Refinement of Hair Geometry by Strand Integration

Ryota Maeda (University of Hyogo),
Kenshi Takayama, Takafumi Taketomi (CyberAgent)

CyberAgent AI Lab

CyberAgent AI Lab

Digital Human

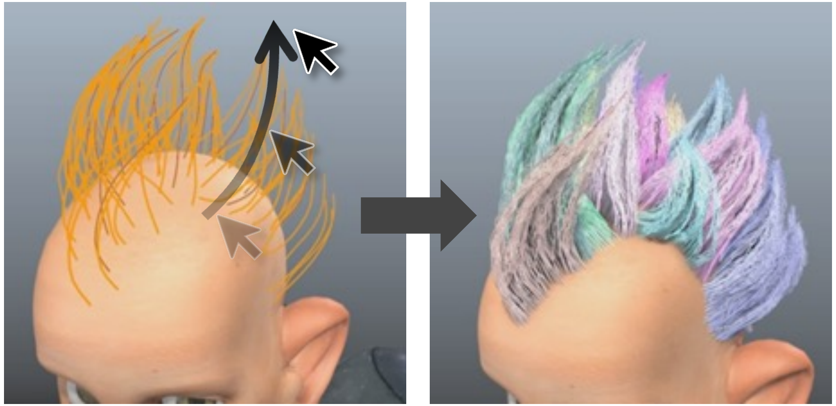
- Create 3D models of real people by capturing images

Multi-view camera

Video production

Difficulty of Hair Modeling

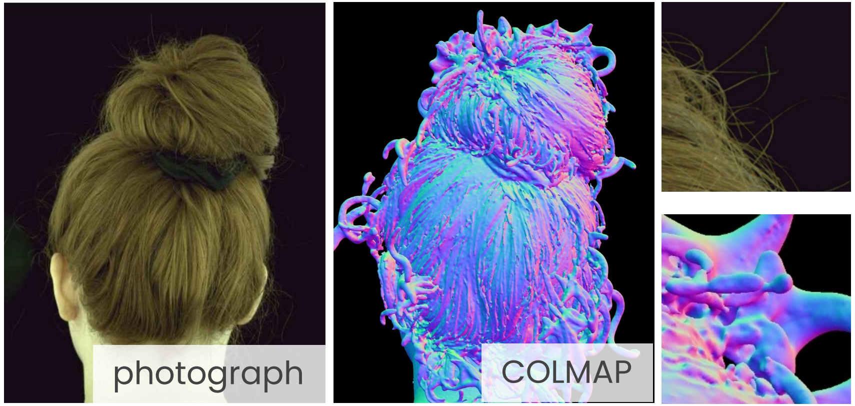
Hand modeling by CG artists



[Maya XGen Official Doc.]

- Technical expertise and artistic skill
- Laborious and time-consuming task

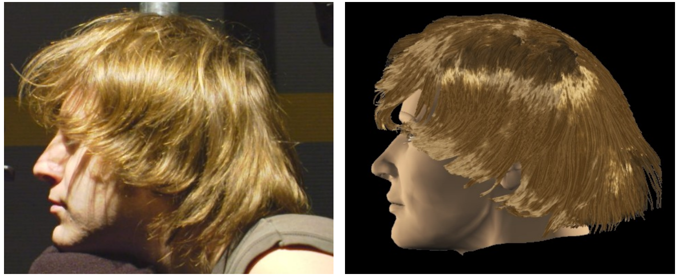
Using multi-view images



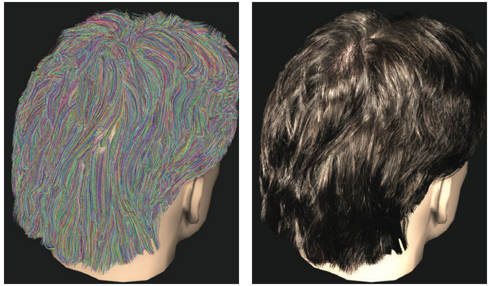
[Nam et al.]

- Conventional MVS does not work

Related Works : Hair Reconstruction from Multi-view Images

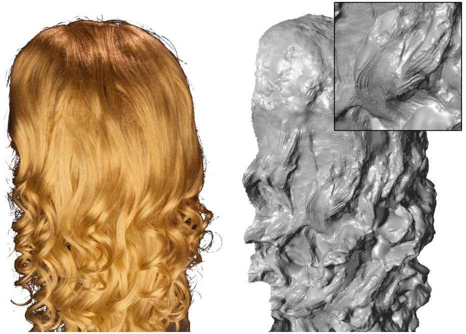


[Paris et al, SIGGRAPH2004]

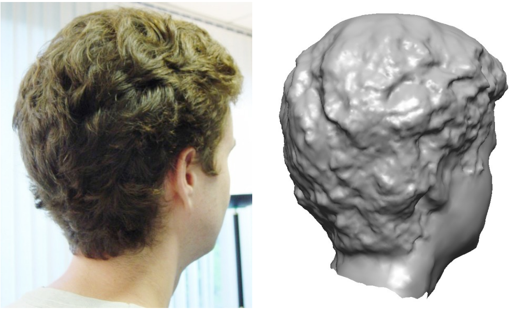


[Paris et al, SIGGRAPH2008]

[Hu et al, SIGGRAPH2014]



[Luo et al, CVPR2012]



[Luo et al, CVPR2013]

[Luo et al, SIGGRAPH2013]

LPMVS (Line-based PatchMatch Multi-View Stereo) [Nam et al., CVPR2019]

Strand-accurate Multi-view Hair Capture

Giljoo Nam*[1] Chenglei Wu[2] Min H. Kim[1] Yaser Sheikh[2]
[1]KAIST [2]Facebook Reality Labs, Pittsburgh

Abstract

Hair is one of the most challenging objects to reconstruct due to its micro-scale structure and a large number of repeated strands with heavy occlusions. In this paper, we present the first method to capture high-fidelity hair geometry with strand-level accuracy. Our method takes three stages to achieve this. In the first stage, a new multi-view stereo method with a slanted support line is proposed to solve the hair correspondences between different views. In detail, we contribute a novel cost function consisting of both photo-consistency term and geometric term that reconstructs each hair pixel as a 3D line. By merging all the depth maps, a point cloud, as well as local line directions for each point, is obtained. Thus, in the second stage, we feature a novel strand reconstruction method with the mean-shift to convert the noisy point data to a set of strands. Lastly, we grow the hair strands with multi-view geometric constraints to elongate the short strands and recover the missing strands, thus significantly increasing the reconstruction completeness. We evaluate our method on both

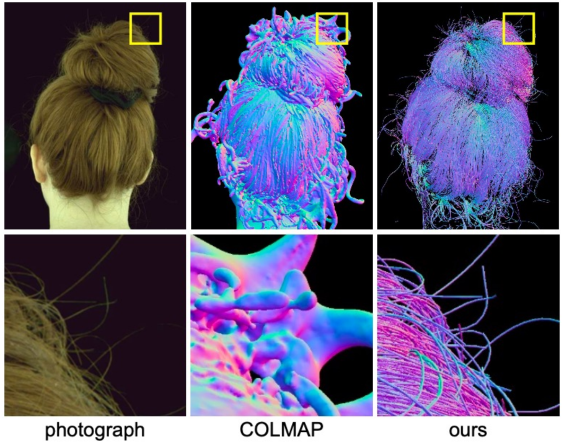
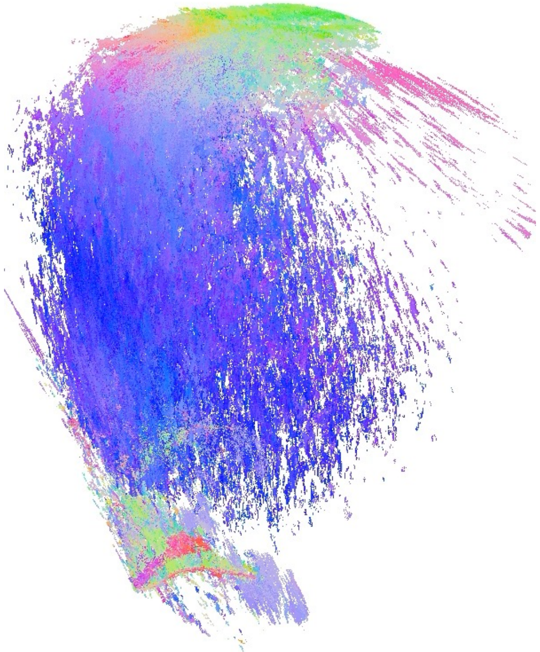


Figure 1. (Left) One of the photographs from multi-view capture. (Middle) Final geometry by traditional MVS (COLMAP [33]). (Right) Final geometry by our method. Our method can produce high-fidelity hair geometry with strand-level accuracy.

