CompNVS: Novel View Synthesis with Scene Completion — Supplementary Materials —

Zuoyue Li[1], Tianxing Fan[2], Zhenqiang Li[3], Zhaopeng Cui[2], Yoichi Sato[3], Marc Pollefeys[1,4], and Martin R. Oswald[1,5]

[1] ETH Zürich [2] Zhejiang University [3] The University of Tokyo
[4] Microsoft [5] University of Amsterdam

In this supplementary material, we provide additional quantitative and qualitative experiment results (Sec. 1), as well as more details on the network architecture and evaluation (Sec. 2).

1 Additional Experiments

1.1 Larger Missing Area

To test the trained model reported in the main paper on the more challenging validation scenes, we conduct experiments on Replica with a larger gap between input views, where the missing pixel ratio in the target view is 50%-55%. Exemplary results are shown below, the video SSIM/LPIPS scores are **0.696/0.347** for ours and 0.579/0.387 for PixelSynth. Generally, both models perform worse than in Tab. 3 of the main paper but PixelSynth drops more. This may be due to PixelSynth completing within the limited image range, where the input information is reduced and the missing area is larger.

1.2 Model Generalization

Our method is not only applicable for the two-view interpolation task shown in the main paper, but is also generalizable and can be adapted to other settings, depending on what kind of training data is provided to the pipeline. We provide exemplary results of applying the pipeline to tasks of single-view extrapolation in Fig. 2 (completion outside the red box) and multi-view completion in Fig. 3 (3 views for room completion), using models that are slightly fine-tuned from our original one used in the main paper.

Fig. 1. Completion results on two input views with a larger gap. Please use **Adobe Reader / KDE Okular** to see ***animations***.

Fig. 2. Results of single-view extrapolation on Replica dataset. The proposed pipeline can also be used to outpaint scenes when only single views are given.

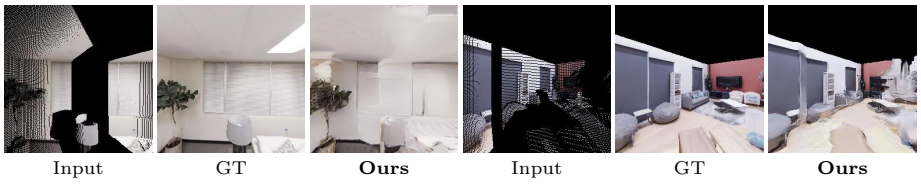

Fig. 3. Results of multi-view completion on Replica data. Our method is also generalizable to the task of scene completion from three disjoint views. Please use **Adobe Reader / KDE Okular** to see ***animations***.

Fig. 4. Comparison of the rendering results in two different resolutions. Please use **Adobe Reader / KDE Okular** to see ***animations***.

1.3 Higher Resolution Rendering

The training with the discriminator requires the image to be fully rendered on the fly. However, the computational efficiency of the high-resolution rendering is limited, which makes the training in high resolution very time-consuming and memory hungry. Therefore, the rendered image resolution is set to be 256×256 (Replica) or 256×192 (ARKitScenes) in the main paper. Nonetheless, we can still render higher resolution images (512×512 for Replica and 512×384 for ARKitScenes) during inference, and several exemplary results are shown below. We see that the improvement of visualization is limited which is likely due to the unchanged voxel resolution.

1.4 Qualitative Depth Comparison

We also compare the query frame depth generated by different methods qualitatively in Fig. 5. Generally, our method can generate more accurate geometry that is compatible with the given observations.

