

NeuralFur: Animal Fur Reconstruction From Multi-View Images (Supplementary)

Vanessa Sklyarova^{1,2*}

Berna Kabadayi^{1,4*}
Michael J. Black¹

Anastasios Yiannakidis¹
Justus Thies^{1,3}

Giorgio Becherini¹

¹Max Planck Institute for Intelligent Systems ²ETH Zürich

³Technical University of Darmstadt ⁴University of Tübingen

1. Preliminaries

Parametric Animal Model. SMAL [13] is a parametric model for quadruped animals. Formally, it is a function $M(\beta, \theta, \gamma)$ that outputs a posed 3D animal shape, given shape parameters β , pose parameters θ , and global translation γ . Template vertices $\mathbf{v}_t \in \mathbb{R}^{3889 \times 3}$ are transformed using specified shape parameters β and learned PCA shape space \mathbf{B} , then articulated using θ through LBS with joint regressor \mathbf{J}_r and LBS weights \mathbf{W} :

$$\mathbf{v}_d = \text{LBS}(\mathbf{v}_t + \beta \cdot \mathbf{B}, \theta; \mathbf{W}, \mathbf{J}_r) + \gamma.$$

θ represents the relative rotations of joints in the kinematic tree. Translation γ is applied to the root joint.

Gaussian Splatting. To use 3D Gaussian Splatting for soft-rasterization of fur, we force Gaussians to lie on line segments. We define the mean and covariance matrix of each individual Gaussian as:

$$\mu_i^l = \frac{1}{2}(\mathbf{p}_i^l + \mathbf{p}_i^{l+1}), \quad C_i^l = E_i^l D_i^l (E_i^l D_i^l)^T. \quad (1)$$

Here, $E_i^l = \{b_i^l, t_i^l, n_i^l\}$ is a TBN basis associated with the strand curve, $\mathbf{b}_i^l = \mathbf{d}_i^l / \|\mathbf{d}_i^l\|_2$, where $\mathbf{d}_i^l = \mathbf{p}_i^{l+1} - \mathbf{p}_i^l$ denotes the segment vector and \mathbf{b}_i^l its normalized direction vector. D_i^l is defined as $D_i^l = \text{diag}(\mathbf{f}_i^l, \epsilon, \epsilon)$, where \mathbf{f}_i^l is set to be proportional to the length of \mathbf{d}_i^l and ϵ denotes a small value. Such parametrization allows effective propagation of photometric information into fur geometry.

2. Training details

2.1. Strands parametrization

Basis calculation. We map each strand into a local tangent-bitangent-normal (TBN) basis using the vertices from the

closest face to its root location on the M_{bald} mesh. Each strand consists of $L = 100$ points. The normal is computed on the mesh and is consistent with nearby strands. In contrast, the definition of a tangent vector is problematic. Different to prior works on hair reconstruction [8, 9, 12], we align tangents with the computed directional face field with further sign consistency resolving using parallel transport [10] across shared edges. The bitangent vector is then defined as a cross-product between the normal and the tangent. For directional field estimation, we use the library <https://github.com/avaxman/Directional>.

2.2. Optimization details

We launch the NeuS [11] reconstruction method for 300,000 iterations, which takes around 6 hours on a single A100. For SMAL fitting, we use the Adam optimizer [4] with different learning rates depending on the optimization stage. Following WLDO [1], we minimize the following energy function:

$$L_{\text{total}} = L_{\text{chamfer}} + \mathcal{L}_{\text{laplacian}} L_{\text{laplacian}} + \mathcal{L}_{\text{edge}} L_{\text{edge}} + \mathcal{L}_{\text{normal}} L_{\text{normal}}, \quad (2)$$

where $\mathcal{L}_{\text{laplacian}} = 0.01$, $\mathcal{L}_{\text{edge}} = 0.8$, $\mathcal{L}_{\text{normal}} = 0.02$. For the final optimization of our model, we use the following weights for losses: $\mathcal{L}_{\text{sil}} = 0.1$, $\mathcal{L}_{\text{dir}} = 1000$, $\mathcal{L}_{\text{dir}}^{\text{gpt}} = 1$, $\mathcal{L}_{\text{chm}} = 20$, $\mathcal{L}_{\text{penetr}} = 1$, $\mathcal{L}_{\text{shape}} = 0.01$ and strand width 0.0025. We use $\gamma = 10$ for positional encodings.

