Supplementary Materials: Multi-View Reconstruction using Signed Ray Distance Functions (SRDF)

Pierre Zins[1,2] Yuanlu Xu[2] Edmond Boyer[1,3] Stefanie Wuhrer[1] Tony Tung[2]
[1]Inria centre at the University Grenoble Alpes
[2]Meta Reality Labs, Sausalito, USA
[3]Meta Reality Labs, Zurich, Switzerland
name.surname@inria.fr, merayxu@gmail.com, tony.tung@fb.com

In the following, we provide more details about our photo-consistency network, and specify hyper parameters we used in the experiments. We further provide an ablation study to better understand the benefits of our optimization framework. We then give additional qualitative comparisons with the DTU dataset and the detailed version of the evaluation table. Finally, we show more qualitative comparisons with Renderpeople and real human capture data.

1. Photo-consistency Network

As explained in the paper, we propose a data-driven photo-consistency measure to better handle real images that are noisy and for which the Lambertian assumption is not fully satisfied. This network is composed of 3 main parts. First, features are extracted from the input images by an image encoder composed of convolutional layers, batch normalizations, ReLu activations and max-pooling operations as shown in Figure 1. Given an input 3D point, its per view multi-scale features are obtained by projecting it in the multi-scale feature maps extracted with the image encoder and by concatenating over scales. Next, we use a self-attention module [7] to combine the multi-scale features from all views and obtain therefore a multi-scale/multi-view (MSV) feature. This Pytorch [5] module is parameterized as follows, *d_model* = 115, *nhead* = 1, *dim_feedforward* = 256, *num_layers* = 6. Note that we also apply a mean operation on the output of this self-attention module. Finally, a fully connected network decodes the MSV feature and outputs a photo-consistency score between 0 and 1, as shown in Figure 2. To train the network, we use an MSE loss between the ground truth and predicted photo-consistency scores and the Adam optimizer with a learning rate of $1e^{-4}$.

2. Hyper parameters

Table 1 specifies hyper parameters used in our experiments. They are either fixed or simply scaled to match the unit of the models. The offset o defines the interval for the

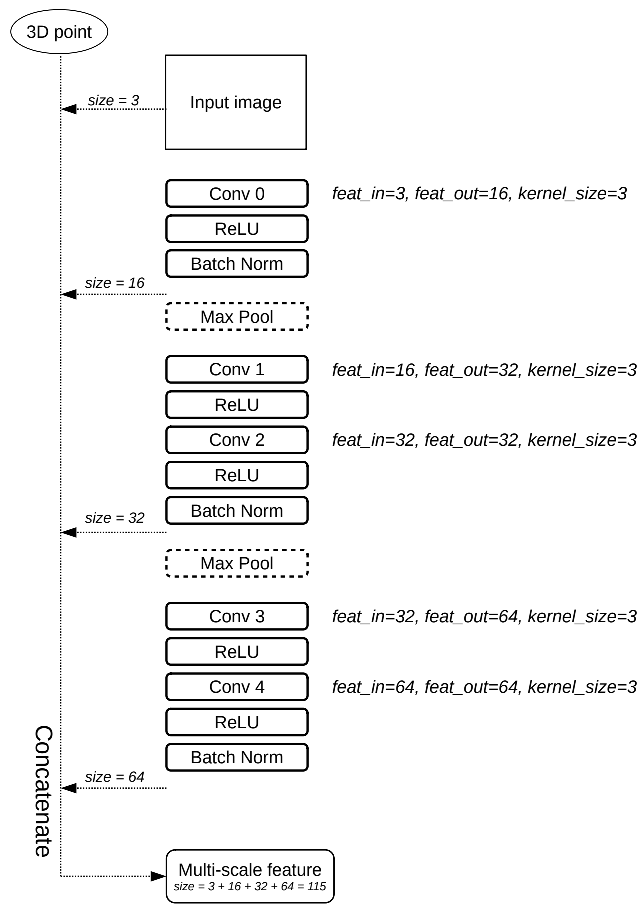

Figure 1. Architecture of the image encoder.

