Topologically Consistent Multi-view 3D Head Reconstruction via Coarse-Guided Layered Surface Sampling

TIMO BOLKART, Google, Switzerland
DAOYE WANG, Google, Switzerland
PRASHANTH CHANDRAN, Google, Switzerland

Fig. 1. **Feed-forward registration.** Given calibrated multi-view images (left; 5 of 13 views shown), SHELLS reconstructs 3D meshes in dense semantic correspondence in 0.08 seconds. Overlaid reconstructions demonstrate precise geometric alignment across diverse subjects and expressions (middle & right). SHELLS generalizes from synthetic training to real multi-view captures, enabling efficient, high-quality registration of large-scale datasets.

We present SHELLS (Semantic Head Estimation via Layered Local Sampling), an efficient feed-forward framework for 3D head reconstruction in dense semantic correspondence from multi-view images. Existing methods typically refine vertices independently via localized feature volumes. This approach couples memory-intensive feature sampling to mesh resolution, which limits scalability for dense topologies ($\geq 10k$ vertices) and introduces surface noise. In contrast, SHELLS decouples feature extraction from mesh resolution via a hierarchical sampling strategy. We extract multi-view features using a DinoV2 backbone with LoRA adaptation, projectively sample a sparse global feature cloud, and predict an intermediate coarse mesh. This coarse prior guides the construction of layered, surface-aware sampling shells that serve as a discrete search space for the final reconstruction. SHELLS maintains surface consistency while using 88% less inference GPU memory (~ 2.4GB vs. ~ 20GB) than volumetric baselines. It reduces median registration error by 21% to 29% with a 3.5× inference speedup (0.08s vs. 0.29s) for 18k-vertex meshes. Notably, our model is trained exclusively on synthetic data yet generalizes effectively to real-world captures, eliminating the need for the costly, pre-registered multi-view datasets common in prior work.

Authors' addresses: Timo Bolkart, Google, Switzerland, tbolkart@google.com; Daoye Wang, Google, Switzerland, daoye@google.com; Prashanth Chandran, Google, Switzerland, prchandran@google.com.

SIGGRAPH Conference Papers '26, July 19–23, 2026, Los Angeles, CA, USA

ACM ISBN 979-8-4007-2554-8/2026/07
https://doi.org/10.1145/3799902.3811201

CCS Concepts: • **Computing methodologies** → **Mesh geometry models**; **Reconstruction**; *Motion capture*.

Additional Key Words and Phrases: Registration

ACM Reference Format:
Timo Bolkart, Daoye Wang, and Prashanth Chandran. 2026. Topologically Consistent Multi-view 3D Head Reconstruction via Coarse-Guided Layered Surface Sampling. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '26), July 19–23, 2026, Los Angeles, CA, USA*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3799902.3811201

1 INTRODUCTION

High-fidelity 3D head reconstruction in dense semantic correspondence is a fundamental requirement for building realistic digital humans [Egger et al. 2020; Zielonka et al. 2026]. Traditional pipelines register unstructured multi-view stereo (MVS) scans to a unified topology [Beeler et al. 2011; Egger et al. 2020]. However, MVS often produces noise and holes in specular regions, necessitating labor-intensive manual cleanup. It also tends to over-smooth concavities like the ears or nostrils and produces unrealistic geometry in hair regions. The subsequent registration step is computationally exhaustive, often requiring several minutes to hours of optimization per frame, and labor-intense as it requires manual clean up and hyperparameter tuning to balance surface fidelity against robustness to scan artifacts [Alexander et al. 2009; Seymour et al. 2017].

To bypass these bottlenecks, recent learning-based frameworks [Bolkart et al. 2023; Filntisis et al. 2026; Li et al. 2024b, 2021; Liu et al.

2022] move toward direct surface regressions from calibrated multi-view images. By eliminating the MVS and registration steps, these methods achieve near-interactive reconstruction speeds, demonstrating great potential for the fully automated processing of large-scale datasets. However, while these methods offer superior scalability, they still lack the geometric details of classical registration and face significant architectural bottlenecks. State-of-the-art methods [Bolkart et al. 2023; Li et al. 2024b, 2021] rely on memory-intense global and per-point feature volumes, which constrain the output resolution to low vertex counts ($\sim 3k$ to $\sim 5k$).

To overcome these challenges, we present SHELLS, a transformer-based framework that predicts higher-resolution 3D heads in dense semantic correspondence from calibrated multi-view images. Our approach builds upon ToFu's [Li et al. 2021] projective multi-view feature sampling strategy, which naturally integrates known camera parameters. However, we replace their memory-intense dense feature volumes with a coarse-guided hierarchical feature sampling strategy. Specifically, SHELLS first employs a sparse global sampling graph to estimate an intermediate low-resolution mesh, which then guides the placement of layered sampling shells displaced along the surface normals. This surface-aware strategy ensures that feature sampling is restricted to the proximity of the target geometry, reducing the sampling of irrelevant features, and effectively decoupling memory consumption from the final mesh resolution.

Furthermore, existing methods [Bolkart et al. 2023; Li et al. 2024b, 2021] lack a global geometric understanding because they refine vertex positions independently, leading to mesh artifacts in regions occluded by hair or clothing. Our transformer-based prediction model instead processes the sampled features holistically, which improves robustness in the presence of severe occlusions. Finally, SHELLS eliminates the need for costly real-world data processing. While prior work requires paired capture data with registration meshes per frame [Li et al. 2021; Liu et al. 2022] or raw scans for joint optimization [Bolkart et al. 2023; Li et al. 2024b], we demonstrate that training SHELLS exclusively on synthetic multi-view data is sufficient to generalize to real-world captures (see Fig. 1).

In summary, we introduce a hierarchical shell-based sampling strategy that reduces GPU memory requirements by 70% for training ($\sim$ 20GB vs. $\sim$ 65GB) and 88% for inference ($\sim$ 2.4GB vs. $\sim$ 20GB) compared to volumetric baselines. Our transformer-based architecture holistically predicts 3D faces in dense semantic correspondence, with a $21\% - 29\%$ lower median registration error on real capture test data and synthetic test data. We demonstrate that training on synthetic data is sufficient to generalize to real multi-view captures.

2 RELATED WORK

Optimization-based registration. Traditional face registration often deforms a template mesh non-rigidly to multi-view stereo (MVS) scans [Egger et al. 2020]. These techniques have matured from neutral expressions [Blanz and Vetter 1999] to high-quality performance sequences [Beeler et al. 2011] and the automatic processing of thousands of identities [Booth et al. 2016] and sequences [Li et al. 2017]. However, MVS reliance introduces noise and holes that necessitate computationally expensive, manually tuned regularization. Direct methods maximize consistency via differentiable rasterization [Qian 2024; Qian et al. 2024], optical flow [Fyffe et al. 2017], or they integrate learnable priors [Bai et al. 2020], neural volume rendering [Wang et al. 2026], or surface-aligned Gaussians [Li et al. 2024a]. While these improve fidelity, their iterative nature remains a bottleneck, requiring minutes to hours per frame. In contrast, SHELLS provides dense correspondence in an efficient, feed-forward manner.

Feed-forward mesh prediction. Feed-forward registration accelerates inference through direct mesh prediction. ToFu [Li et al. 2021] pioneered volumetric feature sampling for sub-second, consistent multi-view face reconstruction. Building on ToFu, TEMPEH [Bolkart et al. 2023] adds head localization and direct scan supervision, GRAPE [Li et al. 2024b] incorporates visual hull initialization, and MOCHI [Filntisis et al. 2026] eliminates registration supervision. However, these volumetric models rely on memory-heavy sampling grids that scale poorly to dense topologies. SHELLS instead employs sparse hierarchical sampling and holistic reconstruction to improve memory efficiency and surface coherence.

Similar to SHELLS, POEM [Yang et al. 2023, 2025] utilizes sparse sampling and a transformer but iteratively refines hand meshes. In contrast, SHELLS is non-iterative and uses joint attention to predict the output in a single pass via an attention-weighted sum of sampling coordinates. While POEM targets low-resolution MANO [Romero et al. 2017] hands (778 vertices), SHELLS reconstructs significantly denser head topologies (18k vertices).

Finally, unlike unconstrained methods [Giebenhain et al. 2025; Vhavle et al. 2025], SHELLS leverages calibrated cameras for metrical accuracy and bypasses the expressiveness limits of linear 3DMMs.

