

Towards High Fidelity Monocular Face Reconstruction with Rich Reflectance using Self-supervised Learning and Ray Tracing

Abdallah Dib^{1*} Cédric Thébault^{1*} Junghyun Ahn¹ Philippe-Henri Gosselin¹
 Christian Theobalt² Louis Chevallier¹

¹InterDigital R&I

²Max-Planck-Institute for Informatics

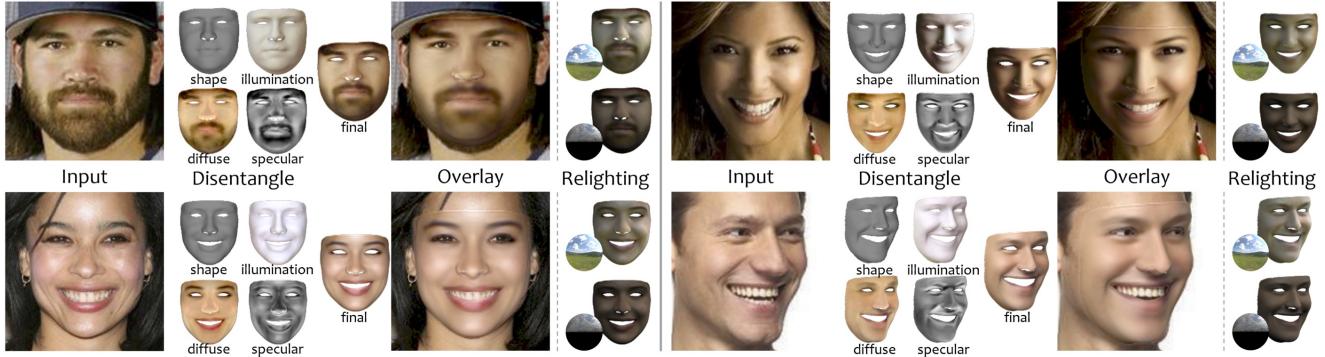


Figure 1. A deep neural network trained in a self-supervised manner with a differentiable ray tracer estimates a complete set of facial attributes – 3D head pose, geometry, personalized albedo (diffuse and specular) from unconstrained monocular image. Reconstruction based on these attributes enables a variety of applications, such as relighting.

Abstract

Robust face reconstruction from monocular image in general lighting conditions is challenging. Methods combining deep neural network encoders with differentiable rendering have opened up the path for very fast monocular reconstruction of geometry, lighting and reflectance. They can also be trained in self-supervised manner for increased robustness and better generalization. However, their differentiable rasterization based image formation models, as well as underlying scene parameterization, limit them to Lambertian face reflectance and to poor shape details. More recently, ray tracing was introduced for monocular face reconstruction within a classic optimization-based framework and enables state-of-the art results. However optimization-based approaches are inherently slow and lack robustness. In this paper, we build our work on the aforementioned approaches and propose a new method that greatly improves reconstruction quality and robustness in general scenes. We achieve this by combining a CNN encoder with a differentiable ray tracer, which enables us to base the reconstruction on much more advanced per-

sonalized **diffuse and specular albedos**, a more sophisticated illumination model and a plausible representation of self-shadows. This enables to take a big leap forward in reconstruction quality of shape, appearance and lighting even in scenes with difficult illumination. With consistent face attributes reconstruction, our method leads to practical applications such as relighting and self-shadows removal. Compared to state-of-the-art methods, our results show improved accuracy and validity of the approach.

1. Introduction

Fast and accurate image-based face reconstruction has many applications in several domains including rig based realistic videoconferencing, interactive AR/VR experiences and special effects for professionals like facial attribute manipulation/transfer or relighting. Also, supporting unconstrained pose and in-the-wild capture conditions without specific hardware, such as multi-view setup, allows for enhanced flexibility and extended applicability. However, captured images reflect the complex interaction between light and faces including shadows and specularities, which pose a real challenge for face reconstruction. Speed

*equal contribution

is also a key factor for interactive scenario and other real-time use cases. Great multi-view approaches exist ([2, 51, 46, 14, 15]), but they may not be easily applied in many applications such as VR or movies / special effects. Significant progress has been made on monocular face reconstruction where most methods resort to some form of parametric prior; high-quality analysis-by-synthesis monocular optimization methods exist ([11, 39, 12]), but besides being rather slow, they would fail for difficult head poses and lighting conditions. More recently, and to improve this robustness against lighting conditions, [8] introduced ray tracing for face reconstruction within an optimization-based framework. But the quality of their reconstruction remains sensitive to the landmarks used for initialization.

Real-time analysis-by-synthesis approaches have also been presented, however they often sacrifice reconstruction details. To increase reconstruction efficiency, CNN based approaches ([42, 41, 44, 43]) that directly regress 3D reconstruction parameters from images have been investigated. To overcome the challenge of creating large amounts of labeled data, while enabling reconstruction on the basis of meaningful scene parameters, methods combining CNNs with differentiable image formation models trained in a self-supervised way have been presented ([42, 41, 40]). They enable reconstruction performance in the range of milliseconds, and can be applied to more general scenes and subjects ([41, 40]). However, even the best of these highly efficient monocular reconstruction methods fall short of the quality and robustness requirements that many professional visual effects pipelines expect. They rely on simple parametric diffuse reflectance models based on low-frequency spherical harmonics (SH) illumination model, whereas more detailed models would be needed to reconstruct at the level of quality that is sometimes required. The inability to model self-shadows is also a prime reason for their instability under challenging scene conditions.

To overcome these limitations, we present a new approach that is the first to jointly provide the following capabilities: it enables monocular reconstruction of detailed face geometry, spatially-varying face reflectance and complex scene illumination at very high speed on the basis of semantically meaningful scene parameters. To achieve this, our model resorts to a parametric face and scene model that represents geometry using 3DMM statistical model, illumination as high-order spherical harmonics and reflectance model with diffuse and specular components. Our new CNN-based approach can be trained in a self-supervised way on unlabeled image data. It features a CNN encoder projecting the input image into the parametric scene representation. We also use an end-to-end differentiable ray tracing image formation model which, in contrast to earlier rasterization-based models, provides a more accurate light-geometry interaction and is able to synthesize images

with complex illumination accounting for self-shadows. To the best of our knowledge, this is the first time a differentiable ray tracer is used for deep-based face reconstruction in an inverse-rendering setup. While ray tracing enables [8] improving the state of the art, their method is based on costly and slow iterative optimization and the final reconstruction remains sensitive to the quality of the landmarks. Our method overcomes these limitations: it absorbs the complexity of ray tracing at training time, achieves robust and competitive results with near real-time performance, and, as it is completely independent of landmarks at test time, is more suitable for in-the-wild conditions. Finally, with an appropriate training strategy, our method is the first self-supervised method to achieve robust face reconstruction in challenging lighting conditions and captures person-specific shadow-free albedo details (such as facial hair or makeup) not restricted by the 3DMM space. Our rich and consistent facial attributes reconstruction naturally allows various types of applications such as relighting, light/albedo edit and transfer. Our comparison with recent state-of-the-art methods shows improved robustness, accuracy and versatility of our method.

2. Related works

While methods such as [51, 2, 46, 14, 15] produce high-quality face reconstruction from multi-view camera and/or from multi-light illumination setup, they are not applicable for in-the-wild images. We focus therefore on monocular image-based face reconstruction approaches that do not require any external hardware setup.

Statistical morphable model To make the highly ill-posed problem of monocular face reconstruction tractable, statistical priors have been introduced [55] such as 3D morphable models (3DMMs) [3, 27, 9]. 3DMM was the main building block for most recent approaches because of its efficiency. However, 3DMM limits face reconstruction to a low-dimensional space and does not capture person specific details (such as beards and makeup), and its statistical albedo prior bakes some illumination in it. Recently, to overcome these limitations, [38] proposes a drop-in replacement to 3DMM statistical albedo model with more sophisticated diffuse and specular albedo priors. In this work, we base our reconstruction on the 3DMM geometry and albedo prior (diffuse and specular) of [38] and we learn an increment, similarly to [43, 8], on top of the albedo prior to capture more person-specific details outside of the statistical prior space.

Optimization-based approaches [12, 11, 39, 50, 8] use optimization formulation for face (geometry and diffuse albedo) reconstruction. In [12, 11, 39, 8], a 3DMM based statistical priors act as optimization regularizer while minimizing a photo-consistency loss function. While such meth-

ods work well for controlled scene conditions, often they do not generalize well for in-the-wild images scenarios and can be computationally expensive.

Illumination and reflectance models The aforementioned methods generally use low-order spherical harmonics (SH) to model light and assume Lambertian surface, while our method uses higher-order SH in order to better model light interaction with non-Lambertian surface (with diffuse and specular). While [25] extracts diffuse and specular albedos from a single image using SH illumination, they do not explicitly model self-shadows and show only results in controlled conditions. [8] introduced a novel virtual light stage to model area lights and used a Cook-Torrance BRDF (diffuse, specular and roughness) to model skin reflectance.

Differentiable rendering A simple and efficient vertex-wise differentiable rendering is proposed in [42]. Two shortcomings of this approach are the simple Lambertian BRDF and its inability to handle self-shadows. To address these limitations, [7, 8] introduce the use of differentiable ray tracing for face reconstruction. It can handle more complex illumination and BRDF models, and naturally accounts for self-shadows. However, in addition to being computationally expensive, differentiable ray tracing exhibits noisy gradients on the geometry edges as they are sampled by very few points (solutions, such as [31, 28], exist but remain computationally expensive).