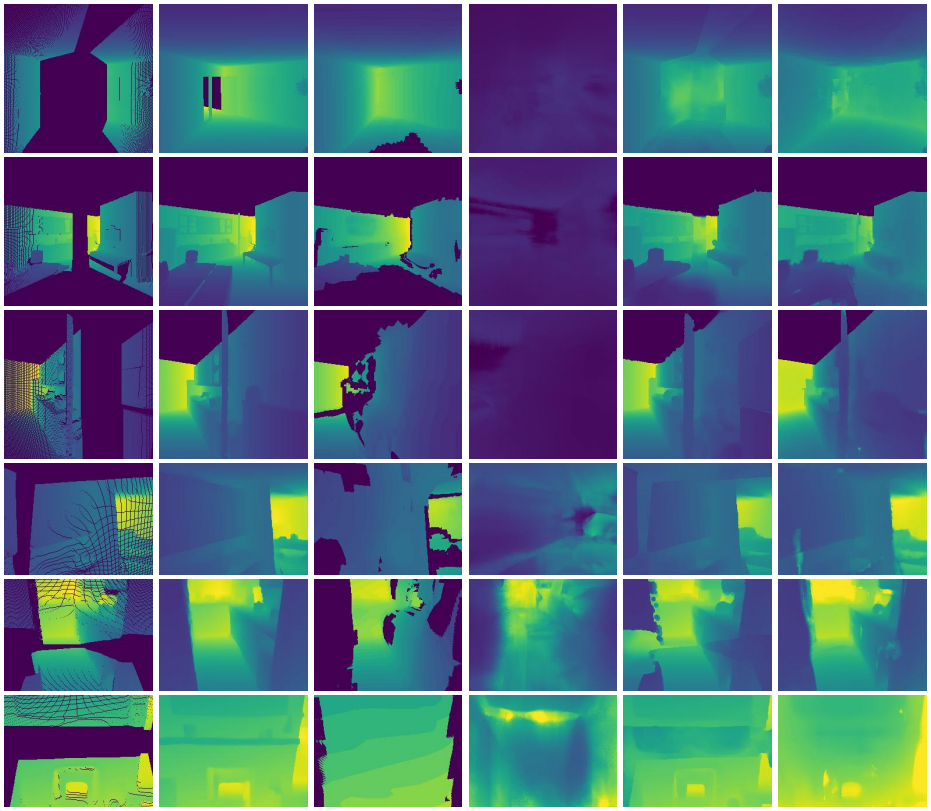

Fig. 5. Qualitative depth comparison. We present exemplary qualitative depth results for the query view. The first 3 rows are from Replica dataset while the last 3 rows are from ARKitScenes dataset.

1.5 Failure Cases

Fig. 6 shows exemplary failure cases mainly due to the following two reasons. For the first and third ones, since we only use the query frame ground truth as the training target, it is likely that there is a missing geometry prediction on the intermediate frames if the trajectory jumps significantly. For the second and fourth ones, the texture is either inconsistent or too blurry, which could result from a bad embedding filling or the camera being too close to the object surface but we have a fixed voxel size.

1.6 Qualitative Baseline Comparison

Additional qualitative baseline comparisons are depicted in Fig. 7 (Replica [9]) and Fig. 8 (ARKitScenes [1]). It can be observed that ours CompNVS can generate more plausible videos with better 3D consistency and contains fewer artifacts.

1.7 Qualitative Ablation Study

Further qualitative ablation study results are provided in Fig. 9. The effectiveness of each added component is shown by clear visual image quality improvements.

2 Additional Details

2.1 Network Architecture Details

Our implementation is mainly based on Minkowski Engine [2] and NSVF [7].
Encoder. The input point cloud is initially voxelized with a size of 0.625cm ($\frac{1}{16}$ of the 10cm voxel size for the sparse voxel field) and processed by 4 dilated convolutions with 32 channels using (1, 2, 4, 8) dilations for a larger receptive field. The main encoder uses a ResNet [5] based backbone with sparse convolutions, having 4 downsampling operations with each stage, (2, 3, 3, 3) bottleneck blocks and (32, 64, 128, 128) channels.
Geometry Predictor. The geometry predictor directly employs the completion network implementation in Minkowski Engine [2], using a U-Net structure with 6 downsampling and upsampling stages. It takes as input the 1-channel sparse input tensor representing the scene occupied locations. Each upsampling layer in the decoder part uses generative transposed convolutions [4]. The network further uses batch normalization and ELU activation.
Texture Inpainter. The texture inpainter directly employs the vanilla U-Net implementation of Minkowski Engine [2], using an encoder backbone of ResNet50 with 4 downsampling stages. The upsampling layers in the decoder part use general transposed convolutions.
Renderer. The renderer directly employs NSVF [7]'s implementation with some necessary adaptions. As mentioned in the main paper, besides RGB values, the embeddings are aggregated to 2D pixels as well as based on the calculated alpha value. Moreover, we use 2 separate MLPs for disentangling RGB and alpha values, with the same structures as the default one used in NSVF [7].
Upsampler. The upsampler contains 3 basic residual blocks for each of the 2 resolutions with a single bicubic interpolation for upsampling.
Discriminator. The discriminator directly employs the implementation used in BicycleGAN [12], which makes predictions on image blocks.

