Low Latency Point Cloud Rendering with Learned Splatting Supplementary Material

Yueyu Hu[1] Ran Gong[2] Qi Sun[1] Yao Wang[1]
[1]Tandon School of Engineering, New York University [2]Tsinghua University
{yyhu, qisun, yaowang}@nyu.edu rangong41@gmail.com

1. Experiments on Texture Adaptability

In addition to the benefit of rendering more accurate surface contour by allowing 3D Gaussians to have elliptical shapes, since the model also takes point color as input, it has the ability to adapt to local texture edges. We compare the rendering results from learned and globally set Gaussian parameters to elucidate this advantage. For the globally set parameters, we choose three different values of Gaussian standard deviation σ relative to $\bar{d}$, the average distance between points in the point cloud. As shown in Figure 1, using a lower global σ can lead to visible holes, while setting a larger σ result in inaccurate edges. The learned model can instead produce spatially varying 3D Gaussian parameters including center translations to provide clearer edges and smooth surface at the same time.

2. Additional Rendering Results

We visualize more rendering results with high-quality (800K points) point clouds from the THuman 2.0 dataset in Figure 3 and 4, outdoor scenes (1.5M points) from the BlendedMVS dataset in Figure 5, and noisy raw captures (1M points) from the CWIPC dataset in Figure 6 and Figure 7.

We also evaluate our method in dynamic quality by rendering the point cloud video sequences from the 8iVFB database [2]. We use a video framebuffer at the resolution of 1024×1024 and the frame rate at 30 fps, in accordance to the original point cloud framerate. Please refer to the supplementary video for the results. As shown, our method can render high-quality videos without visible holes which are observed with the standard OpenGL renderer. Therefore, our method is more suitable for rendering high-quality point cloud videos in real-time applications.

References

[1] Jen-Hao Rick Chang, Wei-Yu Chen, Anurag Ranjan, Kwang Moo Yi, and Oncel Tuzel. Pointersect: Neural rendering with cloud-ray intersection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8359–8369, 2023. 3, 4, 5, 6, 7, 8

[2] Eugene d'Eon, Harrison Bob, Taos Myers, and Philip A. Chou. 8i voxelized full bodies - a voxelized point cloud dataset. In *ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document WG11M40059/WG1M74006*, 2017. 1

Figure 1. Comparison of texture rendering results with our model predicted vs. globally set Gaussian parameters on a quantized point cloud. The insets visualize local details with $3\times$ zooming.

