Self-Supervised Deep Depth Denoising (Supplementary Material)

Vladimiros Sterzentsenko * Leonidas Saroglou * Anargyros Chatzitofis *
Spyridon Thermos * Nikolaos Zioulis * Alexandros Doumanoglou
Dimitrios Zarpalas Petros Daras
Information Technologies Institute (ITI), Centre for Research and Technology Hellas (CERTH), Greece

1. Introduction

In this supplementary material we complement our original manuscript with additional quantitative and qualitative results, which better showcase the advantages of the proposed self-supervised denoising model over traditional filtering and supervised CNN-based approaches. In particular, we present the adopted implementation details used for training our model, as well as additional qualitative results for the two 3D application experiments presented in the original manuscript, namely 3D scanning with KinectFusion [8] and full-body 3D reconstructions using Poisson 3D surface reconstruction [4]. A comparative evaluation with the learning-based state-of-the-art methods on InteriorNet (IN) [7] follows, while an ablation study concludes the document.

The aforementioned results based on all methods presented in the originally manuscript, namely Bilateral Filter (BF [10]), Joint Bilateral Filter (JBF [6]), Rolling Guidance (RGF [12]), and data-driven approaches (DRR [3], DDRNet [11]). Note that for the DRR and DDRNet methods, additional results aim to highlight the over-smoothing effect of the former and the weakness of the latter to denoise depth maps captured by the Intel RealSense D415 sensors. In more detail, DRR is trained on static scenes that contain dominant planar surfaces and, thus tends to flatten (*i.e.* oversmooth) the input data. On the other hand, the available DDRNet model [1] that we used, produces high levels of flying pixels (*i.e.* spraying, see Fig. 1) which can be attributed to background (zero depth values) and foreground blending, even though its predictions are appropriately masked. While the authors have not provided the necessary information, it is our speculation that the available model is trained using Kinect 1 data, which is partly supported by the suboptimal results it produces on Kinect 2 data.

Qualitatively, the remaining traditional filters (BF, JBF, RGF) perform similarly, with RGF showcasing the most competitive results to our method. However, note that RGF utilizes color information in an iterative scheme. It is worth mentioning that depending on the evaluation, *i.e.* KinectFusion or Poisson reconstruction, the difference in the quality among the methods may be more or less distinguishable.

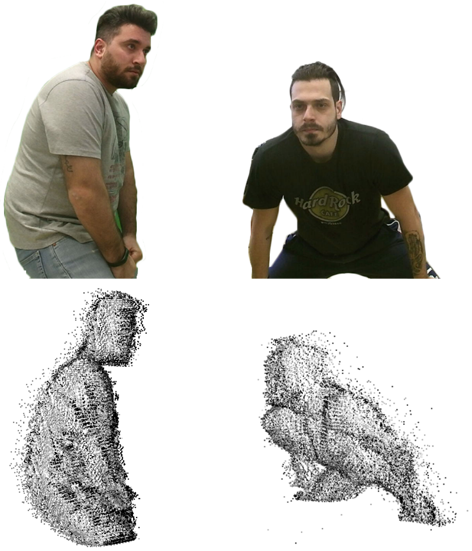

Figure 1. Denoising results using original DDRNet[11] model on Kinect 2 data.

1.1. Implementation Details

The CNN-based autoencoder presented in the original manuscript is implemented using the PyTorch framework [9]. The hyperparameters' initialization follows, while the notation from Section 3.2 (see original manuscript) is adopted. We set $\lambda_1 = 0.85, \lambda_2 = 0.1, \lambda_3 = 0.05$, while α for the photometric loss is set to 0.85. Regarding the outlier estimators, we set $\gamma = 0.447$ for the Charbonnier and $c = 2.2$ for the Tukey penalty, respectively. During training, ELU(a) non-linearity with $a = 1$ is used for all CONV layers except for the output one, while Adam [5] with $\beta_1 = 0.9, \beta_2 = 0.99$ is used for optimization. Xavier

*Equal contribution

[1] https://github.com/neycyanshi/DDRNet

initialization [2] is used for the network weights. The network is trained with learning rate set to 0.0002 and a mini-batch size of 2. Training converges after about $102k$ iterations. The network is trained with depth and color images of 640×360 resolution, while no data augmentation is performed. All collected depth maps are thresholded to $3m$ and thus $\sigma_D = 3$. Note that the mean inference time on a GeForce GTX 1080 graphics card is 11ms.

