SIRA: Relightable Avatars from a Single Image Supplemental Document

Pol Caselles[1,2,3] Eduard Ramon[1,2,*] Jaime Garcia[1] Xavier Giro-i-Nieto[2,3*]

Francesc Moreno-Noguer[3] Gil Triginer[1]

[1]*Crisalix SA* [2]*Universitat Politècnica de Catalunya* [3]*Institut de Robòtica i Informàtica Industrial, CSIC-UPC*

1. Physically-based rendering model

Our physically-based rendering model is based on the implementation of [18]. Here, we provide a summary of the model for completeness. We compute the radiance r^{pb} emitted from a surface point $\mathbf{x}$ with normal $\mathbf{n}$ in the viewing direction $\boldsymbol{\omega}_{\mathrm{o}}$ using the non-emitting rendering equation

$$r^{\mathrm{pb}}(\boldsymbol{\omega}_{\mathrm{o}},\mathbf{x}) = \int_{\Omega} l(\boldsymbol{\omega}_{\mathrm{i}})\big(f^{\mathrm{d}}(\mathbf{x})+k_{\mathrm{s}}\, f^{\mathrm{s}}(\mathbf{x},\boldsymbol{\omega}_{\mathrm{i}},\boldsymbol{\omega}_{\mathrm{o}})\big)(\boldsymbol{\omega}_{\mathrm{i}}\cdot\mathbf{n})\mathrm{d}\boldsymbol{\omega}_{\mathrm{i}}, \tag{1}$$

where $l(\boldsymbol{\omega}_{\mathrm{i}})$ is the incident light from direction $\boldsymbol{\omega}_{\mathrm{i}}$, the functions $f^{\mathrm{d}}, f^{\mathrm{s}}$ are the diffuse and specular components of the BRDF respectively, and the scalar $k_{\mathrm{s}} \in [0,1]$ controls their relative weight. The integral is computed over the hemisphere $\Omega = \{\boldsymbol{\omega}_{\mathrm{i}} : \boldsymbol{\omega}_{\mathrm{i}} \cdot \mathbf{n} > 0\}$.

We express all the terms in the rendering equation using spherical gaussians, which allows us to compute the integral in a closed form. A spherical gaussian (SG) is a spherical function of the form

$$G(\boldsymbol{\nu};\boldsymbol{\xi},\lambda,\boldsymbol{\mu}) = \boldsymbol{\mu}\ \exp(\lambda(\boldsymbol{\nu}\cdot\boldsymbol{\xi}-1)), \tag{2}$$

where $\boldsymbol{\nu} \in \mathbb{S}^2$ is the normalised input direction, $\boldsymbol{\xi} \in \mathbb{S}^2$ is the direction of the lobe, $\lambda \in \mathbb{R}_+$ is the lobe sharpness, and $\boldsymbol{\mu} \in \mathbb{R}^n_+$ the lobe amplitude.

We represent the environment map $l(\boldsymbol{\omega}_{\mathrm{i}})$ as a mixture of N_{l} spherical gaussians:

$$l(\boldsymbol{\omega}_{\mathrm{i}}) = \sum_{l}^{N_{\mathrm{l}}} G(\boldsymbol{\omega}_{\mathrm{i}};\boldsymbol{\xi}_l,\lambda_l,\boldsymbol{\mu}_l). \tag{3}$$

The diffuse component of the BRDF is a scaled spatially varying RGB albedo, $\mathbf{a} \in \mathbb{R}^3$, with no angular dependence:

$$f^{\mathrm{d}}(\mathbf{x}) = \mathbf{a}(\mathbf{x})/\pi \tag{4}$$

The specular BRDF, f^{s}, used in [18] is based on the Cook-Torrance model, and has the form:

$$f^{\mathrm{s}}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}}) = \mathcal{M}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}})\mathcal{D}(\mathbf{h}) \tag{5}$$

where $\mathbf{h} = (\boldsymbol{\omega}_{\mathrm{o}}+\boldsymbol{\omega}_{\mathrm{i}})/\|\boldsymbol{\omega}_{\mathrm{o}}+\boldsymbol{\omega}_{\mathrm{i}}\|_2$. The normalized distribution function $\mathcal{D}$ is expressed as a single SG:

$$\mathcal{D}(\mathbf{h}) = G(\mathbf{h};\mathbf{n},\frac{2}{R^4},\frac{1}{\pi R^4}) \tag{6}$$

where R is the roughness parameter. This can be expressed as a function of $\boldsymbol{\omega}_{\mathrm{i}}$ by a spherical warping:

$$\mathcal{D}(\boldsymbol{\omega}_{\mathrm{i}}) \approx G(\boldsymbol{\omega}_{\mathrm{i}};\mathbf{n},\frac{1}{2R^4\mathbf{n}\cdot\boldsymbol{\omega}_{\mathrm{o}}},\frac{1}{\pi R^4}) \tag{7}$$

The function $\mathcal{M}$ accounts for the Fresnel and shadowing effects, as is usual in the Cook-Torrance model:

$$\begin{aligned}
\mathcal{M}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}}) &= \frac{\mathcal{F}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}})\mathcal{G}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}})}{4(\mathbf{n}\cdot\boldsymbol{\omega}_{\mathrm{o}})(\mathbf{n}\cdot\boldsymbol{\omega}_{\mathrm{i}})}\\
\mathcal{F}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}}) &= s + (1-s)\ 2^{-(5.55473\boldsymbol{\omega}\mathbf{h}+6.8316)\boldsymbol{\omega}_{\mathrm{o}}\mathbf{h})}\\
\mathcal{G}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}}) &= \frac{\boldsymbol{\omega}_{\mathrm{o}}\mathbf{n}}{\boldsymbol{\omega}_{\mathrm{o}}\mathbf{n}(1-k)+k}\ \frac{\boldsymbol{\omega}_{\mathrm{i}}\mathbf{n}}{\boldsymbol{\omega}_{\mathrm{i}}\mathbf{n}(1-k)+k}\\
k &= (R+1)^2/8
\end{aligned} \tag{8}$$

To integrate the rendering equation, $\mathcal{M}$ is assumed to be smooth over $\boldsymbol{\omega}_{\mathrm{i}}$, and is approximated by its value at $\boldsymbol{\omega}_{\mathrm{i}} = 2(\boldsymbol{\omega}_{\mathrm{o}}\cdot\mathbf{n})\mathbf{n}-\boldsymbol{\omega}_{\mathrm{o}}$, which corresponds to the peak of the normalized distribution function $\mathcal{D}$. Putting equations 5, 7 and 8 together, we get

$$\begin{aligned}
f^{\mathrm{s}}(\boldsymbol{\omega}_{\mathrm{o}},\boldsymbol{\omega}_{\mathrm{i}}) &\approx G(\boldsymbol{\omega}_{\mathrm{i}};\mathbf{n},\lambda_{\mathrm{s}},\mu_{\mathrm{s}})\\
\lambda_{\mathrm{s}} &= \frac{1}{2R^4\mathbf{n}\cdot\boldsymbol{\omega}_{\mathrm{o}}}\\
\mu_{\mathrm{s}} &= \mathcal{M}\big(\boldsymbol{\omega}_{\mathrm{o}},2(\boldsymbol{\omega}_{\mathrm{o}}\cdot\mathbf{n})\mathbf{n}-\boldsymbol{\omega}_{\mathrm{o}}\big)/(\pi R^4)
\end{aligned} \tag{9}$$

Finally, the clamped cosine term can also be approximated with a single SG [16]:

$$\boldsymbol{\omega}_{\mathrm{in}}\cdot\mathbf{n} \approx G(\boldsymbol{\omega}_{\mathrm{i}};0.0315,\mathbf{n},32.7080) - 31.7003. \tag{10}$$

Since all the components of the rendering equation are expressed as SGs, it can be integrated in closed form as explained in [16].

