with k_s being the spatially-varying specular gain and α a global roughness blend factor for the Blinn-Phong distribution term D of the 2-lobe mix (D_{12} and D_{48}) suggested by [37]. G denotes the geometry term of the Cook-Torrance BRDF model. We use Shlick's approximation [40] for the Fresnel term F:

$$F(\mathbf{n}, \omega) = F_0 + (1 - F_0)(1 - \mathbf{n}^\top \omega)^5. \quad (3)$$

To model the skin's diffuse response, we implement the BRDF model proposed by Ashikhmin and Shirley [2, 3], that accounts for the fact that a portion of the light has already scattered before penetrating the skin surface:

$$f_d(\mathbf{x}, \omega) = \frac{28 k_d(\mathbf{x})}{23\pi}(1 - F_0)(1 - (1 - \frac{\mathbf{n}^\top \omega}{2})^5)^2, \quad (4)$$

where $F_0 = 0.04$ is the reflectance of the skin at normal incidence. Indirect light bouncing from the capture environment and on the captured face itself might have a significant contribution to pixel intensity at grazing angles, so we also add a Fresnel-modulated ambient term to our BRDF f:

$$f_a(\mathbf{x}, \omega) = k_a(\mathbf{x})(1 - (1 - F_0)(1 - (1 - \frac{\mathbf{n}^\top \omega}{2})^5)^2), \quad (5)$$

with an ambient map k_a which is regularized to be smooth via a total variation loss and close to zero.

Note that using a diffuse scattering model for the optimization is compatible with state-of-the-art physically-based subsurface scattering skin shading [5, 49], as shown in Figure 1. Production-ready subsurface scattering models typically include an albedo inversion stage, which takes a diffuse albedo as input, and converts it to extinction coefficients for the volume rendering random walk.

3.4. Optimization

The objective of the photometric optimization step is to minimize the difference between rendered images $\hat{I}$ and color-corrected target images I:

$$\mathcal{L}(\hat{I}, I) = \left| W \cdot \left(\hat{I} - I \right) \right|, \quad (6)$$

with $\hat{I} = \mathcal{M} \cdot L_o$, where $\mathcal{M}$ is the pre-computed light attenuation map, that accounts for uneven light distribution in different directions. We apply a per-pixel loss weight W based on the respective mip level and the angle between viewing direction and normal $\mathbf{n}^\top \omega$ to improve sharpness. Specifically, to ensure that distant or grazing angle observations do not blur the resulting textures, for each pixel that is projected from the target image to texture space, we calculate which mip level l would need to be looked up in classical forward rendering. W is set to $(\mathbf{n}^\top \omega)(1 - l)$ if the pixel corresponds to a mip level below 1, and zero otherwise.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
NLT [55]	31.51	**0.96**	0.11
NextFace [9]	22.85	0.89	0.31
Ours	**32.37**	**0.96**	**0.10**

Table 1. We compare our method to NLT and NextFace on validation frames over 10 different subjects.

We optimize $\mathcal{L}(\hat{I}, I)$ in two steps, using a coarse-to-fine optimization strategy in each. In the first step, we only use the cross-polarized images to optimize the spatially-varying diffuse albedo texture $k_d(\mathbf{x})$ and an initial tangent-space normal map $n(\mathbf{x})$, while assuming $f_s(\cdot) = 0$ for the specular term. In the second step, we fix the diffuse texture and optimize for specular gain $k_s(\mathbf{x})$, specular roughness α, and the final normal map $n(\mathbf{x})$. To account for potentially different light attenuation in the cross and parallel-polarized filter settings, we also optimize per-channel scaling factors for the diffuse texture. The optimization is performed entirely in texture space. In each step, we employ a four-level coarse-to-fine optimization strategy, starting with a texture resolution of 512×512, and increasing the size by a factor of two after convergence of each level, up to the final resolution of 4096×4096.

We implement our optimization framework in PyTorch, using nvdiffrast [24] as our differentiable renderer. We optimize on batches of 4 images, using Adam with an initial learning rate $lr_0 = 10^{-3}$ for all parameters at the beginning of every coarse-to-fine step, and updating it to $lr = lr_0 \cdot 10^{-0.001t}$ in every iteration t. We scale the FLAME mesh to unit size and set the light intensity to 10. The total optimization time is about 90 minutes.

