Supplemental Material for 360MVSNet: Deep Multi-view Stereo Network with 360° Images for Indoor Scene Reconstruction

Ching-Ya Chiu[1] Yu-Ting Wu[2] I-Chao Shen[3] Yung-Yu Chuang[1]
[1]National Taiwan University [2]National Taipei University [3]The University of Tokyo

In this document, we first describe additional implementation details of our method. We then show some example triplets of the 360° color, depth, and mask images of our *EQMVS* dataset in Section 2. Next, we provide more results and comparisons in Section 3, including equal-effort qualitative comparison (Section 3.1), evaluation on the number of cameras (Section 3.2), and ablation study on multi-scale cost volumes (Section 3.3).

1. Implementation Details

Training. During training, we use Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We first train the first stage for 10 epochs, then train all the three stages for another 10 epochs with batch size 4. The initial learning rate is set to 0.001, then decays with the rate of $1/2$ at the 10th, 12th, and 14th epoch. While previous works [3] use view selection strategy to favor certain angles between camera views, we choose the source views that have a smaller baseline distance to the reference view because 360° images are omnidirectional by nature. Our input image resolution is 1024×512. The spatial resolution of the feature maps at s^{th} stage is $(1/2)^{3-s}$ of the initial image resolution and the weight of the loss at each s^{th} stage is 2^{s-2}.

Post-processing. For geometric consistency, we check the mutual projection among multiple views. In particular, for a pixel p from the reference view, we project it through its depth d_p to pixel q at another view and we back-project pixel q by its depth d_q to pixel $p^{'}$. We set the thresholds so that $|p - p^{'}| < \epsilon_p$ and $|d_p - d_{p'}|/d_p < \epsilon_d$, where $\epsilon_p = 10$ (pixel) and $\epsilon_d = 0.1$.

2. EQMVS

Figure 1 shows three example scenes of our *EQMVS* dataset. For each example scene, we plot the positions of placed cameras to illustrate the camera distribution in our dataset. On the right side of each scene, we show several example triplets, each containing the RGB image, depth map, and mask corresponding to a camera.

Table 1: Comparisons of the reconstruction quality of our model with and without multi-scale cost volumes.

Methods	Acc.(m)↓	Comp.(m)↓	Overall(m)↓
w/o multi-scale	0.0453	0.1099	0.0776
w/ multi-scale	0.0810	0.0579	**0.0694**

3. More Results

3.1. Equal-effort Qualitative Comparison

Figure 2 shows more qualitative comparison with other methods in the equal-effort setting. In this experiment, all methods use the input images captured at the same locations to reconstruct the point cloud. In order to conduct a fair comparison, we warp the input test images from equirectangular projection to cubemap projection for previous methods to avoid distortion since previous methods are all designed for normal FoV images. Our method significantly improves completeness and visual quality compared to previous methods.

3.2. Evaluation on the Number of Cameras

In this experiment, we compare the results of our method with 25 images and other methods with different numbers of cameras. To generate input images that mimic the capturing process for MVS tasks with normal FoV images, we randomly place cameras with 70° FoV in the scene and set the elevation angle of the cameras randomly between 30°and 150°. Then, we manually filter out invalid views that are placed within objects. Figure 3 demonstrates the point cloud results of COLMAP [2], OpenMVS [1], and MVS-Net [3] with 100, 200, 300, 400, and 500 input images and the results generated by our method with 25 360°images.

3.3. Ablation Study on Multi-scale Cost Volumes

We show the effectiveness of the multi-scale cost volume with uncertainty estimation. In this experiment, the model with multi-scale is trained with 3 stages where depth hy-

