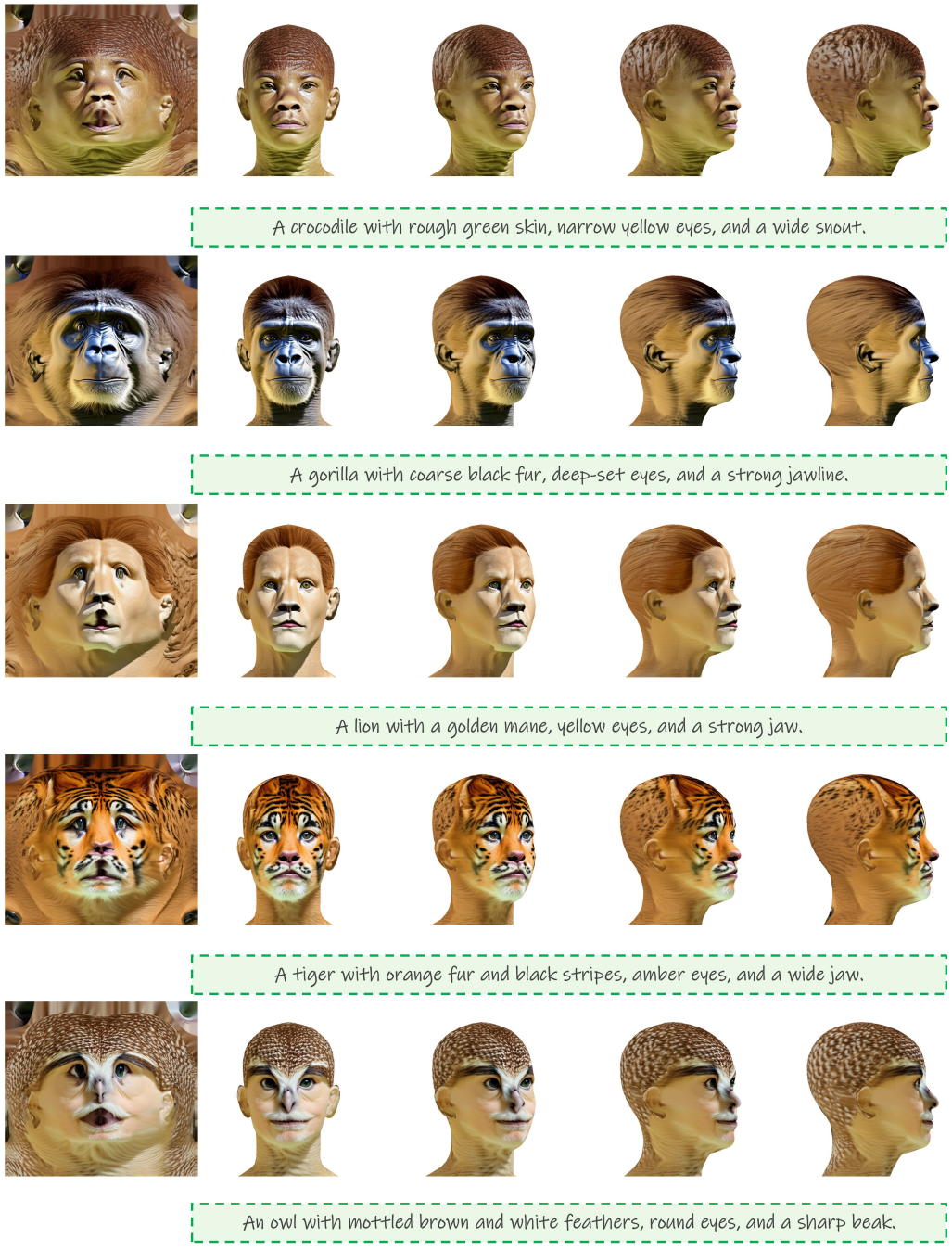

Figure 19: **Animals:** Rendered textures generated from text prompts with "animal" elements, shown from four different views.

