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

Xuanchi Ren
HKUST

Xiaolong Wang
UC San Diego

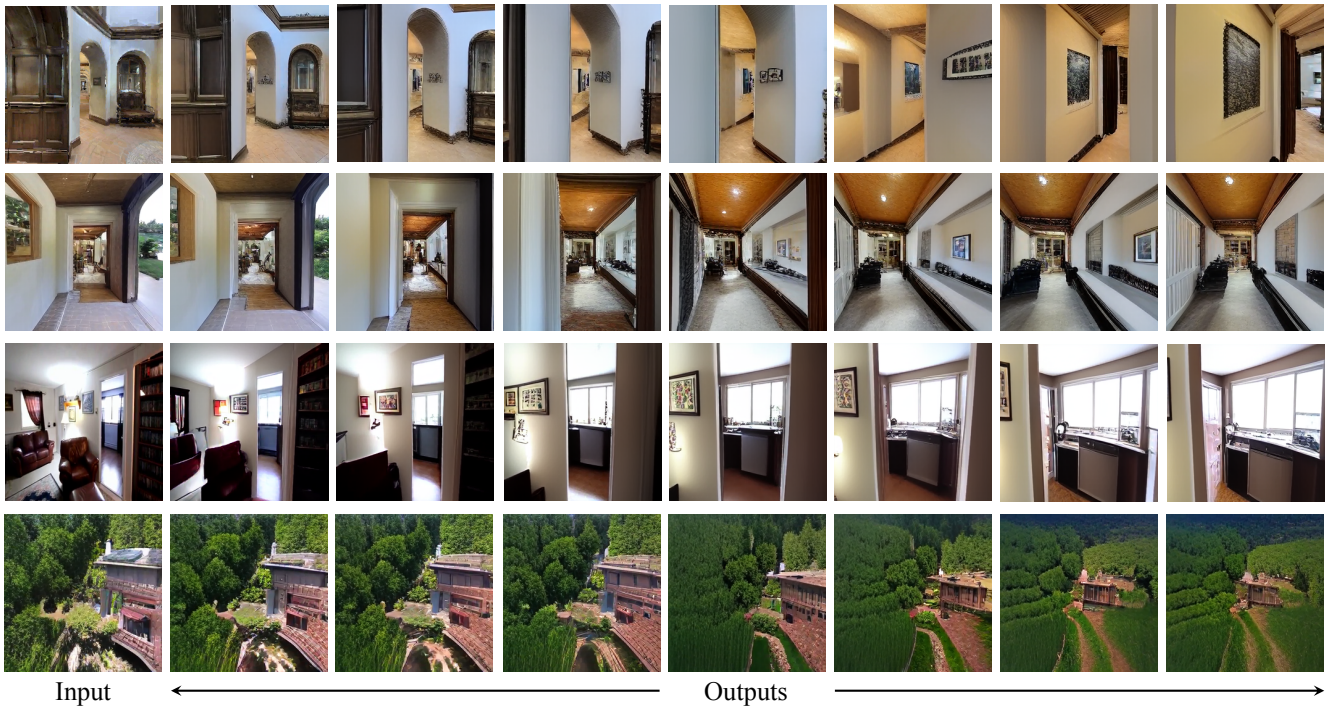


Figure 1. **"Look Outside the Room".** Using a single input image, our method is capable of generating perceptual consistent novel views for a camera trajectory. The top two rows are from the Matterport dataset, and the bottom two rows are from the RealEstate10K dataset.

Abstract

Novel view synthesis from a single image has attracted a lot of attentions recently, and it has been largely advanced by 3D deep learning and rendering techniques. However, most work is still limited by synthesizing new views within relatively small camera motions. In this paper, we propose a novel approach to synthesize a consistent long-term video given a single scene image and a trajectory of large camera motions. Our approach utilizes a autoregressive Transformer to perform sequential modeling of multiple frames, which reasons the relations between multiple frames and the corresponding cameras to predict the next frame. To facilitate learning and ensure the consistency among generated frames, we introduce a locality constraint based on the input cameras to guide self-attention among the large number of patches across space and time. Our method outperforms state-of-the-art view synthesis approaches by a large margin, especially when synthesizing long-term future in indoor 3D scenes.

1. Introduction

Single-image view synthesis has attracted a lot of attentions in computer vision and computer graphics. It brings a photo to life by extrapolating beyond the input pixels and generating new pixels following the geometric structure of the scene. At the same time, the generated pixels need to be semantically coherent with the existing pixels. Current view synthesis methods which learns 3D geometric repre-

sentation has shown encouraging results on generating high quality novel views [37, 56, 70]. However, these approaches can only generate views within a limited range of camera motion. For example, it will be very challenging for current approaches to synthesize what is outside the door of the room shown in the first row of Figure 1.

When synthesizing images with large camera view changes, we would also expect the generated images to be consistent. That is, when we are synthesizing with a path walking towards to the door in a room, we want the surroundings of the path should not change all the time and reveal a single underlying world. To this end, we propose to solve the problem extended based on view synthesis: Given a single image of the 3D scene and a *long-term* camera trajectory as inputs, synthesize a *consistent* video as the output. For example, given a single input image of a room (first row of Figure 1), we synthesize the video on walking towards the door, going through the door, and navigating into a hallway with a painting on the wall. Solving such a task not only has wide applications in content generation and editing, but also help build a differentiable simulator for model-based planning and control in robotics.

To solve this problem, we seek the help from autoregressive models [8, 38, 40, 41, 61] which have shown tremendous success on extrapolating the contents beyond the input image. For example, Rombach et al. [46] proposes to use an autoregressive Transformer to implicitly perform *large* geometric transformation for view synthesis. To handle the uncertainty with large transformation, the model is trained under a probabilistic framework which allows for sampling different novel views with the same camera. While generating realistic novel views even given a large transformation, it also leads to inconsistent and diverse outputs along a given trajectory due to the probabilistic sampling.