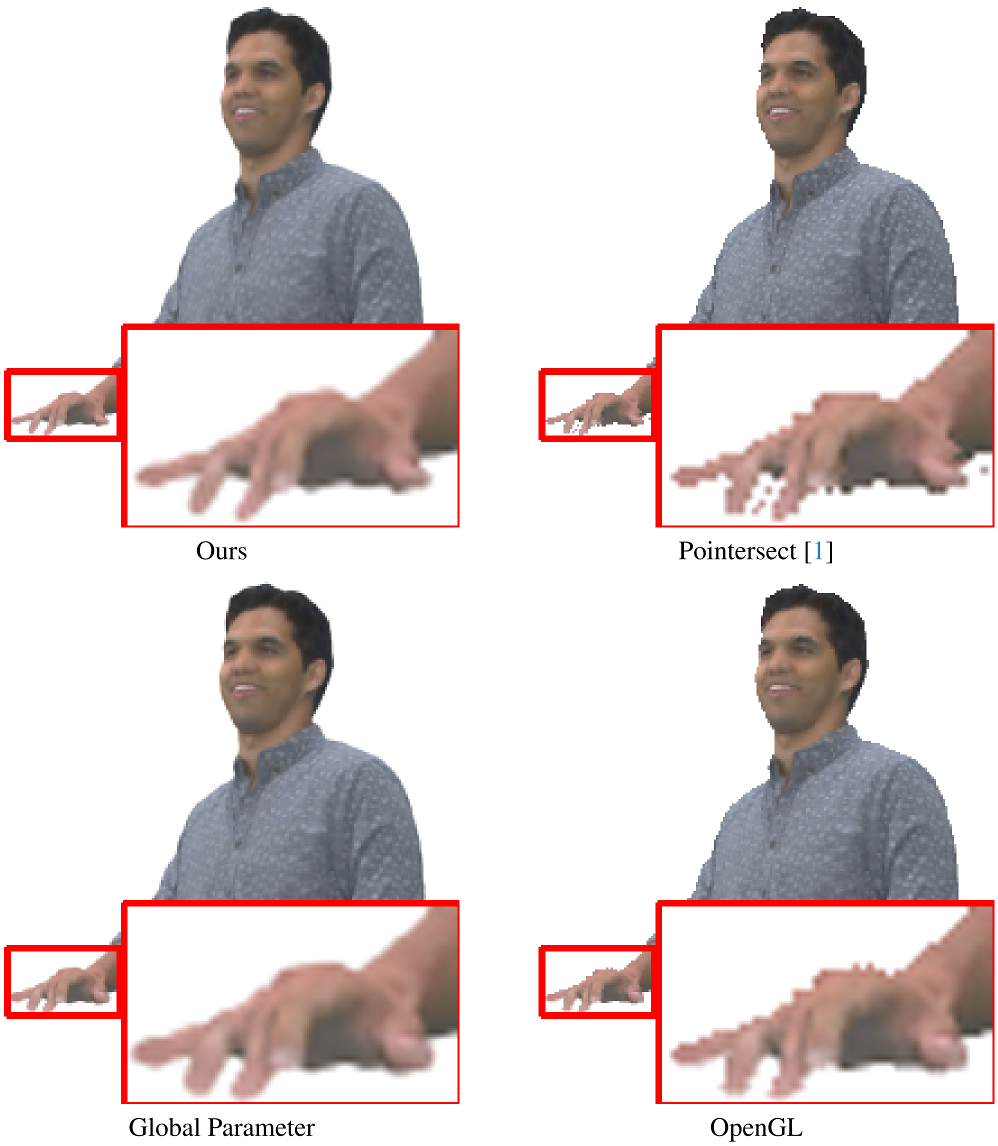

Figure 2. Rendering results of different methods on the sequence *Loot* (610 K points, 23 Mbps) in the 8iVFB database. The insets visualize local details with 3× zooming.