sampling around the current depth $[d_j^i - o; d_j^i + o]$. The real depth $\hat{d_j^i}$ needs to be contained inside this interval for the appearance to guide the geometry optimization. We set o depending on the initialization that is used such that this constraint is satisfied, and adjust o to the unit of the dataset. For datasets captured with cameras distributed all around

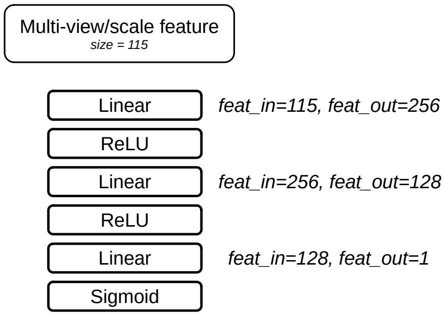

Figure 2. Fully Connected decoder.

the object, we set $o = 50$ for models in mm, while for datasets captured with cameras that observe the object from one side, we set $o = 100$ for models in mm. These values are scaled to the unit of the models.

The sampling density is fixed and set to 51 in our experiments. Γ_{SRDF} and Γ_{Φ} are used for numerical stability and prevent the product over multiple cameras to be very close to zero. We set them to 1 in our experiments. σ_c and σ_d define the strength of the penalty if the prediction of the color or the depth, respectively, from one camera, is inconsistent. σ_c is fixed empirically for each photo-consistency prior ($\sigma_c = 0.05$ for the baseline prior and $\sigma_c = 0.1$ for the learned prior). σ_d is set to 25 for models captured in mm, and scaled to the unit of the models (*e.g.*, $\sigma_d = 0.025$ for models in m).

The learning rate lr represents how much depth predictions change at each optimization iteration, and is fixed to 1 for models in mm and scaled with the unit of models.

	DTU	Renderpeople	Real human capture data
Unit	mm	cm	m
Photo-consistency prior	learned	baseline	learned
o	100	10	0.05
Sampling density	51	51	51
Γ_{SRDF}	1	1	1
Γ_{Φ}	1	1	1
σ_c	0.1	0.05	0.1
σ_d	25	2.5	0.025
lr	1	0.1	0.001

Table 1. Hyper parameters used in our optimization for the different experiments.

3. Ablation Study

To provide more in depth insight into our approach's behavior, we provide a comparison with 2 alternative strategies within our framework. First, we mention in Section 3.2 iofn the main paper that the product over cameras in Equation 3 enforces depths to become consistent across views. To evaluate this aspect we show, in Figure 3, results with an optimization of depths individually per camera, without camera product. Second, to demonstrate the benefit of the volumetric optimization we also show results with a direct search and selection of the photo-consistency maximum along rays without optimization.