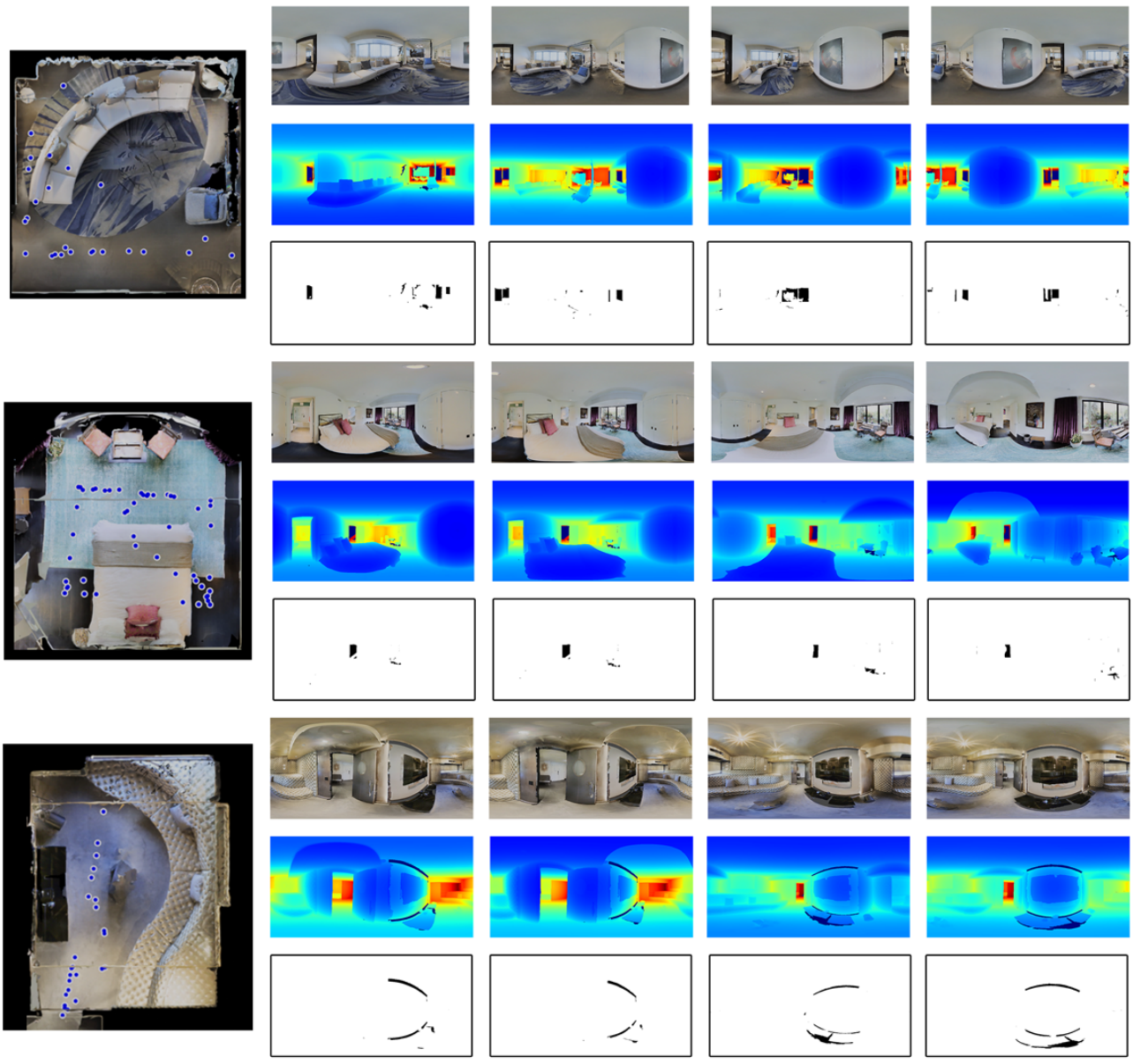

Figure 1: **Example scenes in the *EQMVS* dataset.** We demonstrate three example scenes in our dataset. For each scene, we show all the camera positions (blue points) on the left and four triplets of the color image, depth and mask on the right.

potheses in each stage are 160, 32, 8 , while the model without multi-scale is trained directly with 192 depth hypotheses. Figure 4 demonstrates the qualitative comparisons between the model with and without multi-scale depth estimation. The insets show that the model with multi-scale cost volumes can produce more accurate results, especially for fine structures and regions around object boundaries. The statistics in Table 1 provide the reconstruction quality of these two methods. Multi-scale cost volumes achieve better performance in terms of completeness and overall quality.

