

---

# GLVD: Guided Learned Vertex Descent

---

**Pol Caselles Rico**

Institut de Robotica i Informatica Industrial, CSIC-UPC  
Crisalix SA  
Barcelona, Spain  
pcaselles22@gmail.com

**Francesc Moreno Noguera\***

Amazon  
Barcelona, Spain  
cescmore@amazon.es

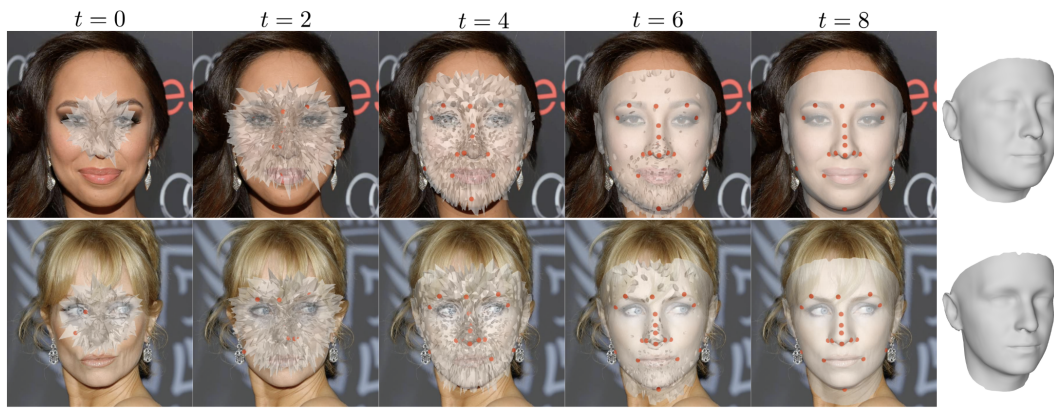


Figure 1: Qualitative results for two in-the-wild subjects reconstructed using GLVD .

## Abstract

Existing 3D face modeling methods usually depend on 3D Morphable Models, which inherently constrain the representation capacity to fixed shape priors. Optimization-based approaches offer high-quality reconstructions but tend to be computationally expensive. In this work, we introduce **GLVD**, a hybrid method for 3D face reconstruction from few-shot images that extends Learned Vertex Descent (LVD) [11] by integrating per-vertex neural field optimization with global structural guidance from dynamically predicted 3D keypoints. By incorporating relative spatial encoding, GLVD iteratively refines mesh vertices without requiring dense 3D supervision. This enables expressive and adaptable geometry reconstruction while maintaining computational efficiency. GLVD achieves state-of-the-art performance in single-view settings and remains highly competitive in multi-view scenarios, all while substantially reducing inference time.

## 1 Introduction

High-fidelity 3D face modeling from images is a long-standing challenge in the computer vision community, with broad impact across applications such as Virtual Reality, Augmented Reality, healthcare, entertainment, and security. Reconstructing an accurate and coherent digital human representation from a few input images is a highly ill-posed task —particularly in uncontrolled environments— often requiring geometry-aware methods guided by strong prior assumptions. Adding to this challenge is the scarcity of abundant, high-quality 3D training data captured under such

---

\*This work was conducted independently and does not relate to the author’s position at Amazon

unconstrained conditions, which limits the generalization ability of existing models in diverse real-world scenarios.

Statistical priors based on parametric 3D Morphable Models [2, 16, 41, 51, 54, 55, 62, 63, 64, 69, 17, 81, 80, 66, 30] have become the standard approach for few-shot 3D face reconstruction. By encoding facial geometry using a low-dimensional set of parameters, 3DMMs provide a robust and efficient framework, particularly effective in scenarios with limited or single-view image input. However, their effectiveness is hindered by two key limitations: a bias toward the mean shape [59], and the inherently constrained expressiveness of parametric models. These models typically operate within fixed low-dimensional subspaces, making it difficult to capture fine-grained details or adapt to out-of-distribution variations.

Model-free representations using voxels [26], meshes, point clouds, or Gaussian splatting [28] offer greater flexibility and high reconstruction accuracy, but they face scalability and resolution trade-offs due to memory and topology constraints. Neural fields address these challenges by encoding geometry and appearance as continuous functions via neural networks. These methods can reconstruct detailed surfaces from images without requiring 3D supervision, but they typically depend on multi-view inputs and suffer from high inference costs [38, 4]. Recent works [42, 67, 6] have made significant progress in reducing computational overhead. However, converting such representations into well-structured, topologically consistent meshes suitable for animation or rendering often necessitates additional post-processing, commonly involving template fitting.

Optimization-based approaches produce accurate and detailed results through iterative refinement [38, 67, 6, 52, 8, 9], while feed-forward methods [57, 58, 23, 22] offer faster inference at the cost of robustness and accuracy, especially under out-of-distribution conditions [37]. More recently, Learned Vertex Descent (LVD) [11] introduced a hybrid strategy that uses pixel-aligned image features to guide iterative template fitting. Despite its effectiveness, LVD relies on large-scale training data with posed images and corresponding 3D geometry, and it lacks explicit global structure, predicting vertex trajectories independently and depending on the image encoder for implicit coherence.

To overcome these limitations, we propose leveraging a 3D face landmark estimator derived from a 2D image-based predictor to guide 3D shape refinement. We introduce GLVD, a learning-based optimization approach that fuses local and global cues by combining per-vertex neural fields with dynamically predicted 3D keypoints. Each neural field predicts 3D displacements for its associated vertex based on local features sampled at its current position, while the keypoint ensemble provides global structural guidance that informs and regularizes the optimization process. Central to our method is a relative encoding scheme, where each vertex is transformed based on the current keypoint estimates, allowing the network to learn geometry-aware updates that are conditioned on the evolving global structure.

The combination of local neural fields and global keypoint-based guidance in GLVD enables more precise control and adaptive refinement of 3D facial geometry, as shown in Figure 1. Leveraging this fusion, we conduct a comprehensive evaluation on both single-view and multi-view 3D face reconstruction benchmarks. Our approach achieves state-of-the-art performance in single-image reconstruction and remains competitive with optimization-based methods in multi-view scenarios, demonstrating its robustness, accuracy, and broad applicability.

## 2 Related work

**3D Morphable Models (3DMM).** The use of 3D Morphable Models (3DMMs) has become the standard paradigm for reconstructing 3D facial geometry from images, particularly in single-view or few-shot scenarios. These statistical models [46, 32, 3] are widely adopted and mainly focus on the facial region. In single-image settings, several methods have demonstrated effective reconstruction performance [62, 17, 81, 66, 80, 30].

Recent advancements in single-image 3D face reconstruction have explored both parametric and non-parametric strategies to improve accuracy, robustness, and detail preservation. 3DDFAv2 [21] proposes a regression-based approach combining a lightweight architecture with meta-joint optimization to achieve real-time performance while maintaining alignment accuracy. Building upon this, 3DDFAv3 [68] introduces Part Re-projection Distance Loss, which leverages dense facial part segmentation as a strong geometric prior for guiding 3D reconstruction, especially under extreme

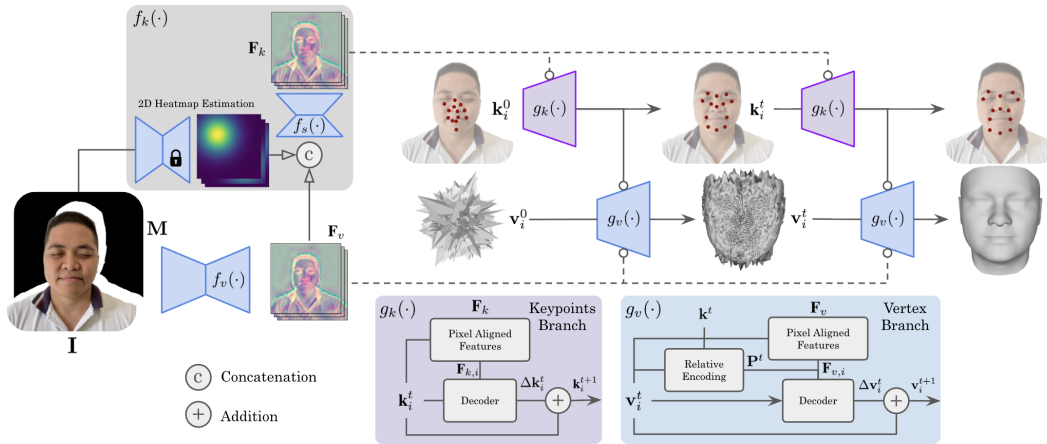


Figure 2: **Overview of GLVD** . Given one or more input images, each paired with a head mask and calibrated camera parameters, the method reconstructs a 3D face mesh through two branches. (1) The 3D Keypoint Branch predicts a set of facial keypoints by extracting localized image features and estimating their 3D displacements iteratively. (2) The 3D Vertex Branch refines the full-face geometry by leveraging these keypoints to encode relative spatial information for each surface vertex. This branch extracts pixel-aligned features and predicts vertex-wise displacements in an iterative optimization process.

expressions where landmarks are unreliable. SADNet [56] introduces a self-aligned dual-regression framework that disentangles pose-dependent and pose-independent features and fuses them through an occlusion-aware alignment strategy. HRN [30] proposes a hierarchical representation network that disentangles geometric components and incorporates high-frequency priors, enabling the reconstruction of fine facial details, such as wrinkles and skin texture, from in-the-wild images.