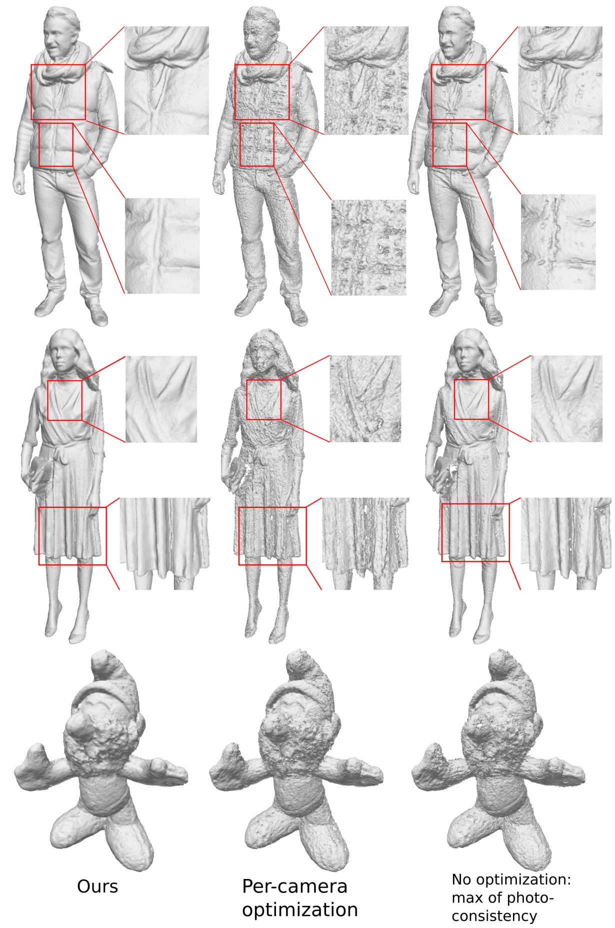

Figure 3. Ablation study with 2 alternative strategies. Reconstructions with data from Renderpeople [1] (two top rows) and DTU [4] (bottom row).

In Figure 3, it can be observed that optimizing depth per-camera, in the first alternative, is prone to local minima and that the reconstructed surfaces are quite noisy even when considering synthetic images from Renderpeople. Moreover, a global search for the maximum of the photo-consistency along each camera ray, in the second alternative, yields somewhat good results with Renderpeople data despite some noise. On the other hand, results are very noisy with real data from DTU. For both real and synthetic data, our proposed strategy that optimizes depth based on a volumetric representation clearly outperforms the two alternatives considered here.

4. Multi-View Reconstruction from Real Data

In Table 2 we show the detailed version of Table 1 of the main paper.

In Figure 4 we also give additional qualitative visual comparisons between our method and the baselines COLMAP [6], ACMMP [10], IDR [11], Neus [9], NeuralWarp [2], PatchmatchNet [8] and CasMVSNet [3] on the DTU [4] dataset. The reconstruction settings are similar to the comparison in Section 5.3 of the main paper.

5. Multi-View Reconstruction from Synthetic Data

In Figure 5, we provide additional visual comparisons between our method with the baseline photo-consistency prior defined in Section 3.3 of the main paper, and COLMAP, ACMMP, IDR, NeuS, PatchmatchNet and CasMVSNet. We use 19 synthetic images rendered from the Renderpeople [1] meshes. The reconstruction settings are similar to the comparison in Section 5.4 of the main paper. We can observe that our method is able to reconstruct very accurate and detailed meshes. Our results contain more details (*e.g*. faces, cloth wrinkles) and less noise than the other methods.

6. Multi-View Reconstruction from Real Human Capture Data

In Figure 6, we provide additional visual comparisons between our method and COLMAP, ACMMP, NeuS, PatchmatchNet and CasMVSNet. The reconstruction settings are similar to the comparison in the Section 5.5 of the main paper. We can observe that our method reconstructs detailed surfaces with limited noise even on some difficult parts as the black pants on the fourth column. COLMAP also performs quite well but has difficulties with the black bag, the pants and the hair. ACMMP is less precise; a single optimization iteration was used due to RAM limitation, even with 64GB. NeuS reconstructs a watertight surface but lacks high-frequency details and exhibits poor geometries at different locations due to appearance ambiguities. The deep MVS methods PatchmatchNet and CasMVSNet have much more difficulties reconstructing accurate surfaces. This illustrates the generalization issue with the full end-to-end learning based methods when the inference scenario is substantially different from the training one (*i.e*. DTU).

7. Societal impact

We do not see any immediate negative societal impact of our method, but we still need to be very cautious as accurate 3D models of humans could be used maliciously, without the consent of the person who is modeled.