Unstructured points prediction. Learning-based multi-view reconstruction significantly improves 3D fidelity [Gu et al. 2020; Im et al. 2019; Kar et al. 2017; Qiu et al. 2024; Sitzmann et al. 2019; Yao et al. 2018]. Transformer models like DUSt3R [Wang et al. 2024], MASt3R [Leroy et al. 2024], and VGGT [Wang et al. 2025] predict point maps without explicit calibration but produce unstructured outputs lacking semantic labels or consistent topology. While VGGT and St4RTrack [Feng et al. 2025] incorporate tracking, correspondence remains restricted to the sequence level. Consequently, performing reconstruction across different subjects fails to provide inter-subject correspondence. In contrast, SHELLS regresses meshes in a fixed mesh topology to ensure dense semantic correspondence across both time and identities.

Synthetic data training. Synthetic head datasets support diverse applications, from 2D landmark prediction [Wood et al. 2022] and face parsing [Wood et al. 2021] to scan segmentation [Chen et al. 2025] and neural avatar construction [Saunders et al. 2025; Zielonka et al. 2025]. This versatility motivates our use of synthetic data for the multi-view prediction task.

3 METHOD

Given n_i time-synchronized input images $\{\mathcal{I}_k \in \mathbb{R}^{h_i \times w_i \times 3}\}_{k=1}^{n_i}$, each of height h_i and width w_i and their corresponding camera parameters $\{C_k\}_{k=1}^{n_i}$ (extrinsics, intrinsics, and lens distortions), SHELLS infers a 3D head mesh $M_f := (\mathbf{V}_f, \mathbf{T})$ with vertices $\mathbf{V}_f \in \mathbb{R}^{n_v \times 3}$ in a fixed mesh topology $\mathbf{T}$. The fixed mesh topology with a constant number of vertices n_v ensures that all reconstructions are in dense

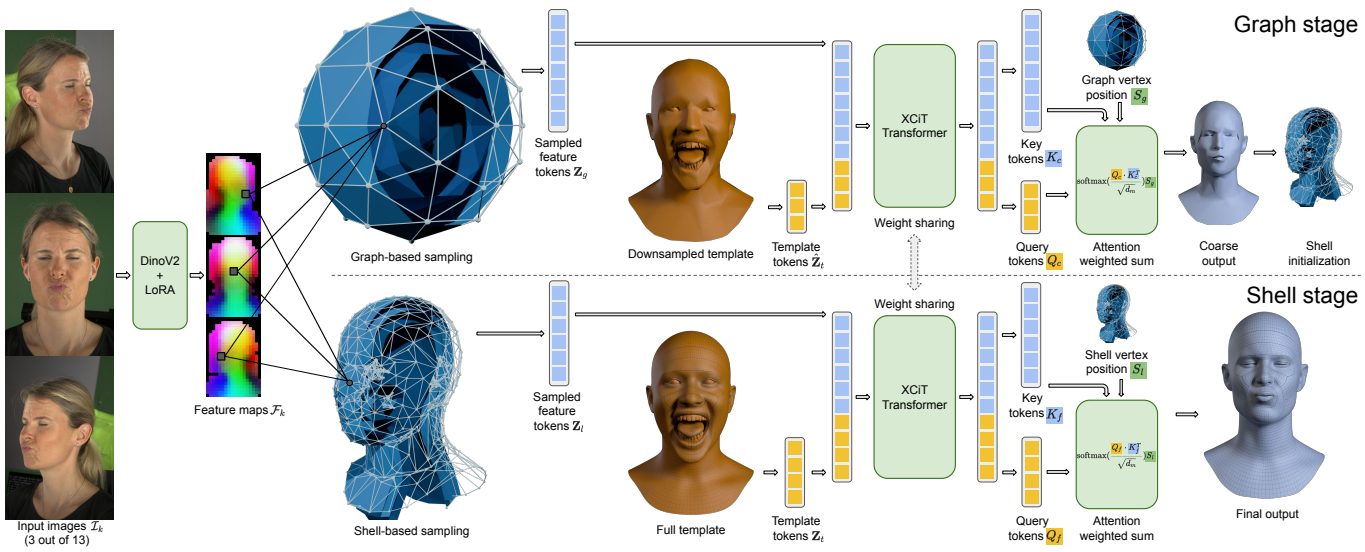

Fig. 2. **Overview of SHELLS**. A shared DinoV2 backbone with LoRA adaptation extracts per-view feature maps from the input images (left). The graph stage (top) projectively samples features for a sparse graph and processes them alongside a downsampled tokenized template using an XCiT-based transformer. From the transformer output, a coarse mesh is regressed as an attention-weighted sum over the sampling graph coordinates. This coarse prediction is displaced along its normals to construct sampling shells for surface-aware feature sampling. Finally, the shared transformer aggregates these shell-based features with a full-resolution tokenized template to predict the high-fidelity mesh as an attention-weighted sum of dynamic shell coordinates (bottom).

semantic correspondence. As shown in Fig. 2, SHELLS consists of two stages, trained end-to-end. The first stage predicts an intermediate coarse mesh $\hat{M}_c := (\hat{\mathbf{V}}_c, \hat{\mathbf{T}})$ with n_c vertices $\hat{\mathbf{V}}_c \in \mathbb{R}^{n_c \times 3}$, which guides feature sampling for the subsequent prediction stage. The second stage then builds layers of sampling shells around $\hat{M}_c$ and predicts the final mesh M_f from the sampled multi-view features.

Feature extraction. Each image $\mathcal{I}_k$ is processed by a shared feature extraction network $F_{\text{img}}(.)$ using a frozen DinoV2 [Oquab et al. 2023] backbone to extract 2D feature maps $F_{\text{img}}(\mathcal{I}_k) \rightarrow \mathcal{F}_k \in \mathbb{R}^{h_f \times w_f \times d_f}$. We incorporate trainable LoRA [Hu et al. 2022] layers with rank r as residuals in each linear DinoV2 layer to adapt the backbone to the reconstruction task. The spatial dimensions are downsampled by a factor of 14, such that $h_f = h_i/14$ and $w_f = w_i/14$.

3.1 Graph-based coarse prediction

Feature sampling. To localize the face within the capture volume when its 3D position is unknown, we employ a sparse sampling strategy. Instead of utilizing a dense 3D grid [Bolkart et al. 2023; Li et al. 2021], we define a sparse point cloud $\mathbf{S}_g \in \mathbb{R}^{n_g \times 3}$ comprising vertices of equidistant concentric spheres, each approximated by a twice-subdivided icosahedron. For each input image $\{\mathcal{I}_k\}_{k=1}^{n_i}$, every sampling point $\mathbf{s} \in \mathbb{R}^3$ in $\mathbf{S}_g$ is perspectively projected into the image plane using the camera projection $\Pi_k : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ with the given camera parameters C_k. Per-view feature vectors $\mathbf{f}_k \in \mathbb{R}^{d_f}$ are extracted from the feature maps $\mathcal{F}_k$ at the projected 2D locations via bilinear sampling and subsequently fused across all views.

Feature fusion. Following ToFu [Li et al. 2021], we fuse these per-view vectors by computing the element-wise mean $\boldsymbol{\mu} \in \mathbb{R}^{d_f}$ and variance $\boldsymbol{\sigma}^2 \in \mathbb{R}^{d_f}$. The resulting fused feature vector $\mathbf{f} = [\boldsymbol{\mu}; \boldsymbol{\sigma}^2] \in \mathbb{R}^{2d_f}$ is the concatenation of $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$. Performing the feature sampling for all sampling points of $\mathbf{S}_g$ yields the global feature point cloud $\mathbf{F}_g \in \mathbb{R}^{n_g \times 2d_f}$.

Transformer-based mesh prediction. We utilize a transformer-based architecture to predict $\hat{\mathbf{V}}_c$ from the sparse feature point cloud. We define a fixed template mesh $M_t := (\mathbf{V}_t, \mathbf{T})$, with vertices $\mathbf{V}_t \in \mathbb{R}^{n_v \times 3}$, which establishes the mesh topology of the output prediction. For computational efficiency, the template mesh is downsampled to a lower resolution $\hat{M}_t = (\hat{\mathbf{V}}_t, \hat{\mathbf{T}})$ with n_c vertices using iterative surface simplification [Garland and Heckbert 1997]. Similar to ToFu's [Li et al. 2021] coarse stage, this makes the intermediate prediction independent of the final mesh resolution.

Following Ranjan et al. [2018], we derive a downsampling matrix $\mathbf{D} \in \{0,1\}^{n_v \times n_c}$ and an upsampling matrix $\mathbf{U} \in \mathbb{R}^{n_c \times n_v}$ based on barycentric interpolation. The coarse template vertices $\hat{\mathbf{V}}_t := \mathbf{D}\mathbf{V}_t \in \mathbb{R}^{n_c \times 3}$ are processed by an MLP to generate tokens $\hat{\mathbf{Z}}_t \in \mathbb{R}^{n_c \times d_m}$.

