Look Outside the Room: Synthesizing A Consistent Long-Term 3D Scene Video from A Single Image

Supplementary Material

Xuanchi Ren  
HKUST

Xiaolong Wang  
UC San Diego

A. Additional View Synthesis Results

A.1. Our Qualitative Results

Figure 2 and 3 provide additional long-term 3D scene videos synthesized by our methods. Our method is able to synthesize consistent novel views with large camera transformations while maintaining high fidelity.

A.2. Qualitative Comparison with Baselines

Figure 4 provide additional comparison with previous methods, including SynSin [10], SynSin-6x [7], GeoGPT [8] and Appearance Flow [12]. The details of the baselines are introduced in Sec. D. Our method is able to generate more consistent and clear.

A.3. Additional Visual Ablation Study

Figure 1 provides additional visual ablation study to validate the effectiveness of beam search strategy.

B. Additional Experiment

B.1. Comparison with Additional Baselines

Infinite Nature [4]. Our paper focuses on indoor scenes while Infinite Nature proposed by Liu et al. [4] focuses on nature scenes and the training code is currently not available online. Our problem is also more challenging given more structural constraints in indoor scenes. As an approximation, following the suggestion by Rockwell et al. [7], we compare to a method applying SynSin [10] in a sequential manner, namely SynSin-Sequential. We report the FID results on Matterport3D in Table 1. We achieve significant improvements on image quality.

Video Antoencoder [3]. We also compare to Video Autoencoder proposed by Lai et al. [3] on Matterport3D. As shown in Table 1, it performs worse than our method. However, it is worthy to note that Video Autoencoder does not require camera ground-truths during training, which is a more challenging setting.

<table>
<thead>
<tr>
<th>Method</th>
<th>FID ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>Video Antoencoder</td>
<td>229.68</td>
</tr>
<tr>
<td>SynSin-Sequential</td>
<td>158.31</td>
</tr>
<tr>
<td>Ours</td>
<td>57.22</td>
</tr>
</tbody>
</table>

Table 1. Comparison on Matterport3D.

B.2. Time consumption.

We measure the average time to generate a frame during inference, as shown in Table 2.

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (sec/image)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PixelSynth</td>
<td>24.71</td>
</tr>
<tr>
<td>GeoGPT</td>
<td>8.73</td>
</tr>
<tr>
<td>Ours</td>
<td>13.13</td>
</tr>
</tbody>
</table>

Table 2. Average inference time (sec/image).

C. More Implementation Details

We provide more implementation details of our method.

Transformer. We follow GPT-2 architecture [6] to implement our Transformer. We set the hidden dimension $d_e$ to 1024, set the number of attention heads to 16, and use a two-layer MLP with hidden size of 4096 inside each transformer block. For an autoregressive Transformer, we adopt the teacher-force strategy [5] with autoregressive masks during training to enable parallel computing.

VQ-GAN. We adopt the architecture and training strategy from [2] for our VQ-GAN part. And we use a downsampling factor of 16, such that an image of resolution $256 \times 256$ is encoded to $16 \times 16$ tokens.

1https://github.com/CompVis/taming-transformers
D. Details of Baselines

**SynSin** [10]. SynSin utilizes a point cloud as an intermediate geometric representation. We also consider a baseline, **SynSin-6x**, which is a version of SynSin trained on much larger view changes. However, these two baselines can only perform inpainting and can not generalize to large view changes. We adopt the official implementation\(^2\).

**PixelSynth** [7]. Based on SynSin, PixelSynth proposes to perform outpainting with the help of VQ-VAE2 and autoregressive model [9]. However, though it can perform outpainting, it still can not apply to the long-term view synthesis as our method does. For the implementation, we adopt the official one\(^3\).

**GeoGPT** [8]. GeoGPT is a geometry-free method, which models two adjacent views as a probabilistic model. However, GeoGPT can not ensure consistency and does not explore the locality constraint in the autoregressive Transformer. For the implementation, we adopt the official one\(^4\).

**Appearance Flow** [12]. Besides the baselines used in the main paper, we also compare our method with Appearance Flow, which is also a geometry-free baseline. Appearance Flow predicts a flow field that warps the original image into a novel view. However, this method can not work well on large camera changes since there are large missing areas after warping. We adopt the implementation provided by SynSin.

---

\(^2\)https://github.com/facebookresearch/synsin  
\(^3\)https://github.com/crockwell/pixelsynth  
\(^4\)https://github.com/CompVis/geometry-free-view-synthesis
Figure 2. **Long-term view synthesis** on RealEstate10K [11]. Our method is able to synthesize consistent novel views with large camera transformations while maintaining high fidelity.

Figure 3. **Long-term view synthesis** on Matterport3D [1]. Our method is able to synthesize consistent novel views with large camera transformations while maintaining high fidelity.
Figure 4. **Long-term view synthesis compared with baselines.** Previous methods are not capable of synthesizing a consistent long-term scene video. Our method can synthesize long-term views of perceptual consistency and high-fidelity.
References