References

[1] Renderpeople, 2018. https://renderpeople.com/3d-people/. 2, 3

[2] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6260–6269, 2022. 3, 4

[3] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020. 3, 4

[4] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413, 2014. 2, 3, 4

[5] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. 1

[6] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016. 3, 4

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. 1

[8] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14194–14203, 2021. 3, 4

[9] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 3, 4

[10] Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5483–5492, 2019. 3, 4

[11] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In *Advances in Neural Information Processing Systems*, volume 33, 2020. 3, 4

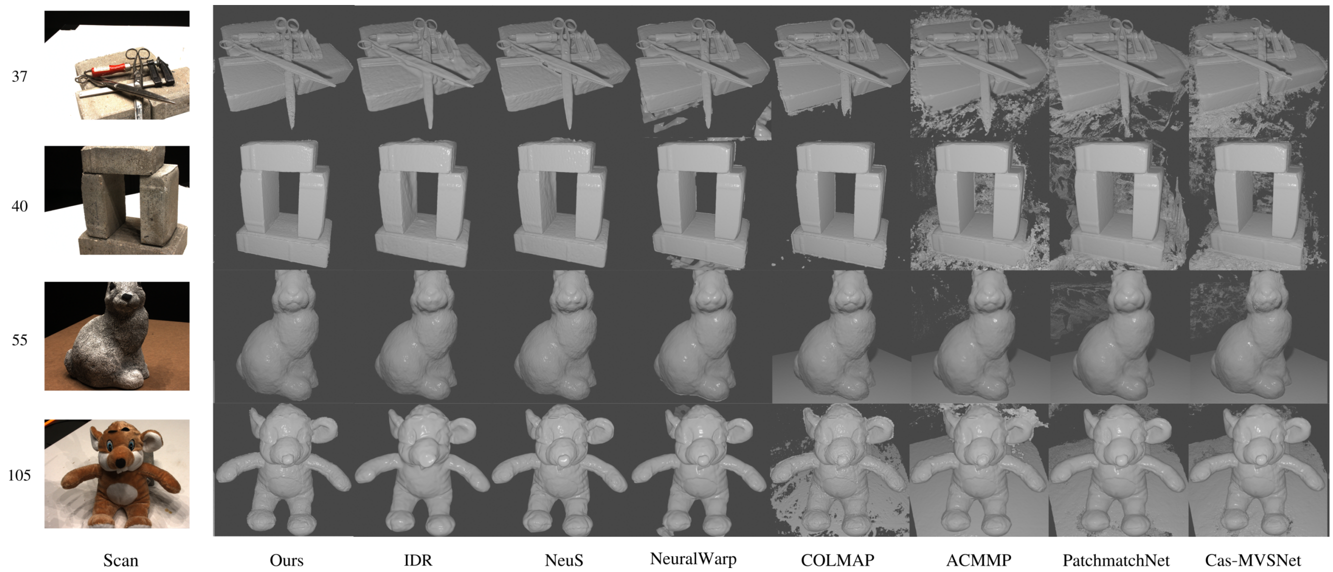

Figure 4. Qualitative comparisons on DTU.