In this paper, to synthesize consistent long-term videos, we propose to leverage the autoregressive Transformer for sequential modeling in time with locality constraints. Instead of learning the autoregressive model between only two views of the scene [46], our work leverages the continuity in videos and perform sequential modeling with multiple video frames. Given a sequence of input images $\{x_1, x_2, ..., x_{t-1}\}$ and the cameras between every two consecutive frames $\{C_2, C_3, ..., C_{t-1}\}$ and the camera for future frame C_t. We provide a probabilistic framework to predict the future frame via sampling from $p(x_t|x_1, C_2, x_2, C_3, ..., x_{t-1}, C_t)$. By conditioning multiple frames during sampling, it ensures the consistency between generated view and historical views. When inference with our Transformer model, we can start with input one single image, and gradually increase the inputs using the predicted frames together with previous frame.

However, it is very challenging to learn such a sequential model with the autoregressive Transformer, which uses self-attention to model the large number of relations between every two patches across space and time in the input video. To facilitate training, our key insight is that not every relational pair is equally important, and we can incorporate a locality constraint to guide the model to concentrate on the critical dependencies. Such locality constraints are introduced by the cameras C_i. Intuitively, given a camera between two frames, we can roughly locate where the overlapping pixels are and where are the new pixels to synthesize. To incorporate this knowledge, we compute a bias using an MLP which takes the camera C_i as inputs, namely *Camera-Aware Bias*. We add this bias to the affinity matrix during performing the self-attention operation. In this way, each patch will have a stronger bias on depending on or attending to relevant patches connected by the camera. Empirically, we find the Camera-Aware Bias not only makes the optimization much easier, but also plays an important role in enforcing the consistency between frames during generation.

We perform our experiments on multiple datasets including the RealEstate10K [74] and Matterport3D [6], which mainly focus on 3D indoor scenes. Our model is able to synthesize new views with large camera motion, and generate a long-term video given a single image input as visualized in Figure 1. Our method not only outperforms state-of-the-art approaches on standard view synthesis metrics, but also achieves a significantly better gain when evaluating on long-range future frames. We highlight our main contributions as follows:

- A novel Transformer model on synthesizing a consistent long-term video given a single image and a trajectory as inputs.
- A novel locality constraint using camera-aware bias, which facilitates optimization during learning and enforces the consistency between generated frames.
- State-of-the-art performance in view synthesis. Our method outperforms baselines by a large margin on the long-term frames.

2. Related Work

Novel View Synthesis. View synthesis has been a long studied problem in computer vision and graphics. When synthesizing with multiple input views, 3D structural representations are often leveraged such as classical multi-view geometry [10, 12, 19, 25, 53, 75], deep voxel representations [30, 54] and neural radiance fields [37, 66]. Recently, researchers have also proposed to perform single-image view synthesis to bring a static photo to life [21, 24, 45, 55, 57, 70, 72]. For example, Wiles et al. [70] propose to perform view synthesis using 3D point clouds as intermediate representations. While these approaches work well with small camera changes, they cannot outpaint pixels that are far from the given view. To perform view synthesis with

large camera changes, Rombach et al. [46] propose a Transformer based autoregressive model. While this approach can synthesize diverse and realistic results, it cannot synthesize consistent views along a trajectory. To seek a balance, Rockwell et al. [45] propose to leverage both 3D representation and the autoregressive models to achieve consistent view synthesis in indoor scenes with large camera changes. However, they are not able to generate a long-term future outside the door of the given room like our approach does.

Video Synthesis. Learning to synthesize and predict a video provides an important manner to capture the dynamics of the world. Researchers have been studied on synthesizing videos from a random noise vector [50, 58, 63], predicting the future frames based on one or multiple previous frames [13,18,26,35,64,65], and translating one video from modality to another [5, 44, 67, 68]. However, most video synthesis approaches do not consider the underline 3D geometry of the scene when predicting the pixels. Our work is mostly related to [29], which proposes an approach to synthesize a long-term video of outdoor nature environments given a single image and a trajectory as inputs. Different from them, we focus on 3D indoor scenes which requires more structural reasoning when performing outpainting.

Image Extrapolation and Outpainting. Image outpainting [23,69,71] synthesizes pixels beyond current input images in 2D. Specifically, our work is related to the autoregressive models [36,43,51,59,61] which performs outpainting the next pixels in a sequential manner. However, learning to predict pixels one by one introduces a large complexity in training and inference. Recently, Razavi et al. [42] propose a novel representation with Vector Quantized Variational AutoEncoder (VQ-VAE), which performs autoregressive modeling in latent space instead of pixel space. This largely reduces the complexity in sequential modeling, and it enables Generative Adversarial Networks [16,28] for synthesizing high resolution images with Transformers. Our work is highly inspired these works, besides forwarding only image tokens to Transformers, we also add cameras as tokens in sequential modeling similar to [46].

Transformers. With the success of Transformer in machine translation [40, 62] and language-modeling [14], it is also recently introduce into multiple recognition tasks in computer vision [1–3, 15, 17, 31, 32]. Besides recognition, it has also been widely used together with autoregressive models for image and video generations [16, 28, 39, 46]. However, it is still very challenging to optimize the self-attention module in Transformer when modeling long sequence of visual tokens. In this paper, we propose to introduce a novel camera-aware bias as a locality constraint for better sequential modeling.

3. Method

In this section, we present our method, which models single-image scene synthesis autoregressively with the transformer architecture in the latent space, as shown in Figure 2.

3.1. Autoregressive Scene Synthesis

Given a single input image x_1 together with a sequence of desirable camera transformations $\{C_2, C_3, ..., C_T\}$, our goal is to synthesize a sequence of images $\{x_2, ..., x_T\}$ with unconstrained length, ensuring high-quality and perceptual consistency without any 3D information.

