






Stylize My Wrinkles: Bridging the Gap from Simulation to Reality Supplementary Material

S. Weiss*¹  and J. Stanhope*²  and P. Chandran¹  and G. Zoss¹  and D. Bradley¹ 

¹ DisneyResearchStudios, Switzerland ² ETH Zürich

Authors marked with * contributed equally

A Light Source Estimation

In Section 4.2 of the main paper we introduced the light source estimation used in the proposed patch scanning setup. The idea is to position two mirror spheres next to the cutout for the skin patch and having two witness cameras that observe both mirror spheres. We assume that the camera intrinsics and extrinsics are calibrated and the size and location in space of the mirror spheres are known. We now provide additional detail on how to estimate the light source position.

Each of the two witness cameras sees one reflection of the light in each of the two mirror spheres. With blob detection we find the center pixel of each reflection and then project it onto the surface of the sphere. Let O_i be the point of reflection in 3D space, N_i the normal vector of the sphere at that location, and V_i the directions from the mirror sphere to the camera. Then, the direction from the mirror sphere intersection to the light source is given as

$$L_i = 2(N_i \cdot V_i)N_i - V_i. \quad (1)$$

Next, we are searching for the point \hat{P} that minimizes the distance to the four light rays $R_i = O_i + t_i L_i$ (two cameras with two reflections each). Chen *et al.* [CGS06] chose to optimize parameters t_i using SVD. Instead, we opt to minimize the distance of point \hat{P} from the light rays R_i for $i = 1, 2, \dots$ directly by solving the following least squares problem, to directly express the distance to be minimized:

$$\hat{P} = \operatorname{argmin}_P \|AP - b\|^2 \quad (2)$$

$$\text{with } A = \sum_{i=1}^4 \mathbb{1} - L_i L_i^T, \quad (3)$$

$$b = \sum_{i=1}^4 (\mathbb{1} - L_i L_i^T) O_i. \quad (4)$$

The optimized light source position \hat{P} is found for each captured frame, which gives us a light direction for each captured image of a skin patch. These light directions are important for conducting photometric stereo in a later step.

B Simulation Fitting

In the following we extend upon the classification and generation architecture introduced in Section 5.2 and present additional qualitative and quantitative evaluations.

B.1 Network Architecture Details

Dataset Creation. To train the classification and generation networks used to fit simulation parameters from scanned displacement maps, we create a database of 50,000 simulated displacement maps of resolution 512^2 . We found that using two levels in the simulation leads to the best result. Only one level did not add sufficient detail to match real scans and three levels did not add a noticeable improvement.

To keep the combinatoric complexity in bounds, we randomly sample parameter values from a subset of 11 parameters of all available parameters in the simulation that we found have the largest impact on the appearance of the wrinkles. We refer to Weiss *et al.* [WMC*23] for a detailed description of each of those parameters. First, we sample the pore distance of the first simulation level. The distance at the second level is fixed to half of the distance of the first level. Second and third, the primary orientation of wrinkles and the strength of the directionality compared to a uniform orientation distribution is encoded using two scalars $uv \in \mathbb{R}^2, \sqrt{u^2 + v^2} \leq 1$. The angle is the given as $\alpha_\theta = \text{atan2}(v, u)$ and the uniform orientation strength as $\alpha_s = 1 - \sqrt{u^2 + v^2}$. The other parameters that are sampled and optimized are the noise strength and frequency $\alpha_{\text{fnoise}}, \alpha_{\text{fscale}}$, the pore and wrinkle width $\alpha_{\text{pore-width}}, \alpha_{\text{wrinkle-width}}$, the pore blending factor α_{blend} , the pore skewing factor α_{skew} , and the simulation continuation weight and distance exponent $\alpha_{\text{cont}}, \alpha_{\text{dist}}$. All other parameters are fixed to their default values. We refer to Appendix C and Fig. 6 for examples of fitted results and values of all the parameters.

Classification. For the classification, we build upon the VGG-11 architecture by Simonyan *et al.* [SZ14] with an added global average pooling layer [LMW*22, SK22] to support larger input sizes. The detailed architecture is listed in Table 1. “conv3- x ” indices a convolutional layer with a filter size of 3×3 and x output channels, “maxpool2” represents a max-pooling layer with a filter size of 2×2 , “FC- x ” is a fully-connected layer with x output features. The output of the classification network are directly the continuous-space simulation parameters, no additional binning is necessary. We train the network using an MSE loss between the ground truth and predicted simulation parameter using Adam with a learning rate of $1e^{-4}$, a batch size of 32 images and 50 epochs over the dataset of 50,000 synthetic displacement maps. The whole optimization takes around 7 hours.

