

# High-quality Animatable Eyelid Shapes from Lightweight Captures

JUNFENG LYU, School of Software and BNRIst, Tsinghua University, China

FENG XU, School of Software and BNRIst, Tsinghua University, China

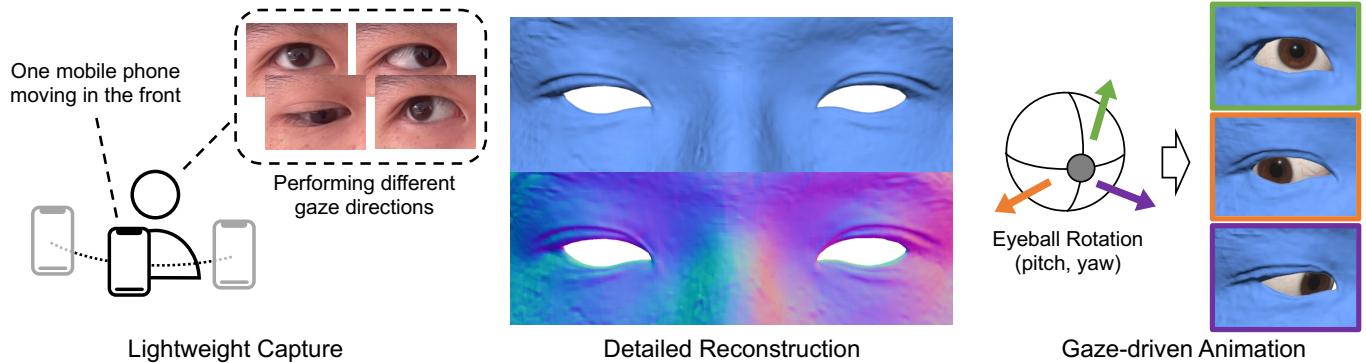


Fig. 1. Our approach can leverage lightweight capture to achieve detailed reconstruction and gaze-driven animation of the eyelids.

High-quality eyelid reconstruction and animation are challenging for the subtle details and complicated deformations. Previous works usually suffer from the trade-off between the capture costs and the quality of details. In this paper, we propose a novel method that can achieve detailed eyelid reconstruction and animation by only using an RGB video captured by a mobile phone. Our method utilizes both static and dynamic information of eyeballs (e.g., positions and rotations) to assist the eyelid reconstruction, cooperating with an automatic eyeball calibration method to get the required eyeball parameters. Furthermore, we develop a neural eyelid control module to achieve the semantic animation control of eyelids. To the best of our knowledge, we present the first method for high-quality eyelid reconstruction and animation from lightweight captures. Extensive experiments on both synthetic and real data show that our method can provide more detailed and realistic results compared with previous methods based on the same-level capture setups. The code is available at <https://github.com/StoryMY/AniEyelid>.

CCS Concepts: • Computing methodologies → Reconstruction; Animation.

Additional Key Words and Phrases: Eyelid Reconstruction, Gaze-driven Animation, RGB Videos

#### ACM Reference Format:

Junfeng Lyu and Feng Xu. 2024. High-quality Animatable Eyelid Shapes from Lightweight Captures. In *SIGGRAPH Asia 2024 Conference Papers (SA Conference Papers '24)*, December 3–6, 2024, Tokyo, Japan. ACM, New York, NY, USA, 15 pages. <https://doi.org/10.1145/3680528.3687583>

Authors' addresses: Junfeng Lyu, School of Software and BNRIst, Tsinghua University, Beijing, China, [ljf19@mails.tsinghua.edu.cn](mailto:ljf19@mails.tsinghua.edu.cn); Feng Xu, School of Software and BNRIst, Tsinghua University, Beijing, China, [feng-xu@tsinghua.edu.cn](mailto:feng-xu@tsinghua.edu.cn).

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SA Conference Papers '24, December 3–6, 2024, Tokyo, Japan

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-1131-2/24/12

<https://doi.org/10.1145/3680528.3687583>

## 1 INTRODUCTION

Realistic digital humans are widely used in computer games, films, and metaverse. As “the windows to the soul”, eyes play a key role in digital humans, which are significant for creating vivid animation and conveying mental activities. However, although face reconstruction and animation have received much attention from the community, the quality of eyelids still needs to be improved to overcome the uncanny valley.

Obtaining high-quality eyelids is challenging in two aspects. First, the shape of eyelids varies across identities and has subtle details that are hard to capture. Second, the dynamics of eyelids are also very complicated as complex non-rigid deformations like skin folding and unfolding often occur during the movement.

Due to these difficulties, previous methods [Kerbirou et al. 2022; Wood et al. 2016a,b] have to make a trade-off between the capture costs and the detail quality. Typically, in order to reconstruct detailed eyelids, Bermano et al. [2015] propose a spatio-temporal eyelid reconstruction method that depends on a multi-camera setup. However, this method does not support animating reconstructed eyelids, and the requirement of a high-cost system strictly limits its daily applications. On the other hand, based on the pre-designed blendshapes, Wen et al. [2017] reconstructs eyelids by a single RGB-D camera while retaining some subtle details like folds and bulges, but the reconstructed geometry is inevitably bottlenecked by the capability of the model space. Recently, the advantages of SDF-based [Wang et al. 2021; Yariv et al. 2021, 2023] and Gaussian-based methods [Guédon and Lepetit 2024; Huang et al. 2024; Kerbl et al. 2023; Yu et al. 2024] over traditional reconstruction methods [Barnes et al. 2009; Broadhurst et al. 2001; Furukawa and Ponce 2009; Kazhdan et al. 2006; Seitz and Dyer 1999] motivate us to employ new representations for eyelid modeling. Considering the ability of both static and dynamic reconstruction, SDF is the superior choice due to its competitive performance in static geometry modeling and its proven effectiveness in dynamic scenes [Cai et al. 2022; Shao et al. 2023], while Gaussian-based methods require further exploration

in dynamic geometry reconstruction. Compared with widely used meshes, SDF can naturally avoid self-interaction and seamlessly integrate with continuous neural fields [Park et al. 2019], which is cost-effective and worth exploring in eyelid reconstruction.

In this paper, we propose a novel SDF-based method that achieves high-quality geometry reconstruction and animation of eyelids with an RGB video recorded by a single mobile phone. The input video contains an actor-oriented sequence with the camera moving in front and the actor performing different gaze directions, which is only a sparse sampling of view-gaze pairs and thus easy to shoot. With such a video, we learn a dynamic neural SDF field to model the dynamic eyelids. The dynamics are modeled by a canonical hyper-space [Park et al. 2021b] cooperating with an invertible deformation field [Cai et al. 2022]. The key to our method is to consider eyeball influence on eyelids from both perspectives of eyeball shapes and motions. These can provide additional constraints over the limited inputs and help to reconstruct gaze-related details during eye movements. Specifically, for eyeball shapes, we propose a contact loss to make the eyelids tightly adhere to the eyeball surface, which provides geometry priors to compensate for the lack of depth information in monocular RGB recording. For eyeball motions, we learn a novel gaze-dependent adaptive anchor grid for eyelid representation. This provides a gaze-varying geometry encoding to better model the subtle details of dynamic eyelids.

To acquire eyeball information to assist eyelid reconstruction, we design an automatic eyeball calibration method to identify the 3D eyeball positions based on differentiable mesh rendering [Laine et al. 2020]. Different from previous methods [Lu et al. 2020; Sun et al. 2015; Wang and Ji 2017; Wen et al. 2020], our calibration does not need 2D or 3D gaze targets as input. It works with videos containing sparse samples of view-gaze pairs, so no additional recording is required. By leveraging the idea that only the correct 3D eyeball position can depict the appearance of all gazes, we optimize the eyeball position and per-frame rotations via aligning the rendering results with the eyeball appearance (Sec. 3.1).

Apart from reconstruction, the eyeball information also benefits the eyelid animation. As the movement of eyelids often contains identity-dependent deformation, it is costly to cover all the deformation by man-made patterns [Wen et al. 2017] or empirical modeling [Wood et al. 2016b]. To solve this, we train an eyelid control network (Sec. 3.3) for each identity to model the identity-specific deformation driven by the eyeball rotation. We learn the mapping along with the dynamic neural SDF field, making it possible to control the complicated eyelid movements by simple and semantic parameters (i.e., eyeball rotations). As there are usually some eye-irrelevant motions (e.g., moving head, pursing lips) during real-data capture, we introduce a freely learnable latent code to model this part of motions and a strategy to disentangle the two kinds of motions.

In summary, our main contributions are listed as follows:

- To the best of our knowledge, we are the first to achieve detailed eyelid reconstruction and animation by only using an RGB video from one mobile phone as input.
- We utilize eyeball information to improve eyelid reconstruction, including both eyeball shapes and motions, by a contact loss and a gaze-dependent adaptive anchor grid, respectively.

- We develop an eyelid control module to achieve semantic animation control over the identity-specific eyelid deformation which is well trained in a disentanglement strategy.

## 2 RELATED WORKS

### 2.1 3D Head Avatars

Early works [Garrido et al. 2016; Thies et al. 2016] adopt the 3D morphable model [Blanz and Vetter 1999] to recover the shape and appearance of human faces. Recently, many works [Chen et al. 2023; Gafni et al. 2021; Gao et al. 2022; Xu et al. 2023; Zhao et al. 2023; Zielonka et al. 2023] utilize the neural radiance field [Mildenhall et al. 2021] to model the human face, while other studies [Ma et al. 2024; Qian et al. 2024; Xiang et al. 2024; Xu et al. 2024] employ 3D Gaussian splats [Kerbl et al. 2023] for the same purpose. However, these methods usually focus on the photorealistic appearance and can hardly create accurate geometry of the faces. For better geometry, Cao et al. [2022] utilize high-quality priors to generate photorealistic avatars. Neural Head Avatars [Grassal et al. 2022] represents detailed geometry through per-vertex offsets on top of a coarse template. IMAvatar [Zheng et al. 2022] learns a canonical space and a series of expression deformations to model the dynamic geometry, while PointAvatar [Zheng et al. 2023] employs point cloud representation to achieve faster rendering and better appearance. FLARE [Bharadwaj et al. 2023] apply the physically-based rendering to disentangle the material and illumination, resulting in a relightable avatar. However, these methods mainly focus on the whole face, and the geometry of eye region is relatively coarse.