Simultaneously, the coordinates of the fixed sampling graph $\mathbf{S}_g$ are processed through a separate MLP to form a geometric positional embedding. This embedding is element-wise added to the multi-view features $\mathbf{F}_g$ and projected to d_m via a linear layer to produce feature tokens $\mathbf{Z}_g \in \mathbb{R}^{n_g \times d_m}$.

The joint set of tokens $\mathbf{Z}_c = [\hat{\mathbf{Z}}_t; \mathbf{Z}_g] \in \mathbb{R}^{(n_c+n_g) \times d_m}$ is processed by a sequence of transformer layers F_{pred}. To circumvent the quadratic memory complexity of standard self-attention, each layer adopts the Cross-Covariance Image Transformer (XCiT) architecture [Ali et al. 2021], following its successful application to 3D face modeling [Chandran et al. 2022]. Utilizing a parallel block design for enhanced efficiency, layer-normalized input tokens are processed

concurrently by a cross-covariance attention (XCA) layer and a feed-forward network (FFN). By computing attention across the feature dimension rather than the token count, the XCA layer significantly reduces computational complexity when processing large point sets. The outputs of these XCA and FFN branches are summed and integrated with the original input via a residual connection.

The transformer output $F_{\text{pred}}(\mathbf{Z}_c) \rightarrow [\mathbf{Q}_c; \mathbf{K}_c]$ is decomposed into query tokens $\mathbf{Q}_c \in \mathbb{R}^{n_c \times d_m}$ and $\mathbf{K}_c \in \mathbb{R}^{n_g \times d_m}$ (after separate linear projections), representing the refined template and feature tokens, respectively. The coarse vertices $\hat{\mathbf{V}}_c$ are then regressed as an attention-weighted sum of the sampling graph coordinates $\mathbf{S}_g$: $\hat{\mathbf{V}}_c = \text{Softmax}(\mathbf{Q}_c \mathbf{K}_c^T / \sqrt{d_m}) \mathbf{S}_g$.

3.2 Shell-based prediction

Feature sampling. The coarse mesh $\hat{M}_c$ is used to construct a sampling graph that layers the estimated surface. Specifically, we compute surface layers displaced along the vertex normal directions. Given the coarse predicted mesh $\hat{M}_c$, we compute the vertex normals $\hat{\mathbf{N}}_c \in \mathbb{R}^{n_c \times 3}$. We build a shell-based sampling point cloud by stacking the displaced vertices $\mathbf{S}_l = [\hat{\mathbf{V}}_c; \hat{\mathbf{V}}_c + d_l \hat{\mathbf{N}}_c; \hat{\mathbf{V}}_c - d_l \hat{\mathbf{N}}_c] \in \mathbb{R}^{3n_c \times 3}$, where d_l is the layer displacement distance. Again, each sampling point $\mathbf{s} \in \mathbb{R}^3$ of $\mathbf{S}_l$ is projected into each feature map $\{\mathcal{F}_k\}_{k=1}^{n_i}$ to get per-view feature vectors $\mathbf{f}_k \in \mathbb{R}^{d_f}$ via bilinear sampling.

Surface-aware feature fusion: To fuse these per-view features, we adopt the visibility-aware aggregation strategy of TEMPEH [Bolkart et al. 2023]. Unlike the graph stage which treats all views equally, this surface-aware fusion weights each view based on the local surface geometry of $\hat{M}_c$. For a sampling point $\mathbf{s}$ of $\mathbf{S}_l$ associated with a vertex $\mathbf{v}$ of $\hat{M}_c$, we compute a view-dependent weight $\phi_k = \text{Softplus}(\delta_k \cdot \cos\theta_k)$. Here, $\delta_k \in \{0, 1\}$ denotes the visibility of $\mathbf{v}$ from the k-th camera, determined via a depth-buffer check on the intermediate mesh $\hat{M}_c$. The term $\cos\theta_k = \mathbf{n}^T \mathbf{d}_k$ is the dot product between the vertex normal $\mathbf{n}$ of $\mathbf{v}$ and the viewing direction $\mathbf{d}_k = (\mathbf{c}_k - \mathbf{v})/\|(\mathbf{c}_k - \mathbf{v})\|$, where $\mathbf{c}_k$ is the k-th camera center. The Softplus function ensures positive weights and maintains non-zero gradients across all views. Using these weights, we compute the weighted mean $\boldsymbol{\mu}$ and weighted variance $\boldsymbol{\sigma}^2$ across all n_i views, which are concatenated to form the fused feature vector $\mathbf{f} = [\boldsymbol{\mu}; \boldsymbol{\sigma}^2]$. Performing this feature fusion for all points in $\mathbf{S}_l$ yields the shell feature point cloud $\mathbf{F}_l \in \mathbb{R}^{3n_c \times 2d_f}$.

Transformer-based mesh prediction. In the second stage, we predict the full-resolution vertices $\mathbf{V}_f \in \mathbb{R}^{n_v \times 3}$ by attending over the shell-based features. The output resolution is established by the template vertices $\mathbf{V}_t \in \mathbb{R}^{n_v \times 3}$, which are processed by an MLP to generate template tokens $\mathbf{Z}_t \in \mathbb{R}^{n_v \times d_m}$.

To obtain consistent semantic embeddings of the shell point cloud, we define fixed template shells $\mathbf{S}_t \in \mathbb{R}^{3n_c \times 3}$ by displacing the downsampled template vertices $\hat{\mathbf{V}}_t$ along their vertex normals. Unlike the dynamic sampling shells $\mathbf{S}_l$, which depend on the coarse prediction, this template shell is static and serves to encode the relative spatial relationships of the sampling points. The coordinates of $\mathbf{S}_t$ are similarly processed through an MLP to form geometric positional embeddings. These embeddings are element-wise added to the fused multi-view features $\mathbf{F}_l$ and projected to d_m to produce shell feature tokens $\mathbf{Z}_l \in \mathbb{R}^{3n_c \times d_m}$.

Identical to the graph stage, the joint set of tokens $\mathbf{Z}_f = [\mathbf{Z}_t; \mathbf{Z}_l] \in \mathbb{R}^{(n_v + 3n_c) \times d_m}$ is processed by the transformer model F_{pred}, which is shared across the coarse and final stage. The transformer output is partitioned and passed through separate linear layers, shared with the graph stage, to obtain query tokens $\mathbf{Q}_f \in \mathbb{R}^{n_v \times d_m}$ and key tokens $\mathbf{K}_f \in \mathbb{R}^{3n_c \times d_m}$. The final vertices $\mathbf{V}_f$ are then regressed as the attention-weighted sum of the dynamic sampling shell coordinates $\mathbf{S}_l$: $\mathbf{V}_f = \text{Softmax}(\mathbf{Q}_f \mathbf{K}_f^T / \sqrt{d_m}) \mathbf{S}_l$.

While ToFu [Li et al. 2021] and TEMPEH [Bolkart et al. 2023] utilize 512 (8^3) volumetric samples per vertex (totaling 9×10^6) for independent local refinement, SHELLS regresses vertex positions as a weighted combination of only 9,000 layered shell points. By reducing the total number of 3D sampling locations across both stages to 11,592, we drastically minimize the sampling operations. This significantly lowers GPU memory overhead and accelerates both training and inference.

3.3 Loss functions

We train SHELLS end-to-end using a combination of vertex-to-vertex and point-to-plane distances between the reconstructed vertices and the ground-truth vertices. To supervise the first stage, the predicted coarse vertices $\hat{\mathbf{V}}_c$ are upsampled to the full mesh resolution using the barycentric upsampling matrix $\mathbf{V}_c = \mathbf{U}\hat{\mathbf{V}}_c \in \mathbb{R}^{n_v \times 3}$. We define the displacement matrices for the upsampled coarse prediction $\Delta\mathbf{V}_c = \mathbf{V}_c - \mathbf{V}_{\text{gt}}$ and the final predictions $\Delta\mathbf{V}_f = \mathbf{V}_f - \mathbf{V}_{\text{gt}}$.

Vertex-to-vertex (V2V) loss. The V2V loss minimizes the Euclidean distances between predicted and ground truth vertex positions. To allow for spatially varying importance across the mesh surface (e.g., to prioritize facial features), we introduce a diagonal weight matrix $\boldsymbol{\Omega} = \text{diag}(\omega_1, \ldots, \omega_{n_v})$, where ω_i denotes the individual weight of the i-th vertex. The loss is defined as:

$$\mathcal{L}_{\text{v2v}} = \lambda_c \left\|\boldsymbol{\Omega}\Delta\mathbf{V}_c\right\|_F^2 + \lambda_f \left\|\boldsymbol{\Omega}\Delta\mathbf{V}_f\right\|_F^2, \tag{1}$$

where λ_c and λ_f balance the coarse and final prediction stages.

