Discontinuity-preserving Normal Integration with Auxiliary Edges - Supplementary Document -

Hyomin Kim* Yucheol Jung* Seungyong Lee

POSTECH

{min00001, ycjung, leesy}@postech.ac.kr

This supplementary document provides additional details for the proposed method and a gallery of results that includes more visual comparisons with other methods Accompanying supplementary video provides visualization of our iterative optimization.

1. Normal Maps Captured from a Perspective Camera

For simplicity, in the main paper, our formulations for the normal integration and the iterative optimization assume that the normal map has been captured with an orthographic camera. When the normal map is captured with a perspective camera, a correct depth map can be recovered by applying minor modifications to the original formulation.

We employ the formulation for normal integration using the perspective camera in [1]. Given the normal map captured with a perspective camera of focal length f and image center coordinates (c_u, c_v), the relation between the gradient of the depth map $\mathbf{d}$ and the surface normal $\mathbf{n}$ is

$$\nabla\tilde{\mathbf{d}} = \tilde{\mathbf{g}} \tag{1}$$

where

$$\tilde{\mathbf{d}} = \log \mathbf{d}, \tag{2}$$

$$\tilde{\mathbf{g}} = \frac{\mathbf{n}_{xy}}{\tilde{n}_z}, \tag{3}$$

and the z component of the normal n_z is adjusted to form $\tilde{n}_z$ according to image coordinate (u, v):

$$\tilde{n}_z = n_x(u - c_u) + n_y(v - c_v) + n_z f. \tag{4}$$

For a more detailed explanation of the derivation of this formulation, refer to [1].

Using this result, the normal integration for the normal map captured from a perspective camera can be solved similarly to the orthogonal case by introducing minor changes. First, the z component of the input normal is modified using Eq. (4). Then, the log depth map $\tilde{\mathbf{d}}$ is acquired by performing the normal integration for the orthographic case using the modified normal as the input. Finally, the resulting depth map is obtained with exponential.

Iterative optimization Based on the relation between the depth and the gradient in Eq. (1), our proposed iterative optimization method for the discontinuity and the depth map is modified accordingly. In the perspective case, the data term $E_{data} = E_v + E_a$ is defined as

$$E_v(\hat{\mathbf{g}}, \tilde{\mathbf{d}}) = \sum_{e \in \mathcal{E}_v} \|\tilde{n}_z(e)(D_e\tilde{\mathbf{d}} - \tilde{\mathbf{g}}(e))\|^2, \tag{5}$$

$$E_a(\mathbf{g}', \tilde{\mathbf{d}}) = \sum_{e \in \mathcal{E}_a} \|\tilde{n}'_z(e)(D_e\tilde{\mathbf{d}} - \mathbf{g}'(e))\|^2, \tag{6}$$

and subsequently, the energy E_{disc} becomes

$$E_{disc}(\mathbf{g}', \tilde{\mathbf{d}}, \mathbf{w}) = \sum_{e \in \mathcal{E}_a} w_e\|\tilde{n}'_z(e)(D_e\tilde{\mathbf{d}} - \mathbf{g}'(e))\|^2, \tag{7}$$

where the value $\tilde{n}'_z$ is simply a constant $\tilde{n}'_z = f$. Note that this value is the result of Eq. (4) when the normal is $(n_x, n_y, n_z) = (0, 0, 1)$.

2. Results for Various Inputs

Inputs with noise and hole Fig. 1 presents results for normal maps with noise. Our method is robust to noise because our method uses combination of explicit gradient filtering and least-squares method; when the gradient filtering produces faulty gradient edit $\mathbf{g}'$ due to noise, the least-squares in the depth solving step can heal the damage. Fig. 2 presents results for normal maps with holes. Our method robustly handles normal maps containing small holes (top row) and a large hole (bottom row).

*Equal contribution.

