Neural Haircut: Prior-Guided Strand-Based Hair Reconstruction

Vanessa Sklyarova[1] Jenya Chelishev[2,*] Andreea Dogaru[3,*] Igor Medvedev[1]
Victor Lempitsky[4] Egor Zakharov[1]

[1]Samsung AI Center [2]Rockstar Games [3]FAU Erlangen-Nürnberg [4]Cinemersive Labs

Multi-view images **Stage I: Volumetric hair reconstruction** **Stage II: Strand-based reconstruction**

Figure 1: We propose a two-stage pipeline for image-based hair reconstruction. Our first stage reconstructs coarse hair, head, and shoulder geometry using volumetric representations. The second stage fits hair strands to the coarse reconstruction via a joint optimization process that incorporates rendering-based losses and priors learned on the synthetic data.

Abstract

Generating realistic human 3D reconstructions using image or video data is essential for various communication and entertainment applications. While existing methods achieved impressive results for body and facial regions, realistic hair modeling still remains challenging due to its high mechanical complexity. This work proposes an approach capable of accurate hair geometry reconstruction at a strand level from a monocular video or multi-view images captured in uncontrolled lighting conditions. Our method has two stages, with the first stage performing joint reconstruction of coarse hair and bust shapes and hair orientation using implicit volumetric representations. The second stage then estimates a strand-level hair reconstruction by reconciling in a single optimization process the coarse volumetric constraints with hair strand and hairstyle priors learned from the synthetic data. To further increase the reconstruction fidelity, we incorporate image-based losses into the fitting process using a new differentiable renderer. The combined system, named Neural Haircut, achieves high realism and personalization of the reconstructed hairstyles. For video results, please refer to our project page[†].

1. Introduction

We propose a new image-based modeling method that recovers human hair from multi-view photographs or video frames. Hair reconstruction remains one of the most challenging problems in human 3D modeling because of its highly complex geometry, physics, and reflectance. Nevertheless, it is critical for many applications, such as special effects, telepresence, and gaming.

In computer graphics, the dominant representation for hair is 3D polylines, or *strands*, which can facilitate both realistic rendering and physics simulation [6]. At the same time, modern image- and video-based human reconstruction systems often model hairstyles using data structures that have fewer degrees of freedom and are easier to estimate, such as meshes with fixed topology [12, 20] or volumetric representations [4, 7, 9, 10, 11, 15, 27, 28, 31, 39, 40, 43, 48, 49, 56, 59, 60]. As a result, these methods often obtain over-smoothed hair geometries and can only model the "outer shell" of the hairstyle without its inner structure.

Accurate strand-based hair reconstruction can be accomplished via controlled lighting equipment and dense capture setup with synchronized cameras, i.e. using *light stages* [8]. Recently, impressive results were achieved [30, 33, 44, 50, 51] by relying on uniform or structured lighting and camera calibration to facilitate the reconstruction process. The latest work [44] further utilized manual frame-wise annotation

* Work done at Samsung AI Center
† https://samsunglabs.github.io/NeuralHaircut/

of the hair growth directions to achieve physically plausible reconstructions. However, despite the impressive quality of the results, the sophisticated capture setup and the manual pre-processing requirements make such methods unsuitable for many practical applications. Some learning-based methods for hairstyle modeling [5, 16, 22, 45, 54, 55, 58, 61] incorporate hair priors learned from the strand-based synthetic data to ease the acquisition process. However, the accuracy of these methods naturally depends on the size of the training dataset. Existing datasets [16, 54] typically consist of only a few hundred samples and are inadequately small for handling the diversity of human hairstyles, leading to the low fidelity of the reconstructions.

In this work, we propose a method for hair modeling that uses only image- or video-based data without any additional manual annotations and works in uncontrolled lighting conditions. To achieve that, we have designed a two-stage reconstruction pipeline. The first stage, *coarse* volumetric hair reconstruction, employs implicit volumetric representations and is purely data-driven. The second stage, *fine* strand-based reconstruction, operates at the level of hair strands and relies heavily on priors learned from a small-scale synthetic dataset.

During the first stage, we reconstruct implicit surface representations [38] for hair and bust (head and shoulders) regions. Additionally, we learn a field of hair growth directions, which we call 3D orientations, by matching them through a differentiable projection with hair directions observed in the training images or 2D orientation maps. While this field can facilitate a more accurate hair shape fitting, its primary use case is to constrain the optimization of hair strands during the second stage. To calculate the hair orientation maps from the input frames, we use a classic approach based on image gradients [37].

The second stage relies on pre-trained priors to obtain strand-based reconstructions. We employ an improved parametric model learned from the synthetic data using an auto-encoder [44] to represent individual strands and combine it with a new diffusion-based prior [14, 18] to model their joint distribution, i.e. a complete hairstyle. This stage thus reconciles the coarse hair reconstruction obtained in the first stage with the learning-based priors through an optimization process. Lastly, we improve the fidelity of reconstructed hairstyles via differentiable rendering using a new hair renderer based on soft rasterization [26].

To summarize, our contributions are:

- Human head 3D reconstruction method for bust and hair regions, which includes hair orientations;
- Improved training procedure for the strand prior;
- Latent diffusion-based prior for global hairstyle modeling, which "interfaces" with the parametric strand prior;
- Differentiable soft hair rasterization technique that leads to more accurate reconstructions than the previous rendering methods;
- Strand-fitting process that incorporates all the components discussed above to produce high-quality reconstructions of human hair at the level of strands.

We validate the efficacy of our method on synthetic [57] and real-world data, for which we use multi-view images from a 3D scanner operating in unconstrained lighting conditions [43] and monocular videos from a smartphone.

2. Related work

Human head reconstruction. Modern approaches have achieved impressive results in modeling static and dynamic human subjects using image and video data. These methods primarily rely on shape priors trained using synthetic datasets or 3D scans. Among the most widespread priors are parametric models [24, 29, 35, 36, 41] that represent the head as a rigged mesh with a fixed topology. However, these models do not include hair, as it is known to be notoriously difficult to scan. This sparked the development of methods that extend parametric models to include hair using volumetric representations and image or video-based finetuning [10, 12, 20, 49, 59, 60] or, more recently, using higher quality 3D scanners [11, 43]. While successfully reconstructing the facial geometry, these methods still achieve low fidelity of hair. They also model only the visible *outer* hair surface and do not reconstruct the inner geometry, limiting the downstream applications.

Hair reconstruction using volumetric representations still has important advantages. Modern volumetric reconstruction methods can handle challenging lighting conditions due to view-dependent modeling of radiance [32]. They can also hallucinate the geometry of the outer surface regions unseen in the training samples [13, 48, 56], which is useful for reconstruction using video data. We found volumetric reconstruction methods ideal for use in the first stage, during coarse hair modeling. We further extend them to represent hair and bust geometries separately, which is not done in the previous works. That allows us to incorporate additional supervision for the geometry and more effectively constrain hair strand optimization during the second stage. In addition, the separate bust geometry is used in an auxiliary way to introduce proper occlusion handling into the volumetric and strand-based hair rendering.

