i3DMM: Deep Implicit 3D Morphable Model of Human Heads Supplementary Material

Tarun Yenamandra[1,2], Ayush Tewari[2], Florian Bernard[1,2], Hans-Peter Seidel[2], Mohamed Elgharib[2], Daniel Cremers[1], and Christian Theobalt[2]
[1]TU Munich, [2]MPI Informatics, Saarland Informatics Campus

1. Model Visualization

Our implicit 3D morphable model models the identity, expression, and hairstyle components of the geometry; as well as the identity and hairstyle components of the color of the head with independent parameteric controls. We visualize these components in Fig. 1. We perform PCA on the identity geometry and color spaces in order to compute the principal components. The expression component is visualized by moving along the directions of the training expressions, since they are semantically well defined. As we model hairstyles that include caps, we show the joint space of geometry and color for hairstyles. Note that the hairstyle geometry can only take four discrete values – short, long, cap1, or cap2. Any variation within these categories is modelled by the identity-geometry component. Similarly, hairstyle color can only take the values – nocap, cap1, or cap2. The color of hair without any cap is determined by the identity-color component.

2. Experiments

Here, we provide more details on the evaluations in the main paper, and include further evaluations.

2.1. Sampling i3DMM

One of the important features of a 3D Morphable Model is the ability to randomly sample shapes in the parametric space. This has been used for generating synthetic data for training CNNs [1, 5, 6].

We achieve sampling in i3DMM by performing principal component analysis (PCA) over the training latent codes for color-identity, geometry-identity, and expressions. We weigh the singular values using a Gaussian random variable $\mathcal{N}(0, 0.1)$ for color-identity, and geometry-identity, and $\mathcal{N}(0, 0.25)$ for expressions. Since the latent codes for hair shapes and colors can take very limited numbers of values and are very well defined semantically, we sample these from their training values. We show several results in the supplemental video.

As can be seen, our model is biased towards generating male heads. This is likely due to our gender-biased training dataset with 46 males and 18 females. While this does not lead to any clear loss of quality when fitting to female test scans, see Fig. 6 in the main paper, it might lead to biased quality of results in other problems, for eg., if random samples from the model was used for training another network.

3. Comparisons

As mentioned in Sec. 4.4 of the main paper, we compare our model to two existing models, BFM [4] and FLAME [2]. We show more qualitative model fitting results in Fig. 7. Next, we provide more details on the fitting algorithm used.

Fitting: We paint the face region of each model's template mesh to create a mask as shown in Fig. 7. We use these masked regions of the models for fitting the models to head scans. To initialize, we mark 8 landmarks (eye corners, nose, lip corners, and chin) on the template mesh and rigidly align the template to each ground truth scan using Procrustes algorithm. We allow for translation, rotation, and scaling. We use the rigid alignment as initialization and optimize for the parameters of each model using a modified iterative closest point (ICP) algorithm which also updates the model parameters. The fitting algorithm maintains the initial scale and translation but optimizes for rotation. In each optimization step, we first compute the correspondences as the closest points from the masked region in the model, shown in Fig. 7, to the scan data. We compute the loss as shown in Eq. (1) and update the model parameters along with Euler angles for global rotation. We run the following optimization program up to convergence to fit the models to our scans:

Figure 1. Principal components of different spaces i3DMM models. We show color renders at the top, geometry renders in the middle, and correspondences at the bottom.

$$\underset{\theta,K,\alpha,\beta,\gamma}{\operatorname{argmin}} \sum_{i=1}^{N} \left(\left\| sR(\alpha,\beta,\gamma)x_i(\theta) + t - x_i \right\|_2 + \left\| Kc_i(\theta) - c_i \right\|_2 \right) + w_l \sum_{j=1}^{L} \left\| sR(\alpha,\beta,\gamma)l_j(\theta) + t - l_j \right\|_2, \quad (1)$$

where, $x_i(\theta) \in \mathbb{R}^3$ is a vertex i in the masked region of the model (containing N vertices), x_i is the point on the scan data corresponding to $x_i(\theta)$; $R(\alpha,\beta,\gamma) \in \mathbb{R}^{3\times3}$ is the global rotation matrix computed using the Euler angles, $\alpha, \beta,$ and γ ($\in \mathbb{R}$); $s \in \mathbb{R}$, $t \in \mathbb{R}^3$ are the global scale and translation computed during initialization; $c_i(\theta) \in \mathbb{R}^3$, $c_i \in \mathbb{R}^3$ are the colors at the vertices $x_i(\theta)$ and x_i respectively; $l_j \in \mathbb{R}^3$ and $l_j(\theta) \in \mathbb{R}^3$ are the $L(=8)$ ground truth and model landmarks respectively, as described earlier; and $K \in R^{3\times3}$ is a diagonal matrix. We set $w_l = 0.1$ during the fitting process.