Figure 20: **Gemstones/Materials (I):** Rendered textures generated from text prompts with "gemstone/material" elements, shown from four different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

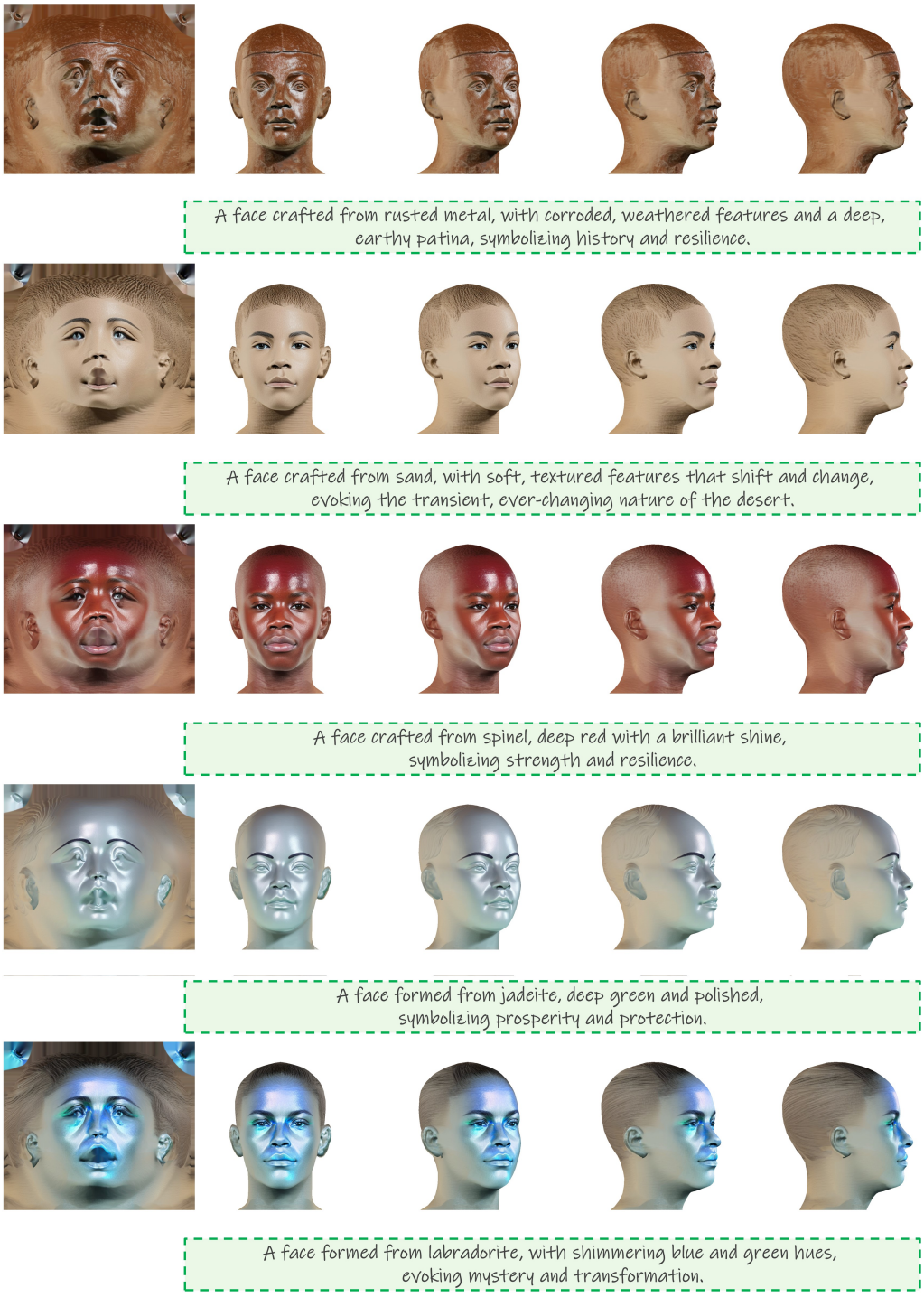

Figure 21: **Gemstones/Materials (II):** Rendered textures generated from text prompts with "gemstone/material" elements, shown from four different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