[*] This work was done prior to joining Amazon.

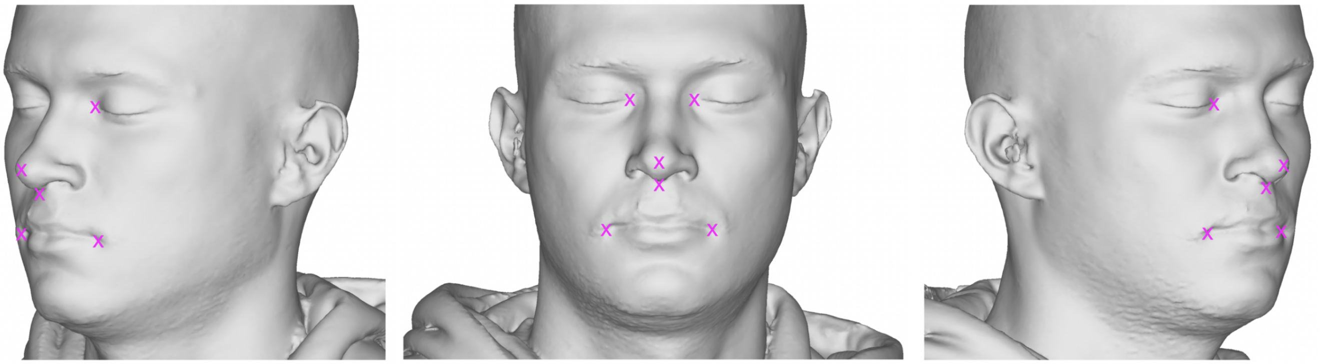

Figure 1: **Six 3D landmark** annotations used in the landmark consistency loss $\mathcal{L}_{\mathrm{Lm}}$. Subject is from the H3Ds dataset.

2. Reconstructing geometry from a single image

In order to optimize $\mathbf{z}_{\mathrm{sa}}$ and $\boldsymbol{\theta}_{\mathrm{sa}}$, we minimize the following loss [17]:

$$\mathcal{L} = \mathcal{L}_{\mathrm{RGB}} + \lambda_{10}\mathcal{L}_{\mathrm{Mask}} + \lambda_{11}\mathcal{L}_{\mathrm{Eik}}, \tag{11}$$

where λ_{10} and λ_{11} are hyperparameters.

We next describe each component of this loss. Let $\mathcal{P}$ be a mini-batch of pixels from image $\mathbf{I}$, $\mathcal{P}_{\mathrm{RGB}}$ the subset of pixels whose associated ray intersects the surface defined by f^{sdf} and which have a nonzero foreground mask value, and $\mathcal{P}_{\mathrm{Mask}} = \mathcal{P} \setminus \mathcal{P}_{\mathrm{RGB}}$. The $\mathcal{L}_{\mathrm{RGB}}$ is the photometric error, computed as $\mathcal{L}_{\mathrm{RGB}} = |\mathcal{P}|^{-1}\sum_{p\in\mathcal{P}_{\mathrm{RGB}}} |\mathbf{I}(p) - \mathbf{c}(p)|$. The loss $\mathcal{L}_{\mathrm{Mask}}$ accounts for silhouette errors, $\mathcal{L}_{\mathrm{Mask}} = \frac{1}{\alpha|\mathcal{P}|}\sum_{p\in\mathcal{P}_{\mathrm{Mask}}} \mathrm{CE}(\mathbf{M}(p), s_\alpha(p))$, where $s_\alpha = \mathrm{sigmoid}(-\alpha \min_{t\geq 0} f^{\mathrm{sdf}}(\mathbf{r}_t))$ is the estimated silhouette, CE is the binary cross-entropy and α is a hyperparameter. Lastly, $\mathcal{L}_{\mathrm{Eik}}$ encourages f^{sdf} to approximate a signed distance function.

3. Implementation Details

We implement equations 1, 2b, 4, 9a, 9b from the main paper, as well as the illumination decoder $\mathbf{l}_{\theta_l,\mathbf{z}_l}$ and the albedo refinement network $\mathbf{a}^r_{\theta_{ar}}$, using MLPs with one skip connection from the input of the network to the input of a hidden layer, as in [11]. We use a SoftPlus activation function in all the hidden layers. We apply positional encoding (PE) [15] to some of the inputs of the networks. Find details in Table 1.

3.1. SA-SM Training

The SA-SM pretraining optimization is iterated for 100 epochs using the Adam optimizer [8] with standard parameters, learning rate of 10^{-4} and learning rate step decay of 0.5 every 15 epochs. The loss hyperparameters are set to $\lambda_1 = 0.1$, $\lambda_2 = \lambda_3 = \lambda_5 = 10^{-3}$, $\lambda_4 = 1$. We automatically annotate six 3D facial landmarks for each scene (Fig. 1), which are used for the landmark consistency loss.

The weigths of the reference SDF network (Eq. 2b) are initialized using the geometric initialization described in [2]. The weights of the deformation and rendering networks are initialized as multivariate gaussians of zero mean and variance 10^{-4}. The latents $\mathbf{z}_{\mathrm{sdf}}$ and $\mathbf{z}_{\mathrm{r}}$ are initialized as zero vectors.

We use a progressive masking of the positional encoding of the input to the reference SDF [10, 12, 6], so as to minimize artifacts on the reference shape and make the training process more stable. Initially masking the higher frequency bands acts as a dynamic low-pass filter, allowing the model to reach robust coarse solutions before adding high-frequency content. At a given step, we compute the parameter $\alpha \in [0, L]$ proportional to the progress of the training, where L is the total number of frequencies used in the PE. The Fourier embedding of frequency k is then multiplied by a scalar $w_k(\alpha)$:

$$w_k(\alpha) = \begin{cases} 0 & \alpha \leq k \\ (1-\cos{(\alpha-k)\pi})/2 & 0 \leq \alpha - k \leq 1 \\ 1 & \alpha - k \geq 1 \end{cases}. \tag{12}$$

We start masking all frequencies in the PE, and unmask them progressively between epochs 20 and 30, by increasing the parameter α linearly from 0 to L.