The previous method focuses on learning an autoregressive model between only two adjacent views of the scene [46]. In our task, it would be a more natural choice to take all the previous frames and cameras into account. In our method, inspired by the success of sequential modeling in reinforcement learning [7], we propose to leverage its power in scene synthesis. To synthesis x_t, we need to accumulate the likelihood of generating $\{x_i\}_{i=1}^{t}$ autoregressively. This is formally written as:

$$\begin{aligned} p(x_t|x_1, \{C_i\}_{i=2}^{T}) &= \prod_{t,i} p(x_{t,i}|x_{t,<i}, x_{<t}, \{C_i\}_{i=2}^{T}) \\ &= \prod_{t,i} p(x_{t,i}|x_{t,<i}, x_{<t}, \{C_i\}_{i=2}^{t}), \end{aligned} \tag{1}$$

where $t \in [1, T]$ indicates time step and $i \in [1, HW]$ indicates the index inside a flattened image coordinate. Based on sequential modeling, we can sample x_t from from the distribution:

$$x_t \sim p(x_t|x_1, C_2, x_2, C_3, ..., x_{t-1}, C_t). \tag{2}$$

However, different from the simple case that models only two adjacent views [29, 46], sequential modeling poses two problems: (i) Only self-attention does not ensure that the relationship between every two patches across space and time are modeled. (ii) More careful designs should be taken into account to ensure a consistent long-term synthesis. For the first problem, we propose an adaptive bias in self-attention as a 3D inductive bias (Sec. 3.3). For the second problem, we propose several key techniques for both training and inference (Sec. 3.2 & Sec. 3.4).

3.2. Network Architecture

Overview. Direct learning the distribution in Eq. 1 in an end-to-end manner is difficult because the model needs to capture interactions inside the sequence and guarantee high-quality generation at the same time. To tackle this problem, we follow previous methods [16, 45] to adopt a two-stage training. For the first stage, we pretrain a VQ-GAN mapping the images to "tokens", consisting an encoder E that

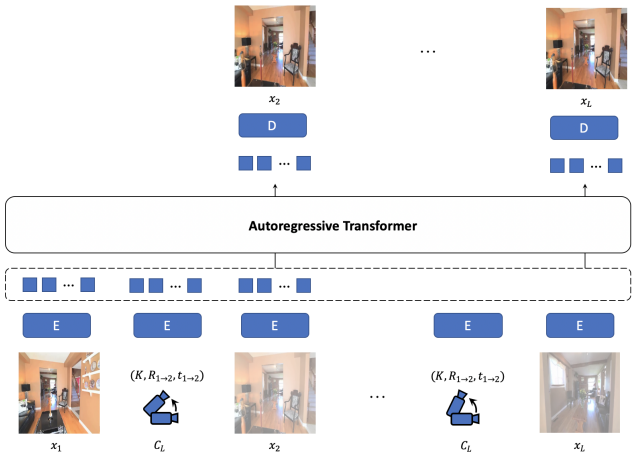


Figure 2. Overview of our method. During training, images $\{x_l\}_{l=1}^{L}$ and camera transformations $\{C_l\}_{l=2}^{L}$ are first encoded to modality-specific tokens and a decoupled positional embedding is added. Tokens are then fed into an autoregressive Transformer that predicts images. During inference, given a single image, x_1 and a camera trajectory $\{C_l\}_{l=2}^{L}$, novel views can be generated autoregressively by using the Transformer.

encode images to discrete representations, a decoder D that map the representations to high-fidelity outputs, and a codebook $\mathcal{B} = \{b_i\}_{i=1}^{|\mathcal{B}|}$ of discrete representations $b_i \in \mathbb{R}^{d_b}$. After processing the images into "tokens", we use a GPT architecture [40] with proposed locality constraint, which modifies the Transformer [62] architecture with a causal self-attention mask to enable autoregressive generation. We then introduce each of the modules used by our system in detail.

Image Encoder E. For an input sequence of images $\{x_l\}_{l=1}^{K}$, the l-th frame $x_l \in \mathbb{R}^{H\times W\times 3}$ can be converted into the latent space by the pretrained VQ-GAN encoder denoted as:

$$y_l = E(x_l), \tag{3}$$

where $y_l \in \mathbb{R}^{(hw)\times d_b}$ is the latent variable with $h \times w$ tokens. Then y_l can be quantized to get a sequence of integers $z_l \in \mathbb{R}^{hw}$ which index the learned codebook $\mathcal{B}$:

$$z_{l,k} = \arg\min_j \|y_{l,k} - b_j\|^2, \tag{4}$$

where $y_{l,k}$ is the k-th token of y_l.

Image Decoder D. Given the nearest indexes z_l, we can decode it back to high-fidelity image using the pretrained VQ-GAN deocder. z_l is first embedded by the codebook $\mathcal{B}$:

$$b_l = B[z_l], \tag{5}$$

where $b_l \in \mathbb{R}^{(hw)\times d_B}$. Then, b_l can be decoded to reconstruct the original image:

$$\hat{x} = D(b_l), \tag{6}$$

where $\hat{x} \in \mathbb{R}^{H\times W\times 3}$. In this way, we can model the Eq. 1 with discrete representation z in the latent space of the VQ-GAN. Moreover, the discrete representation is well-aligned with the "word" in NLP and thus suitable for efficiently training GPT-like architecture [16].

Camera Encoder E^C. For the camera model, we follow previous work [46, 70] to assume it as a pinhole one, such that a desired geometric transformation between two images can be determined by the intrinsic camera matrix K, a rotation matrix R and a translation matrix t.

Canonical Modeling. In our method, to improve consistency, we propose to use *canonical modeling*, such that the first image is assumed as a canonical view. Thus, the input sequence of camera transformation $\{C_l\}_{l=2}^{L}$ is relative to the canonical view, i.e., $C_l = (K, R_{1\to l}, t_{1\to l})$. We encode C_l to latent representation $C_l^e \in \mathbb{R}^{M\times d_e}$ by:

$$C_l^e = E^C(C_l), \tag{7}$$

where the camera parameters inside C_l are flattened and concatenated to shape $M \times 1$ and E^C is a linear layer.