Strand-based hair reconstruction. Starting with the seminal work [37] on hair reconstruction, image-based methods [16, 22, 33, 44, 45, 54, 55, 61] have relied on hair orientation maps [37], or gradients in the image space, to estimate 3D hair strands. These orientation maps can quite effectively bridge the sim-to-real gap and allow some of these methods [22, 45, 61] to train using *only* synthetic data, while still generalizing to the real-world images. However, they have multiple practical issues. In order to obtain high-quality orientation maps, hair must be uniformly lit

and have no specular highlights. This assumption is quite strong and allows impressive results [33, 44] to be achieved using light stages for data capture. However, it limits the practicality of such methods for real-world reconstruction cases where lighting conditions are hard to control. Non-uniform lighting and low effective resolution of the real-world data lead to orientation maps having excessive noise levels or lacking details. Furthermore, orientation maps encode "sign-free" local orientations and do not capture the hair growth direction, which adds extra ambiguity to the reconstruction process.

Some methods [16, 44] address this using manual annotation of the exact growth direction, which makes the reconstruction process labor-intensive. Others [5, 22] employed regressors trained via manual annotation to predict the hair growth directions. However, the solution and the dataset presented in these works are closed-source and remain challenging and costly to reproduce. Lastly, the majority of the strand-based reconstruction method [5, 22, 33, 45, 54, 55, 58, 61] model hair strands without explicitly attaching them to the head scalp, which limits the realism of the resulting reconstructions. Our method addresses all these issues by introducing new hairstyle priors that ensure the physical realism of the reconstructed strands and a new coarse-to-fine optimization pipeline that uses prior-guided optimization and differentiable rendering to obtain personalized reconstructions even in non-uniform lighting conditions.

3. Method

3.1. Overview

We reconstruct the strand-based hair geometry given a single monocular video or multi-view images in the form of polylines in 3D: $\mathbf{S} = \{\mathbf{p}^l\}_{l=1}^{L}$. Our hair reconstruction pipeline consists of two stages. First, we obtain a coarse volumetric hair reconstruction in the form of implicit fields. We then reconstruct fine hair strands using optimization of coarse geometry-based, rendering-based, and prior-based terms. The hairstyle prior is obtained separately during pre-training on a synthetic dataset.

Hair prior training. Following [44], we parameterize the hairstyle using a *latent geometry texture* defined on the head scalp and denoted as $\mathbf{T}$. The mapping between hair strands and their latent embeddings is provided by the hair parametric model. It has the same architecture and training procedure as the original approach [44], besides a modified data term that improves the fidelity of curly hair reconstructions. We denote the decoder that produces strands given their latent embeddings as $\mathcal{G}$ and an encoder as $\mathcal{E}$.

We then train a latent diffusion-based prior $\mathcal{D}$, defined on the geometry texture maps $\mathbf{T}$. We use EDM [18] formulation that outperforms previous approaches such as DDPM [14]. We introduce multiple data augmentations that preserve the realism of the hairstyle while training on a small dataset of hairstyles [16] consisting only of a few hundred samples.

Stage I: coarse volumetric reconstruction. We approach coarse reconstruction by estimating hair and bust geometry as signed distance functions (SDFs) $f_{\text{hair}}, f_{\text{bust}} : \mathbb{R}^3 \to \mathbb{R}$. We train them via volumetric ray marching [48] using a shared view-dependent color field $c : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$. We employ supervision via semantic segmentation masks to ensure that hair and bust regions are non-overlapping. Also, to correctly reconstruct the head scalp, which is typically not visible on training samples, we fit a FLAME [24] head mesh to the scene and use it as a prior for the bust SDF. Lastly, to facilitate strand-based reconstruction, we train an additional field of 3D hair orientations $\beta : \mathbb{R}^3 \to \mathbb{S}^2$ using the hair signed distance function and match its projections with observed hair strand orientations.

Stage II: fine strand-based reconstruction. We reconstruct hair strands as a geometry texture $\mathbf{T}$, i.e. a dense two-dimensional map of latent hair vectors, where the position on the map corresponds to the position of the hair root on the scalp. At each iteration, we sample N random embeddings $\{\mathbf{z}_i\}_{i=1}^{N}$ from the texture $\mathbf{T}$ and obtain corresponding strands $\{\mathbf{S}_i\}_{i=1}^{N}$ using a pre-trained decoder $\mathcal{G}$. These strands are then used to evaluate geometric and rendering-based constraints. In the geometric loss, we penalize strands outside the hair volume and ensure that the visible part of the surface defined by f_{hair} is uniformly covered. We also match the orientations $\mathbf{b}_i^l$ of the predicted strands, defined as the normalized difference between two consecutive points, to the orientation field β. Here, $\mathbf{b}_i^l = \mathbf{d}_i^l / \|\mathbf{d}_i^l\|_2$ and $\mathbf{d}_i^l = \mathbf{p}_i^{l+1} - \mathbf{p}_i^l$.

Besides the geometric constraints, we also employ silhouette-based and neural rendering losses. The rendered hair silhouette $\hat{\mathbf{m}}$ and RGB image $\hat{\mathbf{I}}$ are then obtained using neural soft hair rasterization denoted as $\mathcal{R}$. The renderer employs a bust surface estimated from f_{bust} to handle occlusions. The silhouette $\hat{\mathbf{m}}$ is predicted directly from the sampled strands, while the image render is obtained via a neural hair rendering pipeline inspired by [44].

Lastly, prior-based regularization is applied directly to the geometry texture $\mathbf{T}$ using a pre-trained diffusion model. Specifically, we apply random noise to the geometry map and denoise it using a diffusion model $\mathcal{D}$. We then evaluate the reconstruction error of the input map $\mathbf{T}$ and back-propagate the gradient of this loss back into the texture. This pipeline is inspired by the DreamFusion method [42], albeit with some modifications which facilitate training from a small dataset of hairstyles.

The scheme of the fine reconstruction stage is shown in Figure 2. Below, we describe the parts of our approach in more detail.