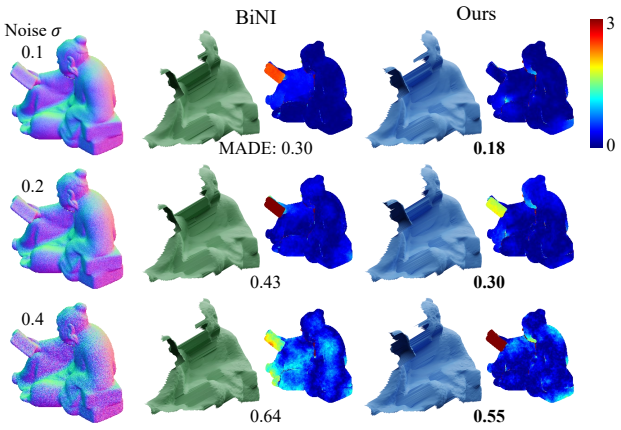

Figure 1. Normal integration when zero-mean Gaussian noise with different standard deviations σ is added to gradients. Our method produces plausible visual results in all three noise levels. Our method also shows better accuracy than BiNI [1] in terms of mean absolute depth error (MADE) to the GT depth map.

Inputs with surface details Fig. 3 presents results for two bas-relief normal maps. Bas-relief contains many subtle details that introduce discontinuities. Our method handles such subtle details correctly.

Objects with different surface frequencies In Table 1 of the main paper, for instances with mostly smooth surfaces (*Bear*, *Cat*, *Cow*), our results have similar errors to BiNI's. For instances with mid-scale details (*Buddah*, *Harvest*, *Reading*), our results have significantly lower errors than BiNI's. We show examples and evaluate the errors for surfaces with fine details in Fig. 3. We visualise results for different frequencies of teeth in Fig. 4. We obtained better or similar results to BiNI [1] in all cases. The comb-like normal maps contain sparse discontinuities only at the boundaries of the teeth.

3. Experimental Analysis of Hyperparameters

All examples in our main paper and the supplementary use the same hyperparameters. Table 1 shows quantitative evaluation results for the DiLiGenT dataset with different hyperparameters. Parameters except λ_{soft} and λ_{hard} can stay fixed as they do not change results notably.

4. Additional Technical Details

GT depth from GT mesh in the main paper's Fig. 8 GT mesh is the original 3D mesh from which the input normal map is rendered. GT depth is the visualization of the depth map with the same dimension as the rendered normal map, using each pixel as the vertex.

[1]"Relief", CC BY 4.0. https://sketchfab.com/3d-models/relief-47f53fd51c7042c59b42ffc8b01621ad

[2]"Apollo", CC BY 4.0. https://sketchfab.com/3d-models/apollo-941a94bfba164dc6a9f682d4820e0dae

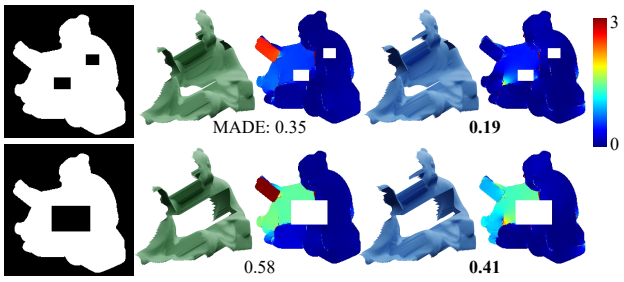

Figure 2. Normal integration results for surfaces with holes. Our method produces better results than BiNI [1] in terms of mean absolute depth error (MADE).

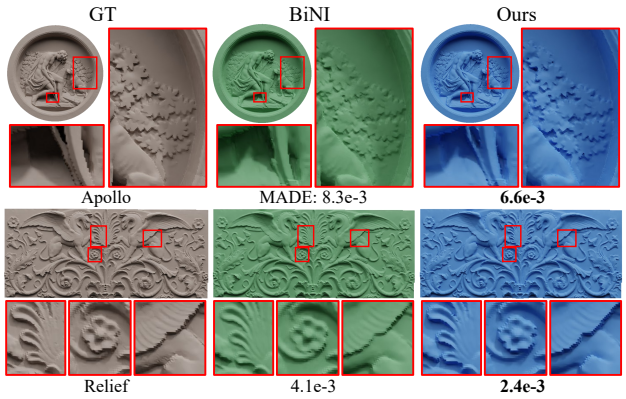

Figure 3. Normal integration results for surfaces with details[1] [2]. Our method produces better visual results capturing subtle GT details as can be observed with the appearance of shadows. Our method also produces better results than BiNI [1] in terms of mean absolute depth error (MADE).