In this section, we further compare the results between the method that directly upsamples the low-resolution depth map and the one that uses a multi-scale cost volume. Table 2 shows that the method with three stages can achieve better completeness and overall score. Figure 5 shows qualitative comparisons of the point cloud results in different stages and the ground truth point cloud. This comparison demonstrates that the completeness of reconstruction is

Figure 2: **Results of equal-effort qualitative comparison.** We use 25 360° images for scene (a), (h) and 49 360° images for scene (b)-(g). We convert the equirectangular images into cubemaps as the input images for COLMAP[2], OpenMVS[1], and MVSNet[3]. In total, for compared methods, we use 150 images for scene (a), (h) and 294 images for scene (b)-(g). We show the scores of accuracy/completeness/overall quality underneath each method's result (lower is better) and mark the best result in red. Our method achieves the best completeness and overall scores across all examples.

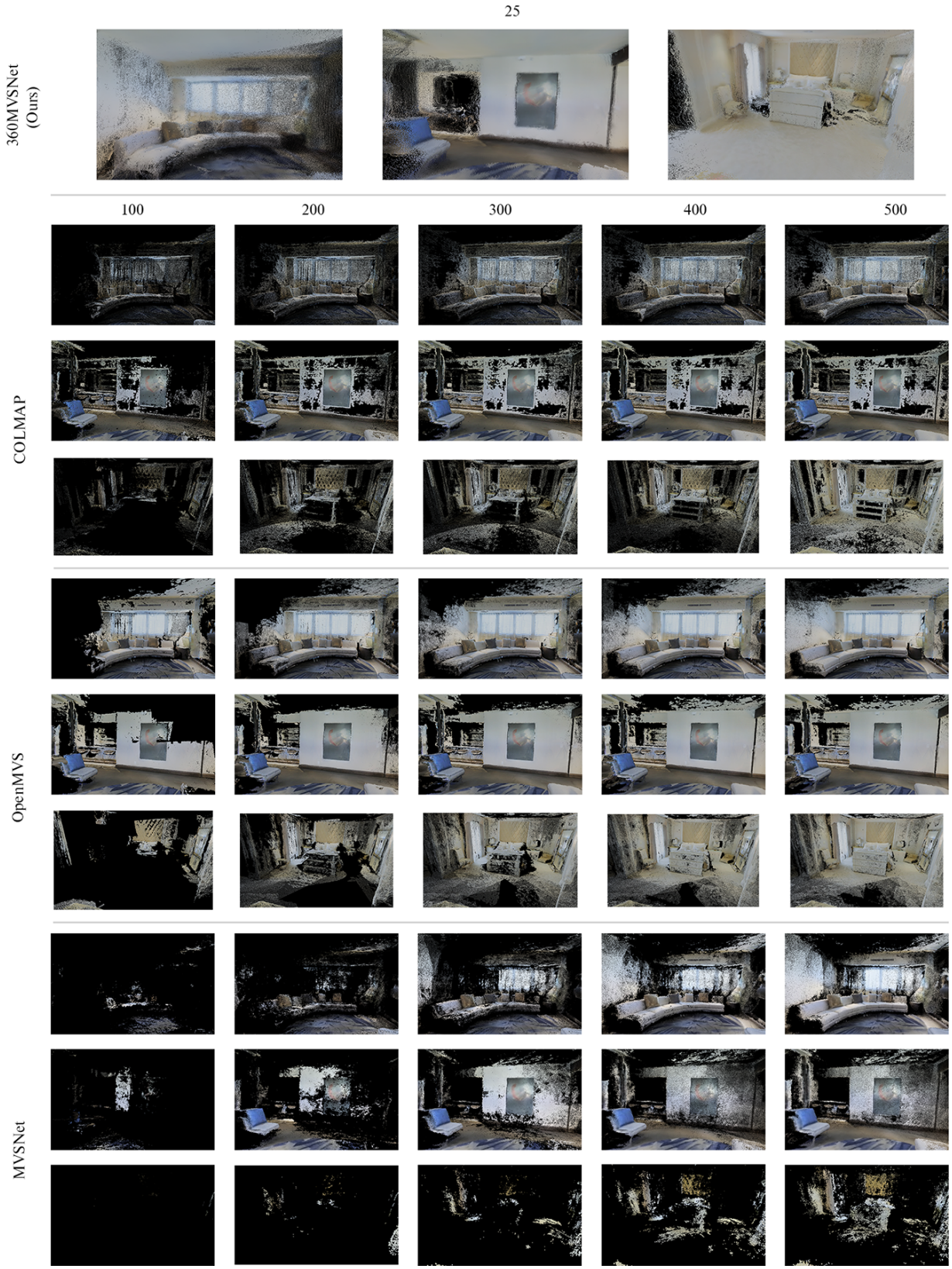

Figure 3: **Comparisons of the results with different numbers of cameras.** We show three scenes reconstructed using our method and other methods (COLMAP [2], OpenMVS [1], and MVSNet [3]). For each compared method, we show the reconstructed point cloud using different numbers of images in each row.