Vertex-to-plane loss: To improve surface alignment, we incorporate a vertex-to-plane loss, formulated as:

$$\mathcal{L}_{\text{v2p}} = \lambda_c \left\|\boldsymbol{\Omega}(\Delta\mathbf{V}_c \odot \mathbf{N}_{\text{gt}})\mathbf{1}_3\right\|_2^2 + \lambda_f \left\|\boldsymbol{\Omega}(\Delta\mathbf{V}_f \odot \mathbf{N}_{\text{gt}})\mathbf{1}_3\right\|_2^2, \tag{2}$$

where $\odot$ denotes the Hadamard product, and $\mathbf{1}_3 \in \mathbb{R}^3$ is a column vector of ones used to perform row-wise summation, effectively computing the dot product between displacement vectors and ground-truth vertex normals $\mathbf{N}_{\text{gt}} \in \mathbb{R}^{n_v \times 3}$. This loss penalizes only the displacement component orthogonal to the target surface, which allows vertices to distribute along the geometry by avoiding penalties for "sliding" along the local tangent planes.

Total loss. The model is trained by minimizing:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{v2v}}\mathcal{L}_{\text{v2v}} + \lambda_{\text{v2p}}\mathcal{L}_{\text{v2p}}, \tag{3}$$

with weights λ_{v2v} and λ_{v2p} of the individual losses.

Optimization-based frameworks [Egger et al. 2020] and TEMPEH [Bolkart et al. 2023] typically minimize point-to-surface (P2S) distances, which permit tangential sliding and mesh distortions that

necessitate explicit regularization. In contrast, SHELLS leverages a vertex-to-vertex (V2V) loss based on dense semantic correspondence. This provides strong implicit regularity, maintaining surface integrity without requiring additional regularization.

4 IMPLEMENTATION DETAILS

Fig. 3. **Synthetic dataset.** (Left) A single subject rendered from 13 camera views simulating a multi-view capture environment. (Right) Random samples demonstrating the diversity in identities and expressions, augmented with randomized backgrounds and assets including clothing and hair.

Synthetic dataset. We adopt the procedural approach of Wood et al. [2021] to construct a synthetic dataset (see Fig. 3) with paired calibrated multi-view images and meshes in a unified mesh topology. First, we select a mesh from an internal dataset with registered 3D head meshes (with 17,821 vertices each) of $\geq$ 2500 identities. Each mesh is assigned a skin texture and a randomized blend of facial expressions. To increase the diversity and realism of the training data, these textured meshes are augmented with various assets including clothing, facial and scalp hair, and accessories. We render the scenes using Blender's Cycles engine from 13 predefined camera views, with each render composited over a randomly selected background. Each image is rendered with resolution 1536×1024. The camera configuration covers the frontal hemisphere of the face, simulating our physical multi-view capture environment. In total, the dataset consists of 300,000 data pairs across 2,064 unique identities (varying in face shape and appearance). The identities represent these demographics: 46%/54% female/male; age groups 20–35 (48%), 36–55 (39%), and 56+ (12%); and ethnicities comprising White (38%), East Asian (24%), South Asian (12%), Hispanic/Latino (9%), Black, (9%), Southeast Asian (5%),Middle Eastern (4%), and others (1%).

Training data. We partition the synthetic dataset into training, validation, and test sets using an 80%/10%/10% ratio, ensuring disjoint identities across splits. In total, the training (validation) data consists of 251,816 (27,295) samples across 1,674 (181) identities. For training, input images are downsampled by a factor of 4 to an effective resolution of 384×256.

Parameter settings: The feature extractor F_{img} utilizes a DinoV2-B backbone [Oquab et al. 2023] with four registers and LoRA [Hu et al. 2022] residuals ($r = 5$). We concatenate four evenly spaced backbone layers and project them to $d_f = 98$ via 1×1 convolutions and pixel shuffling to output feature maps with $w_f = 27$ and $h_f = 18$ for the 384×256 input. The shared transformer F_{pred} uses an XCiT architecture following the ViT-S configuration (12 layers, 6 heads, $d_m = 384$) with each layer's FFN using a 1,536-dimension inner layer and GELU activation. For prediction, the graph stage ($n_c = 3,000$) employs a sampling graph $\mathbf{S}_g$ of 16 concentric shells of twice-subdivided icosahedra with 25mm radial spacing, totaling $n_g = 2,592$ points. The final stage uses surface-aware shells $\mathbf{S}_l$ displaced at ±4mm (i.e., $d_l = 4$) around $\hat{\mathbf{V}}_c$ ($3n_c = 9,000$ points). Template vertices are tokenized via a shared MLP consisting of three linear layers ($2d_m$ projection) with GELU activations and a final linear projection to d_m. The sampling points for positional embeddings $(\mathbf{S}_g, \mathbf{S}_t)$ are processed by two linear layers with an intermediate GELU activation.

The model is implemented in PyTorch [Paszke et al. 2019] and optimized using AdamW [Loshchilov and Hutter 2019] with a batch size of 3 for 900,000 steps. Training takes approximately 2 weeks on a single NVIDIA H100 80GB HBM3 GPU. We use a 10k-step linear warmup to a 1e-4 learning rate, held constant for 100,000 steps, followed by exponential decay. Training is phased: the first 500k iterations focus on the coarse and LoRA layers ($\lambda_c = 1.0, \lambda_f = 0.0$), after which the full model is trained end-to-end ($\lambda_f = 1.0$). Vertex-to-vertex and vertex-to-plane losses are weighted equally ($\lambda_{\text{v2v}} = \lambda_{\text{v2p}} = 1.0$), with vertex weights $\boldsymbol{\Omega}$ prioritizing facial features (5.0 for lips/eyelids, 3.0 for skin, eyebrows, ears, and nose), and 1.0 for the rest (e.g., mouth interior, teeth, neck, scalp).

Data augmentation includes independent per-view color augmentations (brightness ±0.2, contrast/saturation 0.9–1.1, hue ±0.02) and global geometric transformations (±45° rotation, 0.9–1.4× scaling), with camera intrinsics updated accordingly. To ensure camera robustness, we randomly sample 8–13 views per batch and apply random rotations to the coarse sampling graph.

5 EVALUATION

Test data. SHELLS is evaluated on held-out synthetic data and real-world multi-view capture. **(1)** *Synthetic test.* This set comprises 30,889 procedural samples across 209 identities disjoint from the training and validation sets. **(2)** *Real test.* We utilize 9,617 multi-view frames from 303 subjects recorded in approximately 50 static expressions using a calibrated 13-camera system. The system provides full camera parameters, including extrinsics, intrinsics, and lens distortions. To establish reference geometry, we reconstruct unstructured scans via deep MVS [Qiu et al. 2024] and register them by fitting a 3DMM, guided by predicted dense per-view landmarks, followed by non-rigid surface deformation. Both raw scans and registrations are used for evaluation.

Baselines. We compare SHELLS against TEMPEH [Bolkart et al. 2023], a 3DMM regressor, and a traditional multi-view fitting.

(1) *TEMPEH.* TEMPEH performs feed-forward multi-view registered mesh inference. We train it on our synthetic data for 1.4M steps (~30 days on an NVIDIA H100) using a batch size of one due to its high training memory demand (~65GB). To ensure a fair comparison, we adapted the original implementation to use the same supervision, loss functions, and per-vertex weighting as SHELLS. Following our schedule, we optimized its global stage for 500k iterations before training end-to-end.