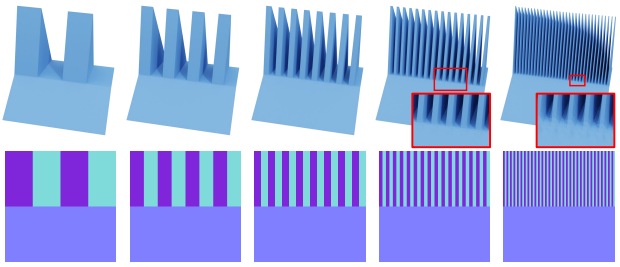

Figure 4. Our results for comb-like normal maps. The input normal map for the third column is the same as the input used in BiNI's Fig. 4 [1]. Our method produces correct results for variations of comb-like normal maps with different frequencies.

Time consumption and convergence For the DiLiGeNT dataset, BiNI took 10 seconds on average to converge, and additional iterations do not notably enhance the accuracy. Our method takes 6 seconds on average to reach the same

Table 1. Quantitative evaluation results for the DiLiGenT dataset with different hyperparameters. For all examples in our main paper and the supplementary, we use the hyperparameters $k = 1000$, $\lambda_{soft} = 0.2$, $\lambda_{hard} = 1.2$ (Ours, first row).

	Bear	Buddha	Cat	Cow	Harvest	Pot1	Pot2	Reading	Goblet
Ours $(k, \lambda_{soft}, \lambda_{hard})$	0.45	0.67	0.24	**0.06**	**2.44**	**0.57**	0.19	**0.15**	9.02
(100, 0.2, 1.2)	0.49	**0.41**	0.20	0.21	2.94	0.60	**0.17**	0.42	9.15
(2000, 0.2, 1.2)	0.48	0.78	0.23	0.10	2.98	0.62	0.20	0.16	9.00
(100, 0.1, 1.1)	0.36	1.64	**0.14**	0.13	4.38	0.58	0.20	0.40	9.04
(1000, 0.1, 1.1)	0.30	2.40	0.16	0.13	4.93	0.70	0.24	0.52	7.95
(2000, 0.1, 1.1)	**0.14**	2.44	0.15	0.15	5.21	0.81	0.25	0.65	**7.90**

level of depth error as BiNI, and produces more accurate results with additional iterations. Using a fixed number of iterations $N_{max} = 1300$, which takes approximately 50 seconds, our method achieves higher accuracy than BiNI in most cases.

5. Gallery of Results

In Figs. 5, 6, and 7, we provide additional visual comparisons to the Poisson method [2] and BiNI [1] using normal maps in THuman2.0 [3] dataset. For BiNI, we use the hyperparameter $K = 2$ used in the paper. Due to our explicit representation of the jumps across discontinuities, our method recovers sparse discontinuities accurately even in small and subtle jumps, e.g., across wrinkles of clothes and hair.

References

[1] Xu Cao, Hiroaki Santo, Boxin Shi, Fumio Okura, and Yasuyuki Matsushita. Bilateral normal integration. In *Proc. ECCV*. Springer, 2022. 1, 2, 3, 4, 5, 6

[2] Berthold KP Horn and Michael J Brooks. The variational approach to shape from shading. *Computer Vision, Graphics, and Image Processing*, 33(2):174–208, 1986. 3, 4, 5, 6

[3] Tao Yu, Zerong Zheng, Kaiwen Guo, Pengpeng Liu, Qionghai Dai, and Yebin Liu. Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR2021)*, 2021. 3