Transformer. Given the encoded images embeddings $\{z_l\}_{l=1}^{L}$ and camera embeddings $\{C_l^e\}_{l=2}^{L}$, we use a transformer to model the conditional probability in Eq. 1 in the latent space.

Decoupled Positional Embedding. To deal with spatial-temporal relationship, we propose a *decoupled positional embedding*. The tokens of the image are first calculated with the consideration of spatial information:

$$g_l^I = \lambda(z_l) + P^I, \tag{8}$$

where $\lambda(.)$ is a embedding function that maps z_l into the latent space $\in \mathbb{R}^{hw\times d_e}$ of transformer and $P^I \in \mathbb{R}^{hw\times d_e}$ is the learnable spatial positional embedding sharing across all the images. Similarly, the tokens of the camera are calculated as:

$$g_l^C = C_l^e + P^C, \tag{9}$$

where $P^C \in \mathbb{R}^{M\times d_e}$ is the learnable camera positional embedding sharing across all the cameras transformation. Then, the input tokens to transformer are calculated as:

$$v = [g_0^I, g_1^C, ..., g_{L-1}^I, g_L^C, g_L^I] + P^T, \tag{10}$$

where $v \in \mathbb{R}^{N\times d_e}$ is the input tokens to transformer and $P^T \in \mathbb{R}^{N\times d_e}$ is the sinusoidal position embedding indicating order of tokens, modeling temporal relationship.

Now, a transformer $\mathcal{T}$ can be trained in an autoregressive way, as denoted:

$$h_n = \mathcal{T}(v_{<n}), \tag{11}$$

where h_n is the n step of output hidden states $h \in \mathbb{R}^{N\times d_e}$. Then we feed it to a linear layer to get the logits $\hat{z}_n$:

$$p(\hat{z}_n|v_{<n}) = \text{Softmax}(\text{Linear}(h_n)), \tag{12}$$

where the linear layer maps $\mathbb{R}^{d_e}$ to $\mathbb{R}^{|\mathcal{B}|}$. In fact, we only select hidden states for image prediction tokens as h^I and feed it to a linear layer to get the logits $\hat{z}^I$ of predicted image tokens, as denoted:

$$p(\hat{z}^I|x_1, \{C_i\}_{i=2}^L) = \text{Softmax}(\text{Linear}(h^I)), \quad (13)$$

where $\hat{z}^I \in \mathbb{R}^J$, $h \in \mathbb{R}^{J \times d_e}$ and $J = (L-1)hw$.

Finally, the transformer together with the camera encoder are trained using cross-entropy loss, leading to the training objective:

$$\mathcal{L} = -\frac{1}{J}\sum_{j=1}^{J} z_j^I log(p(\hat{z}^I)), \quad (14)$$

where z_j^I is index in codebook $\mathcal{B}$ of the j-th visual token.

3.3. Camera-Aware Bias in Transformer

Self-attention captures global dependency, which is a desirable property for novel view synthesis, especially for outpainting [46]. Since only self-attention and MLP are applied in the Transformer, there is a lack of inductive bias on 3D [22], especially facing thousands of tokens, including information interaction between adjacent tokens in either spatial or time, which is not confined to perceptual consistency. Thus, it is crucial to inject inductive bias on the spatio-temporal relations in Transformer.

In the convolution network, 3D convolution serves as a 3D-aware inductive bias, with the constraint on locality in both spatial and time [4]. Recently, researchers interest in injecting 2D convolution layers in Transformer to help short-range relations in spatial [11, 27]. An intuitive way to introduce 3D-aware inductive bias is to inject 3D convolutions. However, it poses two problems: (i) Convolution layers are non-trivial in an autoregressive Transformer due to the irregular kernel design [60]. (ii) The motion between two adjacent views may be so large that the overlapping pixels in geometric transformation are not in the local window. Our key insight to solve this problem is that there is a clear relationship between frames in the video, such that the correspondence between frame x_i and frame x_j is determined by relative camera transformation $(K, R_{i\to j}, t_{i\to j})$. We can incorporate such spatial-temporal dependency between pixels as a 3D-aware inductive bias in Transformer. Inspired by the exploration on relative position bias in computing affinity matrix in self-attention based on image coordinate [31], we model the observed relationship as a novel camera-aware adaptive position bias, denoted as:

$$\boldsymbol{a}_{i,j} = \boldsymbol{q}_i \boldsymbol{k}_j + \phi([K, R_{i\to j}, t_{i\to j}]), \quad (15)$$

where $q_i \in \mathbb{R}^{hw \times d_e}$ is the query corresponding to i-th frame and $k_j \in \mathbb{R}^{hw \times d_e}$ is the key corresponding to k-th frame, and $a_{i,j} \in \mathbb{R}^{hw \times hw}$ indicates the similarity between the hidden state corresponding to i-th frame and j-th frame respectively. Note that our design is applicable to causal self-attention with a mask. We empirically set $\phi : \mathbb{R}^M \longrightarrow \mathbb{R}^{hw \times hw}$ to $\gamma \text{tanh}(\text{Linear}(.))$, where γ is a scalar. For the similarity between frames and cameras, we do not apply any bias. In this way, each patch will have a stronger bias depending on relevant patches connected by the camera, which serves as a 3D-aware inductive bias.

3.4. Training and Inference Details

We then introduce several key techniques for training and inference in our method.

Overlapping Iterative Modeling. In our task, we target generating long-term 3D scene video with unconstrained length T. However, it is never possible to set the length of the training sequence L to infinity. Thus, we choose an iterative modeling strategy. Given a single image x_1, we first generate $x_2, ...x_L$ in an autoregressive manner. Then, instead of only using x_L, we aggregate information from $x_2, ...x_L$ to generate x_{L+1} and so on. This overlapping iterative modeling allows us to inference for unconstrained length and maintains perceptual consistency. As we show in Sec. 4.4, this strategy is sufficient for a consistent long-term 3D scene video even with a small L.