1.2. KinectFusion Reconstruction

KinectFusion [8] reconstructs 3D surfaces by temporally aggregating and fusing depth maps, also implicitly denoising the outcome through the Truncated Signed Distance Functions (TSDFs) fusion process. Therefore, the method's most typical failure case corresponds to an insufficient registration of an input frame, either attributed to difficult to track motion (pure rotation, fast translation) or noisy input. Our results are offered in the exact same sequences, and thus the former source of error is removed, with any tracking failures attributed to noisy inputs.

Consequently, even noisy depth observations may result in high quality 3D scans. Although the original depth estimates from D415 are noisy, in most cases, KinectFusion manages to reconstruct a relatively smooth 3D mesh surface. This is illustrated in Fig. 4 in the first row, where the resulting meshes using the raw depth input are presented. It is worth noting that D415 depth map denoising is proven challenging for the data-driven methods. In particular, DRR tends to over-smooth the surfaces, while DDRNet is completely incompatible with the depth data.

KinectFusion on DDRNet denoised data was repeatedly failing to make correspondences in consecutive frames due to the increased amount of "spraying" in the denoised output. Thus, the fact that our proposed method does not fall into the same limitations as the other data-driven methods can be considered an advantage. KinectFusion-based results are shown in Fig. 4. For comparison, the last row of Fig. 4 shows 3D reconstruction using frames acquired by Microsoft Kinect 2 device, which captures higher quality depth.

We further experimented with DDRNet by re-implementing its denoising part, using traditional and partial convolutions, denoted as DDRNet-TC and DDRNet-PC, respectively. The model was trained using our dataset, which resulted in better results due to the sparse nature of the data. Since our dataset does not contain ground-truth depth-maps, we employ forward-splatting (see Section 3.1 of the original original manuscript) in order to produce cleaner depth-maps to use as near ground-truth. As Table 1(top) shows, our model outperforms DDRNet retrained models, in both regular and partial convolution by a wide margin, which can be attributed to the different behaviour of splatting color images compared to depth maps. For completeness, we qualitatively evaluated the performance of DDRNet-PC using KinectFusion (see Fig. 4, middle). Note that even if denoising is improved compared to the original DDRNet, the reconstructed output quality is still low.

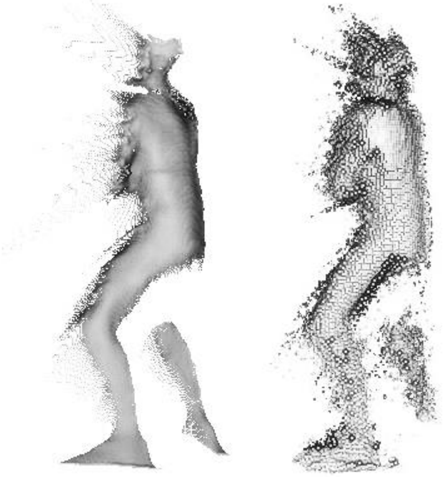

Figure 2. Projected depth maps to 3D domain, after denoising with DRR [3] (left) and original DDRNet [11] (right). This figure showcases the spray at the boundaries, which leads Poisson reconstruction method to fail (see Section 1.3).

Table 1. Top to bottom sections: a) DDRNet trained with splatted depth, b) learning-based methods evaluation on IN, c) ablation results of the proposed denoising model.

Model	**MAE**	**RMSE**	**M(°)↓**	**10(%)↑**	**20(%)↑**	**30(%)↑**
DDRNet-TC	121.83	265.10	56.17	1.41	5.79	13.41
DDRNet-PC	75.68	241.58	40.46	5.38	18.83	36.13
DDRNet (IN)	140.80	198.45	59.86	1.72	6.07	11.32
DRR (IN)	86.88	144.97	25.84	26.72	48.62	65.19
Ours (IN)	**33.44**	**81.28**	**20.08**	**39.53**	**64.12**	**77.37**
AE	26.35	59.92	36.30	7.78	25.75	45.70
P+N	28.04	60.20	34.32	8.73	28.55	49.46
P+D	26.43	**58.31**	**31.71**	**9.62**	**31.39**	**53.98**
P-only	25.96	58.30	32.13	9.39	30.69	53.11
P+D+N (best)	**25.11**	58.95	32.09	9.61	31.34	53.65