**Neural Fields for Face Reconstruction.** Neural fields have emerged as a leading approach for 3D reconstruction, offering continuous and high-fidelity representations of geometry and appearance. They have been successfully applied to full-head and facial modeling tasks using techniques such as volume rendering and surface priors [40, 44, 45, 52, 8, 9, 6]. Hybrid models combine parametric approaches such as 3DMMs with neural implicit functions to increase control and expressiveness. For instance, IMFace [74] and IMFace++ [75] introduce implicit displacement fields to refine a 3DMM geometry, and NeuFace [76] proposed an approximated BRDF integration and a low-rank prior for human face rendering. In [7], authors combine geometry-aware features with image features that output a signed distance field. However, these approaches tend to collapse and generate artifacts.

When several input images are available, a line of research [20, 78, 79, 5, 82, 19] aims to obtain animatable full-head avatars from videos. Building on the recent success of Gaussian Splatting [28], several works [35, 15, 49, 61] have integrated this representation to improve rendering efficiency and visual fidelity. Combining them with 3DMMs has been explored in recent works, with methods such as HeadGAP [77] and GPHM [71] that learns parametric head models using Gaussian-splatting-based models. While effective, these methods often rely on dense multi-view input. In contrast, model-free feed-forward approaches use pixel-aligned features [57] for faster inference from sparse views.

Model-free methods leveraging pixel-aligned features have gained popularity for fast 3D reconstruction, as they avoid the need for test-time optimization [57, 58, 23, 1, 60, 13, 22]. PIFu [57] introduced pixel-aligned implicit functions, using 2D image features to predict 3D occupancy from single or multiple views, and PHORHUM [1] extended this by employing signed distance fields for surface modeling. JIFF [7] proposed combining features from a face morphable model and pixel-aligned features. In contrast, using volumetric rendering, KeypointNeRF aggregates pixel-aligned features with a relative spatial encoder. More recently, LVD [11] emerged as a learning-based optimization approach that leverages pixel-aligned features to guide an iterative-based template fitting. While it acquires a good trade-off between computation requirements and accuracy for human mesh recovery, it remains unexplored for 3D face modeling. However, single feed-forward methods based on pixel-aligned features still lag behind optimization-based approaches in terms of reconstruction quality [9].

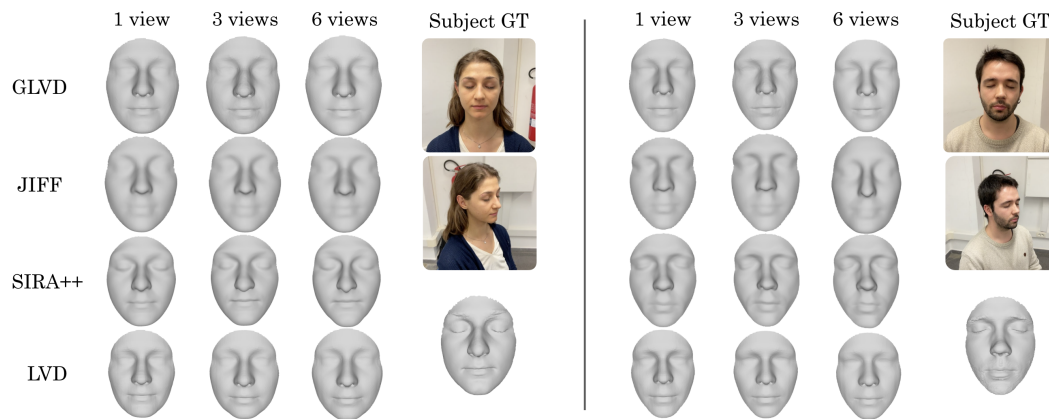


Figure 3: **Qualitative results on two subjects of the H3DS dataset**, for LVD [11], SIRA++[9], JIFF [7], and GLVD , with an increasing number of input views.

**Encoding Representations for 3D Reconstruction.** Previous approaches have explored various spatial encoding strategies to enhance learning. PVA [50] and PortraitNeRF [18] utilize face-centric coordinate systems, while ARCH [25] and ARCH++ [24] adopt canonical body coordinates. KeypointNeRF [39] proposes to encode relative spatial 3D information in the form of depth via sparse 3D keypoints. In [14], the authors introduced a three-step pipeline of landmark selection, low-dimensional embedding via MDS, and distance-based triangulation to embed points. GLVD follows a similar idea by selecting identity-specific facial keypoints and encoding mesh vertices through their Euclidean distances to these keypoints within the learned space. In this work, we conduct a thorough investigation of spatial encoding and find that a simple yet effective encoding based on relative distances w.r.t 3D keypoints [12] yields effective results in combination with neural fields guided by pixel-aligned features. We adopt a canonical aligned space to stabilize training. In addition to achieving state-of-the-art results on 3D face reconstruction from as few as a single input image, our approach can also be used beyond face modeling.

**Mesh Recovery for body.** Several works [34, 72, 36, 33, 73] focus on full-body mesh recovery and remain unexplored in the context of face modeling. These methods often operate in single-image settings [34, 72, 36, 33] or on video sequences [31]. METRO [34] and DeFormer [72] use transformers to jointly process mesh joints and vertices, differing in attention aggregation but both relying on global self-attention. METRO further conditions on a global image embedding, discarding pixel-aligned spatial detail. In contrast, 3D Virtual Markers [36] predicts latent 3D markers and reconstructs the mesh as a linear combination, while One-Stage Mesh Recovery [33] directly regresses SMPL parameters via a transformer. Both are constrained by the limited expressiveness of low-dimensional latent spaces. PyMAF-X [73] introduces an iterative refinement approach that samples pixel-aligned features from prior vertex predictions, but does not leverage explicit landmark or keypoint constraints for structural supervision. GART [31] leverages skeletal priors and temporal video input for avatar reconstruction, whereas GLVD is purely image-driven and effective in both monocular and few-shot settings. Finally, [53] replaces PCA-based 3D face models with convolutional mesh autoencoders to learn shape priors, but focuses on latent mesh encoding rather than direct facial geometry estimation from RGB images as in GLVD

### 3 Method

In this section, we present our method for 3D face reconstruction using learned keypoint guidance. We first review the Learned Vertex Descent (LVD) framework [11] and then introduce our key architectural innovations. The section concludes with details on training and inference. An overview is shown in Figure 2, with implementation details in the supplementary material.



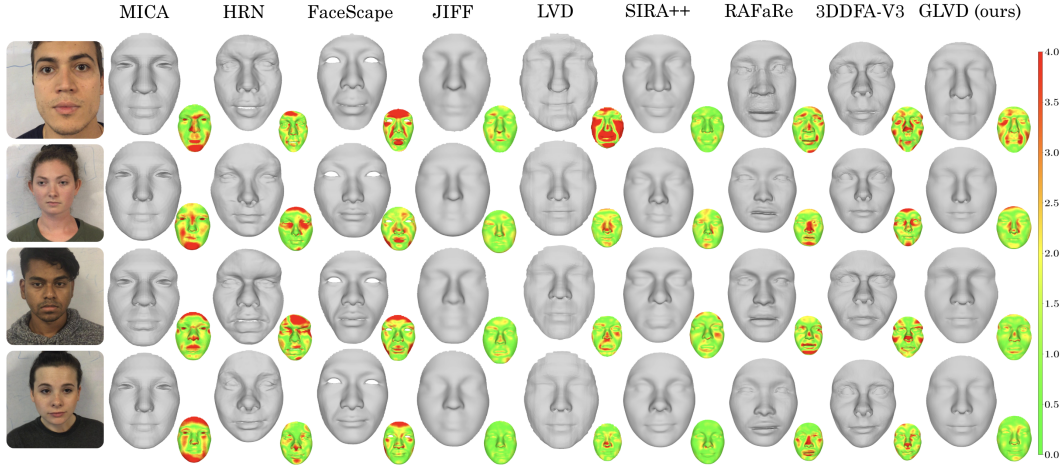


Figure 4: **Qualitative results on the 3DFAW dataset for a single input image.** Each 3D reconstructed face is accompanied by a heatmap, where reddish areas indicate larger errors in mm.

### 3.1 Background: Learned Vertex Descent

Learned Vertex Descent (LVD) [11] is an optimization-based method for 3D human shape reconstruction from single-view images or scans. While it has been applied to full-body and hand reconstruction, we explore its potential for 3D face modeling. The model learns a transformation  $g(\cdot)$  which takes the current 3D vertex position at iteration  $t$  and the associated 2D image features  $\mathbf{f}_i$  for vertex  $i$  as input, and outputs a displacement vector  $\Delta \mathbf{v}_i$ :

$$g : (\mathbf{v}_i^t, \mathbf{f}_i) \mapsto \Delta \mathbf{v}_i. \quad (1)$$

This vector represents the correction needed to align the vertex with its ground truth position  $\hat{\mathbf{v}}_i$ . The updated vertex position is then given by  $\mathbf{v}_i^{t+1} = \mathbf{v}_i^t + \Delta \mathbf{v}_i$ . The reconstruction process involves iteratively applying this update to refine the 3D shape.