LPMVS (Line-based PatchMatch Multi-View Stereo) **[Nam et al., CVPR2019]**

Input: Multi-view images

LPMVS
[Nam+, CVPR2019]

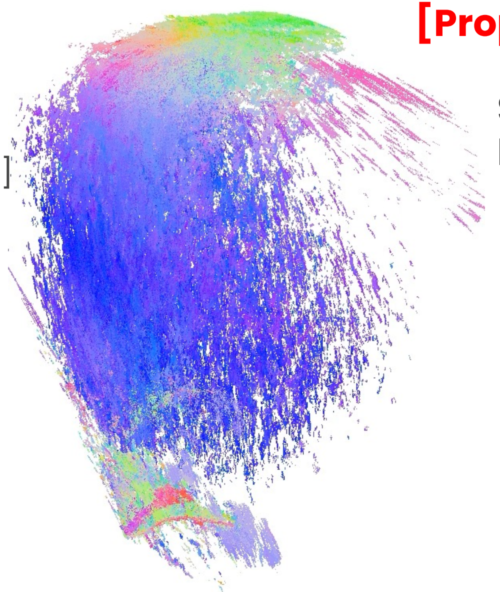


Output: 3D lines (*unfilterd*)

Our Contribution: Strand Integration

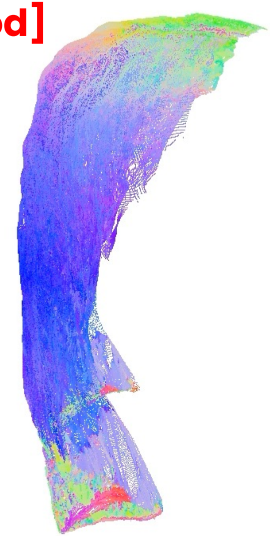
Input: Multi-view images

LPMVS
[Nam+, CVPR2019]



[Proposed method]

Strand Integration

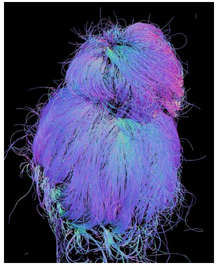


General Pipeline for Reconstruction of Hair Geometry

Reconstruction of Hair Geometry from Multi-view Images

Input → **Reconstruction Algorithm** → **Output**

Multi-view images



Hair geometry

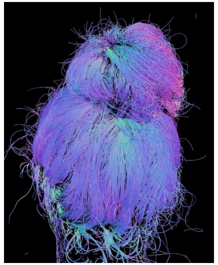
Downstream tasks
(i.e. neural rendering)
[Rosu et al., ECCV2022]
[Wang et al., CVPR2022]

Reconstruction of Hair Geometry from Multi-view Images

Input → **Esitimate 3D lines** → **Merge into long strands** → Output

Our target

Multi-view images



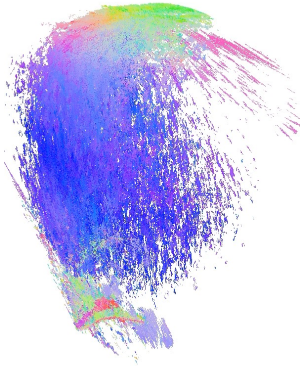
Hair geometry

Reconstruction of Hair Geometry from Multi-view Images

Esitimate 3D lines

Input → **Estimate depth and direction map for every view** → **Merge all views into a set of 3D lines**

Multi-view images

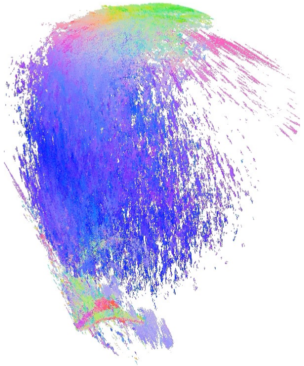


Reconstruction of Hair Geometry from Multi-view Images

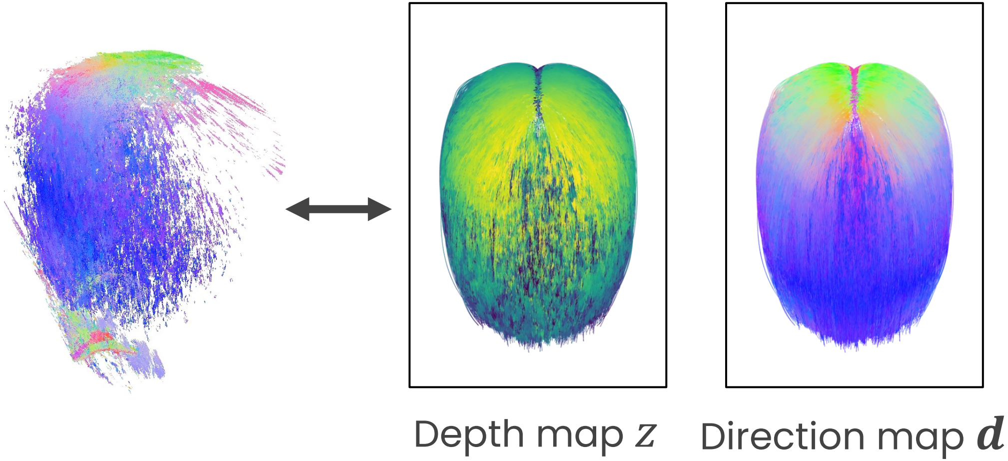
Esitimate 3D lines

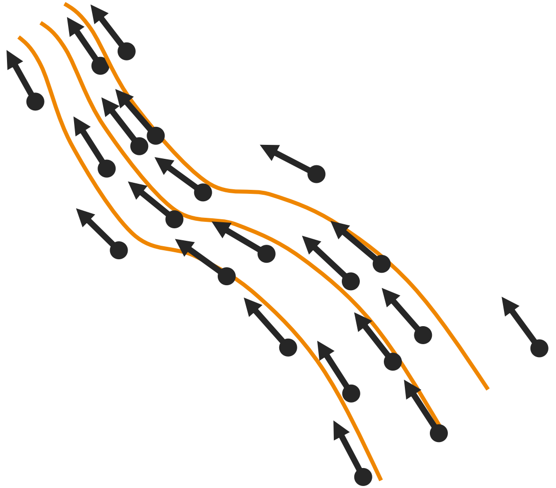
Input → Estimate depth and direction map for every view → **[Proposed method] Strand Integration** → Merge all views into a set of 3D lines

Multi-view images



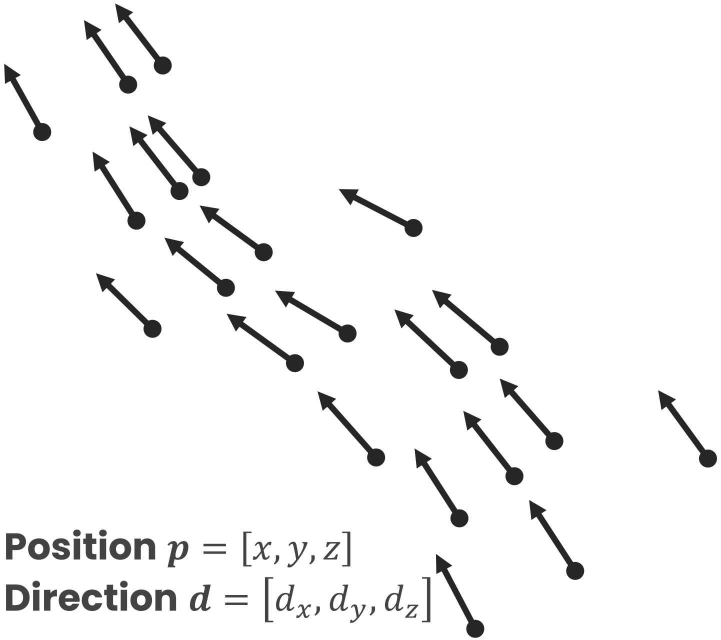
Representation of Hair Geometry by 3D Lines





- **Position** $\boldsymbol{p} = [x, y, z]$
- **Direction** $\boldsymbol{d} = [d_x, d_y, d_z]$

Problem of the Existing Method (LPMVS [Nam et al.])