2.2 Evaluation Metric on Masked Image

For Tab. 1, 2, and 4 in the main paper, we report evaluations on the masked area in the center frame. Here the masked area refers to the generated scene part in the query view that is not visible in the input observations. For per-pixel PSNR and depth metrics, the masked pixels are extracted for evaluation. For other metrics, we calculate as follows.
FID [6]. Since the inception network uses max-pooling layers, the metric can evaluate the features activated by the masked area. The unmasked area in the

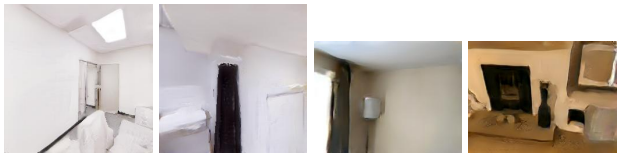

Fig. 6. Exemplary failure cases. The first two examples are from Replica and the rest two ARKitScenes. Please use **Adobe Reader / KDE Okular** to see ***animations***.

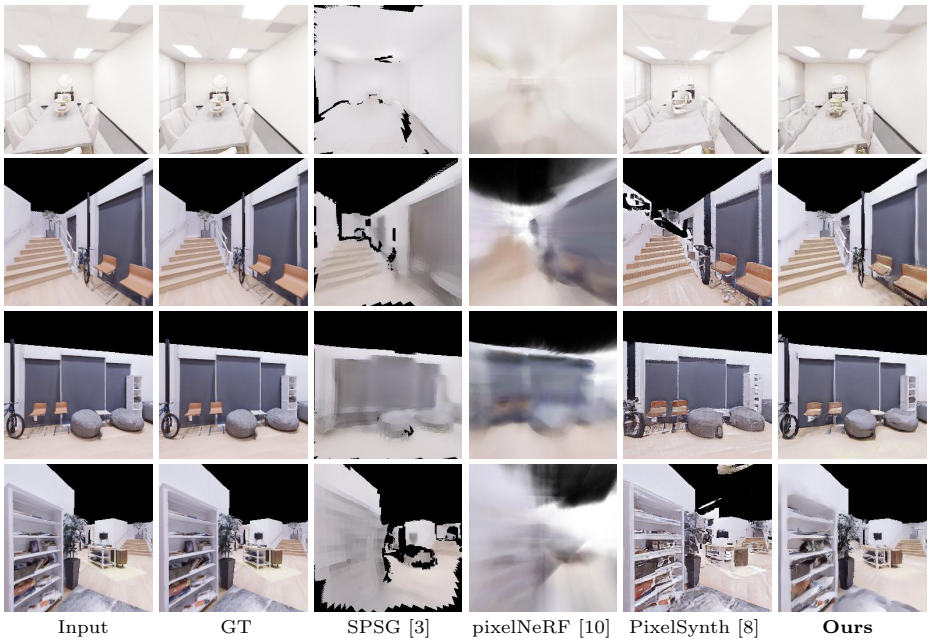

Fig. 7. Supplementary qualitative baseline comparison on the Replica dataset. We show more additional comparisons to state of the arts. Ours generates more realistic videos with better temporal consistency and contains fewer artifacts. Please use **Adobe Reader / KDE Okular** to see ***animations***.

prediction is filled with gray color before the calculation. The resulting distribution is compared with the GT image distribution (without gray color filling).

SSIM. The unmasked area is filled with the corresponding GT image and the SSIM score is then calculated on the masked area only.

LPIPS [11]. The unmasked area is filled with the corresponding GT image and the perceptual similarity is then calculated as general and further divided by the ratio of the masked area. Here we assume that the perceptual similarity for the whole image is a weighted score average of masked and unmasked areas.