### 3.2 Problem Definition

Our goal is to recover a 3D face surface  $\mathcal{S}$  from a small set of input images  $\{\mathbf{I}_n\}_{n=1}^N$ , where each image  $\mathbf{I}_n$  is paired with a head mask  $\mathbf{M}_n$  and calibrated camera parameters  $\mathbf{T}_n$ . The surface  $\mathcal{S}$  is represented by a fixed topology of 7,225 vertices and 14,342 faces.

We aim to incorporate global 3D-aware guidance into the per-vertex optimization by leveraging a relative encoding based on the Euclidean distances between vertices and keypoints. As a result, we propose a two-stage architecture to address limited multi-view input images: (1) a 3D keypoint estimation module that defines spatial keypoints on the facial surface by estimating displacements, and (2) a vertex prediction module that encodes vertices relative to these keypoints to estimate vertex updates. Our formulation does not rely on a predefined parametric model or fixed joint sets, making it adaptable to arbitrary topologies. A sparse set of surface points is conveniently selected to act as ground-truth keypoints.

### 3.3 Learning-Based Keypoint Estimation

Our goal is to establish a reference space to guide vertex optimization towards the target surface. To achieve this, we estimate a fixed set of  $K$  3D keypoints  $\{\mathbf{k}_j^t\}_{j=1}^K$ , where  $\mathbf{k}_j^t \in \mathbb{R}^3$  represents the  $j$ -th keypoint on the target surface, at iteration  $t$ . These keypoints are used for encoding the relative positions of query vertices  $\mathbf{v}_i$ .

The 3D keypoint branch consists of two components: one responsible for extracting image features  $\mathbf{F}_k$ , and another focused on learning 3D keypoint displacements  $\Delta \mathbf{k}_i$ . To compute  $\mathbf{F}_k$ , we first generate facial keypoint heatmaps using off-the-shelf HRNet [65], which are then concatenated with vertex image features  $\mathbf{F}_v$  obtained from the first stack of the Hourglass network [43]. This combined feature map is subsequently refined using a single-stack Hourglass module  $f_s(\cdot)$ :

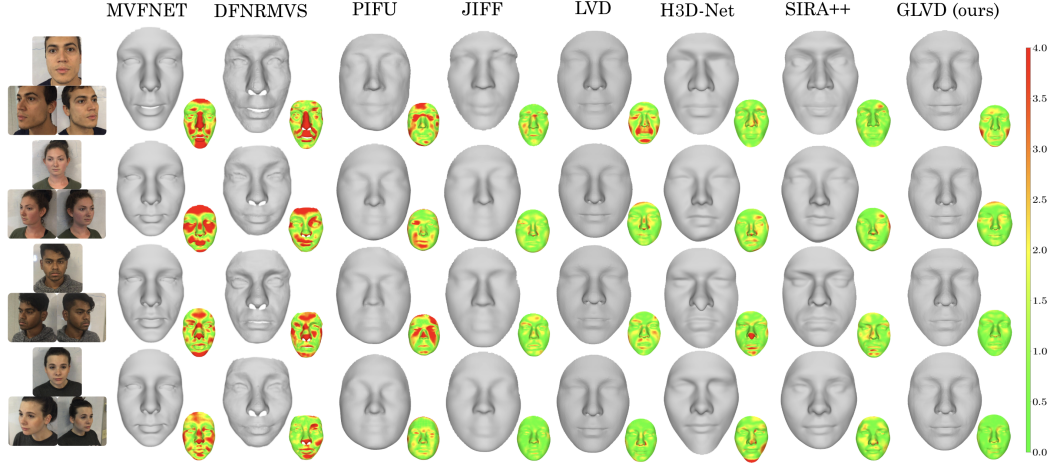


Figure 5: **Qualitative results on the 3DFAW dataset for three input images.** Each 3D reconstructed face is accompanied by a heatmap, where reddish areas indicate larger errors in mm.

$$f_k : (\mathbf{I}, \mathbf{M}, \mathbf{F}_v) \mapsto \mathbf{F}_k. \quad (2)$$

The keypoints employed to guide the vertex branch do not necessarily coincide with the facial landmarks defined by HRNet. We adopt a strategy of predicting vertex displacements from local features, as this approach has been demonstrated to yield more accurate geometric detail [57, 58, 10, 11]. To estimate the 3D keypoints, we implement  $g_k(\cdot)$  (Eq. 1) with a 3-layer MLP that takes as input the current estimate of keypoint  $\mathbf{k}_i^t$  and its local  $F$ -dimensional local features  $\mathbf{F}_{k,i}$  extracted at the projection of  $\mathbf{k}_i^t$  on the image plane, and predicts the displacement  $\Delta \mathbf{k}_i^t$ . In the first step, we begin by uniformly sampling  $\mathbf{k}_i^0$  within a volume of size 2 centered at the origin.

### 3.4 Vertex Displacement Prediction

The vertex branch consists of two modules: a local feature extractor  $f_v : (\mathbf{I}, \mathbf{M}) \mapsto \mathbf{F}_v$  that computes image-aligned features for each projected vertex, and a regressor  $g_v(\cdot)$  that predicts the final vertex displacements  $\Delta \mathbf{v}_i$ , which is also implemented as a 3-layer MLP.

We extract local image features  $\mathbf{F}_v$  and  $\mathbf{F}_k$  for each image  $\mathbf{I}$ , and following Sec. 3.3, we estimate a set of predefined 3D keypoints at each iteration  $t$ . Given a query vertex  $\mathbf{v}_i \in \mathbb{R}^3$ , we compute a keypoint-relative encoding matrix  $\mathbf{P} \in \mathbb{R}^{K \times 3}$ , where each row represents a displacement vector  $\mathbf{P}^t = \mathbf{k}^t - \mathbf{v}_i^t$ . As shown in our ablations (Sec. 4.2), this encoding outperforms alternatives such as Euclidean distances, attention, or concatenation.

### 3.5 Displacement Learning and Optimization

The network is trained to learn the parameters of  $f_v(\cdot)$ ,  $f_k(\cdot)$ ,  $g_k(\cdot)$ , and  $g_v(\cdot)$ . We use a dataset of  $N$  training scenes, each with a ground truth mesh with known topology, posed RGB images, and head masks. We sample query points for each scene using a hybrid strategy that combines uniform sampling with points near the mesh surface. Both model components are trained to predict displacements at each iteration  $t$ . The iteration index is not explicitly encoded during training, as the model is exposed to a stochastic distribution of vertex states, making it inherently timestep-independent. At inference time, iteration  $t$  corresponds to updating each vertex by adding the displacement predicted at the previous step (Sec. 3.1). We pre-train the feature encoder on the 3D reconstruction task by augmenting it with a signed distance function (SDF) prediction head. This improves convergence behavior and leads to higher reconstruction accuracy. Further details are provided in the supplementary material.

**Keypoint Displacement Learning.** We select a consistent set of 3D keypoints to guide the surface displacement learning. For each mesh, we choose the same subset of vertices to serve as target keypoints  $\{\mathbf{k}_j^t\}$ . While their selection can be arbitrary, it must remain consistent across training

Table 1: **3D face reconstruction comparison.** Average surface error (in mm) computed over all subjects in 3DFAW and H3DS datasets. We place "-" for not applicable configurations. Optimization-based have been included for reference.

|                | 3DFAW       |             | H3DS 2.0    |             |             |             |
|----------------|-------------|-------------|-------------|-------------|-------------|-------------|
|                | 1 view      | 3 view      | 1 views     | 3 views     | 4 views     | 6 views     |
| MVFNet [69]    | -           | 1.56        | -           | 1.73        | -           | -           |
| DFNRMVS [2]    | -           | 1.69        | -           | 1.83        | -           | -           |
| DECA [17]      | 1.71        | -           | 1.99        | -           | -           | -           |
| MICA [81]      | 1.83        | -           | 2.08        | -           | -           | -           |
| FaceVerse [66] | 1.88        | -           | 2.57        | -           | -           | -           |
| FaceScape [80] | 1.61        | -           | 1.78        | -           | -           | -           |
| HRN [30]       | 1.60        | -           | 1.73        | -           | -           | -           |
| VHAP [48]      | 2.05        | -           | 2.15        | -           | -           | -           |
| 3DDFA-V3 [68]  | 1.45        | -           | 1.65        | -           | -           | -           |
| RAFaRe [22]    | 1.68        | -           | 2.54        | -           | -           | -           |
| PIFU [57]      | 2.19        | 1.99        | 1.98        | 1.70        | 1.85        | 2.03        |
| JIFF [7]       | 1.48        | 1.47        | 1.85        | 1.80        | 1.79        | 1.79        |
| LVD [11]       | 1.58        | 1.26        | 1.39        | 1.45        | 1.39        | 1.37        |
| GLVD (ours)    | <b>1.25</b> | <b>1.22</b> | <b>1.36</b> | <b>1.33</b> | <b>1.34</b> | <b>1.34</b> |
| H3D-Net [52]   | 1.70        | 1.37        | -           | 1.44        | 1.41        | 1.21        |
| SIRA++ [9]     | 1.35        | 1.32        | 1.57        | 1.18        | 1.23        | 1.04        |

scenes and during test-time inference. Each query keypoint, together with the input image  $\mathbf{I}$ , is passed to the model  $(g_k \circ f_k)(x)$ , which predicts 3D keypoint displacements  $\Delta \mathbf{k}_i^t$ .