3.2. AF-SM Training

The AF-SM pretraining optimization is iterated for 600 epochs using the Adam optimizer with standard parameters, learning rate of $5 \cdot 10^{-4}$ and learning rate step decay of 0.5 every 150 epochs. The loss hyperparameters are set to $\lambda_6 = 10^{-3}$, $\lambda_7 = 0.1$ and $\boldsymbol{\epsilon} = 10^{-2}$.

The weights of the diffuse albedo, albedo refinement, specular albedo and light networks are initialized as multivariate gaussians of zero mean and variance 10^{-4}. The la-

Network	Num layers	Layer width	Skip connection index	Input (Dimensions)	Output (Dimensions)	Last activation function
$f^{\mathrm{def}}_{\boldsymbol{\theta}_{\mathrm{def}},\mathbf{z}_{\mathrm{sdf}}}$	5	512	3	$(\mathbf{x}, \mathrm{PE}_6(\mathbf{x}), \mathbf{z}_{\mathrm{sdf}})$ (3, 36, 256)	$(\boldsymbol{\delta}, \boldsymbol{\gamma})$ (3, 128)	sigmoid, -
$f^{\mathrm{ref}}_{\boldsymbol{\theta}_{\mathrm{ref}}}$	3	512	2	$(\mathbf{x}_{\mathrm{ref}}, \mathrm{PE}_6(\mathbf{x}_{\mathrm{ref}}))$ (3, 36)	s, 1	-
$r_{\boldsymbol{\theta}_{\mathrm{r}},\mathbf{z}_{\mathrm{r}}}$	4	512	2	$(\mathbf{x}_{\mathrm{ref}}, \mathbf{v}, \mathrm{PE}_4(\mathbf{v}), \mathbf{n}, \boldsymbol{\gamma}, \mathbf{z}_{\mathrm{r}})$ (3, 3, 24, 3, 128, 128)	$\mathbf{c}$, 3	tanh
$a_{\boldsymbol{\theta}_{\mathrm{a}},\mathbf{z}_{\mathrm{a}}}$	4	512	-	$(\mathbf{x}_{\mathrm{ref}}, \mathrm{PE}_6(\mathbf{x}_{\mathrm{ref}}), \boldsymbol{\gamma}, \mathbf{z}_{\mathrm{a}})$ (3, 36, 128, 128)	$\mathbf{a}$, 3	sigmoid
$a^{\mathrm{r}}_{\boldsymbol{\theta}_{\mathrm{ar}}}$	4	512	2	$(\mathbf{x}_{\mathrm{ref}}, \mathrm{PE}_6(\mathbf{x}_{\mathrm{ref}}), \boldsymbol{\gamma})$ (3, 36, 128)	$\mathbf{a}^r$, 3	tanh
$s^{\mathrm{r}}_{\boldsymbol{\theta}_{\mathrm{s}},\mathbf{z}_{\mathrm{s}}}$	3	256	-	$(\boldsymbol{\gamma}, \mathbf{z}_{\mathrm{s}})$ (128, 64)	$\mathbf{s}$, 1	sigmoid
$l^{\mathrm{r}}_{\boldsymbol{\theta}_{\mathrm{l}},\mathbf{z}_{\mathrm{l}}}$	4	512	-	$\mathbf{z}_{\mathrm{l}}$, 128	$\mathbf{l}$, 128×5	-

Table 1: Implementation details of the architecture. We include the SA-SM and the AF-SM models. $(\cdot)$ denotes concatenation. The number of frequencies k in the positional encoding is denoted as PE_{k}.

tents $\mathbf{z}_a$, $\mathbf{z}_s$ and $\mathbf{z}_l$ are also initialized as zero vectors. However, in some networks we modify the biases of the last layer so that the initial output is different from 0. We set a bias of 0.55 in the diffuse and specular albedo networks. The light network is initialized to output 128 uniformly distributed lobes on a sphere, using the Fibonacci sphere algorithm [7].

3.3. Reconstructing geometry from a single image

At test time, the 3D reconstruction of a scene is done over 2000 epochs using Adam with initial learning rate of 10^{-4} and learning rate step decay of 0.5 at epochs 1000 and 1500. The parameter α in the mask loss $\mathcal{L}_{\mathrm{Mask}}$ is scheduled as in [17]. We use a two-step scheduling where the weights of the deformation and rendering networks are unfrozen at epoch 100.

3.4. Appearance factorization from a single image

The appearance factorization process is done over 10000 epochs using Adam with initial learning rate of $5 \cdot 10^{-4}$ and learning rate step decay of 0.5 at epoch 7000. The loss hyperparameters are set to $\lambda_8 = 3$ and $\lambda_9 = 2$.

We use a scheduling (Fig. 2) suitable for the task of appearance factorization using the pre-learnt AF-SM prior. We initialize the appearance factorization networks with the parameters of the pretrained AF-SM, $\{\boldsymbol{\theta}_{\mathrm{a},0}, \boldsymbol{\theta}_{\mathrm{s},0}, \boldsymbol{\theta}_{\mathrm{l},0}\}$. The parameters of the albedo refinement network, $\boldsymbol{\theta}_{\mathrm{ar}}$, are initialized to yield a zero-mean, low-amplitude random refinement field. The initial latent vectors, $\mathbf{z}_{\mathrm{pb}}$ are picked from a multivariate normal distribution with zero mean and small variance. The vectors $\mathbf{x}_{\mathrm{ref}}$ and $\boldsymbol{\gamma}$ are obtained by evaluating the function f^{def} optimized in the 3D reconstruction step.

During the first epochs, as we begin to minimize $\mathcal{L}$, we freeze all parameters except for the lighting decoder weights and latent vector, $\boldsymbol{\theta}_{\mathrm{l}}$ and $\mathbf{z}_{\mathrm{l}}$. We also disable the specular radiance ($k_{\mathrm{s}} = 0$), as well as the albedo refinement ($k_{\mathrm{r}} = 0$). In this stage, the model learns initial lights to recover coarse shadows. Next, we unfreeze the albedo latent vector $\mathbf{z}_{\mathrm{a}}$, allowing the model learn an albedo within the latent space which matches basic features of the scene, like skin tone and hair color. We do not unfreeze $\boldsymbol{\theta}_{\mathrm{a}}$ in order to prevent baking of unwanted information into the albedo. After this, we gradually enable the specular radiance ($k_{\mathrm{s}} = 1$) and unfreeze the weights and latent vector of the specular albedo decoder, $\boldsymbol{\theta}_{\mathrm{s}}, \mathbf{z}_{\mathrm{s}}$. At this stage, the model adjustes the specular albedo and lights to explain reflections. Finally, we enable the training of the albedo refinement by setting $k_{\mathrm{r}} = 0.5$ and unfreezing the parameters $\boldsymbol{\theta}_{\mathrm{ar}}$ to allow for corrections of up to ± 0.5 on the base albedo network $\mathbf{a}_{\theta_a,\mathbf{z}_a}$. Simultaneously, we freeze the latent of the diffuse albedo decoder, $\mathbf{z}_{\mathrm{a}}$. During this stage, the model captures photo-realistic details in the albedo refinement field, while avoiding baking shades and reflections thanks to $\mathcal{L}_{\mathrm{Reg}}$.Lastly, we let the albedo network be finetuned for the last 50 epochs to make small adjustments to