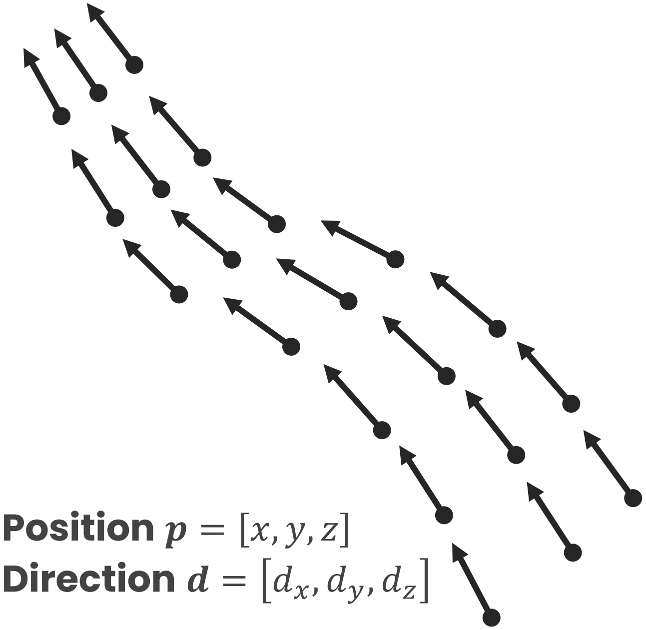


- **Position** $p = [x, y, z]$
- **Direction** $d = [d_x, d_y, d_z]$

Hair strands are smoothly connected…

Problem
Does NOT consider
the spatial coherence of 3D lines.

Our Method: Strand Integration



- **Position** $p = [x, y, z]$
- **Direction** $d = [d_x, d_y, d_z]$

Hair strands are smoothly connected…

Problem
Does NOT consider
the spatial coherence of 3D lines.

Our Method: Strand Integration
Refine the position of hair strands by using the direction.

Strand Integration

Loss Function

Find the depth map z which minimizes

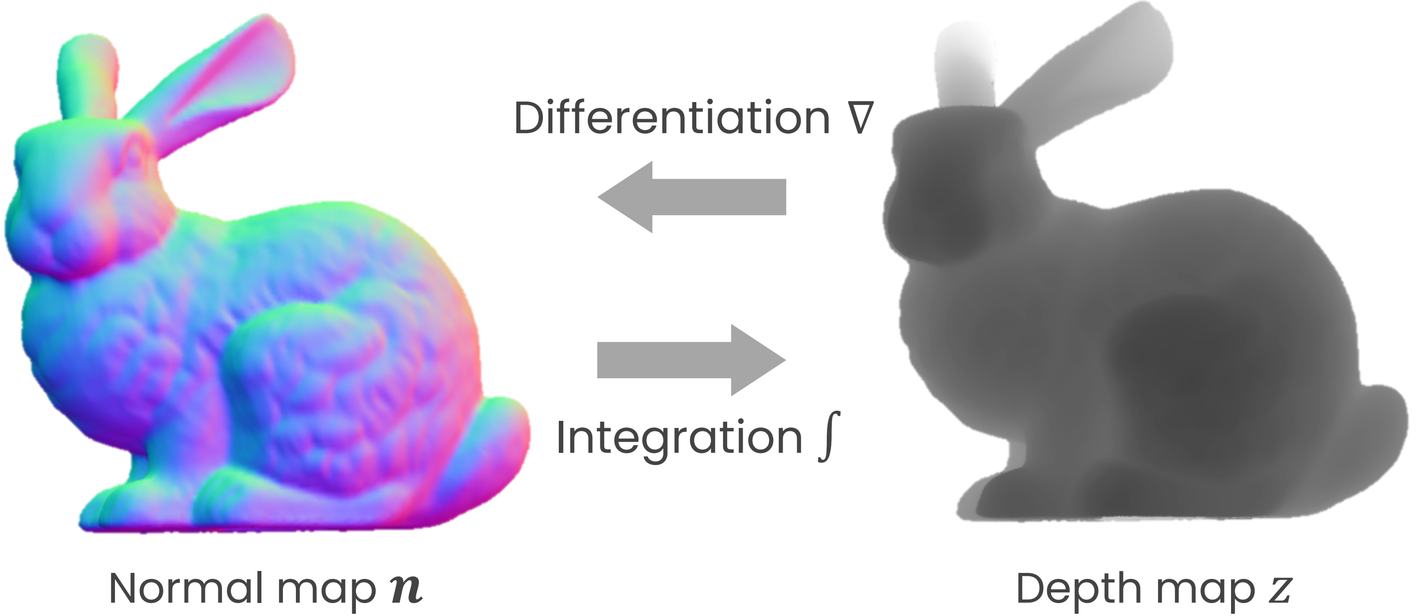
$$\mathcal{L}(z) = \lambda_d \, \mathcal{L}_d(z) + \mathcal{L}_z(z)$$

Direction loss   Depth loss

Direction Loss: $\mathcal{L}_d$

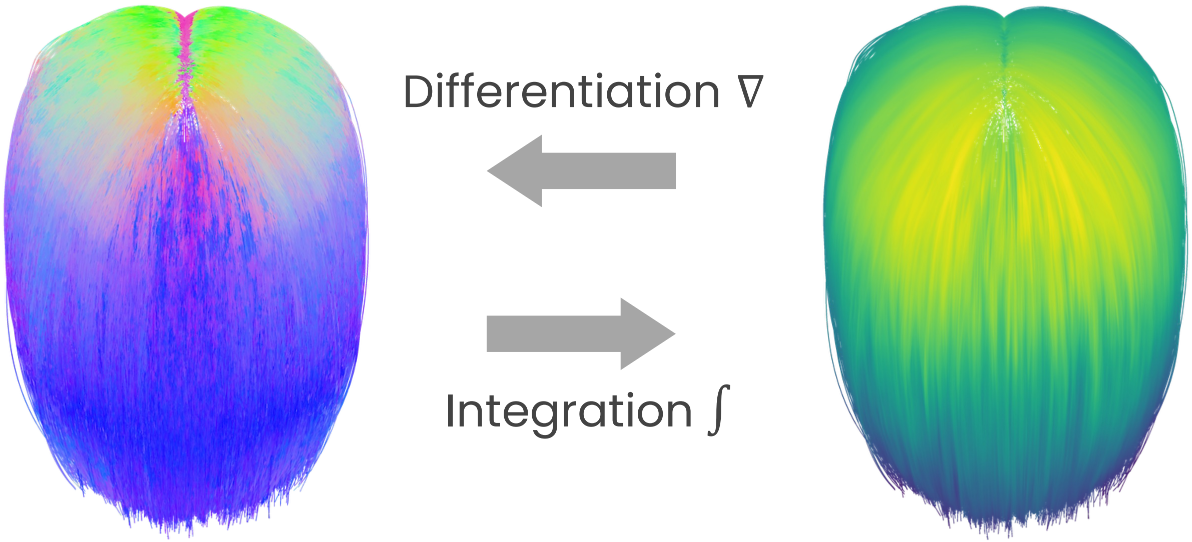
- Harness the **position** and **direction** information for improving geometrical coherence.
- Inspired by **Normal Integration**.

Normal Integration: Depth from Normal



[Image from ICIP2019 Tutorial: Photometric 3D Reconstruction]

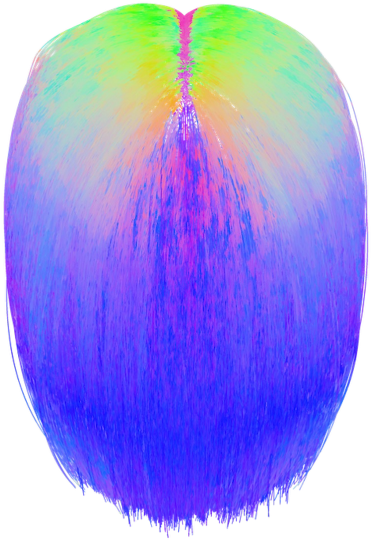
Strand Integration: Depth from Direction



Direction map $\boldsymbol{d}$

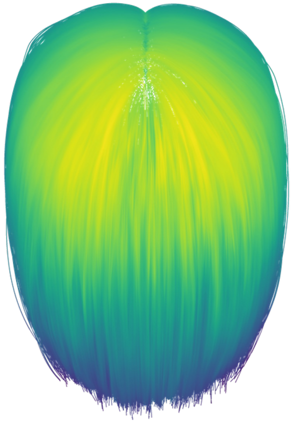
Depth map z

Strand Integration: Depth from Direction



$$= \nabla_{\theta}$$

Orientation
of the strand



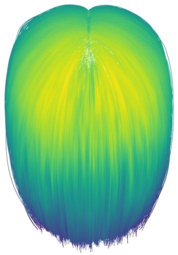
Direction map $\boldsymbol{d}$

Differentiation of depth map $\nabla_{\theta} z$

Direction Loss: $\mathcal{L}_d$

$$\mathcal{L}_d(z) = \sum \left(d_{\text{prior}} - \nabla_\theta(z)\right)^2$$

∇_θ



Loss Function

Find the depth map z which minimizes

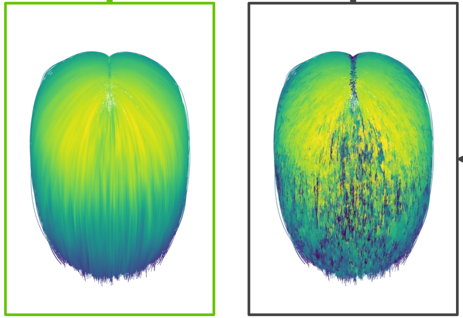
$$\mathcal{L}(z) = \lambda_d \, \mathcal{L}_d(z) + \mathcal{L}_z(z)$$