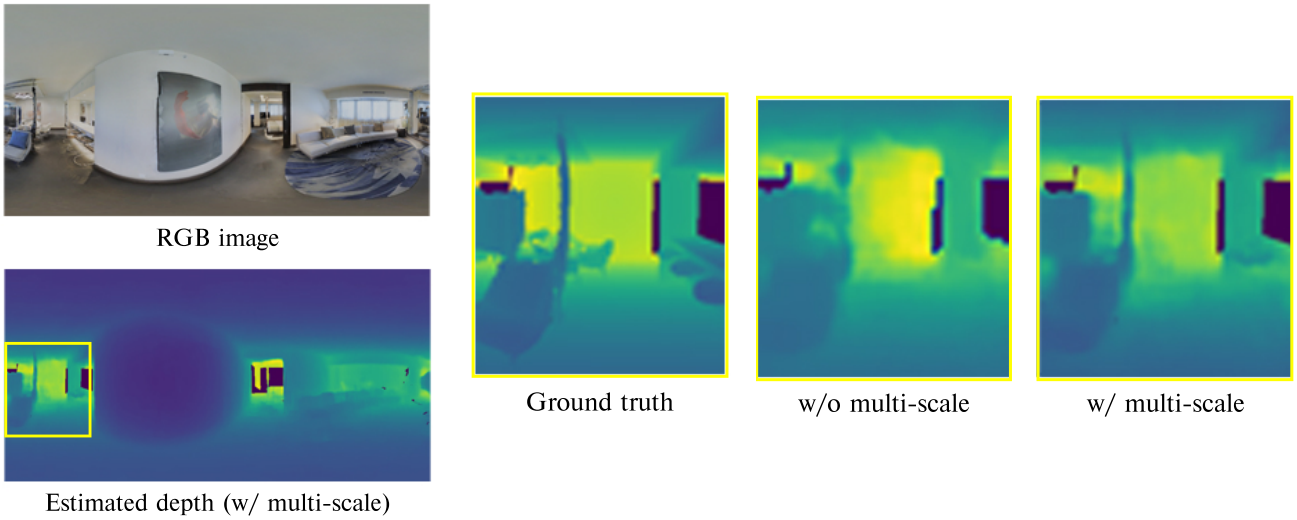

Figure 4: **The comparisons of depth map w/o and w/ multi-scale cost volumes. Top row**: RGB image and the full estimated depth using multi-scale cost volume. **Bottom row**: insets of the ground truth depth map, estimated depth w/o multi-scale, and estimated depth w/ multi-scale. Object boundaries and fine structures are better preserved using mutli-scale cost volume.

Figure 5: **Results of multi-scale cost volume ablation study.**

gradually improved with multiple stages.

References

[1] Dan Cernea. OpenMVS: Multi-view stereo reconstruction library. 2020.

[2] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *Proc. European Conference on Computer Vision (ECCV)*, 2016.

[3] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long

Table 2: Comparisons of the results of different stages.

Methods	Size	Acc.(m)↓	Comp.(m)↓	Overall(m)↓
stage2 w/ upsample	1024×512	0.0666	0.0894	0.0780
stage3	1024×512	0.0810	0.0579	**0.0694**

Quan. MVSNet: Depth inference for unstructured multi-view stereo. *Proc. European Conference on Computer Vision (ECCV)*, 2018.