Figure 22: **Gemstones/Materials (III):** Rendered textures generated from text prompts with "gemstone/material" elements, shown from four different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

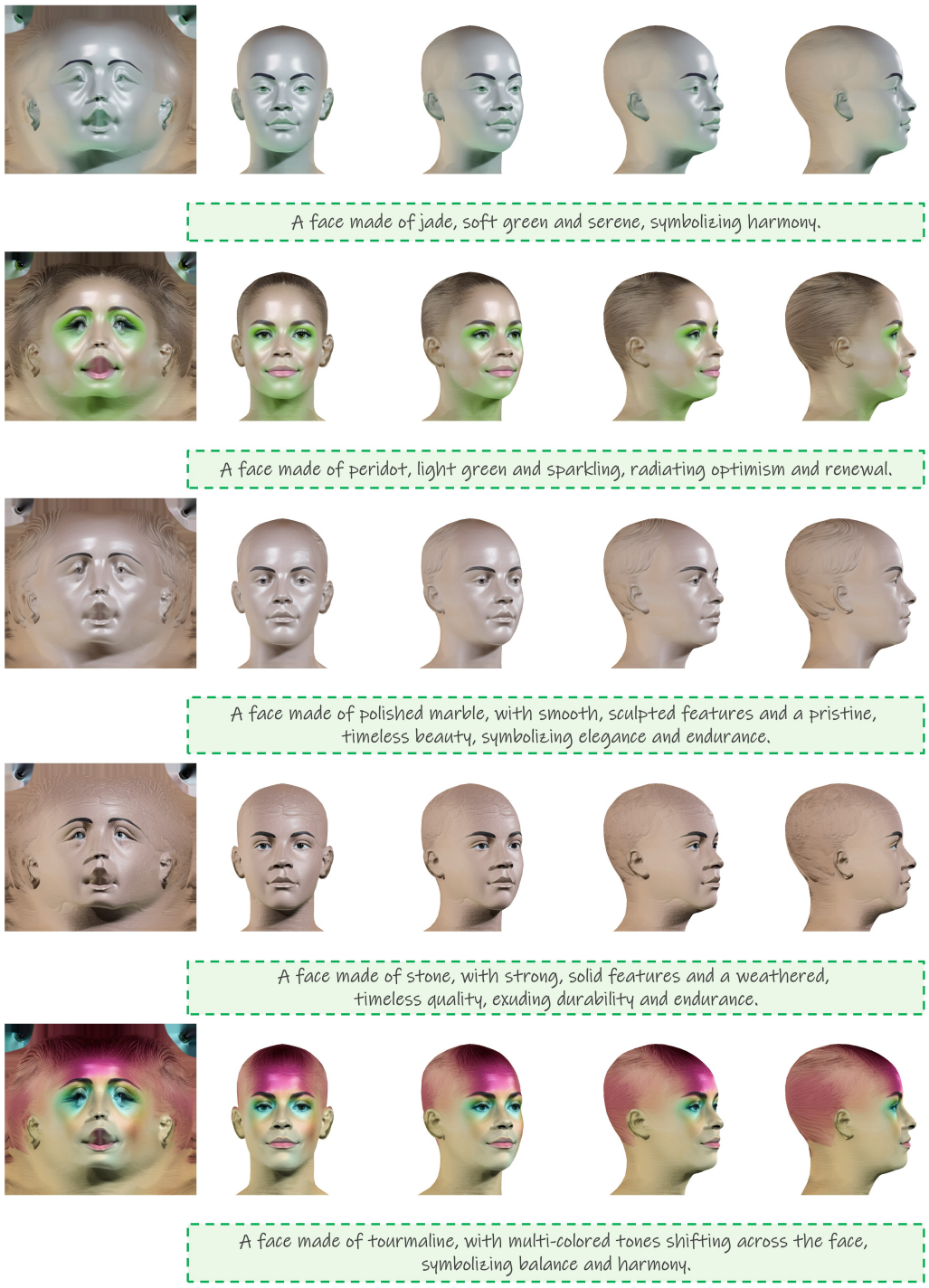

Figure 23: **Gemstones/Materials (IV):** Rendered textures generated from text prompts with "gemstone/material" elements, shown from four different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

Figure 24: **Gemstones/Materials (V):** Rendered textures generated from text prompts with "gemstone/material" elements, shown from four different views. These textures show the consistency and variety of our method in generating imaginative and stylistic facial features.