Direction loss Depth loss

Depth Loss : $\mathcal{L}_z$

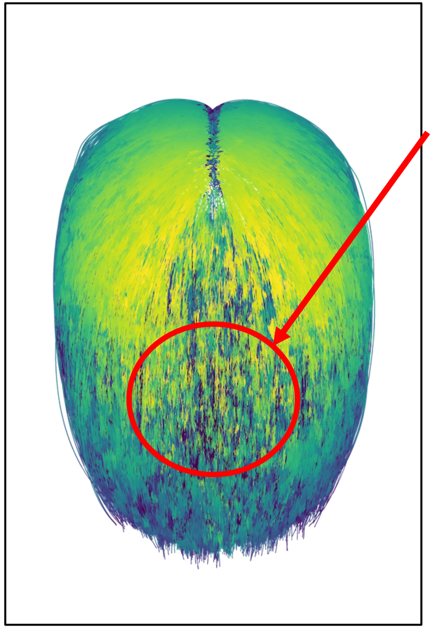
- Use prior depth map as anchor points.
- Incorporate the multi-view constraint into the refinement process.

$$\mathcal{L}_z(z) = \sum \left(z - z_{\mathrm{prior}}\right)^2$$



Contain **inaccurate** depth values

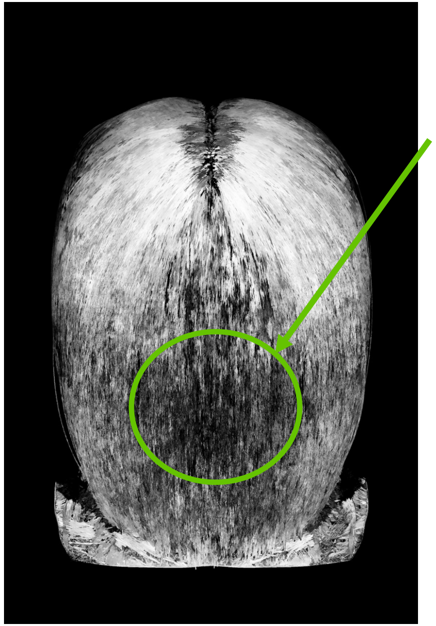
3D Line Consistency Map



Prior depth map

Inaccurate depth

distort
the refinement result



Consistency Map
with respect to
neighboring views

Low consistency

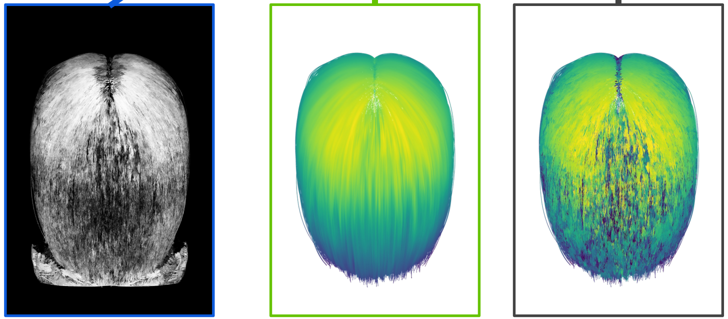
use as per-pixel weight

to ignore
the inaccurate depth

Depth Loss (w/ Consistency Map) : $\mathcal{L}_z$

- Use prior depth map as anchor points.
- Incorporate the multi-view constraint into the refinement process.

$$\mathcal{L}_z(z) = \sum c\left(z - z_{\mathrm{prior}}\right)^2$$



Loss Function

Find the depth map z which minimizes

$$\mathcal{L}(z) = \lambda_d \underbrace{\mathcal{L}_d(z)}_{\text{Direction loss}} + \underbrace{\mathcal{L}_z(z)}_{\text{Depth loss}}$$

Loss Function with Normal Loss

Find the depth map z which minimizes

$$\mathcal{L}(z) = \lambda_n \mathcal{L}_n(z) + \lambda_d \mathcal{L}_d(z) + \mathcal{L}_z(z)$$

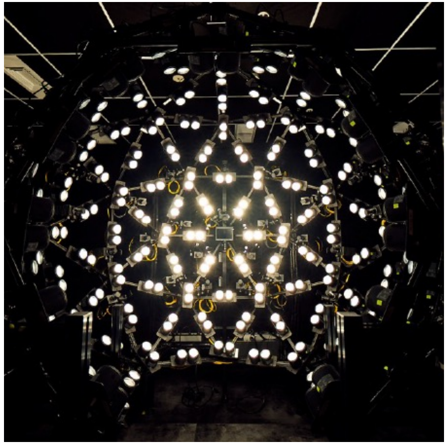
Normal loss (optional)

Direction loss

Depth loss

Normal Loss (optional): $\mathcal{L}_n$

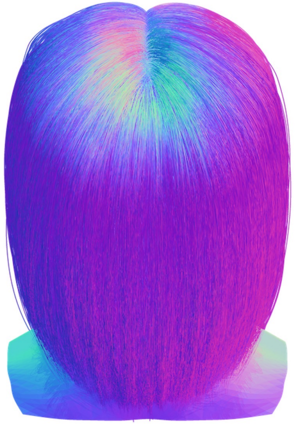
- If we obtain the **normal map** , we can use it as **optional loss.**
- Normal should be perpendicular to the strand direction.



Light stage system

Photometric images

Photometric stereo

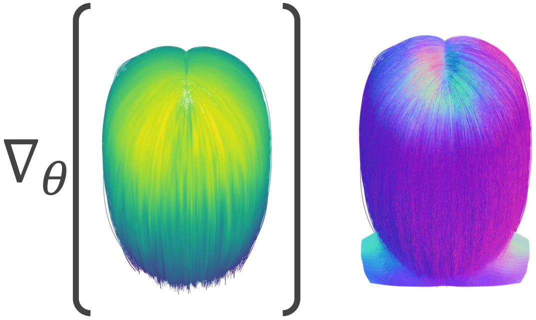


Normal map

Normal Loss (optional): $\mathcal{L}_n$

- If we obtain the **normal map** , we can use it as **optional loss.**
- Normal should be perpendicular to the strand direction.

$$\mathcal{L}_n(z) = \sum \left(\boldsymbol{d} \cdot \boldsymbol{n}_{\mathrm{prior}}\right)^2$$



Loss Function with Normal Loss

Find the depth map z which minimizes

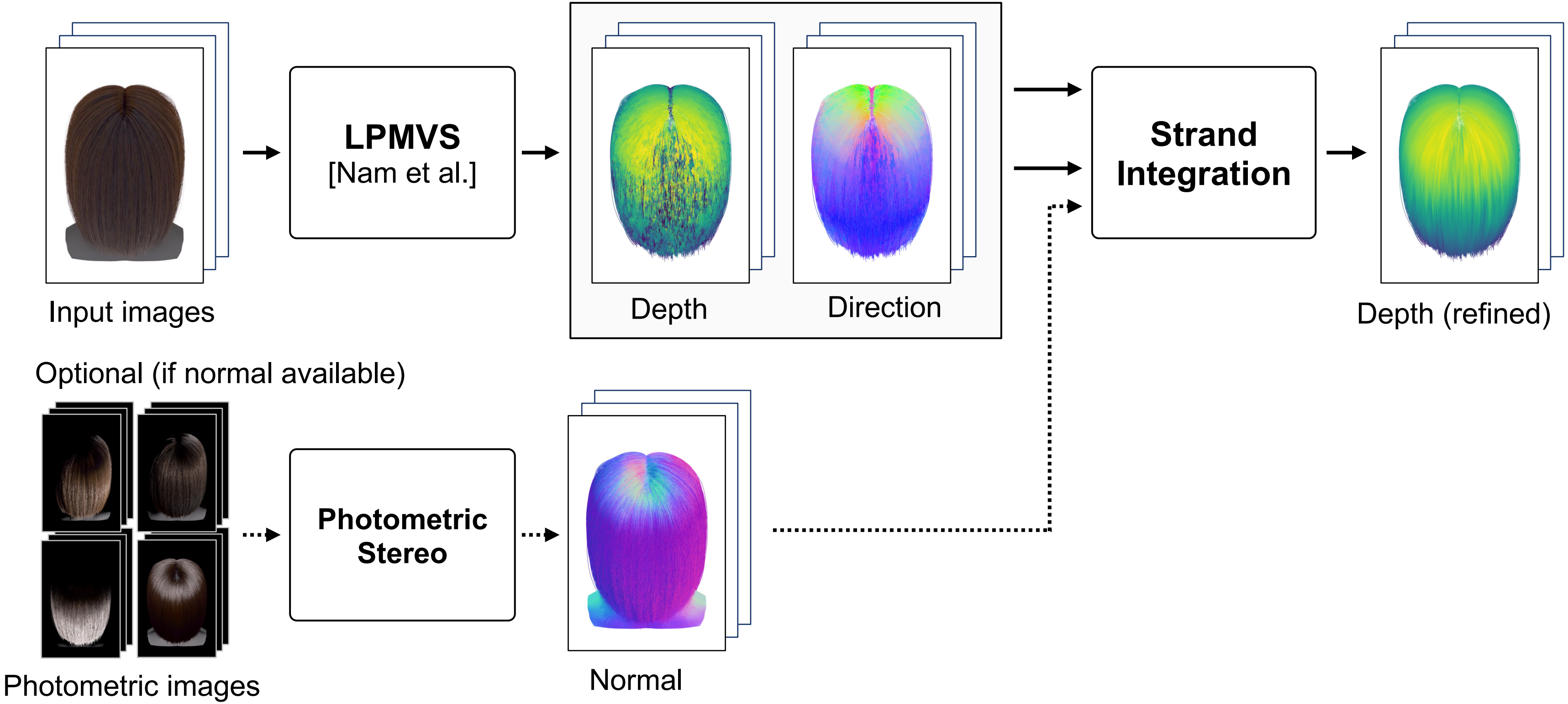
$$\mathcal{L}(z) = \lambda_n \mathcal{L}_n(z) + \lambda_d \mathcal{L}_d(z) + \mathcal{L}_z(z)$$