2.3. De-furring process

For the de-furring process, we convert the mesh to an SDF. We first estimate the effective strand thickness for each region using ChatGPT, then smooth these values via neighborhood averaging. The mesh is converted to an SDF, shrunk according to the predicted thickness values, and converted back using marching cubes with resolution 256. Finally, we resample the resulting mesh to 160,000 faces and

* Equal contribution

apply basic mesh repair operations to remove noise and artifacts.

2.4. Prompting

To obtain annotations of an animal, we sent two images from frontal and side views and then provided a set of prompts.

Length annotations. We annotate the fur length using the following prompt: “Here, you could see images of an animal. Could you estimate the accurate fur length in cm for each part of the animal: “leg_front”, “leg_rear”, “paws”, “front_paws”, “belly”, “neck”, “face”, “ears”, “under_tail”, “tail”, “body”, “paw_pads”, “inner_earcanal”, “eyes”, “nosetip”? Also, does this animal have some fur near the neck that grows significantly beyond the underlying body? If so, could you add the estimates for length and fur thickness as well for “mane” and include it at the end of the previous part names; otherwise, do not include it. Is the ear canal visible in the image? If the inner ear canal is not visible, use the same value for length as for the outer ear. Please provide results in dict format, where the key is part name and the value is your estimate.”

Effective fur thickness. To determine the effective fur thickness for the subsequent de-furring step, we employ the following prompt: “I want to create a furless animal. To do that, I have a 3D model of an animal with fur, from which I want to subtract the region covered by fur for each part. Could you provide the number of effective fur thickness for each part that I need to subtract from the full geometry? Please provide results in dict format where the key is part name and the value is your estimate.” Finally, we verify that the estimates are correct and correspond to the image using the following prompt: “Are you sure of the initial length estimations and fur thickness to abstract? Could you double-check with the image? Please provide results in a several-dict format where the key is part name and the value is your estimate.”

Hair growing direction. To determine the hair growth direction, we provide the coordinate space, specify its relation to the animal, and then ask: “Could you also estimate what the fur growing direction looks like in 3d space? I want to know which parts are oriented along the gravity vector and which are against. Also, if fur for regions grows from left to right or from right to left. I have the following coordinate system: x from the right side of animal towards the left, y - opposite to gravity direction, and z from back to the front of the animal. The second image is aligned with the 3D coordinate system. I want for each part: “leg_front”, “leg_rear”, “paws”, “front_paws”, “belly”, “neck”, “face”, “ears”, “under_tail”, “tail”, “body”, “paw_pads”, “inner_earcanal”, “eyes”, “nosetip”, “mane” (if appeared) obtains a vector that defines the approximate

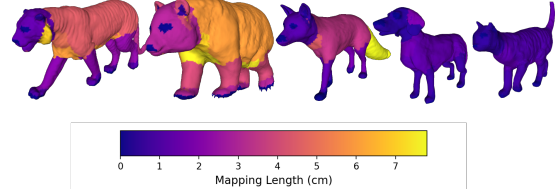


Figure 1. Length annotations obtained from ChatGPT for all animals.

growing direction in the coordinate system. Please double-check the proposed directions several times.”

Eyeballs annotations. To annotate the distance between the centers of the eyes, we employ the following prompt: “Could you measure the distance between eyeballs in cm? You could also use the real-world understanding of this animal class.”