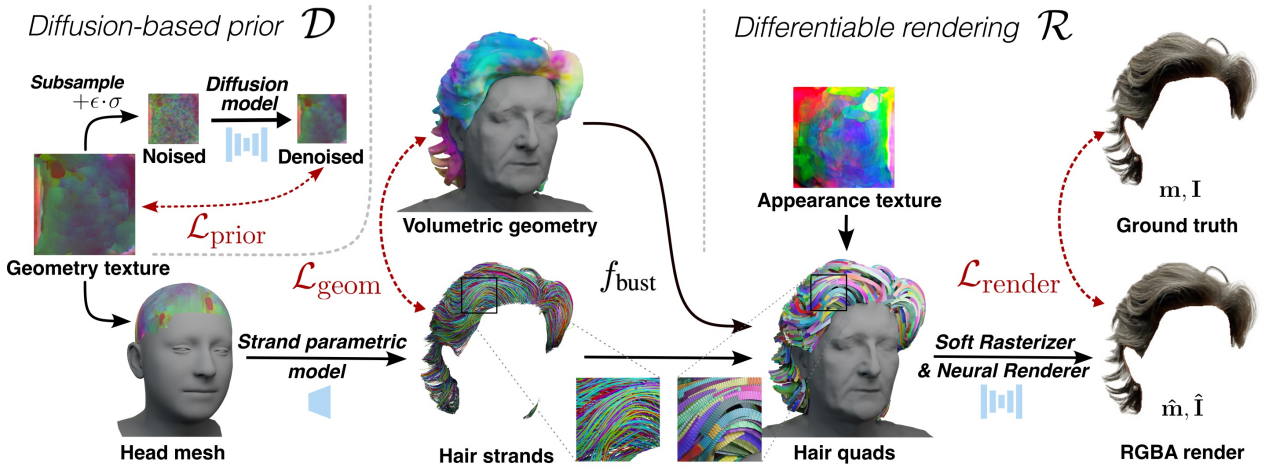

Figure 2: The overview of the second stage of our approach (fine strand-based reconstruction). We use shape texture to represent hair strands and utilize multiple objectives to optimize it. We apply $\mathcal{L}_{\text{prior}}$ as a regularization penalty using a diffusion network pre-trained on synthetic hairstyles. Then, we use $\mathcal{L}_{\text{geom}}$ to match the reconstructed strands to geometry and orientation fields parameterized by the implicit function. Lastly, $\mathcal{L}_{\text{render}}$ is used to match the rendered hair to the ground truth image.

3.2. Hair prior training

Our global hairstyle prior is trained using a strand parametric model and a latent diffusion network, which interface with each other via the geometry texture $\mathbf{T}$.

Hair strand parametric model. To obtain the latent representations for the hair strands, we follow [44] and train a variational autoencoder, which maps a strand $\mathbf{S} = \{\mathbf{p}^l\}_{l=1}^L$ into a latent vector $\mathbf{z}$ via encoder $\mathcal{E}$, and back via decoder $\mathcal{G}$. During training, we employ the reparameterization trick: $\mathbf{z} = \mathbf{z}_\mu + \epsilon \cdot \mathbf{z}_\sigma$, where $\mathbf{z}_\mu$ and $\mathbf{z}_\sigma$ are parameters of the Gaussian distribution, and $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The model is trained using the data term $\mathcal{L}_{\text{data}}$ and KL divergence [21] term $\mathcal{L}_{\text{KL}}$ on a synthetic dataset of hair strands. The data term consists of an L2 error between predicted $\hat{\mathbf{p}}^l$ and ground-truth points $\mathbf{p}^l$, as well as the cosine distance for the orientations $\hat{\mathbf{b}}^l$ and $\mathbf{b}^l$, similarly to [44]. Inspired by [61], we additionally match curvatures $\hat{\mathbf{g}}^l$ and $\mathbf{g}^l$ of predicted and ground-truth strands to better model curly hair, where

$$\mathbf{g}^l = \left\| \mathbf{b}^l \times \mathbf{b}^{l+1} \right\|_2. \tag{1}$$

Our data term is therefore defined as follows:

$$\begin{aligned} \mathcal{L}_{\text{data}} = \sum_{l=1}^{L} \left\| \hat{\mathbf{p}}^l - \mathbf{p}^l \right\|_2^2 + \lambda_d \left(1 - \hat{\mathbf{b}}^l \mathbf{b}^l\right) \\ + \lambda_c \left\| \hat{\mathbf{g}}^l - \mathbf{g}^l \right\|_2^2, \end{aligned} \tag{2}$$

and the final loss is:

$$\mathcal{L}_{\text{VAE}} = \mathcal{L}_{\text{data}} + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}} \left(\mathcal{N}(\mathbf{z}_\mu, \mathbf{z}_\sigma) \,\middle\|\, \mathcal{N}(\mathbf{0}, \mathbf{I}) \right). \tag{3}$$

Hairstyle diffusion model. We obtain the latent representation of the synthetic hairstyle consisting of N strands $\{\mathbf{S}_i\}_{i=1}^N$ by first estimating their latent descriptors $\{\mathbf{z}_i\}_{i=1}^N$ using the pre-trained strand encoder $\mathcal{E}$. We then convert them into a dense texture $\mathbf{T}$ using nearest neighbor interpolation. To increase the diversity of training samples, we employ augmentations that preserve the realism of the hairstyle before encoding. Also, to speed up the training and further diversify the inputs of the diffusion model, we subsample the full texture into a low-resolution map $\mathbf{T}_{\text{LR}}$.

We use the Elucidating Diffusion Moldel (EDM) [18] to train the denoiser $\mathcal{D}$. Below, we denote a training sample $\mathbf{T}_{\text{LR}}$ as $\mathbf{y}$ to be consistent with [18], and obtain a noised input: $\mathbf{x} = \mathbf{y} + \epsilon \cdot \sigma$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and σ is a noise strength. We then predict a denoised input:

$$\mathcal{D}(\mathbf{x}, \sigma) = c_{\text{skip}}(\sigma) \cdot \mathbf{x} + c_{\text{out}}(\sigma) \cdot \mathcal{F}\big(c_{\text{in}}(\sigma) \cdot \mathbf{x}, c_{\text{noise}}(\sigma)\big), \tag{4}$$

where the c_{skip}, c_{out}, c_{in} and c_{noise} are part of preconditioning approach proposed in [18], which improves the robustness of $\mathcal{D}$ to the low noise strength σ, and $\mathcal{F}$ is a neural network. Our training objective also follows [18]:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{y}, \sigma, \epsilon} \left[\lambda_{\text{diff}}(\sigma) \cdot \left\| \mathcal{D}(\mathbf{x}, \sigma) - \mathbf{y} \right\|_2^2 \right], \tag{5}$$

where $\lambda_{\text{diff}}(\sigma)$ is a weighting function, and the expectation is approximated via sampling.

3.3. Coarse volumetric reconstruction

We represent the subject's coarse head geometry in a segmented form using the hair and the bust neural signed distance functions [38] (SDFs). We use the volumetric ray marching approach for neural implicit surfaces, NeuS [48] to fit them. We modify NeuS to accommodate multi-label

reconstruction as we reconstruct the hair geometry and the bust (head and shoulders) geometry as separate shapes. The training proceeds by approximating a pixel's color $\mathbf{c}$ using the radiance at N points $\mathbf{x}_i$ sampled along the corresponding ray $\mathbf{v}$. The color is predicted as follows:

$$\hat{\mathbf{c}} = \sum_{i=1}^{N} T_i \cdot \alpha_i \cdot c\left(\mathbf{x}_i, \mathbf{v}, \mathbf{l}, \mathbf{n}\right), \quad T_i = \prod_{j=1}^{i-1}(1-\alpha_i), \quad (6)$$

where T_i is the accumulated transmittance, α_i is the opacity, $\mathbf{l}$ and $\mathbf{n}$ - the blended hair with bust features and normals correspondingly, and c is the view-dependent radiance field. We calculate the opacity α_i of each point along the ray by blending the individual opacities of hair and bust:

$$\alpha_i = \min\left(\alpha_i^{\text{hair}} + \alpha_i^{\text{bust}}, 1\right). \quad (7)$$