Table 1: Modified VGG-11 architecture [SZ14] for classification of a displacement map. The 11 simulation parameters are directly predicted as continuous-space outputs.

Layer (s)	Dimensions $H \times W \times C$
Input	$512 \times 512 \times 1$
conv3-64, relu, maxpool2	$256 \times 256 \times 64$
conv3-128, relu, maxpool2	$128 \times 128 \times 128$
conv3-256, relu	$128 \times 128 \times 256$
conv3-256, relu, maxpool2	$64 \times 64 \times 256$
conv3-512, relu	$64 \times 64 \times 512$
conv3-512, relu, maxpool2	$32 \times 32 \times 512$
conv3-512, relu	$32 \times 32 \times 512$
conv3-1024, relu, maxpool2	$16 \times 16 \times 1024$
global average pooling	$1 \times 1 \times 1024$
FC-2048, relu, dropout	2048
FC-2048, relu, dropout	2048
FC-11	11

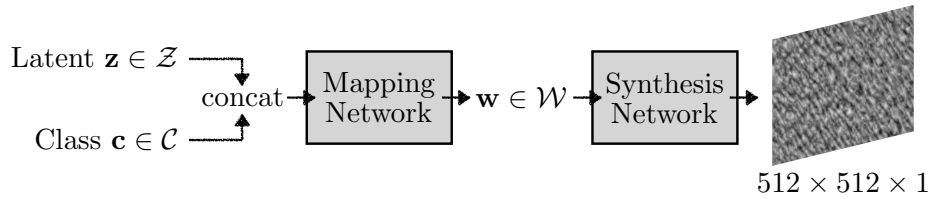


Figure 1: Generator architecture based on StyleGAN2 [KLA*20] to predict displacement maps from simulation parameters. The simulation parameters are directly passed to the network as an 11-dimensional, continuous-space class label $\mathbf{c} \in \mathcal{C} = \mathbb{R}^{11}$. The last layer of the generator is changed to predict a 1-channel grayscale displacement map instead of a 3-channel rgb image.

StyleGAN generator. For the generation architecture, we use the StyleGAN2 architecture [KLA*20] with two changes. First, the output layer is changed to predict a single-channel image for the displacements instead of a 3-channel rgb image. Second, the 11-dimensional simulation parameters are treated as class labels and concatenated with the regular latent vector, see Fig. 1. Experiments with other architectures, e.g. StyleGAN3 [KAL*21] led to inferior results. We use the same dataset of 50,000 displacement maps and train the network for roughly 5 days.

Once the generator is trained, we keep the weights frozen when optimizing the class labels for a given scanned displacement map through the generator. We use Adam with a learning rate of $1e^{-2}$, 400 epochs, a style loss as the main loss plus an MSE loss with a weight of $1e^{-4}$ and an LPIPS loss with weight $1e^{-3}$ for regularization. For the style loss, we use the same loss as Weiss *et al.* [WMC*23], that is, comparing the mean and variance of the features of a pre-trained VGG network. This optimization takes around 80 seconds per input displacement map.

B.2 Additional Evaluation of the Simulation Fitting

In Section 5.1 of the main paper, a combination of an image classification and an image generation network was presented to fit the scanned displacement maps to the simulation data. Since the fitting method is not tied to the presented patch scanning setup, we conducted further evaluation on the displacement maps provided by Graham *et al.* [GTB*13] in Fig. 2. In previous work by Weiss *et al.* [WMC*23], particle swarm optimization (PSO) was used to fit the simulation parameters (see Fig. 2). This optimization scheme, however, can get stuck in suboptimal local minima in some cases, see, for example, the cheek or nose patches, where the directionality of the input wrinkles is underestimated. In comparison, the proposed neural network fitting pipeline of performing a StyleGAN embedding from an initialization obtained from a classification network, produces much closer fits (see Fig. 2, second row). Samples from the database of synthetic displacement maps used to train the two networks are depicted in Fig. 3.

Additional qualitative and quantitative comparisons of our method (Classifier-initialization + StyleGAN refinement) against PSO, Classifier-only, and StyleGAN-only fitting on six patches from the presented patch scanning setup can be found in Fig. 4. Qualitatively, PSO and Classifier-only fail to extract the directionality of the scanned input patches. StyleGAN-only often struggles with extracting the correct pore density. Quantitatively, our method reports the best LPIPS score in four out of the six examples, compared against PSO, Classifier-only and StyleGAN-only. In the remaining two examples, Classifier-only leads to the lowest LPIPS score as the improved (visual) match of the directional component of the wrinkles harms the LPIPS score.