The final length annotations for each animal part obtained from ChatGPT are shown in centimeters in Table 1. Figure 1 illustrates the distribution of fur lengths across all animals, while the distribution of effective fur thickness is shown in Figure 2.

| Body Part | “Cat” | “Beagle dog” | “Fox” | “Panda” | “White Tiger” |
|------------------|-------|--------------|-------|---------|---------------|
| “leg_front” | 0.8 | 0.7 | 1.35 | 4 | 2.1 |
| “leg_rear” | 0.8 | 0.7 | 1.75 | 4 | 2.3 |
| “paw_pads” | 0.1 | 0.0 | 0 | 0 | 0 |
| “paws” | 0.45 | 0.4 | 0.6 | 3 | 1.1 |
| “front_paws” | 0.45 | 0.4 | 0.6 | 3 | 1.1 |
| “belly” | 1.25 | 1.2 | 3.5 | 7.5 | 6 |
| “neck” | 1.2 | 1.5 | 4.5 | 7 | 5 |
| “face” | 0.5 | 0.4 | 1.1 | 3 | 1.35 |
| “ears” | 0.3 | 1.8 | 0.6 | 4 | 1.5 |
| “inner_earcanal” | 0.15 | 1.8 | 0.25 | 0.5 | 0.4 |
| “under_tail” | 1 | 1.2 | 5 | 5 | 3 |
| “eyes” | 0 | 0 | 0 | 0 | 0 |
| “tail” | 1.35 | 1.8 | 7.8 | 6 | 3.25 |
| “nosetip” | 0 | 0.0 | 0 | 0 | 0 |
| “body” | 1 | 1.3 | 3.5 | 6 | 4.5 |
| “mane” | - | - | - | - | 7 |

Table 1. Part lengths annotations for different animals obtained from ChatGPT in cm. Missing values are indicated by “-”. Naming of animals taken from the Artemis [5] dataset.

2.5. Data preprocessing

We use multi-view images of animals with ground-truth silhouettes, along with camera parameters provided in the Artemis [5] dataset. After obtaining a full mesh reconstruction from NeuS [11], denoted as M_{NeuS} , we fit SMAL to it to transfer body part labels. Next, we use ChatGPT to obtain effective length and thickness annotations for each

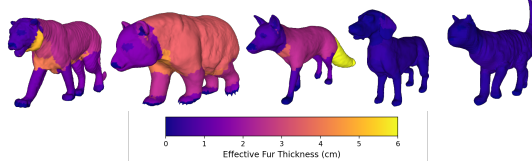


Figure 2. Effective fur thickness annotations obtained from ChatGPT for all animals.

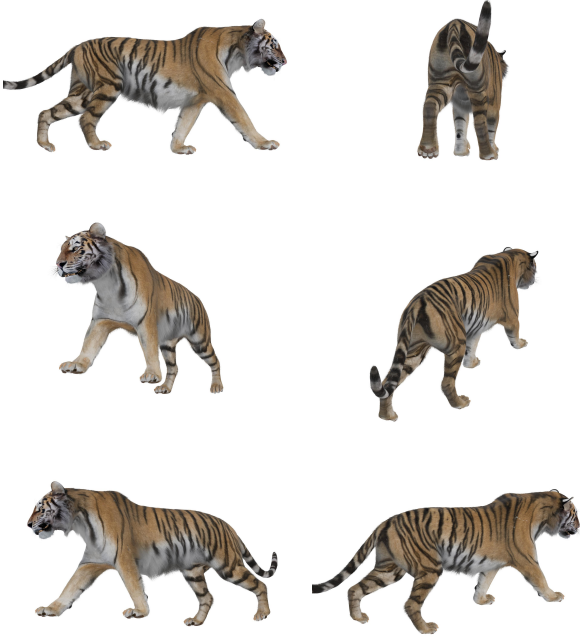


Figure 3. Rendered in Blender [2] asset used for quantitative evaluation.

part. Based on this information, we reconstruct the defurred geometry M_{bald} . We then re-fit SMAL to M_{bald} to obtain vertex-level length and direction annotations, which are subsequently used during the optimization process. Finally, we compute the SDF field for M_{bald} , which is required for the penetration loss.

2.6. Baselines

The GenZoo [7] results were provided by the corresponding authors, while NeuS [11], AniMer [6], and GaussianHair-cut [12] were run using their respective public repositories.