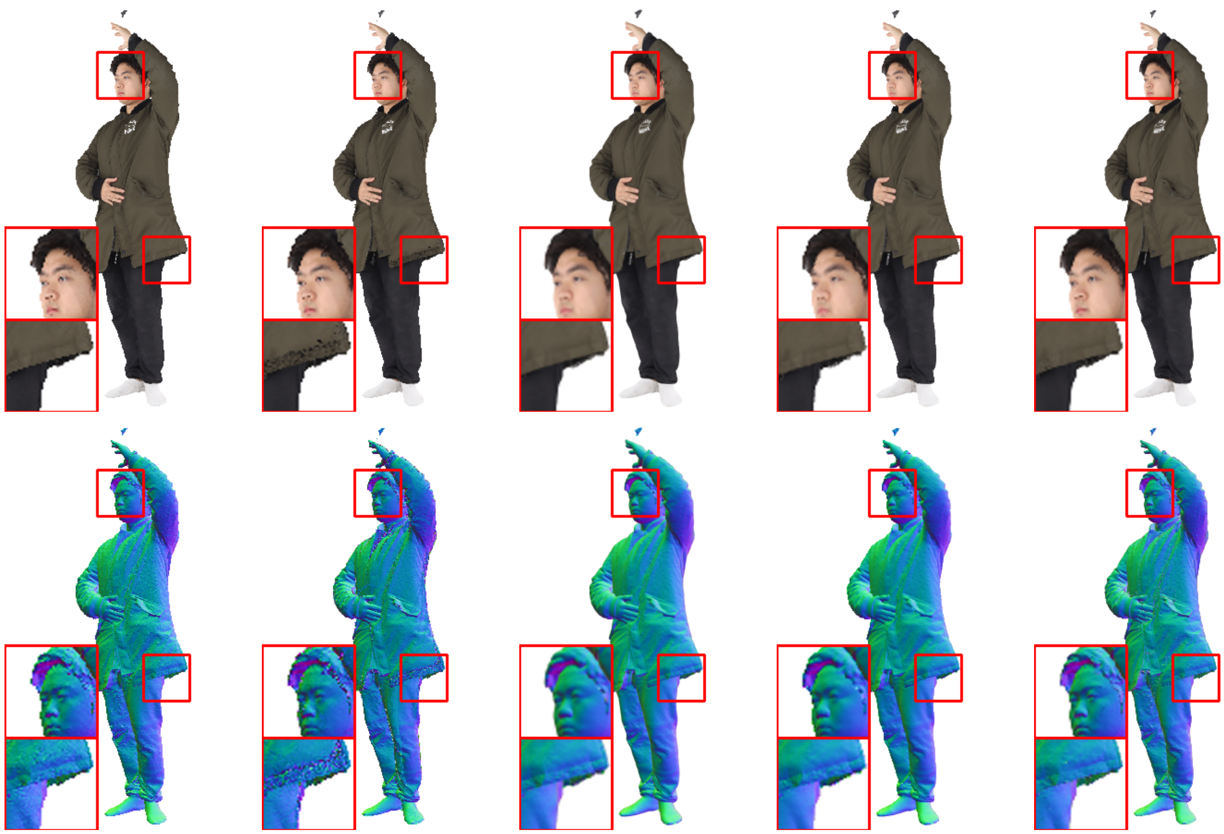

Figure 3. Rendering results of high quality point clouds in THuman 2.0 dataset. The insets visualize local details with 2× zooming.