Error Accumulation. As pointed in [29,47], a key challenge in generating long sequences is dealing with the accumulation of errors. Even a small perturbation in each iteration can eventually lead to predictions outside the distribution and thus undesirable results. For an autoregressive Transformer, though we still need teacher forcing in training, we can partially simulate the error accumulation process during inference. We can first sample the predicted novel views from the predicted logits with the image decoder D and then finetune the model with its own predicted outputs, which improves the visual quality for long-term synthesis as shown in Sec. 4.4.

Beam Search. During inference, we need to sample next frame x_t from Eq. 1. Considering the consistency, we need to choose x_t with the most likely sequences of tokens. However, decoding the most likely output sequence is exponential in the length of the output sequence. It is intractable (NP-complete) to search thoroughly since there are thousands of possible choices ("vocabulary") [48]. We find that greedily take the most likely next step as the sequence leads to unnatural artifacts. Thus, we adopt a beam search strategy [49]. Starting with the k most likely codes in the VQ codebook as the first step in the sequence, we expand the top k possible next steps instead of all possible in original algorithms for faster speed. Then we keep the k most likely ones and repeat. In this way, we find a more optimal sample than a greedy search.

4. Experiments

In this section, we provide an empirical evaluation of our method. We demonstrate the power of our approach with an

autoregressive Transformer on the view synthesis task.

4.1. Experimental setup

Datasets. We follow the common protocol [24, 45, 70] to evaluate our method on *Matterport3D* [6] and *RealEstate10K* [74]. Matterport3D consists of 3D models of scanned and reconstructed building-scale scenes, of which 61 are for training, and 11 are for testing. To generate long-term episodes, we use an embodied agent in Habitat [52] from one point in the scene to another point. In total, we render 6000 videos for training and 500 videos for testing. RealEstate10K is a collection of videos of footages of real estates (both indoor and outdoor). We follow [24] to use $10,000$ videos for training and $5,000$ videos for testing.

Baselines. We compare our approach with three state-of-the-art single-image novel view synthesis work: *SynSin* [70], *PixelSynth*[1] [45] and *GeoGPT* [46]. SynSin and PixelSynth utilize point cloud as a geometric representation. We also adopt an improved version of SynSin, named *SynSin-6x*, provided by [45], which trained on larger view change. GeoGPT is a geometry-free method with probabilistic modeling between two adjacent views.

Implementation Details. For preprocessing, we resize all images into a resolution of $H \times W = 256 \times 256$. For our experiments on both Matterport3D and RealEstate10K, we adopted the VQ-GAN from [46] pretrained on RealEstate10K. The number of entries in the codebook $\mathcal{B}$ is 16384. For the Transformer, we adopt a GPT-like architecture [40] with a stack of 32 transformer blocks containing casual self-attention modules. During training, the training video clip consists $L = 3$ frames, which will be discussed in Sec. 4.4. The encoded image is of shape $h \times w = 16 \times 16$ and the camera embedding is of length $M = 30$, which lead the total sequence length $N = 828$. We train our transformer using a batch size of 16 for $200K$ iterations with an AdamW optimizer [33] (with $\beta_1 = 0.9$, $\beta_2 = 0.95$). We set the initial learning rate to 1.5×10^{-4} and apply a cosine-decay learning rate schedule [34] towards zero. During inference, we set $k = 3$ for beam search. We defer more details to the supplementary material.

4.2. Evaluation on Short-Term View Synthesis

We evaluate our method against the baselines on short-term view synthesis in the considered range of previous novel view synthesis methods. In this setting, we adopt the standard metrics in view synthesis task: PSNR and LPIPS [73]. PSNR measures pixel-wise differences between two images, and LPIPS measures the perceptual similarity in deep feature space. As pointed by [45], PSNR and LPIPS also measure *consistency* for a unimodal task, such as the short-term view synthesis. For both datasets, we randomly select test sequences with an input frame and 5 subsequent ground-truth frames.

[1]Current implementation of PixelSynth only support 8 discrete direction. We compare against it in Sec. 4.3 following their setting.

Method	Matterport3D		RealEstate10K	
	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑
SynSin [70]	3.53	13.92	2.55	14.77
SynSin-6x [45]	3.59	14.33	2.62	14.89
GeoGPT [46]	3.09	15.24	2.68	14.42
Ours	**2.97**	**16.06**	**2.53**	**15.60**

Table 1. Quantitative evaluation on *short-term* view synthesis.

Method	Matterport3D		RealEstate10K	
	A/B vs. Ours	FID↓	A/B vs. Ours	FID↓
SynSin [70]	82.0%	152.51	92.5 %	75.47
SynSin-6x [45]	87.0%	153.96	88.5 %	48.71
GeoGPT [46]	81.5 %	99.06	68.5 %	53.82
Ours	–	**57.22**	–	**32.88**
PixelSynth [45]	69%	146.54	63%	98.87
Ours	–	**75.96**	–	**82.51**

Table 2. Image quality and scene consistency evaluation on *long-term* view synthesis. For the A/B test, each cell lists the fraction of pairwise comparisons in which scenes synthesized by our approach were rated more consistent than scenes synthesized by the corresponding baseline.

Method	Matterport3D		RealEstate10K	
	LPIPS ↓	PSNR ↑	LPIPS ↓	PSNR ↑
SynSin [70]	3.85	13.51	3.41	12.18
SynSin - 6x [45]	3.85	**14.03**	3.42	12.28
GeoGPT [46]	3.71	11.43	3.44	10.61
Ours	**3.54**	12.89	**3.20**	**12.36**

Table 3. Quantitative evaluation on *long-term* view synthesis. Though PSNR and LPIPS are poor metrics for extrapolation tasks [23, 45], we report them for reference.