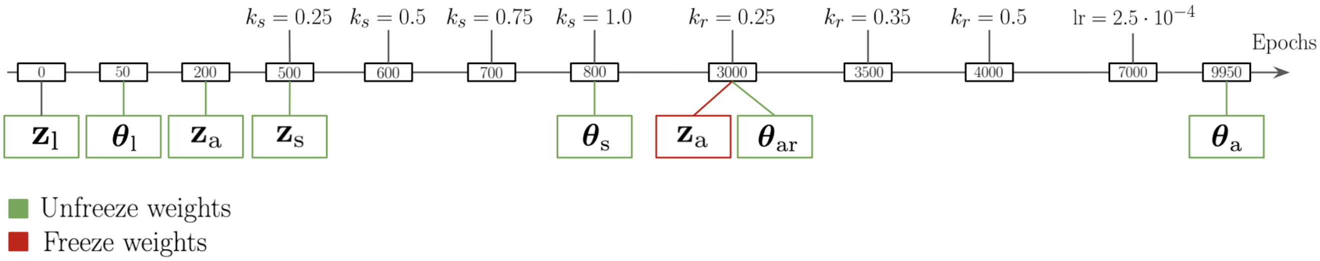

Figure 2: **Scheduling** during the appearance factorization decomposition. At the beginning all the network weights and latent vectors are frozen. The initial learning rate is set to $5 \cdot 10^{-4}$ and the k_r and k_s parameters to 0.

the global color.

4. Datasets

Prior training. We train the SA-SM and AF-SM on a non-released dataset [14] made of 3D head scans and corresponding posed images from 10,000 individuals with an average of 6 photos per scene. The scans are low resolution, incomplete and non-watertight. The dataset is perfectly balanced in gender and diverse in age and ethnicity.

3DFAW. Videos of human heads paired with 3D reconstructions of the facial area, at two different resolutions. We select from the low-resolution set the same 10 cases as in [14], and 17 subjects from the high-resolution set provided by [4].

H3DS [14]. 23 human head scenes with multi-view posed images, masks, and full-head 3D textured scans. The dataset consists of 13 men and 10 women.

Wikihuman Project. We evaluate our appearance factorization method on the Digital Emily scene [1]. In our comparisons we only use the diffuse and specular albedos, and the frontal photo, as in [4].

5. Evaluation details

5.1. 3D reconstruction

The predicted 3D reconstruction for all methods is roughly aligned with the ground truth mesh using manually annotated landmarks, and then refined with rigid ICP [3]. Surface error is computed as the unidirectional Chamfer distance from the reconstruction to the ground truth. For a fair comparison, all methods are evaluated on the same face region. This is defined by cutting both the reconstructions and the ground truth using a sphere of 95 mm radius and with center at the tip of the nose of the ground truth mesh, and refining the alignment with ICP.

5.2. Appearance factorization

We first find the intersection mask among all the evaluated methods. We use it to mask the final render, as well as diffuse and specular albedos provided by each baseline, in order to provide results evaluated on the same pixels. Due to the existing scale ambiguity in inverse rendering problems, for each method we apply a scaling that minimizes the mean squared error channel-wise between the ground truth and the predicted images.

6. Additional results

We provide extended results on our ablation study in figures 3, 4. Figure 5 shows additional results on our 3D reconstruction method. In figures 6 and 7 we provide extended results on our appearance factorization method. Figure 8 shows extended relightning results and in figure 9 we provide diverse results in the Celeb-HQ dataset.

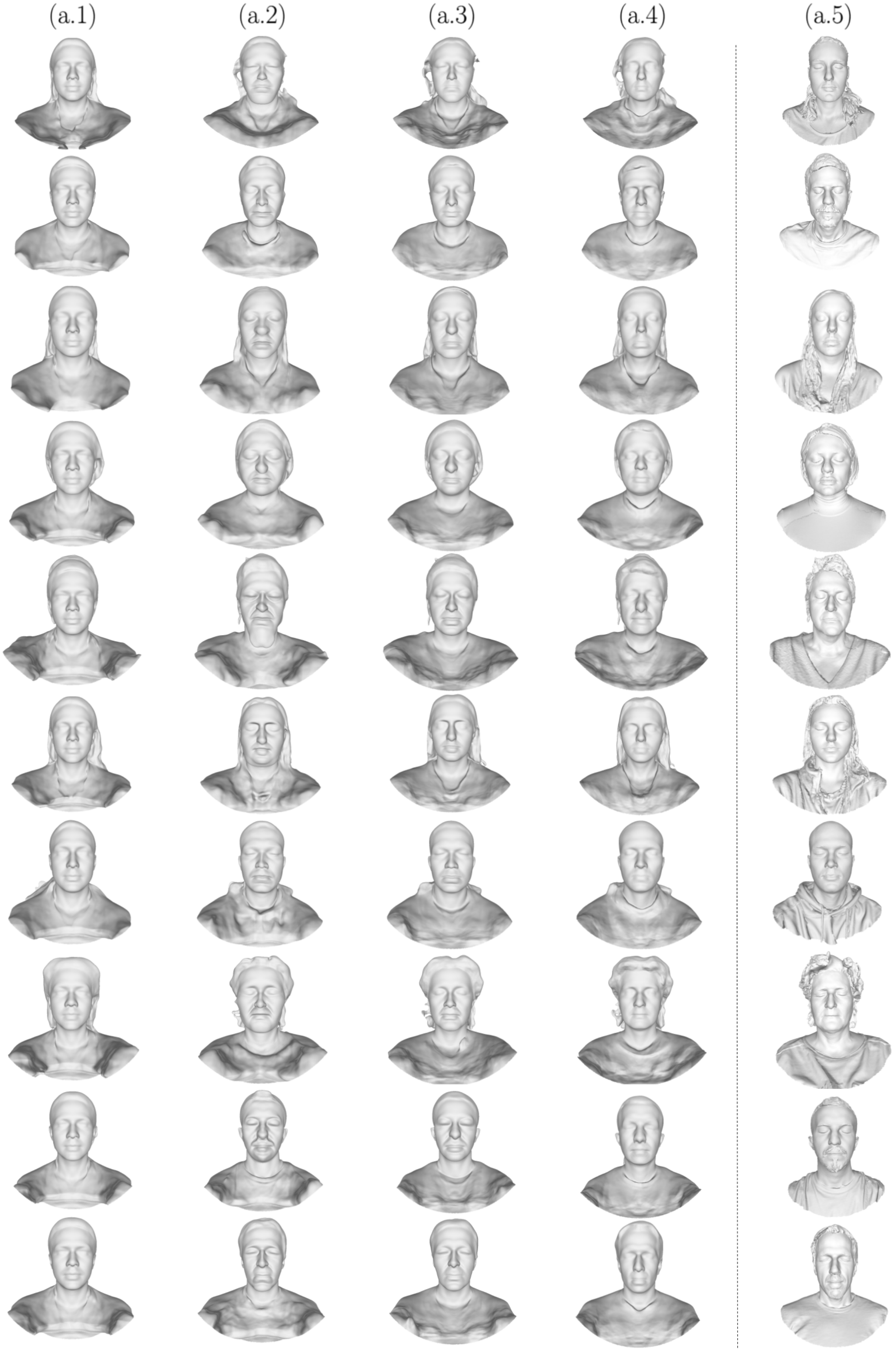

Figure 3: **Ablation study:** We ablate the 3D reconstruction method. Extension of Figure 4a from the main paper.