1.3. Poisson Reconstruction

The second method used to qualitatively compare the aforementioned methods is the well-established 3D Poisson reconstruction [4, **?**]. The setup is realized as 4 RealSense D415 sensors placed in a cross-like setup to capture a static subject in a full 360° manner. Poisson reconstruction utilizes surface information (oriented point-clouds) in order to recover the original 3D shape, constituting an appropriate application to compare denoising results while preserving geometric details in a qualitative manner. While the produced reconstructions are watertight, "balloon" like artifacts can be observed in empty areas where the proximal surface information is inconsistent or noisy. This can be seen in Fig. 5 (1st row-"Raw Depth"). The curvature of these proximal patches is an indicator of the smoothness across the boundary of the hole (empty area). Fig. 5-7 demonstrate the results of full-body 3D reconstructions using 4 depth maps denoised with each evaluated method, as well as the original raw input (1st row). Note that results using DDRNet (both original and retrained) and DRR methods are omitted, as their denoised depth maps are affected by "boundary spraying" (see Fig. 2), which leads to highly cluttered 3D reconstructions. Depth maps denoised using RGF are also affected by slight spraying, which is easily removed manually in order to present a fair quality result. From the presented results, it can be seen that 3D reconstruction from raw (noisy) depth maps preserves little to no geometric details, while using depth maps from BF and JBF methods leads to local region smoothing. On the other hand, RGF leads to higher quality results as which are comparable to those produced by our model. It should be noted though that our model infers using only depth input while RGF requires color information and an appropriate selection of parameters.

1.4. Evaluation on InteriorNet

IN consists of 22M layouts of synthetic indoor scenes with varying lighting configuration. We use 6K samples from the first 300 scenes. We corrupt these clean ground truth depth maps with two artificial noise patterns in order to create noisy-ground truth data pairs; a) a noise similar to the one presented in [1], and b) a ToF-like, non-linear (distance-dependent), bi-directional noise distribution along the ray. The quantitative results of the learning - based methods are shown in Table 1(middle). As for qualitative results on this task, we provide the original, ground truth and denoised images in Fig. 3.

1.5. Ablation Study

Finally, we perform an ablation study of various aspects of out deep depth denoising model. Spacifically, we examine cases of a) training the model as a plain autoencoder (AE) without bell and whistles (only reconstruction loss used), b) training the AE with photometric loss only (P-only), c) regularizing supervision using the BerHu depth loss (P+D), d) using normal priors to guide the supervision (P+N), instead of depth regularization, and e) combining photometric supervision with depth and surface normals losses (P+D+N), as presented in the original manuscript.

The results of the aforementioned cases evaluated on our dataset are presented in Table 1(bottom). These results indicate that photometric supervision is a better alternative than a plain autoencoder train with a reconstruction loss, as well as that depth regularization is important as it aids photometric supervision by constraining it when its assumptions break (no texture, etc.). Further, note that the normals smoothness prior leads to a significant improvement of the MAE, while achieving the second best performance in the rest of error metrics. Based on this analysis, we adopt the last training scheme for our depth denoising model.

References

[1] Jonathan T. Barron and Jitendra Malik. Intrinsic scene properties from a single RGB-D image. In *CVPR*, pages 17–24, 2013.

[2] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proc. International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[3] Junho Jeon and Seungyong Lee. Reconstruction-based pairwise depth dataset for depth image enhancement using CNN. In *ECCV*, pages 438–454, 2018.

[4] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics*, 32(3):29, 2013.

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[6] Johannes Kopf, Michael F. Cohen, Dani Lischinski, and Matt Uyttendaele. Joint bilateral upsampling. *ACM Transactions on Graphics*, 26(3), 2007.

[7] Wenbin Li, Sajad Saeedi, John McCormac, Ronald Clark, Dimos Tzoumanikas, Qing Ye, Yuzhong Huang, Rui Tang, and Stefan Leutenegger. InteriorNet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. In *BMVC*, 2018.

[8] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *ISMAR*, pages 127–136, 2011.

[9] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. 2017.

[10] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *ICCV*, pages 839–846, 1998.