Variants	IDR [11]			NeuS [9]			NeuralWarp [2]			COLMAP [6]			ACMMP [10]			PatchmatchNet [8]			CasMVSNet [3]			Ours		
	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg	Acc	Comp	Avg
Scan024	1.76	1.50	1.63	0.90	0.75	0.83	0.52	0.47	0.50	0.32	0.50	0.41	0.39	0.33	0.36	0.33	0.26	0.30	**0.29**	0.29	**0.29**	0.35	**0.25**	0.30
Scan037	2.16	1.55	1.86	1.09	0.88	0.98	0.80	0.61	0.70	0.57	0.66	0.62	0.66	0.44	0.55	0.56	0.45	**0.51**	**0.47**	0.58	0.52	0.61	**0.43**	0.52
Scan040	0.65	0.61	0.63	0.58	0.54	0.56	0.38	0.37	0.38	0.27	0.43	0.35	0.37	0.28	0.33	0.28	0.29	0.29	**0.24**	0.34	0.29	0.29	**0.25**	**0.27**
Scan055	0.57	0.37	0.47	0.40	0.34	0.37	0.40	0.37	0.39	0.25	0.44	0.35	0.26	0.27	0.27	0.27	**0.25**	**0.26**	0.32	0.39	0.36	**0.25**	0.26	**0.26**
Scan063	1.43	0.63	1.03	1.62	0.64	1.13	1.00	0.58	0.79	0.70	0.45	0.58	1.35	0.35	0.85	0.84	**0.26**	0.55	0.64	0.27	0.45	**0.45**	0.40	**0.43**
Scan065	0.88	0.69	0.78	0.68	**0.51**	0.59	0.80	0.82	0.81	0.32	1.60	0.96	0.32	0.72	0.52	0.34	0.98	0.66	**0.27**	1.42	0.84	0.50	**0.51**	**0.50**
Scan069	0.88	0.66	0.77	0.68	0.52	0.60	0.92	0.73	0.82	0.39	0.52	0.46	0.43	0.37	0.40	0.38	0.32	0.35	**0.31**	0.33	**0.32**	0.44	**0.27**	0.36
Scan083	1.10	1.55	1.32	1.33	1.57	1.45	0.85	1.55	1.20	0.48	0.62	0.55	0.47	0.56	0.51	0.57	**0.50**	0.54	0.36	0.51	**0.43**	**0.32**	1.02	0.67
Scan097	1.30	0.99	1.15	1.06	0.84	0.95	0.85	1.33	1.09	0.57	0.56	0.57	0.46	0.39	0.43	0.58	**0.31**	0.45	**0.42**	0.32	**0.37**	0.51	0.34	0.42
Scan105	-	-	0.64*	0.78	0.78	0.78	0.59	0.78	0.69	0.46	0.63	0.54	0.50	0.52	0.51	0.55	0.48	0.52	**0.33**	0.51	0.42	0.34	**0.27**	**0.31**
Scan106	0.73	0.60	0.66	0.53	0.52	0.52	0.57	0.78	0.67	0.29	0.57	0.43	0.32	**0.33**	0.32	0.31	0.34	0.33	**0.25**	0.40	0.33	**0.25**	0.34	**0.29**
Scan110	1.09	0.68	0.89	1.71	1.16	1.44	0.90	0.57	0.73	0.44	0.43	0.44	0.45	0.34	0.39	0.49	**0.20**	0.34	**0.34**	0.23	**0.29**	0.41	0.36	0.38
Scan114	0.45	0.38	0.41	0.34	0.38	0.36	0.42	0.41	0.41	0.26	0.36	0.31	0.26	0.27	0.27	0.39	**0.18**	0.29	**0.23**	0.19	**0.21**	0.26	0.20	0.23
Scan118	0.54	0.46	0.50	0.48	0.43	0.45	0.71	0.55	0.63	0.30	0.50	0.40	0.30	0.34	0.32	0.37	**0.25**	0.31	**0.28**	0.39	0.33	0.30	0.26	**0.28**
Scan122	0.72	0.43	0.57	0.57	0.41	0.49	0.55	0.46	0.50	0.30	0.45	0.37	0.30	0.31	0.31	0.34	**0.22**	0.28	0.26	0.34	0.30	**0.26**	**0.22**	**0.24**
Mean	1.02	0.79	0.89	0.85	0.68	0.77	0.68	0.69	0.69	0.40	0.58	0.49	0.46	0.39	0.42	0.44	**0.35**	0.40	**0.34**	0.43	0.38	0.37	0.36	**0.36**

Table 2. Quantitative evaluation on DTU [4] (49 or 64 images per model). Best scores are in **bold**. (* pre-trained model issue with Scan105, we report the IDR paper results).