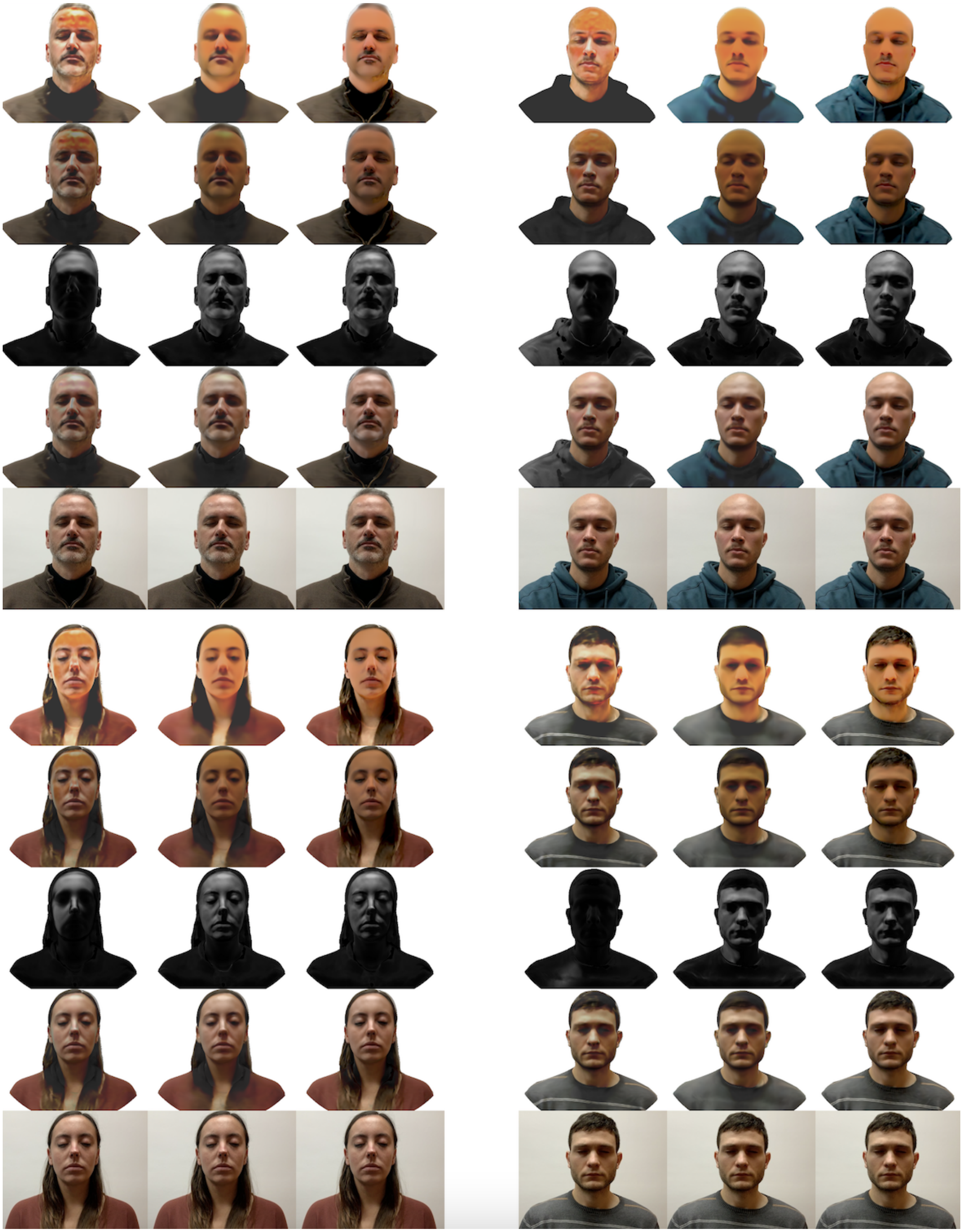

Figure 4: **Ablation study:** We ablate the appearance factorization method on 4 subjects from the H3DS dataset. The 1st and 4th columns correspond to (b.1), the 2nd and 5th columns to (b.2) and the 3rd and 6th columns to (b.3). Extension of Figure 4b from the main paper.