[11] Shi Yan, Chenglei Wu, Lizhen Wang, Feng Xu, Liang An, Kaiwen Guo, and Yebin Liu. DDRNet: Depth map denoising and refinement for consumer depth cameras using cascaded CNNs. In *ECCV*, pages 155–171, 2018.

[12] Qi Zhang, Xiaoyong Shen, Li Xu, and Jiaya Jia. Rolling guidance filter. In *ECCV*, pages 815–830, 2014.

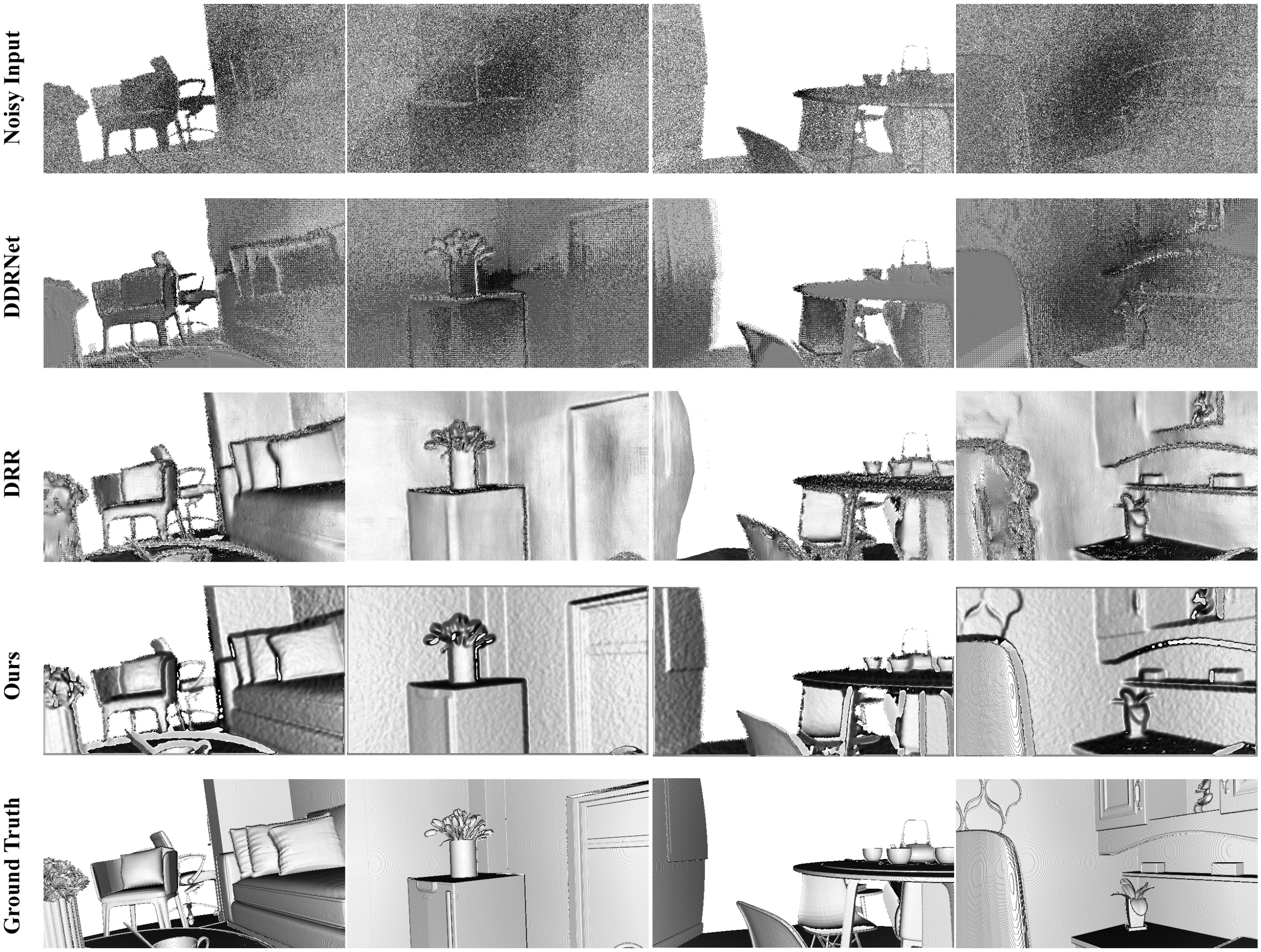

Figure 3. Quantitative results of learning-based methods in rendered images from InteriorNet layouts. The first row and last row show the noisy input and ground truth respectively. DDRNet fails to remove most of the noise. On the other hand, DRR shows promising results, although it tampers the shape of some objects and fails to preserve fine details. Our model shows superior results comparing to other methods on these data.

