Learning Neural Orientation Field for Volumetric Hair Reconstruction

Computer Vision Project Proposal

Fangjun Zhou	Weiran Xu	Zhenyu Zhang
fzhou48	weiran	zhenyuz5

Saturday 9th November, 2024

1 Introduction

Reconstructing human hair is one of the most challenging yet critical process in rendering photorealistic digital human. Unlike other parts of the human body, human hair is highly detailed and often intertwined together. Therefore, it's difficult to use traditional photogrammetry method to reconstruct its structure.

Before machine learning is used in this field, artists often hand crafted splines on skulls to represent hair strands. Each strand is then textured and rendered to mimic the hair volume. This workflow requires a lot of experience as it's non-trivial for artists to infer the final render result from hair stand splines. To reduce the workload and improve the accuracy of hair reconstruction, machine learning models are used to generate hair strand from captured photos.

2 Related Work

Previous attempt to achieve this goal mainly focus on learning based hair strand generation. This includes some studies about single view hair synthesis [**saito˙3d˙2018**, **zheng˙hairstep˙2023**, **wu˙neuralhdhair˙2022**, **ma˙single-view˙nodate**]. Since the image only contains hair structure from one viewing angle, it's impossible to reconstruct entire hair accurately. These models often use pretrained image encoders such as ResNet-50 [**saito˙3d˙2018**] to encode the abstract hair style into a feature vector, then use generative models such as U-Net [**zheng˙hairstep˙2023**], VAE [**saito˙3d˙2018**], and diffusion [**sklyarova˙neural˙2023**]. These models also struggle with generating curly hair as there's only limited information about growing direction after feature extraction.

In [**sklyarova˙neural˙2023**] and [**rosu˙neural˙2022**], the authors also tried hair syntheses from multi-view images. However, these two studies still failed to capture finer detail.

Another study about this topic tried to tackle this problem by expanding the traditional PatchMatch MVS (PMVS) algorithm to a Line-based PatchMatch MVS (LPMVS) [**nam˙strand-accurate˙nodate**]. This method, despite its high accuracy, doesn't capture the volumetric property of human hair.

NeRF [**mildenhall˙nerf˙2020**] is a method used for 3D reconstruction from 2D images, utilizing neural networks to predict the color and density of points in 3D space. Our approach references the design of NeRF at various stages, transforming the color prediction task into a direction vector prediction task to achieve the goal of reconstructing hair flow directions.

In this study, we aim to develop a new method for hair reconstruction. The major goal of this model is to provide detailed growth vector for direct volumetric rendering or as a guidance for hair strand generation.

3 Method

3.1 Data Pre-process

Our model will be trained on images of human bust from multiple viewing angles. The preprocessing kernel will extract the projection of the hair growing direction onto the camera viewing plane. This can be done by a simple edge detection kernel such as Gabor filter used in related studies.

3.2 Backbone

The backbone network is similar to NeRF [**mildenhall˙nerf˙2020**]'s backbone, primarily consisting of multiple MLP layers and residual connections. The difference is that the input to the network we constructed is the spatial coordinates (x, y, z) of the point to be evaluated, and the output is the direction vector of the hair at that point $v = (v_x, v_y, v_z)$, as well as the opacity parameter σ. The direction vector is used to describe the growing direction of the hair at this point, while the opacity parameter sigma is used in loss construction to model the occlusion relationships of the hair which will be discussed later.

3.3 Loss Defination

The construction of the model's loss is divided into two parts: occlusion relationship modeling and plane projection of the direction vector. In occlusion relationship modeling, the weighted 3D direction vector is obtained by integrating the direction vectors along the observation direction ray, weighted by the opacity parameter σ and an function describe the accumulative transmittance, similar to NeRF [**mildenhall˙nerf˙2020**]. We can simulate the integration process through sampling specific points on the observation direction ray and discrete the original integral.

For the second part, we project the weighted 3D direction vector obtained from the first part along the viewing direction onto the camera plane, resulting in the 2D direction vector

observed by the camera. The final loss function is constructed by calculating the MSE loss between the 2D direction vector and the ground truth. We will use mini-batched gradient descent to optimize the loss function.

4   Experiment

Given a set of images of human busts from multiple viewing angles, we will first use COLMAP, a general-purpose MVS pipeline that processes a set of images to generate a point cloud, as our baseline. Next, we will evaluate the quality enhancements provided by our model compared to both the Line-based PatchMatch MVS [**nam˙strand-accurate˙nodate**] and generative models. Since the final output of our model differs from that of [**nam˙strand-accurate˙n** and the generative models, we will not compare their performances directly. Instead, we will evaluate performance relative to COLMAP. Finally, we will demonstrate the tunability of our model's output using Blender.