### 2.2 Eye Reconstruction and Animation

The research of eye modeling mainly focus on the eyeball and eyelid. The high-quality eyeball reconstruction and rigging is well addressed by a series of work [Bérard et al. 2016, 2019, 2014]. For eyelid modeling, Bermano et al. [2015] reconstruct detailed eyelids based on a multi-camera capture setup, which is too expensive for consumer-level applications. To solve this, Wood et al. [2016a] build a 3D morphable model of the eye from high-fidelity scans, which can be fitted with a single image. Similarly, Wen et al. [2017] design a set of eyelid blendshapes and propose a real-time eyelid tracking method based on one RGB-D camera, but the quality of subtle details is constrained by the capability of the model space. As for eyelid animation, gaze or eyeball rotation is a common parameter for controlling eyelid deformation in traditional scheme [Kerbirou et al. 2022; Neog et al. 2016; Wood et al. 2016b]. Different from these methods, our eyelid control is based on a neural architecture like Schwartz et al. [2020], Li et al. [2022], and Li et al. [2024], but with a more lightweight input and more attention to the geometry. To the best of our knowledge, we are the first to make detailed eyelid reconstruction and animation based on lightweight captures.

### 2.3 Neural Implicit Surfaces

Different from neural radiance field [Mildenhall et al. 2021] that models the scene by density, neural implicit surfaces [Wang et al. 2021; Yariv et al. 2021, 2023] represent the geometry as a neural SDF field [Park et al. 2019], achieving better static surface reconstruction with multi-view RGB inputs. NDR [Cai et al. 2022] and Tensor4D

[Shao et al. 2023] extend the method to dynamic scenes by applying deformation field [Fang et al. 2022; Park et al. 2021a,b; Pumarola et al. 2021; Tretschk et al. 2021], but rely on RGB-D and multi-camera setup to guarantee the geometry quality. Despite their good results on the large-scale geometry, some subtle details are still missing. For detailed reconstruction, recent works [Cai et al. 2023; Fu et al. 2022; Li et al. 2023; Rosu and Behnke 2023] have made explorations in static reconstruction. In this paper, we propose an gaze-dependent adaptive anchor grid to extend the static techniques to dynamic scenes, achieving detailed reconstruction of the moving eyelids.

### 3 METHOD

The architecture of the proposed method is illustrated in Fig. 3. The input is an actor-oriented RGB video captured by one camera (e.g., mobile phone) moving in the front of the actor performing different gaze directions. Additionally, we adopt the video segmentation method [Lin et al. 2022] to get the mask of the actor. Our goal is to get a dynamic neural SDF field for eyelid reconstruction and animation. In order to acquire high-quality results with such limited inputs, we leverage the eyeball information (geometry and motions) to provide additional guidance. Specifically, we obtain the masks of the eyelid and iris by the commercial service of SenseTime<sup>1</sup> and design an eyeball calibration method to get the eyeball parameters we need. Note that the calibration suits the same capture setup and does not require additional inputs (Sec. 3.1). Then, we use the eyeball parameters to assist the learning of dynamic neural SDF field (Sec. 3.2) and achieve the semantic animation control of eyelids (Sec. 3.3). The optimization details can be found in Sec. 3.4. Note that our method can process both eyes of the human face, but to simplify the explanation, most descriptions are in the context of one single eye.

#### 3.1 Eyeball Calibration

The goal of the eyeball calibration is to get the eyeball positions and rotations for assisting the eyelid reconstruction. As human eyeballs have strong prior in both shapes and dimensions [Kaufman et al. 2003; Von Helmholtz 1925], we leverage these physiological priors to simplify the calibration problem. The key idea of our method is that only the eyeball with correct position can depict the appearance of all gazes. Therefore, given a 3D eyeball template  $\mathcal{T}$  with physiologically appropriate ratio of the iris and eyeball radius, we optimize one eyeball position, one uniform scale, and n-frame eyeball rotations by aligning the rendering results with the appearance of each frame. Our eyeball template is shown in Fig. 2. It contains a cut sphere and a convex cornea that satisfies the mathematical modeling in [Nishino and Nayar 2004]. As the iris is the noticeable feature of human eyes, we take the iris mask as the target of alignment, which can be described as

$$E(\mathbf{P}^e, \mathbf{R}_i^e, s; \mathbf{P}_i^c, \mathcal{T}, \mathcal{M}) = \sum_i^n \|\mathcal{M}_i - \hat{\mathcal{M}}_i\|_2^2 \quad (1)$$

$$\hat{\mathcal{M}}_i = \Pi(\mathcal{T}, s, \mathbf{P}^e, \mathbf{P}_i^c, \mathbf{R}_i^e) \quad (2)$$

where  $\mathbf{P}^e \in \mathbb{R}^3$  is the eyeball position, and  $s \in \mathbb{R}$  is the scale of eyeball template. They are shared by all frames.  $n$  is the number of frames.  $\mathbf{P}_i^c \in SE(3)$  is the camera pose of the i-th frame.  $\mathbf{R}_i^e =$

<sup>1</sup>[www.sensetime.com](http://www.sensetime.com)

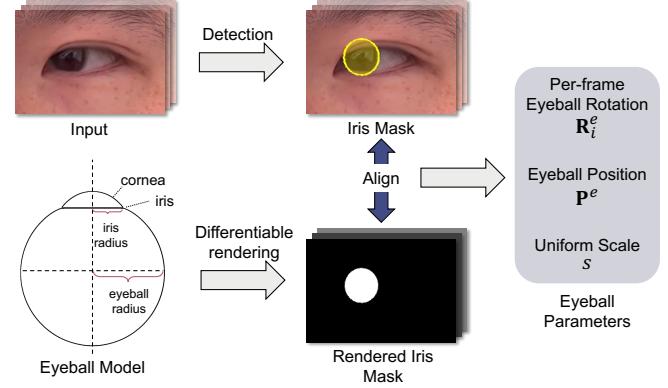


Fig. 2. The eyeball calibration is based on an 3D eyeball model with physiological prior. We apply differentiable rendering to optimize eyeball parameters by aligning iris masks.

$[p_i, y_i] \in \mathbb{R}^2$  (represented as pitch and yaw in the model space) is the eyeball rotation of the  $i$ -th frame.  $\mathcal{M}$  is the ground-truth iris mask.  $\hat{\mathcal{M}}$  is the rendered iris mask of the 3D eyeball template, which is got by the differentiable mesh rendering  $\Pi(\cdot)$  [Laine et al. 2020]. Note that we solve the parameters of the two eyes separately.

The above optimization relies on good initial values. In order to avoid the local minimal, we have some strategies to initialize the parameters. Details can be found in the supplementary materials.

#### 3.2 Dynamic Neural SDF Field for Eyelid

We model the moving eyelids (together with the other facial regions) as a dynamic neural SDF field. Similar to the previous works [Park et al. 2021a,b], we construct the deformation between each observation space and the canonical space. Specifically, the dynamic modeling is achieved by a canonical hyper-space [Park et al. 2021b] cooperating with the invertible deformation field [Cai et al. 2022]. Further, we leverage the eyeball information to provide extra constraints on geometry and a gaze-varying encoding for subtle details.

**3.2.1 Preliminaries: NeuS for Dynamic Scenes.** NeuS [Wang et al. 2021] is a neural implicit surface reconstruction method that can reconstruct the surface of a static object by inputting posed images. It represents the geometry as the zero-level set of signed distance function (SDF)  $S = \{\mathbf{x} \in \mathbb{R}^3 | f(\mathbf{x}) = 0\}$ . For a particular 3D point  $\mathbf{p}$  in the space, its SDF value and geometric feature  $\mathbf{z}$  are predicted by an MLP  $F_s(\mathbf{p})$ . To train this MLP, they propose a probability function to convert the SDF values of  $N$  points sampled along a ray  $r = \{\mathbf{p}_k = \mathbf{o} + t_k \mathbf{v} | k = 1, \dots, N, t_k < t_{k+1}\}$  to density value for volume rendering:

$$\hat{C} = \sum_k^N T_k \alpha_k \mathbf{c}_k, \quad T_k = \prod_{j=1}^{k-1} (1 - \alpha_j) \quad (3)$$

Here,  $\mathbf{c}_k$  is the color of the  $k$ -th points, which is predicted by another trainable MLP  $F_c(\mathbf{p}_k, \mathbf{n}_k, \mathbf{v}, \mathbf{z}_k)$  from normal  $\mathbf{n}_k$ , view direction  $\mathbf{v}$ , and geometric feature  $\mathbf{z}_k$ . The discrete opacity values  $\alpha_k$  is calculated