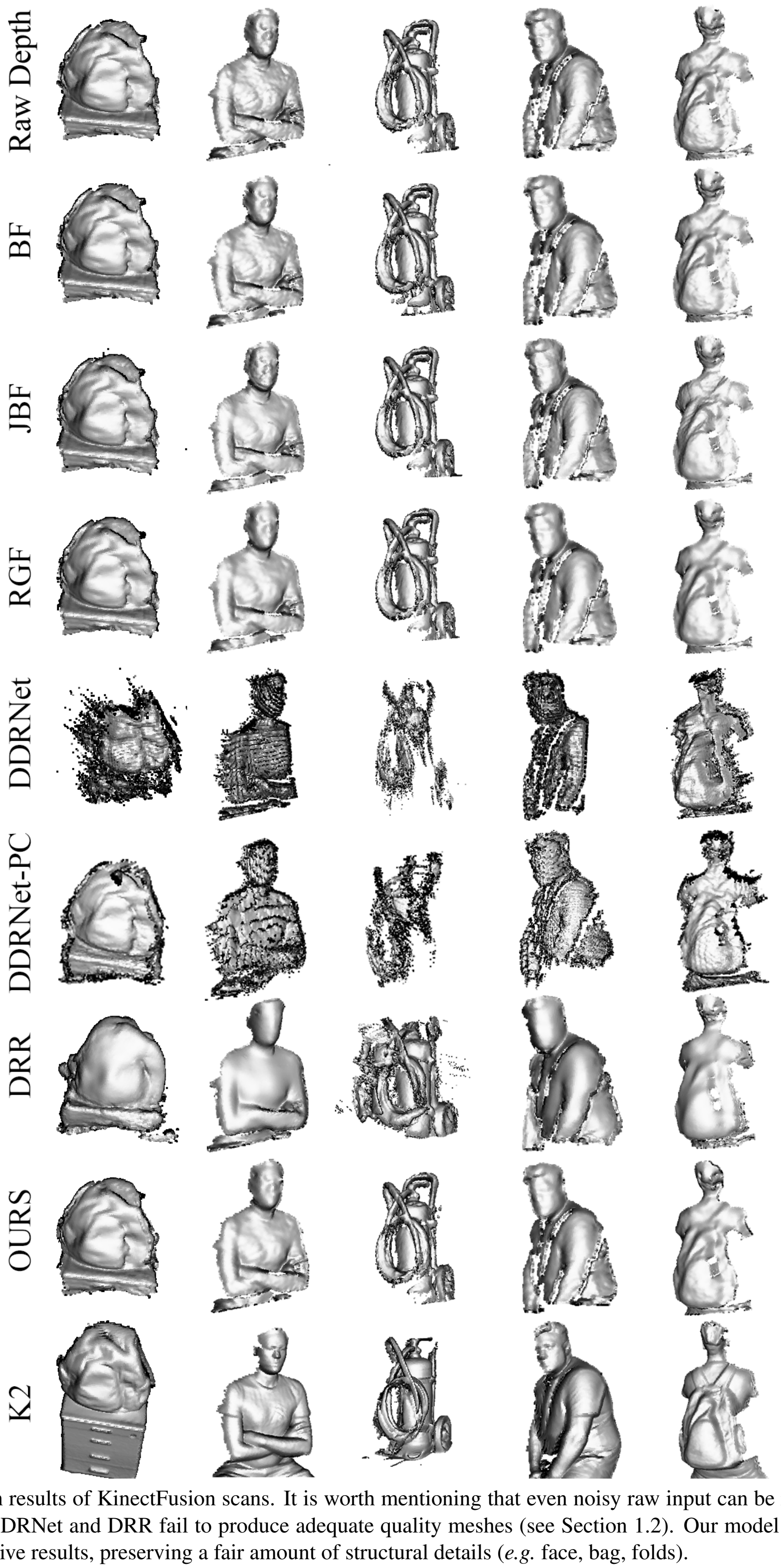

Figure 4. Reconstruction results of KinectFusion scans. It is worth mentioning that even noisy raw input can be reconstructed into a high quality mesh (row 1). DDRNet and DRR fail to produce adequate quality meshes (see Section 1.2). Our model along with K2 and RGF produce the best qualitative results, preserving a fair amount of structural details (*e.g.* face, bag, folds).