Table 1 shows the quantitative results for our method. Without an intermediate geometry, our method can still outperform the methods with explicit geometric modeling in terms of short-term view synthesis. Moreover, our method also outperforms the geometry-free baseline, GeoGPT, by a large margin since this method does not ensure consistency.

4.3. Evaluation on Long-Term View Synthes

We then evaluate our method on the *long-term* view synthesis task. Prior work [23, 45] points out that PSNR and LPIPS are poor metrics for scene extrapolation tasks cause there are multiple possibilities for the output. Thus, for image quality, we follow [29] to use FID [20], which is a distribution-level similarity measurement between generated images and real images. For consistency, we follow [45] to conduct user study on Amazon Mechanical Turk following the A/B test protocol [9]. During the user study, each

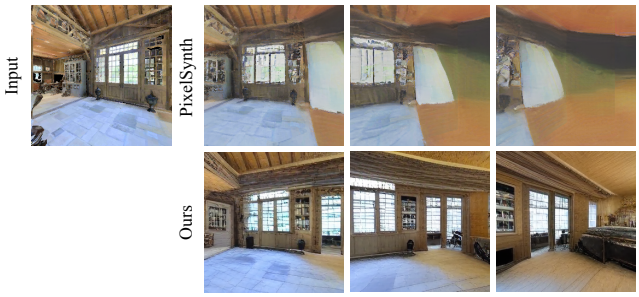


Figure 3. Qualitative comparison between our method and PixelSynth [45]. Though PixelSynth conducts outpainting explicitly, it is not capable of synthesizing a long-range view. Sequential modeling instead creates realistic and consistent views since it facilitates outpainting conditioned on pixels from previous frames.

user is presented with a video generated by our method and a baseline simultaneously. Then the user needs to choose a more consistent video. For both datasets, we randomly select test sequences with an input frame and 20 subsequent ground-truth frames with significant camera motion, of which each covers an extended range of footage. For the comparison with PixelSynth [45], we randomly sample an input frame and several outpainting directions to form a test sequence since it currently only supports 10 discrete directions.

We report the quantitative comparisons in Table 2. For both image quality and consistency, our method is significantly better than other baselines, including geometry-based and geometry-free ones. This is consistent with qualitative results, as shown in Figure 3 and Figure 5. On Matterport3D, the gap is even more prominent due to the camera angle changes being more significant. Notably, methods without geometric modeling achieve better image quality on long-term view synthesis. SynSin-6x performance is still not good, indicating that training previous methods on larger camera changes helps but does not account for the main issue.

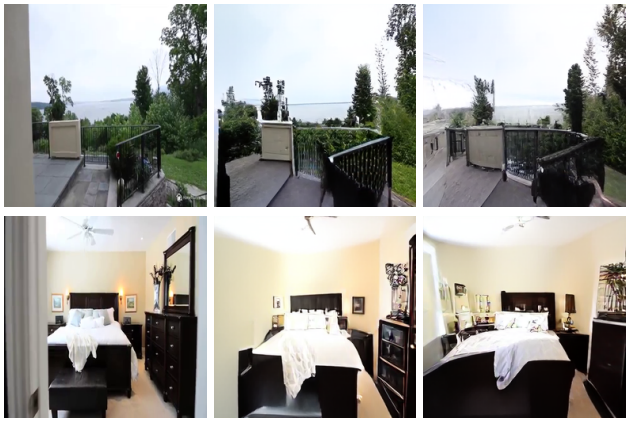
In addition, we follow past work to report PSNR and LPIPS in Table 3, which are poor measures for extrapolation tasks [23, 45]. For example, though SynSin-6x usually produces entirely gray results, its PSNR is good.

4.4. Ablation Study

We conduct an ablation study on our method in terms of long-term view synthesis on the Matterport3D dataset.

Camera-aware bias. As shown in Figure 4, the camera-aware bias improves the image quality and the consistency between frames. Table 4 also confirms this observation, indicating that bringing locality into autoregressive Transformer is critical, especially for consistency.

Decoupled positional embedding. We compare our decoupled positional embedding (P.E.) with a vanilla learnable positional embedding. As shown in Table 4, both image quality and consistency drop.



Input Ours Ours w/o Bias

Figure 4. Visual ablation study. The proposed camera-aware bias benefits both the consistency between frames and image quality.

Method	A/B vs. Ours	FID↓
Ours (Full Model)	–	**57.22**
– Decoupled P.E.	65.0%	70.47
– Camera-Aware Bias	73.8%	66.42
– Error Accumulation	56.3%	66.81

Table 4. Ablation study on Matterport3D in terms of *long-term* view synthesis. We ablate aspects of our model to investigate their influence on the results.

	L=2	L=3	L=4	L=5
FID↓	70.62	57.22	62.34	93.85
A/B vs. Ours	97.0%	–	54.0%	49.0%

Table 5. Ablation study on length of video clips L.

Error accumulation. As shown in Table 4, finetuning the model by stimulating error accumulation benefits the long-term view synthesis.

Length of video clips. We compare our default length of video clips with variants that modify the length during training. As shown in Table 5, the consistency improves significantly when the length increase from 2 to 3. When the length further increases to 5, the consistency remains nearly unchanged. For the image quality, there is a significant drop when we expand the length to 5. We hypothesize that the numbers of tokens are too large that the Transformer is difficult to optimize. Considering the computation resource and performance, we set the length of video clips to 3.