4. Results

In this section, we present texture reconstruction and rendering results on several subjects. Figure 5 shows the texture reconstruction on several actors of different ethnicity. Our method is able to reconstruct pore-level detail in the diffuse, specular and normal maps. Further, we evaluate the quality of our reconstructed textures by rendering the mesh from novel views and under novel illumination. Figure 6 shows that our method faithfully reconstructs the skin's appearance under novel views and lighting. *We recommend the reader to use the zoom function of the PDF viewer to inspect the details.*

Comparison to state of the art. We perform both a qualitative and quantitative evaluation of our method and compare to state-of-the-art methods for relighting and texture reconstruction. During optimization, we hold out a validation frame on which we compute image metrics.

Neural Light Transport. Neural Light Transport [55] is a deep learning-based method that takes as input pre-

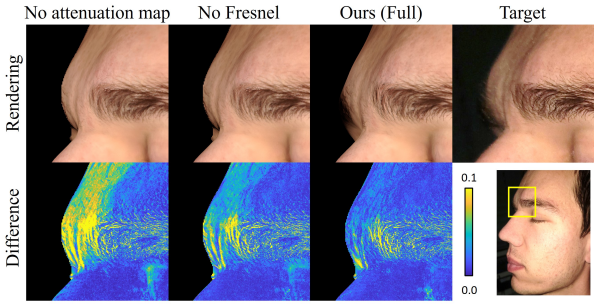

Figure 8. Ignoring the angle-dependent flashlight attenuation or the Fresnel effect leads to an incorrect diffuse map reconstruction, that can no longer reproduce the shading from all views. We account for both to closely match the target data.

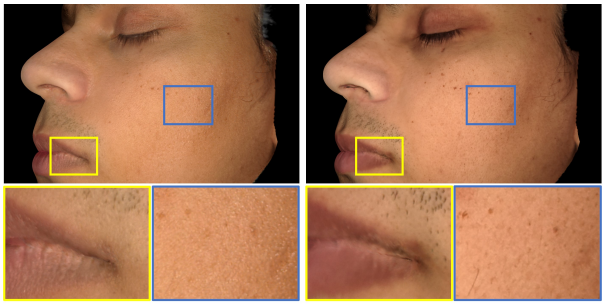

Figure 9. We run a joint optimization of all textures and compare a purely diffuse render to our full approach. As visible in the figure, optimizing the textures jointly will leak specular and normal map information into the diffuse texture.

skin's diffuse response at all angles. In Figure 9, we show that optimizing textures without cross-polarization will leak specular information into the diffuse texture.

Coarse-to-fine optimization and mipmapping. The pixels of the target images may have different footprints in uv-space, depending on distance and angle between the camera and the surface. Weighting the loss of each pixel equally would lead to blurriness in the reconstructed textures. Optimizing coarse-to-fine, where at each resolution we use only pixels with the corresponding uv-space footprint, helps us reconstruct additional detail in the textures. Figure 10 shows a comparison between our full approach and one where we directly optimize the highest resolution texture. We additionally show the decrease in quality when optimizing only on the video frames (w/o photographs).

Runtime and memory consumption. Including one hour that is spent on MVS, our method takes about 2.5h to reconstruct a person's face. The photo-metric skin texture reconstruction takes about 90 minutes on an Nvidia RTX A6000. We reconstruct the facial geometry with Metashape using an average of 420 video frames and 70 photographs. At a texture resolution of 4096×4096 and target image