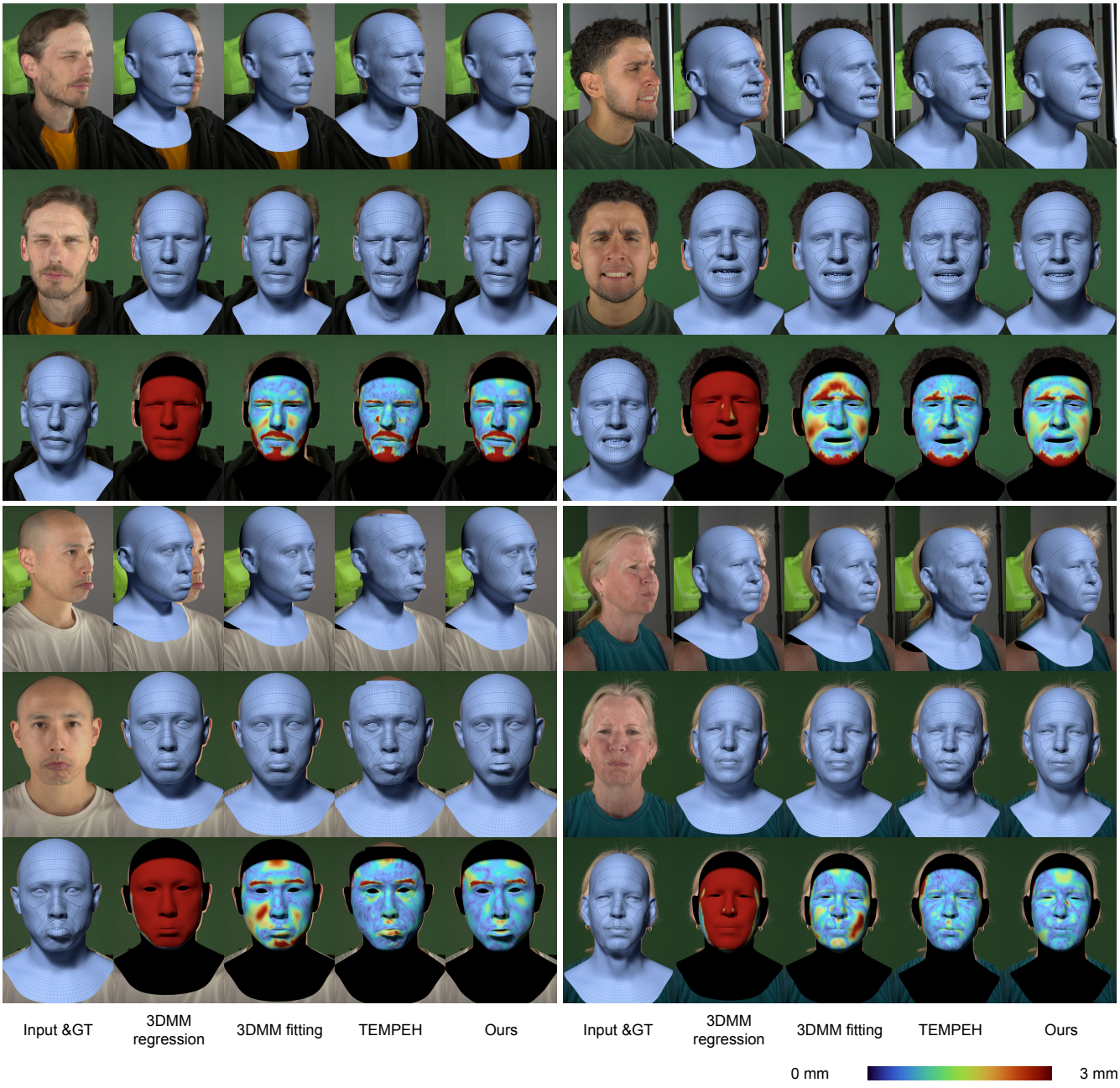

Fig. 4. **Baseline comparisons.** Comparison to the 3DMM regression, 3DMM fitting [Wood et al. 2021], and TEMPEH [Bolkart et al. 2023]. For each sample, we show one side view, a frontal view, and a rendering of the reference registration overlaid with the frontal image. The error visualizes the color coded (range $0 - 3$ mm) point-to-surface distance of each point in the reconstructed mesh and the closest point in the surface of the reference scan.

(2) *3DMM regressor.* We implement a baseline that regresses 116 identity and 197 expression shape parameters of a 3DMM. The model combines linear blend skinning for neck and eyeball articulation with linear identity and expression blend shapes, following existing models [Bednarík et al. 2024; Li et al. 2017; Wood et al. 2021]. For each multi-view input, we detect dense landmarks [Wood et al. 2022], crop the facial region, and process each image through a monocular 3DMM parameter regressor. The final multi-view result is obtained by averaging the predicted parameters across all views.

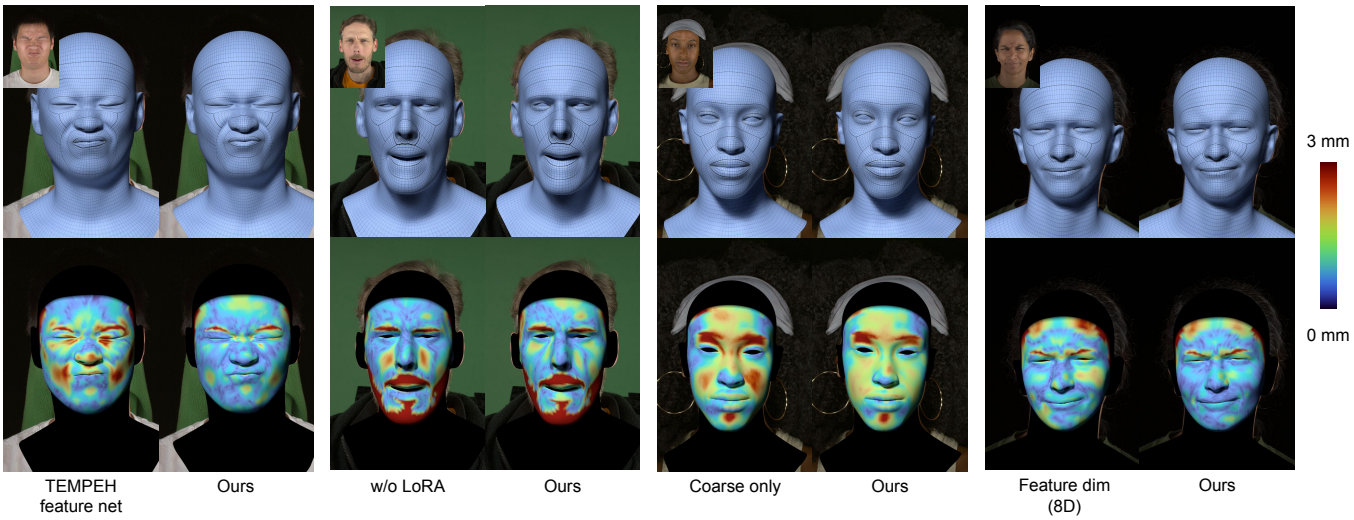

Fig. 5. **Ablations.** We show qualitative comparisons of SHELLS (Ours) to different ablated model variants.

(3) *Multi-view fitting.* We jointly fit the 3DMM to dense per-view landmarks, following Wood et al. [2022], taking around 35 seconds.

5.1 Qualitative evaluation

Figure 4 qualitatively compares SHELLS with 3DMM regression, 3DMM fitting [Wood et al. 2021], and TEMPEH [Bolkart et al. 2023]. The 3DMM regression baseline generally lacks the metric accuracy required for multi-view reconstruction. While the 3DMM fitting achieves better alignment via per-view landmark guidance, it results in poor fits in the cheek and forehead regions (Fig. 4, top right) and fails to capture subtle dynamics such as lip rolling (bottom left) or cheek blowing (bottom right) as effectively as our method. TEMPEH recovers more geometric detail and fits the scan surface more closely, as is reflected in the error visualizations, however, it lacks global surface coherence and frequently produces noisy mesh estimates (see Figure 6). Instead, SHELLS consistently produces smooth and visually coherent meshes.

5.2 Quantitative evaluation

Data and metrics. We evaluate reconstruction accuracy on the synthetic test set (D_{syn}) and the real capture test set (D_{real}) using four metrics. On D_{syn}, providing topologically consistent ground truth, we compute Euclidean vertex-to-vertex (V2V) distances. Since D_{real}lacks ground truth, we report V2V distances relative to reference registrations. For geometric fidelity against MVS scans, we compute point-to-surface (P2S) distances between skin-labeled scan vertices and the predicted mesh, where skin labels are obtained by projecting 2D face parsing results onto the 3D scans. While V2V assesses semantic correspondence accuracy, P2S evaluates the geometric fidelity relative to the unstructured MVS data. We also quantify mesh quality through triangle distortion and orientation. Distortion is measured as the mean absolute difference in triangle area between prediction and the ground truth. To detect mesh artifacts, we evaluate orientation by identifying inverted normals. We report this as a percentage of triangles per mesh to indicate the frequency of triangle flips.

Table 1. **Ablations and comparisons on synthetic data.** Ablations of individual model components and TEMPEH [Bolkart et al. 2023] baseline comparisons. We report the vertex-to-vertex (V2V) ground truth distance on the face region (left) and the entire mesh (right).