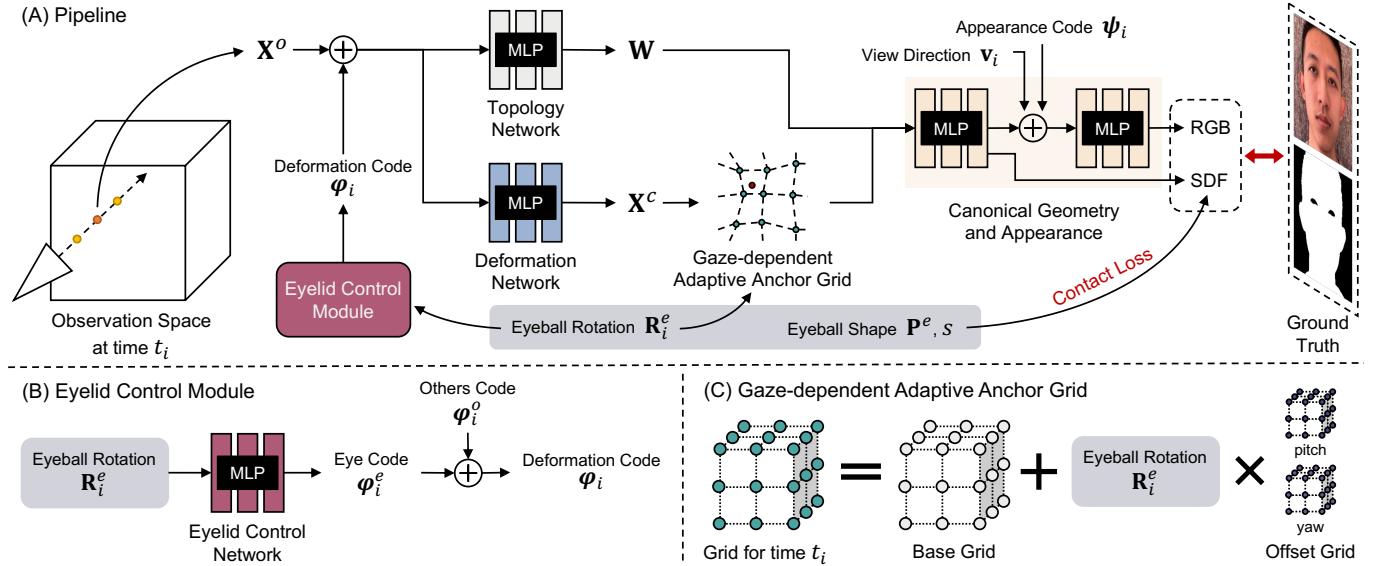


Fig. 3. An overview of our method. (A) Our method models the moving eyelids as a dynamic neural SDF field, which is achieved by a canonical hyper-space with deformation. For a sampled 3D point in the observation space, the topology and deformation network convert it to the canonical hyper-space, where the SDF field is defined. Then, its point color and SDF predicted by MLPs are used to apply volume rendering for training on RGB images. (B) This module divides the dynamic information into eye motions and others. The eye motions are modeled by the latent code mapped from eyeball rotations, and the other movements are modeled by a learnable latent code. (C) We encode the geometry feature of a 3D canonical point by the positions of neighbor anchors. For each frame, the anchor positions are determined by a learnable base grid plus the linear combination of two learnable offset grids based on eyeball rotations.

by

$$\alpha_k = \max\left(\frac{\Phi_s(f(\mathbf{p}(t_k))) - \Phi_s(f(\mathbf{p}(t_{k+1})))}{\Phi_s(f(\mathbf{p}(t_k))), 0}\right) \quad (4)$$

The  $\Phi_s(x)$  is defined as  $\Phi_s(x) = (1 + e^{-sx})^{-1}$ . The  $s$  is a trainable parameter. The training is based on the photometric loss that minimizes the difference between the rendered images and input images.

To adapt NeuS [Wang et al. 2021] to dynamic scenes, NDR [Cai et al. 2022] construct the deformation between observation and canonical space like dynamic NeRF methods [Park et al. 2021a,b]. Further, they employ a strictly invertible bijective mapping based on the continuous homeomorphic mapping  $\mathcal{D}_i : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ . It maps the observation point  $\mathbf{X}^o = [x_i, y_i, z_i]$  at time  $t_i$  back to the 3D canonical position  $\mathbf{X}^c = [x, y, z]$ . As  $\mathcal{D}_i$  is invertible, the deformed surface at time  $t_i$  can be obtained by

$$U_i = \{\mathcal{D}_i^{-1}([x, y, z]) | \forall [x, y, z] \in U\} \quad (5)$$

and the correspondence between deformed points can be factorized as

$$[x_j, y_j, z_j] = \mathcal{D}_j^{-1} \circ \mathcal{D}_i([x_i, y_i, z_i]) \quad (6)$$

However, the invertible representation only effectively models the topology-preserving motions. To better model the changes in topology, Cai et al. [2022] introduce the topology-aware design [Park et al. 2021b] into their deformation field. Specifically, the 3D position  $\mathbf{X}^o$  observed at time  $t_i$  is mapped to topology coordinates by the mapping  $\mathcal{W} : \mathbb{R}^3 \rightarrow \mathbb{R}^m$ . Cooperating with the 3D canonical position  $\mathbf{X}^c$ , the canonical hyper-space is represented as

$$\mathbf{X} = [\mathbf{X}^c, \mathbf{W}] \quad (7)$$

where  $\mathbf{W} \in \mathbb{R}^m$  is the topology coordinates. In practice, the invertible deformation mapping and the topology mapping are modeled by two separate MLPs:

$$\mathbf{X}^c = F_d(\mathbf{X}^o, \varphi_i), \quad \mathbf{W} = F_w(\mathbf{X}^o, \varphi_i) \quad (8)$$

where  $\varphi_i$  is the trainable deformation code at time  $t_i$ . Then, the neural implicit surface is defined as the zero-level set of an SDF on the canonical hyper-space:

$$S = \{\mathbf{x} \in \mathbb{R}^{3+m} | f(\mathbf{x}) = 0\} \quad (9)$$

Similar to NeuS [Wang et al. 2021], the SDF value is converted to density for training based on volume rendering. The point color at time  $t_i$  is predicted from the 3D canonical position  $\mathbf{X}^c$ , the normal  $\mathbf{n}^c$ , the view direction  $\mathbf{v}^c$ , and the geometric feature  $\mathbf{z}^c$  conditioned a time-varying appearance code  $\psi_i$ , formulated as

$$\mathbf{c}_i = F_c(\mathbf{X}^c, \mathbf{n}^c, \mathbf{v}^c, \mathbf{z}^c, \psi_i) \quad (10)$$

**3.2.2 Gaze-dependent Adaptive Anchor Grid.** Despite the impressive results of NDR [Cai et al. 2022] on large-scale geometry, it still struggles to recover fine-grain geometry of dynamic scenes. To reconstruct subtle details, some solutions [Cai et al. 2023; Fu et al. 2022; Li et al. 2023; Rosu and Behnke 2023] are proposed for static reconstruction, but few works solve it in dynamic scenes. One naive solution is to directly apply the static techniques on the 3D canonical space, but this scheme inevitably misses some time-varying information and generates low-quality results (Sec. 4.2).

Instead, we develop a gaze-dependent adaptive anchor grid to improve subtle details of dynamic eyelid reconstruction. Following NeuDA [Cai et al. 2023], we attribute the shortage of details to the

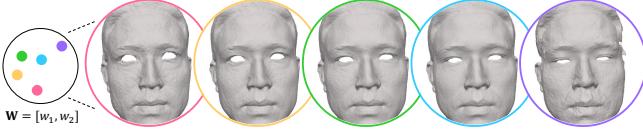


Fig. 4. Topology coordinates model various shape templates of the eyelids.

poor utilization of the spatial context. For example, NeuS [Wang et al. 2021], as a method purely based on MLPs, implicitly utilizes the spatial context through the continuity of MLPs, which is an indirect way and often leads to over-smooth results. Voxel grid approaches [Fridovich-Keil et al. 2022; Liu et al. 2020; Sun et al. 2022; Takikawa et al. 2021] forward a step by explicitly encoding the geometry feature of a particular 3D point using the surrounding eight vertices, followed by the hierarchical structure [Müller et al. 2022; Takikawa et al. 2021; Wang et al. 2023] for different levels of representation. However, as they uniformly split the 3D space, the distribution of vertices may not be optimal for a particular object because the distribution of object details is spatially variant, which means we need model the geometry features adaptively (e.g., dense for subtle details, sparse for large-scale geometry). This issue may also occur when making local modeling based on facial landmarks [Chen et al. 2023], which are manually defined and not fully adaptive. NeuDA [Cai et al. 2023] solve this problem by introducing a multi-level learnable anchor grid. For a specific sample point  $\mathbf{p} \in \mathbb{R}^3$ , the geometry feature of level  $l$  is represented as the interpolated frequency embedding of the 8-nearest neighbor vertices (anchors):

$$G^l(\mathbf{p}, \mathbf{A}) = \sum_u w_u^l \cdot \gamma(\mathbf{a}_u^l), \quad u = 1, \dots, 8 \quad (11)$$

$$\gamma(\mathbf{a}^l) = (\sin(2^l \pi \mathbf{a}^l), \cos(2^l \pi \mathbf{a}^l)) \quad (12)$$

where  $\mathbf{A} \in \mathbb{R}^{N_a \times 3}$  denotes the anchor grid.  $\mathbf{a}_u^l \in \mathbb{R}^3$  is the learnable anchors of level  $l$  around  $\mathbf{p}$ , and  $w_u^l$  is the trilinear interpolation weight.  $\gamma(\cdot)$  is the frequency embedding proposed by [Mildenhall et al. 2021]. Finally, the geometry encoding is represented as the concatenation of per-level frequency embeddings:

$$F_g(\mathbf{p}, \mathbf{A}) = [\mathbf{p}, G^1(\mathbf{p}, \mathbf{A}), G^2(\mathbf{p}, \mathbf{A}), \dots, G^l(\mathbf{p}, \mathbf{A})] \quad (13)$$

Different from NeuDA [Cai et al. 2023] that learns a fixed anchor grid, we learn a gaze-dependent anchor grid by modeling the relationship between eyeball rotations and anchor positions. As we represent the deformation field by the hyper-space (3D canonical space plus the topology coordinates), it can be taken as a set of 3D canonical templates defined by the topology coordinates. Similar to HyperNeRF [Park et al. 2021b], if we visualize the hyper-space by sampling different topology coordinates (Fig. 4), we can find the hyper-space models a family of shapes. This motivates us to learn time-varying anchor grids to represent the geometry of the moving eyelids. A direct way to achieve this is to relate the anchor grid with the topology coordinates. However, this operation will make a explosive growth in the computation graph, which brings a great burden to the device capacity. Considering the topology changes are highly relevant with the gaze poses, we associate the anchor grid with the eyeball rotations instead. In practice, we model the anchor grid as the linear combination of a base anchor grid  $\mathbf{A}_0$  and

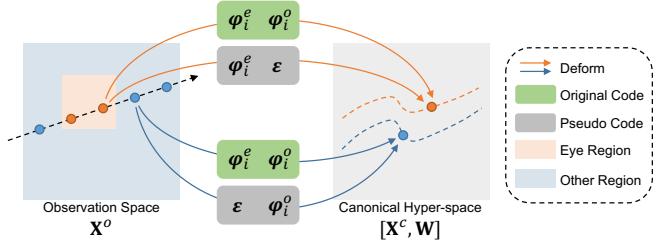


Fig. 5. Illustration of disentanglement strategy. We generate pseudo deformation codes by replacing one of the parts of the original deformation code with a random latent code  $\epsilon$ . The control region of the unchanged part is encouraged to have the same hyper-space coordinates.

two offset grid  $\mathbf{A}_p, \mathbf{A}_y$  that stores anchor offset related to the pitch and yaw angle, respectively. The final anchor grid of the  $i$ -th frame can be formulated as

$$\mathbf{A}_i = \mathbf{A}_0 + p_i \cdot \mathbf{A}_p + y_i \cdot \mathbf{A}_y \quad (14)$$

### 3.3 Eyelid Control

In previous methods [Cai et al. 2022; Park et al. 2021b], the per-frame deformation code  $\phi_i$  is a freely learnable latent code that cannot be controlled semantically. In order to better animate the reconstructed eyelids, we develop an eyelid control module to model the eyelid deformation as a function of the eyeball rotation.