5. Discussion

In this paper, we propose an autoregressive Transformer based model to solve novel view synthesis, especially when synthesizing long-term future in indoor 3D scenes. This method leverages a locality constraint based on the input

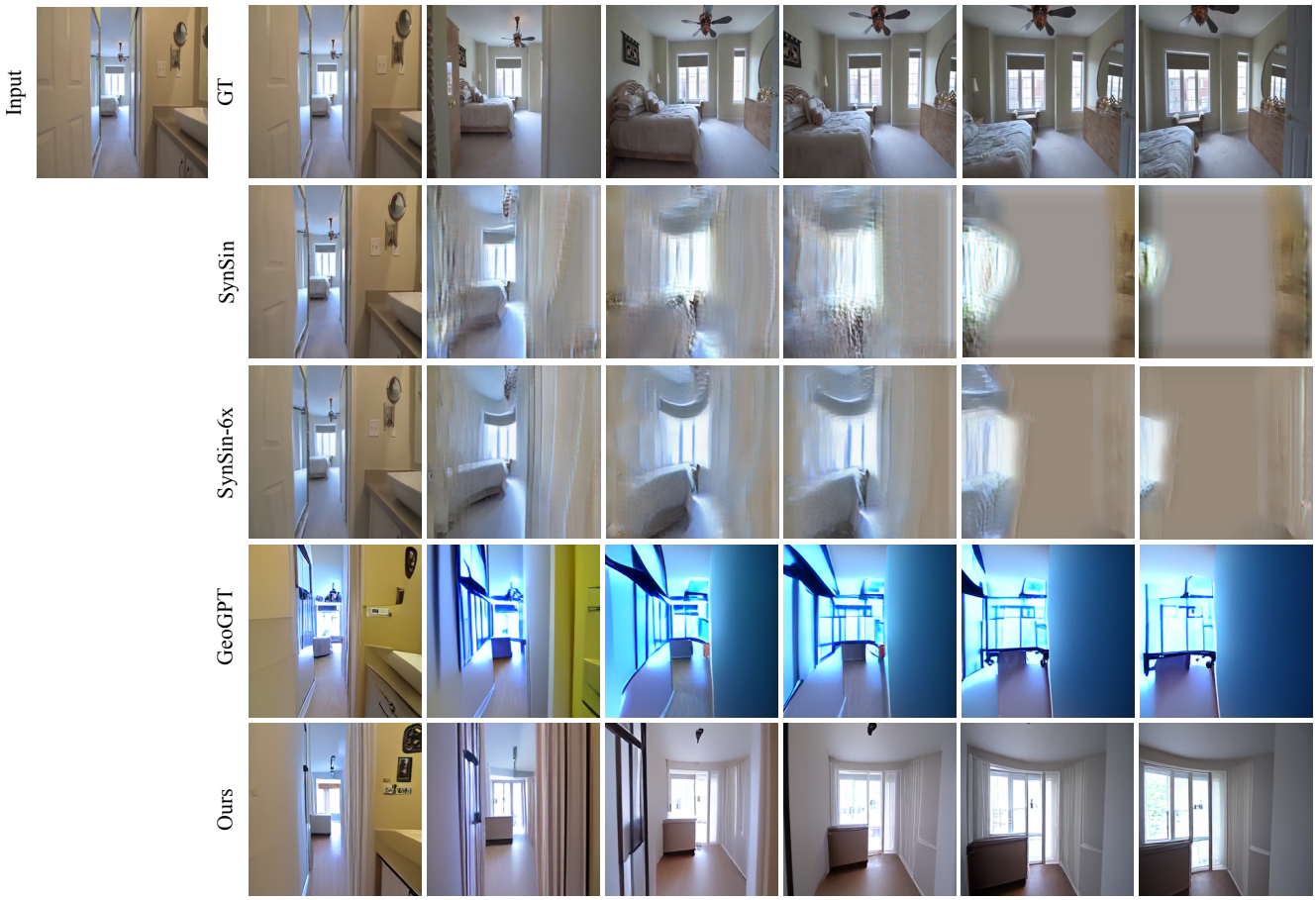


Figure 5. Qualitative comparison between our method and baselines in terms of *long-term* view synthesis. Prior work is not capable of synthesizing a consistent long-term scene video. Though our result is not the same as ground truth, it is perceptual consistency and of high-fidelity. For more results, please refer to supplementary materials.

cameras in self-attention to ensure consistency among generated frames. Our method can get superior performance in novel view synthesis compared to both geometry-based and geometry-free approaches. To conclude, we take a further step to explore the capabilities of geometry-free methods and manage to synthesize consistent high-fidelity 3D scenes. Nevertheless, many challenges remain.

First, though we improve the consistency significantly compared to previous methods when synthesizing long trajectories, there are still inconsistent for small viewpoint changes, as shown in Figure 4, which are mainly brought by compression of VQ-GAN [46]. However, decreasing the compression rate will alleviate this problem but bring higher computational cost and more token numbers for Transformer. Second, the current inference speed of the autoregressive model is slower than vanilla models. Further advancements in autoregressive model architecture still call for need. Thirdly, introducing more 3D-aware inductive bias in the geometry-free method is a faithful direction.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691*, 2021. 3

[2] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021. 3

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3

[4] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, 2017. 5

[5] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A. Efros. Everybody dance now. In *ICCV*, 2019. 3

[6] Angel X. Chang, Angela Dai, Thomas A. Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from RGB-D data in indoor environments. In *3DV*, 2017. 2, 6

[7] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *arXiv preprint arXiv:2106.01345*, 2021. 3

[8] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020. 2

[9] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017. 6

[10] Shenchang Eric Chen and Lance Williams. View interpolation for image synthesis. In *SIGGRAPH*, 1993. 2

[11] Stéphane d'Ascoli, Hugo Touvron, Matthew L. Leavitt, Ari S. Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In Marina Meila and Tong Zhang, editors, *ICML*, 2021. 5

[12] Paul E. Debevec, Camillo J. Taylor, and Jitendra Malik. Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In *SIGGRAPH*, 1996. 2

[13] Emily L. Denton and Vighnesh Birodkar. Unsupervised learning of disentangled representations from video. In *NeurIPS*, 2017. 3

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 3

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[16] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. 3, 4

