Supplementary Materials - Skyeyes: Ground Roaming using Aerial View Images

Zhiyuan Gao[1,2,*] Wenbin Teng[1,2,*] Gonglin Chen[1,2] Jinsen Wu[1,2]
Ningli Xu[3] Rongjun Qin[3] Andrew Feng[2] Yajie Zhao[1,2,†]

[1]University of Southern California [2]Institute for Creative Technologies [3]The Ohio State University
{gaozhiyu, wenbinte, gonglinc, jinsenwu}@usc.edu
{xu.3961}@buckeyemail.osu.edu {Qin.324}@osu.edu {feng, zhao}@ict.usc.edu

In this supplementary material, we provide more details of our dataset collection in Section 1. After that, we provide additional qualitative result in Section 2 and additional ablation studies in Section 3. In addition, based on the limitation of this work introduced in the main paper, we discuss our potential future work in Section 4.

1. Dataset Collection

1.1. CARLA Simulator

The CARLA Simulator [1] provides comprehensive Python API to facilitate interactions between users and environment. We leverage the Python API to build connections with the CarlaUE4 server, load the target map, add ego vehicle and multiple sensor cameras. We design customized trajectories for the vehicle and render the whole scene with sensor cameras. The first 4 rows of Figure 1 illustrates the top down view of each town that we extract data from together with an example of aerial/ground pairs. For more examples, please see our supplementary video. We separate each scene with multiple lanes. Within each lane, we spawn cameras to capture a color image for every 2 meters. Camera positioning is automatically optimized by CARLA to adhere to constraints like maintaining a safe distance from buildings, and the camera orientation is adjusted to face the direction of travel. For each point sampled on the lane, we set the yaw value of camera rotation to vary within $k\pi/4$, where $k = 0...7$. The altitude of aerial sequence is set to be 52 meters while the altitude of ground sequence is 2 meters. Pitch value of camera rotation is set to be -45 degrees for aerial views whereas 0 for ground views. For training-evaluation purposes, we set all the extracted data from Town04 for evaluation and all the rest for training.

1.2. CitySample

As discussed in the main paper, we follow the same data extraction pipeline as MatrixCity [2]. We only manipulate the rotation and position of camera trajectories to extract our customized data. Similarly, please refer to the last row of Figure 1 and our supplementary video for examples of CitySample dataset. The data extraction protocol mirrors the data extraction strategy employed in the CARLA Simulator, where the starting and ending points of each lane were manually determined. The configuration of camera poses in this environment closely aligns with those in CARLA, with the notable distinction that aerial sequences are captured at an altitude of 100 meters. We choose region 5 as the test set, and region 1, 2, 3, 4 and 6 as the training set (See Figure 3 in our main paper)

2. Additional Visualization

2.1. Additional Results

Figure 2 provides visualization of addition evaluation of our method on the test set of CARLA and CitySample dataset

2.2. Long Video Generation

As discussed in the main paper, we first use appearance control module to generate the first image, which is further applied as a condition to generate the following frames. For longer video generation, we simply use the last frame generated from current sequence as the new condition to generate next sequence. Please refer to our supplementary video for long video generation results.

3. Additional Ablation

Figure 3 provides visualization of ablation experiments of view consistency module. Also, please refer to our supplementary video for more detailed visualizations.