Besides the color, we also render the bust and the hair masks:

$$\hat{\mathbf{o}}_{\text{hair}} = \sum_{i=1}^{N} T_i \cdot \alpha_i^{\text{hair}}, \quad \hat{\mathbf{o}}_{\text{bust}} = \sum_{i=1}^{N} T_i \cdot \alpha_i^{\text{bust}}. \quad (8)$$

Our training losses include a photometric L1 loss $\mathcal{L}_{\text{color}}$, which matches $\hat{\mathbf{c}}$ and $\mathbf{c}$, a mask-based loss $\mathcal{L}_{\text{mask}}$ that applies binary cross-entropy between the predicted masks and the ground-truth $\mathbf{m}_{\text{hair}}$ and $\mathbf{m}_{\text{bust}}$, and the regularizing Eikonal term [13] $\mathcal{L}_{\text{reg}}$, which is applied for both f_{hair} and f_{bust}.

Our additional losses include a regularization for the bust shape. Before proceeding with the coarse reconstruction of the subject, we fit a FLAME [24] head mesh into the scene using optimization based on 2D facial landmarks [3]. Using this mesh, we ensure that f_{bust} includes the head scalp surface region by applying the regularizing constraints denoted as $\mathcal{L}_{\text{head}}$ that match the SDF to the mesh. To implement this loss, we follow the previous works [1, 13, 46] on fitting neural SDFs using mesh-based data and provide its full description in the supplementary materials.

Lastly, we incorporate an additional field of hair growth directions, β, into the coarse reconstruction. We train it via a differentiable surface rendering [9] of f_{hair}. Following [9], we obtain the intersection point $\mathbf{x}_{\text{s}}$ of the ray $\mathbf{v}$ with the hair surface. We then project the 3D orientation field $\beta(\mathbf{x}_{\text{s}})$ into the camera $\mathcal{P}$ using Plucker line coordinates [53]. The projected direction $\mathbf{l}(\mathbf{x}_s, \beta(\mathbf{x}_{\text{s}}); \mathcal{P})$ is then matched to the 2D orientation map [37], estimated from the training images using Gabor filters. The matching loss $\mathcal{L}_{\text{dir}}$ follows previous works [37] on strand-based reconstruction and penalizes the minimum angular difference between the projected and ground-truth orientations. Please refer to the supplementary materials for more details.

Overall, the training objective for the coarse reconstruction is as follows:

$$\begin{aligned}\mathcal{L}_{\text{coarse}} = \mathcal{L}_{\text{color}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}} \\ + \lambda_{\text{head}}\mathcal{L}_{\text{head}} + \lambda_{\text{dir}}\mathcal{L}_{\text{dir}}.\end{aligned} \quad (9)$$

3.4. Fine strand-based reconstruction

To reconstruct the hair strands, we learn a latent hair geometry texture $\mathbf{T}$ [44], from which a hairstyle can be decoded using a pre-trained network $\mathcal{G}$. However, instead of directly optimizing this map, we parameterize it with a UNet-like neural network using the so-called deep image prior [47]. We found such parameterization to not require additional smoothing [44] of the sparse gradients from the decoded strands. Below, we denote such new parameterization as $\mathbf{T}_\theta$.

The training proceeds by sampling N points on the scalp part of the fitted FLAME mesh and decoding them into strands $\{\mathbf{S}_i\}_{i=1}^{N}$, each strand consisting of L points: $\mathbf{S}_i = \{\mathbf{p}_i^l\}_{l=1}^{L}$. We then evaluate the following objectives: geometry-based losses $\mathcal{L}_{\text{geom}}$ that match the strands to the coarse geometry, photometric constraints $\mathcal{L}_{\text{render}}$ calculated via differentiable rendering, and finally, a diffusion-based prior loss $\mathcal{L}_{\text{prior}}$. Below we describe them in more detail.

Geometry-based losses. To ensure that the optimized strands lie inside the coarse hair volume, we employ a loss $\mathcal{L}_{\text{vol}}$ that penalizes the points on the strands that stray outside of it:

$$\mathcal{L}_{\text{vol}} = \sum_{i=1}^{N}\sum_{l=1}^{L} \mathbb{I}\left[f_{\text{hair}}(\mathbf{p}_i^l) > 0\right]\left(f_{\text{hair}}(\mathbf{p}_i^l)\right)^2, \quad (10)$$

where $\mathbb{I}$ denotes the indicator function.

Additionally, to make the learned strands densely cover the visible part of the coarse hair surface, denoted as $\mathcal{S}$, we minimize the error between K random points $\mathbf{x}_k$ sampled on this surface and their nearest points on the strands, denoted as $\mathbf{p}_k$. This loss $\mathcal{L}_{\text{chm}}$ is exactly equal to the one-way Chamfer distance between the visible part of the coarse hair surface and the learned strands:

$$\mathcal{L}_{\text{chm}} = \sum_{k=1}^{K} \left\|\mathbf{x}_k - \mathbf{p}_k\right\|_2^2, \quad (11)$$

Lastly, we calculate the distance between the hair orientations and the implicit field β at all points on the strands that are closer to the visible hair surface $\mathcal{S}$ than some small threshold τ. We denote these M points as $\mathbf{p}_m$ and their orientations as $\mathbf{b}_m$. The resulting orientations loss $\mathcal{L}_{\text{orient}}$ can be written as follows:

$$\mathcal{L}_{\text{orient}} = \sum_{m=1}^{M}\left(1 - \left|\mathbf{b}_m \cdot \beta(\mathbf{p}_m)\right|\right). \quad (12)$$

We penalize the orientations of strands near the outer hair surface because the photometric nature of the orientation loss $\mathcal{L}_{\text{dir}}$ makes the field β learn accurate orientations only in this region. We describe the procedure for estimating this surface using f_{hair} and f_{bust} in the supplementary materials.

Overall, the total geometry loss is the following:

$$\mathcal{L}_{\text{geom}} = \mathcal{L}_{\text{vol}} + \lambda_{\text{chm}}\mathcal{L}_{\text{chm}} + \lambda_{\text{orient}}\mathcal{L}_{\text{orient}}. \quad (13)$$

Rendering-based losses. We have developed a new approach for the differential rendering of hair strands to improve the visible hair geometry. We note that the previous hair rasterization approaches [44] rely on graphics API [52] line rasterization algorithms, e.g. Bresenham's line algorithm [2]. While being computationally efficient, such methods only provide the gradients w.r.t. the first element of the line segments z-buffer, Figure 3 (a). At the same time, for the task of mesh inverse rendering, it was shown to be highly beneficial [26] to propagate the gradient into multiple z-buffer elements. Inspired by the success of this *soft rasterization* method [26] for meshes, we adapt it for the differentiable rendering of hair strands Figure 3 (b).