C Simulation Preset Parameters

In the main paper we presented the *Simulation Preset Library* consisting of artistically-designed presets and presets fitted from scanned patches. These presets can then be placed at arbitrary points on the face and the parameters are smoothly interpolated in-between. Fig. 5 and Fig. 6 show samples from this library with their parameters and a

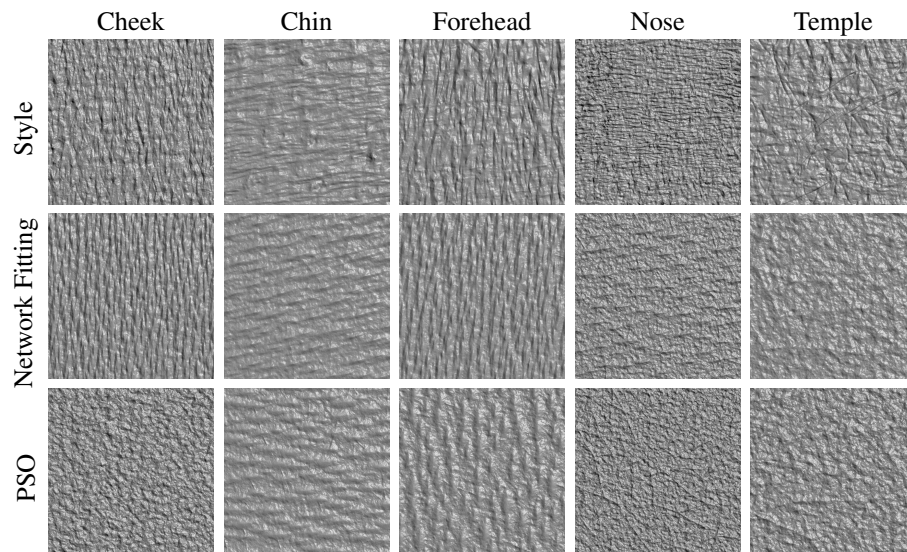


Figure 2: Results of the parameter fitting on displacement maps provided by Graham *et al.* [GTB*13] (Subject 1). The first row shows the original displacement map that is used as as input to the fitting and also as style image. The second row the re-simulated fitting results using the proposed neural network pipeline, the third row the fitting results using the particle swarm optimization (PSO) presented by Weiss *et al.* [WMC*23]. The patches are shown shaded with a light source coming slightly from the left for visualization purpose.