[*] Equal Contribution
[†] Corresponding Author

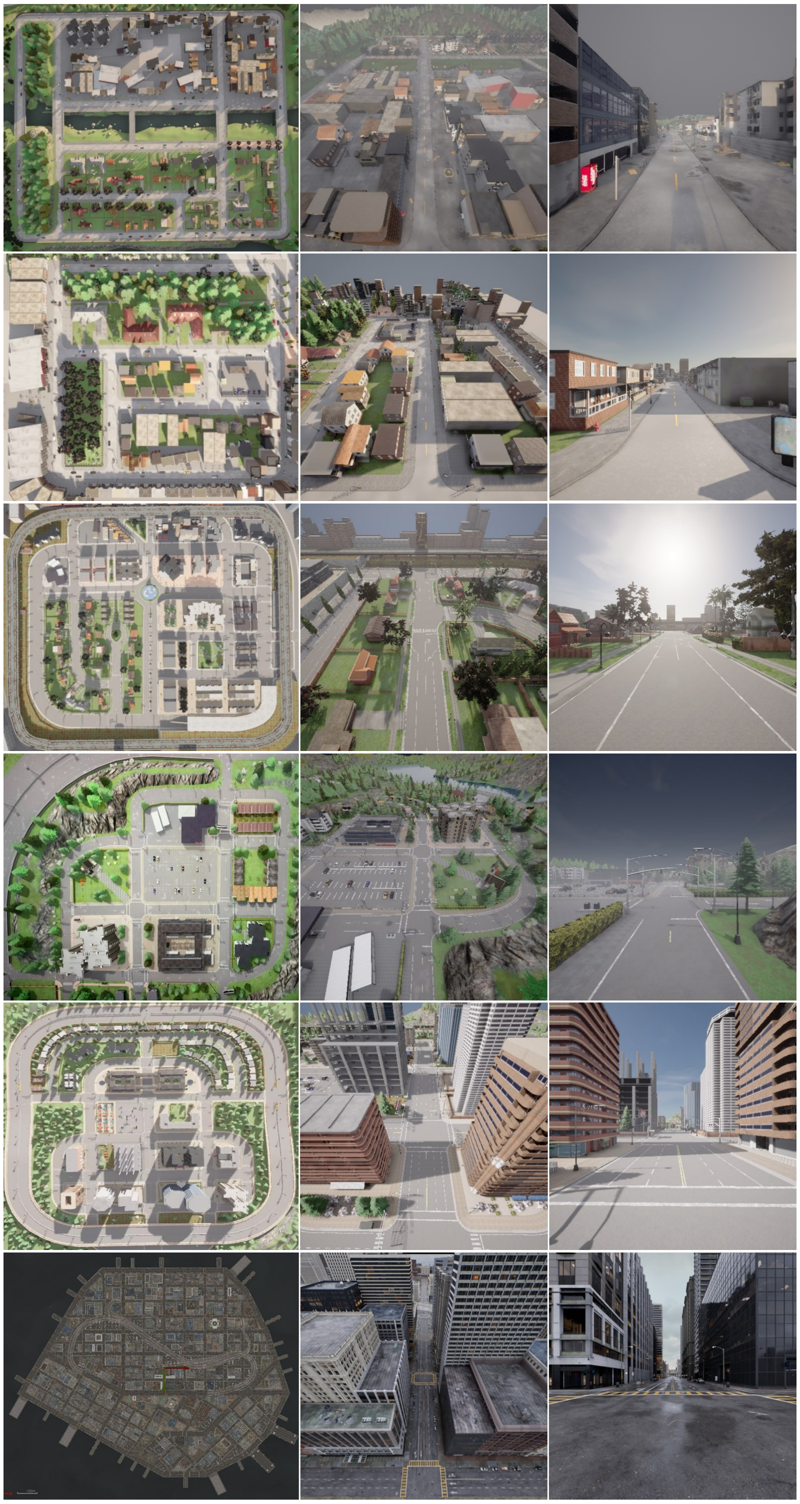
4. Discussion and Future Work

As discussed in the main paper, our proposed method does not generalize well to the realistic data. This is mainly due to the lack of scale and variety of the training data, which is currently limited to synthetic data with two open source platforms. However, the extraction of large amount of geo-aligned aerial-to-ground pairwise data is very costly and the acquisition should abide by the local rules and policies. Therefore, our next step is to perform unsupervised domain adaptation in diffusion model to bridge the gap between synthetic and realistic dataset.

References

[1] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017. 1, 3

[2] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3205–3215, 2023. 1



(a) Map (b) Aerial (c) Ground

Figure 1. **Dataset Visualization.** The first four rows are Town01, Town02, Town03, Town04 and Town05 of CARLA Simulator [1], respectively. The last row is a visualization of CitySample dataset.