Note that the color loss is only enforced for BFM, as FLAME does not model colors. Further, as the color intensities of BFM and our scans differ, we globally scale the color values using channel-specific scalars arranged as a diagonal matrix K which we optimize for along with the model parameters.

Evaluation details:

We describe the evaluation metrics in Sec. 4.4 of the

Metric	i3DMM	BFM'09	BFM'17	BFM'19
Chamfer(mm)	1.02	0.96	**0.8**	0.89
F-Score	99.31	97.66	**99.47**	98.63
Color	**0.07**	0.09	0.08	0.09

Figure 2. Comparison of BFM models with i3DMM.

main paper. Here, we present details about the masks used to evaluate these metrics.

Face region: We manually paint face masks on the ground truth scans to obtain the ground truth masks. We exclude the mouth interior of the ground truth scans. We copy this mask to the i3DMM fits. We do that by annotating a vertex in i3DMM reconstruction if the nearest point from that vertex on the ground truth scan is in the masked region. We show the face masks used to fit BFM and FLAME to ground truth scans in Fig. 7. We obtain the symmetric metrics presented in Table. 1 of the main paper for the face region in the following way. In one direction, we compute the errors from masked region of ground truth to the closest points on the (unmasked) models fit to the scan. In the other direction, we compute the errors from the masked region of the models to the (unmasked) ground truth scan. We compute errors between the masked regions of one mesh to unmasked regions of other mesh to avoid large error metrics due to annotation mistakes during manual mask painting.

Full Head: We only fit to FLAME full head model as BFM does not model the entire head. We remove the neck region from FLAME as shown in Fig. 7 as the ground truth head scans do not have neck regions. We also remove the vertices used to close the neck from ground truth as FLAME has a hole in the mesh at the neck. We compute the metrics as we do for the face region between these two full head meshes. We report the full head metrics for our model in the entire head region, including the closed hole at the neck mesh.

Comparison Results with BFM'17 and BFM'19. Towards a comprehensive comparison with state-of-the-art BFM models, we also show additional comparison results on BFM'17, and BFM'19 in Fig. 2. The main limitation of all the BFM models is that they cannot model hair.
It must be noted that we optimized the fitting method in order to obtain the best quantitative results. Many fitting approaches use a statistical regularizer, which encourages the reconstructions to be closer to the mean shape. This would lead to smoother and more realistic results (see Fig. 3), but with slightly larger quantitative errors (color error: 0.1 with reg., 0.09 w/o; identical geometry errors).

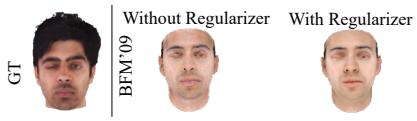

Figure 3. Comparison of BFM'09 with and without regularizer.

3.1. Ablative Analysis

In Fig. 8, we show additional qualitative results for the ablative analysis. We also show the quantitative results for full head i3DMM fit in comparison to i3DMM variants. We compare the four models that evaluate our design choices in Table 1. The error metrics are computed for the face region using manually annotated face masks as described in Sec. 3. We only evaluate the face region, as the ground truth for hair is noisy, and small quantitative differences are not very indicative of degradation in quality. Although the geometric reconstruction accuracy is marginally better without the landmark supervision loss, as compared to i3DMM, the color reconstruction accuracy of i3DMM is higher. Also, as mentioned in the main paper, Sec. 4.3, texture transfer results around the ear regions with landmark supervision loss are worse compared to i3DMM.