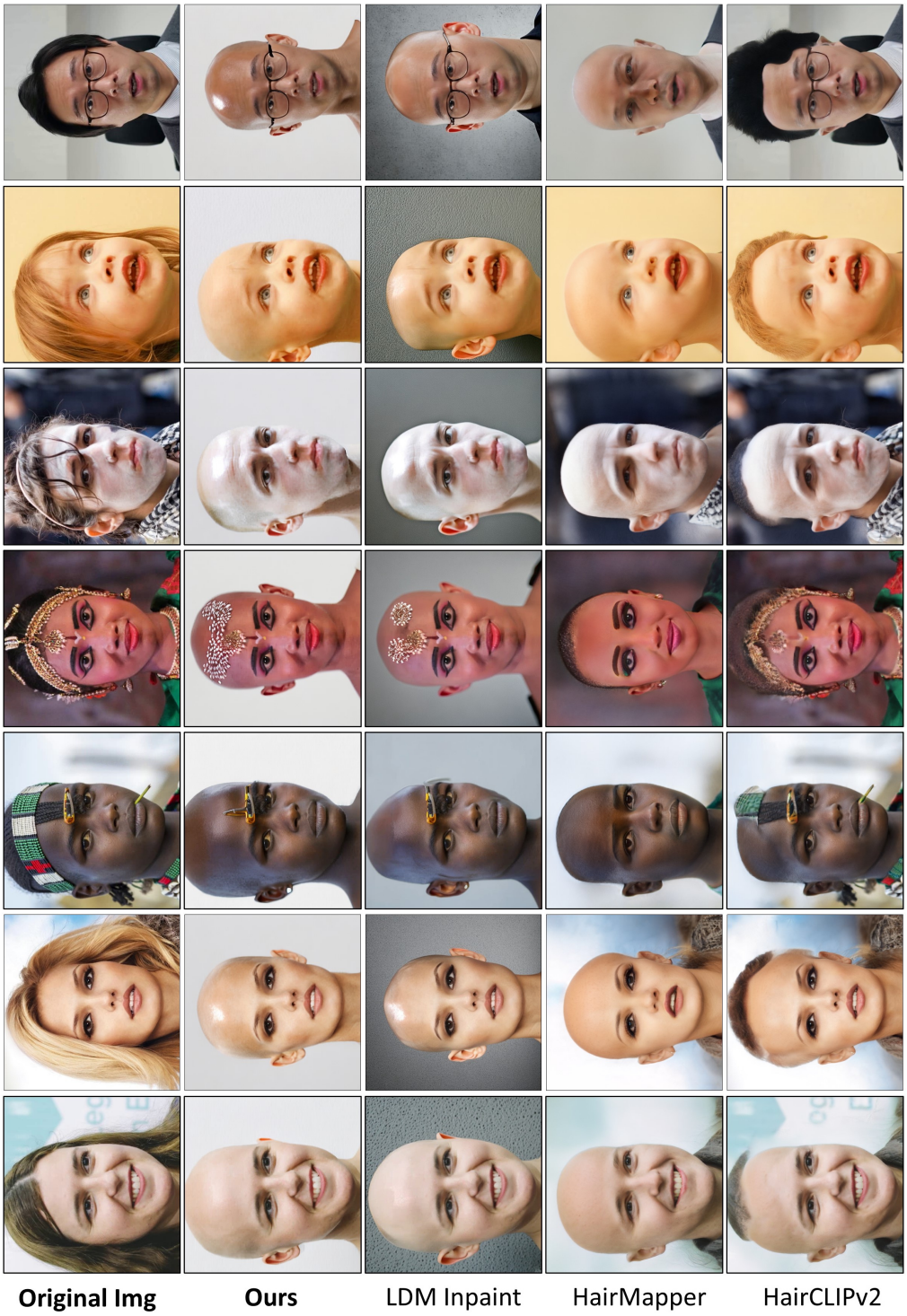

Figure 25: **Comparisons with State-of-the-Art 2D Bald Proxy Methods (I):** Our approach (with CS 0.45) is shown alongside bald proxy and hairstyle editing methods. LDM Inpaint [54] uses the SD 2.1 inpainting model with a full background mask, similar to ours. HairMapper [67] and HairCLIPv2 [66] (prompt: "bald") are bald proxy and hairstyle editing methods, though both degrade image quality; HairCLIPv2 generates hair due to limited bald data in training. Our method addresses this limitation, while preserving head shapes and poses more accurately.