Normal loss
(optional)

Direction loss

Depth loss

Overall Pipeline of Strand Integration



Experiment

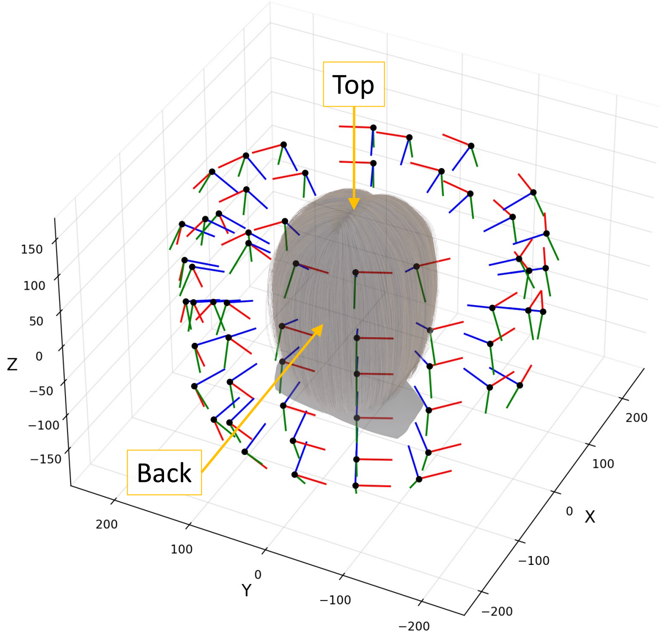
Implementation

- LPMVS [Nam et al.]
  - The official implementation is not publicly available.
  - CPU-based reimplementation in C++.
- Strand Integration
  - Use PyTorch as a general gradient descent solver.
  - 20 min for each view on Apple M1 Max. 11 MP(2730x4096) image.

Synthetic Data

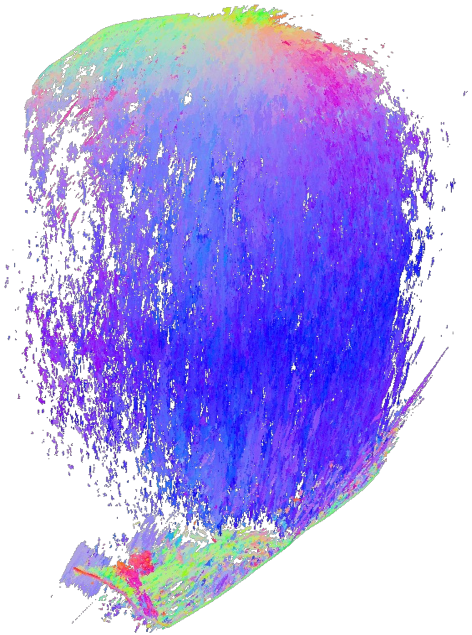
- Rendered with pbrt-v4
- 60 views
- 2730x4096
- 4 hairstyles

Straight Curly Wavy Wavythin



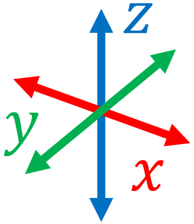
Result: Straight Hair

Ground Truth

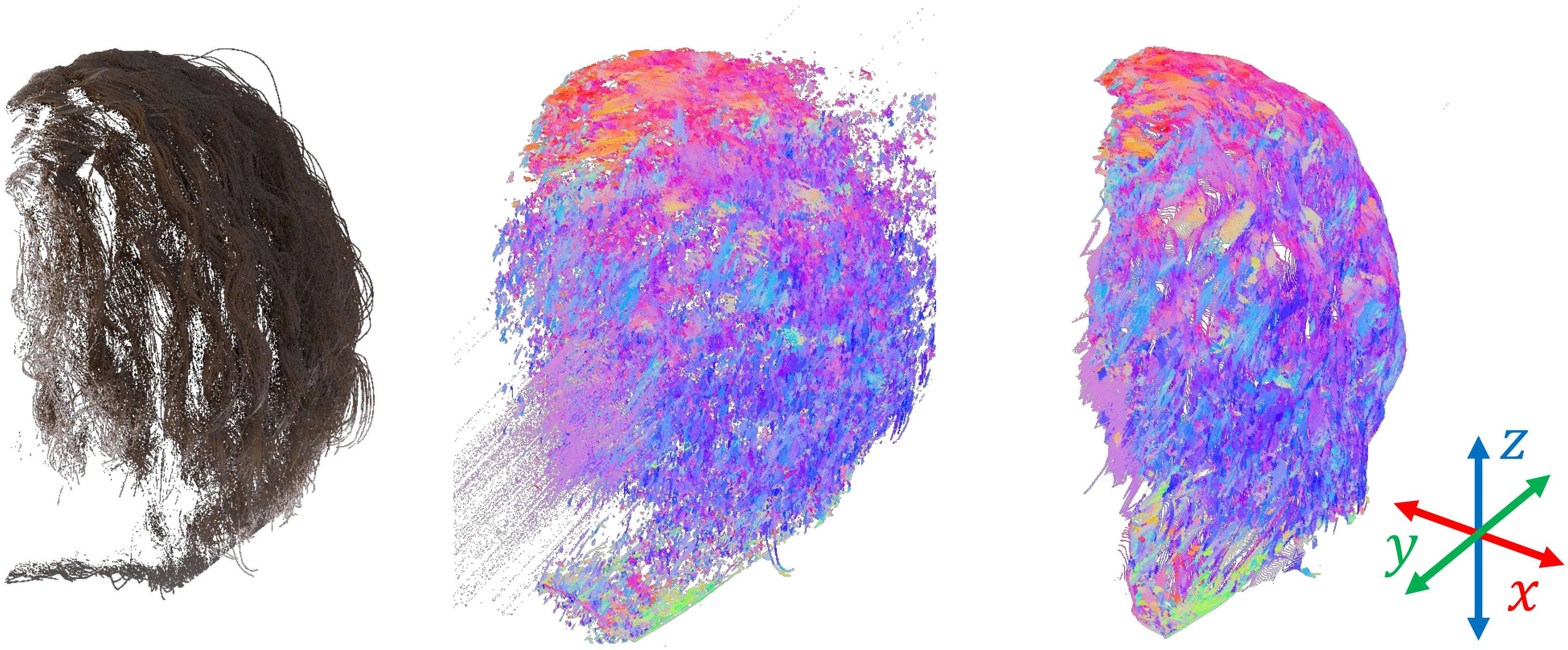


LPMVS
[Nam et al.]