3.2. Correspondence Evaluation

We quantitatively evaluate the correspondences predicted by i3DMM by using the FLAME and BFM fits as ground truth correspondences. To this end, we first find the closest points from the vertices of the (masked) model fits to the i3DMM reconstructions for different scans. We will call these correspondences ground truth annotations here. We use a KD tree algorithm for efficiency. The masked face region contains 26370 vertices for BFM, and 1873 vertices for FLAME. We also transfer the annotations for one i3DMM reconstruction, to all the other reconstructions. This process is same as that described in annotation transfer application (see Sec. 4.5 of main paper). We compute the correspondence error as the average of error between the transferred and the ground truth annotations. Note that we transfer annotations from one i3DMM fit to every other i3DMM fit. Therefore, we compute a symmetric error metric.

The resulting distribution of error is shown in Fig. 4 (evaluating with BFM as ground truth is plotted in red, while FLAME is plotted in blue). The mean and median of errors for BFM is 5.08mm and 3.02mm respectively. The mean and median of errors for FLAME is 2.36mm and 1.83mm respectively. Note that, this error does not only capture the error in i3DMM's correspondence predictions but also the error in registrations of FLAME and BFM fits, see Table. 1 in the main paper.

	Uniform sampling	No landmark supervision	Independently trained color and geometry	i3DMM
Chamfer (mm) ↓	1.1065	**0.9775**	1.0319	1.0143
F-score ↑	98.5031	**99.5339**	99.1276	99.3101
Color ↓	0.0734	0.0681	0.0796	**0.0655**

Table 1. Quantitative results for ablation study. The columns, from left to right, show results obtained with uniform sampling for SDF instead of landmark-based sampling, without sparse pairwise landmark supervision loss, independently training for representing geometry and color, final model (i3DMM), and ground truth.

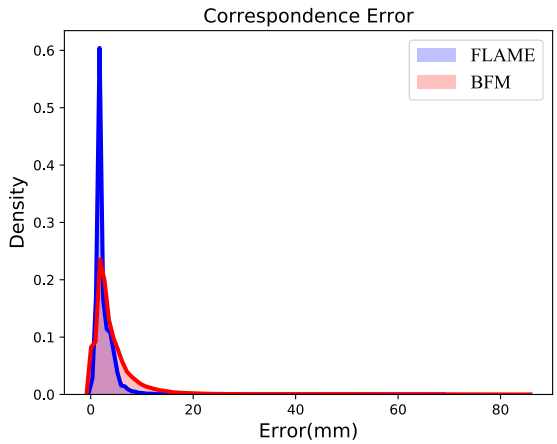

Figure 4. Distribution of errors in correspondences predicted by i3DMM computed using FLAME fits (face) as ground truth (blue), and using BFM fits (face) as ground truth (red).

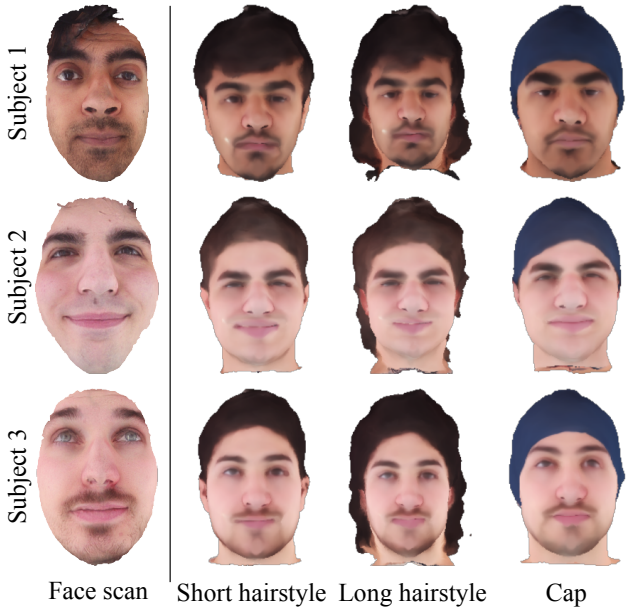

Figure 5. Completing face scans (left) with different hair styles (short hairstyle (middle-left), long hairstyle (middle-right), and cap (right)) using i3DMM as prior.

4. Applications

4.1. Full Head Completion

We use the i3DMM prior to complete face scans with different hairstyles as shown in Fig. 5. To obtain the face meshes from our head meshes, we delete all the vertices that are outside a sphere around the the tip of the nose. We learn the latent vector for the given test scan using the SDF samples of the face mesh, as described in Sec. 4.1 of the main paper. Additionally, we semantically control the hair style of the completed scan by adding a regularizer that enforces the learned $\mathbf{z}_{\text{geoH}}$ and $\mathbf{z}_{\text{colH}}$ to be close to the hairstyle latent vectors learned during training.