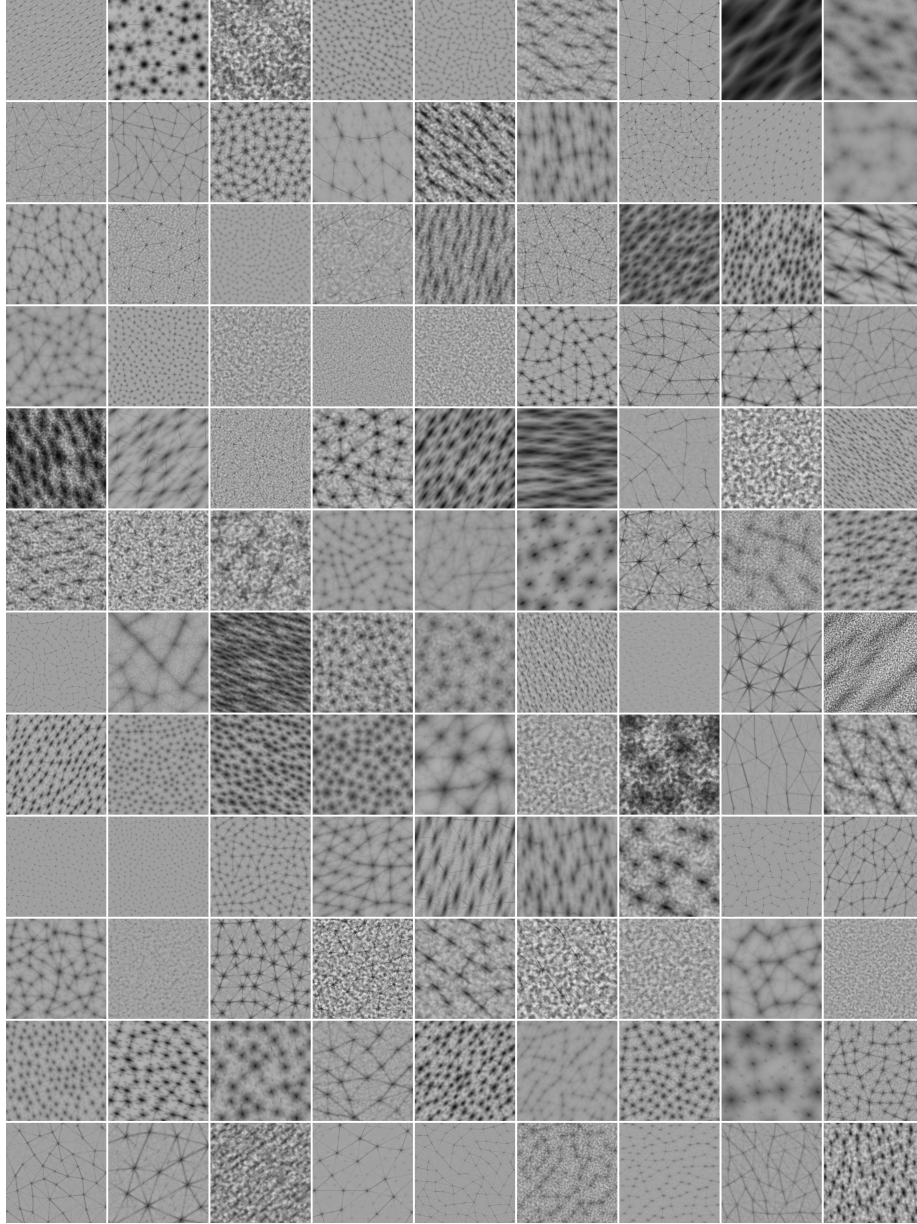


Figure 3: Random samples from the database with 50,000 synthetic displacement maps used to train the classification and generation network.

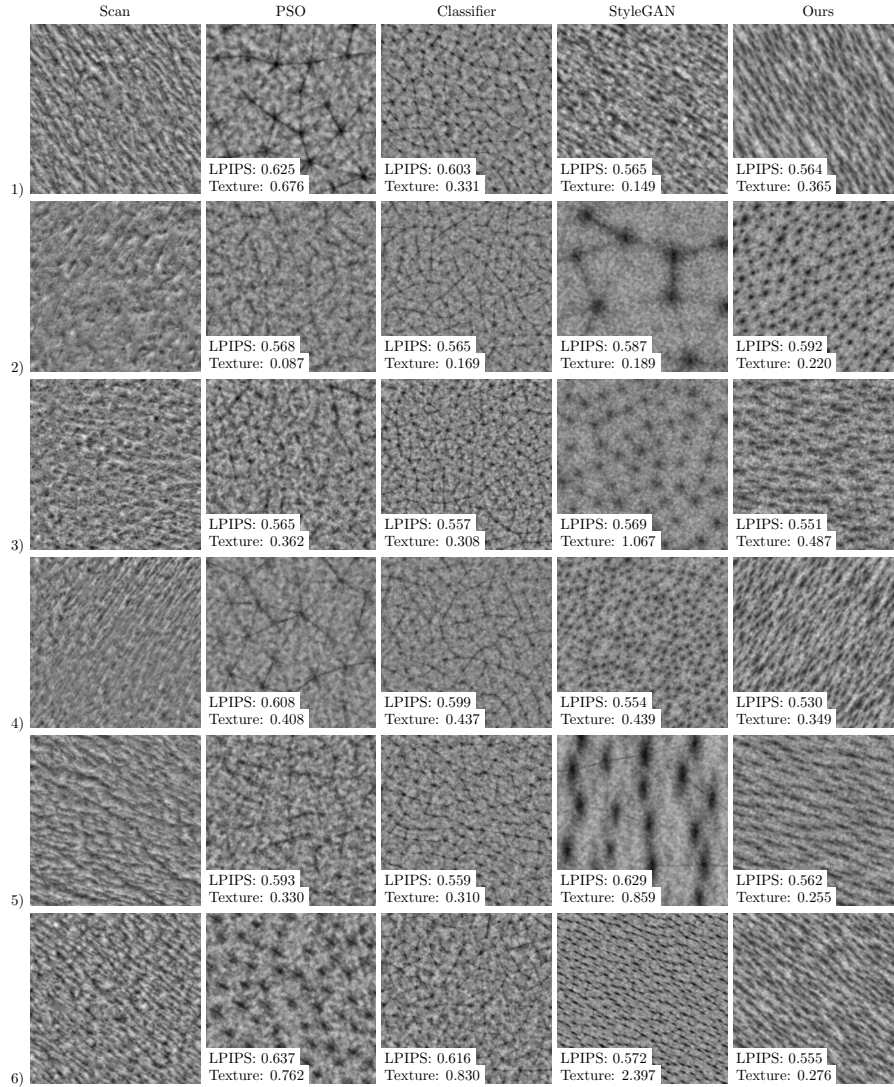


Figure 4: Additional qualitative and quantitative comparison of our method (Classifier-initialization + StyleGAN refinement) against PSO [WMC*23], Classifier-only, and StyleGAN-only fitting. For each sample, we report the LPIPS [ZIE*18] and Texture-loss score compared to the scanned displacement map. For the texture loss we compare the Gram-matrix features of a pre-trained VGG network [GEB16].

visualization of the generated skin patches. For details on the individual parameters, we refer to Weiss *et al.* [WMC*23].