Strand Integration
(ours)



Result: Curly Hair

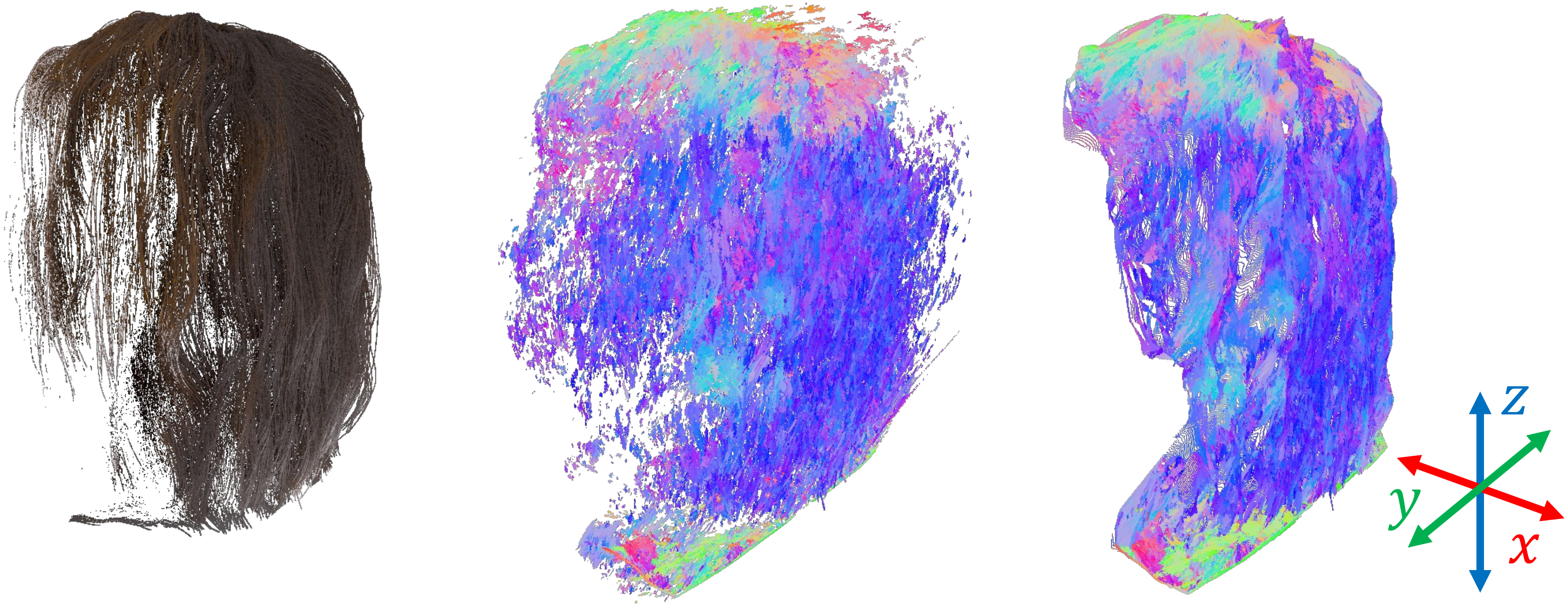


Ground Truth

LPMVS
[Nam et al.]

Strand Integration
(ours)

Result: Wavy Hair



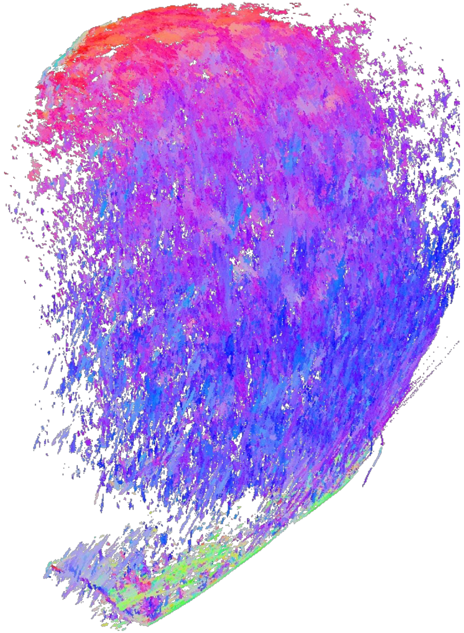
Ground Truth

LPMVS
[Nam et al.]

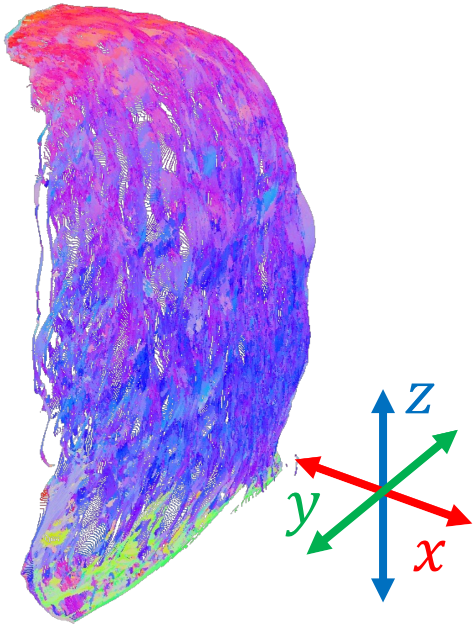
Strand Integration
(ours)

Result: Wavythin Hair

Ground Truth

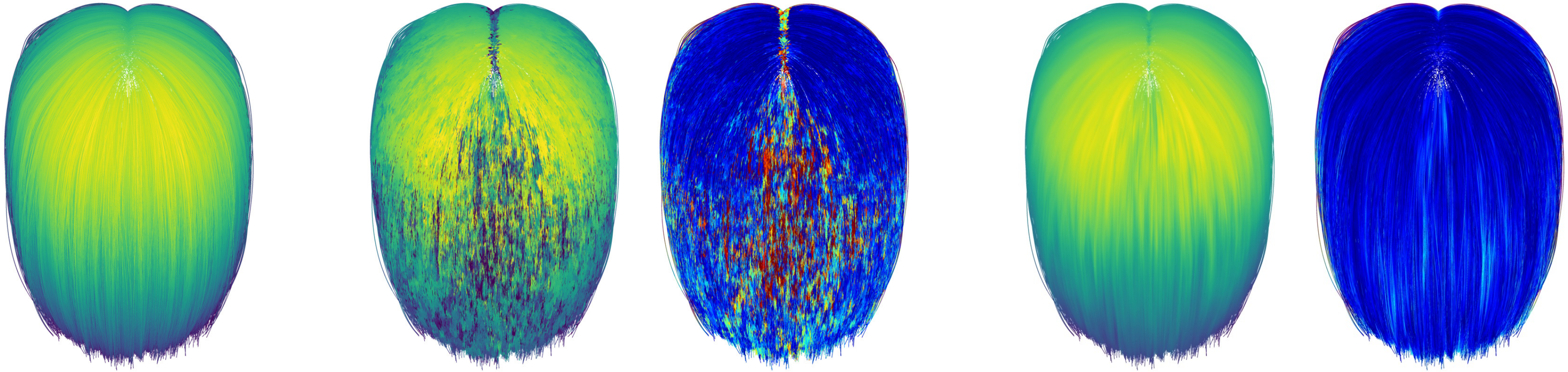


LPMVS
[Nam et al.]



Strand Integration
(ours)

Error Analysis on Depth Map



Abs. error

Abs. error

Ground Truth

LPMVS [Nam et al.]