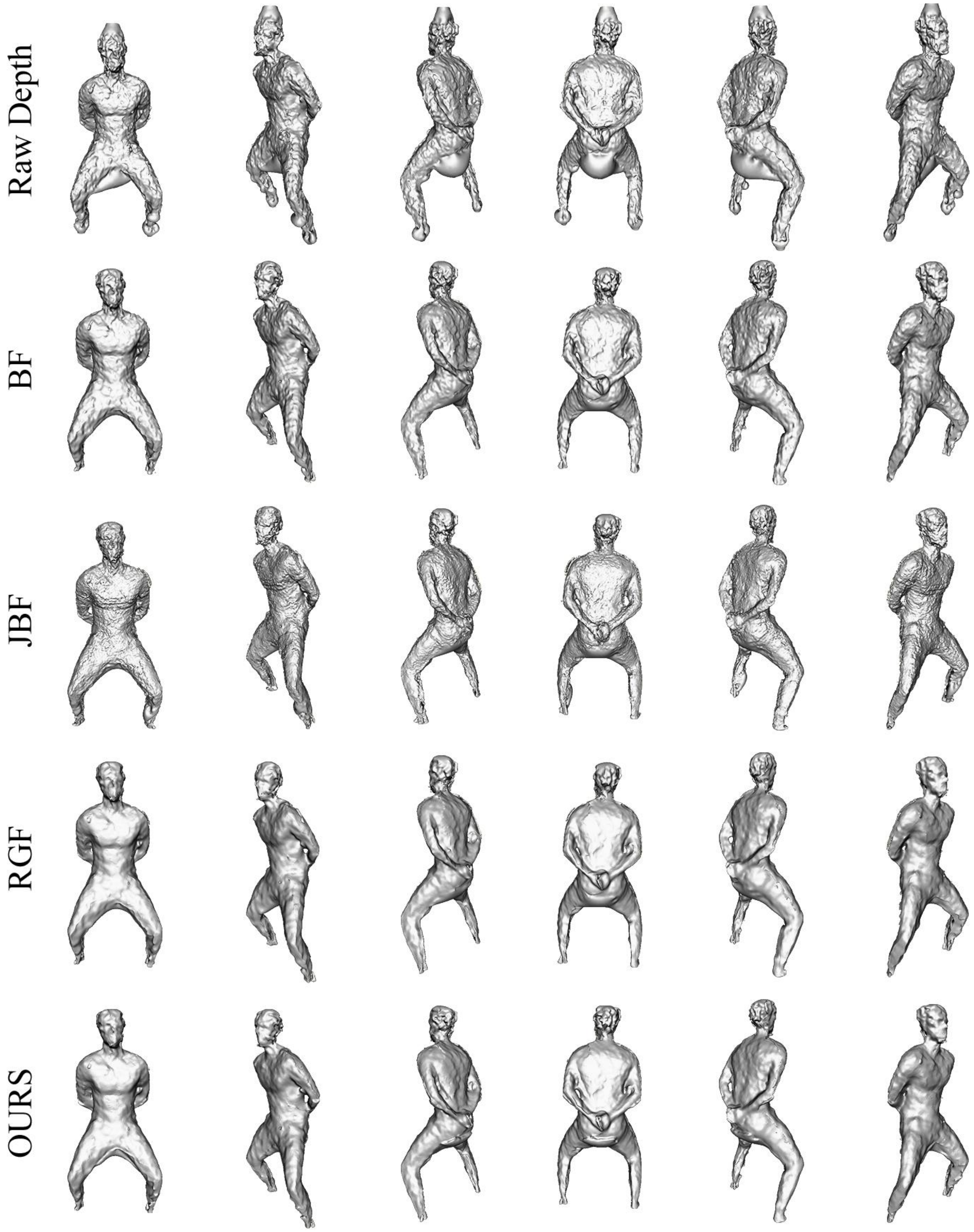

Figure 5. Poisson reconstruction sample. BF and JBF lead to low quality reconstruction due their inability to understand the global context of the scene. Our method and RGF lead to higher quality reconstructions, restoring face details that are hardly spotted in “Raw Depth” reconstruction.