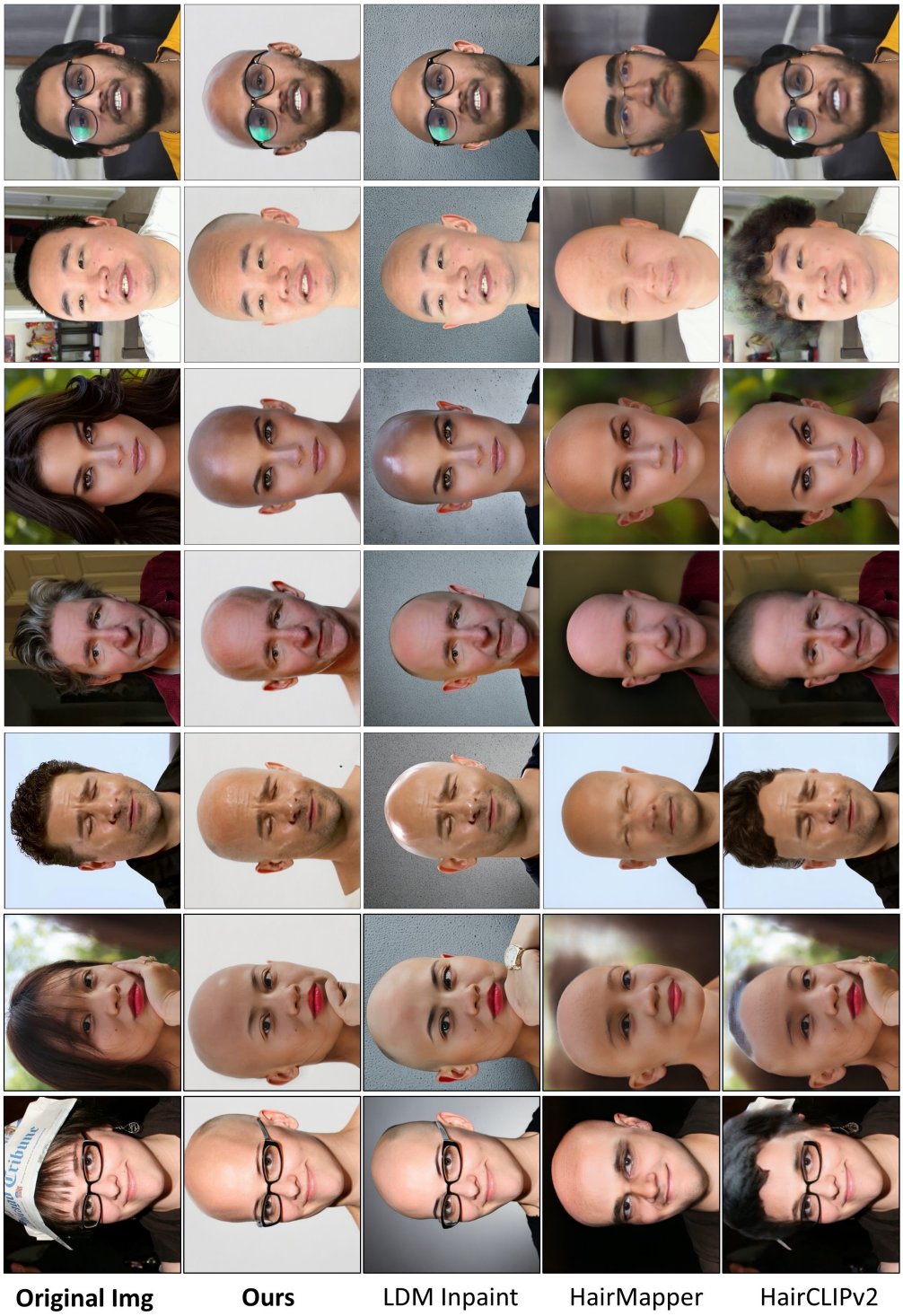

Figure 26: **Comparisons with State-of-the-Art 2D Bald Proxy Methods (II):** Our approach (with CS 0.45) is shown alongside bald proxy and hairstyle editing methods. LDM Inpaint [54] uses the SD 2.1 inpainting model with a full background mask, similar to ours. HairMapper [67] and HairCLIPv2 [66] (prompt: "bald") are bald proxy and hairstyle editing methods, though both degrade image quality; HairCLIPv2 generates hair due to limited bald data in training. Our method addresses this limitation, while preserving head shapes and poses more accurately.