Deep learning based approaches Convolution neural networks (CNNs) autoencoders (AEs) have proven to be among the most successful non-linear functions *learner* from large corpus of datasets such as [30, 20]. Several novel approaches have been proposed ([44, 43, 42, 41, 40, 22, 37]). [42] was among the first to show the use of self-supervised autoencoder-like inverse rendering based architecture to *infer* semantic attributes. These unsupervised learning approaches have the potential to leverage the vast amount of face images. They are however limited by their differentiable rasterizer, relying on a low-order spherical harmonics parametrization and a pure-Lambertian skin reflectance model, which limits their performance and reconstruction quality under challenging lighting conditions. [42] settles for diffuse albedo obtained from 3DMM, which restricts generalization of the true diversity in face reflectance. [41] uses self-supervision to learn a corrective space to capture more person-specific albedo/geometry details outside of the 3DMM space. More recently, [5] learns user-specific expression blendshapes and dynamic albedo maps by predicting personalized corrections on top of a 3DMM prior. However, these methods do not separate diffuse and specular albedos, which get mixed in their final estimated reflectance. They also do not show reconstruction under challenging lighting conditions. Our method uses self-supervision to train a CNN and extracts personalized diffuse

and specular albedos, outside of the 3DMM space. In contrast to the aforementioned methods, our method uses high-order SH for better light approximation and an image formation layer based on a differentiable ray tracing that handle advanced lighting and self-shadows. All this contributes to a robust face reconstruction even in scenes with challenging lighting conditions. Our designed loss functions and training strategy allow us to obtain personalized diffuse and specular albedos, outside the span of 3DMM, with faithful separation between them and avoid baking residual self-shadows in the albedo. More recently, [44, 43] use supervised training to learn a non-linear 3DMM model to produce more detailed albedo and geometry. [36, 23, 1] show vast improvements in geometry reconstruction. [18] improves on these approaches further by inferring mesoscopic facial attributes given monocular facial images, an attribute we do not model in our final reconstruction. Finally, [52] and [24] use image-to-image translation networks to directly regress diffuse and specular albedos but they do not model light. Self-shadows can be observed in the albedo they obtain, whereas we model self-shadows implicitly. Additionally, these methods require ground-truth data to train the generative model, which is not easy to acquire.

3. Method

Our method is inspired by the work of [8], which achieves face reconstruction with personalized albedos outside the statistical albedo prior space and in challenging lighting conditions using ray-tracing within an optimization framework. While their method achieves state-of-the art results, it lacks robustness against initial starting point and is inherently slow, which makes their method not suitable for in-the-wild conditions. Our method overcomes these limitations by leveraging the generalization capacity of self-supervised learning.

Our method is composed of two stages as shown in Figure 2. In stage I, an input image \mathcal{I}^R is passed through a deep network \mathbf{E}^1 (a pre-trained ResNet-152) followed by a fully connected layer to predict a semantic attribute vector χ for 3DMM shape (α), expression blendshapes (δ), camera pose ($\phi = \{R, T\}$, composed of rotation and translation), light (γ) modeled with 9 spherical harmonics bands, albedo prior (β), which encodes diffuse and specular priors from [38]. Statistical diffuse \mathcal{D} and specular \mathcal{S} textures are obtained from β . These parameters are fed to a differentiable ray tracer to generate a ray traced image \mathcal{I}_1^S . This encoder \mathbf{E} is trained end-to-end in a non-supervised manner to obtain a ‘base’ reconstruction. At this level, the estimated albedos only capture low-frequency skin attributes. In stage II, to enhance these albedo priors, we train two additional decoders, \mathbf{D}_1 and \mathbf{D}_2 , in a self-supervised way to estimate

¹For the sake of conciseness, in the rest of the paper, we use \mathbf{E} to denote the combination of this encoder with the fully connected layer.

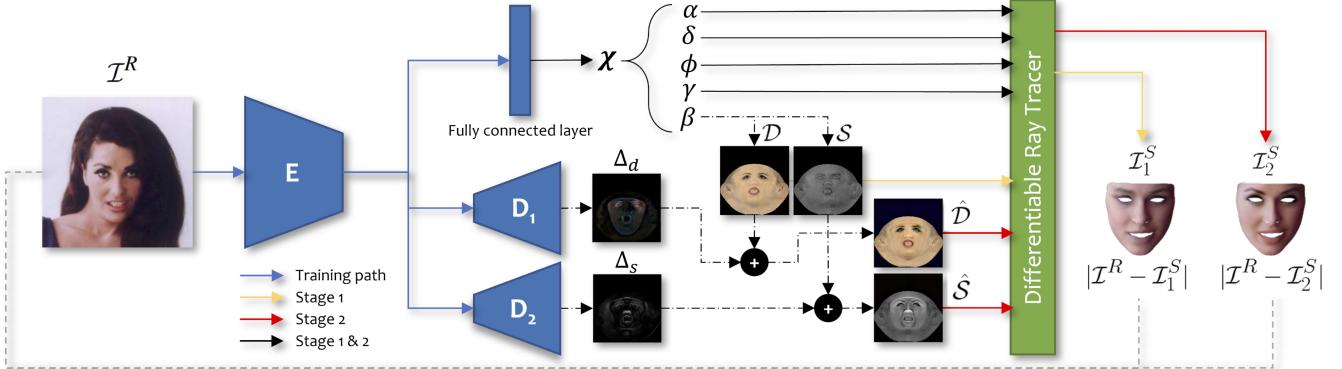


Figure 2. Method overview: A network \mathbf{E} regresses semantic face attributes (shape, expression, light, statistical diffuse and specular albedos and camera). Two additional networks $\mathbf{D}_1, \mathbf{D}_2$ are used to estimate increments Δ_d, Δ_s on top of the statistical albedos to capture personalized reflectance (diffuse and specular) outside of the statistical prior space. A differentiable ray tracer is used for image formation.

diffuse Δ_d and specular Δ_s increments to be added on top of the previously estimated textures, \mathcal{D} and \mathcal{S} , respectively. The resultant textures, $\hat{\mathcal{D}}$ and $\hat{\mathcal{S}}$, are used to generate a new image \mathcal{I}_2^S . The challenges is to avoid mixing the diffuse and specular parts and baking unexplained shadows (residual) in the personalized albedos. In the next section, we briefly describe the scene attributes used for image formation, and then we discuss the training of the different networks.

3.1. Scene attributes

Geometry We use [3, 13]’s statistical face model, where identity is given by $e = a_s + \Sigma_s \alpha$. e a vector of face geometry vertices with N vertices. The identity-shape space is spanned by $\Sigma_s \in \mathbb{R}^{3N \times K_s}$ composed of $K_s = 80$ principal components of this space. $\alpha \in \mathbb{R}^{K_s}$ describes weights for each coefficient of the 3DMM and $a_s \in \mathbb{R}^{3N}$ is the average face mesh. We use linear blendshapes to model face expressions over the neutral identity e . $v = e + \Sigma_e \delta$, where v is the final vertex position displaced from e by blendshape weights vector $\delta \in \mathbb{R}^{K_e}$ and $\Sigma_e \in \mathbb{R}^{3N \times K_e}$ composed of $K_e = 75$ principal components of the expression space.

Reflectance A simplified Cook-Torrance BRDF [6, 47], with a constant roughness term, is used to model skin reflectance. In contrast to Lambertian BRDF, the Cook-Torrance BRDF can model specular reflections, and thus defines for each vertex v_i a diffuse $c_i \in \mathbb{R}^3$ and a specular $s_i \in \mathbb{R}^3$ albedos. The statistical diffuse albedo $c \in \mathbb{R}^{3N}$ is derived from 3DMM as $c = a_r + \Sigma_r \beta$, where $\Sigma_r \in \mathbb{R}^{3N \times K_r}$ defines the PCA diffuse reflectance with $K_r = 80$ and $\beta \in \mathbb{R}^{K_r}$ the coefficients. a_r is the average skin diffuse reflectance. Similarly to [8], we employ the statistical specular prior introduced by [38] to model the specular reflectance: $s = a_b + \Sigma_b \beta$ where $\Sigma_b \in \mathbb{R}^{3N \times K_r}$ defines the PCA specular reflectance. a_b is the average specular reflectance. We use the same coefficients β to sample for the diffuse and specular albedos as suggested by [38]. In unwrapped (UV) image texture space, $\mathcal{D} \in \mathbb{R}^{M \times M \times 3}$ and

$\mathcal{S} \in \mathbb{R}^{M \times M \times 3}$ are the statistical diffuse and specular albedos, with $M \times M$ being the texture resolution.

Illumination Specular reflections are much more sensitive to light direction than diffuse reflections. So while in the literature 3-order spherical harmonics have been widely used in conjunction with Lambertian BRDF, we resort to nine SH bands with the Cook-Torrance model. We actually found SH to be better suited to our deep learning framework than an explicit spatial representation as the one used by [8]. We show in section B that high-order SH produces better shadows estimation than low-order SH. To use with the ray tracer, an environment map of 64×64 is derived from this light representation. We define $\gamma \in \mathbb{R}^{9 \times 9 \times 3}$ as the light coefficients to be predicted by the network \mathbf{E} .

Camera We use the pinhole camera model with rotation $R \in SO(3)$ and translation $T \in \mathbb{R}^3$. We define $\phi = \{T, R\}$ the parameters predicted by \mathbf{E} .