**Surface Displacement Learning.** We train  $f_v(\cdot)$  and  $g_v(\cdot)$  separately from the 3D keypoint branch. We encode vertices relative to the ground-truth 3D keypoints. To simulate prediction uncertainty, we perturb the sampled keypoints with noise drawn from a zero-mean multivariate Gaussian distribution with spherical covariance  $\sigma^2$ . These noisy keypoints are then used for encoding. Since depth errors in camera-aligned show higher variance, we apply noise with standard deviation  $3\sigma$  along the depth axis in the camera frame.

Given a ground-truth mesh  $\hat{V} = [\hat{\mathbf{v}}_1, \dots, \hat{\mathbf{v}}_N]$  and its corresponding image  $I$ . We randomly sample  $M$  3D points  $X = \{\mathbf{x}_1, \dots, \mathbf{x}_M\}$ , and compute the loss for each of the  $M$  points:

$$\mathcal{L} = \frac{1}{M} \sum_{i=1}^M \frac{1}{N} \sum_{j=1}^N \left[ \lambda_1 \left( 1 - \frac{\Delta \mathbf{x}_i^j \cdot \hat{\Delta} \mathbf{x}_i^j}{\|\Delta \mathbf{x}_i^j\|_2 \|\hat{\Delta} \mathbf{x}_i^j\|_2} \right) + \lambda_2 \left| \|\Delta \mathbf{x}_i^j\|_2 - \|\hat{\Delta} \mathbf{x}_i^j\|_2 \right| \right] \quad (3)$$

where  $\mathbf{x}_i$  is the  $i$ -th 3D query point,  $N$  is the number of ground-truth vertices,  $\Delta \mathbf{x}_i^j$  is the predicted displacement from  $\mathbf{x}_i$  to the  $j$ -th point, and  $\hat{\Delta} \mathbf{x}_i^j$  is the corresponding ground truth displacement. The symbol  $\cdot$  denotes the dot product, and  $\|\cdot\|_2$  represents the Euclidean (L2) norm. The parameter  $\lambda_1$  controls the contribution of the directional loss term, which minimizes the angular deviation  $1 - \cos(\theta)$  between the predicted and ground truth vectors. The parameter  $\lambda_2$  weights the magnitude loss, which penalizes differences in the length of the displacement vectors. To promote locality in the extracted image features, this term is clipped during training. This weighted combination encourages both directional consistency and training stability. During training, we apply binary dropout to the image features  $\mathbf{F}_v$  to enhance robustness against unreliable predicted neural fields. Additionally, we model the 3D keypoints used for encoding vertices in the vertex branch as stochastic variables, introducing noise to the ground-truth keypoints only during training.

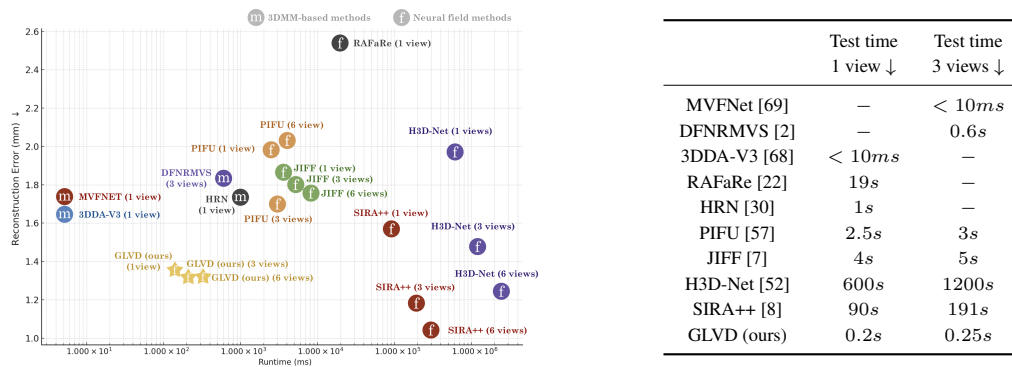


Figure 6: **Quantitative comparison on H3DS dataset with one and three input views.** **Left.** Reconstruction error (mm) is plotted against runtime for various state-of-the-art methods under different view configurations. **Right.** Inference times for single and multi-view settings.

## 4 Experiments

**Training face dataset.** We employ a proprietary dataset of 3D head scans collected from 10,000 individuals, balanced by gender and diverse in age and ethnicity. All scans are aligned to a template 3D model using non-rigid Iterative Closest Point (ICP) registration for consistency.

**H3DS 2.0.** [52, 9] It contains 60 high-quality 3D full-head scans, including hair and shoulders, paired with posed RGB images. Each image includes a foreground mask and calibrated camera parameters.

**3DFAW.** [47] This dataset provides videos recorded as well as mid-resolution 3D ground truth of the facial region. We select 5 male and 5 female scenes and use them to evaluate only the facial region.

**CelebA-HQ.** [27] This dataset comprises 30k high-quality images at 1024×1024 resolution, derived from the original CelebA dataset. We selected a subset of 6 subjects for our qualitative evaluation.

### 4.1 3D Face estimation

We conducted a comprehensive comparison of our method with several 3DMM-based reconstruction works, including MVFNet [2], DFNRMVS [69], DECA [17], MICA [81], FaceScape [80], FaceVerse [66], HRN [30], 3DDFA-v3 [68] and VHAP [48]. Additionally, we compared our approach to the model-free methods PIFU [57], JIFF [7], RAFaRe[22], H3D-Net [52] SIRA++ [9] and hybrid method LVD [11]. We used the unidirectional Chamfer distance for the quantitative evaluation, measuring the surface error from the ground truth to the predictions. The results of this comparison are summarized in Table 1. Qualitative results for 3DFAW subjects are presented in Figure 4 for the single view and in Figure 5 for the multiview setting. Results on H3DS are presented in Figure 3. We show in Figure 7 the estimated 3D face and the guiding keypoints.

GLVD demonstrates consistently strong performance across 3DFAW and H3DS evaluations. It leverages the structured topology of 3DMMs while explicitly addressing the constraint of shape representation to a predefined model space and the resulting bias toward average mean shape. As a result, GLVD outperforms 3DMM-based approaches, particularly in the single-view setting (Figure 4). In comparison to model-free single forward pass methods such as PIFU, JIFF and RAFaRe, it demonstrates superior performance in surface reconstruction, yielding surfaces with reduced errors and a more realistic shape appearance.

Optimization-based methods are considered state-of-the-art for face reconstruction, particularly in multi-view settings where the problem becomes less ill-posed. However, their high computational cost remains a key limitation. In contrast, GLVD achieves comparable accuracy with over two orders of magnitude faster inference and without requiring postprocessing or template registration. Inference times are reported in Figure 6 with 10 iterative update steps.

## 4.2 Ablation

We conduct detailed ablation study on 3DFAW and H3DS 2.0 datasets to assess the impact of key design choices and demonstrate the effectiveness of the proposed method.

Table 2 reports quantitative results using various subsets of 3D keypoints to guide GLVD. While the method is agnostic to landmark topology and supports flexible selection, we evaluate the impact of using different subsets derived from the template 3DMM. The specific landmark sets used are detailed in the supplementary material. Notably, results show that a small set of well-chosen keypoints provides effective global structural guidance, though limited coverage can introduce noise. While increasing the number of keypoints improves performance, gains in reconstruction accuracy diminish beyond a certain point. We attribute this to the fact that the keypoints are estimated. Higher-quality landmark supervision could further enhance reconstruction fidelity. In Figure 7, we provide face reconstruction results with the selected keypoints indicated for reference.

Table 2: **Reconstruction quality comparison** using a single view with varying numbers of keypoints. Chamfer distance is reported in millimeters (mm).

|           | 3DFAW↓      | H3DS↓       |
|-----------|-------------|-------------|
| GLVD - 4  | 1.92        | 1.51        |
| GLVD - 6  | 1.85        | 1.36        |
| GLVD - 12 | 1.57        | 1.42        |
| GLVD - 18 | <b>1.25</b> | <b>1.36</b> |
| GLVD - 24 | 1.58        | 1.33        |

**Keypoints encoding.** In the original LVD formulation, global structure is not explicitly modeled, as point trajectories are predicted independently, with structural coherence learned implicitly through 2D feature volumes. This results in ambiguity, as the model must resolve all possible correspondences per query without clear global guidance. In GLVD, we address this by introducing a landmark ensemble that serves as a compact global prior. These keypoints act as spatial anchors that guide vertex displacements, reducing ambiguity and promoting consistent topology. We ablate various encoding strategies in Table 3. Replacing the per-vertex head with a global attention layer (k) introduces noise and instability, presumably due to long context. Inspired by the concept of skinning weights, we model vertex-to-keypoint relations via learnable attention (j), achieving performance on par with the standard encoding (l). Concatenation (h) and distance-based encoding (i) offer no gains, while removing absolute positions and using only relative encoding (g) leads to a performance drop.