A straightforward solution is to predict the deformation code  $\phi_i$  from the eyeball rotation  $\mathbf{R}_i^e$  using an MLP, formulated as  $\phi_i = F_e(\mathbf{R}_i^e)$ . However, this design assumes that the dynamic scenes only contain eye-related movements, which is almost unattainable in real-world applications because the input video inevitably contains some unconscious movements of the other facial regions.

To solve this problem, we divide the deformation into two parts (eye movements and others), and simultaneously learn a latent code  $\phi_i^o$  to model the movements of the other facial regions. Therefore, the final deformation code is represented as

$$\phi_i = [\phi_i^e, \phi_i^o] = [F_e(\mathbf{R}_i^e), \phi_i^o] \quad (15)$$

To ensure each part has a proper control region, we introduce a disentanglement strategy as shown in Fig. 5. During training, we will generate several pseudo deformation codes by replacing one of the parts with a random latent code  $\epsilon$ , and encourage the hyper-space coordinates in the control region of the unchanged parts keep the same as before. Taking eye-region points (orange in Fig. 5) as an example, although it is mapped to hyper-space by two different codes (the original code and the pseudo code with the second part changed), the two results should be the same. This can prevent the second part from controlling eye-region movements. The control region is split by the eye bounding box in the observation space, which can be automatically generated based on the eyeball parameters acquired in Sec. 3.1. Note that this strategy can be extended from {eye, others} to {left eye, right eye, others} by further splitting  $\phi_i^e$  into two sub-codes for the left/right eye, respectively. It can also be combined with conditions to model closed eyes for generating natural eyelid movements like blinking. Details can be found in the supplementary materials.

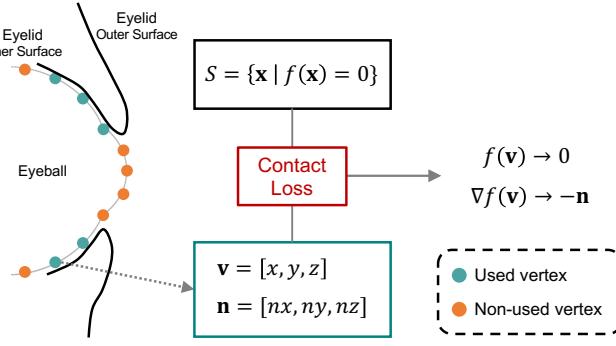


Fig. 6. The contact loss encourages the inner surface of eyelid tightly adhere to the eyeball surface, which provides additional geometric constraints. The used and non-used vertices are chosen by excluding all back vertices and visible frontal vertices.

### 3.4 Optimization Details

Given an RGB video with the mask of the actor and the eyeball parameters  $\{\mathbf{R}_i^e, \mathbf{P}^e, s\}$  acquired by eyeball calibration (Sec. 3.1), we optimize the parameters of MLPs  $\{F_s, F_c, F_d, F_w, F_e\}$ , the anchor grids  $\{\mathbf{A}_0, \mathbf{A}_p, \mathbf{A}_y\}$ , and latent codes  $\{\varphi_i^o, \psi_i\}$ . We assume the batch size is  $R$ , and we sample  $N$  points for each ray.

We minimize the difference between the rendered and ground-truth pixel color by

$$\mathcal{L}_{color} = \frac{1}{R} \sum_r \|C(r) - \hat{C}(r)\|_1 \quad (16)$$

where  $r$  is a specific ray in the volume rendering formulation [Mildenhall et al. 2021].

To focus on the actor, we also optimize mask loss defined as

$$\mathcal{L}_{mask} = BCE(M(r), \sum_k T_{r,k} \alpha_{r,k}) \quad (17)$$

where  $BCE(\cdot)$  denotes the binary cross entropy loss, and  $M(r)$  is the mask label of ray  $r$ .

An Eikonal loss [Gropp et al. 2020] is introduced to regularize the signed distance function by

$$\mathcal{L}_{reg} = \frac{1}{RN} \sum_{r,k} (\|\nabla f(\mathbf{p}_{r,k})\|_2 - 1)^2 \quad (18)$$

Inspired by [Verbin et al. 2022], we optimize a normal regularization loss by auxiliarily predicting a normal vector  $\hat{\mathbf{n}}_{r,k}$  using the MLP  $F_s$ . The predicted normal is encouraged to be close to the gradient of SDF:

$$\mathcal{L}_{norm} = \sum_{r,k} T_{r,k} \alpha_{r,k} \|\nabla f(\mathbf{p}_{r,k}) - \hat{\mathbf{n}}_{r,k}\|_1 \quad (19)$$

In addition to the above terms, we also leverage the eyeball information to provide geometry constraints for the eyelids. As shown in Fig. 6, our method models both outer and inner surfaces of eyelids. We encourage the inner surface of eyelid to tightly adhere to the eyeball surface. In practice, we use the positions of eyeball vertices and encourage the SDF field of eyelid to have zero SDF value in

these positions while holding opposite normal directions:

$$\mathcal{L}_{contact} = \frac{1}{V} \sum_i^V \|f(\mathbf{v}_i)\|_1 + \frac{1}{V} \sum_i^V \|\nabla f(\mathbf{v}_i) \cdot \mathbf{n}_i + 1\|_1 \quad (20)$$

where  $\mathbf{v}_i$  is the sampled position on the eyeball surface, and  $\mathbf{n}_i$  is the corresponding eyeball normal.  $V$  is the number of samples. Note that all back and visible frontal vertices are excluded because they do not contact the eyelids. The former is determined by normal directions in the eyeball model space, while the latter is frame-specific and determined by whether their projected 2D positions fall within the eyelid mask.

To ensure the disentanglement strategy described in Sec. 3.3, we introduce disentangle loss defined as

$$\begin{aligned} \mathcal{L}_{det} = & \frac{1}{RN} \sum_{\mathbf{X}^o \in B^e} \|H(\mathbf{X}^o, [\varphi_i^e, \epsilon]) - H(\mathbf{X}^o, [\varphi_i^e, \varphi_i^o])\|_1 \\ & + \frac{1}{RN} \sum_{\mathbf{X}^o \in B^o} \|H(\mathbf{X}^o, [\epsilon, \varphi_i^o]) - H(\mathbf{X}^o, [\varphi_i^e, \varphi_i^o])\|_1 \end{aligned} \quad (21)$$

where  $H(\mathbf{X}^o, \varphi_i) = [F_d(\mathbf{X}^o, \varphi_i), F_w(\mathbf{X}^o, \varphi_i)]$  and  $\epsilon$  is a random latent code sampled by standard normal distribution.  $B^e$  and  $B^o$  are the sets of point in the eye region and the other region, respectively.

Overall, the full objective is formulated as

$$\begin{aligned} \mathcal{L} = & \mathcal{L}_{color} + \lambda_{mask} \mathcal{L}_{mask} + \lambda_{reg} \mathcal{L}_{reg} + \lambda_{norm} \mathcal{L}_{norm} \\ & + \lambda_{contact} \mathcal{L}_{contact} + \lambda_{det} \mathcal{L}_{det} \end{aligned} \quad (22)$$

where  $\lambda$  is the weights for different terms. In our experiments, we set  $\lambda_{mask}, \lambda_{reg}, \lambda_{norm}, \lambda_{contact}$ , and  $\lambda_{det}$  as 0.1, 0.1,  $10^{-5}$ , 0.01, and 0.1, respectively.

## 4 EXPERIMENTS

In this section, we first present the datasets and our implementation details. Then, we compare our method with the state-of-the-art methods qualitatively and quantitatively. Finally, we evaluate our key contributions via ablation study. We refer readers to our video for input examples and dynamic results. As our contributions focus on geometry reconstruction, most experiments only show geometry results. Appearance results and comparisons can be found in our supplementary materials.

**Datasets.** We use a mobile phone (iPhone 13) to collect the RGB videos. In total, we recruit six participants of different ethnicities and genders. The participants are asked to perform different gaze directions while keeping the other facial parts as static as possible. The resolution of the collected data is 1080p, and the whole capture duration is about one minute. We down-sample the video in the time axis and get RGB sequences with around 500 frames. We then use Agisoft MetaShape [Agisoft 2023] to get the camera intrinsics and extrinsics. In order to evaluate the geometry results quantitatively, we additionally make four synthetic videos based on MetaHuman [Games 2023]. The videos contains 450 frames with the similar content as the real data.