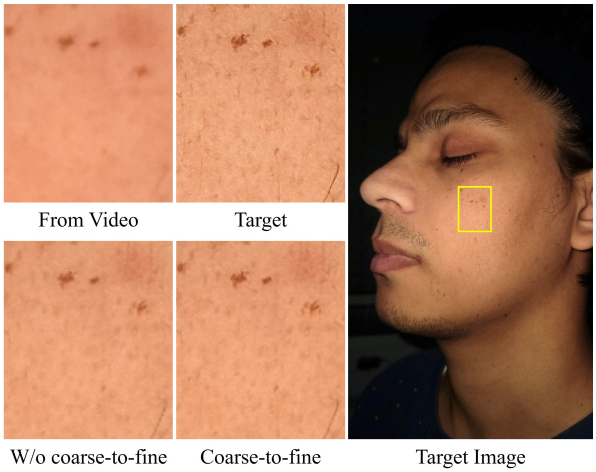

Figure 10. Optimizing from video frames produces blurry textures. To increase sharpness, we optimize from photographs in a coarse-to-fine manner, where at each texture resolution we consider only the input pixels with the appropriate mip level.

resolution of 3840×2160, the photometric optimization requires 30GB of GPU memory. In comparison, NLT takes about 10h and NextFace about 6h given the same number of frames that our method uses.

Discussion & Limitations. Our method is able to reconstruct high-quality face textures with a low-cost capture routine. However, our method is restricted to static expressions, i.e., it does not handle dynamically changing face geometry and textures. An avenue for future research is the reconstruction of dynamic expressions by fitting a parametric model with consistent mesh topology to each frame, and optimizing over the entire non-rigid sequence. Our method does not explicitly handle global illumination and indirect light bounces. The use of a differentiable path tracer could potentially improve results in the concavities of the eye region. As we assume a static face with closed mouth and closed eyes, we only recover the skin area of a face. Eyes, mouth interior and hair are a subject of future work.

5. Conclusion

We have presented a practical and inexpensive method of capturing high-resolution textures of a person's face by coupling commodity smartphones and polarization foils. The co-location of the camera lens and light source allows us to reduce the problem complexity and separate material from shading information. As a result, we obtain high-resolution textures of the skin area of the human face. We believe that polarization is a powerful tool for material recovery in the real world, and future smartphones could benefit from including filters directly in the hardware. Overall, we believe that our work is a stepping stone towards democratizing the creation of digital human face assets by making it more accessible to smaller production studios or individual users.

REFERENCES

[2] Michael Ashikhmin and Peter Shirley. An anisotropic phong light reflection model. *University of Utah Computer Science Technical Report*, 2000. 5

[3] Michael Ashikhmin and Peter Shirley. An anisotropic phong brdf model. *Journal of Graphics Tools 5 (2)*, 25–32, 2002. 5

[5] Matt Jen-Yuan Chiang, Peter Kutz, and Brent Burley. Practical and controllable subsurface scattering for production path tracing. In *ACM SIGGRAPH 2016 Talks*, pages 1–2. 2016. 2, 5

[9] Abdallah Dib, Junghyun Ahn, Cedric Thebault, Philippe-Henri Gosselin, and Louis Chevallier. S2f2: Self-supervised high fidelity face reconstruction from monocular image. *arXiv preprint arXiv:2203.07732*, 2022. 3, 5, 7

[24] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6), 2020. 3, 5

[37] Jérémy Riviere, Paulo Gotardo, Derek Bradley, Abhijeet Ghosh, and Thabo Beeler. Single-shot high-quality facial geometry and skin appearance capture. *ACM Trans. Graph.*, 39(4), jul 2020. 2, 5

[40] Christophe Schlick. An inexpensive brdf model for physically-based rendering. *Computer Graphics Forum*, 13(3):233–246, 1994. 5

[49] Magnus Wrenninge, Ryusuke Villemin, and Christophe Hery. Path traced subsurface scattering using anisotropic phase functions and non-exponential free flights. Technical report, Tech. Rep. 17-07, Pixar. https://graphics.pixar.com/library . . . , 2017. 2, 5

[55] Xiuming Zhang, Sean Fanello, Yun-Ta Tsai, Tiancheng Sun, Tianfan Xue, Rohit Pandey, Sergio Orts-Escolano, Philip Davidson, Christoph Rhemann, Paul Debevec, Jonathan T. Barron, Ravi Ramamoorthi, and William T. Freeman. Neural light transport for relighting and view synthesis. *ACM Trans. Graph.*, 40(1), jan 2021. 3, 5, 7