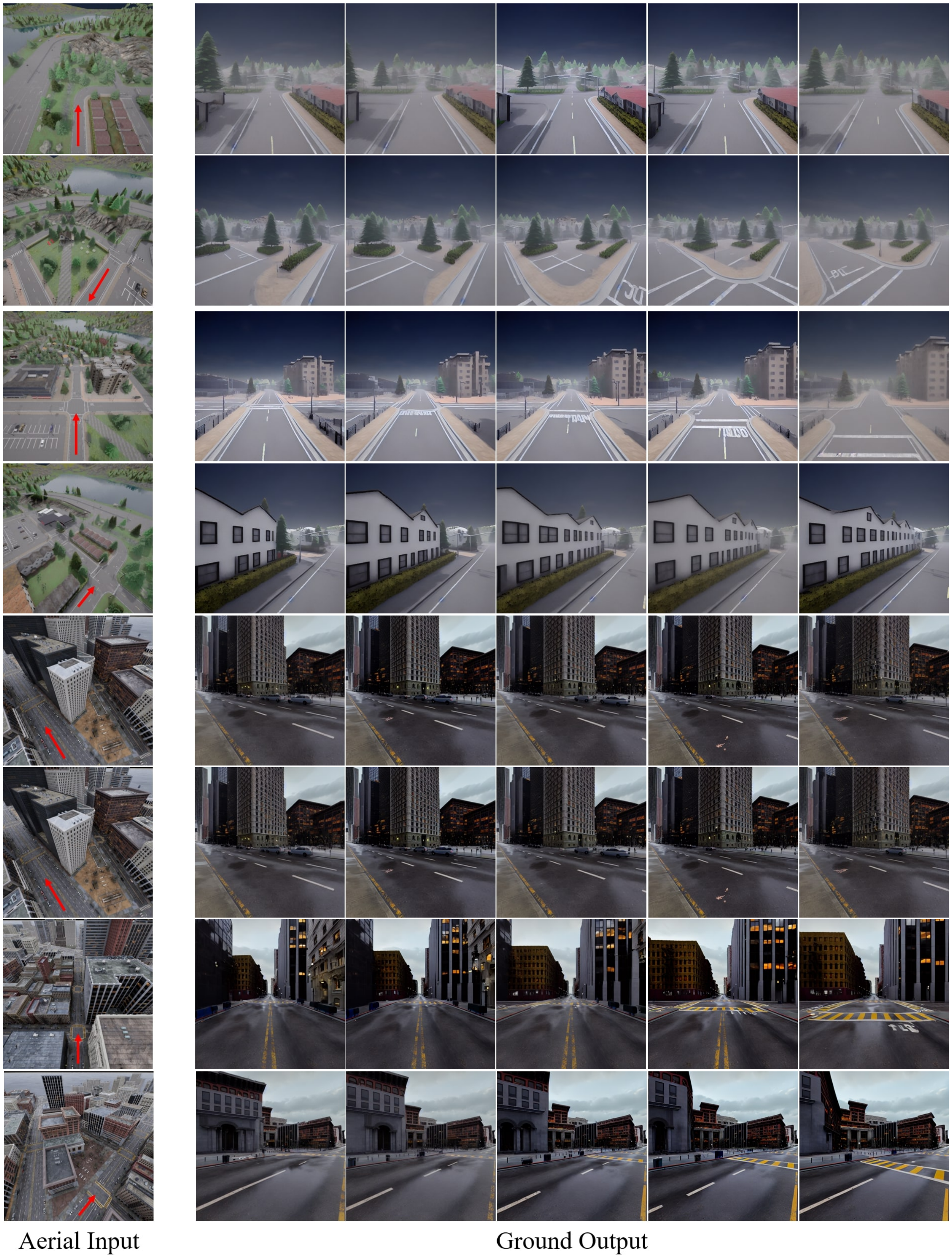


Figure 2. Additional visualizations of aerial view to ground view synthesis on CARLA and CitySample datasets