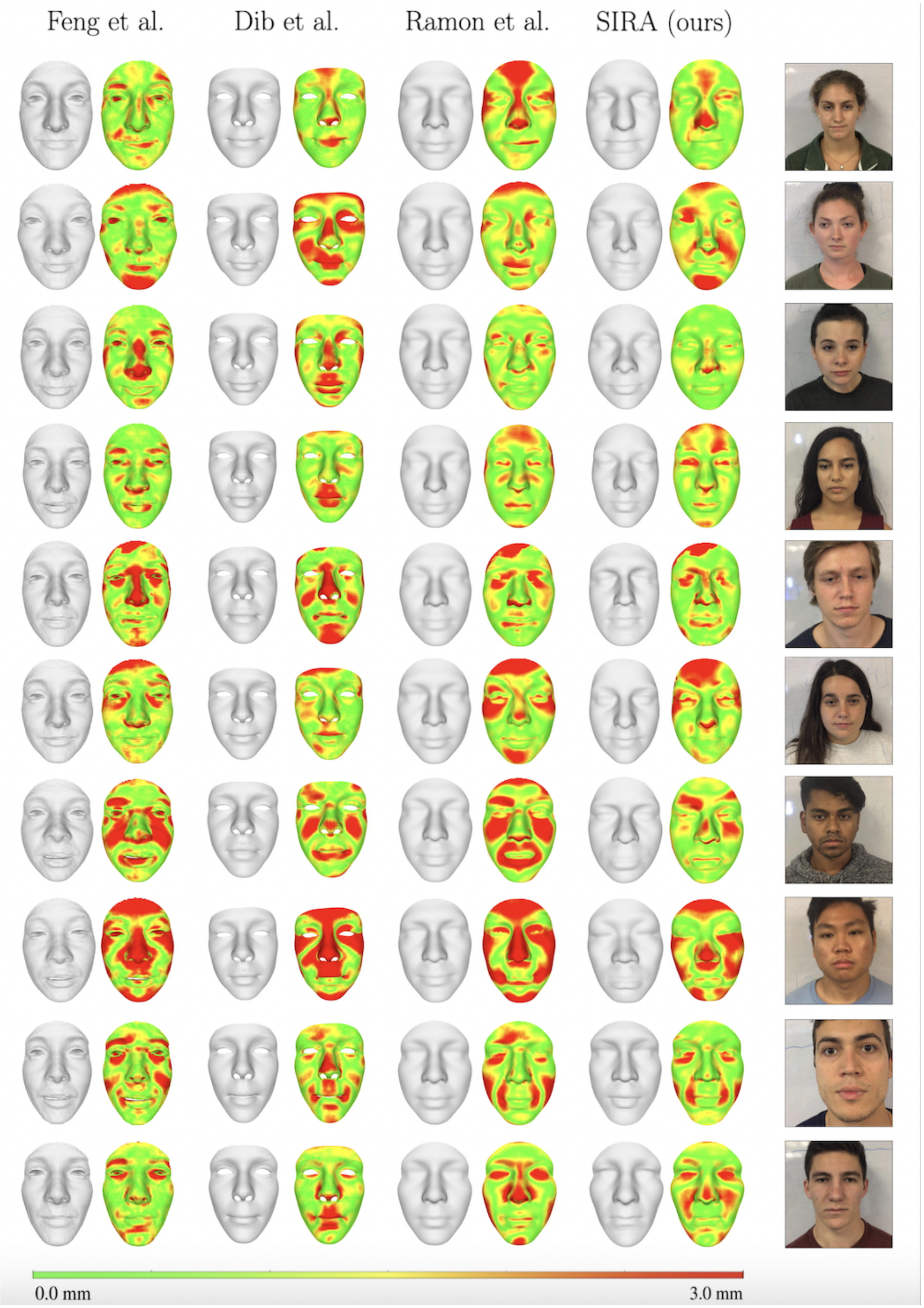

Figure 5: **Single-view 3D reconstruction:** Subjects from 3DFAW dataset [13]. Comparison: Feng 2021 [5], Dib 2021 [4], Ramon 2021 [14], SIRA (ours). Extension of Figure 5 from the main paper.

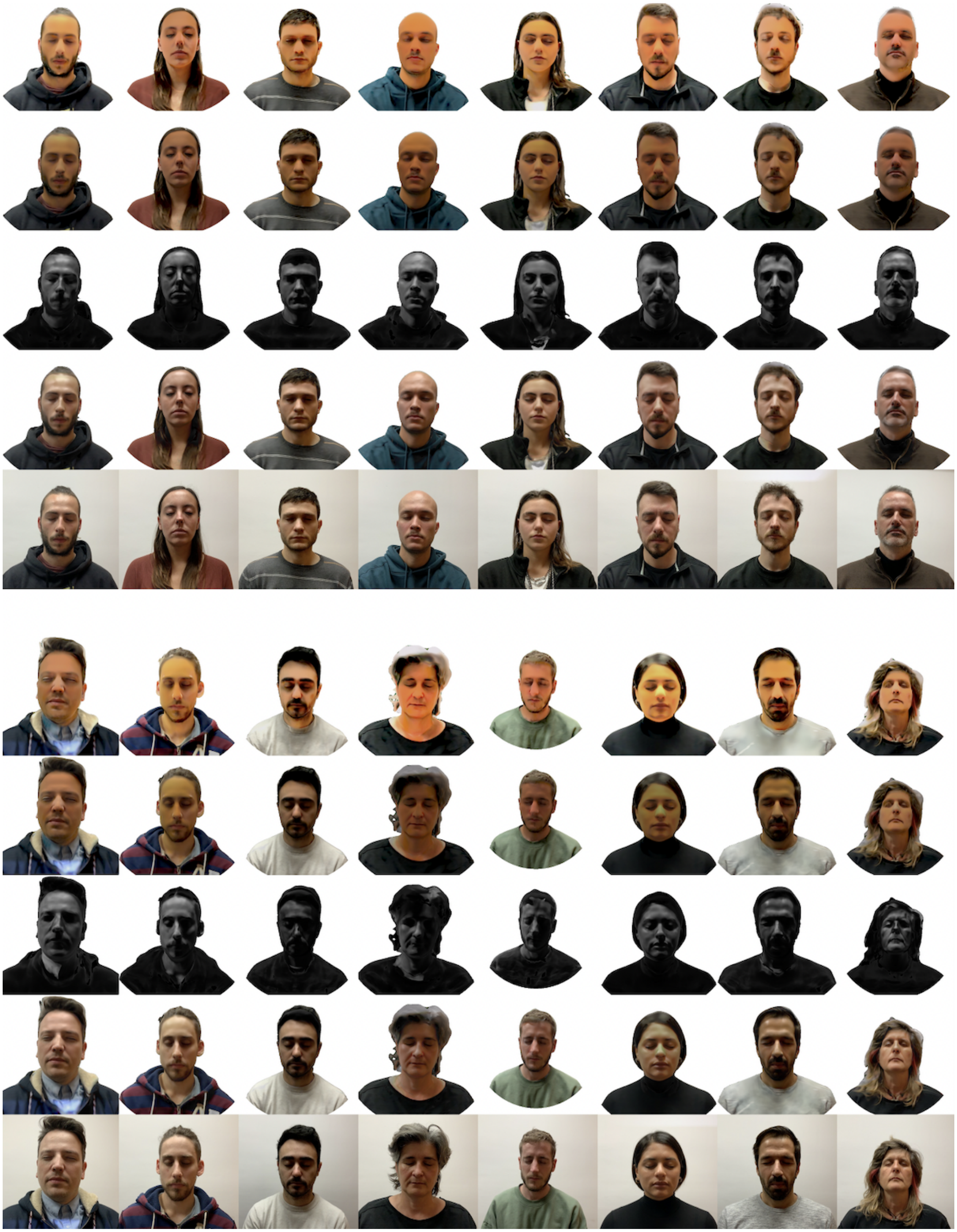

Figure 6: **Appearance factorization:** 16 subjects from H3DS [14] dataset. Input images (5th and 10th rows) are decomposed into diffuse albedo (1st and 6th rows), diffuse radiance (2nd and 7th rows), specular radiance (3rd and 8th rows), and final render (4th and 9th rows). Extension of Figure 7 from the main paper.