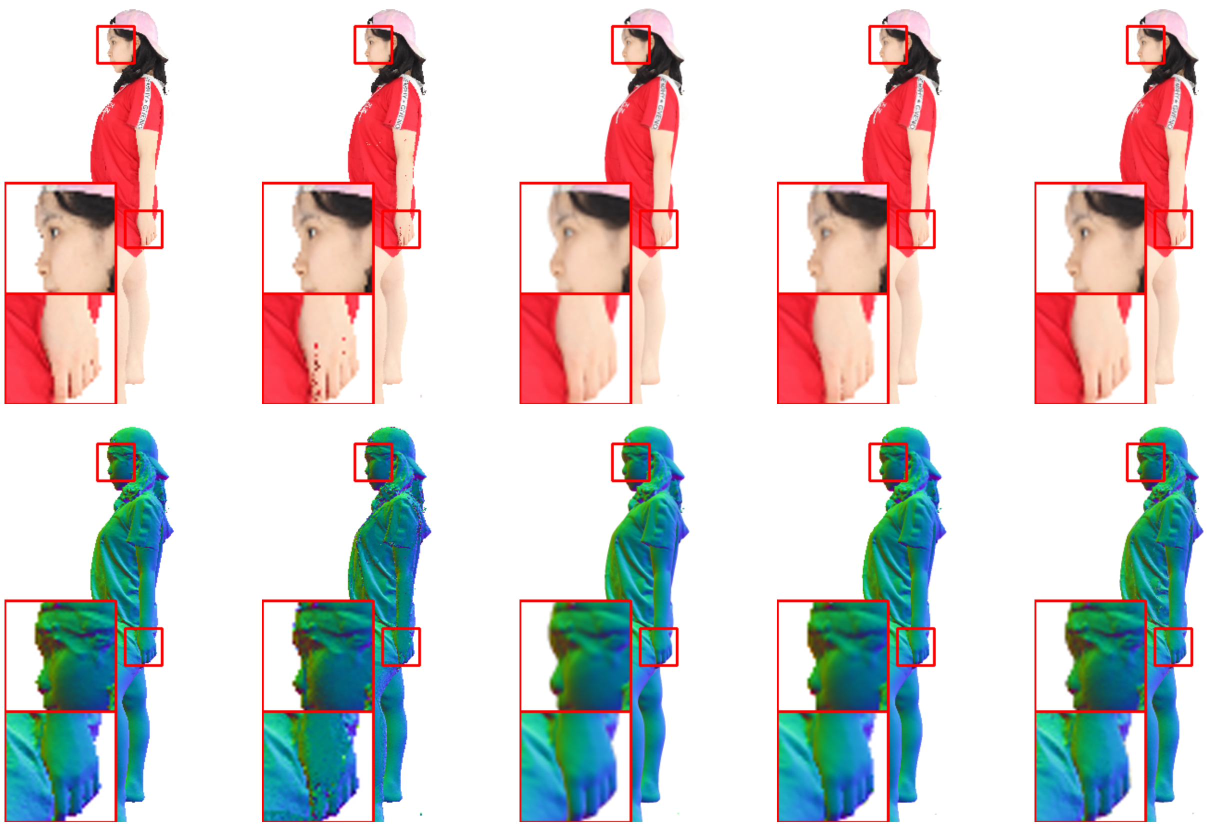

Figure 4. Rendering results of high quality point clouds in THuman 2.0 dataset. The insets visualize local details with 3× zooming.