MAE : 30.05
RMSE: 52.57

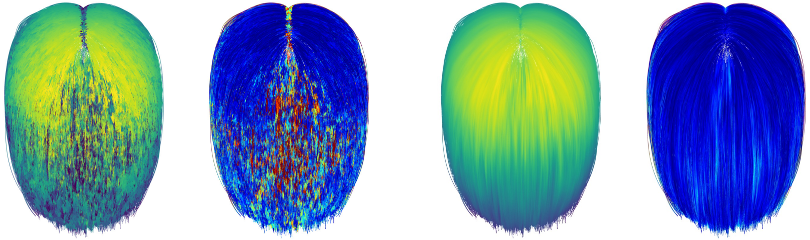
Strand Integration

MAE : **9.67**
RMSE: **14.92**

Error Analysis on Depth Map

MAE / RMSE

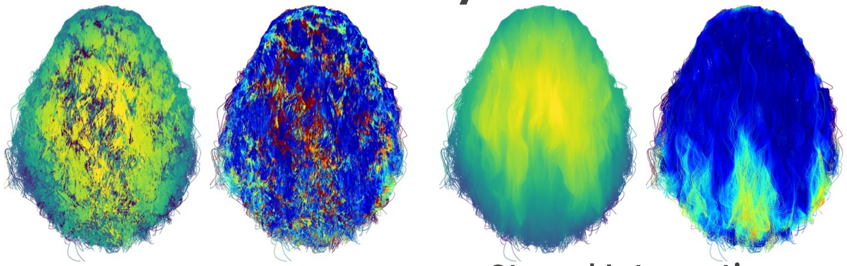
Straight



LPMVS
30.05 / 52.57

Strand Integration
9.76 / 15.03

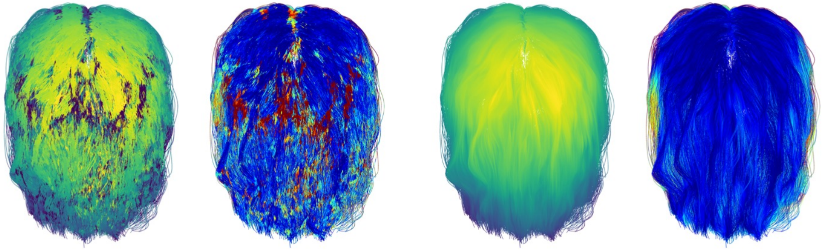
Curly



LPMVS
41.47 / 67.87

Strand Integration
18.10 / 27.07

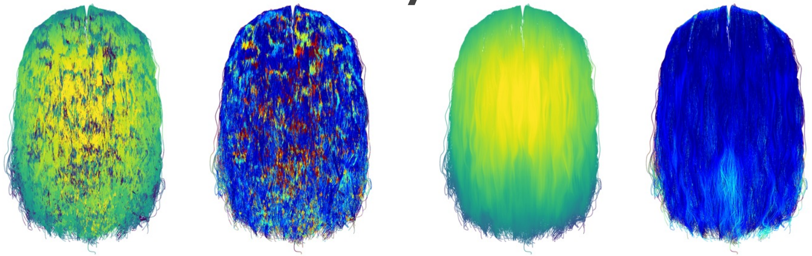
Wavy



LPMVS
34.21 / 60.30

Strand Integration
12.20 / 20.57

Wavythin



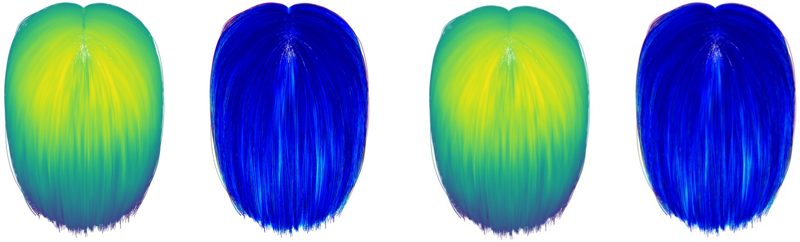
LPMVS
34.96 / 58.54

Strand Integration
11.34 / 19.36

Ablation Study of Normal Loss

MAE / RMSE

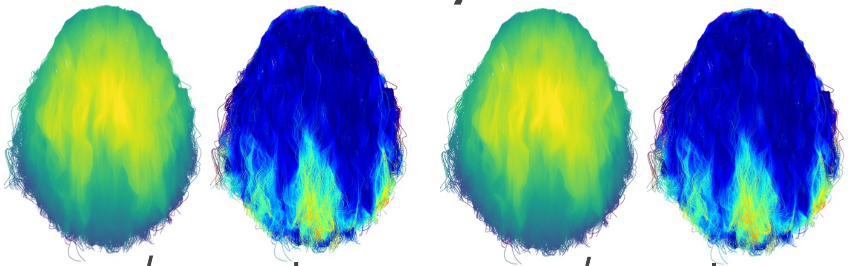
Straight



w/o normal
9.76 / 15.03

w/ normal
9.67 / 14.92

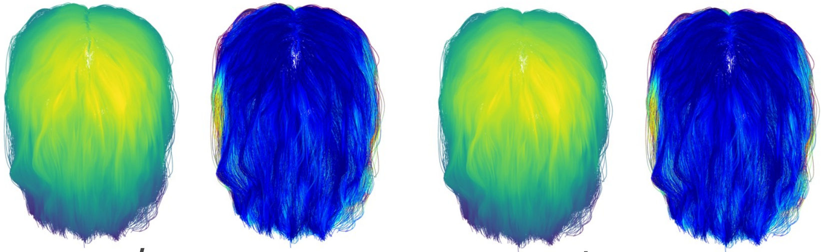
Curly



w/o normal
18.10 / 27.07

w/ normal
17.86 / 26.97

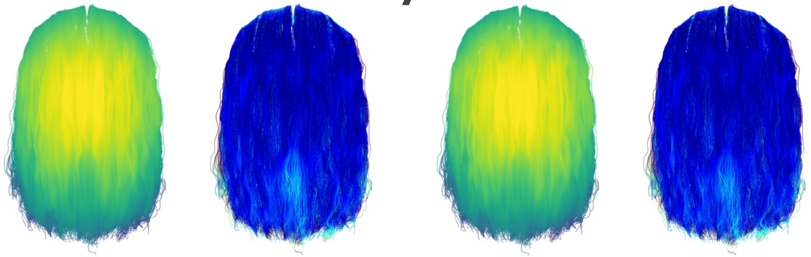
Wavy



w/o normal
12.20 / 20.57

w/ normal
12.25 / **20.53**

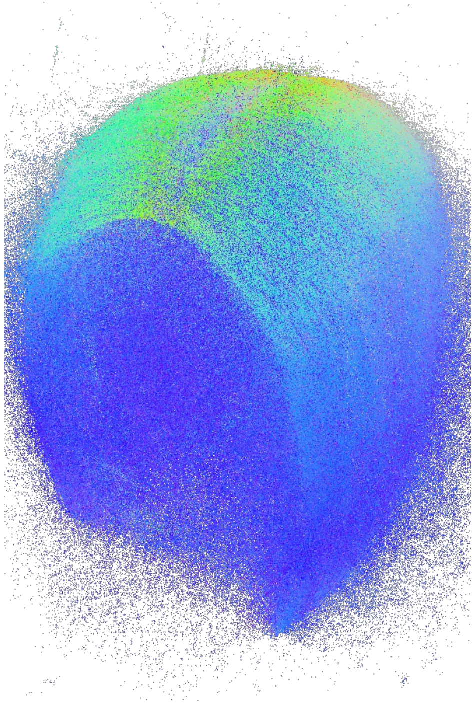
Wavythin



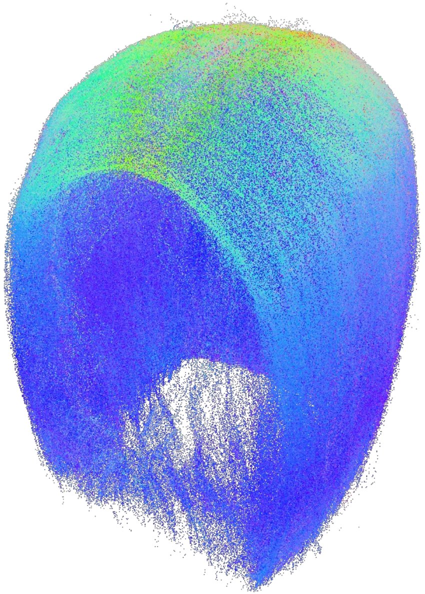
w/o normal
11.34 / 19.36

w/ normal
11.23 / 19.13

Result: Merged Point Cloud



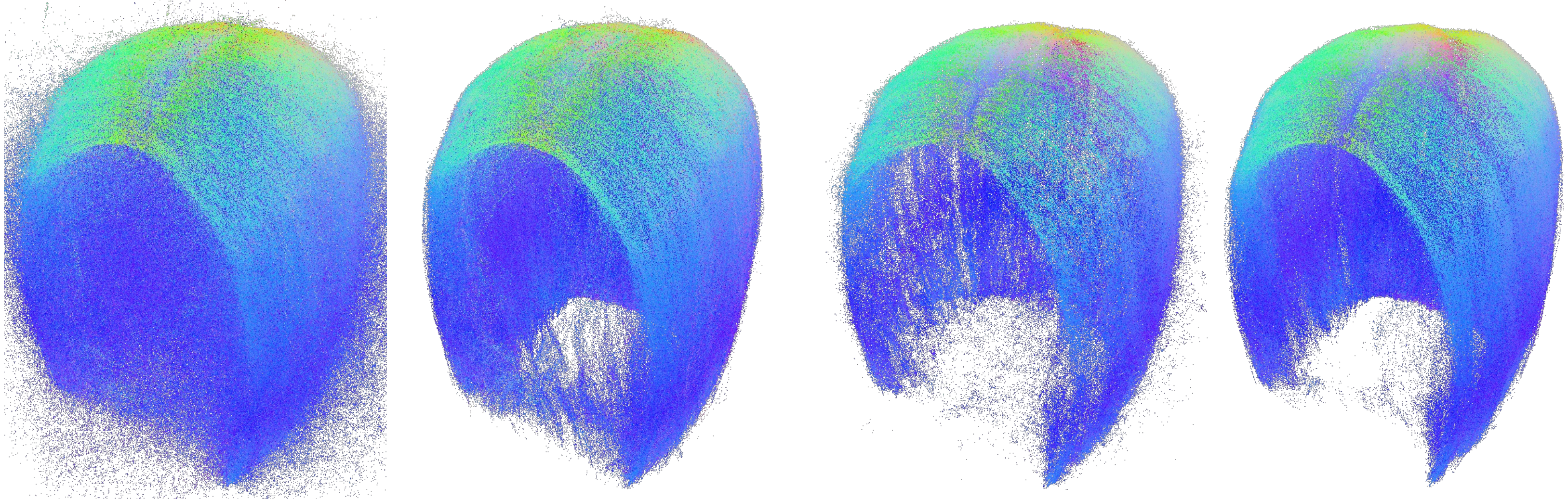
LPMVS [Nam+]