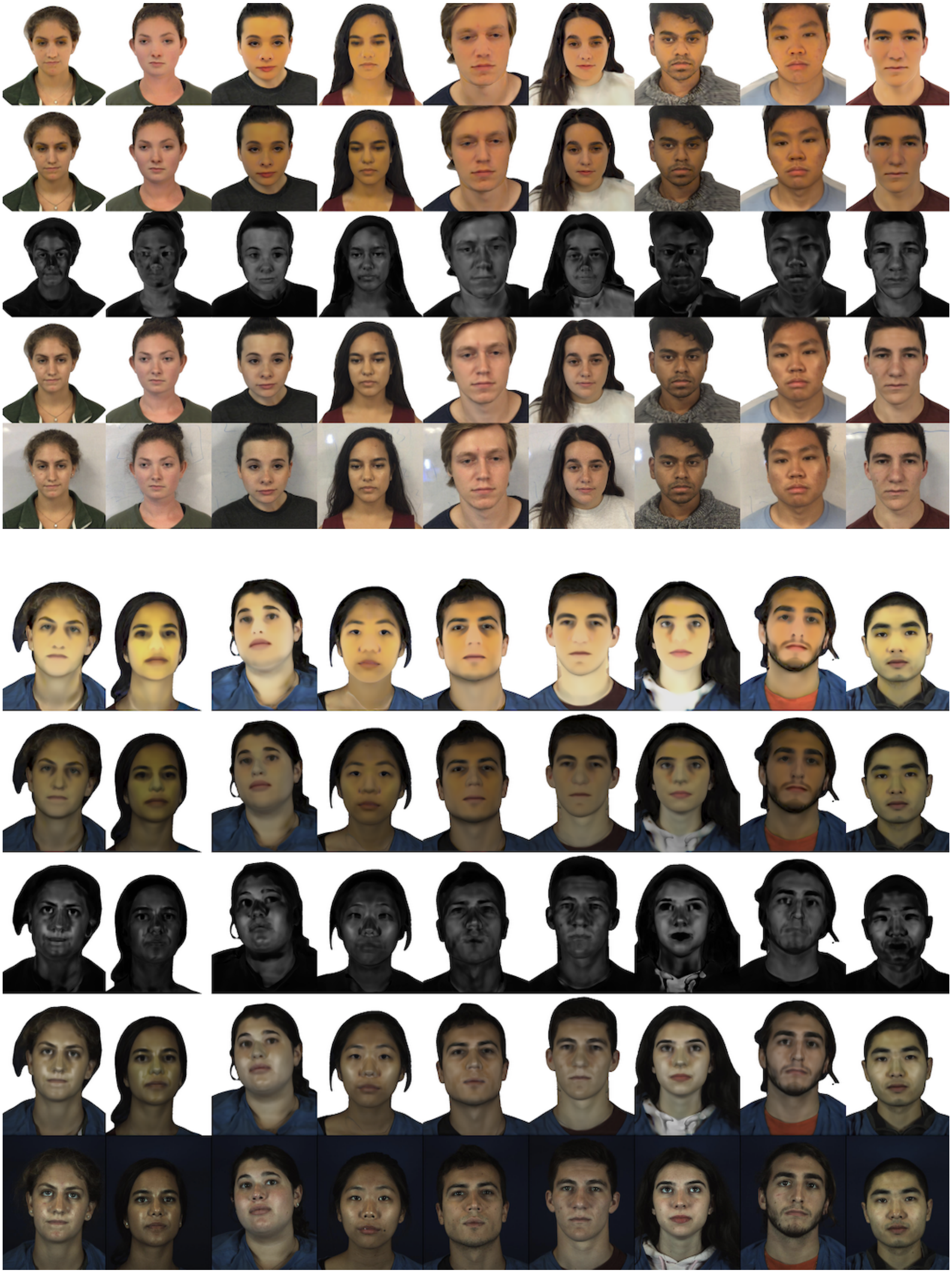

Figure 7: **Appearance factorization:** 18 subjects from 3DFAW [13] dataset. Rows 1-5 are from the low resolution subset and rows 6-10 are from the high resolution one. Input images (5th and 10th rows) are decomposed into diffuse albedo (1st and 6th rows), diffuse radiance (2nd and 7th rows), specular radiance (3rd and 8th rows), and final render (4th and 9th rows). Extension of Figure 7 from the main paper.

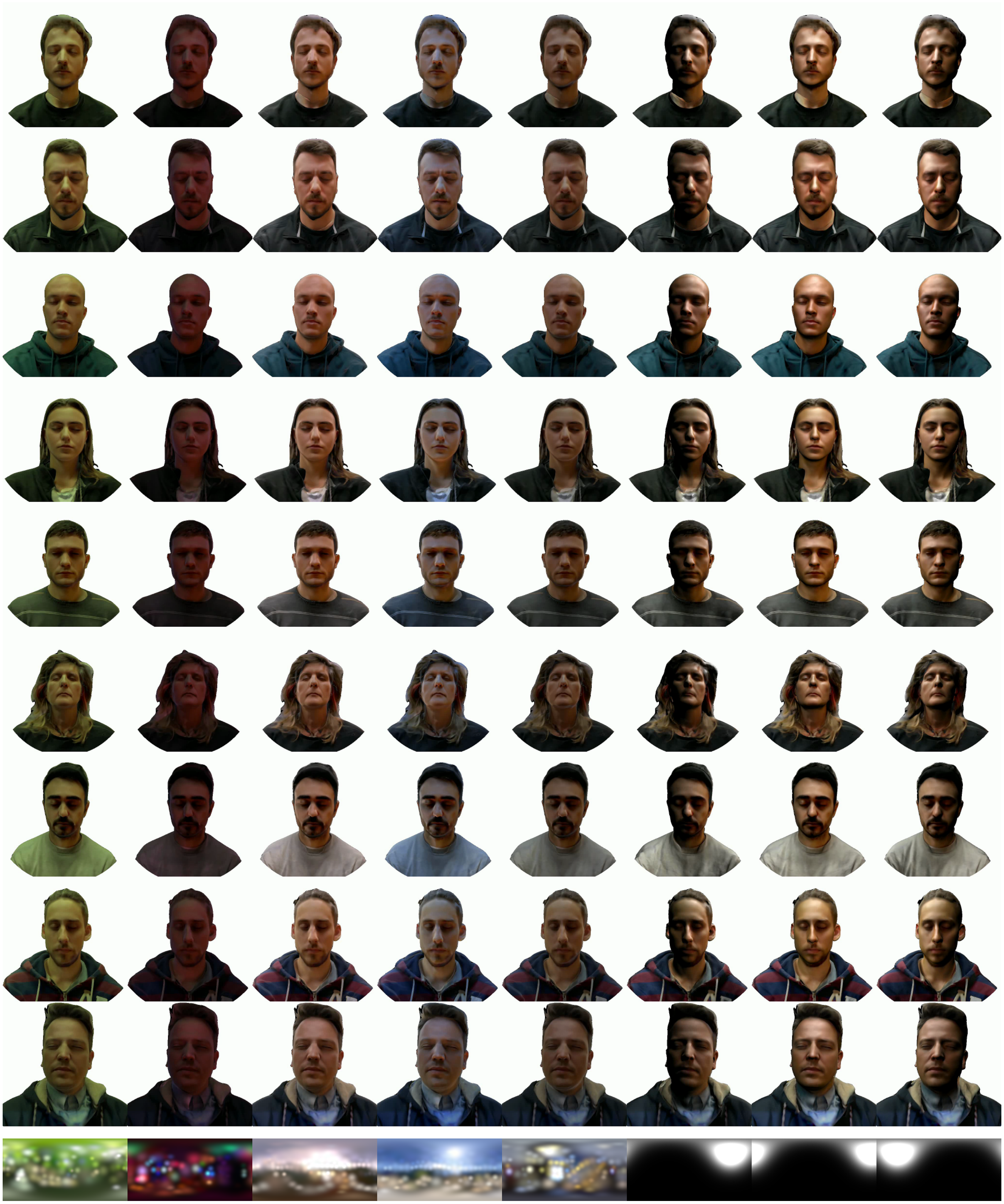

Figure 8: **Relighting** of inverse-rendered scenes. Subjects from the H3DS dataset. Rows 1-9 are relightings from diferent subjects. Row 10 is the aproximation with 128 Spherical Gaussians (SG) of different environment maps. Extension of Figure 8 from the main paper.