**Implementation Details.** The experiments are implemented with Pytorch. Similar to [Atzmon and Lipman 2020], we initialize  $f(x)$  as a unit sphere. We optimize the neural networks by Adam optimizer [Kingma and Ba 2014] with a learning rate of  $5 \times 10^{-4}$ . We sample 1,024 rays per batch, and the number of per-ray sampled points

grows from 64 to 128 by 4-times up-sampling like NeuS [Wang et al. 2021]. The level of our adaptive anchor grid is set to 8. The dimensions of  $\varphi_i^e$  is 32 (16 for left and 16 for right). We learn separate  $F_e$  for the left and right eye with one shared latent code  $\varphi_i^o$  of 32 dimensions. For coarse-to-fine training, we adopt an incremental positional encoding strategy like Nerfies [Park et al. 2021a]. The experiments run  $1.2 \times 10^5$  iterations for about 12 hours on one NVIDIA RTX 3090.

Table 1. Quantitative comparisons on synthetic data.

Methods	Depth Error ↓				Chamfer Distance ↓			
	ID-1	ID-2	ID-3	ID-4	ID-1	ID-2	ID-3	ID-4
NDR	1.76	1.39	1.90	1.29	0.131	0.219	0.241	0.187
Tensor4D	1.29	1.11	1.22	1.39	0.204	0.226	0.221	0.201
PointAvatar	3.49	3.58	3.42	3.54	2.450	2.139	1.477	1.409
FLARE	1.79	1.81	1.26	1.48	0.584	0.527	0.261	0.382
Wen et al.	2.43	2.87	1.77	2.14	1.029	1.406	0.540	0.849
Ours	<b>0.79</b>	<b>0.48</b>	<b>0.60</b>	<b>0.49</b>	<b>0.044</b>	<b>0.038</b>	<b>0.062</b>	<b>0.034</b>

#### 4.1 Comparisons

As we model the moving eyelid as a dynamic neural SDF field, we compare our method with the other dynamic neural SDF methods [Cai et al. 2022; Shao et al. 2023]. In addition, we make comparisons with the recent face-related methods based on lightweight capture [Bharadwaj et al. 2023; Zheng et al. 2023], as well as the model-based eyelid tracking method [Wen et al. 2017].

The qualitative results of real datasets are shown in Fig. 9. Most results are rendered with a directional light by rasterization. As PointAvatar [Zheng et al. 2023] is a point-based method, we use its normal outputs to composite the shading results under the same directional light. It shows that PointAvatar [Zheng et al. 2023] can only reconstruct the coarse contours of facial details. One possible reason is that the number of input images is one or two thousand in the original settings of PointAvatar [Zheng et al. 2023], but our input only contains hundreds of images. It may not perform well with such limited inputs. Wen et al. [2017] can generate subtle details like double-folds, but as they are model-based method, the shape of eye region is still constrained by the model capacity. FLARE [Bharadwaj et al. 2023] allows the 3D model to deform and can generate more realistic shapes, but the results suffer serious self-intersection. The results of Tensor4D [Shao et al. 2023] is bumpy, while the results of NDR [Cai et al. 2022] is over-smooth. Our approach can not only reconstruct more plausible geometry but also preserve more subtle details in the eye region.

For quantitative comparisons, we use the synthetic data to calculate the geometric metrics (depth error and Chamfer distance). Specifically, for neural implicit methods (NDR [Cai et al. 2022], Tensor4D [Shao et al. 2023], and ours), we export the mesh by marching cubes [Lorensen and Cline 1998] with a resolution of 512. Then, similar to the mesh-based methods [Bharadwaj et al. 2023; Wen et al. 2017], the mesh is rasterized to depth map and converted to point cloud for calculating the metrics. For PointAvatar [Zheng et al. 2023], we directly use the points to calculate Chamfer distance and project the points to get a sparse depth map for calculating depth error. Note that we only calculate the metrics in the eye region, which is located by an extended bounding box of the eye mask. The

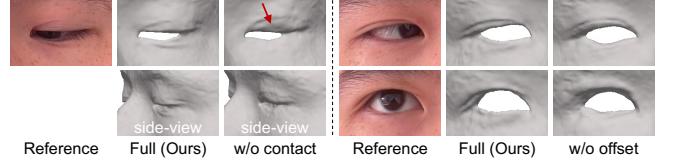


Fig. 7. Ablation for reconstruction. Left: removing contact loss results in sunken shape on the upper eyelids. Right: the details becomes worse when removing the offset grids.

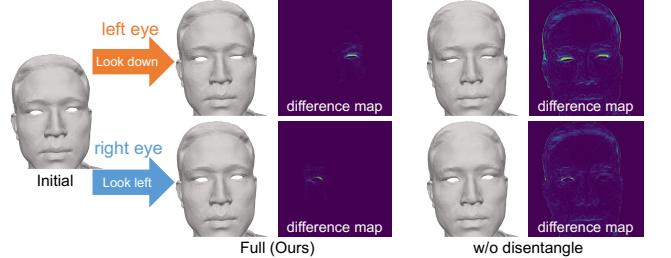


Fig. 8. Ablation for animation. Without disentangle loss, the eyeball rotation cannot control the correct region of eyelids.

Table 2. Ablation study on synthetic data.

Methods	Depth Error ↓				Chamfer Distance ↓			
	ID-1	ID-2	ID-3	ID-4	ID-1	ID-2	ID-3	ID-4
Full (Ours)	<b>0.79</b>	<b>0.48</b>	<b>0.60</b>	<b>0.49</b>	<b>0.044</b>	<b>0.038</b>	<b>0.062</b>	<b>0.034</b>
w/o contact	0.88	0.53	0.66	0.72	0.098	0.054	0.080	0.085
w/o offset	0.95	0.62	0.69	0.79	0.087	0.048	0.071	0.041
w/o both	1.00	0.70	0.90	0.85	0.122	0.106	0.113	0.106

average metrics of the sequence are reported in Tab. 1. Our method has a noticeable improvement compared with the previous methods.

#### 4.2 Ablation Study

In order to evaluate our technical contributions, we design experiments to demonstrate the effectiveness of our proposed techniques for both reconstruction (contact loss and gaze-dependent adaptive anchor grid) and animation (disentanglement strategy).

For reconstruction, the first ablation (w/o contact) removes the contact loss. Results in Fig. 7 show that without the contact loss, the algorithm fails to solve the depth uncertainty with such limited inputs, which leads to the sunken shape on the upper eyelid. The second ablation (w/o offset) removes the offset grid and only optimizes a fixed anchor grid. As shown in Fig. 7, it is hard for a fixed anchor grid to represent the gaze-varying shapes, which leads to the lost of the subtle details as well as the misalignment of the eyelid contour. The quantitative results in Tab. 2 also demonstrate the advantages of the proposed method. Note that we also add a third ablation (w/o both) that removes both the contact loss and the offset grid, which can be taken as a naive combination of NDR [Cai et al. 2022] and NeuDA [Cai et al. 2023].

For animation, we compare the proposed method with the one without disentangle loss. As shown in Fig. 8, when animating the reconstructed result by changing parameters of eyeball rotations,

the one without disentangle loss cannot control the eye region correctly. The difference maps clearly show that its results contain some unexpected motions like pursing lips and moving head.

## 5 DISCUSSION

Our approach is not perfect and may miss some details like shallow creases which introduce serious ambiguity between geometry and appearance (networks can use smooth shapes with dark colors to satisfy the photometric loss). Similar issues also occur when reconstructing people with dark skins or in dark scenes. It is also hard for our method to reconstruct high-quality eyelashes, eyebrows, and bangs, which is challenging even for the high-cost method [Bermano et al. 2015] and needs further exploration. The training and testing time of our method is still too long, which needs to be improved by some speed-up techniques [Müller et al. 2022; Wang et al. 2023] and extended by neural parameterization methods [Lin et al. 2023; Ma et al. 2022] for registration if necessary. Our method only considers the influence of eyeballs towards eyelids but neglects the reverse influence of eyelids towards eyeballs. It may be beneficial to optimize the eyelid and eyeball parameters jointly in the future.

## 6 CONCLUSION

In this paper, we propose a novel method that can make high-quality eyelid reconstruction and animation by only using lightweight captures. To achieve this, we leverage eyeball information to compensate for the limited inputs from both perspectives of eyeball shapes and motions. For animation, we model the eyelid deformation as a function of the eyeball rotation and introduce a disentanglement strategy to ensure the correct semantic control. Besides, we present a lightweight eyeball calibration method to acquire the eyeball parameters automatically, which does not need additional inputs. To our best knowledge, we present the first method for high-quality eyelid reconstruction and animation from lightweight inputs. We believe this technique could reduce the cost and broaden the application scenarios of realistic digital humans.

## ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China (2023YFC3305600), the NSFC (No. 61822111, 62021002), the Zhejiang Provincial Natural Science Foundation (No. LDT23F02024F 02), and the Key Research and Development Project of Tibet Autonomous Region (XZ202101ZY0019G). This work was also supported by THUIBCS, Tsinghua University, and BLBCI, Beijing Municipal Education Commission. Feng Xu is the corresponding author.