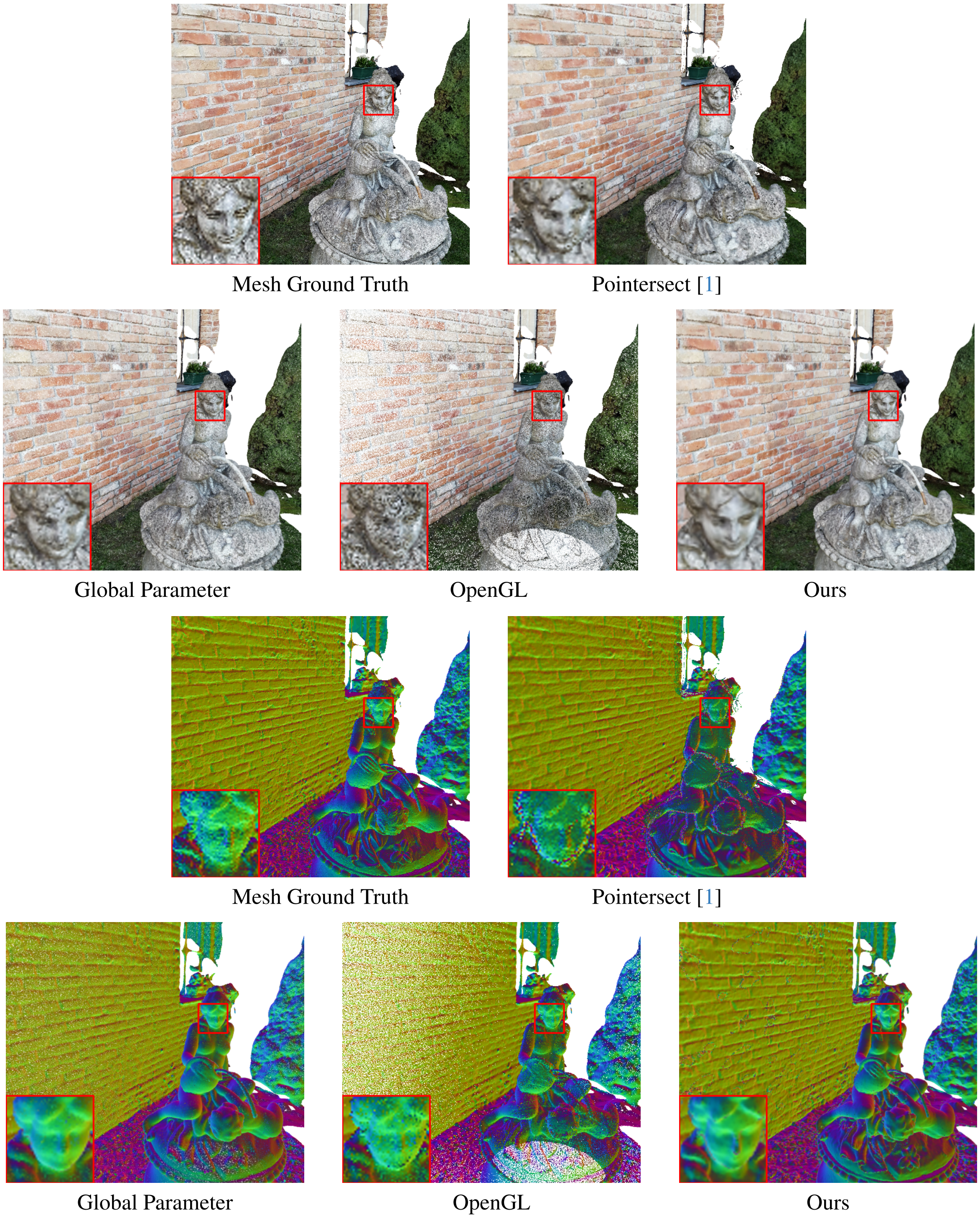

Figure 5. Rendering results of high quality point clouds in BlendedMVS dataset. The insets visualize local details with 3× zooming.