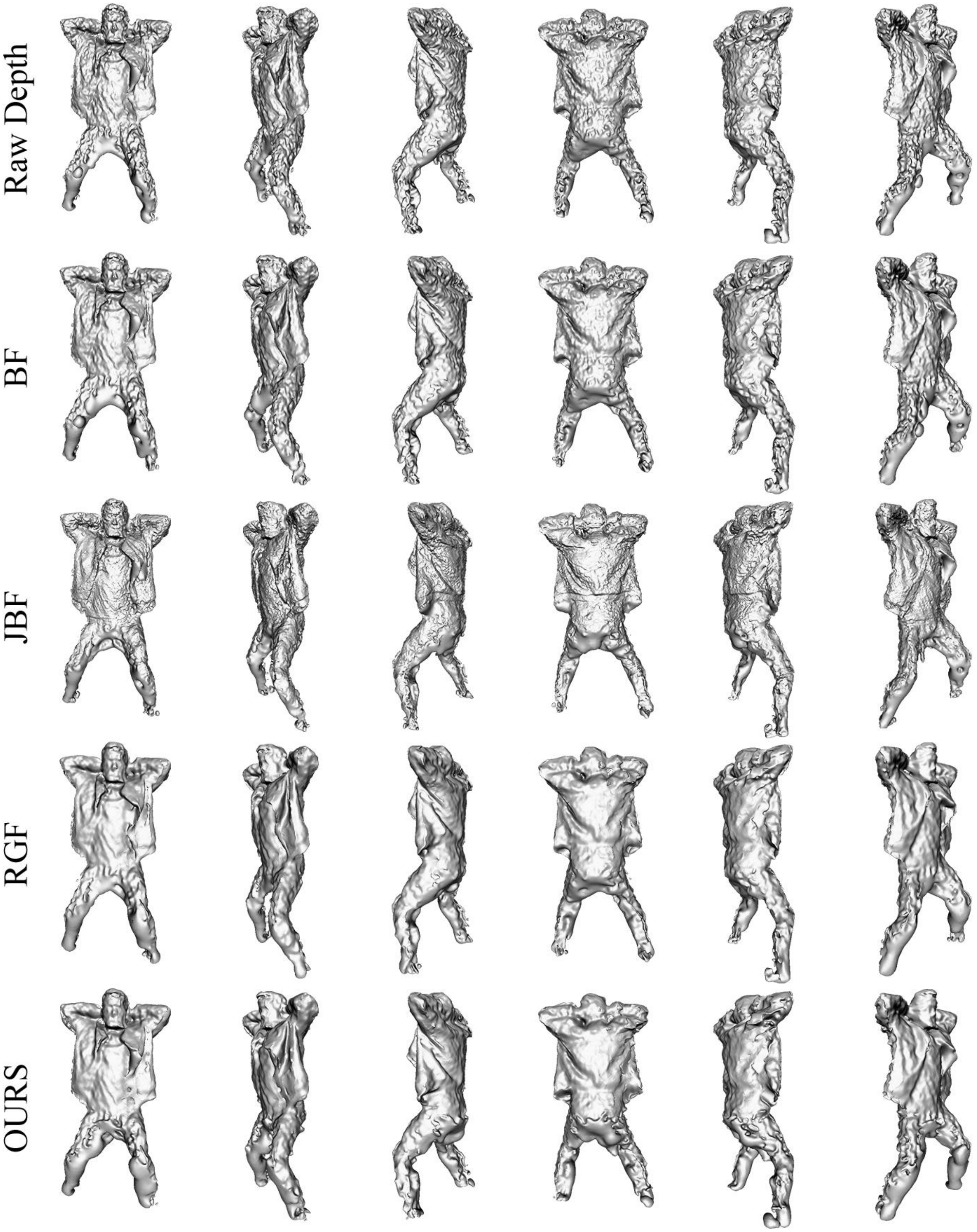

Figure 6. Poisson reconstruction sample with the sensors placed higher (looking downwards) and slightly further away from the target. This leads to erroneous surface estimations at the approximate leg region, mainly due to the partial visibility and data sparseness. Despite the challenging setup, our method was able to successfully remove noise and preserve fine details (*e.g.* face, jacket folds).