## REFERENCES

- Agisoft. 2023. *Agisoft Metashape Professional*.
- Matan Atzmon and Yaron Lipman. 2020. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2565–2574.
- Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. 2009. Patch-Match: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* 28, 3 (2009), 24.
- Pascal Bérard, Derek Bradley, Markus Gross, and Thabo Beeler. 2016. Lightweight eye capture using a parametric model. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.
- Pascal Bérard, Derek Bradley, Markus Gross, and Thabo Beeler. 2019. Practical Person-Specific Eye Rigging. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 441–454.
- Pascal Bérard, Derek Bradley, Maurizio Nitti, Thabo Beeler, and Markus H Gross. 2014. High-quality capture of eyes. *ACM Trans. Graph.* 33, 6 (2014), 223–1.
- Amit Bermano, Thabo Beeler, Yeara Kozlov, Derek Bradley, Bernd Bickel, and Markus Gross. 2015. Detailed spatio-temporal reconstruction of eyelids. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–11.
- Shrishti Bharadwaj, Yufeng Zheng, Otmar Hilliges, Michael J Black, and Victoria Fernández Abrevaya. 2023. FLARE: Fast Learning of Animatable and Relightable Mesh Avatars. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–15.
- V Blanz and T Vetter. 1999. A Morphable Model for the Synthesis of 3D Faces. In *26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 1999)*. ACM Press, 187–194.
- Adrian Broadhurst, Tom W Drummond, and Roberto Cipolla. 2001. A probabilistic framework for space carving. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, Vol. 1. IEEE, 388–393.
- Bowen Cai, Jinchu Huang, Rongfei Jia, Chengfei Lv, and Huan Fu. 2023. NeuDA: Neural Deformable Anchor for High-Fidelity Implicit Surface Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8476–8485.
- Hongrui Cai, Wanquan Feng, Xuetao Feng, Yan Wang, and Juyong Zhang. 2022. Neural surface reconstruction of dynamic scenes with monocular rgbd camera. *Advances in Neural Information Processing Systems* 35 (2022), 967–981.
- Chen Cao, Tomas Simon, Jin Kyu Kim, Gabe Schwartz, Michael Zollhoefer, Shun-Suke Saito, Stephen Lombardi, Shih-En Wei, Danielle Belko, Shouo-I Yu, Yaser Sheikh, and Jason Saragih. 2022. Authentic volumetric avatars from a phone scan. *ACM Trans. Graph.* 41, 4, Article 163 (jul 2022), 19 pages. <https://doi.org/10.1145/3528223.3530143>
- Chuhan Chen, Matthew O’Toole, Gaurav Bharaj, and Pablo Garrido. 2023. Implicit neural head synthesis via controllable local deformation fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 416–426.
- Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. 2022. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia 2022 Conference Papers*. 1–9.
- Sara Fridovich-Keil, Alex Yu, Matthew Tancil, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. 2022. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5501–5510.
- Qiancheng Fu, Qingshan Xu, Yew Soon Ong, and Wenbing Tao. 2022. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *Advances in Neural Information Processing Systems* 35 (2022), 3403–3416.
- Yasutaka Furukawa and Jean Ponce. 2009. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence* 32, 8 (2009), 1362–1376.
- Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. 2021. Dynamic Neural Radiance Fields for Monocular 4D Facial Avatar Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8649–8658.
- Epic Games. 2023. *Metahuman creator*.
- Xuan Gao, Chenglai Zhong, Jun Xiang, Yang Hong, Yudong Guo, and Juyong Zhang. 2022. Reconstructing personalized semantic facial nerf models from monocular video. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–12.
- Pablo Garrido, Michael Zollhöfer, Dan Casas, Levi Valgaerts, Kiran Varanasi, Patrick Pérez, and Christian Theobalt. 2016. Reconstruction of personalized 3D face rigs from monocular video. *ACM Transactions on Graphics (TOG)* 35, 3 (2016), 1–15.
- Philip-William Grassal, Malte Prinzel, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. 2022. Neural head avatars from monocular rgb videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18653–18664.
- Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. 2020. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099* (2020).
- Antoine Guédon and Vincent Lepetit. 2024. SuGaR: Surface-Aligned Gaussian Splatting for Efficient 3D Mesh Reconstruction and High-Quality Mesh Rendering. *CVPR* (2024).
- Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2024. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery. <https://doi.org/10.1145/3641519.3657428>
- P.L. Kaufman, A. Alm, and F.H. Adler. 2003. *Adler’s Physiology of the Eye: Clinical Application*. Mosby. <https://books.google.com.hk/books?id=2YlsAAAAMAAJ>
- Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. 2006. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, Vol. 7. 0.
- Glenn Kerbiriou, Quentin Avril, Fabien Danieau, and Maud Marchal. 2022. Detailed Eye Region Capture and Animation. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 279–282.
- Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics* 42, 4 (July 2023).
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

- Tobias Kirschstein, Shengan Qian, Simon Giebenhain, Tim Walter, and Matthias Nießner. 2023. NeRSeMble: Multi-View Radiance Field Reconstruction of Human Heads. *ACM Trans. Graph.* 42, 4, Article 161 (jul 2023), 14 pages. <https://doi.org/10.1145/3592455>
- Samuli Laine, Janne Hellsten, Tero Karra, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. 2020. Modular Primitives for High-Performance Differentiable Rendering. *ACM Transactions on Graphics* 39, 6 (2020).
- Gengyan Li, Abhimitra Meka, Franziska Mueller, Marcel C Buehler, Otmar Hilliges, and Thabo Beeler. 2022. EyeNeRF: a hybrid representation for photorealistic synthesis, animation and relighting of human eyes. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–16.
- Gengyan Li, Kripasindhu Sarkar, Abhimitra Meka, Marcel Buehler, Franziska Mueller, Paulo Gotardo, Otmar Hilliges, and Thabo Beeler. 2024. ShellNeRF: Learning a Controllable High-resolution Model of the Eye and Periorbital Region. *Computer Graphics Forum* (2024). <https://doi.org/10.1111/cgf.15041>
- Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unterath, Ming-Yu Liu, and Chen-Hsuan Lin. 2023. Neuralangelo: High-Fidelity Neural Surface Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8456–8465.
- Connor Lin, Koki Nagano, Jan Kautz, Eric Chan, Umar Iqbal, Leonidas Guibas, Gordon Wetzstein, and Sameh Khamis. 2023. Single-shot implicit morphable faces with consistent texture parameterization. In *ACM SIGGRAPH 2023 Conference Proceedings*. 1–12.
- Shanchuan Lin, Linjie Yang, Imran Saleemi, and Soumyadip Sengupta. 2022. Robust high-resolution video matting with temporal guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 238–247.
- Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. 2020. Neural sparse voxel fields. *Advances in Neural Information Processing Systems* 33 (2020), 15651–15663.
- William E Lorensen and Harvey E Cline. 1998. Marching cubes: A high resolution 3D surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*. 347–353.
- Conny Lu, Praneeth Chakravarthula, Yujie Tao, Steven Chen, and Henry Fuchs. 2020. Improved vergence and accommodation via purkinje image tracking with multiple cameras for ar glasses. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 320–331.
- Li Ma, Xiaoyu Li, Jing Liao, Xuan Wang, Qi Zhang, Jue Wang, and Pedro V Sander. 2022. Neural parameterization for dynamic human head editing. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–15.
- Shengjie Ma, Yanlin Weng, Tianjia Shao, and Kun Zhou. 2024. 3D Gaussian Blendshapes for Head Avatar Animation. In *ACM SIGGRAPH 2024 Conference Papers*. 1–10.
- Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.
- Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)* 41, 4 (2022), 1–15.
- Debangha R Neog, João L Cardoso, Anurag Ranjan, and Dinesh K Pai. 2016. Interactive gaze driven animation of the eye region. In *Proceedings of the 21st International Conference on Web3D Technology*. 51–59.
- Ko Nishino and Shree K Nayar. 2004. The world in an eye [eye image interpretation]. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, Vol. 1. IEEE, 1–1.
- Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. Deep sdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 165–174.
- Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. 2021a. Nerfes: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5865–5874.
- Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. 2021b. HyperNeRF: A Higher-Dimensional Representation for Topologically Varying Neural Radiance Fields. *ACM Trans. Graph.* 40, 6, Article 238 (dec 2021).
- Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. 2021. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10318–10327.
- Shengan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. 2024. Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 20299–20309.
- Radu Alexandru Rosu and Sven Behnke. 2023. Permutosdf: Fast multi-view reconstruction with implicit surfaces using permutohedral lattices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8466–8475.
- Gabriel Schwartz, Shih-En Wei, Te-Li Wang, Stephen Lombardi, Tomas Simon, Jason Saragih, and Yaser Sheikh. 2020. The eyes have it: An integrated eye and face model for photorealistic facial animation. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 91–1.
- Steven M Seitz and Charles R Dyer. 1999. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision* 35 (1999), 151–173.
- Ruizhi Shao, Zerong Zheng, Hanzhang Tu, Boning Liu, Hongwen Zhang, and Yebin Liu. 2023. Tensor4d: Efficient neural 4d decomposition for high-fidelity dynamic reconstruction and rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16632–16642.
- Cheng Sun, Min Sun, and Hwann-Tzong Chen. 2022. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5459–5469.
- Li Sun, Zicheng Liu, and Ming-Ting Sun. 2015. Real time gaze estimation with a consumer depth camera. *Information Sciences—Informatics and Computer Science, Intelligent Systems, Applications: An International Journal* 320, C (2015), 346–360.
- Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. 2021. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11358–11367.
- Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. 2016. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2387–2395.
- Edgar Treitschke, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. 2021. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 12959–12970.
- Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. 2022. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 5481–5490.
- Hermann Von Helmholtz. 1925. *Helmholtz's treatise on physiological optics*. Vol. 3. Optical Society of America.
- Kang Wang and Qiang Ji. 2017. Real time eye gaze tracking with 3d deformable eye-face model. In *Proceedings of the IEEE International Conference on Computer Vision*. 1003–1011.
- Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. *Advances in Neural Information Processing Systems* 34 (2021), 27171–27183.
- Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. 2023. NeuS2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3295–3306.
- Quan Wen, Derek Bradley, Thabo Beeler, Seonwook Park, Otmar Hilliges, Junhai Yong, and Feng Xu. 2020. Accurate Real-time 3D Gaze Tracking Using a Lightweight Eyeball Calibration. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 475–485.
- Quan Wen, Feng Xu, Ming Lu, and Jun-Hai Yong. 2017. Real-time 3D eyelids tracking from semantic edges. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 1–11.
- Erroll Wood, Tadas Baltrušaitis, Louis-Philippe Morency, Peter Robinson, and Andreas Bulling. 2016a. A 3d morphable eye region model for gaze estimation. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I* 14. Springer, 297–313.
- Erroll Wood, Tadas Baltrušaitis, Louis-Philippe Morency, Peter Robinson, and Andreas Bulling. 2016b. Learning an appearance-based gaze estimator from one million synthesised images. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. 131–138.
- Jun Xiang, Xuan Gao, Yudong Guo, and Juyong Zhang. 2024. FlashAvatar: High-fidelity Head Avatar with Efficient Gaussian Embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1802–1812.
- Yuelang Xu, Benwang Chen, Zhe Li, Hongwen Zhang, Lizhen Wang, Zerong Zheng, and Yebin Liu. 2024. Gaussian head avatar: Ultra high-fidelity head avatar via dynamic gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1931–1941.
- Yuelang Xu, Lizhen Wang, Xiaochen Zhao, Hongwen Zhang, and Yebin Liu. 2023. Avatarmav: Fast 3d head avatar reconstruction using motion-aware neural voxels. In *ACM SIGGRAPH 2023 Conference Proceedings*. 1–10.
- Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. 2021. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems* 34 (2021), 4805–4815.
- Lior Yariv, Peter Hedman, Christian Reiser, Dor Verbin, Pratul P Srinivasan, Richard Szeliski, Jonathan T Barron, and Ben Mildenhall. 2023. Bakedsdf: Meshing neural sdfs for real-time view synthesis. In *ACM SIGGRAPH 2023 Conference Proceedings*. 1–9.
- Zehao Yu, Torsten Sattler, and Andreas Geiger. 2024. Gaussian Opacity Fields: Efficient High-quality Compact Surface Reconstruction in Unbounded Scenes.