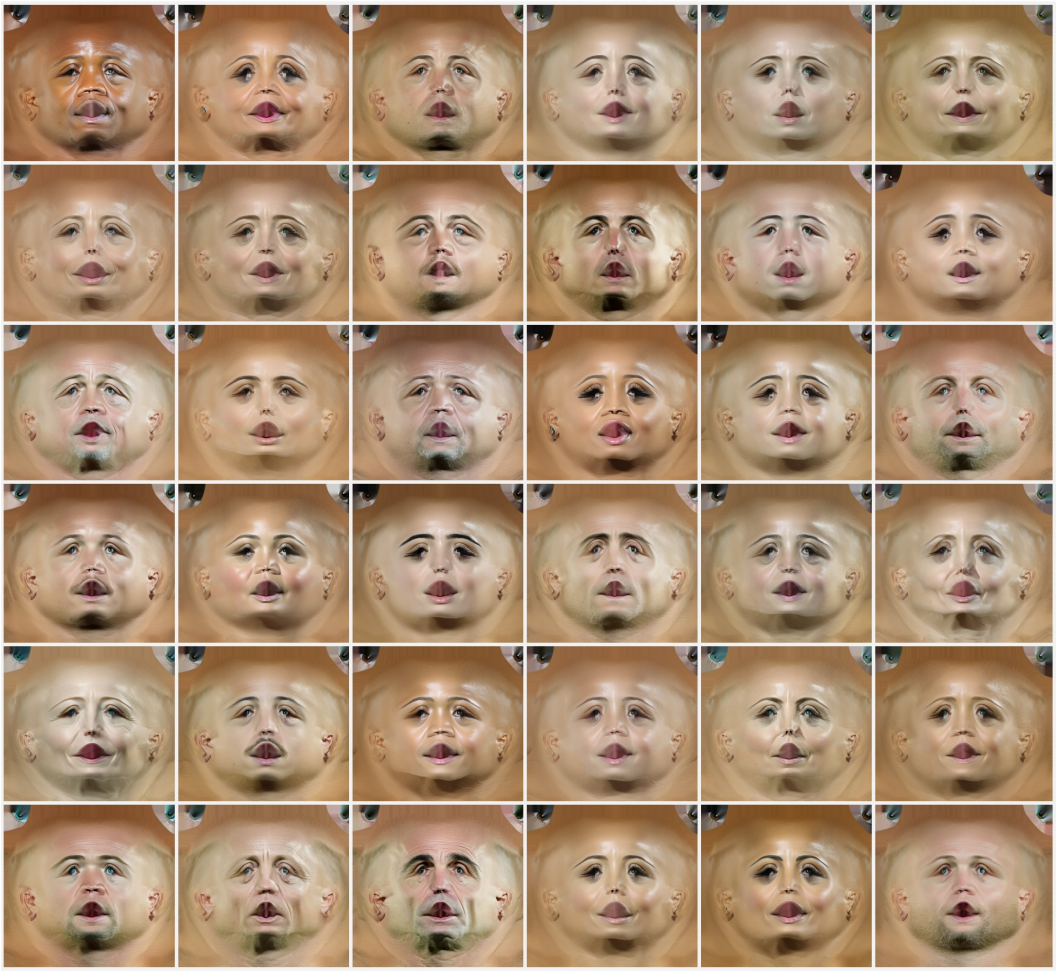

Figure 27: **Additional Texture Maps - Generated Results:** Examples of textures generated from text prompts of random celebrity names. The results include a diverse range of skin tones and facial features, highlighting the effectiveness of our method in generating a large variety of faces.

A.6 User Study

We ran a perceptual evaluation on Amazon Mechanical Turk to assess avatar plausibility. Each HIT presented a worker with four rendered views (frontal, two sides, and back) of avatars for two different identities, generated by 13 methods. Participants were asked, "Which head avatar looks most plausible?" and were paid $1 USD per completed subject. A screenshot of the study interface is shown in Figure 28.