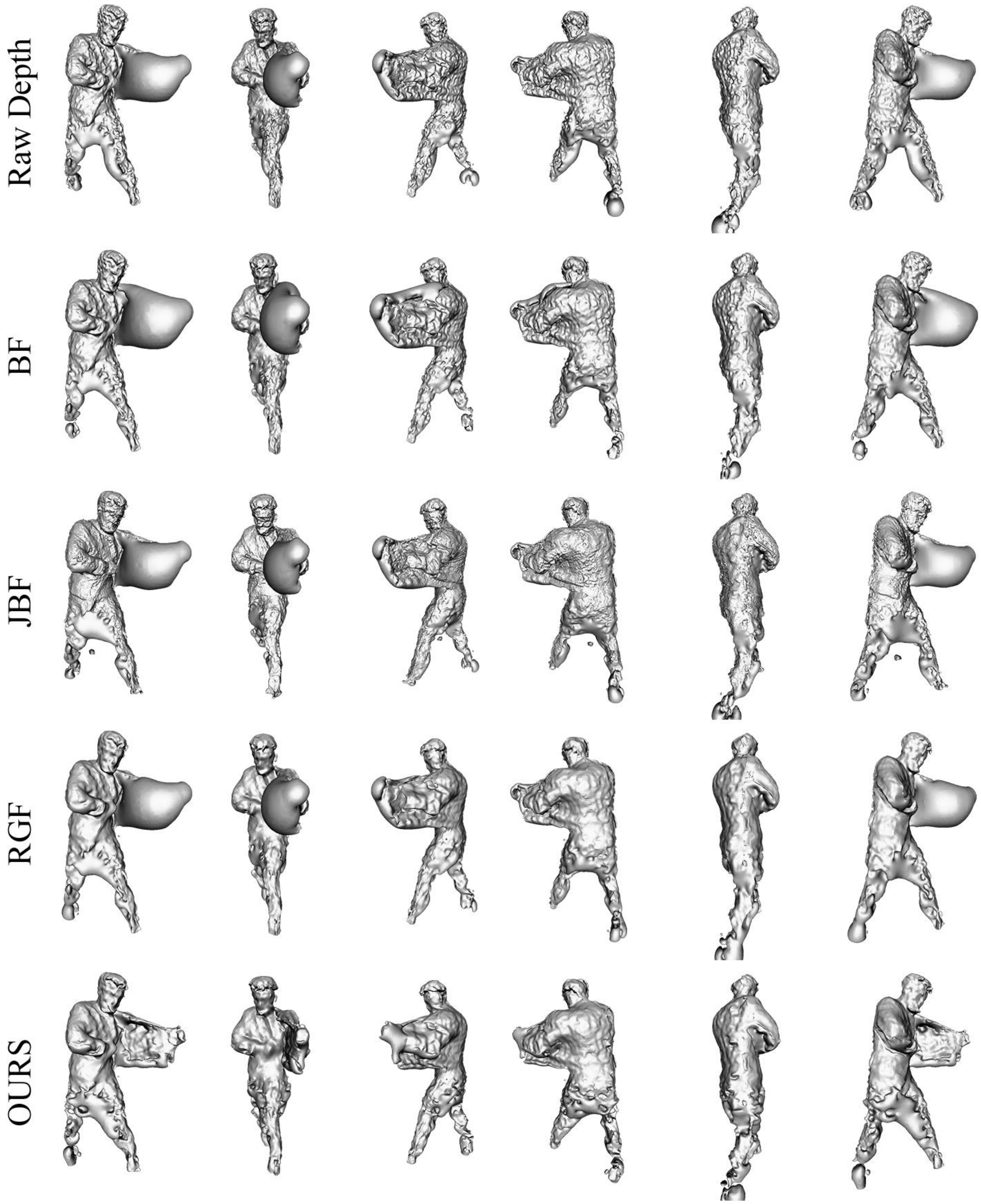

Figure 7. Poisson reconstruction sample using the setup described in Fig. 6. Our method is the only one to remove the “balloon” noise at the inner side of the jacket caused by noisy depth measurements.