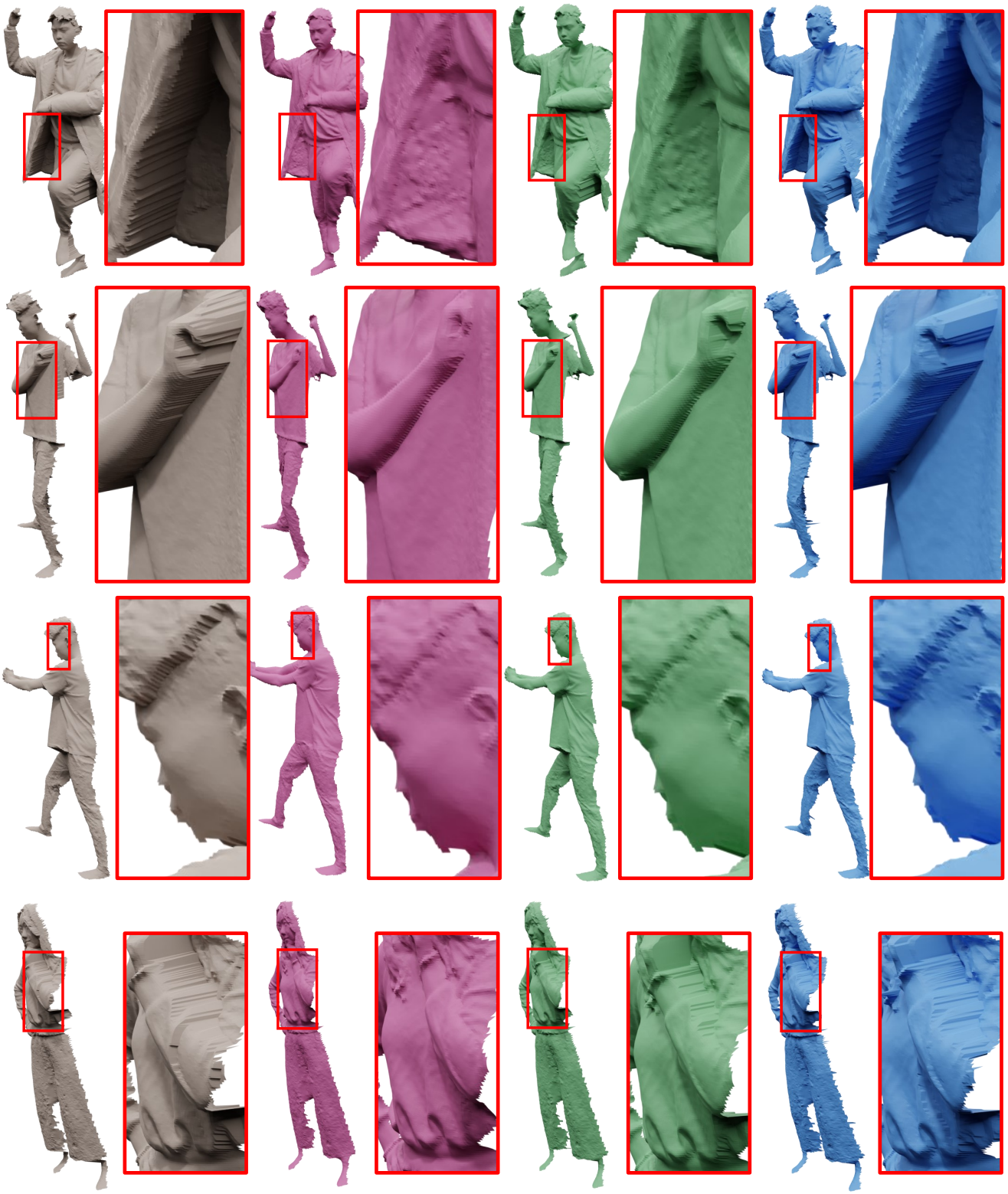

Figure 5. Qualitative comparison to previous normal integration methods using a human body dataset.

Figure 6. Qualitative comparison to previous normal integration methods using a human body dataset.

