Deep Space-Time Prior for Novel View Synthesis

Zain Shah
Vidi Labs
http://vidi.cam
hello@vidi.cam

Abstract

In this work we frame total scene capture as a spatio-temporal novel view synthesis problem. From a monocular smartphone camera video where camera poses are inferred from visual inertial odometry, our task is to construct photo-realistic imagery for new poses and timestamps that were not originally captured, but are cohesive with the overall scene. To this end, we propose that a space-time convolutional neural network could act as a strong prior to learn plausible scene reconstruction. For each scene we fit a 4D space-time tensor with learned projection to directly generate any image captured from the scene given the camera's pose and timestamp. While this is a computationally expensive fixed cost per scene, at test time the space-time volume has already been fitted, so we simply run a single forward pass through the network to generate the necessary imagery. [1]

1. Introduction

Our goal is total scene capture - to be able to share with high fidelity the complete dynamic, spatial and visual experience of any scene - including the waves lapping on the shore at the beach, and the opalescent seashells at ones' feet. Specifically, we define a visual scene as all possible images that could be captured within a given space-time volume. While this may sound impossible in theory, in practice it is straightforward to capture sparse samples of an appropriately bounded scene by e.g. filming a monocular video while exploring the scene.

Unfortunately densely capturing the complete scene the same way is impossible because the camera can only look in one direction at a given time. Even a spherical camera can only completely capture a small space-time volume. That is, because it cannot capture all displacements of the camera at all times, but it can capture all orientations of the camera in a given position.

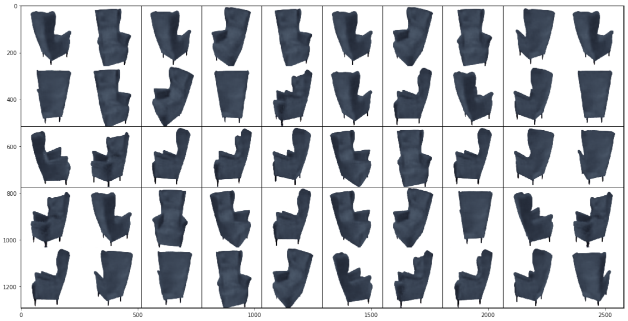

Figure 1. Novel view synthesis samples from our model on the synthetic armchairs dataset. This model was trained in 41 minutes on a single NVIDIA K80 GPU.

With the recent advances in deep generative modeling, specifically with generative adversarial networks, synthetic imagery has received renewed attention and substantial progress. We are especially inspired by the recent success in utilizing deep convolutional neural networks as an effective prior for a variety of image tasks without supervision, finding embeddings for 3D objects, rendering novel views of scenes only from imagery found "in the wild", and high fidelity consistent prediction of imagery from synthetic scenes. Most of these works, however, either do not scale to natural scenery or do not adequately account for dynamism. To that end, we propose our deep space-time prior - a scene specific learned 4D tensor combined with learned projection and up-sampling units.

Our developments build on a rich history of previous work in capturing and understanding the physical world, across disciplines such as novel view synthesis, 3D representation learning, generative modeling, inverse graphics, image based rendering, and photogrammetry. Most recently, we extend on the work of [8] in finding a scene specific representation for novel view synthesis. Their method produces high quality renderings and cohesive novel view synthesis, but does not lend itself well to dynamic scenes with motion. They employ a GRU to update a scene specific

[1] Supplementary animations available at this url.

embedding for the object in latent voxel space, then use that latent voxel space to decode. In our work we similarly learn a scene specific embedding, but instead of requiring part of the model to learn an encoding mechanism, we fit the input space-time tensor with gradient descent. This is similar to the approach in [7], where they learn a constant 4D tensor at train time, which they manipulate with Adaptive Instance Normalization to fit the desired parameters for a particular object. [9] attempt to learn a partial graphics pipeline, applying neural networks to the task of rendering from a coarse proxy geometry. Our approach is similar, except that our proxy geometry contains a temporal dimension, and is learned end-to-end rather than generated externally.

Meanwhile, [6] attacks the same problem, namely that of ”total scene capture” - but because their approach also relies on a proxy geometry input, it lends itself especially well to particular sets of dynamics, namely the dynamic appearance of a static scene under different conditions. To mitigate this, they use a semantic segmentation mask as input, so the network can learn to ignore known dynamic objects such as people, cars, etc. In contrast, our approach seeks to render the scene with all dynamic objects present, but we do not consider such a wide variety of ambient scene appearances.