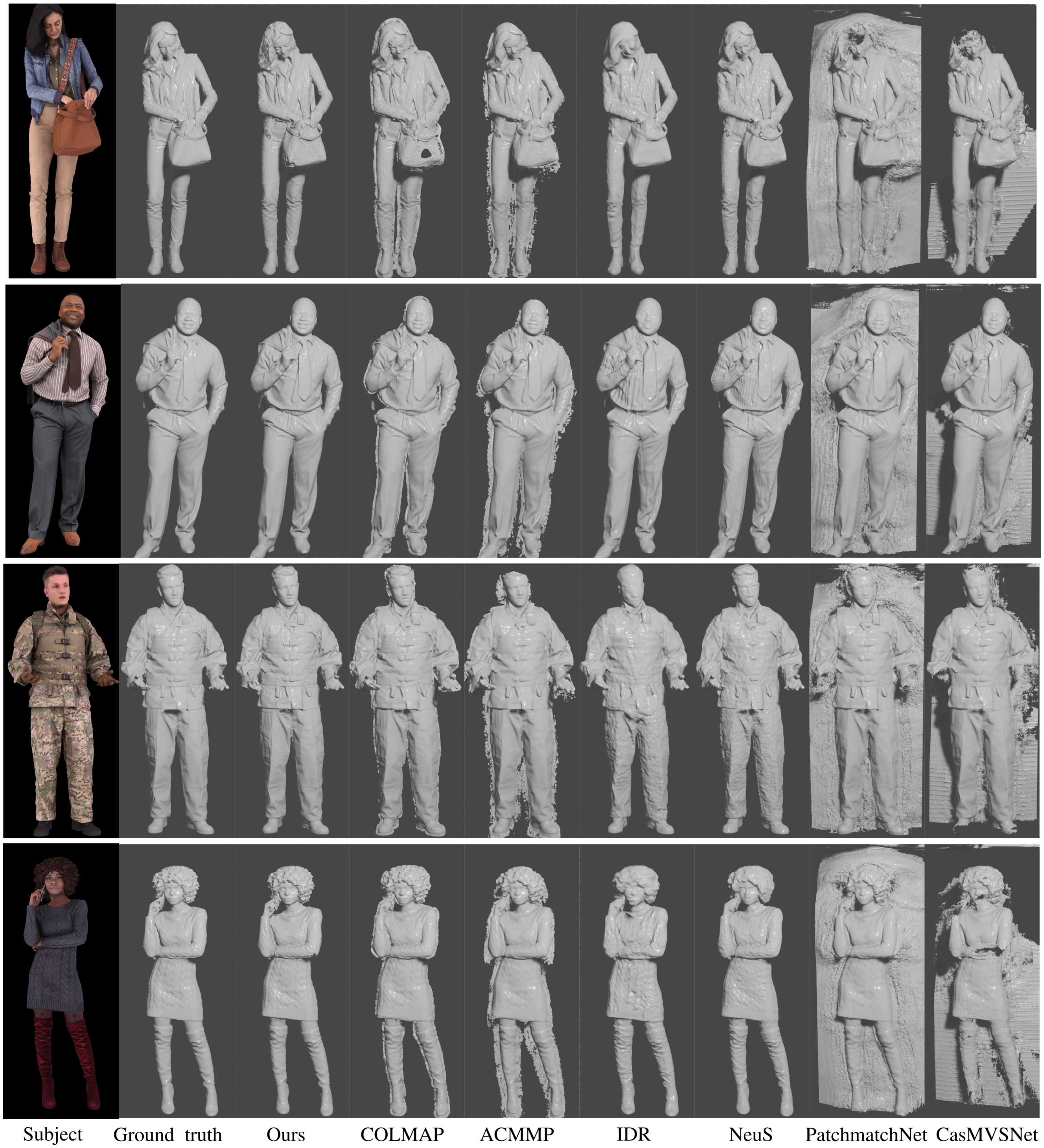

Figure 5. Qualitative comparisons on Renderpeople.