2.7. Datasets

Our model is evaluated on five scenes from the Artemis [5] dataset. For quantitative evaluation we use synthetic asset with ground-truth geometry from https://superhivemarket.com/products/complete-realistic-tiger---texture--fur--armature?search_id=41647510. Figure 3

illustrates several views generated by rendering the scene in Blender [2].

2.8. Metrics

For quantitative evaluation on a synthetic asset, we compute metrics using 100,000 strands sampled from both the ground-truth and generated assets. While the reconstructed furred animal appears visually close to the ground truth, we observe relatively high metric values due to (1) inaccuracies in the defurring procedure and (2) inaccuracies in the strand length prediction process from VLM. Including additional synthetic assets for benchmarking is a potential direction for future work.

To assess the consistency of the generated strands, we compute self-supervised metrics for local and global consistency in terms of length, direction, and curvature. Let M be the number of points per strand. To compute the length of each strand, we use the following equation:

$$L_i = \sum_{l=0}^{M-2} \|\mathbf{p}_i^{l+1} - \mathbf{p}_i^l\|_2. \quad (3)$$

We calculate the strand’s average tangent direction by:

$$\tilde{\mathbf{t}}_i = \frac{1}{M-1} \sum_{l=0}^{M-2} \mathbf{b}_i^l \quad (4)$$

$$\mathbf{t}_i = \frac{\tilde{\mathbf{t}}_i}{\|\tilde{\mathbf{t}}_i\|_2}. \quad (5)$$

Based on these measurements, we define the following metrics:

Local length variance: The local length variance is defined as follows:

$$\sigma_{\text{loc}}(L) = \sqrt{\frac{1}{N} \sum_{i=1}^N \frac{1}{k} \sum_{j \in \text{knn}(i)} (L_i - L_j)^2}. \quad (6)$$

Global curvature variance: The global curvature variance is defined analogously to the optimization loss in the main paper: $\text{Var}_{\text{glob}}(\kappa) \equiv \mathcal{L}_{\text{shape}}$.

Local curvature variance: The local curvature variance is formulated in a similar manner, but restricted to the k nearest neighbors:

$$\text{Var}_{\text{loc}}(\kappa) = \frac{1}{N} \sum_{i=1}^N \frac{1}{k} \sum_{j \in \text{knn}(i)} \frac{1}{M-2} \sum_{l=1}^{M-2} (\theta_i^l - \theta_j^l)^2. \quad (7)$$

Local direction consistency: Similarly, we define local direction consistency. We first evaluate the consistency score based on the initial segment directions:

$$\text{Var}_{\text{loc}}^{\text{dir}}(\text{dir}) = \frac{1}{N} \sum_{i=1}^N \frac{1}{k} \sum_{j \in \text{knn}(i)} \|\mathbf{b}_i^0 - \mathbf{b}_j^0\|_2^2. \quad (8)$$

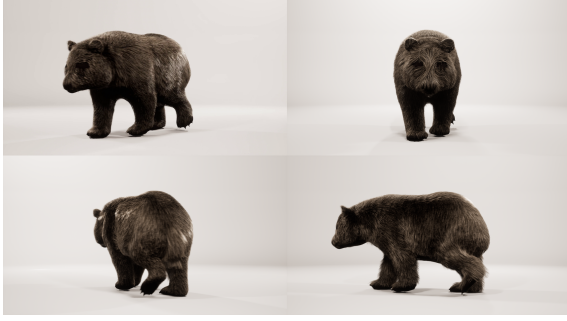


Figure 4. **Fur editing results.** Top-left: original image; top-right: face edited; bottom-left: tail edited; bottom-right: legs edited

Local direction variance: The local direction variance is calculated as follows:

$$\text{Var}_{\text{loc}}(\text{dir}) = \frac{1}{N} \sum_{i=1}^N \frac{1}{k} \sum_{j \in \text{knn}(i)} \|\mathbf{t}_i - \mathbf{t}_j\|_2^2. \quad (9)$$

Maximum bending angle: Lastly, the maximum bending angle is given by the following expression:

$$\kappa_{\max} = \max_{i,l} \theta_i^l. \quad (10)$$

3. Applications

Rendering. For rendering in Unreal Engine [3], we use 500,000 strands. Figure 8 shows the results of importing reconstructed furless animals and fur into Unreal Engine and rendering them with the specified colors.