D User Study: Questions

To evaluate the proposed method, we conducted a user study with 58 participants. Each participant was asked to choose one image of an A/B pair on which one looks more like natural skin. Each participant saw the same images but in a randomized order. The 56 image pairs shown to the participants together with the results for each pair can be found in Fig. 7 to Fig. 9

References

- [CGS06] CHEN T., GOESELE M., SEIDEL H.-P.: Mesostructure from specularity. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (June 2006), vol. 2, pp. 1825–1832.
- [GEB16] GATYS L. A., ECKER A. S., BETHGE M.: Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).
- [GTB*13] GRAHAM P., TUNWATTANAPONG B., BUSCH J., YU X., JONES A., DEBEVEC P., GHOSH A.: Measurement-based synthesis of facial microgeometry. In *Computer Graphics Forum* (2013), vol. 32, pp. 335–344.
- [KAL*21] KARRAS T., AITTALA M., LAINE S., HÄRKÖNEN E., HELLSTEN J., LEHTINEN J., AILA T.: Alias-free generative adversarial networks. 852–863. [arXiv:2106.12423](https://arxiv.org/abs/2106.12423).
- [KLA*20] KARRAS T., LAINE S., AITTALA M., HELLSTEN J., LEHTINEN J., AILA T.: Analyzing and improving the image quality of stylegan. 8110–8119.
- [LMW*22] LIU Z., MAO H., WU C.-Y., FEICHTENHOFER C., DARRELL T., XIE S.: A ConvNet for the 2020s. 11976–11986. [arXiv:2201.03545](https://arxiv.org/abs/2201.03545).
- [SK22] SADEGHZADEH H., KOOHI S.: Translation-invariant optical neural network for image classification. *Scientific Reports* 12, 1 (Oct. 2022), 17232.
- [SZ14] SIMONYAN K., ZISSERMAN A.: Very deep convolutional networks for Large-Scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556).
- [WMC*23] WEISS S., MOULIN J., CHANDRAN P., ZOISS G., GOTARDO P., BRADLEY D.: Graph-Based synthesis for skin micro wrinkles. In *COMPUTER GRAPHICS forum* (2023), vol. 42.
- [ZIE*18] ZHANG R., ISOLA P., EFROS A. A., SHECHTMAN E., WANG O.: The unreasonable effectiveness of deep features as a perceptual metric. [arXiv:1801.03924](https://arxiv.org/abs/1801.03924).