Strand Integration

Result: Merged Point Cloud

All point cloud | **After filtered**



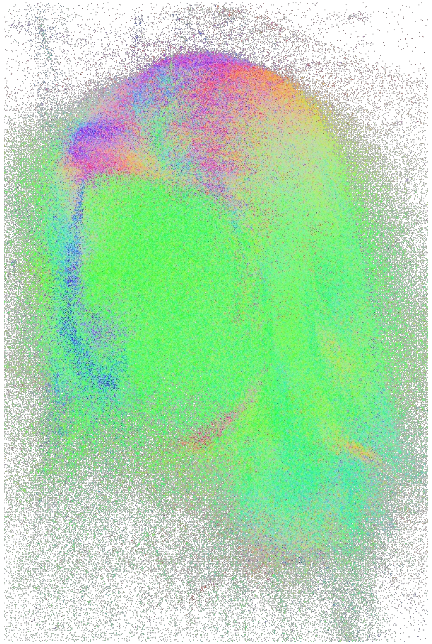
Number of points

297×10^6	297×10^6	85×10^6	175×10^6
LPMVS [Nam+]	Strand Integration	LPMVS [Nam+]	Strand Integration

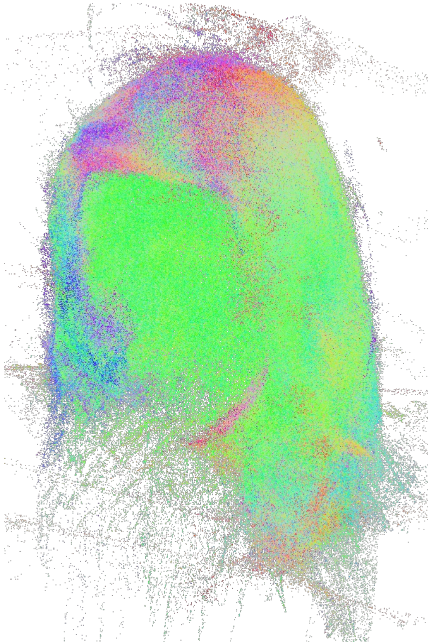
Real Data

- 60 views
- 5315x8001
- Long hair

Result: Real Data



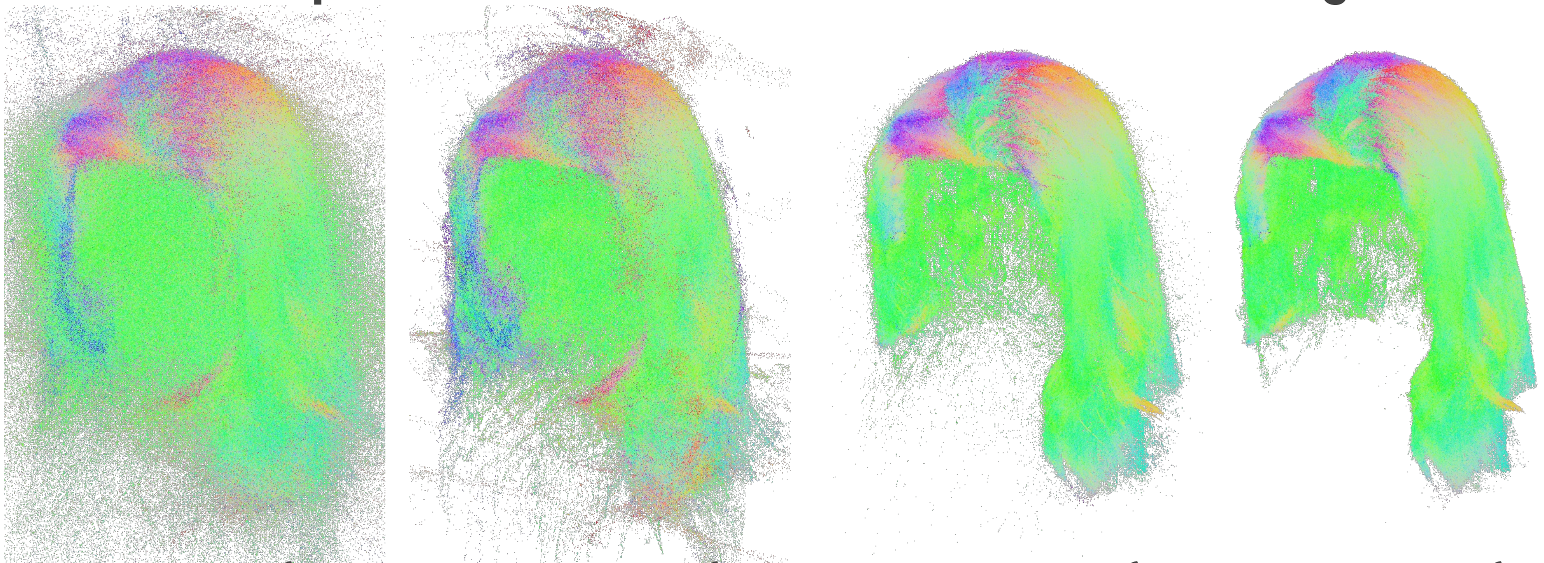
LPMVS [Nam+]



Strand Integration

Result: Real Data

All point cloud | **After filtering**



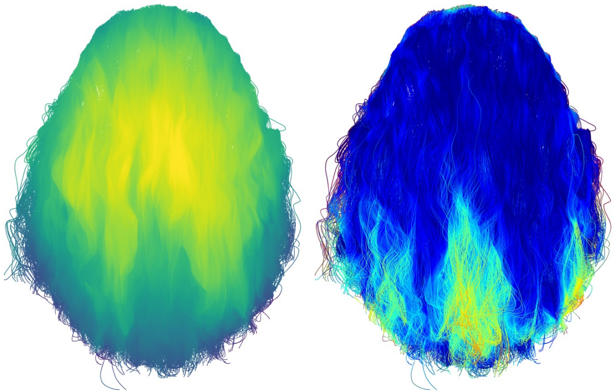
Number of points

	LPMVS [Nam+]	Strand Integration	LPMVS [Nam+]	Strand Integration
Number of points	1056×10^6	1056×10^6	214×10^6	300×10^6

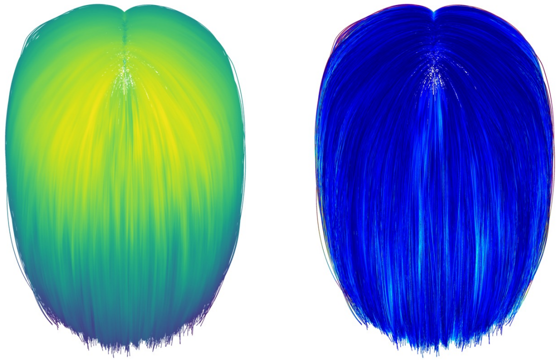
Limitation

- Assume that hair is continuous and coherent everywhere.
- Break when the hair is strongly curled or scattered.

Curly



Straight



MAE / RMSE

18.10 / 27.07

9.76 / 15.03

Official Code on GitHub

Official implementation
Strand Integration

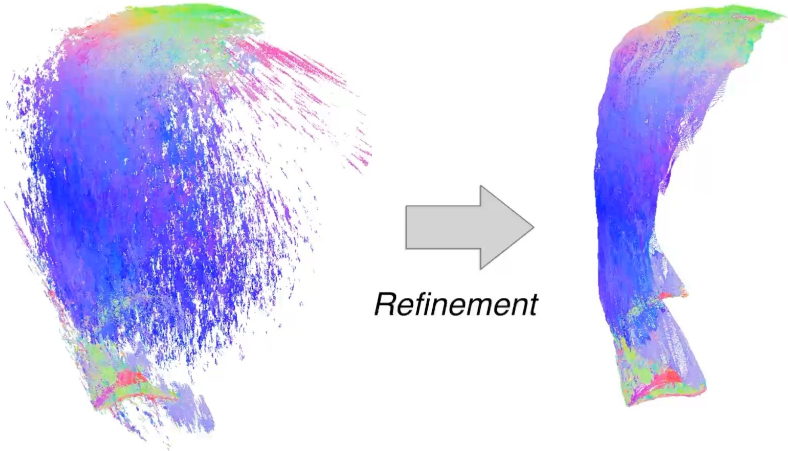
Unofficial implementation
LPMVS
[Nam el al., CVPR2019]

Synthetic data of multi-view images

Project page

Conclusion

- **Strand Integration: Refine the inaccurate hair strand by integrating the gradient along the hair strand.**



LPMVS [Nam+, CVPR'19] **Strand Integration (ours)**

Project page