First, we convert the hair strands into the so-called *hair quads* [57]. They consist of a stripe-like mesh, which follows the strand trajectory and has normals oriented toward the camera, see close-ups in Figure 2. The vertices of the resulting quad mesh are fully differentiable w.r.t. the strands, and we include the quad generation algorithm into the supplementary materials. We then render this mesh using soft rasterization. We include the zero iso-surface of f_{bust} obtained using Marching Cubes [23] into the rendering pipeline to handle hair-bust occlusions. Contrary to the previous rasterization methods, in our approach the segmentation mask for the hair is *directly* predicted from the hair geometry using a soft silhouette shader [17], which allows unconstrained gradient flow into the geometry from the mask-based objectives. To render the color, we follow [44] and use a neural rendering approach that can handle the view-dependent reflectance of the hair. Specifically, we train a neural appearance texture $\mathbf{A}$ similarly to the geometry texture $\mathbf{T}$ and use it in conjunction with a rendering U-Net to produce the renders, similarly to [44].

As a result of the hair rasterization pipeline $\mathcal{R}$ described above, we obtain both the hair silhouette $\hat{\mathbf{m}}$ and the images $\hat{\mathbf{I}}$ in a fully differentiable way:

$$\hat{\mathbf{m}}, \hat{\mathbf{I}} = \mathcal{R}_{\phi}\big(\{\mathbf{S}_i\}_{i=1}^{N}, f_{\text{bust}}, \mathcal{P}\big), \quad (14)$$

where ϕ denotes the trainable parameters of the appearance texture and a rendering UNet, and $\mathcal{P}$ are the camera parameters. We then apply L1 losses $\mathcal{L}_{\text{mask}}$ and $\mathcal{L}_{\text{rgb}}$ to match the predicted silhouette and the color to the ground truth $\mathbf{m}$ and $\mathbf{I}$. The final rendering loss is the weighted sum of these terms:

$$\mathcal{L}_{\text{render}} = \mathcal{L}_{\text{rgb}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}. \quad (15)$$

Diffusion-based prior. To apply the pre-trained diffusion prior, we use a Score Distillation Sampling (SDS) approach from the DreamFusion work [42]. In this method, the pre-trained diffusion model is used to guide the optimization of a neural radiance field [32] by providing it with the gradients in the image space. These gradients originate from the same loss used to train a diffusion model, in our case, $\mathcal{L}_{\text{diff}}$, Eq. 5. However, instead of back-propagating this loss through the denoising neural network $\mathcal{F}$, the SDS approach assumes the gradients w.r.t. the noised input $\mathbf{x}$ to be identity: $\partial\mathcal{F}/\partial\mathbf{x} = \mathcal{I}$. However, we found such a trick to be required only for DDPM [14] training formulation used in DreamFusion, while for the EDM [18] that we use, proper back-propagation through the denoising network $\mathcal{F}$ leads to better results. Therefore, in our case, the prior regularization term $\mathcal{L}_{\text{prior}} \equiv \mathcal{L}_{\text{diff}}$.

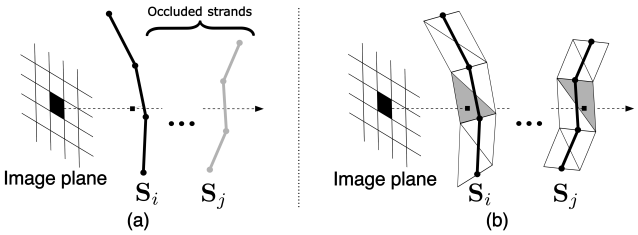

Figure 3: (a) Differentiable hair rasterization algorithm of [44] propagates the gradient only into the first element of z-buffer. (b) Our proposed hair rasterization based on quads leverages soft hair rasterization [26] and passes gradients into multiple elements of the z-buffer to achieve better reconstructions.

To calculate this loss, we employ the same procedure as during the training of the diffusion model. We sample random noise ϵ and the noise level σ and apply them to the geometry map. Then, we perform random sub-sampling to decrease the resolution of $\mathbf{T}_\theta$ before forwarding it through the diffusion model. We back-propagate the loss $\mathcal{L}_{\text{prior}}$ directly into the parameters θ of the geometry texture $\mathbf{T}_\theta$ while keeping the weights of the denoiser frozen.

Overall, the optimization objective for the strand-based reconstruction stage is the following:

$$\mathcal{L}_{\text{fine}} = \mathcal{L}_{\text{geom}} + \lambda_{\text{render}}\mathcal{L}_{\text{render}} + \lambda_{\text{prior}}\mathcal{L}_{\text{prior}}. \quad (16)$$

4. Experiments

We use the USC-HairSalon [16] dataset to pre-train the strand parametric model and hairstyle diffusion module. This dataset consists of 343 hairstyles aligned with the template bust mesh. We then evaluate our method using both synthetic and real-world data. We use two synthetic scenes [57] to conduct a quantitative comparison using the ground-truth strand-based geometry. For the real-world data, we use H3DS Dataset [43] of multi-view images with non-uniform lighting and monocular video data captured using a smartphone.