Looking further back, the rich field of image-based rendering contains numerous attempts to completely capture the necessary information to reconstruct a dynamic scene. [4] uses structure from motion to first reconstruct a proxy geometry, then uses image-based rendering techniques to re-render the input video from an optimized smooth camera trajectory. Meanwhile, works like [2] use spatio-temporal regularization and optimization techniques to find the optimally consistent pixel from the input videos for each output pixel. In contrast, we do not explicitly constrain our results with any regularization, nor require images to be selected from the input video.

Earlier still are the foundational works in image based rendering for object and scene capture, the [1] and [5]. In these techniques enough data is collected to form a reasonable posterior estimate of the scene. In [1], by collecting sufficient samples of the lightfield entering the volume containing an object, one can then reconstruct the dense lightfield and sample arbitrary viewpoints. This requires a sufficiently dense sampling of the lightfield, and it assumes the object is static, with static lighting. [5] uses dense sampling of the objects’ viewpoints to form a posterior estimate of the object’s voxel occupancy grid. This also assumes the scene is static, and requires dense sampling of viewpoints in order to render realistic imagery.

2. Method

Our core contributions are two-fold:

1. We frame the bounded spatio-temporal scene capture problem as novel view synthesis
2. To enable real-time inference on commodity hardware, we devise to fit a single high dimensional space-time tensor per scene.

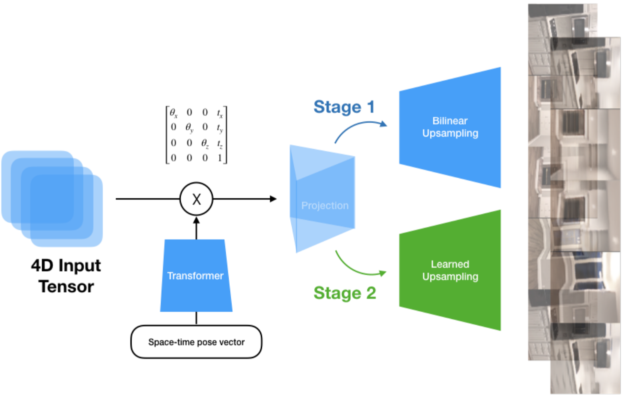
$\begin{bmatrix} \theta_x & 0 & 0 & t_x \\ 0 & \theta_y & 0 & t_y \\ 0 & 0 & \theta_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$
X
4D Input Tensor
Transformer
Space-time pose vector
Projection
Stage 1
Bilinear Upsampling
Stage 2
Learned Upsampling

Figure 2. Overview of the proposed architecture - including the learned 4D input tensor, transformer, and learned upsampling layer. Blue indicates components trained during Stage 1 of training, Green indicates components trained during Stage 2.

Consider the total set of video frames possible to be captured from a given scene, $S \in \mathbb{R}^{T \times X \times Y \times Z \times \phi \times \theta \times \psi}$ where $T, X, Y, Z, \phi, \theta, \psi$ denote the time, position and orientation of the camera. Let s_t be the pose for a frame at time t, with associated extrinsic camera parameters $x, y, z, \phi, \theta, \psi$.

This leads us to define the local camera transfer function $\hat{I_{s_t}} = F(T, X, Y, Z, \phi, \theta, \psi)$. We seek to find a transfer function such that we minimize the loss:

$$E(\hat{I_{s_t}}, I_{s_t}) = ||\hat{I_{s_t}} - I_{s_t}|| \qquad (1)$$

over the full set of captured frames S.

2.1. Our Model

Our model is straightforward. As in [7] we fit a constant 4D tensor of size 32 × 32 × 32 × 32. This 4D tensor is then projected to a canonical view volume as in [3] by a fully-connected multilayer perceptron. This multilayer perceptron generates a transform matrix from the input camera pose and timestamp, which is then applied via trilinear sampling to the learned input tensor. With this transformation, the 4D spacetime volume is now in a canonical perspective for rendering. Once transformed, the view is processed by a learned projection unit, consisting of several 1x1 convolutions to collapse the temporal and spatial dimensions into a 2D image. In our experiments, training was stable in all cases, but we managed to get higher fidelity results in less time with component-wise training.