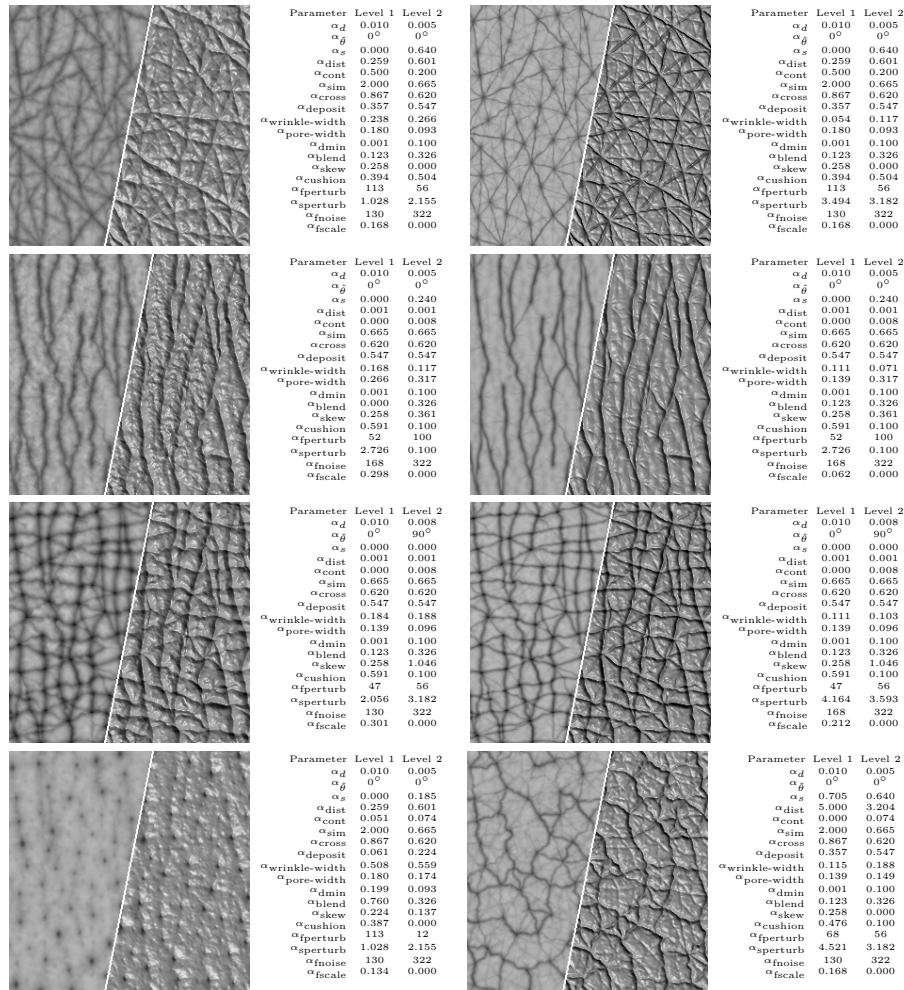


Figure 5: Samples from the Simulation Preset Library, showing the displacement map and a shaded render for each sample together with the parameters defining the *user* preset.

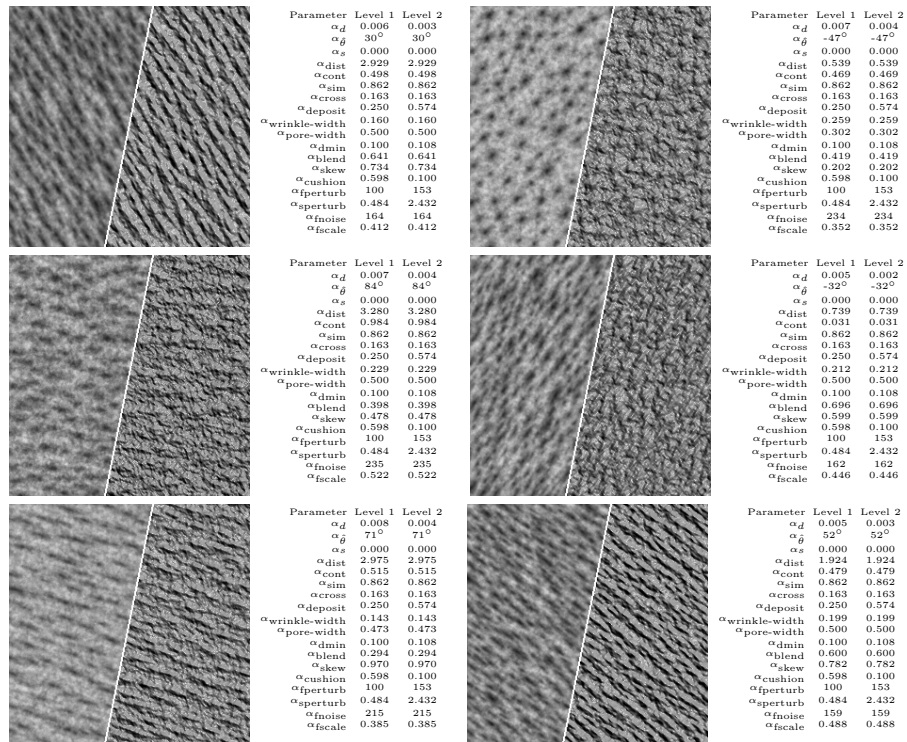


Figure 6: Cont.: Samples from the Simulation Preset Library, showing the displacement map and a shaded render for each sample together with the parameters defining the *fitted* preset.