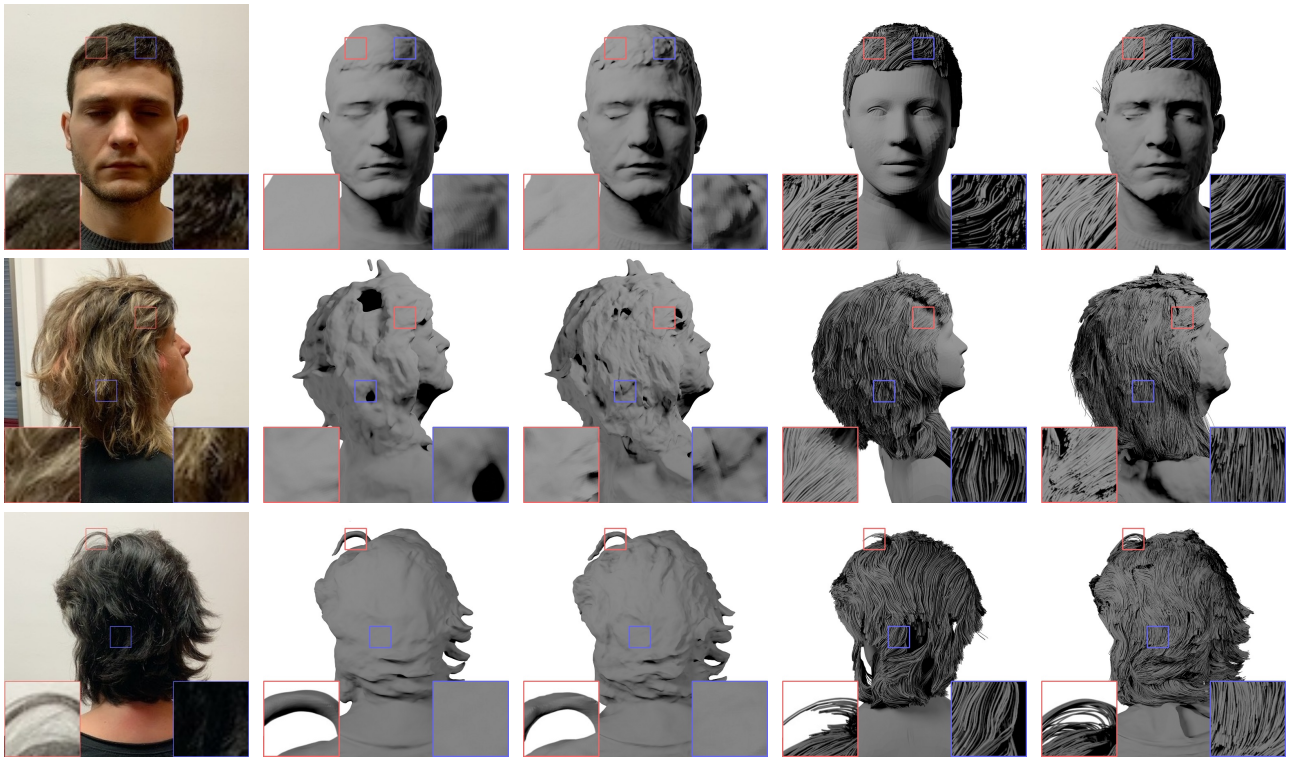

Figure 4: We compare our method with volumetric and strand-based 3D reconstruction systems using a real-world multi-view dataset [43]. While baseline volumetric approaches [34, 48] can only produce coarse hair geometry, our method is able to reconstruct fine details using strands. We also achieve more robust and accurate results than the existing multi-view hair reconstruction methods [22]. For additional results, please refer to the supplementary materials. Digital zoom-in is recommended.

4.1. Implementation details

To train our method on real-world data we use off-the-shelf methods [19, 25] to obtain segmentation masks for the hair and bust. To parameterize the geometry texture map, we use a UNet network that predicts it from a constant mesh grid. Similarly to the diffusion model training, we calculate the denoising error $\mathcal{L}_{\text{prior}}$ by averaging over the mini-batch of different offsets, noise ϵ, and noise levels σ. In total, our reconstruction pipeline takes three days per subject on a single NVIDIA RTX 4090: one day for the first, and two days for the second stage. For more training details and hyperparameters, please refer to the supplementary materials.

4.2. Real-world evaluation

Baselines. We compare our method against popular 3D reconstruction approaches [34, 48], as well as methods [22, 54] designed for strand-based reconstruction, using publicly available scenes from the H3DS [43] dataset. **NeuS** [48] is a multi-view reconstruction approach that learns the scene geometry as the zero level-set of a signed distance function using volume rendering. This method can reconstruct non-Lambertian surfaces, making it well-suited for hair reconstruction. **UNISURF** [34] is another multi-view approach based on occupancy fields learned via volume rendering. This method is specifically tailored to handle semi-transparent objects, such as hair. **DeepMVSHair** [22] is a multi-view image-guided method for realistic strand-based reconstruction that can operate in a sparse-view scenario and under non-uniform lighting conditions. Due to memory constraints, this method is trained to produce reconstructions using twelve views. We also provide comparisons with a single image **NeuralHDHair** [54] method in Figure 6.

The qualitative results are shown in Figure 4. Note that our segmented modeling approach allows us to reconstruct realistic hair alongside the accurate bust geometry. It sets us apart from all the baseline methods, which can only reconstruct coarse hair geometry. Furthermore, our approach is able to reconstruct strands in poorly visible regions.

Finally, we evaluate our method in a challenging case of monocular video capture. The results are provided in Figure 5. Our method is fully capable of handling this challenging use case and keeps the high fidelity and realism of the reconstructed hair. We include more examples in the supplementary materials.

Figure 5: Our method can obtain high-fidelity hair reconstructions even from a monocular video. For more results, please refer to the supplementary materials.

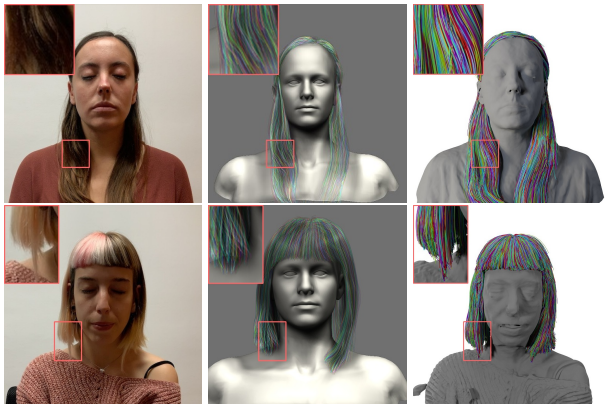

Figure 6: Comparison of our multi-view method (right) with a single-shot NeuralHDHair [54] system (middle). Digital zoom-in is recommended.

4.3. Ablation study

We conduct an ablation study using both synthetic and real-world datasets. Here, we also include the comparison with **Neural Strands** [44], since our approach, in some aspects, builds on top of it. However, we cannot compare our method against it directly as this method is closed-source, requires manual annotation of the hair growth directions, and relies on an MVS-based hair reconstruction method [33], which is sensitive to non-uniform lighting conditions and poorly handles hair specularities. We have re-implemented a strand generator network, a differentiable rasterizer, neural hair rendering, texture parametrization, and a training procedure. Due to the unavailability of the ground-truth 3D strand segments and hair growth directions, we replace their geometry loss with our $\mathcal{L}_{\text{geom}}$. We refer to the resulting method as Neural Strands*.

We employed the same approach for quantitative evaluation as in [44]. We first render the ground-truth strands using Blender [6] and reconstruct them. We then follow [33, 44] and measure precision, recall, and F-score between our predicted strands and ground truth using both distance and angular errors as thresholds. The comparison results are shown in Table 1. First, we see that our rendering loss $\mathcal{L}_{\text{rgb}}$ improves over the base $\mathcal{L}_{\text{render}}$ from [44] in terms of precision, while the rendering mask loss $\mathcal{L}_{\text{mask}}$ achieves better recall and an aggregated F-score.

Finally, our complete model $\mathcal{L}_{\text{fine}}$, which combines together geometric, rendering, and diffusion-based losses, further improves these results, achieving the highest recall and F-score across all experiments. In Figure 8, we show the comparison with Neural Strands*. The method suffers from unrealistic curls and artifacts in reconstructions. For more details please refer to the supplementary materials.