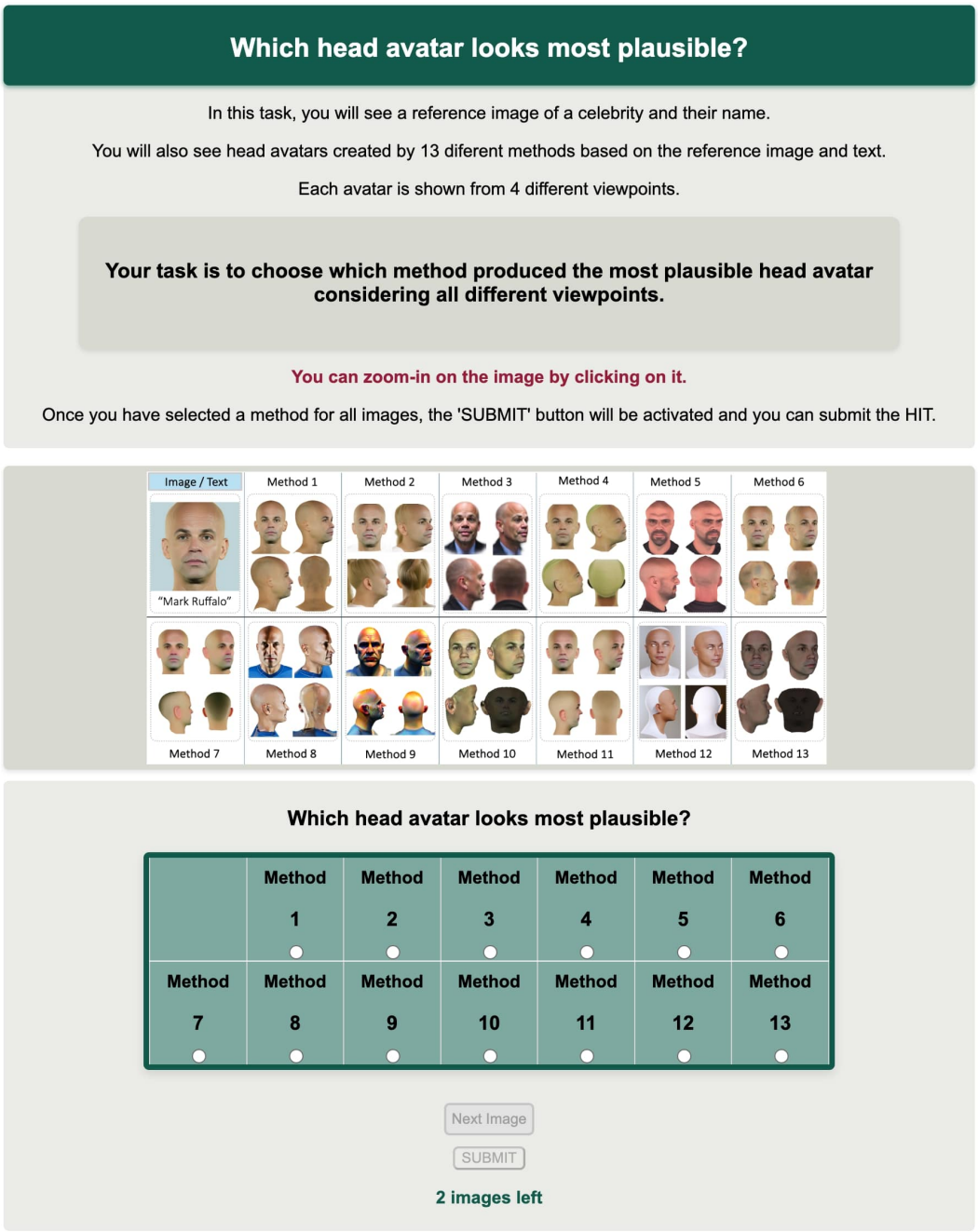

Figure 28: Layout of our user study.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: Yes, we thoroughly evaluate our method qualitatively, quantitatively, and through multiple ablation studies and a user study considering the claims.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Yes, a separate paragraph on Limitations exists.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [NA]

Justification: Our work does not contain theoretical results.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Yes, we disclose all the information needed to reproduce the main experimental results of the paper (see the Method section and main references, i.e. ControlNet, Stable Diffusion).

   Guidelines:

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

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

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

   Answer: [Yes]

Justification: We use a publicly available dataset of faces to fine-tune our diffusion prior. We will release our code, fine-tuned model, and a generative dataset of textures.

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Yes, for the diffusion fine-tuning part we follow the guidelines from the ControlNet and Stable Diffusion papers. The rest is described in the Method section.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: Yes, we quantitatively compare our method against state-of-the-art considering its run-time, method complexity, a high-level metric of image quality, and through a user study.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Yes, for the diffusion fine-tuning part we follow the guidelines from the ControlNet and Stable Diffusion papers. Time of execution is mentioned in the Experiments section.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: Yes, the paper follows the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We mention a potential malicious usecase of our method, and biases that are inherited from the large diffusion prior.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [NA]

    Justification: We use a standard publicly released dataset to fine-tune a prior model.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We cite all of the used datasets/code/models.

    Guidelines:

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

13. **New assets**

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

Answer: [No]

Justification: We will release our code, model, and a generative dataset (created by our method), as described in the paper (abstract, intro, method). We do not introduce any new assets to train our model.

Guidelines:

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

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

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

Answer: [Yes]

Justification: Yes, we provide details of our user study (Experiments section, Appendix).

Guidelines:

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

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

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

Answer: [NA]

Justification: Potential risks and IRB approval were not applicable to our user study.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [NA]

Justification: LLMs were not used for method development in our work.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.