3.2. Training

First stage (Base reconstruction) In this stage, we use statistical geometry and albedo priors to obtain a first estimation of geometry, reflectance, light and camera attributes. We define $\chi = \{\alpha, \delta, \phi, \gamma, \beta\}$ as the semantic attribute vector used by the differentiable ray tracer to obtain a ray traced image \mathcal{I}_1^S . We train a deep encoder \mathbf{E} to directly regress χ . We use a pixel-wise photo-consistency loss between \mathcal{I}_1^S and the input image \mathcal{I}^R :

$$E_{ph}(\chi) = \sum_{i \in \mathcal{I}} |\mathbf{p}_i^S(\chi) - \mathbf{p}_i^R| \quad (1)$$

Here, $\mathbf{p}_i^S, \mathbf{p}_i^R \in \mathbb{R}^3$ are ray traced and real image pixel colors, respectively. Rendered pixel colors are given by $\mathbf{p}_i^S = \mathcal{F}(\chi)$, where \mathcal{F} is the Monte Carlo estimator of the rendering equation [19]. We also define a sparse landmark loss E_{land} , which measures the distance between the projection of $L = 68$ facial landmarks and their corresponding pixel projections \mathbf{z}_l on the input image (more details in

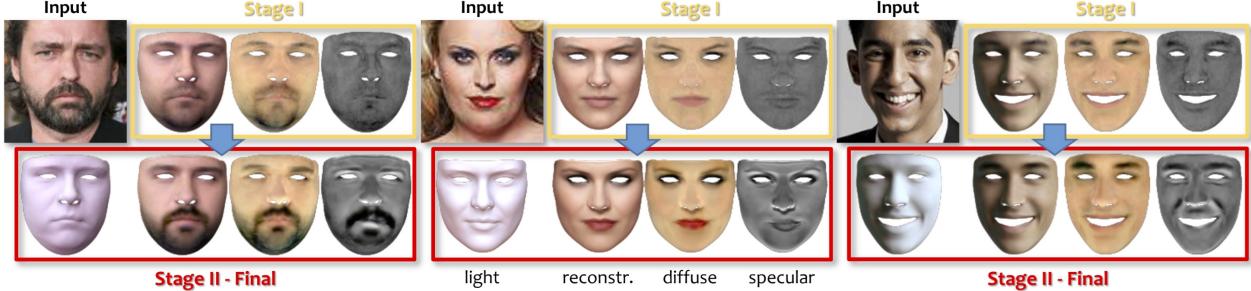


Figure 3. Yellow box: Base reconstruction (Stage I section 3) with the estimated statistical albedo priors. Red box: Final reconstruction (Stage II section 3) with the final albedos and light.

[8]). These landmarks are obtained using an off-the-shelf landmarks detector [4]. E_{land} plays an important role as it helps stabilizing the training, and it mitigates the effect of the noisy gradients on the edges introduced by the ray tracer by guiding the training toward the correct shape around the edges – especially mouth and eyes. During training, we minimize the following energy function:

$$\operatorname{argmin}_{(\alpha, \delta, \phi, \gamma, \beta)} E_d(\chi) + E_p(\alpha, \beta) + E_b(\delta) \quad (2)$$

$E_d(\chi) = E_{ph}(\chi) + \alpha_1 E_{land}(\chi)$ and E_p is the statistical face (shape and albedo) prior ([8]) that regularizes against implausible face geometry and reflectance deformations. $E_b(\delta)$ is a soft-box constraint that restricts δ to range $[0, 1]$.

Second stage (personalized albedos) In Figure 3 (yellow boxes) we show the final reconstruction together with the estimated statistical albedos (D and S) obtained by the base reconstruction of stage I. This result misses subject specific skin features, like beards and make-up, as 3DMM does not support them. We aim to personalize the albedo. However the challenges are to avoid mixing the diffuse and specular parts and to avoid baking unexplained shadows (residual) in the personalized albedos. We train two additional networks D_1 and D_2 , which take as input the latent space of E and estimate a diffuse and specular increments Δ_d and Δ_s which are added to D and S , respectively. $\hat{D} = D + \Delta_d$ and $\hat{S} = S + \Delta_s$ are then used by the differentiable ray tracer to generate a new synthetic image I_2^S . Predicting an increment on statistical texture priors instead of directly estimating a complete texture is important to force E to produce a good albedo prior while at the same time estimating a good increment over these priors. We define $\hat{\chi} = \{\alpha, \delta, \phi, \gamma, \hat{D}, \hat{S}\}$ which is used in the photo-consistency loss (eq 7). The following energy function is minimized:

$$\begin{aligned} & \operatorname{argmin}_{(\Delta_d, \Delta_s)} E_d(\hat{\chi}) + w_1(E_s(\hat{D}) + E_s(\hat{S})) + \\ & w_{2D} E_c(\hat{D}, D) + w_{2S} E_c(\hat{S}, S) + \\ & w_3(E_m(\hat{D}) + E_m(\hat{S})) + \\ & w_4(E_b(\hat{D}) + E_b(\hat{S})) \end{aligned} \quad (3)$$

where E_b is the soft box constraint that restricts the albedos to remain in an acceptable range $[0, 1]$. E_m is a constraint term which ensures local smoothness at each vertex, with respect to its first ring neighbors in the UV space, and is given by $E_m(\hat{A}) = \sum_{x_j \in N_{x_i}} \|(\hat{A}(x_j) - \hat{A}(x_i))\|_2^2$, where N_{x_i} is 4-pixel neighborhood of pixel x_i .

We use regularization similar to [43, 8] to prevent residual shadows to leak into the albedos. $E_s(\hat{A}) = \sum_{i \in M} |\hat{A}(x_i) - \text{flip}(\hat{A}(x_i))|_1$ is a symmetry constraint, where $\text{flip}()$ is the horizontal flip operator. $E_c(\hat{A}, A)$ is a consistency regularizer, which weakly regularizes \hat{A} with respect to the previously optimized albedo A based on the chromaticity κ of each pixel in the texture given by $E_c(\hat{A}, A) = \sum_{i \in M} |\kappa(\hat{A}(x_i)) - \kappa(A(x_i))|_1$.

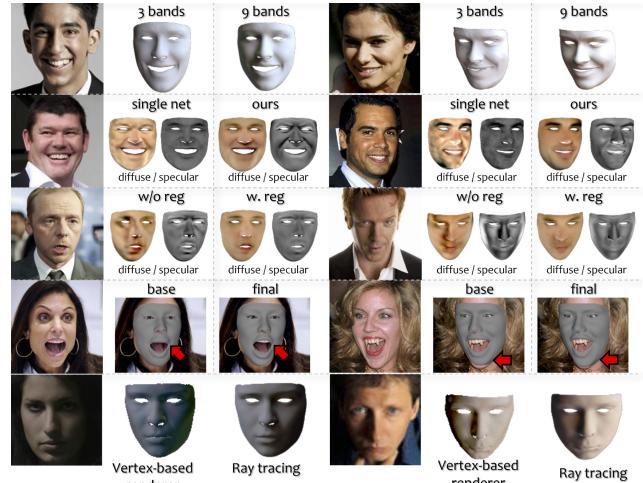


Figure 4. Row I: Comparison of the estimated light using 3 and 9 SH bands. Row II: Comparison of a single network vs. our dual-network approach to estimate albedo increments. Row III: Comparison of results obtained with and without the symmetry and consistency regularizers. Row IV: Comparison of base vs. final geometry reconstruction. Row V: comparison of vertex-based renderer and ray tracing.

Training strategy Obtaining faithful and personalized diffuse and specular albedos (outside of the statistical prior space) in uncontrolled lighting conditions is an ill-posed

problem. For instance, one can easily overfit the diffuse albedo increment Δ_d to explain the remaining information in the input image. To avoid this case, we proceed with the following strategy to separate the diffuse and specular albedos: After training \mathbf{E} for few epochs, we fix \mathbf{E} and start training \mathbf{D}_1 and \mathbf{D}_2 with a high regularization weight w_{2D} for the diffuse consistency regularizer in order to keep $\hat{\mathcal{D}}$ closer to D . Next, we progressively relax the w_{2D} constraint during the training to let the diffuse increment Δ_s capture more details. Finally, we train for all networks jointly (\mathbf{E} , \mathbf{D}_1 and \mathbf{D}_2). So, for a given image, we generate two images \mathcal{I}_1^S and \mathcal{I}_2^S and minimize the energy functions in 2 and 3 respectively and back-propagate over the whole attributes. This last step is important to stress \mathbf{E} to produce better priors, light and pose while at the same time pushing \mathbf{D}_1 and \mathbf{D}_2 to capture more details in the albedos.

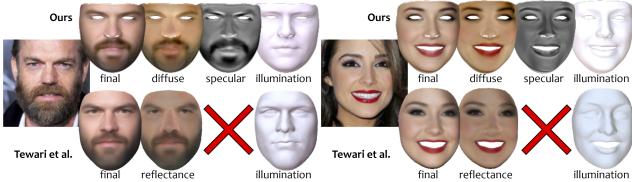


Figure 5. Comparison against [41] (subjects from authors paper).

4. Results

For training, we used a total of $250K$ images, partially from CelebA dataset [29]. For \mathbf{E} , we use a pre-trained ResNet-152, and both \mathbf{D}_1 and \mathbf{D}_2 networks use a cascade of 7 convolution layers. Ray tracing is based on the method of [28]. Because ray tracing is very memory consuming, we use 256×256 as resolution for the texture and the input images. The inference takes 54 ms (47 ms for \mathbf{E} and 7 ms for $\mathbf{D}_1, \mathbf{D}_2$). During training, we use 8 samples per pixels for ray tracing the images. Other implementation details can be found in the supplementary material (section I). Please note that images used in the figures were not used for training. The attributes estimated by our method are compatible with most rendering engines, nevertheless all results shown in the paper were rendered using ray tracing.

Figure 3 (red boxes) show reconstruction results from in-the-wild images for different subjects with various face attributes, head pose and lighting conditions. Our method successfully captures the facial hair and the lipstick – outside of statistical albedo prior space – for subjects on the left and in the middle respectively. Subject on the right is under challenging lighting conditions. Our method robustly estimates meaningful albedos and avoid baking residual shadows in the final albedos. We also show on Figure 3 the effectiveness of albedo personalization provided by the networks \mathbf{D}_1 and \mathbf{D}_2 (red box) to refine the estimated statistical priors obtained by \mathbf{E} (yellow box) from outside of the statistical albedo prior space. More results are shown on Figure 1 and in supplementary material (section 5). Finally,

this robust estimation of geometry, reflectance and illumination under general lighting conditions provides explicit control over these attributes, with several practical applications: relighting, light/albedo transfer, shadow and specularity editing, and image texture filters addition. We show relighting application on Figure 1 and other application results in supplementary material (section 6).