Simulations. We simulate the reconstructed fur using the fitted SMAL [13] model along with its kinematic tree. Additional results are provided in the Supplementary Video.

Editing. Our 3D part-based annotations allow region-specific editing, providing fine-grained control over fur on different body areas. For example, in Figure 4, we show edits where the tail fur length is changed from 6 cm to 12 cm, leg fur from 4 cm to 8 cm, and face fur from 3 cm to 5 cm, while fur on other regions remains unchanged. More results are available in the Supplementary Video.

4. Additional experiments

4.1. Reconstructed results

In Figure 9, we present additional results comparing our method with coarse reconstruction baselines, including SMAL [13], GenZoo [7], AniMer [6] and NeuS [11]. We also compare against the strand-based hair reconstruction method, Gaussian Haircut [12], as shown in Figure 10. For

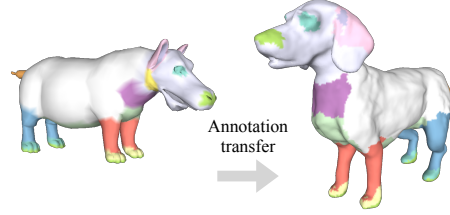


Figure 5. Transfer of the part annotations from the template (left) to a dog (right). Errors in body part transfer are mainly due to limitations of the fitted SMAL model, particularly for animals with large ears.

Gaussian Haircut, comparisons are provided after the second and third stages, denoted as GH (2nd stage) and GH (3rd stage), respectively.

4.2. Extended ablation

Quantitative and qualitative results. In Figure 7, we present an additional ablation study examining strand length, the strand encoder, and the impact of the de-furring strategy. Table 2 reports self-supervised metrics on an additional scene.

| Method | "Cat" | | | | | | | | |
|---|------------------------------------|-------------------------------------|---|--|----------------------------|--|---|-----------------|--|
| | Fur length | | Fur curvature | | | Fur orientation | | Fur silhouette | |
| | $\mu_L \pm \sigma_L$ (cm) | $\sigma_{\text{loc}}(L) \downarrow$ | $\text{Var}_{\text{glob}}(\kappa) \downarrow$ | $\text{Var}_{\text{loc}}(\kappa) \downarrow$ | $\kappa_{\max} \downarrow$ | $\text{Var}_{\text{loc}}(\text{dir}) \downarrow$ | $\text{Var}_{\text{glob}}(\text{dir}) \downarrow$ | CD \downarrow | |
| Ours | 0.889 \pm 0.29 | 0.046 | 0.0006 | 0.000082 | 1.70 | 0.027 | 0.048 | 0.000354 | |
| GH (2nd stage) | 3.830 \pm 2.77 | 0.945 | 0.0005 | 0.000098 | 2.79 | 0.096 | 0.093 | 0.001907 | |
| GH (3rd stage) | 5.036 \pm 2.59 | 1.591 | 0.0967 | 0.151766 | 3.13 | 0.575 | 0.632 | 0.003013 | |
| w/ opt. length | 8.255 \pm 8.38 | 0.817 | 0.0066 | 0.000635 | 3.04 | 0.032 | 0.044 | 0.001143 | |
| w/ fix. length | 1.000 \pm 0.00 | 0.000 | 0.0008 | 0.000098 | 2.06 | 0.033 | 0.046 | 0.000389 | |
| w/ Unet | 0.889 \pm 0.29 | 0.046 | 0.0003 | 0.000375 | 2.68 | 0.077 | 0.119 | 0.001778 | |
| w/o defur | 0.906 \pm 0.29 | 0.046 | 0.0010 | 0.000313 | 3.01 | 0.046 | 0.088 | 0.001045 | |
| w/o \mathcal{L}_{chm} | 0.889 \pm 0.29 | 0.046 | 0.0007 | 0.000094 | 2.32 | 0.030 | 0.056 | 0.000445 | |
| w/o $\mathcal{L}_{\text{dir}}^{\text{opt}}$ | 0.889 \pm 0.29 | 0.046 | 0.0010 | 0.000218 | 2.84 | 0.033 | 0.046 | 0.000360 | |
| w/o $\mathcal{L}_{\text{shape}}$ | 0.889 \pm 0.29 | 0.046 | 0.0034 | 0.000448 | 3.02 | 0.029 | 0.085 | 0.000365 | |