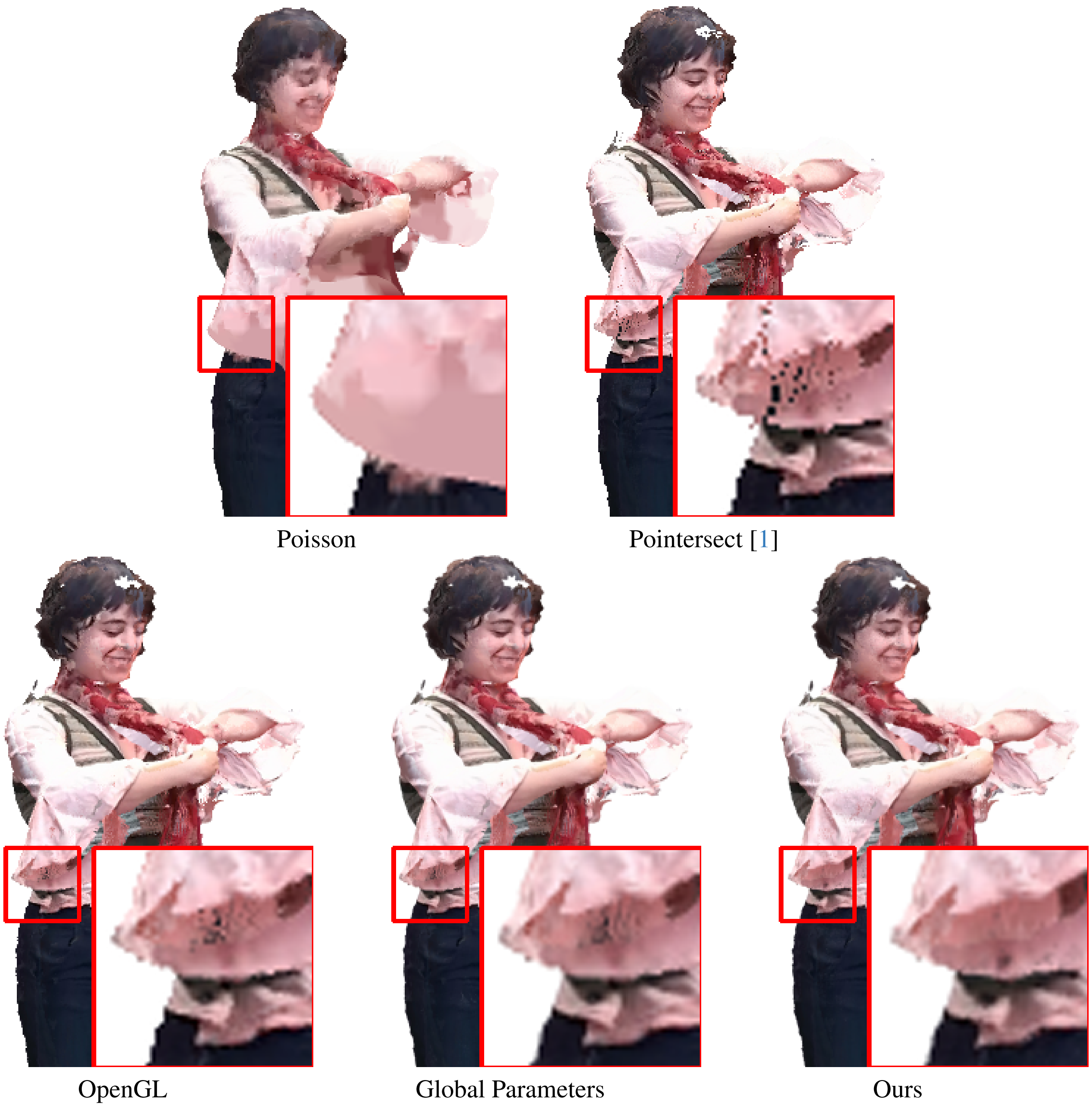

Figure 6. Rendering results from noisy raw point clouds in the CWIPC dataset. The insets visualize local details with $2\times$ and $3\times$ zooming, respectively.

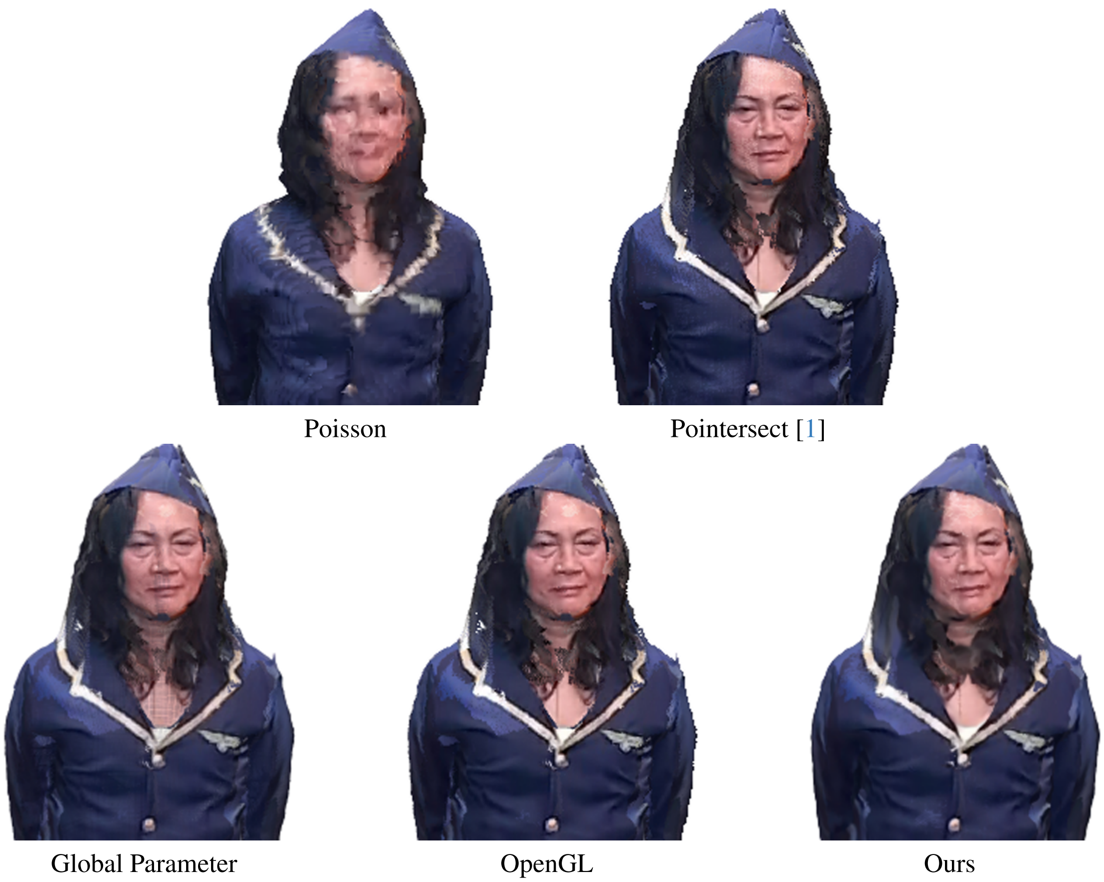

Figure 7. Rendering results from noisy raw point clouds in the CWIPC dataset.