Method	V2V (mm) - face only			V2V (mm) - full mesh		
	Mean	Median	Std	Mean	Median	Std
Feature extraction						
(1) TEMPEH feature net	2.00	1.82	1.06	2.49	2.24	1.38
(2) W/o downsampling	1.97	1.65	1.23	2.41	2.12	1.41
(3) Feature dim. (8D)	1.56	1.39	0.85	2.03	1.79	1.18
(4) W/o LoRA	1.62	1.46	0.86	2.11	1.87	1.21
Mesh prediction						
(5) Samp. density (w/o subdiv)	1.85	1.65	1.01	2.57	2.27	1.53
(6) Samp. density (1 subdiv)	1.56	1.39	0.86	2.20	1.94	1.31
(7) Coarse res. (500)	1.43	1.26	0.82	1.93	1.66	1.20
(8) Coarse only	1.50	1.32	0.83	2.00	1.76	1.19
TEMPEH						
Global	2.09	1.87	1.11	2.67	2.36	1.55
Refinement	1.94	1.71	1.09	2.59	2.25	1.56
SHELLS (ours)						
Coarse	1.47	1.30	0.83	1.92	1.66	1.25
Final	1.39	1.22	0.79	1.83	1.59	1.12

Baseline comparison. As shown in Tab. 1, SHELLS outperforms TEMPEH [Bolkart et al. 2023] on synthetic data, achieving a 29% (28%) lower median (mean) V2V error. This performance gain generalizes to capture data (Tab. 2), with a 21% (20%) median (mean) V2V error reduction compared to TEMPEH. A notable distinction arises in the P2S metrics on capture data. While SHELLS achieves a 5% lower mean P2S error, TEMPEH exhibits an 18% lower median P2S distance. This indicates that TEMPEH's per-vertex refinement effectively pulls individual points toward the scan boundary, but at the cost of global surface coherence. The higher V2V error for

Table 2. **Comparisons on capture data.** Baseline comparisons against 3DMM regression, 3DMM fitting [Wood et al. 2022], and TEMPEH [Bolkart et al. 2023]. We report the vertex-to-vertex (V2V) distance to registrations and the point-to-surface (P2S) distance between MVS scans and the predictions. V2V 3DMM fitting results are in brackets, as the registrations use the fitting results for initialization, creating an inherent bias.

Method	V2V (mm)			P2S (mm)		
	Mean	Median	Std	Mean	Median	Std
3DMM regression	30.33	30.20	1.52	17.23	18.31	9.37
3DMM fitting	(1.53)	(1.33)	(0.94)	2.05	1.03	3.31
TEMPEH						
Coarse	2.28	2.06	1.19	1.66	1.09	2.27
Final	2.13	1.90	1.16	1.19	0.62	2.10
SHELLS (ours)						
Coarse	1.74	1.53	1.00	1.14	0.78	1.17
Final	1.71	1.50	0.97	1.13	0.76	1.17

Table 3. **Ablations on capture data.** We evaluate the contribution of individual components in the feature extraction and mesh prediction stages. We report the vertex-to-vertex (V2V) distance to registrations and the point-to-surface (P2S) distance between MVS scans and the predictions.

Method	V2V (mm)			P2S (mm)		
	Mean	Median	Std	Mean	Median	Std
SHELLS (Ours)	1.71	1.50	0.97	1.13	0.76	1.17
Feature extraction						
(1) TEMPEH feature net	2.31	2.08	1.23	1.70	1.12	2.35
(2) W/o downsampling	2.22	1.92	1.28	1.34	0.96	1.29
(3) Feature dim. (8D)	1.80	1.59	1.01	1.25	0.86	1.28
(4) W/o LoRA	1.82	1.62	0.98	1.27	0.86	1.27
Mesh prediction						
(5) Samp. density (w/o subdiv)	2.07	1.85	1.13	1.54	1.11	1.49
(6) Samp. density (1 subdiv)	1.79	1.61	0.95	1.25	0.84	1.27
(7) Coarse res. (500)	1.71	1.50	0.97	1.18	0.80	1.20
(8) Coarse only	1.77	1.58	0.95	1.17	0.81	1.17

TEMPEH confirms that while its vertices may lie closer to the scan surface, they do not maintain accurate semantic correspondence. In contrast, SHELLS's holistic prediction ensures superior correspondence across the entire head. For the baseline comparisons on capture data (Tab. 2), the 3DMM regressor performs poorly, primarily because the monocular parameter estimation lacks metric accuracy and simple averaging across views fails to resolve depth ambiguities, leading to high V2V errors. Furthermore, 3DMM fitting results in a 36% (81%) higher median (mean) P2S error compared to SHELLS, indicating an overall worse face geometry reconstruction.

Quantifying mesh quality, SHELLS significantly outperforms TEMPEH, achieving a 31% lower triangle deformation score (0.38 vs. 0.55) and nearly half the triangle flip rate (0.08% vs. 0.15%). SHELLS performs similarly to the 3DMM regressor across both metrics (deformation: 0.44, orientation: 0.08%). The optimization-based 3DMM fitting yields lower scores (deformation: 0.29, orientation: 0.07%), but requires orders of magnitude more computation time.

Note that the reference-based scores for the 3DMM fitting are inherently biased. The reference registrations used for evaluation are initialized via the 3DMM fitting process and utilize the same dense landmarks during optimization. Consequently, the reference geometry is predisposed toward the fitting baseline.

5.3 Ablation experiments

We evaluate our architectural design choices through ablation experiments on the synthetic test set (D_{syn}) and the capture test set (D_{real}), as summarized in Table 1 and Table 3. Unless otherwise noted, our discussion focuses on the median errors and face-only V2V metrics.

Feature extraction. We first evaluate the impact of the image feature extractor through several modifications: **(1)** *TEMPEH feature net*: Replacing our DinoV2-based feature extractor with the ResNet34-UNet architecture used in TEMPEH [Bolkart et al. 2023] increases V2V by 49% on $D_{\text{syn}}$and 39% on D_{real}, while P2S increases by 47%. This validates the superior geometric cues provided by foundation model features. **(2)** *W/o downsampling*: Upsampling DinoV2 features to the original input resolution (8D output) via linear layers and pixel shuffling increases V2V by 35% on $D_{\text{syn}}$and 28% on D_{real}, while P2S increases by 26%. This suggests that the native transformer feature resolution is more robust for surface alignment. **(3)** *Feature dimension (8D)*: Reducing the feature dimensionality to 8D (following TEMPEH) degrades performance, increasing V2V by 14% on $D_{\text{syn}}$and 6% on D_{real}, while P2S increases by 13%. This indicates that lower-dimensional features lack the necessary detail for dense surface regression. **(4)** *W/o LoRA*: Freezing the backbone without task-specific adaptation leads to a 20% V2V increase on $D_{\text{syn}}$and 8% on D_{real}, while P2S increases by 13%, demonstrating the importance of the LoRA layers in adapting the general-purpose backbone to facial geometry reconstruction.

Mesh prediction. We further ablate the transformer-based prediction stages: **(5** & **6)** *Samp. density (w/o subdiv / 1 subdiv)*: Varying the resolution of the coarse sampling graph $\mathbf{S}_g$ (using 192 or 672 points) demonstrates that the initial sampling density is critical. With lower density (w/o subdivision), V2V increases by 35% on $D_{\text{syn}}$and 23% on D_{real}, while P2S increases by 46%. Even with a single subdivision, V2V is 14% higher on $D_{\text{syn}}$and 7% higher on D_{real}, with a 5% P2S increase. **(7)** *Coarse res. (500)*: Reducing the intermediate mesh to $n_c = 500$ vertices marginally increases V2V on $D_{\text{syn}}$by 3%, while yielding identical V2V on $D_{\text{real}}$and a 5% P2S increase. This indicates that while the intermediate mesh guides the sampling shells, the final stage is robust to the specific resolution of the intermediate mesh. **(8)** *Graph stage only*: Omitting the second stage and regressing all 17,821 vertices directly from the global graph results in an 8% V2V increase on D_{syn}, a 5% increase on D_{real}, and a 7% P2S increase. As shown in Fig. 5, while the coarse-only stage captures the general head shape, it lacks the fine surface detail recovered by our hierarchical shell-based refinement.

Finally, we find that training with an additional Laplacian [Taubin 1995] loss (with a weight of 1.0) yields no accuracy gains. V2V errors (face only) remain identical on the synthetic test set (mean/median/std: 1.39/1.22/0.79 mm). Results on capture test data are similarly unchanged, with V2V errors of 1.71/1.49/0.97 mm and P2S errors of 1.14/0.76/1.18 mm (mean/median/std).

6 DISCUSSION

Applications. SHELLS bypasses traditional registration, enabling efficient 3DMM construction after rigid stabilization [Bednarík et al. 2024] (Fig. 7). It further facilitates large-scale performance capture, yielding temporally consistent reconstructions even when applied frame-by-frame, without requiring temporal filtering or post-processing. Figure 8 displays predictions for sample frames, while the supplementary video highlights the temporal stability of the results for multiple subjects and facial performances by visualizing the shared mesh topology across entire sequences.

Limitations. SHELLS fails to reconstruct certain tongue expressions (Fig. 9) due to limited synthetic training diversity. Improving these requires more varied tongue training configurations.