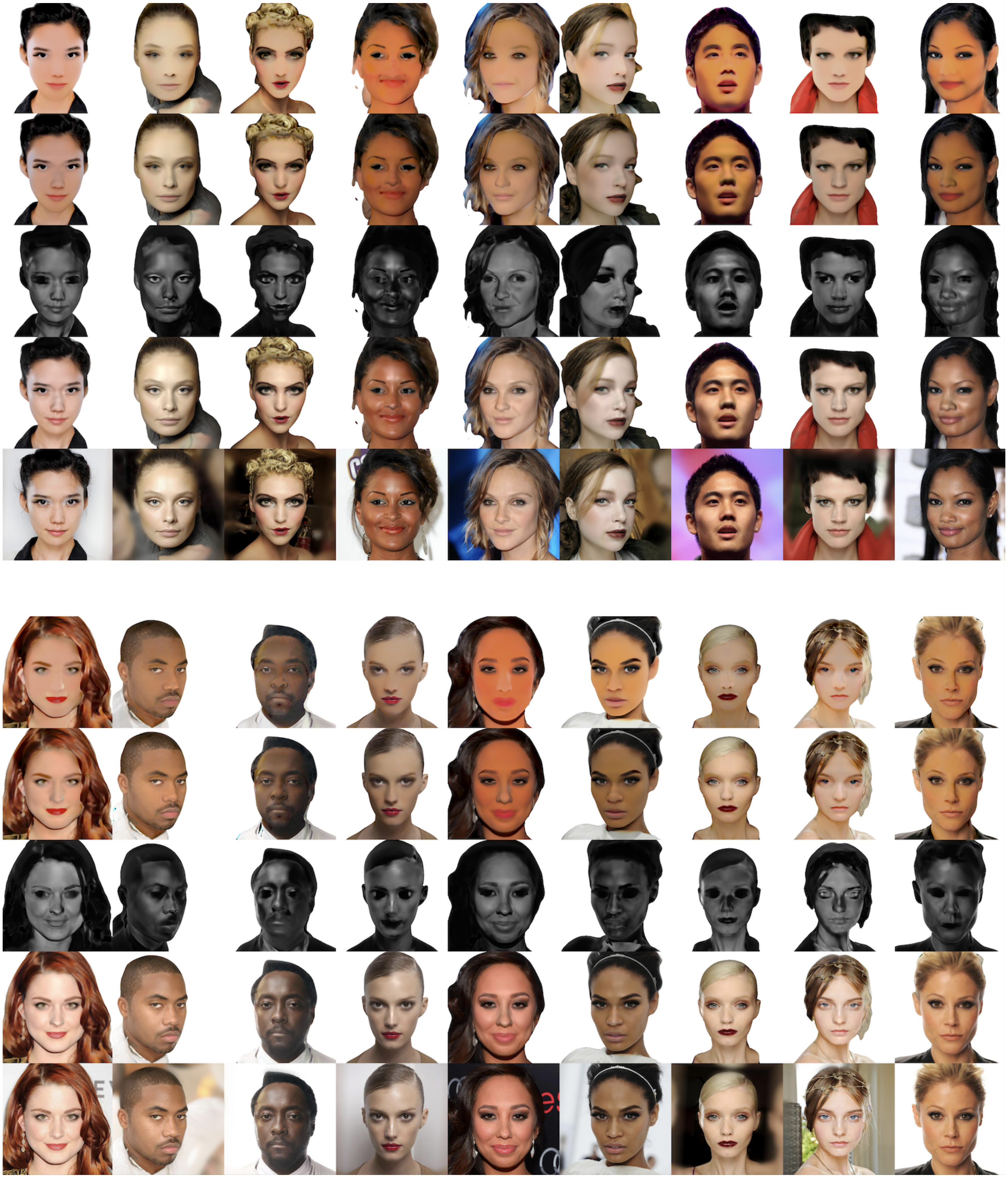

Figure 9: **Appearance factorization:** 18 subjects from Celeb-HQ [9] dataset. Input images (5th and 10th rows) are decomposed into diffuse albedo (1st and 6th rows), diffuse radiance (2nd and 7th rows), specular radiance (3rd and 8th rows), and final render (4th and 9th rows).

References

[1] Emily. the wikihuman project. https://vgl.ict.usc.edu/Data/DigitalEmily2/. Accessed: 2022-03-05.

[2] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In CVPR, 2020.

[3] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In Sensor fusion IV: control paradigms and data structures, volume 1611, pages 586–606. Spie, 1992.

[4] Abdallah Dib, Cedric Thebault, Junghyun Ahn, Philippe-Henri Gosselin, Christian Theobalt, and Louis Chevallier. Towards high fidelity monocular face reconstruction with rich reflectance using self-supervised learning and ray tracing. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 12819–12829, 2021.

[5] Yao Feng, Haiwen Feng, Michael J Black, and Timo Bolkart. Learning an animatable detailed 3d face model from in-the-wild images. ACM Transactions on Graphics (TOG), 40(4):1–13, 2021.

[6] Amir Hertz, Or Perel, Raja Giryes, Olga Sorkine-Hornung, and Daniel Cohen-Or. Sape: Spatially-adaptive progressive encoding for neural optimization. In Thirty-Fifth Conference on Neural Information Processing Systems, 2021.

[7] Benjamin Keinert, Matthias Innmann, Michael Sänger, and Marc Stamminger. Spherical fibonacci mapping. ACM Transactions on Graphics (TOG), 34(6):1–7, 2015.

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[9] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[10] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. arXiv preprint arXiv:2104.06405, 2021.

[11] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[12] Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 5865–5874, October 2021.

[13] Rohith Krishnan Pillai, László Attila Jeni, Huiyuan Yang, Zheng Zhang, Lijun Yin, and Jeffrey F Cohn. The 2nd 3d face alignment in the wild challenge (3dfaw-video): Dense reconstruction from video. In ICCV Workshops, 2019.

[14] Eduard Ramon, Gil Triginer, Janna Escur, Albert Pumarola, Jaime Garcia, Xavier Giro-i Nieto, and Francesc Moreno-Noguer. H3d-net: Few-shot high-fidelity 3d head reconstruction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 5620–5629, 2021.

[15] Matthew Tancik, Pratul P Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. arXiv preprint arXiv:2006.10739, 2020.

[16] Jiaping Wang, Peiran Ren, Minmin Gong, John Snyder, and Baining Guo. All-frequency rendering of dynamic, spatially-varying reflectance. In ACM SIGGRAPH Asia 2009 papers, pages 1–10. 2009.

[17] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. Advances in Neural Information Processing Systems, 33:2492–2502, 2020.

[18] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5453–5462, 2021.