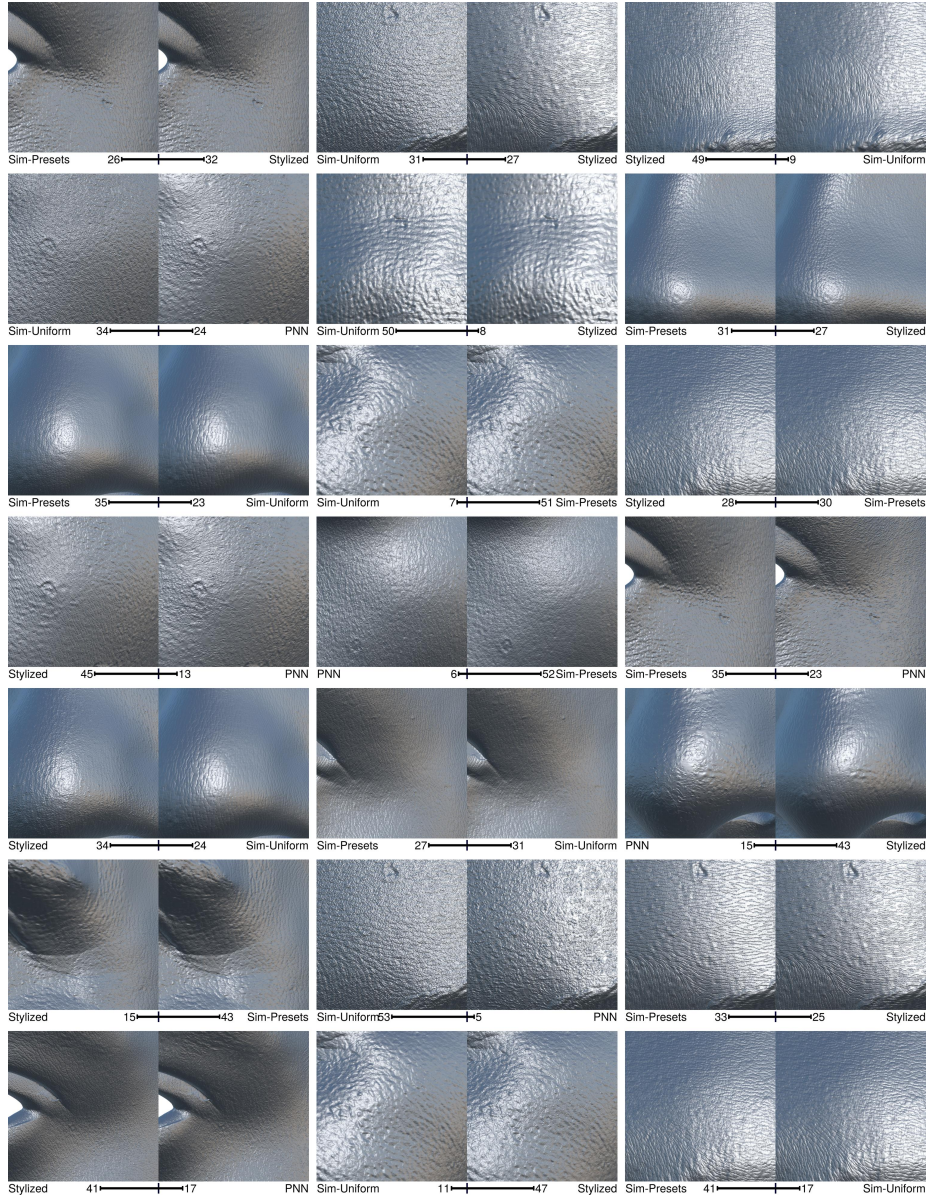


Figure 7: Image pairs used in the user study. Each image is captioned with the method that are used on the left and right side and the values indicate how many participants chose that particular option. Captions were not shown to the participants, only the images.

