3D Neural Scene Representations for Visuomotor Control

Yunzhu Li*, Shuang Li*, Vincent Sitzmann, Pulkit Agrawal, Antonio Torralba
MIT CSAIL
{liyunzhu, lishuang, sitzmann, pulkitag, torralba}@mit.edu

Abstract

Humans have a strong intuitive understanding of the 3D environment around us. The mental model of the physics in our brain applies to objects of different materials and enables us to perform a wide range of manipulation tasks that are far beyond the reach of current robots. In this work, we desire to learn models for dynamic 3D scenes purely from 2D visual observations. Our model combines Neural Radiance Fields (NeRF) and time contrastive learning with an autoencoding framework, which learns viewpoint-invariant 3D-aware scene representations. We show that a dynamics model, constructed over the learned representation space, enables visuomotor control for challenging manipulation tasks involving both rigid bodies and fluids, where the target is specified in a viewpoint different from what the robot operates on. When coupled with an auto-decoding framework, it can even support goal specification from camera viewpoints that are *outside the training distribution*. We further demonstrate the richness of the learned 3D dynamics model by performing future prediction and novel view synthesis. Finally, we provide detailed ablation studies regarding different system designs and qualitative analysis of the learned representations.

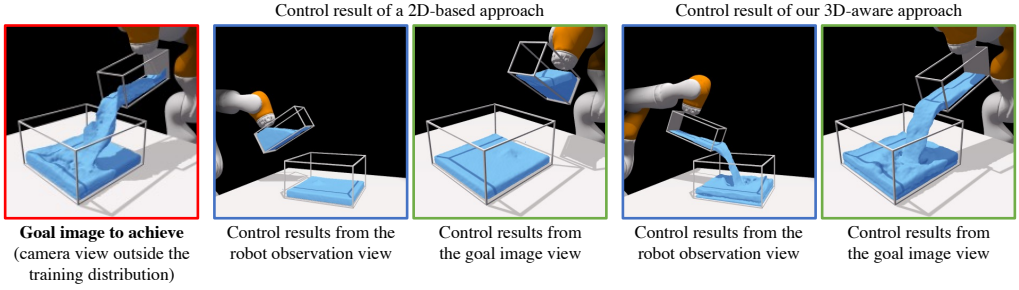


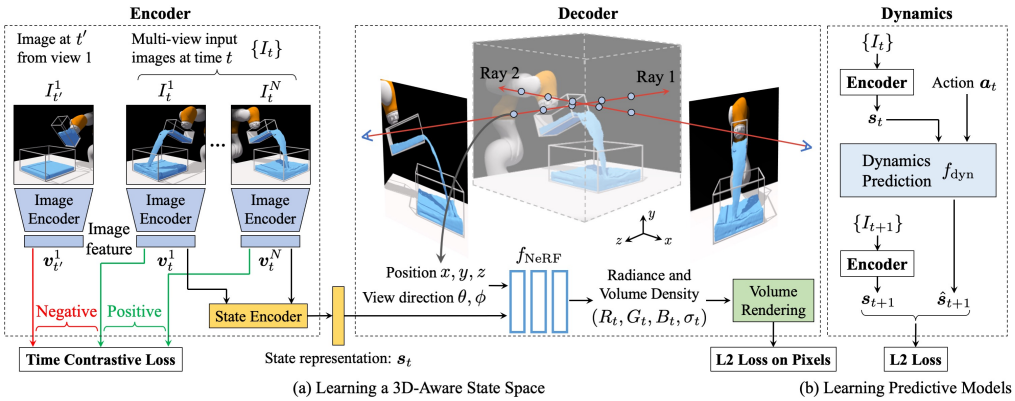
Figure 1: **Comparison of the control results between a 2D-based baseline and our 3D-aware approach.** The task here is to achieve the configuration shown on the left, observed from a viewpoint that is outside the training distribution. The agent only takes a single-view visual observation as input (images with blue frames) from a viewpoint that is vastly different from the goal. Our method generalizes well in this scenario and outperforms the 2D-based baseline, demonstrating the benefits of the learned 3D-aware scene representations.

1 Introduction

Existing state-of-the-art model-based systems operating from vision treat images as 2D grids of pixels. The world, however, is three-dimensional. Modeling the environment from 3D enables amodal completion and allows the agents to operate from different views. Therefore, it is desirable to obtain good 3D-aware representations of the environment from 2D observations to achieve better task performance when an accurate inference of 3D information is essential, which can further make it easier to specify tasks and learn from third-person videos, etc.