It can be inferred from the results in Fig. 5 that our model learns a good prior distribution, generating plausible heads for the given faces. i3DMM offers user-guided control for head completion and can be used to turn existing face-only 3DMMs into full head 3DMMs. Further, our method can also be used as a prior distribution for applications such as monocular 3D reconstruction [7].

4.2. Annotation Transfer

We show more results for annotation transfer described in Sec. 4.5 of the main paper, in Fig. 6.

4.3. Visualization Details

The output of the i3DMM is a signed distance field. Generally, marching cubes algorithm is used to reconstruct the surface from a SDF. However, based on the resolution used, marching cubes algorithm introduces unpleasant surface artifacts. To avoid these artifacts, we used a sphere tracer to render our results. We used the Blinn-Phong reflection model to shade our geometry results. We apply a gamma correction with $\gamma = 0.65$ for the color renders. Optimizing for the latent code of i3DMM given a test mesh takes about 60s on RTX8000. Our implementation of sphere tracer takes about 40s on a RTX8000 GPU to render a 256x256 image (including network evaluations and shading) with our (unoptimized) code. We use Redner [3] for rendering meshes.

References

[1] Hyeongwoo Kim, Michael Zollhöfer, Ayush Tewari, Justus Thies, Christian Richardt, and Christian Theobalt. Inverse-

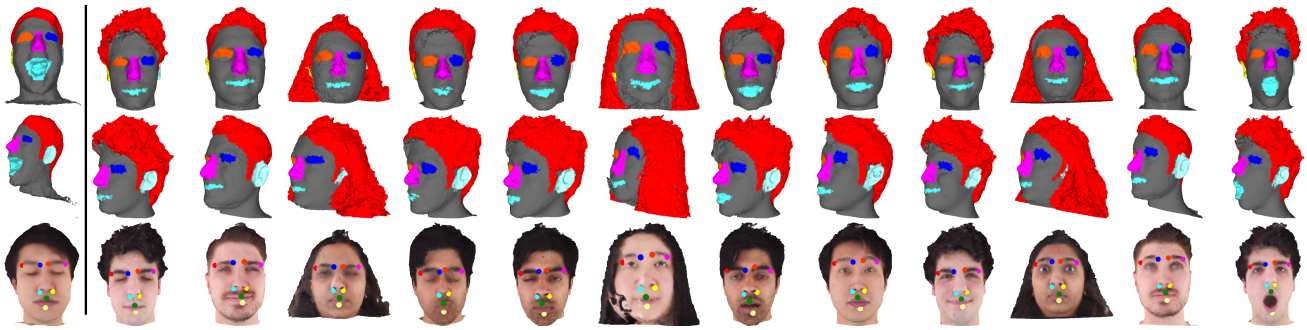

Figure 6. Additional annotation transfer results. Top: segmentation transfer front view. Middle: segmentation transfer side view. Bottom: landmark transfer. Left column shows i3DMM reconstructions with manual annotations. Right part shows annotations transferred to head scans using i3DMM.

FaceNet: Deep Single-Shot Inverse Face Rendering From A Single Image. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[2] 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), 2017. 1

[3] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, 37(6):222:1–222:11, 2018. 4

[4] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3d face model for pose and illumination invariant face recognition. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 296–301, Sep. 2009. 1

[5] Elad Richardson, Matan Sela, Roy Or-El, and Ron Kimmel. Learning detailed face reconstruction from a single image. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1

[6] M. Sela, E. Richardson, and R. Kimmel. Unrestricted facial geometry reconstruction using image-to-image translation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1585–1594, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society. 1

[7] Ayush Tewari, Michael Zollhöfer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Theobalt Christian. MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction. In *ICCV*, 2017. 4

Figure 7. Additional comparison results between i3DMM (full head) fits, BFM (face) fit, and FLAME (full head and face) fits.

Figure 8. Additional ablation results. From left to right, i3DMM without landmark-based sampling, i3DMM without landmark supervision, i3DMM with independent color and geometry training, i3DMM, and ground truth results are shown.