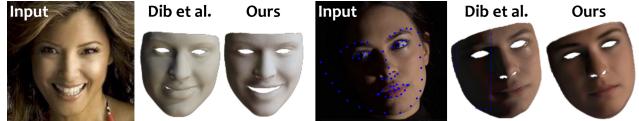


Figure 6. Comparison against [8] (right subject from authors paper).

5. Ablation

Low-order SH parameterization In this experiment, we compare 9-order with 3-order SH. Estimated illuminations are shown in Figure 4 (first row). This shows that using higher-order spherical harmonics results in better shadow reconstruction, which prevents baking residual shadows in the albedo. In a similar experiment, 9-order SH has proven itself to be slightly better than 7-order.

Dual networks In this experiment, we trained a single network to regress diffuse Δ_d and specular Δ_s increments. As shown in Figure 4 (second row), a single network produces poor albedo separation compared to our approach that uses two separate networks, possibly because the diffuse and specular components capture different skin features that interfere when using a single network.

Symmetry and consistency regularizers Figure 4 (row III) shows the importance of the regularizers used in equation 3. In fact, because our light model may not always perfectly capture the real shadows, these regularizers prevent from baking residual shadows in the albedo.

Joint training Training \mathbf{E} jointly with \mathbf{D}_1 and \mathbf{D}_2 (refer to section 3) improves the final geometry as shown in Figure 4 (row IV). The 'final' mesh better fits the input image compared to the 'base' mesh obtained from the base reconstruction (when training \mathbf{E} only).

Vertex-based renderer In this experiment, we train the same architecture, except that we used a 'vertex-based' renderer, sharing the illumination model (9-order SH) [32] and the simplified Cook-Torrance BRDF (more details on this implementation in the supp. material, section 2). Figure 4 (last row) shows that shadows are more accurately captured by the ray tracing version (especially around the nose).

6. Comparison

In this section, we compare, qualitatively and quantitatively, our method against most recent state-of-the-art methods and to the ground-truth Digital Emily project.

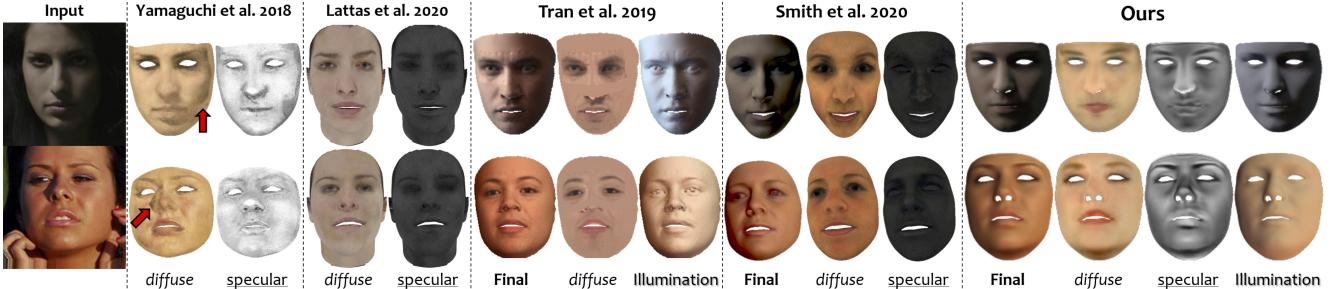


Figure 7. Comparison between methods - [52], [24], [43], [38], and ours. The red arrows indicate the shadow baked in diffuse albedo.

Visual comparison Figure 5 shows comparison results of our method against [41] (results and images are taken from authors paper). Both methods achieve visually comparable results on the final reconstructed images. However, our method separates diffuse and specular albedos while [41] only estimates a reflectance map, which mixes diffuse and specular components; This makes our method more suitable for applications such as relighting.

Figure 6 shows comparison against [8]. For the left subject, their method infers incorrect shape (around the mouth), head pose and illumination estimation. For the right subject, their estimated shape is inaccurate (right part of the head). This illustrates the sensitivity of [8] to the landmarks quality (as reported by the authors). Our method, completely independent of landmarks, is more robust to the lighting conditions and generates more convincing results. In the supp. material (section 4), we show that our method achieves visually comparable results ([8] being slightly better) while being an order of magnitude faster (54 ms vs. 6.4 min). Finally, we note that [8] can capture shadows projected by point lights, while our method can only handle lights at infinite distance; nevertheless, our training strategy, together with carefully designed loss functions prevent our method from baking unexplained shadows in the final personalized albedos at the expense of some albedo details (please refer to supp. material for more results in challenging lighting conditions).

Figure 7 shows comparison against [52, 24, 43, 38]. Results for [38] are obtained using their open-source implementations. For [43, 52, 24], results are from original authors. [52] and [24] methods do not estimate light and thus final reconstruction is not available. Because [52] does not model light, they bake some shadows in the diffuse albedo for both subjects. [38] estimates light using 3-order SH but their reconstruction is bounded by the statistical albedo space and cannot capture personalized albedos outside of this space. Compared to [43], our method has ‘visually’ better light estimation and smoother shadows (for first subject). Additionally, their diffuse and geometry have some artifacts visible around the nose. Also their method does not estimate the specular component. [24] produces convincing shadow-free diffuse albedo. However, as they do

Table 1. Vertex position and normal error μ and standard deviation σ for each method on all subjects.

| | [52] | [24] | [43] | [8] | Ours |
|------------------------------|-------|-------|-------|--------------|-------|
| Position error μ (cm) | 0.236 | 0.184 | 0.176 | 0.174 | 0.181 |
| Position error σ (cm) | 0.114 | 0.072 | 0.065 | 0.064 | 0.069 |
| Normal error μ (rad) | 0.152 | 0.156 | 0.159 | 0.139 | 0.148 |
| Normal error σ (rad) | 0.051 | 0.043 | 0.046 | 0.046 | 0.048 |

not estimate the light and head pose, they may not be suitable for most applications.

Geometric comparison We evaluate the quality of the reconstructed geometry on 23 images with 3D ground truth (GT) mesh from 3DFAW [35], AFLW2000 fitting package [54][16][17], and from the wikihuman project² [10]. We compared our method to [52, 24, 43, 8] (Figure 8 and Table 1), using vertex position and normal errors calculated with respect to the GT meshes. Table 1 reports, for each method on all subjects, the average error (μ) and standard deviation (σ) for vertex position and normal direction. For vertex-position error, [8] scores the best average error (0.174), while [43] reports the second lowest measure. Comparing to the above methods, our method shows similar performance (0.181). We note that [43] learns a non-linear 3DMM model to improve the geometry while our method, which only uses 3DMM geometry, achieves a comparable results. Our score is also on par with optimization-based method [8], while being order of magnitude faster. For normal error, our method reports the second best error (0.148), very close to [8]. Also, our score is better than [43] (0.159), since the mesh estimated by the latter has noise that appears around the nose (visible in Figure 7 and Figure 8).

Digital Emily As shown in Fig.9, we compare our method with Digital Emily [10] ground truth (GT). We note that our method bakes some albedo in the estimated light (as shown in the recovered environment map). Disentangling light color from skin color from a single image is an ill-posed problem. The limitations of 3DMM statistical albedo prior (unable to model all skin types) make this separation even more difficult. Adding light color regularization, together with a more complete albedo prior can mitigate this.

²More details in supp. material section 3.

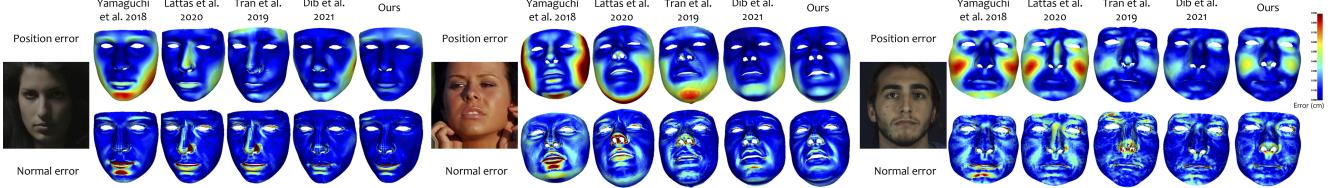


Figure 8. Vertex position and normal error for each method (from left [52], [24], [43], [8], and Ours) compared to GT mesh.

We also compare quantitatively our image reconstruction quality against state-of-the-art (see Table 2). For each method, we compute SSIM [48] and PSNR scores versus GT, for final render, diffuse, and specular. Since each method uses a different UV mapping, we compare the projection of the albedo on the input image (using the GT camera pose) and not on the unwrapped texture. For the ‘Final’ rendered image, our method is on par with the method of [8] and achieves better performance than [38]. Since [52] and [24] do not estimate scene light, they do not have a Final render image, so no comparison is available for these methods. For diffuse and specular albedos, [52] is globally better than the other methods except for ‘Diffuse SSIM’ and ‘Specular SSIM’, where [8] and our method measure higher similarity, respectively.

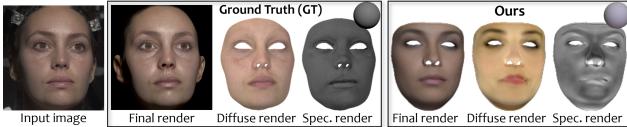


Figure 9. Our final, diffuse, and specular render images (Ours box) compared to GT (GT box). All GTs are rendered by Maya.

Table 2. Final render, diffuse and specular albedos in comparison with GT Maya renders. SSIM and PSNR (dB): higher is better.

| vs GT Render | Final (SSIM) | Final (PSNR) | Diffuse (SSIM) | Diffuse (PSNR) | Spec. (SSIM) | Spec. (PSNR) |
|-----------------|-----------------|-----------------|-------------------|-------------------|-----------------|-----------------|
| Ours | 0.933 | 35.932 | 0.653 | 29.548 | 0.642 | 29.274 |
| [8] | 0.965 | 36.390 | 0.722 | 29.812 | 0.547 | 29.670 |
| [38] | 0.906 | 35.389 | 0.639 | 29.006 | 0.452 | 28.833 |
| [24] | - | - | 0.540 | 28.633 | 0.516 | 28.926 |
| [52] | - | - | 0.679 | 30.061 | 0.604 | 30.923 |

7. Limitations, Future works and Conclusion

Limitations Our data-driven method inherits the bias of the training data, which does not provide high diversity in lighting and facial expression (most subjects in the dataset are with smiling faces, eyes opened and well lit). This leads to sub-optimal reconstruction for less frequent expression/lighting (incorrect light estimation for left subject in Figure 10). This limitation can be mitigated by using a more balanced dataset while the method remains the same. Also, with images with extreme shadows some artifacts may appear in the estimated albedos (right subject in Figure 10). Finally, in some cases, our regularization does not prevent hairs or glasses from getting baked into the albedos.