Table 2. Unsupervised geometry consistency metrics for length, direction, and curvature, evaluated for both local and global cases, as well as distance between roots and the outer surface.

Number of views. In Figure 6, we present an ablation study on the number of views used for supervision during the second stage of strand-based fur optimization. Results are shown for 1, 2, 4, 8, and 16 input views, compared against the full 36-view setting. Note that all results are rendered from novel viewpoints.

5. Limitations

Our goal is to automatically annotate parts of various animals by fitting a parametric SMAL [13] model. Nevertheless, errors in the fitted SMAL model may result in incorrect annotations, such as the dog ear being labeled as "face" in Figure 5. Still, our method provides explicit control over fur growth: it does not grow any fur in regions that naturally lack fur, such as paw pads or nose tips (see Figure 9), showing that it might potentially generalize to hairless or

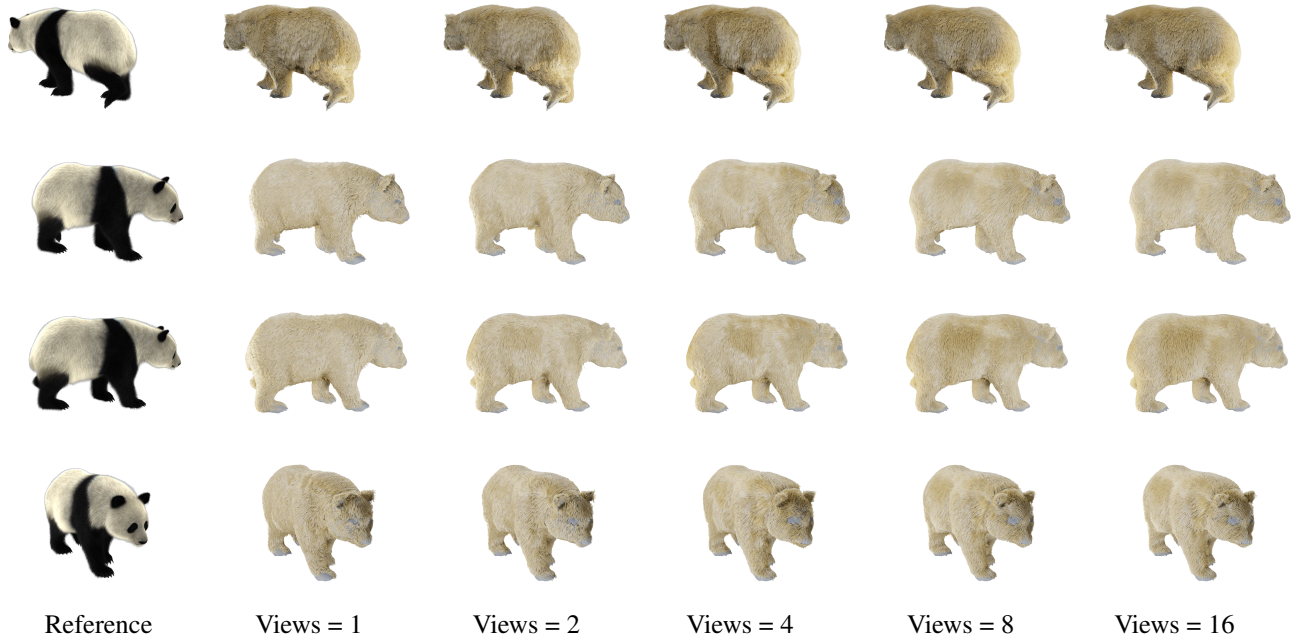


Figure 6. **Extended ablation study.** Qualitative evaluation of our method with different number of views. Even with a low number of views, our method is able to output reasonable fur reconstructions. However, the quality increases with more input views available.