[17] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. *arXiv preprint arXiv:2104.11227*, 2021. 3

[18] Chelsea Finn, Ian J. Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *NeurIPS*, 2016. 3

[19] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. In *SIGGRAPH*, 1996. 2

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurPIS*, 2017. 6

[21] Ronghang Hu, Nikhila Ravi, Alexander C Berg, and Deepak Pathak. Worldsheet: Wrapping the world in a 3d sheet for view synthesis from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12528–12537, 2021. 2

[22] Manjin Kim, Heeseung Kwon, Chunyu Wang, Suha Kwak, and Minsu Cho. Relational self-attention: What's missing in attention for video understanding. In *NeurIPS*, 2021. 5

[23] Dilip Krishnan, Piotr Teterwak, Aaron Sarna, Aaron Maschinot, Ce Liu, David Belanger, and William T. Freeman. Boundless: Generative adversarial networks for image extension. In *ICCV*, 2019. 3, 6, 7

[24] Zihang Lai, Sifei Liu, Alexei A. Efros, and Xiaolong Wang. Video autoencoder: self-supervised disentanglement of static 3d structure and motion. In *ICCV*, 2021. 2, 6

[25] Marc Levoy and Pat Hanrahan. Light field rendering. In *SIGGRAPH*, 1996. 2

[26] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Flow-grounded spatial-temporal video prediction from still images. In *ECCV*, 2018. 3

[27] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. In *ICCV*, 2021. 5

[28] Chieh Hubert Lin, Hsin-Ying Lee, Yen-Chi Cheng, Sergey Tulyakov, and Ming-Hsuan Yang. Infinitygan: Towards infinite-resolution image synthesis. *arXiv preprint arXiv:2104.03963*, 2021. 3

[29] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *ICCV*, 2021. 3, 5, 6

[30] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. In *NeurIPS*, 2020. 2

[31] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3, 5

[32] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *NeurIPS*, 2020. 3

[33] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *ICLR*, 2017. 6

[34] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[35] Michaël Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016. 3

[36] Jacob Menick and Nal Kalchbrenner. Generating high fidelity images with subscale pixel networks and multidimensional upscaling. In *ICLR*, 2019. 3

[37] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 2

[38] Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders. *arXiv preprint arXiv:1606.05328*, 2016. 2

[39] Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. ATISS: autoregressive transformers for indoor scene synthesis. In *NeurIPS*, 2021. 3

[40] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018. 2, 3, 4, 6

[41] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. In *Advances in neural information processing systems*, pages 14866–14876, 2019. 2

[42] Ali Razavi, Aäron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *NeurIPS*, 2019. 3

[43] Scott E. Reed, Aäron van den Oord, Nal Kalchbrenner, Sergio Gomez Colmenarejo, Ziyu Wang, Yutian Chen, Dan Belov, and Nando de Freitas. Parallel multiscale autoregressive density estimation. In *ICML*, 2017. 3

[44] Xuanchi Ren, Haoran Li, Zijian Huang, and Qifeng Chen. Self-supervised dance video synthesis conditioned on music. In *ACM MM*, 2020. 3

[45] Chris Rockwell, David F. Fouhey, and Justin Johnson. Pixelsynth: Generating a 3d-consistent experience from a single image. In *ICCV*, 2021. 2, 3, 6, 7

[46] Robin Rombach, Patrick Esser, and Björn Ommer. Geometry-free view synthesis: Transformers and no 3d priors. In *ICCV*, 2021. 2, 3, 4, 5, 6, 8

[47] Stéphane Ross, Geoffrey J. Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, 2011. 5

[48] Kimmo Rossi. Handbook of natural language processing and machine translation. *Mach. Transl.*, 2013. 5

[49] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach (4th Edition)*. Pearson, 2020. 5

[50] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017. 3

[51] Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P. Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. In *ICLR*, 2017. 3

[52] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *ICCV*, 2019. 6

[53] Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006. 2

[54] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhöfer. Deepvoxels: Learning persistent 3d feature embeddings. In *CVPR*, 2019. 2

[55] Jie Song, Xu Chen, and Otmar Hilliges. Monocular neural image based rendering with continuous view control. In *ICCV*, 2019. 2

[56] Richard Tucker and Noah Snavely. Single-view view synthesis with multiplane images. In *CVPR*, 2020. 2

[57] Shubham Tulsiani, Richard Tucker, and Noah Snavely. Layer-structured 3d scene inference via view synthesis. In *ECCV*, 2018. 2

[58] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *CVPR*, 2018. 3

[59] Aäron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Koray Kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In *NeurIPS*, 2016. 3

[60] Aäron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Koray Kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In *NeurIPS*, 2016. 5

[61] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *ICML*, 2016. 2, 3

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3, 4

[63] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NeurIPS*, 2016. 3

[64] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *ECCV*, 2016. 3

[65] Jacob Walker, Ali Razavi, and Aäron van den Oord. Predicting video with vqvae. *arXiv preprint arXiv:2103.01950*, 2021. 3

[66] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P. Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas A. Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *CVPR*, 2021. 2

[67] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Bryan Catanzaro, and Jan Kautz. Few-shot video-to-video synthesis. In *NeurIPS*, 2019. 3

[68] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Nikolai Yakovenko, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *NeurIPS*, 2018. 3

[69] Yi Wang, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Wide-context semantic image extrapolation. In *CVPR*, 2019. 3

[70] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *CVPR*, 2020. 2, 4, 6

[71] Zongxin Yang, Jian Dong, Ping Liu, Yi Yang, and Shuicheng Yan. Very long natural scenery image prediction by outpainting. In *ICCV*, 2019. 3

[72] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *CVPR*, 2021. 2

[73] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 6

[74] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *ACM Trans. Graph.*, 2018. 2, 6

[75] C. Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon A. J. Winder, and Richard Szeliski. High-quality video view interpolation using a layered representation. *ACM Trans. Graph.*, 2004. 2