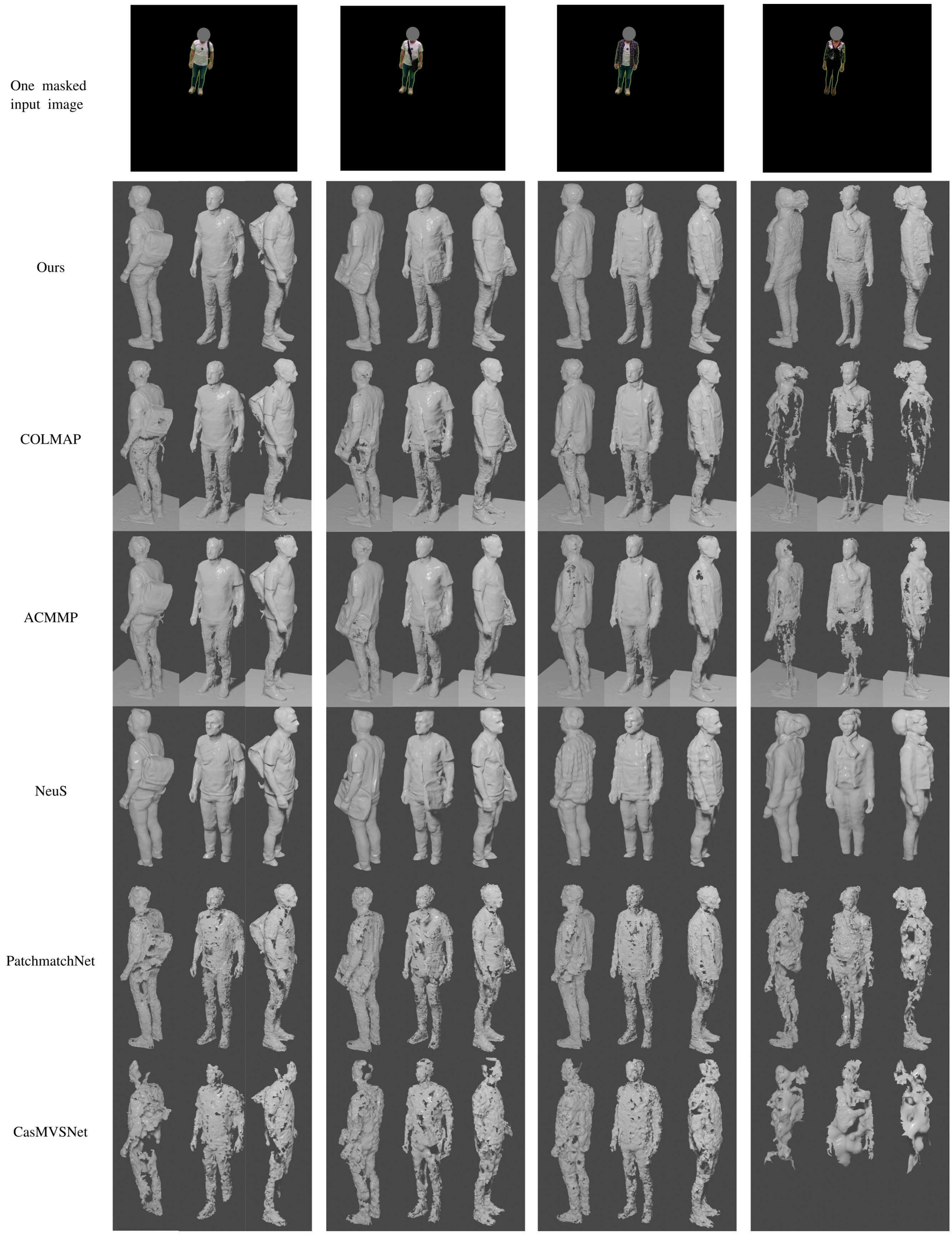

Figure 6. Qualitative comparison using 65 images from a multi-camera platform.