partially hairless species. Nonetheless, its applicability is constrained by the SMAL model itself, which represents quadrupeds and cannot accurately represent animals with very different body structures (e.g., ducks or crocodiles). One possible solution to reconstruct animal fur without relying on SMAL is to bypass SMAL entirely during annotation and instead annotate parts directly on the reconstructed NeuS geometry. Although this limits the automatic transfer of fur length to corresponding body parts, such an approach would eliminate the dependency on a parametric model template and potentially might extend fur reconstruction to a broader range of animal species.



Figure 7. **Extended ablation study.** Qualitative evaluation of our design choices regarding length, fur parametrization, and the importance of the defurring approach for accurate geometry modeling.



Figure 8. **Rendering examples.** Reconstructed strand-based fur could be easily rendered with predefined colors in Unreal Engine [3].

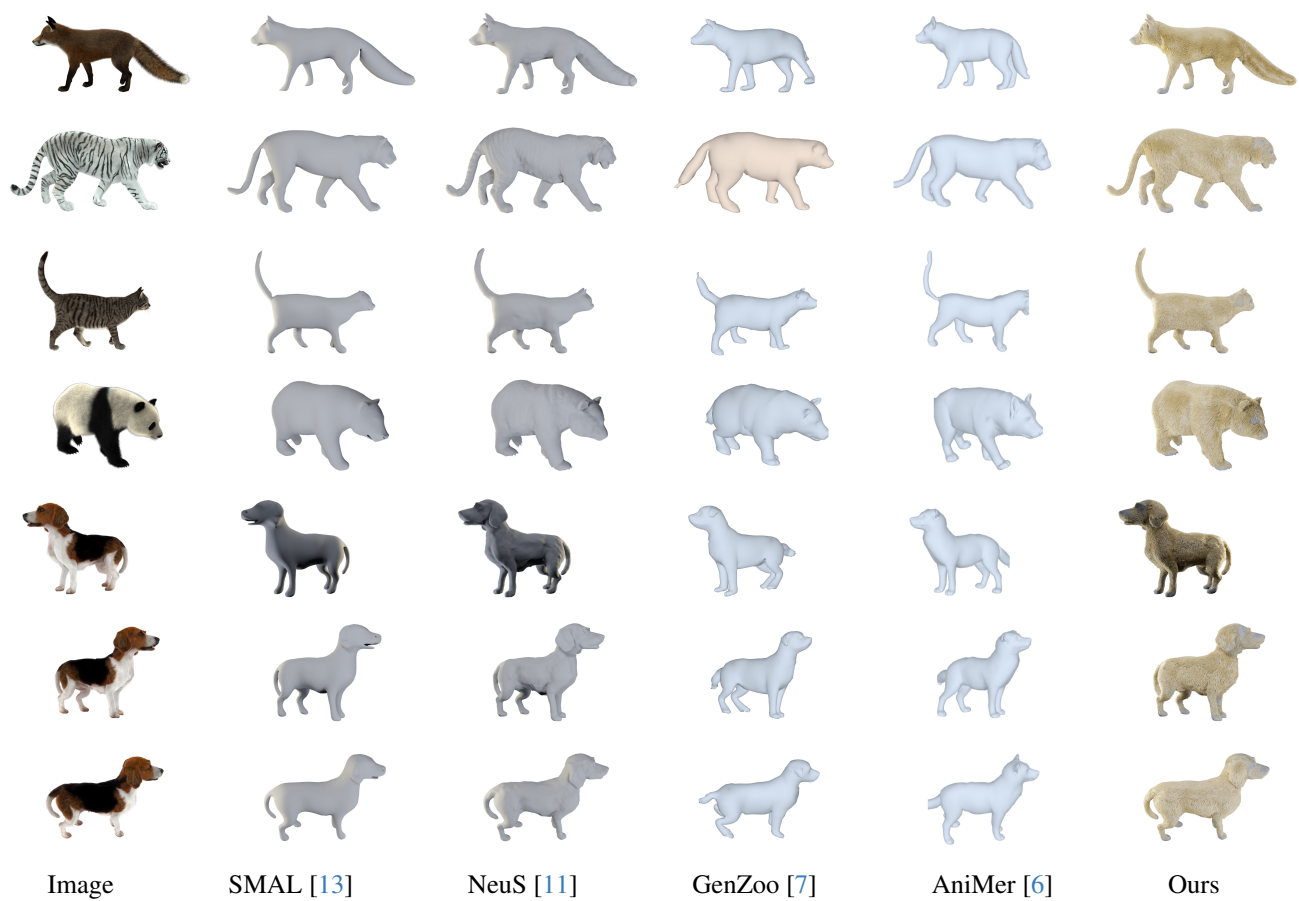


Figure 9. **Extended comparison with baselines.** Surface reconstruction methods produce very coarse geometry.



Figure 10. **Extended qualitative comparison with Gaussian Haircut [12]**, the state-of-the-art strand-based hair reconstruction method. Here, we compare with the results obtained after the second and third stages, see GH (2nd stage) and GH (3rd stage), respectively. Digital zoom-in is recommended.

References

- [1] Benjamin Biggs, Oliver Boyne, James Charles, Andrew Fitzgibbon, and Roberto Cipolla. Who left the dogs out?: 3D animal reconstruction with expectation maximization in the loop. In *ECCV*, 2020. 1
- [2] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2023. 3
- [3] Epic Games. Unreal engine. 4, 7
- [4] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 1