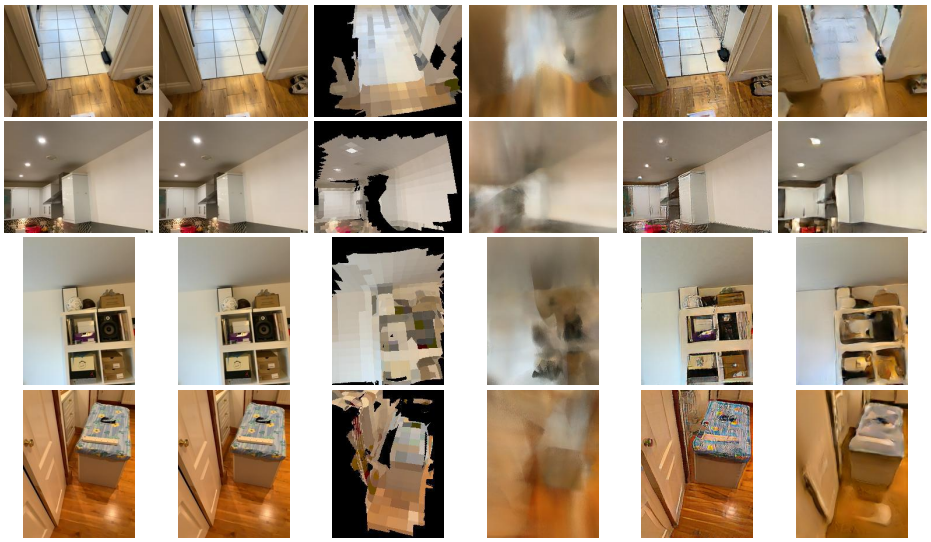

Fig. 8. Supplementary qualitative baseline comparison on the ARKitScenes dataset. We show more additional comparisons to state of the arts. Ours generates more realistic videos with better temporal consistency and contains fewer artifacts. Please use **Adobe Reader / KDE Okular** to see ***animations***.

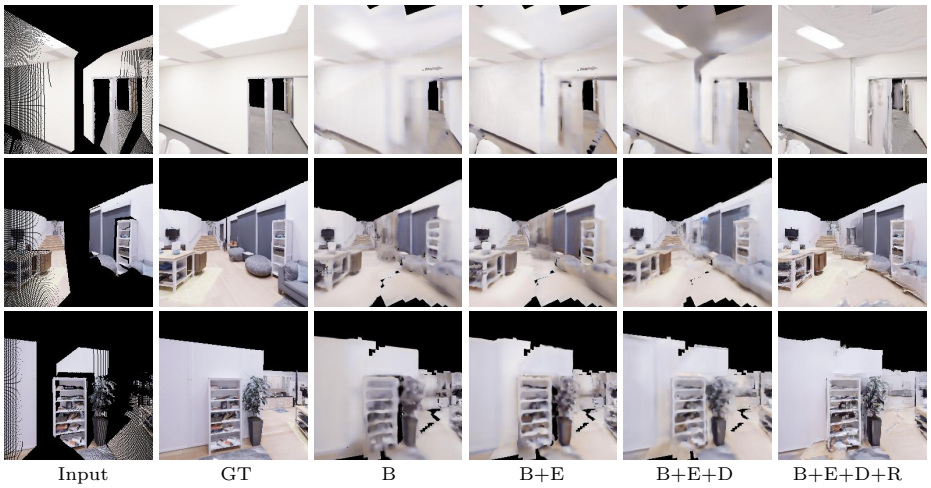

Fig. 9. Supplementary qualitative ablation study. We present more additional exemplary qualitative results for various ablations of our method.

References

1. Baruch, G., Chen, Z., Dehghan, A., Dimry, T., Feigin, Y., Fu, P., Gebauer, T., Joffe, B., Kurz, D., Schwartz, A., Shulman, E.: ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1) (2021)
2. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019)
3. Dai, A., Siddiqui, Y., Thies, J., Valentin, J., Nießner, M.: Spsg: Self-supervised photometric scene generation from rgb-d scans. In: Proc. Computer Vision and Pattern Recognition (CVPR), IEEE (2021)
4. Gwak, J., Choy, C.B., Savarese, S.: Generative sparse detection networks for 3d single-shot object detection. In: European conference on computer vision (2020)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
6. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)
7. Liu, L., Gu, J., Lin, K.Z., Chua, T.S., Theobalt, C.: Neural sparse voxel fields. NeurIPS (2020)
8. Rockwell, C., Fouhey, D.F., Johnson, J.: Pixelsynth: Generating a 3d-consistent experience from a single image. In: ICCV (2021)
9. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., Clarkson, A., Yan, M., Budge, B., Yan, Y., Pan, X., Yon, J., Zou, Y., Leon, K., Carter, N., Briales, J., Gillingham, T., Mueggler, E., Pesqueira, L., Savva, M., Batra, D., Strasdat, H.M., Nardi, R.D., Goesele, M., Lovegrove, S., Newcombe, R.: The Replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
10. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4578–4587 (June 2021)
11. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018)
12. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: Advances in Neural Information Processing Systems (2017)