In Figure 7, we conduct a qualitative ablation study to evaluate the effect of a diffusion prior $\mathcal{L}_{\text{prior}}$ and a curvature loss term in the strand parametric model. Notice that the diffusion-based prior achieves substantially higher realism of the internal part of the hairstyle. For the curvature loss,

Method	Thresholds: mm / degrees								
	2/20	3/30	4/40	2/20	3/30	4/40	2/20	3/30	4/40
	Precision			**Recall**			**F-score**		
$\mathcal{L}_{\text{geom}}$	57.3	81.9	90.4	7.8	13.8	19.8	13.7	23.5	32.5
w/ $\mathcal{L}_{\text{render}}$ [44]	58.6	82.4	91.0	8.0	13.9	21.5	14.1	23.7	34.7
w/ $\mathcal{L}_{\text{rgb}}$	**60.5**	**83.2**	**91.5**	7.6	13.8	21.0	13.5	23.7	34.1
w/ $\mathcal{L}_{\text{mask}}$	56.5	81.5	90.4	8.7	14.7	21.0	15.0	24.9	34.1
$\mathcal{L}_{\text{fine}}$	52.9	78.1	88.4	**9.8**	**17.8**	**26.3**	**16.4**	**28.7**	**40.3**

Table 1: We provide an extensive quantitative evaluation of individual components of our method. Our complete model $\mathcal{L}_{\text{fine}}$, which combines together geometric, rendering, and diffusion-based losses, achieves the highest recall and F-score.

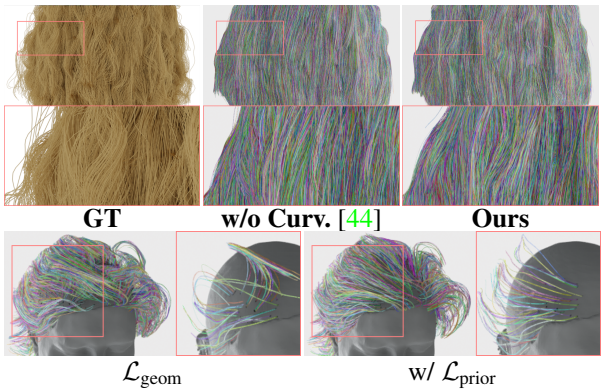

Figure 7: Ablation on curvature (top) and diffusion losses (bottom). The incorporation of curvature loss allows us to better model curly strands, while the diffusion tackles the problems with hair growth directions and unrealistic angles (insets show a subset of hairs for clarity).

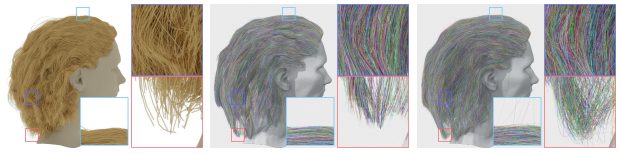

Figure 8: A comparison with Neural Strands [44]. Our method obtains higher-quality hairstyles that lack the unrealistic curls visible in the top and bottom parts of the [44] reconstruction.

its effect is visible when modeling curly hairstyles. Additional qualitative and quantitative results are provided in the supplementary materials.

4.4. Limitations

The main limitations of our method can be seen in Figure 9. Our approach struggles to represent highly curly hair and depends on the accuracy of hair and body segmentation masks to produce the reconstructions. In principle, it is possible to address these limitations by extending the dataset for the hairstyle prior training and employing more robust human matting systems. Furthermore, improving the rendering procedure for better signal propagation and conditioning prior network on hairstyle-related features are very promising directions.

Figure 9: Our system's main limitations are curly hairstyle modeling performance and reliance on hair segmentation masks during volumetric and strand-based reconstruction.

5. Conclusion

We have presented a method capable of detailed and personalized human hair reconstruction from monocular videos with uncontrolled lighting. To achieve that, we employ both volumetric and strand-based hair representations and combine them with differential hair rendering and global hairstyle priors. We demonstrate the efficacy of our approach by conducting extensive qualitative and quantitative evaluations. To the best of our knowledge, our method is the first to achieve detailed strand-based and personalized hair reconstructions from monocular video capture using a smartphone video. We believe this method can also be incorporated into future human avatar systems for better realism and identity preservation of human appearance, in which hairstyles play a vital part.

Acknowledgements

We thank Samsung ML Platform for providing the computational resources for this work. We also sincerely thank Zhiyi Kuang for aiding us with the DeepMVSHair comparison and Youyi Zheng — for providing the reconstructions of NeuralHDHair. We also thank David Svitov for his insightful suggestions on diffusion models.

References

[1] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2562–2571, 2019. 5

[2] J. E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4(1):25–30, 1965. 6

[3] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *International Conference on Computer Vision*, 2017. 5

[4] Chen Cao, Tomas Simon, Jin Kyu Kim, Gabriel Schwartz, Michael Zollhoefer, Shunsuke Saito, Stephen Lombardi, Shih-En Wei, Danielle Belko, Shoou-I Yu, Yaser Sheikh, and Jason M. Saragih. Authentic volumetric avatars from a phone scan. *ACM Transactions on Graphics (TOG)*, 41:1 – 19, 2022. 1

[5] Menglei Chai, Tianjia Shao, Hongzhi Wu, Yanlin Weng, and Kun Zhou. Autohair: fully automatic hair modeling from a single image. *ACM Trans. Graph.*, 35:116:1–116:12, 2016. 2, 3

[6] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2023. 1, 8

[7] Franccois Darmon, B. Bascle, Jean-Clement Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6250–6259, 2021. 1

[8] Paul E. Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000. 1

[9] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 1, 5

[10] Guy Gafni, Justus Thies, Michael Zollhofer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8645–8654, 2020. 1, 2

[11] Simon Giebenhain, Tobias Kirschstein, Markos Georgopoulos, Martin Rünz, Lourdes de Agapito, and Matthias Nießner. Learning neural parametric head models. *ArXiv*, abs/2212.02761, 2022. 1, 2

[12] Philip-William Grassal, Malte Prinzler, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. Neural head avatars from monocular rgb videos. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18632–18643, 2021. 1, 2

[13] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *International Conference on Machine Learning*, 2020. 2, 5

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. 2, 3, 6

[15] Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. Headnerf: A realtime nerf-based parametric head model. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20342–20352, 2021. 1

[16] Liwen Hu, Chongyang Ma, Linjie Luo, and Hao Li. Single-view hair modeling using a hairstyle database. *ACM Transactions on Graphics (TOG)*, 34:1 – 9, 2015. 2, 3, 6

[17] Justin Johnson, Nikhila Ravi, Jeremy Reizenstein, David Novotny, Shubham Tulsiani, Christoph Lassner, and Steve Branson. Accelerating 3d deep learning with pytorch3d. In *SIGGRAPH Asia 2020 Courses*, SA '20, New York, NY, USA, 2020. Association for Computing Machinery. 6

[18] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 3, 4, 6

[19] Zhanghan Ke, Jiayu Sun, Kaican Li, Qiong Yan, and Rynson W.H. Lau. Modnet: Real-time trimap-free portrait matting via objective decomposition. In *AAAI*, 2022. 7

[20] Taras Khakhulin, V. V. Sklyarova, Victor S. Lempitsky, and Egor Zakharov. Realistic one-shot mesh-based head avatars. In *European Conference on Computer Vision*, 2022. 1, 2

[21] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013. 4

[22] Zhiyi Kuang, Yiyang Chen, Hongbo Fu, Kun Zhou, and Youyi Zheng. Deepmvshair: Deep hair modeling from sparse views. *SIGGRAPH Asia 2022 Conference Papers*, 2022. 2, 3, 7