- arXiv:2404.10772* (2024).
- Xiaochen Zhao, Lizhen Wang, Jingxiang Sun, Hongwen Zhang, Jinli Suo, and Yebin Liu. 2023. Havatar: High-fidelity head avatar via facial model conditioned neural radiance field. *ACM Transactions on Graphics* 43, 1 (2023), 1–16.
- Yufeng Zheng, Victoria Fernández Abrevaya, Marcel C Bühler, Xu Chen, Michael J Black, and Otmar Hilliges. 2022. Im avatar: Implicit morphable head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13545–13555.
- Yufeng Zheng, Wang Yifan, Gordon Wetzstein, Michael J Black, and Otmar Hilliges. 2023. Pointavatar: Deformable point-based head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 21057–21067.
- Wojciech Zienonka, Timo Bolkart, and Justus Thies. 2023. Instant volumetric head avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4574–4584.



Fig. 9. Qualitative comparisons on real data. The results cover different ethnicities and genders.

## A DETAILS OF DATA PREPROCESS

Given an RGB input, we first get the face mask by the video matting method [Lin et al. 2022] and detect eye landmarks, including eyelid and iris landmarks, by the commercial service of SenseTime. Then, we generate the eyelid and iris mask by filling the polygon enclosed by detected landmarks. The iris mask is used for eyeball calibration, while the eyelid mask is used to generate the final mask for eyelid reconstruction.

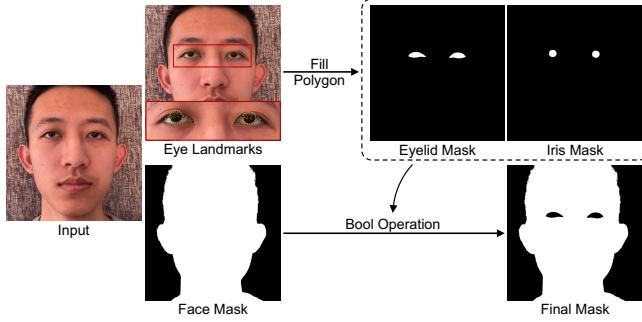


Fig. 10. Mask generation. The final mask used for eyelid reconstruction excludes the eyeball region by the bool operation.

## B DETAILS OF EYEBALL CALIBRATION

In order to acquire good initialization for the iris alignment, we adopt a coarse-to-fine strategy by first optimizing the eyeball position and scale by aligning the iris center and radius while keeping eyeball rotations fixed:

$$E(\mathbf{P}^e, s; \mathbf{P}_i^c, \mathcal{T}, \mathcal{M}) = \sum_i^n \|\mathbf{o}_i - \hat{\mathbf{o}}_i\|_1 + \sum_i^n \|r_i - \hat{r}_i\|_1 \quad (23)$$

where  $\mathbf{o} \in \mathbb{R}^2$  is the 2D center of the ground-truth iris mask.  $r \in \mathbb{R}$  is the 2D bounding box size (i.e.,  $\max(\text{width}, \text{height})$ ) of the ground-truth iris mask. The symbol with hat represents the results of the rendered iris mask. Based on the coarse results, we take eyeball rotations into the optimization and change the objective to aligning the iris mask. Note that Eq. 23 also participate in this stage for regularization. So, the full objective is formulated as

$$\begin{aligned} E(\mathbf{P}^e, \mathbf{R}_i^e, s; \mathbf{P}_i^c, \mathcal{T}, \mathcal{M}) &= \sum_i^n \|\mathcal{M}_i - \hat{\mathcal{M}}_i\|_2^2 \\ &+ \lambda_1 \sum_i^n \|\mathbf{o}_i - \hat{\mathbf{o}}_i\|_1 + \lambda_2 \sum_i^n \|r_i - \hat{r}_i\|_1 \end{aligned} \quad (24)$$

$$\hat{\mathcal{M}}_i = \Pi(\mathcal{T}, s, \mathbf{P}^e, \mathbf{P}_i^c, \mathbf{R}_i^e) \quad (25)$$

where we use  $\lambda_1 = 0.1$  and  $\lambda_2 = 0.01$  in our implementation. The whole calibration costs about ten minutes on one NVIDIA RTX 3090.

## C DISENTANGLEMENT FOR BOTH EYES

To extend the disentanglement strategy from {eye, others} to {left eye, right eye, others}, we further split the eye code  $\varphi_i^e$  into  $\varphi_i^{le}$  and  $\varphi_i^{re}$  for the left and right eye, respectively. The strategy is depicted

in Tab. 3.  $B^{le}$  and  $B^{re}$  denote the bounding boxes of the left and right eye, respectively.  $B^o$  is the other region.

Table 3. Disentanglement strategy.

change			keep same		
$\varphi_i^{le}$	$\varphi_i^{re}$	$\varphi_i^o$	$B^{le}$	$B^{re}$	$B^o$
		✓	✓	✓	
✓				✓	✓
	✓		✓		✓

For example, when changing the left eye code  $\varphi_i^{le}$ , the hyper-space coordinates of points in the  $B^{re}$  and  $B^o$  are encouraged to keep the same as before. Specifically, the disentangle loss is extended as:

$$\begin{aligned} \mathcal{L}_{det} &= \frac{1}{N_1} \sum_{\mathbf{X}^o \in B_1} \|H(\mathbf{X}^o, [\varphi_i^{le}, \varphi_i^{re}, \varepsilon]) - H(\mathbf{X}^o, [\varphi_i^{le}, \varphi_i^{re}, \varphi_i^o])\|_1 \\ &+ \frac{1}{N_2} \sum_{\mathbf{X}^o \in B_2} \|H(\mathbf{X}^o, [\varepsilon, \varphi_i^{re}, \varphi_i^o]) - H(\mathbf{X}^o, [\varphi_i^{le}, \varphi_i^{re}, \varphi_i^o])\|_1 \\ &+ \frac{1}{N_3} \sum_{\mathbf{X}^o \in B_3} \|H(\mathbf{X}^o, [\varphi_i^{le}, \varepsilon, \varphi_i^o]) - H(\mathbf{X}^o, [\varphi_i^{le}, \varphi_i^{re}, \varphi_i^o])\|_1 \end{aligned} \quad (26)$$

where  $H(\mathbf{X}^o, \varphi_i) = [F_d(\mathbf{X}^o, \varphi_i), F_w(\mathbf{X}^o, \varphi_i)]$  and  $\varepsilon$  is a random latent code sampled by standard normal distribution.  $B_1 = B^{le} \cup B^{re}$ ,  $B_2 = B^{re} \cup B^o$ , and  $B_3 = B^{le} \cup B^o$  are the sets of points in the corresponding regions, while  $N_1$ ,  $N_2$ , and  $N_3$  are the corresponding number of points.

## D MODELING OF CLOSING EYES

Closing eyes is an important movement for realistic eyelid animation but cannot be directly controlled by eyeball rotations. In order to model and control it, we additionally introduce two latent codes of 32 dimensions (named closing code) to encode the closing conditions, representing closed and non-closed eyes, respectively. According to the eyelid masks, we can automatically obtain the closing condition of each frame by setting a threshold of the pixel distance between upper and lower eyelids, and we feed the corresponding closing code into the deformation network and topology network during training. When inserting closing eyes into an animation sequence, we linearly interpolate the closing code of closed eyes with that of non-closed eyes.

## E MOTIVATION OF HYPER-SPACE

The hyper-space is designed to handle the topology changes during eyelids moving, especially when participants closing their eyes. To demonstrate its benefits, we add another ablation study that removes hyper-space and only model the deformation by the deformation network  $F_d$ . Results in Fig. 12 show that without hyper-space, the algorithm fails to model the closed eyes. Apart from topology changes, the ability of geometry modeling provided by the hyper-space also benefits the geometry reconstruction of non-closed eyelids. We compare geometry metrics of two experiment settings on our synthetic dataset. Results in Tab. 4 show the advantages of our full method.