Future works Our method could be extended by using a non-linear morphable model such as [26] to improve the geometry at finer level. Using a more complex skin reflectance model such as BSSRDF/dielectric materials [49] is also interesting. Our method could also benefit from a work such as [53], which tackles the foreign (external) shadows challenge. Finally, our method can naturally extend to video-based reconstruction [40], which should improve the accuracy of the estimated facial attributes.



Figure 10. Limitations of our method.

Conclusion In this work, we address the problem of face reconstruction under general illumination conditions, an important challenge to tackle for in-the-wild face reconstruction. For this, we introduced the first deep-based and self-supervised method that achieves state-of-the-art monocular face reconstruction in challenging lighting conditions. We build our work on recent methods, namely [42] which combines deep neural networks with differentiable rendering and [8] which uses ray tracing for face reconstruction within an optimization-based framework. Our method solves the limitations of [42] by using better light and BRDF models and captures personalized diffuse and specular albedos outside of 3DMM space, while being robust against harsh shadows as shown in our results. Our approach also solves the limitations of [8] and achieves near-real time performance (compared to their extremely slow method) while at the same time being completely independent of landmarks at test time which makes our method much more suitable for in-the-wild conditions. Our method, completely self-supervised, naturally benefits from large-scale unlabeled data-sets. By comparing to recent state-of-the-art approaches, our method achieves better results in terms of robustness in scenes with challenging lighting conditions, while producing plausible reconstruction of subject-specific albedos. Beyond its robustness to lighting conditions, the rich reflectance decomposition produced by our method is compatible with existing rendering engines and allows for several style – illumination and albedo – transfer, edit applications, avatar creation and relighting.

References

- [1] Timur Bagautdinov, Chenglei Wu, Jason Saragih, Pascal Fua, and Yaser Sheikh. Modeling facial geometry using compositional vaes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3877–3886, 2018. 3
- [2] Thabo Beeler, Fabian Hahn, Derek Bradley, Bernd Bickel, Paul Beardsley, Craig Gotsman, Robert W Sumner, and Markus Gross. High-quality passive facial performance capture using anchor frames. In *ACM Transactions on Graphics (TOG)*, volume 30, page 75. ACM, 2011. 2
- [3] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999. 2, 4
- [4] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *International Conference on Computer Vision*, 2017. 5
- [5] Bindita Chaudhuri, Noranart Vesdapunt, Linda Shapiro, and Baoyuan Wang. Personalized face modeling for improved face reconstruction and motion retargeting. In *IEEE European Conference on Computer Vision (ECCV)*, 2020. 3
- [6] Robert L Cook and Kenneth E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics (TOG)*, 1(1):7–24, 1982. 4
- [7] Abdallah Dib, Gaurav Bharaj, Junghyun Ahn, Cedric Thébault, Philippe-Henri Gosselin, and Louis Chevallier. Face reflectance and geometry modeling via differentiable ray tracing. *ACM SIGGRAPH European Conference on Visual Media Production (CVMP)*, 2019. 3
- [8] Abdallah Dib, Gaurav Bharaj, Junghyun Ahn, Cédric Thébault, Philippe-Henri Gosselin, Marco Romeo, and Louis Chevallier. Practical face reconstruction via differentiable ray tracing. *Computer Graphics Forum*, 2021. 2, 3, 4, 5, 6, 7, 8, 13, 14
- [9] Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, et al. 3d morphable face models—past, present, and future. *ACM Transactions on Graphics (TOG)*, 39(5):1–38, 2020. 2
- [10] Emily. The wikihuman project. <https://vgi.ict.usc.edu/Data/DigitalEmily2/>, 2017. Accessed: 2020-05-21. 7
- [11] Pablo Garrido, Levi Valgaert, Chenglei Wu, and Christian Theobalt. Reconstructing detailed dynamic face geometry from monocular video. *ACM Transactions on Graphics*, 2013. 2
- [12] Pablo Garrido, Michael Zollhöfer, Dan Casas, Levi Valgaerts, Kiran Varanasi, Patrick Perez, and Christian Theobalt. Reconstruction of Personalized 3D Face Rigs from Monocular Video. *{ACM} Trans. Graph. (Presented at SIGGRAPH 2016)*, 35(3):28:1–28:15, 2016. 2
- [13] Thomas Gerig, Andreas Morel-Forster, Clemens Blumer, Bernhard Egger, Marcel Luthi, Sandro Schönborn, and Thomas Vetter. Morphable face models—an open framework. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 75–82. IEEE, 2018. 4
- [14] Abhijeet Ghosh, Graham Fyffe, Borom Tunwattanapong, Jay Busch, Xueming Yu, and Paul Debevec. Multiview face capture using polarized spherical gradient illumination. In *ACM Transactions on Graphics (TOG)*, volume 30, page 129. ACM, 2011. 2
- [15] P Gotardo, J Riviere, D Bradley, et al. *Practical Dynamic Facial Appearance Modeling and Acquisition*. ACM SIGGRAPH Asia, 2018. 2
- [16] Jianzhu Guo, Xiangyu Zhu, and Zhen Lei. 3ddfa. <https://github.com/cleardusk/3DDFA>, 2018. 7
- [17] Jianzhu Guo, Xiangyu Zhu, Yang Yang, Fan Yang, Zhen Lei, and Stan Z Li. Towards fast, accurate and stable 3d dense face alignment. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 7
- [18] Loc Huynh, Weikai Chen, Shunsuke Saito, Jun Xing, Koki Nagano, Andrew Jones, Paul Debevec, and Hao Li. Mesoscopic Facial Geometry Inference Using Deep Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8407–8416, 2018. 3
- [19] James T. Kajiya. The rendering equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’86, pages 143–150, New York, NY, USA, 1986. ACM. 4
- [20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 3
- [21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 12
- [22] Tatsuro Koizumi and William A. P. Smith. Look ma, no landmarks! – unsupervised, model-based dense face alignment. In *IEEE European Conference on Computer Vision (ECCV)*, 2020. 3
- [23] Samuli Laine, Tero Karras, Timo Aila, Antti Herva, Shunsuke Saito, Ronald Yu, Hao Li, and Jaakko Lehtinen. Production-level facial performance capture using deep convolutional neural networks. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 1–10, 2017. 3
- [24] Alexandros Lattas, Stylianos Moschoglou, Baris Gecer, Stylianos Ploumpis, Vasileios Triantafyllou, Abhijeet Ghosh, and Stefanos Zafeiriou. Avatarme: Realistically renderable 3d facial reconstruction “in-the-wild”. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 760–769, 2020. 3, 7, 8, 13
- [25] Chen Li, Kun Zhou, and Stephen Lin. Intrinsic face image decomposition with human face priors. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 218–233, Cham, 2014. Springer International Publishing. 3
- [26] Rui long Li, Karl Bladin, Yajie Zhao, Chinmay Chinara, Owen Ingraham, Pengda Xiang, Xinglei Ren, Pratusha Prasad, Bipin Kishore, Jun Xing, et al. Learning formation of physically-based face attributes. In *Proceedings of*