[23] Thomas Lewiner, Hélio Lopes, Antônio Wilson Vieira, and Geovan Tavares. Efficient implementation of marching cubes' cases with topological guarantees. *Journal of Graphics Tools*, 8:1 – 15, 2003. 6

[24] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017. 2, 3, 5

[25] Kunliang Liu, Ouk Choi, Jianming Wang, and Wonjun Hwang. Cdgnet: Class distribution guided network for human parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4473–4482, June 2022. 7

[26] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7707–7716, 2019. 2, 6

[27] Stephen Lombardi, Tomas Simon, Jason M. Saragih, Gabriel Schwartz, Andreas M. Lehrmann, and Yaser Sheikh. Neural volumes. *ACM Transactions on Graphics (TOG)*, 38:1 – 14, 2019. 1

[28] Stephen Lombardi, Tomas Simon, Gabriel Schwartz, Michael Zollhoefer, Yaser Sheikh, and Jason M. Saragih. Mixture of volumetric primitives for efficient neural rendering. *ACM Transactions on Graphics (TOG)*, 40:1 – 13, 2021. 1

[29] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Smpl: a skinned multi-

person linear model. *ACM Trans. Graph.*, 34:248:1–248:16, 2015. 2

[30] Morgan McGuire, Tiancheng Sun, Giljoo Nam, Carlos Aliaga, Christophe Hery, and Ravi Ramamoorthi. Human hair inverse rendering using multi-view photometric data. In *Eurographics Symposium on Rendering*, 2021. 1

[31] Marko Mihajlović, Aayush Bansal, Michael Zollhoefer, Siyu Tang, and Shunsuke Saito. Keypointnerf: Generalizing image-based volumetric avatars using relative spatial encoding of keypoints. In *European Conference on Computer Vision*, 2022. 1

[32] 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 *European Conference on Computer Vision*, 2020. 2, 6

[33] Giljoo Nam, Chenglei Wu, Min H. Kim, and Yaser Sheikh. Strand-accurate multi-view hair capture. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 155–164, 2019. 1, 2, 3, 8

[34] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *International Conference on Computer Vision (ICCV)*, 2021. 7

[35] Ahmed A A Osman, Timo Bolkart, and Michael J. Black. STAR: A sparse trained articulated human body regressor. In *European Conference on Computer Vision (ECCV)*, pages 598–613, 2020. 2

[36] Ahmed A A Osman, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. SUPR: A sparse unified part-based human body model. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[37] Sylvain Paris, Héctor M. Briceño, and François X. Sillion. Capture of hair geometry from multiple images. *ACM SIGGRAPH 2004 Papers*, 2004. 2, 5

[38] Jeong Joon Park, Peter R. Florence, Julian Straub, Richard A. Newcombe, and S. Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 165–174, 2019. 2, 4

[39] Keunhong Park, U. Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B. Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5845–5854, 2020. 1

[40] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Trans. Graph.*, 40, 2021. 1

[41] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10967–10977, 2019. 2

[42] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022. 3, 6

[43] Eduard Ramon, Gil Triginer, Janna Escur, Albert Pumarola, Jaime Garcia Giraldez, Xavier Giró i Nieto, and Francesc Moreno-Noguer. H3d-net: Few-shot high-fidelity 3d head reconstruction. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5600–5609, 2021. 1, 2, 6, 7

[44] Radu Alexandru Rosu, Shunsuke Saito, Ziyan Wang, Chenglei Wu, Sven Behnke, and Giljoo Nam. Neural strands: Learning hair geometry and appearance from multi-view images. *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 4, 5, 6, 8, 9

[45] Shunsuke Saito, Liwen Hu, Chongyang Ma, Hikaru Ibayashi, Linjie Luo, and Hao Li. 3d hair synthesis using volumetric variational autoencoders. *ACM Transactions on Graphics (TOG)*, 37:1 – 12, 2018. 2, 3

[46] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 5

[47] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Deep image prior. *Int. J. Comput. Vis.*, 128(7):1867–1888, 2020. 5

[48] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 1, 2, 3, 4, 7

[49] Xueying Wang, Yudong Guo, Zhongqi Yang, and Juyong Zhang. Prior-guided multi-view 3d head reconstruction. *IEEE Transactions on Multimedia*, 24:4028–4040, 2021. 1, 2

[50] Ziyan Wang, Giljoo Nam, Tuur Stuyck, Stephen Lombardi, Chen Cao, Jason M. Saragih, Michael Zollhoefer, Jessica K. Hodgins, and Christoph Lassner. Neuwigs: A neural dynamic model for volumetric hair capture and animation. *ArXiv*, abs/2212.00613, 2022. 1

[51] Ziyan Wang, Giljoo Nam, Tuur Stuyck, Stephen Lombardi, Michael Zollhoefer, Jessica K. Hodgins, and Christoph Lassner. Hvh: Learning a hybrid neural volumetric representation for dynamic hair performance capture. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6133–6144, 2021. 1

[52] Mason Woo, Jackie Neider, Tom Davis, and Dave Shreiner. *OpenGL programming guide: the official guide to learning OpenGL, version 1.2*. Addison-Wesley Longman Publishing Co., Inc., 1999. 6

[53] Bernhard P. Wrobel. Multiple view geometry in computer vision. *Künstliche Intell.*, 15:41, 2001. 5

[54] Keyu Wu, Yifan Ye, Lingchen Yang, Hongbo Fu, Kun Zhou, and Youyi Zheng. Neuralhdhair: Automatic high-fidelity hair modeling from a single image using implicit neural representations. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1516–1525, 2022. 2, 3, 7, 8

[55] Lingchen Yang, Zefeng Shi, Youyi Zheng, and Kun Zhou. Dynamic hair modeling from monocular videos using deep neural networks. *ACM Transactions on Graphics (TOG)*, 38:1 – 12, 2019. 2, 3

[56] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Neural Information Processing Systems*, 2021. 1, 2

[57] Cem Yuksel, Scott Schaefer, and John Keyser. Hair meshes. *ACM SIGGRAPH Asia 2009 papers*, 2009. 2, 6

[58] Meng Zhang and Youyi Zheng. Hair-gan: Recovering 3d hair structure from a single image using generative adversarial networks. *Vis. Informatics*, 3:102–112, 2019. 2, 3

[59] Yufeng Zheng, Victoria Fernández Abrevaya, Xu Chen, Marcel C. Buhler, Michael J. Black, and Otmar Hilliges. I m avatar: Implicit morphable head avatars from videos. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13535–13545, 2021. 1, 2

[60] Yufeng Zheng, Yifan Wang, Gordon Wetzstein, Michael J. Black, and Otmar Hilliges. Pointavatar: Deformable point-based head avatars from videos. *ArXiv*, abs/2212.08377, 2022. 1, 2

[61] Yi Zhou, Liwen Hu, Jun Xing, Weikai Chen, Han-Wei Kung, Xin Tong, and Hao Li. Hairnet: Single-view hair reconstruction using convolutional neural networks. In *European Conference on Computer Vision*, 2018. 2, 3, 4