### Architecture analysis.

The GLVD architecture (Sec. 3) incorporates two core design choices: it estimates keypoints progressively during optimization and encodes vertices relative to these dynamically predicted keypoints. Table 3 presents an ablation of different training strategies. Using an L2 loss with clipping, following [11] (b), results in unstable gradients and weak directional supervision. Incorporating a pre-trained HRNet for facial landmark prediction (c) highly improves performance in both vertex reconstruction and keypoint estimation. Injecting noise during training (d) to model the encoding as a stochastic variable improves robustness and test-time stability in the presence of uncertainty. Applying binary dropout to the 2D features during training (e) encourages reliance on the geometry-aware encoding and leads to better accuracy. Finally, optimizing in canonical space (f), as done in [52, 9, 18], introduces a bias towards the mean shape. In the single-view setting, we use camera coordinates as the reference space, which removes the need for a calibrated camera at test time. We also investigate directly predicting the position of a specific vertex. We augment each query point with an identity feature derived from a Fourier embedding of the vertex’s 3D coordinates in the canonical face template. At test time, each sampled vertex is assigned a unique target identifier, and the model predicts displacements toward all possible targets, after which the displacement corresponding to the specified

Table 3: Comparison of reconstruction quality for different number of keypoints. Chamfer distance in mm.

|                             | 3DFAW↓      | H3DS 2.0↓   |
|-----------------------------|-------------|-------------|
| (a) LVD                     | 1.58        | 1.39        |
| (b) GLVD w/o loss (Eq. 3)   | 1.27        | 1.39        |
| (c) GLVD w/o HRNet          | 1.52        | 1.39        |
| (d) GLVD w/o training Noise | 1.31        | 1.42        |
| (e) GLVD w/o binary dropout | 1.27        | 1.38        |
| (f) GLVD w/ canonical space | 1.51        | 1.52        |
| (g) GLVD w/o vertex pos.    | 1.55        | 1.67        |
| (h) GLVD w/ concat Enc.     | 1.27        | 1.36        |
| (i) GLVD w/ norm Enc.       | 1.26        | 1.37        |
| (j) GLVD w/ attention Enc.  | 1.27        | 1.36        |
| (k) GLVD w/ attention layer | 1.85        | 1.97        |
| (l) GLVD                    | <b>1.25</b> | <b>1.36</b> |





Figure 7: **Qualitative results for one and three input images.** Images are from CelebA-HQ (left), H3DS2.0 (right top), and 3DFAW (right bottom). At the last iteration step, we show the predicted template and the 3D keypoints in red.

identifier is selected. However, we empirically found it to be more stable and to yield better results by encoding trajectories implicitly as done in GLVD.

## 5 Discussion

**Limitations and Future Work:** While GLVD demonstrates strong performance in few-shot 3D face reconstruction, it remains sensitive to occlusions and relies on the accuracy of keypoint predictions, which may degrade under challenging visual conditions. The focus of our method is the reconstruction of the face area by combining a hybrid method for fast and accurate prediction. Therefore, adding facial expressions is an interesting future direction. Future work may explore temporal consistency for video-based reconstruction and topology-adaptive strategies to better capture complex geometry.

**Conclusions.** In this paper, we have introduced GLVD, a hybrid approach for high-fidelity 3D face reconstruction from few-shot images. Our method introduces a novel combination of per-vertex neural fields and dynamically predicted 3D keypoints to provide both local accuracy and global structural guidance. By encoding vertex displacements relative to a sparse set of learned keypoints, our method refines mesh geometry iteratively without requiring parametric shape priors. The thorough evaluation demonstrated that our method achieves state-of-the-art performance in single-view settings and remains highly competitive in multi-view scenarios, all while substantially reducing inference time.

## 6 Acknowledgments

This work has been supported by the project GRAVATAR PID2023-151184OB-I00 funded by MCIU/AEI/10.13039/501100011033 and by ERDF, UE.

## References

- [1] Thiemo Alldieck, Mihai Zanfir, and Cristian Sminchisescu. Photorealistic monocular 3d reconstruction of humans wearing clothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1506–1515, 2022.
- [2] Ziqian Bai, Zhaopeng Cui, Jamal Ahmed Rahim, Xiaoming Liu, and Ping Tan. Deep facial non-rigid multi-view stereo. In *CVPR*, 2020.
- [3] Linchao Bao, Xiangkai Lin, Yajing Chen, Haoxian Zhang, Sheng Wang, Xuefei Zhe, Di Kang, Haozhi Huang, Xinwei Jiang, Jue Wang, Dong Yu, and Zhengyou Zhang. High-fidelity 3d digital human head creation from rgb-d selfies. *ACM Transactions on Graphics*, 2021.
- [4] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-nerf: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19697–19705, 2023.
- [5] Shrisha Bharadwaj, Yufeng Zheng, Otmar Hilliges, Michael J. Black, and Victoria Fernandez Abrevaya. Flare: Fast learning of animatable and relightable mesh avatars. *ACM Transactions on Graphics*, 42:15, December 2023.
- [6] Antonio Canela, Pol Caselles, Ibrar Malik, Eduard Ramon, Jaime García, Jordi Sánchez-Riera, Gil Triginer, and Francesc Moreno-Noguer. Instantavatar: Efficient 3d head reconstruction via surface rendering, 2023.
- [7] Yukang Cao, Guanying Chen, Kai Han, Wenqi Yang, and Kwan-Yee K. Wong. Jiff: Jointly-aligned implicit face function for high quality single view clothed human reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [8] Pol Caselles, Eduard Ramon, Jaime Garcia, Xavier Giro-i Nieto, Francesc Moreno-Noguer, and Gil Triginer. Sira: Relightable avatars from a single image. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 775–784, January 2023.
- [9] Pol Caselles, Eduard Ramon, Jaime Garcia, Gil Triginer, and Francesc Moreno-Noguer. Implicit shape and appearance priors for few-shot full head reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.
- [10] Julian Chibane and Gerard Pons-Moll. Implicit feature networks for texture completion from partial 3d data. In *Computer Vision—ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 717–725. Springer, 2020.
- [11] Enric Corona, Gerard Pons-Moll, Guillem Alenya, and Francesc Moreno-Noguer. Learned vertex descent: A new direction for 3d human model fitting. In *European Conference on Computer Vision*, pages 146–165. Springer, 2022.
- [12] Enric Corona, Albert Pumarola, Guillem Alenya, Gerard Pons-Moll, and Francesc Moreno-Noguer. Smplicit: Topology-aware generative model for clothed people. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11875–11885, 2021.
- [13] Enric Corona, Mihai Zanfir, Thiemo Alldieck, Eduard Gabriel Bazavan, Andrei Zanfir, and Cristian Sminchisescu. Structured 3d features for reconstructing controllable avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16954–16964, 2023.
- [14] Vin De Silva and Joshua B Tenenbaum. Sparse multidimensional scaling using landmark points. Technical report, technical report, Stanford University, 2004.
- [15] Helisa Dharmo, Yinyu Nie, Arthur Moreau, Jifei Song, Richard Shaw, Yiren Zhou, and Eduardo Pérez-Pellitero. Headgas: Real-time animatable head avatars via 3d gaussian splatting. In *European Conference on Computer Vision*, pages 459–476. Springer, 2024.
- [16] Pengfei Dou and Ioannis A Kakadiaris. Multi-view 3d face reconstruction with deep recurrent neural networks. *Image and Vision Computing*, 80:80–91, 2018.
- [17] Yao Feng, Haiwen Feng, Michael J Black, and Timo Bolkart. Learning an animatable detailed 3d face model from in-the-wild images. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021.
- [18] Chen Gao, Yichang Shih, Wei-Sheng Lai, Chia-Kai Liang, and Jia-Bin Huang. Portrait neural radiance fields from a single image. *arXiv preprint arXiv:2012.05903*, 2020.

- [19] Simon Giebenhain, Tobias Kirschstein, Markos Georgopoulos, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Monophm: Dynamic head reconstruction from monocular videos. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [20] Philip-William Grassal, Malte Prinzler, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. Neural head avatars from monocular rgb videos. *arXiv preprint arXiv:2112.01554*, 2021.
- [21] Jianzhu Guo, Xiangyu Zhu, Yang Yang, Fan Yang, Zhen Lei, and Stan Z Li. Towards fast, accurate and stable 3d dense face alignment. In *European Conference on Computer Vision*, pages 152–168. Springer, 2020.
- [22] Longwei Guo, Hao Zhu, Yuanxun Lu, Menghua Wu, and Xun Cao. Rafare: Learning robust and accurate non-parametric 3d face reconstruction from pseudo 2d&3d pairs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 719–727, 2023.
- [23] Tong He, John Collomosse, Hailin Jin, and Stefano Soatto. Geo-pifu: Geometry and pixel aligned implicit functions for single-view human reconstruction. *Advances in Neural Information Processing Systems*, 33:9276–9287, 2020.
- [24] Tong He, Yuanlu Xu, Shunsuke Saito, Stefano Soatto, and Tony Tung. Arch++: Animation-ready clothed human reconstruction revisited. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11046–11056, 2021.
- [25] Zeng Huang, Yuanlu Xu, Christoph Lassner, Hao Li, and Tony Tung. Arch: Animatable reconstruction of clothed humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3093–3102, 2020.
- [26] Aaron S Jackson, Adrian Bulat, Vasileios Argyriou, and Georgios Tzimiropoulos. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. In *ICCV*, 2017.
- [27] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2018.
- [28] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023.
- [29] Diederik P Kingma and Jimmy Ba. *arXiv preprint arXiv:1412.6980*, 2014.
- [30] Biwen Lei, Jianqiang Ren, Mengyang Feng, Miaomiao Cui, and Xuansong Xie. A hierarchical representation network for accurate and detailed face reconstruction from in-the-wild images, 2023.
- [31] Jiahui Lei, Yufu Wang, Georgios Pavlakos, Lingjie Liu, and Kostas Daniilidis. Gart: Gaussian articulated template models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19876–19887, 2024.
- [32] Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017.
- [33] Jing Lin, Ailing Zeng, Haoqian Wang, Lei Zhang, and Yu Li. One-stage 3d whole-body mesh recovery with component aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21159–21168, 2023.
- [34] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1954–1963, 2021.
- [35] Jiahao Luo, Jing Liu, and James Davis. Splatface: Gaussian splat face reconstruction leveraging an optimizable surface. *arXiv preprint arXiv:2403.18784*, 2024.
- [36] Xiaoxuan Ma, Jiajun Su, Chunyu Wang, Wentao Zhu, and Yizhou Wang. 3d human mesh estimation from virtual markers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 534–543, 2023.
- [37] Riccardo Marin, Enric Corona, and Gerard Pons-Moll. Nicp: neural icp for 3d human registration at scale. In *European Conference on Computer Vision*, pages 265–285. Springer, 2024.