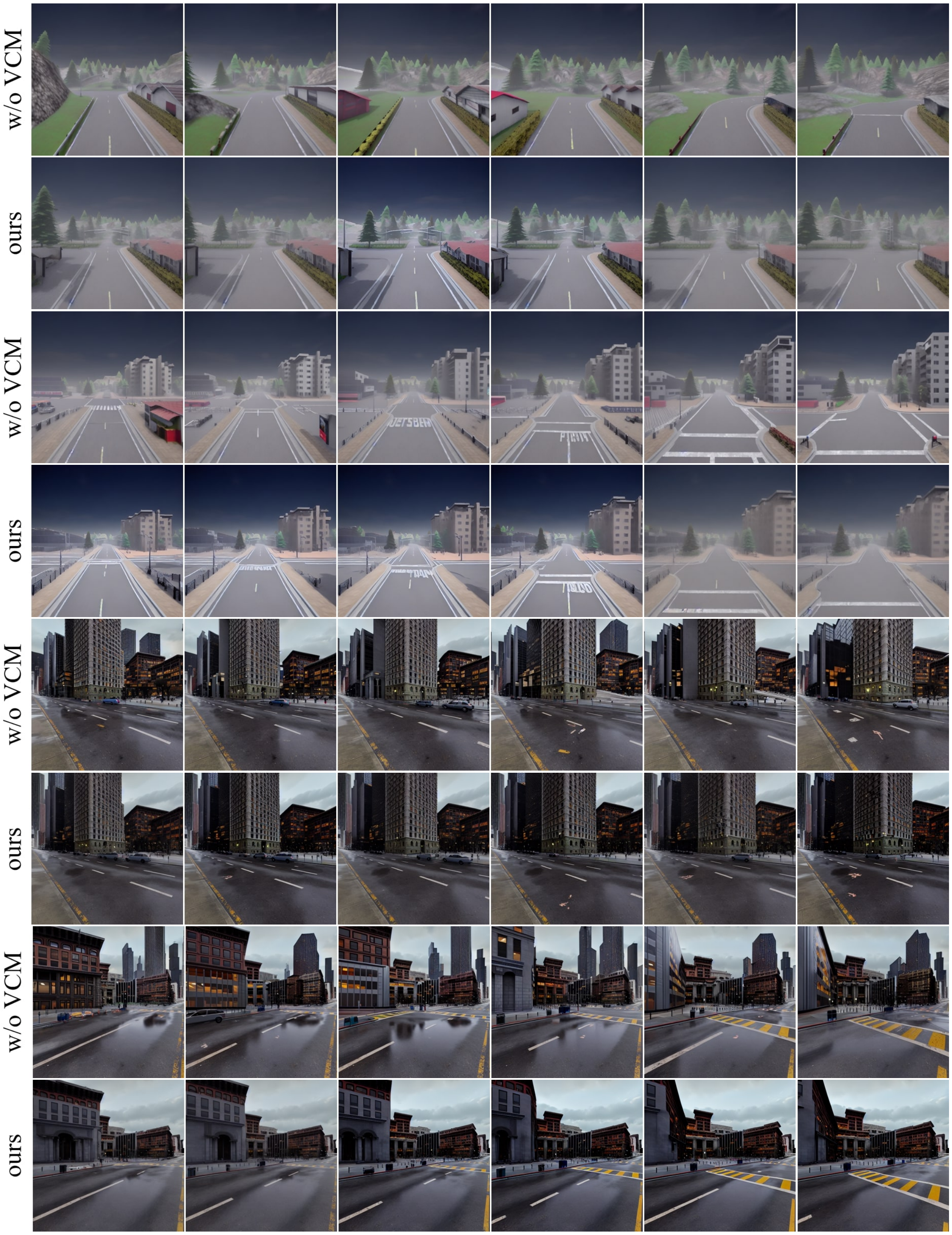


Figure 3. Additional ablation studies on view consistency module (VCM). For better visualizations, we pick 6 nearby positions along 4 different ground view sequences and display generation results for without and with VCM in every other rows.