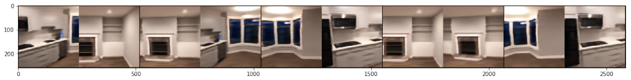

Figure 3. Novel view synthesis samples from our model trained on real imagery captured from a smartphone. This model was trained in 1 hour 35 minutes on a single NVIDIA K80 GPU.

2.2. Component-Wise Training

In component-wise training, we proceed to isolate and train the 4D spatio-temporal tensor separately from the projection units. Specifically, we start by learning the tensor, and using only the collapsing 1x1 convolutions and bilinear upsampling to get a view appropriate for our loss. Once this has converged, we switch to fractionally-strided convolutions to learn upsampling filters to add further textures to the image and increase the visual fidelity.

Thus our training regime breaks down into 2 steps:

1. First we train only the 4D input tensor and transformer. We've found that by isolating this step in training our model is less likely to overfit and more likely to generate geometrically plausible transitions between frames throughout training. In order to compare the transformed input tensor directly to the expected output images, we also train a projection unit of 1x1 convolutions and use bilinear sampling to upscale the resulting projection to the desired resolution.

2. Second, once the loss stops improving, we freeze the 4D input tensor, transformer, and projection layers. We then swap out the bilinear upsampling for a series of transposed convolutions to learn upsampling filters from the data. We expect that because the input tensor and its rotations are already parametrized, it's easier for the model to learn to apply realistic textures to the resulting projection than it is to overfit to the other cues of the camera pose.

2.3. Experiments

Our experiments are in progress but early qualitative results are quite promising. In a fraction of the time (minutes versus hours or days) required to fit DeepVoxels or HoloGAN, our model can photorealistically reproduce novel views at different timestamps and poses. We show some samples from our model in Figures 2 and 3. The armchair samples in 2 are a synthetic dataset used in [8], while the apartment footage samples used in 3 were filmed from an iPhone X with a custom application to record real-time visual inertial odometry for camera extrinsics per frame.

For both experiments, we used component-wise pretraining as defined in 2.2. Specifically, we first fit the 4D input tensor jointly with a transformation unit, a projection unit, and a static bi-linear up-sampling layer. Once this has converged, we freeze the input tensor and transformation unit, and switch to learning a series of fractionally-strided convolutions to up-sample the projected image to a higher visual fidelity. Our final model, with just the projection and up-sampling unit requires only 3MB for the forward pass. This can enable real-time exploration of the rendered scenes on a commodity smartphone.

3. Conclusion

In summary, we demonstrate that novel view synthesis approaches bring us closer to sharing the full visual experience of dynamic scenes. We propose a specific implementation that works well in practice, and demonstrate compelling early results. There are some challenges with fitting the scene specific 4D tensor, namely that the space-time voxel grid does not scale well to larger or more dynamic scenes, and there is no interpretable or explicit geometry in the model. At the same time, we propose component-wise pretraining to mitigate some of these shortcomings, and show some compelling results that will require further investigation. Future work could investigate more compressed representations of the scene, alternative capture methodologies, or techniques that allow the learned components to generalize across scenes.

References

[1] Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. The lumigraph. 1996. 2

[2] Mingming He, Jing Liao, Pedro V. Sander, and Hugues Hoppe. Gigapixel panorama video loops. *ACM Trans. Graph.*, 37(1):3:1–3:15, Nov. 2017. 2

[3] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. pages 2017–2025, 2015. 2

[4] Johannes Kopf, Michael F Cohen, and Richard Szeliski. First-person hyper-lapse videos. *ACM Transactions on Graphics (TOG)*, 33(4):78, 2014. 2

[5] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International journal of computer vision*, 38(3):199–218, 2000. 2

[6] Moustafa Meshry, Dan B. Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. Neural rerendering in the wild. *CoRR*, abs/1904.04290, 2019. 2

[7] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. *CoRR*, abs/1904.01326, 2019. 2

[8] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhöfer. Deepvoxels: Learning persistent 3d feature embeddings. *CoRR*, abs/1812.01024, 2018. 1, 3

[9] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *arXiv preprint arXiv:1904.12356*, 2019. 2