- [38] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7210–7219, 2021.
- [39] Marko Mihajlovic, Aayush Bansal, Michael Zollhoefer, Siyu Tang, and Shunsuke Saito. KeypointNeRF: Generalizing image-based volumetric avatars using relative spatial encoding of keypoints. In *European conference on computer vision*, 2022.
- [40] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020.
- [41] Francesc Moreno-Noguer and Pascal Fua. Stochastic exploration of ambiguities for nonrigid shape recovery. 35(2):463–475, 2013.
- [42] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022.
- [43] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VIII 14*, pages 483–499. Springer, 2016.
- [44] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *CVPR*, 2020.
- [45] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021.
- [46] Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3d face model for pose and illumination invariant face recognition. In *AVSS*, pages 296–301, 2009.
- [47] Rohith Krishnan Pillai, László Attila Jeni, Huiyuan Yang, Zheng Zhang, Lijun Yin, and Jeffrey F Cohn. The 2nd 3d face alignment in the wild challenge (3dfaw-video): Dense reconstruction from video. In *ICCV Workshops*, 2019.
- [48] Shenhan Qian. Vhap: Versatile head alignment with adaptive appearance priors, sep 2024.
- [49] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20299–20309, 2024.
- [50] Amit Raj, Michael Zollhoefer, Tomas Simon, Jason Saragih, Shunsuke Saito, James Hays, and Stephen Lombardi. Pva: Pixel-aligned volumetric avatars. *arXiv preprint arXiv:2101.02697*, 2021.
- [51] Eduard Ramon, Janna Escur, and Xavier Giro-i Nieto. Multi-view 3d face reconstruction in the wild using siamese networks. In *ICCV Workshops*, 2019.
- [52] Eduard Ramon, Gil Triginer, Janna Escur, Albert Pumarola, Jaime Garcia, Xavier Giro-i Nieto, and Francesc Moreno-Noguer. H3d-net: Few-shot high-fidelity 3d head reconstruction. In *ICCV*, pages 5620–5629, 2021.
- [53] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. Generating 3d faces using convolutional mesh autoencoders. In *ECCV*, 2018.
- [54] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. In *3DV*, 2016.
- [55] Elad Richardson, Matan Sela, Roy Or-El, and Ron Kimmel. Learning detailed face reconstruction from a single image. In *CVPR*, 2017.
- [56] Zeyu Ruan, Changqing Zou, Longhai Wu, Gangshan Wu, and Limin Wang. Sadrnet: Self-aligned dual face regression networks for robust 3d dense face alignment and reconstruction. *IEEE Transactions on Image Processing*, 30:5793–5806, 2021.
- [57] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *ICCV*, 2019.

- [58] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 84–93, 2020.
- [59] Akash Sengupta, Ignas Budvytis, and Roberto Cipolla. Synthetic training for accurate 3d human pose and shape estimation in the wild. *arXiv preprint arXiv:2009.10013*, 2020.
- [60] Ruizhi Shao, Hongwen Zhang, He Zhang, Mingjia Chen, Yan-Pei Cao, Tao Yu, and Yebin Liu. Doublefield: Bridging the neural surface and radiance fields for high-fidelity human reconstruction and rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15872–15882, 2022.
- [61] Jiapeng Tang, Davide Davoli, Tobias Kirschstein, Liam Schoneveld, and Matthias Niessner. Gaf: Gaussian avatar reconstruction from monocular videos via multi-view diffusion. *arXiv preprint arXiv:2412.10209*, 2024.
- [62] Ayush Tewari, Michael Zollhofer, Hyeonwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Christian Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *ICCV*, 2017.
- [63] Anh Tuan Tran, Tal Hassner, Iacopo Masi, Eran Paz, Yuval Nirkin, and Gérard G Medioni. Extreme 3d face reconstruction: Seeing through occlusions. In *CVPR*, 2018.
- [64] Anh Tuan Tran, Tal Hassner, Iacopo Masi, and Gérard Medioni. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *CVPR*, 2017.
- [65] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020.
- [66] Lizhen Wang, Zhiyua Chen, Tao Yu, Chenguang Ma, Liang Li, and Yebin Liu. Faceverse: a fine-grained and detail-controllable 3d face morphable model from a hybrid dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [67] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.
- [68] Zidu Wang, Xiangyu Zhu, Tianshuo Zhang, Baiqin Wang, and Zhen Lei. 3d face reconstruction with the geometric guidance of facial part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1672–1682, 2024.
- [69] Fanzi Wu, Linchao Bao, Yajing Chen, Yonggen Ling, Yibing Song, Songnan Li, King Ng Ngan, and Wei Liu. Mvf-net: Multi-view 3d face morphable model regression. In *CVPR*, 2019.
- [70] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.
- [71] Yuelang Xu, Zhaoqi Su, Qingyao Wu, and Yebin Liu. Gphm: Gaussian parametric head model for monocular head avatar reconstruction. *arXiv preprint arXiv:2407.15070*, 2024.
- [72] Yusuke Yoshiyasu. Deformable mesh transformer for 3d human mesh recovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17006–17015, 2023.
- [73] Hongwen Zhang, Yating Tian, Yuxiang Zhang, Mengcheng Li, Liang An, Zhenan Sun, and Yebin Liu. Pymaf-x: Towards well-aligned full-body model regression from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):12287–12303, 2023.
- [74] Mingwu Zheng, Hongyu Yang, Di Huang, and Liming Chen. Imface: A nonlinear 3d morphable face model with implicit neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20343–20352, 2022.
- [75] Mingwu Zheng, Haiyu Zhang, Hongyu Yang, Liming Chen, and Di Huang. Imface++: A sophisticated nonlinear 3d morphable face model with implicit neural representations. *arXiv preprint arXiv:2312.04028*, 2023.
- [76] Mingwu Zheng, Haiyu Zhang, Hongyu Yang, and Di Huang. Neuface: Realistic 3d neural face rendering from multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16868–16877, 2023.



- [77] Xiaozheng Zheng, Chao Wen, Zhaohu Li, Weiyi Zhang, Zhuo Su, Xu Chang, Yang Zhao, Zheng Lv, Xiaoyuan Zhang, Yongjie Zhang, et al. Headgap: Few-shot 3d head avatar via generalizable gaussian priors. *arXiv preprint arXiv:2408.06019*, 2024.
- [78] Yufeng Zheng, Victoria Fernández Abrevaya, Marcel C. Bühler, Xu Chen, Michael J. Black, and Otmar Hilliges. I M Avatar: Implicit morphable head avatars from videos. In *Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [79] Yufeng Zheng, Wang Yifan, Gordon Wetzstein, Michael J. Black, and Otmar Hilliges. Pointavatar: Deformable point-based head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [80] Hao Zhu, Haotian Yang, Longwei Guo, Yidi Zhang, Yanru Wang, Mingkai Huang, Menghua Wu, Qiu Shen, Ruigang Yang, and Xun Cao. Facescape: 3d facial dataset and benchmark for single-view 3d face reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2023.
- [81] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Towards metrical reconstruction of human faces. In *European Conference on Computer Vision (ECCV)*. Springer International Publishing, October 2022.
- [82] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Instant volumetric head avatars. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4574–4584, 2023.

## NeurIPS Paper Checklist

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: We propose an optimization method, GLVD, that archives competitive results for the task of 3D face reconstruction. We encode global awareness through iterative 3D landmarks prediction. All experiments showed in the paper are designed to evaluate the main claims.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The main document has a limitation and future work section. We also add extra experiments violating some of the assumptions that are added in the supplementary material.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [No]

Justification: No theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.

- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide a detailed explanation of the proposed model as well as how it has been trained.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

#### 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Due to legal constraints, we are unable to release the training data. However, we will publish the code along with detailed instructions on how to prepare compatible datasets for training.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).

- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

#### 6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Implementation details are explained in the supplementary material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

#### 7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: While not all prior works report statistical significance, we ensure rigorous evaluation by testing our methods across diverse datasets and varying numbers of input views. We also include qualitative comparisons to visually assess reconstruction quality.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

#### 8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Implementation details as well as training recipe is explained in the supplementary details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

#### 9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

#### 10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We provide societal impact discussion in the appendix.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

#### 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: Access to the models and data is available upon request and subject to the completion of a formal questionnaire and the signing of a legal agreement in advance.

Guidelines:

- The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [\[Yes\]](#)

Justification: We provide proper citation for all prior work on which our approach is based.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

## 13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: We release the full codebase on GitHub, including model implementation, training, and evaluation scripts (to be made public upon publication).

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

## 14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: Although our research involves facial data, we did not collect new human data nor conduct any additional data acquisition for this work.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.



- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

**15. Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [Yes]

Justification: We followed the ethical guidelines established by NeurIPS.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

**16. Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLMs are not involved as any important, original components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

## A Appendix: Guided Learned Vertex Descend

In this appendix, we provide further technical details on

- Experimental setup
- Different Keypoints configuration
- Implementation Details
- Additional qualitative results
- Failure Cases

For video results, including visual comparison to prior work, we refer to our supplementary video. This video includes a demonstration of GLVD for different input images.

### A.1 Experimental Setup

GLVD adapts its reference space based on the number of input views during training and inference. For single-view 3D reconstruction, it operates in the camera coordinate frame, eliminating the need for camera parameter estimation at test time. In the multi-view setting, we canonicalize the 3D reconstruction and train a camera pose estimator on the same dataset used for training GLVD to enable prediction at inference time.

To ensure a fair comparison, PIFU, JIFF, LVD, and SIRA++ are trained using the same data used to train GLVD. While PIFU, LVD, and JIFF were initially designed for full-body reconstruction, we modified their training to align with the data used by GLVD. To enhance robustness during training, we applied data augmentation techniques, including adjustments to brightness, contrast, hue, and saturation, as well as image jittering, blurring, and zooming. These augmentations are applied to the input images used for feature extraction. Additionally, we employed scene symmetrization, doubling the number of training scenes.

### A.2 Keypoints configuration

GLVD requires only RGB images as input to predict the 3D surface. Internally, it operates by estimating 3D keypoints. Figure 8 presents visualizations of four proposed landmark subsets. The method is designed to allow a flexible selection of landmark configurations. In our experiments, we use template vertices registered to the training scenes. To adapt the method to other parametric models, such as FLAME or SMPL-X, joints can be selected as keypoints.

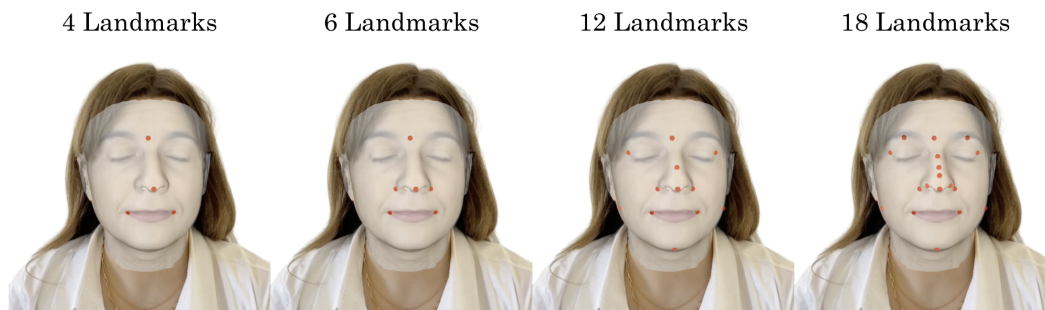


Figure 8: **Qualitative visualization of four keypoints configurations.** Images are from H3DS 2.0. At the last iteration step, we show the predicted template and the 3D keypoints in red.

### A.3 Implementation Details

The function  $f_v(\cdot)$  is a stacked hourglass network [43] composed of four stacks using group normalization [70]. Feature embeddings have a spatial resolution of  $64 \times 64$ , with each containing 64 channels. As a result, each query point is represented by stacking four feature vectors of size  $64 \times 4 = 256$ . We pre-train  $f_v(\cdot)$  to predict the signed distance function (SDF) values using the same training data. We set the clipping directional factor to 0.1, being the scene normalized in the centered cube of size 2.

The function  $f_k(\cdot)$  is implemented by a combination of a facial keypoint heatmap estimator HRNet [65] and a single-stack hourglass network [43]. During training, we keep the weights of the HRNet frozen. To generate  $\mathbf{F}_k$ , we extract the first feature map computed with  $f_v(\cdot)$  and then concatenate it with the heatmaps predicted by HRNet. The combination of the features and the image  $\mathbf{I}$  and the mask  $\mathbf{M}$  is then fed to the hourglass network.

1 view



Figure 9: **Qualitative results for one input image.** Images are from 3DFAW (top), and H3DS 2.0 (bottom). At the last iteration step, we show the predicted template and the 3D keypoints in red.

Feature embeddings have a spatial resolution of  $64 \times 64$ , with each containing 64 channels. We use a 0.1 clipping factor.

Function  $g_v(\cdot)$  produces an output tensor of dimension  $N = 7225 \times 3$ . Given an input surface of size  $7225 \times 3$ , it outputs a tensor of shape  $7225 \times 7225 \times 3$ . Per-vertex displacements ( $7225 \times 3$ ) are extracted from the diagonal and applied to update vertex positions.

GLVD works for different numbers of input images. When several images are used, we adopt a mean aggregation layer among features extracted from a multi-view feature encoder. In particular, we follow a single view forward pass independently of the number of input images until the second layer of the  $g_v(\cdot)$  and  $g_k(\cdot)$ , where we apply a mean operation to aggregate multiview features.

Function  $g_k(\cdot)$  produces an output tensor of dimension  $K = 18 \times 3$ . Given an input surface of size  $18 \times 3$ , it outputs a tensor of shape  $18 \times 18 \times 3$ . Per-vertex displacements ( $18 \times 3$ ) are extracted from the diagonal and applied to update keypoint positions. Both  $g_v(\cdot)$  and  $g_k(\cdot)$  are implemented as a 3-Layer MLP with ReLU activation and weight normalization.

All networks are trained end-to-end using GPU-accelerated hardware (RTX 4090). We use a batch size of 4 and an initial learning rate of 0.001 for 50 epochs, followed by 200 additional epochs with linear learning rate decay. For each scene, we sample 1400 vertices as query points. It takes between 1.5 to 6 days of training, depending on the configuration. We set  $\lambda_1 = \lambda_2 = 0.5$ . Optimization is performed using Adam [29] with  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$ . The total number of parameters is detailed in Table 4.

Table 4: **Total number of parameters.**

|                          | Parameters | Ratio |
|--------------------------|------------|-------|
| $f_v(\cdot)$             | 14.08 M    | 54.9% |
| $g_v(\cdot)$             | 11.57 M    | 45.1% |
| Total                    | 25.64 M    | 100%  |
| $f_k(\cdot)$ (HRNet)     | 9.65 M     | 67.2% |
| $f_s(\cdot)$ (Hourglass) | 4.28 M     | 29.8% |
| $g_k(\cdot)$             | 0.43 M     | 2.96% |
| Total                    | 14.36 M    | 100%  |

#### A.4 Additional results

We provide qualitative results for subjects from 3DFAW and H3DS 2.0 using a single input view in Figure 9, and for the multi-view setting in Figure 10. We also show qualitative results in the wild CelebA-HQ dataset in Figure 7.

Figure 11 reports the reconstruction error on 3DFAW subjects under varying numbers of update steps and clipping factor. At test time, the magnitude of the predicted displacement vector is clipped within the range  $[0.05, 0.5]$ . Results indicate that the number of update steps has a limited impact on reconstruction accuracy, which is strongly influenced by the clipping value used during training. This parameter controls the trade-off between accuracy and computational cost. Our experiments achieve the best performance with 10 steps and a clipping factor of 0.1.

We conducted an ablation study evaluating sequential vs parallel update strategies for the vertex and keypoints refinement modules on the H3DS 2.0 and 3DFAW datasets (Table 5). The results demonstrate that the iterative parallel update scheme yields consistently superior performance compared to the sequential alternative, although the improvement is minor.

3 views



Figure 10: **Qualitative results for three input images.** Images are from 3DFAW. At the last iteration step, we show the predicted template and the 3D keypoints in red.