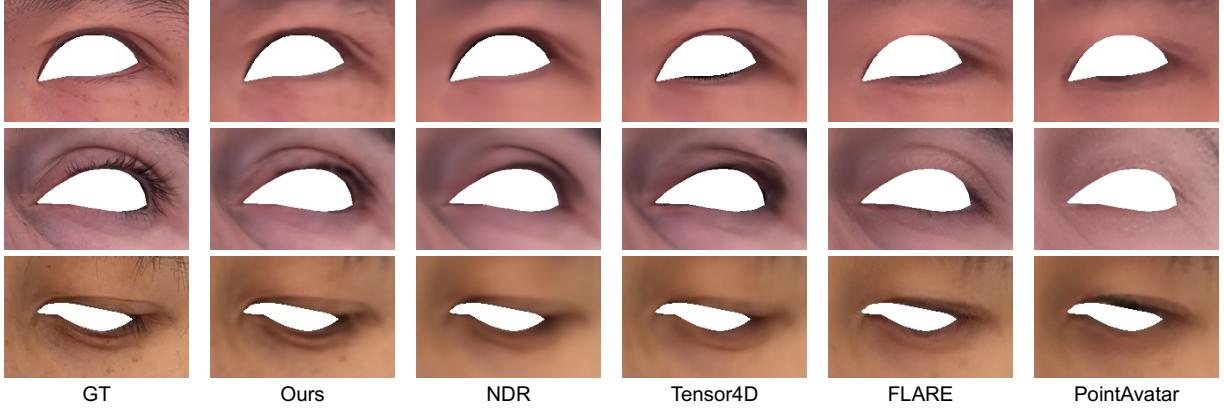


Fig. 11. Rendered results on real data. Benefiting from the geometry improvements, our method outperforms other methods with the same-level capture setups.



Fig. 12. Ablation for ablation. Without hyper-space, the algorithm cannot model the closed eye.

Table 4. Ablation study of hyper-space.

Methods	Depth Error ↓				Chamfer Distance ↓			
	ID-1	ID-2	ID-3	ID-4	ID-1	ID-2	ID-3	ID-4
w/o hyper	1.13	0.72	1.20	1.04	0.112	0.071	0.153	0.119
Full (Ours)	<b>0.79</b>	<b>0.48</b>	<b>0.60</b>	<b>0.49</b>	<b>0.044</b>	<b>0.038</b>	<b>0.062</b>	<b>0.034</b>

## F RENDERING QUALITY

Apart from geometry, our method can also obtain colors from the appearance network  $F_c$ . We compare the rendering quality of different methods both qualitatively and quantitatively. As shown in Fig. 11, our method show better rendering quality in the eye region. Specifically, PointAvatar [Zheng et al. 2023] and FLARE [Bharadwaj et al. 2023] suffer from the misalignment of wrinkles, while NDR [Cai et al. 2022] and Tensor4D [Shao et al. 2023] miss some details. The mean PSNR of all participants in Tab. 5 also demonstrate our advantages.

Table 5. Mean PSNR of different methods in the eye region.

Ours	NDR	Tensor4D	FLARE	PointAvatar
30.29	28.44	28.19	21.97	20.86

## G DIFFERENCES FROM SIMILAR METHODS

Although the linear combination of our adaptive anchor grids is similar with some techniques like NeRFBlendShape [Gao et al. 2022] and NeRSemble [Kirschstein et al. 2023] in formulation, there are

some in-depth differences among these methods, and our method has some advantages. First, the impact of linear combination in NeRFBlendShape [Gao et al. 2022] is different from NeRSemble [Kirschstein et al. 2023] and ours. As they do not use canonical plus deformation architecture, the linear combination in NeRFBlendShape [Gao et al. 2022] aims for capabilities of representing various expressions rather than expression-dependent encodings of features. Second, the combination coefficients of NeRSemble [Kirschstein et al. 2023] are learnable parameters, while ours are semantic parameters (i.e., eyeball rotations) that are fixed once acquired, which is more suitable for animation control. As for reconstruction, we present an ablation that changes our eyeball rotations to learnable parameters just as NeRSemble [Kirschstein et al. 2023]. The quantitative results shown in Tab. 6 show that using learnable parameters has no obvious advantages on geometry reconstruction.

Table 6. Ablation study of learnable parameters.

Methods	Depth Error ↓				Chamfer Distance ↓			
	ID-1	ID-2	ID-3	ID-4	ID-1	ID-2	ID-3	ID-4
learnable gaze	1.12	1.20	0.72	0.80	0.071	0.198	<b>0.053</b>	0.050
Full (Ours)	<b>0.79</b>	<b>0.48</b>	<b>0.60</b>	<b>0.49</b>	<b>0.044</b>	<b>0.038</b>	0.062	<b>0.034</b>

## H QUANTITATIVE COMPARISONS OF OTHER FACIAL REGIONS

Our method models the other facial regions together with eyelids in the same SDF field. The difference is that the other facial regions are only modeled by the naive combination of NDR [Cai et al. 2022] and NeuDA [Cai et al. 2023], while the eye region benefits from the proposed techniques (contact loss and gaze-dependent adaptive anchor grid). In order to fully show the advantages of our method, we supplement the geometry metrics of the other facial regions on synthetic data. As shown in Tab. 7, we compare our method with the two most competitive methods (NDR [Cai et al. 2022] and Tensor4D [Shao et al. 2023]). Our method also achieve the best geometry reconstruction.

Table 7. Quantitative comparisons of other facial regions on synthetic data.

Methods	Depth Error ↓				Chamfer Distance ↓			
	ID-1	ID-2	ID-3	ID-4	ID-1	ID-2	ID-3	ID-4
NDR	0.50	0.64	0.51	0.77	0.050	0.092	0.042	0.109
Tensor4D	1.02	1.41	1.15	1.06	0.149	0.418	0.235	0.183
Ours	<b>0.35</b>	<b>0.40</b>	<b>0.39</b>	<b>0.44</b>	<b>0.040</b>	<b>0.029</b>	<b>0.025</b>	<b>0.078</b>

## I INTERPOLATION BEHAVIORS

Our reconstructed eyelids can be animated by semantic parameters (i.e., eyeball rotations). The valid range of rotation values vary from person to person because different people can rotate their eyeballs at different angles. In general, most case works well with a pitch angle from  $-20^\circ$  to  $20^\circ$  and a yaw angle from  $-30^\circ$  to  $30^\circ$  (an example is shown in Fig. 13). A larger angle may result in artifacts if it is beyond the range of eyeball rotations in the training data.

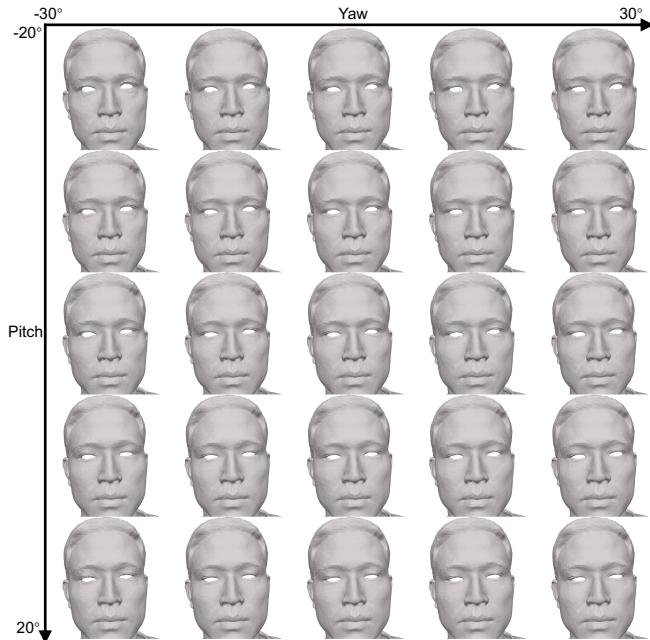


Fig. 13. Visualization of interpolation behaviors. Our reconstructed eyelids can be animated by eyeball rotations.

## J TOLERANCE OF CALIBRATION ERRORS

As our eyelid reconstruction relies on eyeball parameters from eyeball calibration, it is valuable to explore its robustness to calibration errors. However, as we do not have ground-truth eyeball parameters, it is infeasible to get the calibration errors directly. Instead, we make calibration errors manually by adding man-made offsets to the eyeball parameters of our full method. Considering the key to eyeball calibration is to determine the eyeball positions, this experiment mainly focuses on the robustness to position errors. Specifically, the added manual offset is a vector that contains a random direction with a ratio of eyeball radius. For each ID in our synthetic data, we randomly sample the direction and uniformly sample the ratio

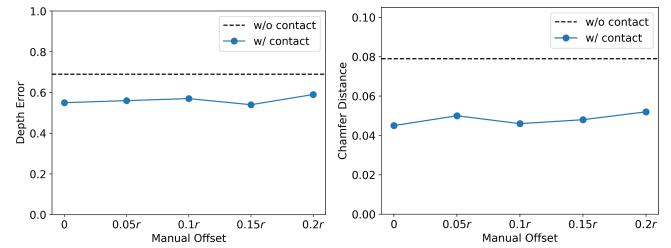


Fig. 14. Trend of reconstruction performance changing with calibration errors. The x-axis is the length (represented by the ratio of eyeball radius) of manual offsets. Using inaccurate parameters to construct contact loss is still beneficial for eyelid reconstruction and outperforms the w/o contact ablation.

from 0.05 to 0.2. The mean depth error and Chamfer distance of setups with different manual offsets are used to reflect the trend of reconstruction performance changing with position errors. As shown in Fig. 14, with the manual offset increases, the reconstruction performance of setups with non-zero manual offsets is still comparable with that of the original setup (0 manual offset). In addition, even with a large offset of 0.2 times the radius length, using the inaccurate parameters to construct contact loss is still beneficial for eyelid reconstruction, outperforming the w/o contact ablation. To some extent, these phenomena demonstrate that our method is robust to calibration errors.

## K MORE RESULTS ON SYNTHETIC DATA

We show qualitative comparisons on synthetic data in Fig. 15. Similar to the results on real data, PointAvatar [Zheng et al. 2023] can only reconstruct coarse contours of facial details, and the results of Wen et al. [2017] are seriously restricted to the model capacity. FLARE [Bharadwaj et al. 2023] can reconstruct some personal details, but the results have severe self-intersection. The results of Tensor4D [Shao et al. 2023] contain some bumpy artifacts, while the results of NDR [Cai et al. 2022] miss some subtle details. Our method outputs more plausible geometry and reconstructs more subtle details in the eye region. The depth error maps in the red rectangle also demonstrate the advantages of our method. Note that PointAvatar [Zheng et al. 2023] is a point-based method and thus only contains sparse values on the error maps.



Fig. 15. Reconstructed results on synthetic data with corresponding depth error map of the eye region.