- the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3410–3419, 2020. 8
- [27] 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 (ToG)*, 36(6):194, 2017. 2
- [28] Tzu-Mao Li, Miika Aittala, Frédéric Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph.*, 37(6):222:1–222:11, Dec. 2018. 3, 6, 12
- [29] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. 6, 12
- [30] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15:2018, 2018. 3
- [31] Guillaume Loubet, Nicolas Holzschuch, and Wenzel Jakob. Reparameterizing discontinuous integrands for differentiable rendering. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 38(6), Dec. 2019. 3
- [32] Dhruv Mahajan, Ravi Ramamoorthi, and Brian Curless. A theory of frequency domain invariants: Spherical harmonic identities for brdf/lighting transfer and image consistency. *IEEE transactions on pattern analysis and machine intelligence*, 30(2):197–213, 2007. 6, 12
- [33] Tomas Möller and Ben Trumbore. Fast, minimum storage ray-triangle intersection. *J. Graph. Tools*, 2(1):21–28, Oct. 1997. 12
- [34] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. 2017. 12
- [35] Rohith Krishnan Pillai, Laszlo A. 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 *In Proceedings of the 2019 IEEE International Conference on Computer Vision Workshops*, October 2019. 7, 12
- [36] Shunsuke Saito, Lingyu Wei, Liwen Hu, Koki Nagano, and Hao Li. Photorealistic facial texture inference using deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5144–5153, 2017. 3
- [37] Soumyadip Sengupta, Angjoo Kanazawa, Carlos D Castillo, and David W Jacobs. Sfsnet: Learning shape, reflectance and illuminance of faces in the wild’. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6296–6305, 2018. 3
- [38] William AP Smith, Alassane Seck, Hannah Dee, Bernard Tiddeman, Joshua B Tenenbaum, and Bernhard Egger. A morphable face albedo model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5011–5020, 2020. 2, 3, 4, 7, 8, 13
- [39] Supasorn Suwajanakorn, Ira Kemelmacher-Shlizerman, and Steven M Seitz. Total moving face reconstruction. In *European Conference on Computer Vision*, pages 796–812. Springer, 2014. 2
- [40] A Tewari, F Bernard, P Garrido, G Bharaj, et al. *FML: Face Model Learning from Videos*. CVPR, 2019. 2, 3, 8
- [41] Ayush Tewari, Michael Zollhöfer, Pablo Garrido, Florian Bernard, Hyeongwoo Kim, Patrick Pérez, and Christian Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18–22, 2018*, pages 2549–2559. IEEE Computer Society, 2018. 2, 3, 6, 7
- [42] Ayush Tewari, Michael Zollöfer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Theobalt Christian. MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 2, 3, 8
- [43] Luan Tran, Feng Liu, and Xiaoming Liu. Towards high-fidelity nonlinear 3d face morphable model. In *In Proceeding of IEEE Computer Vision and Pattern Recognition*, Long Beach, CA, June 2019. 2, 3, 5, 7, 8, 13
- [44] Luan Tran and Xiaoming Liu. Nonlinear 3d face morphable model. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7346–7355, 2018. 2, 3
- [45] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13:376–380, 1991. 12
- [46] Levi Valgaerts, Chenglei Wu, Andrés Bruhn, Hans-Peter Seidel, and Christian Theobalt. Lightweight binocular facial performance capture under uncontrolled lighting. *ACM Trans. Graph.*, 31(6):187–1, 2012. 2
- [47] Bruce Walter, Stephen Marschner, Hongsong Li, and Kenneth Torrance. Microfacet models for refraction through rough surfaces. pages 195–206, 01 2007. 4
- [48] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *Trans. Img. Proc.*, 13(4):600–612, Apr. 2004. 8
- [49] Tim Weyrich, Wojciech Matusik, Hanspeter Pfister, Bernd Bickel, Craig Donner, Chien Tu, Janet McAndless, Jinho Lee, Addy Ngan, Henrik Wann Jensen, and Others. Analysis of human faces using a measurement-based skin reflectance model. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 1013–1024. ACM, 2006. 8
- [50] Chenglei Wu, Derek Bradley, Markus Gross, and Thabo Beeler. An anatomically-constrained local deformation model for monocular face capture. *ACM transactions on graphics (TOG)*, 35(4):1–12, 2016. 2
- [51] Chenglei Wu, Kiran Varanasi, Yebin Liu, Hans-Peter Seidel, and Christian Theobalt. Shading-based dynamic shape refinement from multi-view video under general illumination. In *2011 International Conference on Computer Vision*, pages 1108–1115. IEEE, 2011. 2
- [52] S Yamaguchi, S Saito, et al. *High-fidelity Facial Reflectance and Geometry Inference from an Unconstrained Image*. ACM TOG, 2018. 3, 7, 8, 13
- [53] Xuaner Zhang, Jonathan T. Barron, Yun-Ta Tsai, Xiuming Zhang, Ren Ng, and David E. Jacobs. Portrait shadow manipulation. volume 39, 2020. 8

[54] Xiangyu Zhu, Xiaoming Liu, Zhen Lei, and Stan Z Li. Face alignment in full pose range: A 3d total solution. *IEEE transactions on pattern analysis and machine intelligence*, 2017.

7

[55] Michael Zollhöfer, Justus Thies, Darek Bradley, Pablo Garrido, Thabo Beeler, Patrick Pérez, Marc Stamminger, Matthias Nießner, and Christian Theobalt. State of the Art on Monocular 3D Face Reconstruction, Tracking, and Applications. 2018. 2

A. Implementation details

We implemented the architecture using PyTorch [34] with a GPU-enabled backend. Ray tracing is based on the method of [28], and for training we used Adam [21] as optimizer with default parameters. We used images from CelebA dataset [29] in addition to 40K images collected from the web for a total of 250K images. We keep 2K images for the validation. Images are aligned and cropped to a resolution of 256 × 256. We trained \mathbf{E} for 10 epochs, then we fixed \mathbf{E} and trained \mathbf{D}_1 and \mathbf{D}_2 for 5 epochs. Finally we trained all networks jointly for 5 epochs. We set our regularization weights as following: landmarks weight $\alpha_1 = 1$, $w_i = 0.002$, $w_c = 0.01$, symmetry regularizer $w_1 = 20$, $w_{2S} = 0.01$, smoothness regularizer $w_3 = 0.0001$; and for w_{2D} , we start with $w_{2D} = 0.5$, and decrease it by a factor of 2 at each epoch. For \mathbf{E} , we use a pre-trained *ResNet-152* with latent space dimension equal to 1000. Both \mathbf{D}_1 and \mathbf{D}_2 networks use a cascade of 7 convolution layers. Because ray tracing is very memory consuming, we use a texture resolution of 256 × 256 with batch size equal to 8 and input image of resolution 256 × 256 to fit the GPU memory (12GB on a NVIDIA GeForce RTX 2080 Ti). For the learning rates, we use $1e^{-6}$ for \mathbf{E} and $1e^{-7}$ for \mathbf{D}_1 and \mathbf{D}_2 . For training, it takes 15 hours to do a single epoch. During training, we use 8 samples per pixels for ray tracing the images. We experimented with different numbers of samples per pixel (spp) for ray tracing (8, 16 and 32 spp), but we did not obtain substantial improvements when using more than 8 spp, even though using 16 spp already made the training much slower. Additionally, as skin is generally not a highly specular surface, in our experiments, modeling self-geometry ray bounces did not lead to substantial gain in accuracy; thus we did not use it for training. The inference takes 54 ms (47 ms for \mathbf{E} and 7 ms for $\mathbf{D}_1, \mathbf{D}_2$).

B. Vertex-based renderer implementation

In this section we provide implementation details of the vertex-based renderer that we used to compare against the ray tracer (please refer to section 5 in primary document).

The vertex based renderer computes the irradiance by evaluating spherical harmonics (SH) for each vertex of the face mesh. To model skin reflectance, we use a simplified Cook-Torrance BRDF, thus the final irradiance is the sum of diffuse and specular irradiance terms. For the diffuse term, a spatial convolution with the *half-cosine* is applied to the SH light representation. This corresponds to a multiplication of the SH coefficients (B_{lm}) of the light representation with SH coefficients (A_l) of the *half-cosine* function ([32]). For each vertex, the diffuse irradiance, \mathcal{B}_d , is obtained by

evaluating the resulting SH:

$$\mathcal{B}_d(\mathbf{n}_i, \mathbf{c}_i) = \mathbf{c}_i \cdot \sum_{l=0}^8 \sum_{m=-l}^l A_l \cdot B_{lm} \cdot Y_{lm}(\mathbf{n}_i) \quad (4)$$

where $\mathbf{c}_i \in \mathbb{R}^3$ is the diffuse albedo of a vertex. $\mathbf{n}_i \in \mathbb{R}^3$ is the vertex normal. The specular term is similarly obtained using a spatial convolution of the SH light representation with the BRDF kernel corresponding to the roughness (which is constant in the simplified Cook-Torrance BRDF model we use). The specular irradiance, \mathcal{B}_s , is obtained by evaluating the resulting SH:

$$\mathcal{B}_s(\mathbf{R}_i) = \sum_{l=0}^8 \sum_{m=-l}^l S_l \cdot B_{lm} \cdot Y_{lm}(\mathbf{R}_i) \quad (5)$$

where \mathbf{R}_i is the reflection direction of the viewing vector \mathbf{W}_i according to the surface normal, and S_l are the SH coefficients of the BRDF function corresponding to the roughness [32]. The final irradiance \mathcal{B} is equal to the sum of the diffuse and specular terms weighted by the specular intensity s_i :

$$\mathcal{B}(\mathbf{n}_i, \mathbf{c}_i, \mathbf{R}_i) = (1 - s_i) \cdot \mathcal{B}_d(\mathbf{n}_i, \mathbf{c}_i) + s_i \cdot \mathcal{B}_s(\mathbf{R}_i) \quad (6)$$

$s_i \in \mathbb{R}$ is the specular albedo.

Finally, We use the following vertex-based photo-consistency loss to minimize during the training:

$$E_{ph}(\chi) = \sum_{i=1}^N |\mathcal{B}(\mathbf{n}_i, \mathbf{c}_i, \mathbf{R}_i) - \mathcal{I}^R(\Pi \circ C(v_i))| \quad (7)$$

where N is the number of vertices, $C(v_i)$ is the projection of vertex v_i in the real image, equal to: $R^{-1}(v_i - T)$. Π is the perspective camera matrix that maps a 3D vertex to a 2D pixel.

C. Mesh difference

In this section, we provide more details on how the geometric error is calculated for each method (please refer to Table 1 in primary document).

The mean difference error is computed per-vertex on the entire mesh. We implement a 3D mesh evaluation protocol similar to [35]. For computing the mesh difference, we first align a reconstructed mesh towards a ground truth (GT) mesh. Several feature points, namely, sparse correspondence points are defined on both the GT and reconstructed facial meshes, where vertices are minimally affected by the facial muscles. With the corresponding points ready on both meshes, we use a traditional least-square estimation introduced by [45] to align the two meshes. After this alignment, we compute the distance from each vertex of a mesh to the other via a fast ray-triangle intersection method [33]. The average error is computed for final difference between two meshes.

D. More comparison results

Figure 11 shows comparison results against the method of [8]. For each subject, we show the final reconstruction, estimated diffuse, specular and light for each method. The first two subjects are from the authors of [8].

Quite logically, the iterative optimization-based method of [8] achieves slightly better reconstruction results and captures more details in the estimated albedos. This is because [8] estimates and fine-tunes the facial and scene parameters specifically for each subject, while our method infers them directly without fine-tuning. Nevertheless, our method is almost on par with [8] and can successfully handle some cases where [8] falters. For instance, with the last two subjects of the Figure 11, in presence of shadows and strong expression, landmarks detector deliver less accurate initial starting points for the method of [8] which consequently gets trapped in wrong local minima. This yields poor shape and artefacts in the estimated albedos (highlighted in red boxes). Our method does not suffer from this limitation, proves to be more robust and produces visually more plausible reconstruction. We note that [8] estimates a roughness map, a parameter that we do not estimate. However, as reported by the authors, the missing statistical prior of the estimated roughness may sometimes yield to an over-fitting on this parameter.