Detail reconstruction. The predicted 18k-vertex meshes capture global structure and mid-frequency features but lack geometric details (e.g., fine wrinkles and skin pores) required for photorealistic rendering. To achieve higher realism, a separate synthesis network as in ToFu [Li et al. 2021] could be trained to predict displacement maps and textures atop our output.

Occlusion. Through global attention, SHELLS handles occlusions (e.g., hair, mouth cavity) by correlating visible areas with a learned geometric prior to regress all 18k vertices holistically with a fixed mesh connectivity. Interior mouth vertices are implicitly tucked into the cavity during closure to maintain semantic consistency without adding edges between the lips even if vertices coincide.

Modeling non-skin surfaces. SHELLS is optimized to predict the skin surface beneath hair or clothing. However, neural avatars [Qian et al. 2024; Zielonka et al. 2025] often require mesh proxies aligned with the outer volume of hair or beards. Reconstructing these holistic volumes requires extending our synthetic dataset to include consistent hair and clothing surface labels.

Number of input views. Random camera dropout during training and mean-variance feature fusion make SHELLS robust to varying input image counts. While single-view reconstruction is ill-posed, results are reasonable with as few as two views (Fig. 10).

7 CONCLUSION

We have presented SHELLS, an efficient feed-forward framework for 3D head reconstruction in dense semantic correspondence from calibrated multi-view images. The core of our approach lies in a hierarchical strategy that combines a sparse global sampling graph with dynamic, surface-aware sampling shells. By decoupling feature extraction from the final mesh resolution, this design enables the model to scale to high-resolution topologies (≥18k vertices) while requiring only 12% of the GPU memory used by previous volumetric approaches. Furthermore, by replacing independent per-vertex refinement with a holistic transformer-based prediction, SHELLS maintains global surface consistency and demonstrates superior robustness to occlusions. Notably, the model generalizes effectively from synthetic training to real-world captures, eliminating the need for the costly pre-registered multi-view datasets required by prior work. Experimentally, SHELLS achieves a median registration error 21% – 29% lower than the previous state-of-the-art on both real and synthetic data. With its combination of geometric accuracy and sub-second inference speed, SHELLS provides a scalable solution for real-time multi-view performance capture.

ACKNOWLEDGMENTS

We thank M. Prinzler and V. Choutas for their helpful discussions and proofreading, D. Vicini for assistance with Mitsuba rendering, and E. Wood for support with synthetic data generation.

REFERENCES

Oleg Alexander, Mike Rogers, William Lambeth, Matt Jen-Yuan Chiang, and Paul E. Debevec. 2009. The Digital Emily project: photoreal facial modeling and animation. In *SIGGRAPH Courses*. 12:1–12:15.

Alaaeldin Ali, Hugo Touvron, Mathilde Caron, Piotr Bojanowski, Matthijs Douze, Armand Joulin, Ivan Laptev, Natalia Neverova, Gabriel Synnaeve, Jakob Verbeek, and Hervé Jégou. 2021. XCiT: Cross-Covariance Image Transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*. 20014–20027.

Ziqian Bai, Zhaopeng Cui, Jamal Ahmed Rahim, Xiaoming Liu, and Ping Tan. 2020. Deep Facial Non-Rigid Multi-View Stereo. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 5849–5859.

Jan Bednarík, Erroll Wood, Vasileios Choutas, Timo Bolkart, Daoye Wang, Chenglei Wu, and Thabo Beeler. 2024. Learning to Stabilize Faces. *Computer Graphics Forum (CGF)* 43, 2 (2024).

Thabo Beeler, Fabian Hahn, Derek Bradley, Bernd Bickel, Paul A. Beardsley, Craig Gotsman, Robert W. Sumner, and Markus H. Gross. 2011. High-quality passive facial performance capture using anchor frames. *SIGGRAPH* 30, 4 (2011), 75.

Volker Blanz and Thomas Vetter. 1999. A Morphable Model for the Synthesis of 3D Faces. In *SIGGRAPH*. ACM, 187–194.

Timo Bolkart, Tianye Li, and Michael J. Black. 2023. Instant Multi-View Head Capture through Learnable Registration. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 768–779.

James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David J. Dunaway. 2016. A 3D Morphable Model Learnt from 10,000 Faces. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 5543–5552.

Prashanth Chandran, Gaspard Zoss, Markus Gross, Paulo Gotardo, and Derek Bradley. 2022. Shape Transformers: Topology-Independent 3D Shape Models Using Transformers. *Computer Graphics Forum (CGF)* 41, 2 (2022), 195–207.

Victoria Yue Chen, Daoye Wang, Stephan Garbin, Jan Bednarik, Sebastian Winberg, Timo Bolkart, and Thabo Beeler. 2025. Pixels2Points: Fusing 2D and 3D Features for Facial Skin Segmentation. In *Eurographics 2025 - Short Papers*. The Eurographics Association.

Bernhard Egger, William A. P. Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, Christian Theobalt, Volker Blanz, and Thomas Vetter. 2020. 3D Morphable Face Models - Past, Present and Future. *Transactions on Graphics (TOG)* 39, 5 (2020), 157:1–157:38.

Haiwen Feng, Junyi Zhang, Qianqian Wang, Yufei Ye, Pengcheng Yu, Michael J. Black, Trevor Darrell, and Angjoo Kanazawa. 2025. St4RTrack: Simultaneous 4D Reconstruction and Tracking in the World. In *International Conference on Computer Vision (ICCV)*. IEEE, 8503–8513.

Panagiotis Filntisis, George Retsinas, Radek Danecek, Vanessa Sklyarova, Petros Maragos, and Timo Bolkart. 2026. Registration-Free Learnable Multi-View Capture of Faces in Dense Semantic Correspondence. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Graham Fyffe, Koki Nagano, Loc Huynh, Shunsuke Saito, Jay Busch, Andrew Jones, Hao Li, and Paul E. Debevec. 2017. Multi-View Stereo on Consistent Face Topology. *Computer Graphics Forum (CGF)* 36, 2 (2017), 295–309.

Michael Garland and Paul S. Heckbert. 1997. Surface simplification using quadric error metrics. In *SIGGRAPH*. ACM, 209–216.

Simon Giebenhain, Tobias Kirschstein, Martin Rünz, Lourdes Agapito, and Matthias Nießner. 2025. Pixel3DMM: Versatile Screen-Space Priors for Single-Image 3D Face Reconstruction. *CoRR* abs/2505.00615 (2025). arXiv:2505.00615

Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. 2020. Cascade Cost Volume for High-Resolution Multi-View Stereo and Stereo Matching. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2492–2501.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations (ICLR)*. OpenReview.net.

Sunghoon Im, Hae-Gon Jeon, Stephen Lin, and In So Kweon. 2019. DPSNet: End-to-end Deep Plane Sweep Stereo. In *International Conference on Learning Representations (ICLR)*.

Fig. 6. **Mesh consistency.** TEMPEH [Bolkart et al. 2023] (top) exhibits significant surface noise and artifacts, whereas SHELLS (bottom) maintains global surface consistency and smoothness across various subjects.

Fig. 7. **Application to 3DMM building.** Top row: SHELLS simplifies the generation of registered meshes allowing us to easily build statistical 3DMMs of faces. Bottom row: We sample this 3DMM built from SHELLS outputs to generate novel shapes and expressions.

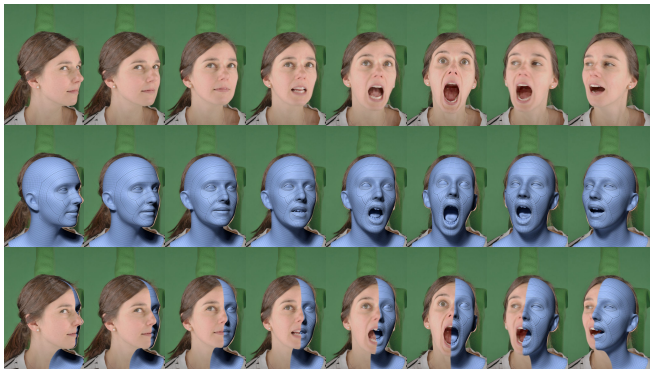

Fig. 8. **Performance registration.** SHELLS can be applied frame-by-frame to dynamic facial performances and produces temporally smooth and expressive performance registrations. See the video for the full performances.

Fig. 9. **Failure cases.** Our framework occasionally struggles with extreme tongue articulations. This is primarily attributed to the limited diversity of tongue expressions within our synthetic training set.