- [5] H. Luo, A. Chen, Q. Zhang, B. Pang, M. Wu, L. Xu, and J. Yu. Convolutional neural opacity radiance fields. In *2021 IEEE International Conference on Computational Photography (ICCP)*, pages 1–12, Los Alamitos, CA, USA, 2021. IEEE Computer Society. 2, 3
- [6] Jin Lyu, Tianyi Zhu, Yi Gu, Li Lin, Pujin Cheng, Yebin Liu, Xiaoying Tang, and Liang An. Animer: Animal pose and shape estimation using family aware transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17486–17496, 2025. 3, 4, 8
- [7] Tomasz Niewiadomski, Anastasios Yiannakidis, Hanz Cuevas-Velasquez, Soubhik Sanyal, Michael J. Black, Silvia Zuffi, and Peter Kulits. Generative zoo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 3, 4, 8
- [8] Radu Alexandru Rosu, Shunsuke Saito, Ziyang Wang, Chenglei Wu, Sven Behnke, and Giljoo Nam. Neural strands: Learning hair geometry and appearance from multi-view images. In *European Conference on Computer Vision*, 2022. 1
- [9] Vanessa Sklyarova, Jenya Chelishev, Andreea Dogaru, Igor Medvedev, Victor Lempitsky, and Egor Zakharov. Neural haircut: Prior-guided strand-based hair reconstruction. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2023. 1
- [10] M. Spivak. *A Comprehensive Introduction to Differential Geometry Vol. II*. Publish or Perish, 1979. 1
- [11] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 1, 2, 3, 4, 8
- [12] Egor Zakharov, Vanessa Sklyarova, Michael Black, Giljoo Nam, Justus Thies, and Otmar Hilliges. Human hair reconstruction with strand-aligned 3d gaussians. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XVI*, page 409–425, Berlin, Heidelberg, 2024. Springer-Verlag. 1, 3, 4, 9
- [13] Silvia Zuffi, Angjoo Kanazawa, David Jacobs, and Michael J. Black. 3D menagerie: Modeling the 3D shape and pose of animals. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 4, 8