Table 5: Ablation of sequential versus iterative (parallel) update strategies on H3DS 2.0 and 3DFAW.

| <b>H3DS 2.0 Dataset</b> | 1 view ↓    | 3 views ↓   | 4 views ↓   | 6 views ↓   |
|-------------------------|-------------|-------------|-------------|-------------|
| LVD                     | 1.55        | 1.49        | 1.44        | 1.42        |
| LVD pre-trained SDF     | 1.39        | 1.45        | 1.39        | 1.37        |
| GLVD Sequential         | 1.38        | 1.36        | <b>1.34</b> | 1.35        |
| GLVD Iterative          | <b>1.36</b> | <b>1.33</b> | <b>1.34</b> | <b>1.34</b> |
| <b>3DFAW Dataset</b>    | 1 view ↓    | 3 views ↓   | —           |             |
| LVD                     | 1.65        | 1.52        | —           |             |
| LVD pre-trained SDF     | 1.58        | 1.26        | —           |             |
| GLVD Sequential         | 1.29        | 1.23        | —           |             |
| GLVD Iterative          | <b>1.25</b> | <b>1.22</b> | —           |             |

We also demonstrate that pre-training the feature encoder on a 3D reconstruction task, where it is trained to predict signed distance functions (SDFs), leads to faster convergence and improved performance. In this setting, (1) we pre-train the feature encoder on the 3D reconstruction task. We represent the surface  $\mathcal{S}$  as the zero-level set of a signed distance function  $f^{\text{sdf}}: (\mathbf{x}, I) \rightarrow s$ , such that  $\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 \mid f^{\text{sdf}}(\mathbf{x}, I) = 0\}$ . Our goal is to estimate  $f^{\text{sdf}}$  through a composition of a feature encoder and a decoder network. The resulting feature encoder is then used within GLVD. To train on the SDF task, we use non-watertight scans from the same training dataset and minimize  $\mathcal{L}_{\text{Surf}}^{(i)}$  on surface points  $N_s$  and  $\mathcal{L}_{\text{Eik}}^{(i)}$  throughout the volume  $N_v$ :

$$\mathcal{L}_{\text{Surf}} = \frac{1}{N_s} \sum_{i=1}^{N_s} |f^{\text{sdf}}(\mathbf{x}_i, I)|, \quad (4)$$

$$\mathcal{L}_{\text{Eik}} = \frac{1}{N_v} \sum_{i=1}^{N_v} \left( \left\| \nabla_{\mathbf{x}} f^{\text{sdf}}(\mathbf{x}_i, I) \right\|_2 - 1 \right)^2. \quad (5)$$

The loss is averaged over the batch. We compare in Table 5 the impact of pretraining. We evaluate LVD with and without SDF-based pretraining of the feature encoder. Results show that this pretraining is crucial for achieving strong performance in both LVD and GLVD.

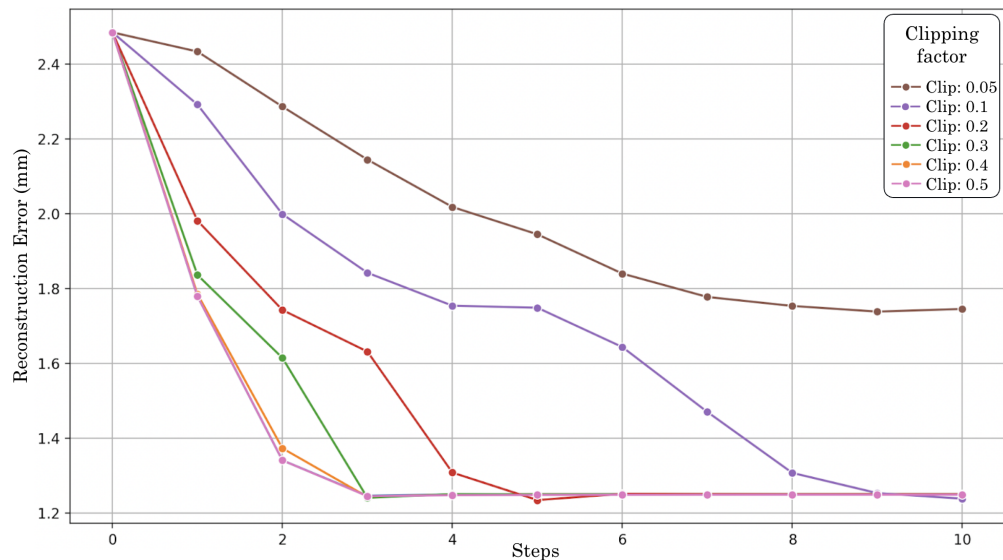


Figure 11: **Reconstruction Error from a Single Input Image.** Results report the mean Chamfer Distance on the 3DFAW dataset.

### A.5 Failure Cases

We present failure cases of GLVD under varying numbers of input views. Figure 12 illustrates qualitative results in extreme scenarios, while Table 6 reports quantitative performance for different viewing angles in the single-view setting. The best performance is observed with front-facing input images. As the viewing angle increases, performance degrades significantly, primarily due to inaccuracies in landmark estimation under self-occlusion conditions.

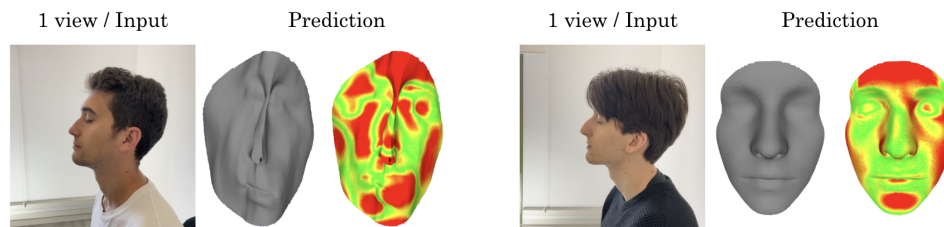


Figure 12: **Reconstruction Error from a Single Input Image.** Results report failure cases.

### A.6 Social Impact

GLVD advances 3D face modeling with high accuracy, enabling applications in graphics, AR/VR, and biometrics. However, it also raises concerns about privacy, surveillance, and identity misuse. High-fidelity face reconstruction can be used without consent or for impersonation, contributing to deepfake risks. To mitigate these issues, responsible deployment, fairness audits, and privacy safeguards are essential. While GLVD is a technical step forward, its societal implications must be carefully considered.

Table 6: **Quantitative Results for a Single Input Image at varying input angles.** Chamfer Distance is reported in millimeters (mm) on the H3DS 2.0 dataset.

|      | 1 view 0° | 1 view 45° | 1 view 90° |
|------|-----------|------------|------------|
| GLVD | 1.31      | 2.07       | 2.11       |



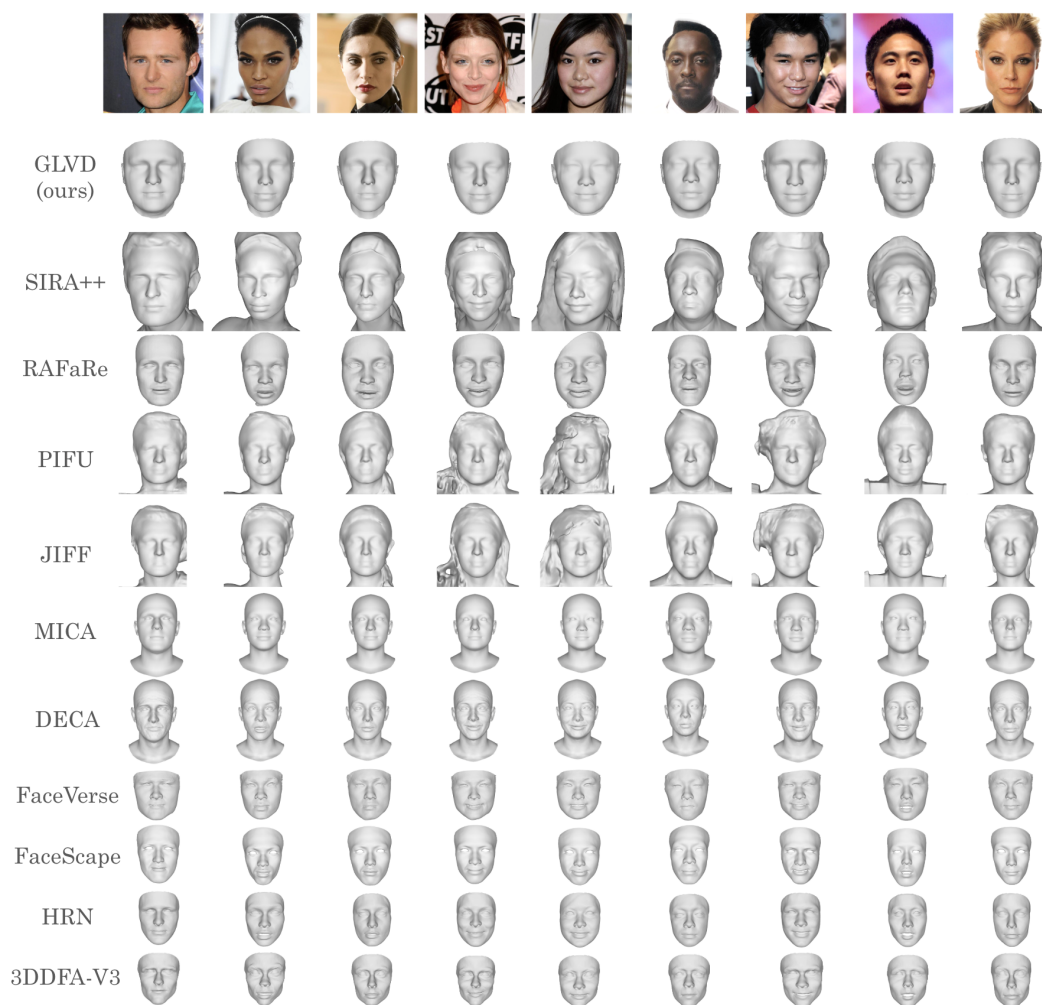


Figure 13: Qualitative results on the CelebA-HQ dataset for a single input image.