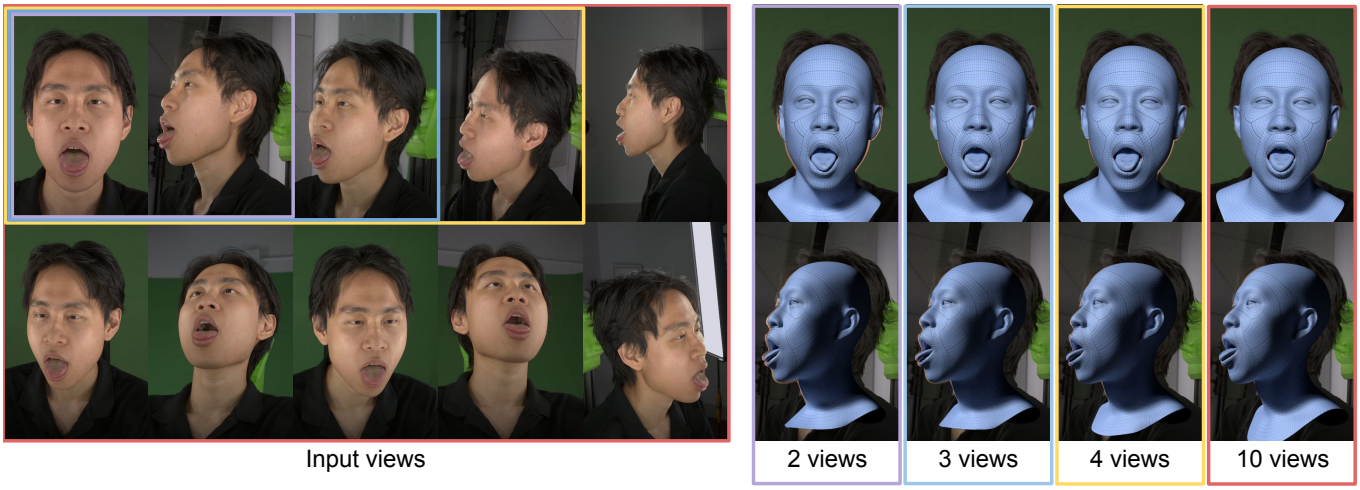

Fig. 10. **Varying views at inference.** SHELLS is robust to the number of input views. Here we show predictions given 2, 3, 4, and 10 input views for the same subject. Our predictions remain plausible even with just 2 input views featuring large disparities that challenge traditional MVS methods.

Abhishek Kar, Christian Häne, and Jitendra Malik. 2017. Learning a Multi-View Stereo Machine. In *Advances in Neural Information Processing Systems (NeurIPS)*. 365–376.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. 2024. Grounding Image Matching in 3D with MASt3R. In *European Conference on Computer Vision (ECCV)*, Vol. 15130. Springer, 71–91.

Jing Li, Di Kang, and Zhenyu He. 2024b. GRAPE: Generalizable and Robust Multi-view Facial Capture. In *European Conference on Computer Vision (ECCV)*. Springer, 403–418.

Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. 2017. Learning a model of facial shape and expression from 4D scans. *Transactions on Graphics, (Proc. SIGGRAPH Asia)* 36, 6 (2017), 194:1–194:17.

Tianye Li, Shichen Liu, Timo Bolkart, Jiayi Liu, Hao Li, and Yajie Zhao. 2021. Topologically Consistent Multi-View Face Inference Using Volumetric Sampling. In *International Conference on Computer Vision (ICCV)*. IEEE, 3824–3834.

Xuanchen Li, Yuhao Cheng, Xingyu Ren, Haozhe Jia, Di Xu, Wenhan Zhu, and Yichao Yan. 2024a. Topo4D: Topology-Preserving Gaussian Splatting for High-fidelity 4D Head Capture. In *European Conference on Computer Vision (ECCV)*. Springer, 128–145.

Shichen Liu, Yunxuan Cai, Haiwei Chen, Yichao Zhou, and Yajie Zhao. 2022. Rapid Face Asset Acquisition with Recurrent Feature Alignment. *Transactions on Graphics, (Proc. SIGGRAPH Asia)* 41, 6 (2022), 214:1–214:17.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations (ICLR)*.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. 2023. DINOv2: Learning Robust Visual Features without Supervision.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Shenhan Qian. 2024. VHAP: Versatile Head Alignment with Adaptive Appearance Priors. https://doi.org/10.5281/zenodo.14988309

Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. 2024. GaussianAvatars: Photorealistic head avatars with rigged 3D Gaussians. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 20299–20309.

Di Qiu, Yinda Zhang, Thabo Beeler, Vladimir Tankovich, Christian Häne, Sean Fanello, Christoph Rhemann, and Sergio Orts-Escolano. 2024. CHOSEN: Contrastive Hypothesis Selection for Multi-View Depth Refinement. In *European Conference on Computer Vision Workshops (ECCV-W)*. Springer.

Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. 2018. Generating 3D faces using Convolutional Mesh Autoencoders. In *European Conference on Computer Vision (ECCV)*. Springer, 725–741.

Javier Romero, Dimitrios Tzionas, and Michael J. Black. 2017. Embodied hands: Modeling and capturing hands and bodies together. *Transactions on Graphics, (Proc. SIGGRAPH Asia)* 36, 6 (2017), 245:1–245:17.

Jack R. Saunders, Charlie Hewitt, Yanan Jian, Marek Kowalski, Tadas Baltrusaitis, Yiye Chen, Darren Cosker, Virginia Estellers, Nicholas Gyde, Vinay P. Namboodiri, and Benjamin E. Lundell. 2025. GASP: Gaussian Avatars with Synthetic Priors. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 271–280.

Mike Seymour, Chris Evans, and Kim Libreri. 2017. Meet Mike: Epic Avatars. In *SIGGRAPH*. ACM, Article 12, 2 pages.

Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhöfer. 2019. DeepVoxels: Learning Persistent 3D Feature Embeddings. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2437–2446.

Gabriel Taubin. 1995. A signal processing approach to fair surface design. In *SIGGRAPH*. ACM, 351–358.

Vishwesh Vhavle, Hiteshi Jain, and Avinash Sharma. 2025. Camera3DMM: Leveraging Perspective Camera for Estimating Parametric 3D Head Models. In *SIGGRAPH Asia Conference Papers*. ACM, 39:1–39:4.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotný. 2025. VGGT: Visual Geometry Grounded Transformer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 5294–5306.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jérôme Revaud. 2024. DUSt3R: Geometric 3D Vision Made Easy. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 20697–20709.

Yating Wang, Ran Yi, Xiaoning Lei, Ke Fan, Jinkun Hao, and Lizhuang Ma. 2026. Reconstructing Topology-Consistent Face Mesh by Volume Rendering from Multi-View Images. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 12507–12511.

Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. 2021. Fake it till you make it: Face analysis in the wild using synthetic data alone. In *International Conference on Computer Vision (ICCV)*. IEEE, 3681–3691.

Erroll Wood, Tadas Baltrusaitis, Charlie Hewitt, Matthew Johnson, Jingjing Shen, Nikola Milosavljevic, Daniel Wilde, Stephan Garbin, Chirag Raman, Jamie Shotton, Toby Sharp, Ivan Stojiljkovic, Tom Cashman, and Julien Valentin. 2022. 3D face reconstruction with dense landmarks. In *European Conference on Computer Vision (ECCV)*. Springer, 160–177.

Lixin Yang, Jian Xu, Licheng Zhong, Xinyu Zhan, Zhicheng Wang, Kejian Wu, and Cewu Lu. 2023. POEM: Reconstructing Hand in a Point Embedded Multi-view Stereo. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 21108–21112.

Lixin Yang, Licheng Zhong, Pengxiang Zhu, Xinyu Zhan, Junxiao Kong, Jian Xu, and Cewu Lu. 2025. Multi-View Hand Reconstruction With a Point-Embedded Transformer. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 47, 11 (2025), 10680–10695.

Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. 2018. MVSNet: Depth Inference for Unstructured Multi-view Stereo. In *European Conference on Computer Vision (ECCV)*, Vol. 11212. Springer, 785–801.

Wojciech Zielonka, Stephan J. Garbin, Alexandros Lattas, George Kopanas, Paulo F. U. Gotardo, Thabo Beeler, Justus Thies, and Timo Bolkart. 2025. Synthetic Prior for Few-Shot Drivable Head Avatar Inversion. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 10735–10746.

Wojciech Zielonka, Tobias Kirschstein, Timo Bolkart, Simon Giebenhain, Vanessa Sklyarova, Xiang Deng, Donglai Xiang, Shunsuke Saito, Yebin Liu, Matthias Nießner, and Justus Thies. 2026. How to Build Digital Humans? From Priors to Photorealistic Avatars. *Computer Graphics Forum (Eurographics State-of-the-Art Report)* 45, 2 (2026).