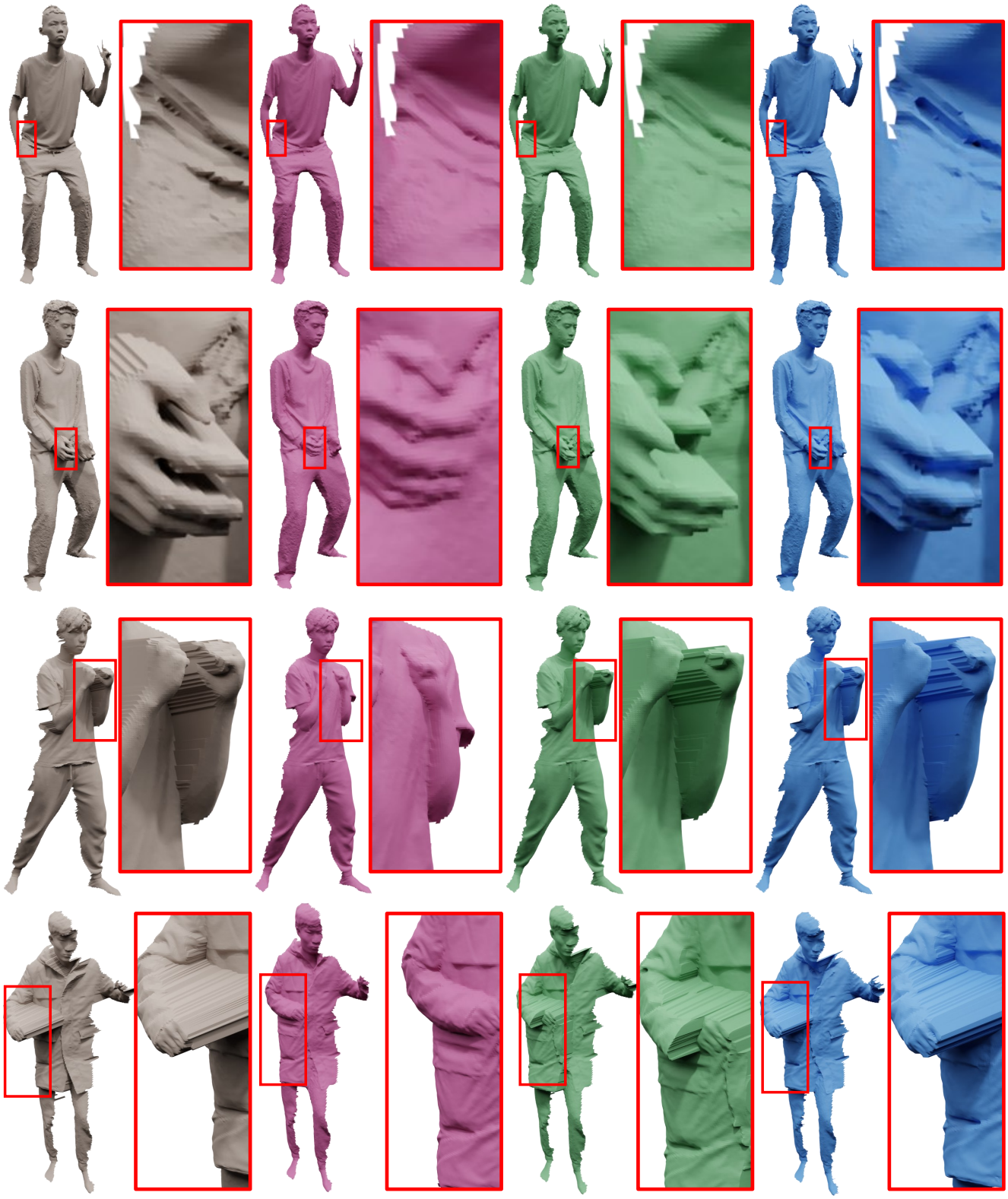

Figure 7. Qualitative comparison to previous normal integration methods using a human body dataset.