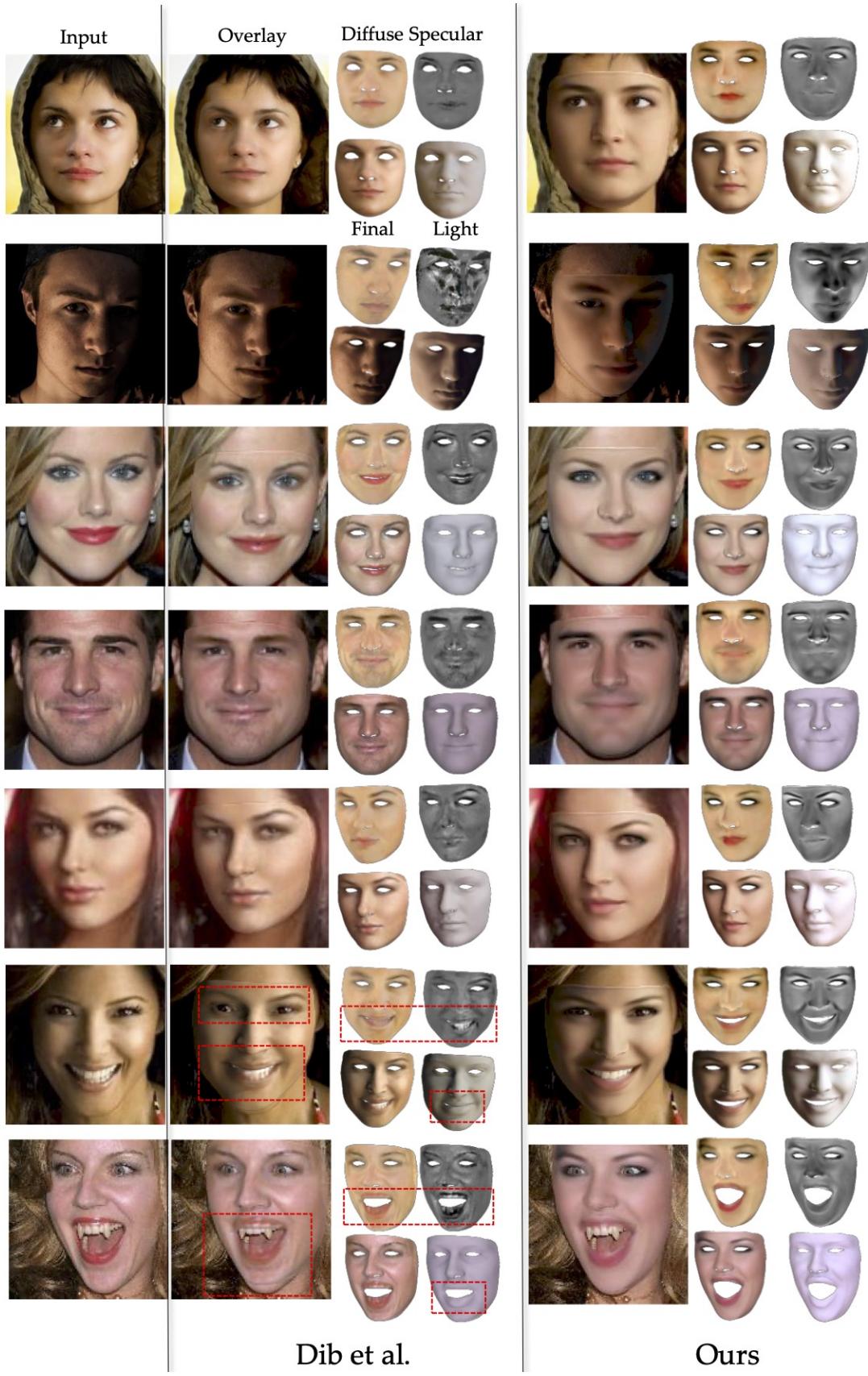
We show more qualitative comparison against [52], [24], [43], and [38] in Figure 12. In Figure 13, we show more quantitative comparison against [52], [24], [43], and [8].

E. Face catalog

In Figure 14, we show more reconstruction results from in-the-wild images. For each subject we show the final reconstruction and the estimated diffuse, specular albedos and illumination. More results are in the accompanied video.

F. More relighting examples

Figure 15 and 16 show more relighting examples where the estimated illumination is replaced with an environment-map. More relighting results are in the accompanied video.



Dib et al.

Ours

Figure 11. Comparison against [8]

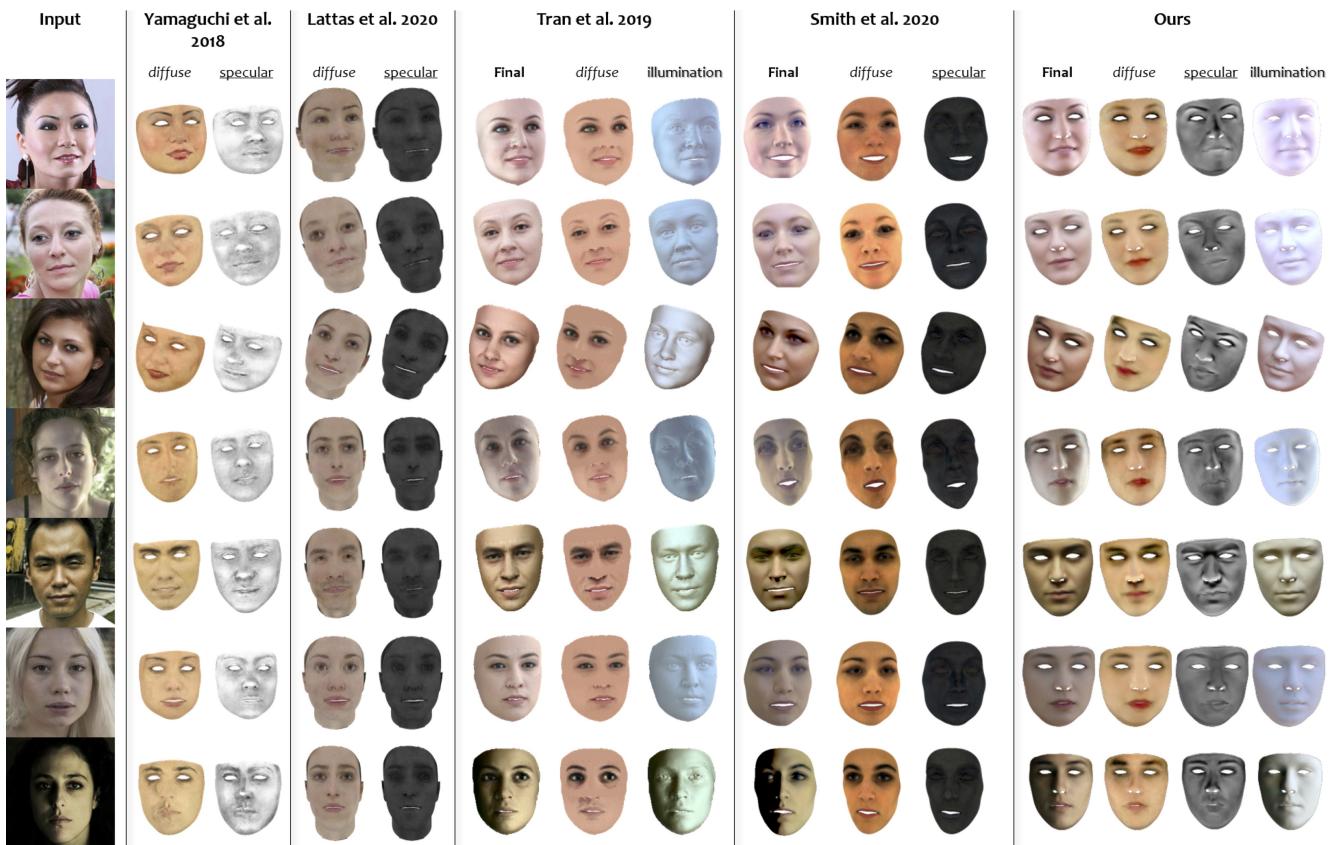
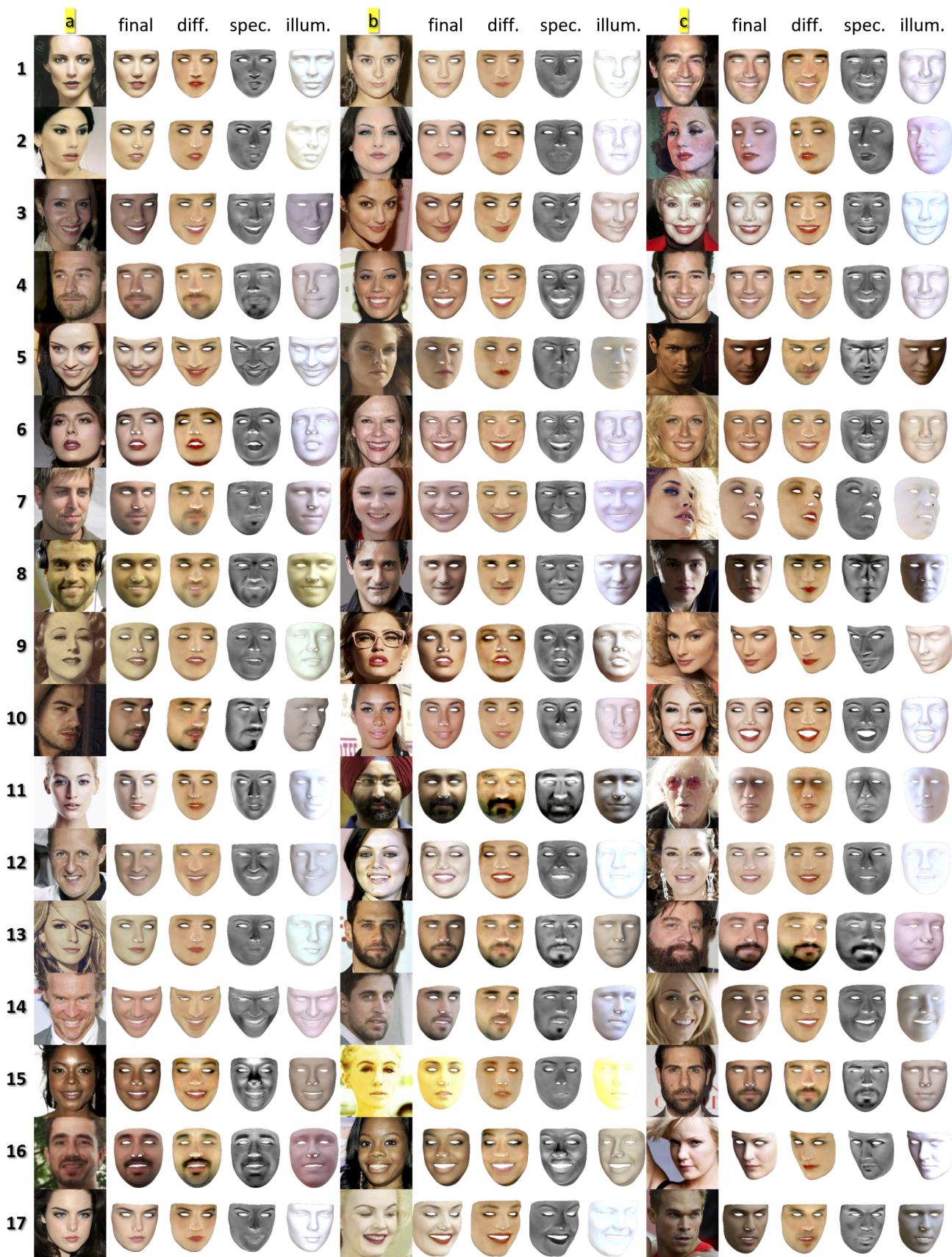