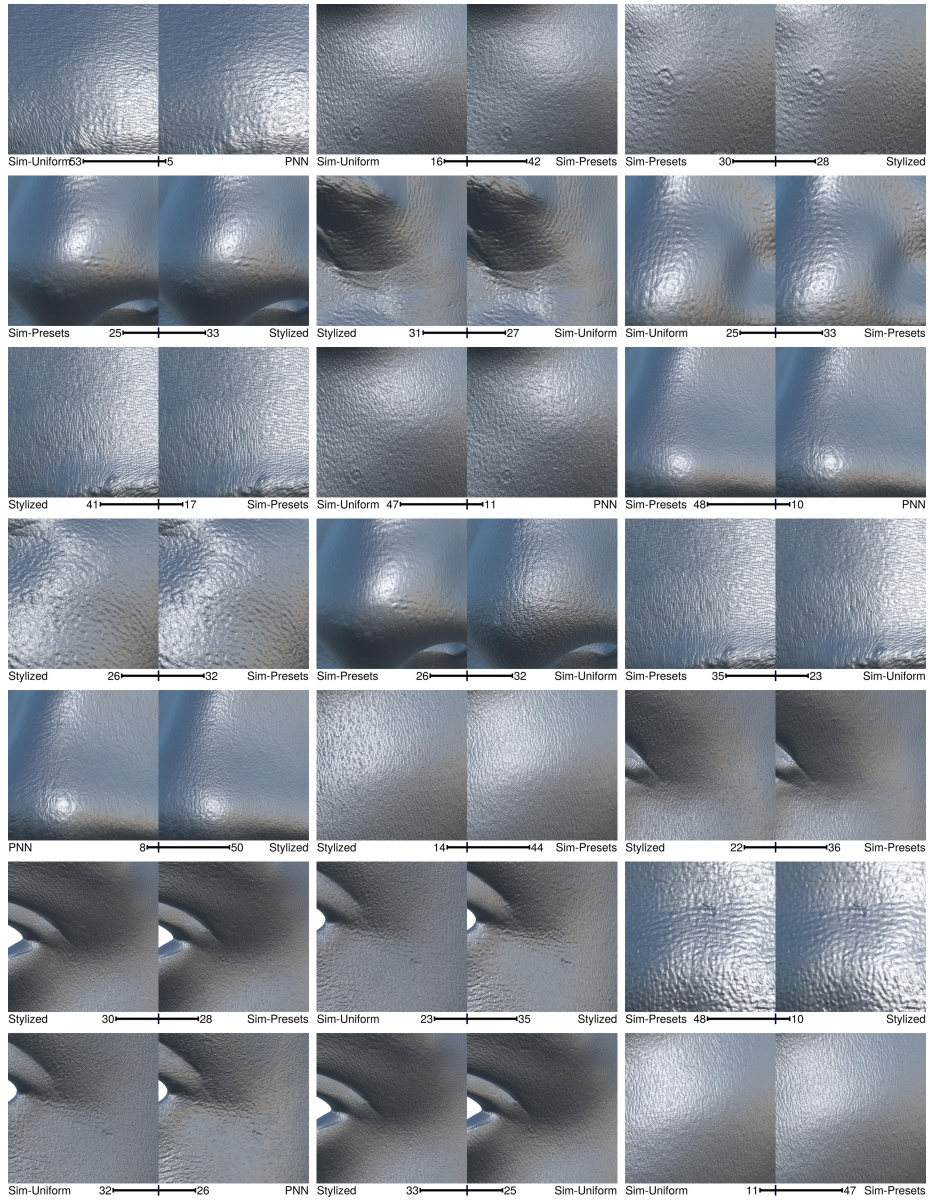


Figure 8: (Cont'd) Image pairs used in the user study.

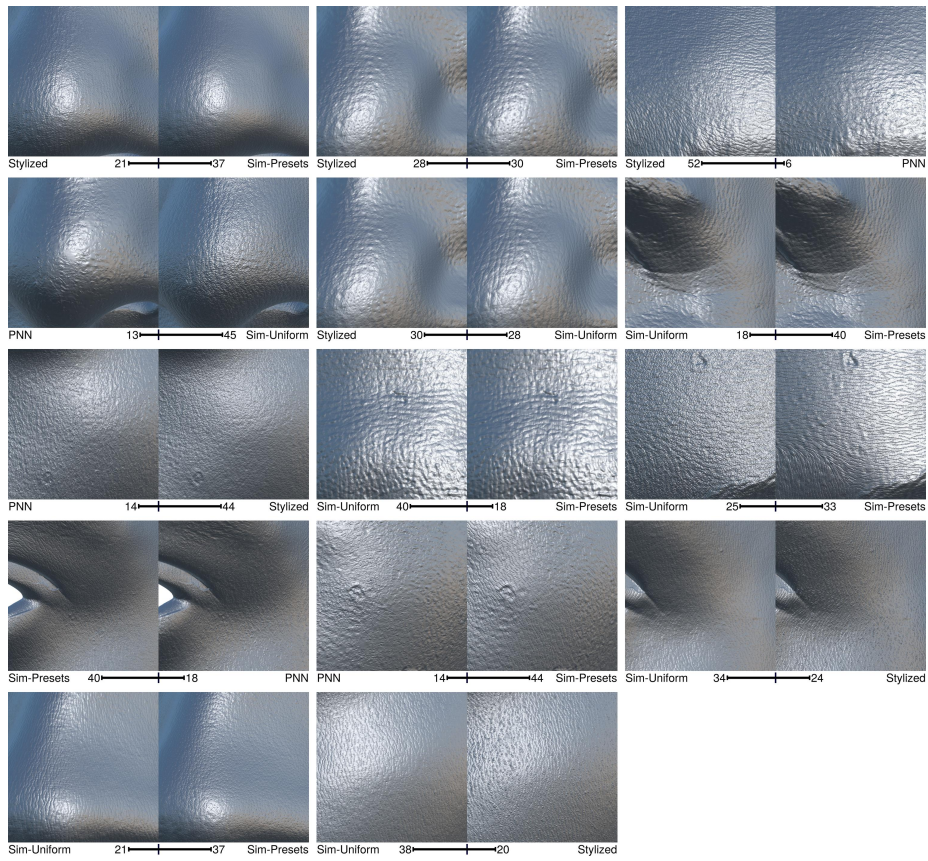


Figure 9: (Cont'd) Image pairs used in the user study.