Figure 12. More visual comparisons against state-of-the-art methods.





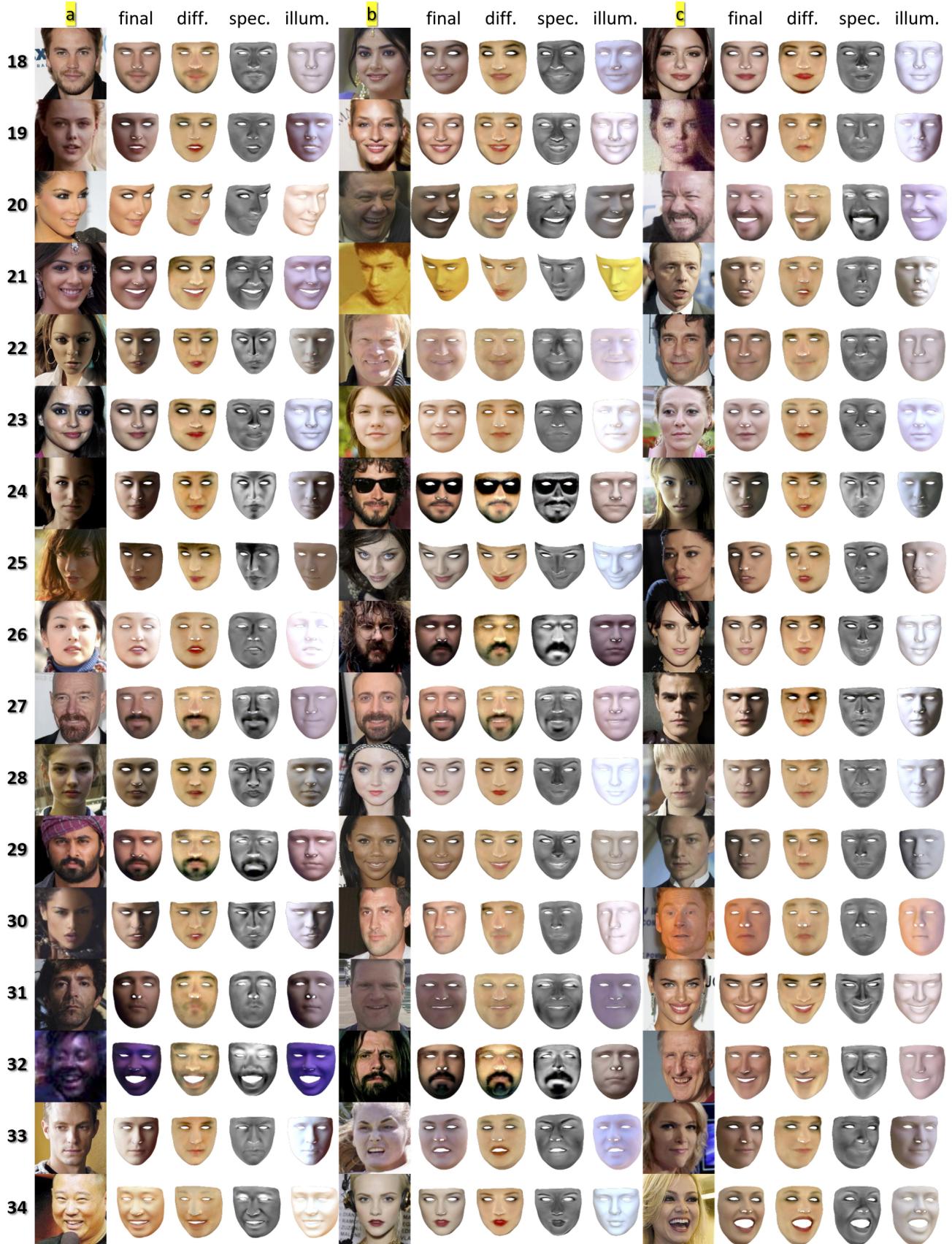


Figure 14. Face catalog of our reconstruction. For each subject, we show the input, final, diffuse, specular, and illumination. More results are in the accompanied video.



Figure 15. Relighting examples (More relighting results are in the accompanied video).

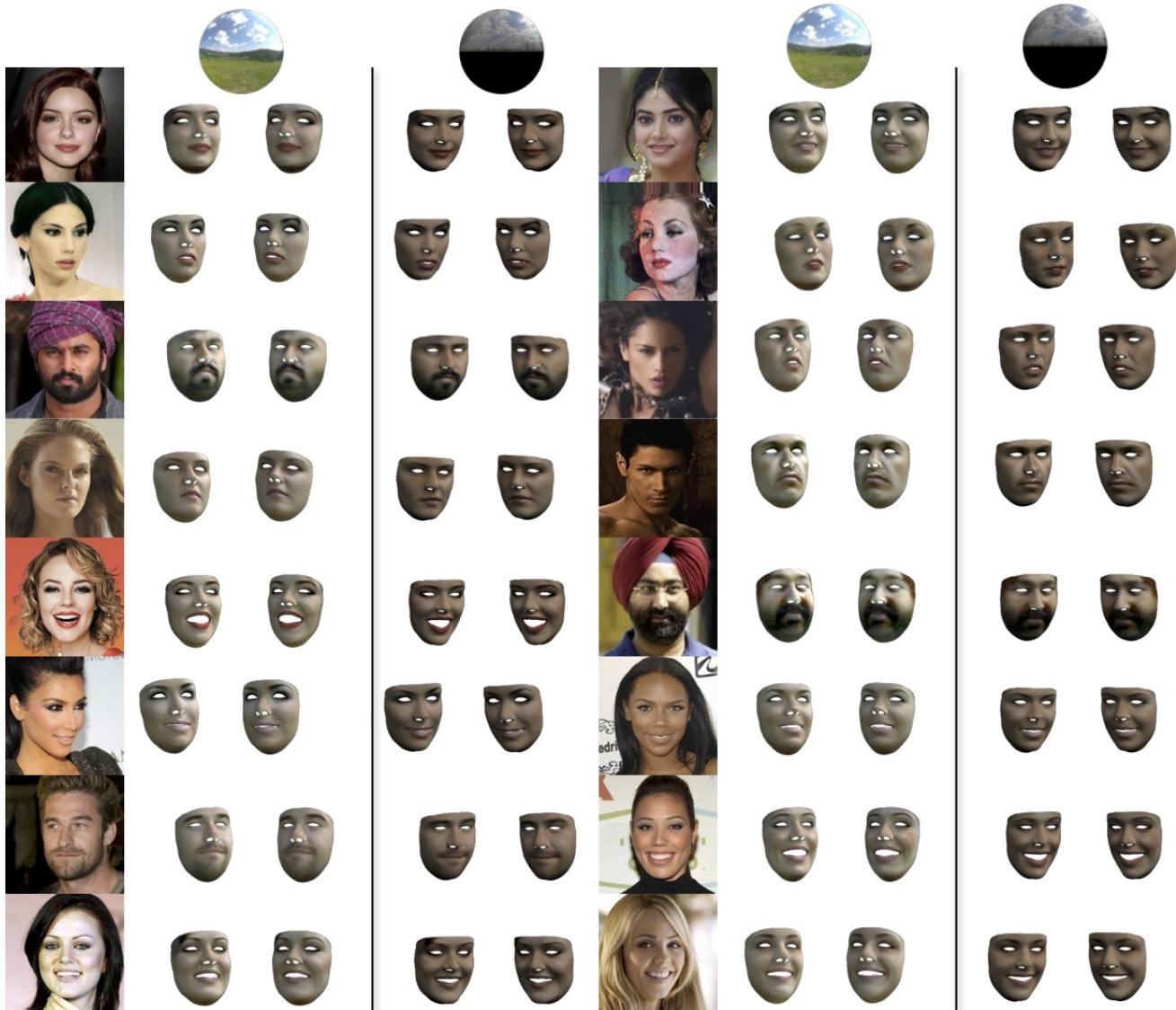


Figure 16. Relighting examples (More relighting results are in the accompanied video).