*equal contribution. Project Page: https://3d-representation-learning.github.io/nerf-dy/

Physical Reasoning and Inductive Biases for the Real World at NeurIPS 2021.



(a) Learning a 3D-Aware State Space (b) Learning Predictive Models

Figure 2: **Overview of the training procedure. Left:** an encoder that maps the input images into a latent scene representation. The images are first sent to an image encoder to generate the image feature representations $\boldsymbol{v}$. Then we combine the image features from the same time step using a state encoder to obtain the state representation $\boldsymbol{s}_t$. A time contrastive loss is applied to enable our model to be invariant to camera viewpoints. **Middle:** a decoder that takes the scene representation as input and generates the visual observation conditioned on a given viewpoint. We use an L2 loss to ensure the reconstructed image to be similar to the ground truth image. **Right:** a dynamics model that predicts the future scene representations $\hat{\boldsymbol{s}}_{t+1}$ by taking in the current representation $\boldsymbol{s}_t$ and action $\boldsymbol{a}_t$. We use an L2 loss to enforce the predicted latent representation to be similar to the scene representation $\boldsymbol{s}_{t+1}$ extracted from the true visual observation I_{t+1}.

In this work, we aim to leverage recently proposed 3D-structure-aware implicit neural scene representations for visuomotor control tasks. We thus propose to embed neural radiance fields [1] in an auto-encoder framework, enabling tractable inference of the 3D-structure-aware scene state for dynamic environments. By additionally enforcing a time contrastive loss on the estimated states, we ensure that the learned state representations are viewpoint-invariant. We then train a dynamics model that predicts the evolution of the state space conditioned on the input action, enabling us to perform control in the learned state space. Though the representation itself is grounded in the 3D implicit field, the convolutional encoder is not. At test time, we overcome this limitation by performing inference-via-optimization [2, 3], enabling accurate state estimation even for out-of-distribution camera poses and, therefore, control of tasks where the goal view is specified in an entirely unseen camera perspective. These contributions enable us to perform model-based visuomotor control of complex scenes, modeling both 3D dynamics of rigid objects and fluids. Through comparison with various baselines, the learned representation from our model is more precise at describing the contents of 3D scenes, which allows it to accomplish control tasks involving both rigid objects and fluids with significantly better accuracy (Figure 1). Please check our supplementary video for better visualization.

We summarize our contributions as follows: (i) We extend an autoencoding framework with a neural radiance field rendering module and time contrastive learning that allows us to learn 3D-aware scene representations for dynamics modeling and control purely from visual observations. (ii) By incorporating the auto-decoder mechanism at test time, our framework can adjust the learned representation and accomplish the control tasks with the goal specified from camera viewpoints outside the training distribution. (iii) We are the first to augment neural radiance fields using a time-invariant dynamics model, supporting future prediction and novel view synthesis across a wide range of environments with different types of objects.

2 3D-Aware Representations for Dynamics Modeling

Inspired by Neural Radiance Fields (NeRF) [1], we propose a framework that learns a viewpoint-invariant model for dynamic environments. As shown in Figure 2, our framework has three parts: (1) an encoder that maps the input images into a latent state representation, (2) a decoder that generates an observation image under a certain viewpoint based on the state representation, and (3) a dynamics model that predicts the future state representations based on the current state and the input action.

2.1 3D-Aware Scene Representation Learning

Neural Radiance Field. Given a 3D point $\boldsymbol{x} \in \mathbb{R}^3$ in a scene and a viewing direction unit vector $\boldsymbol{d} \in \mathbb{R}^3$ from a camera, NeRF learns to predict a volumetric radiance field. This is represented using a differentiable rendering function f_{NeRF} that predicts the corresponding density σ and RGB color $\boldsymbol{c}$

using $f_{\text{NeRF}}(\boldsymbol{x}, \boldsymbol{d}) = (\sigma, \boldsymbol{c})$. To render the color of an image pixel, NeRF integrates the information along the camera ray using $\hat{\boldsymbol{C}}(\boldsymbol{r}) = \int_{h_{\text{near}}}^{h_{\text{far}}} T(h)\sigma(h)\boldsymbol{c}(h)dh$, where $\boldsymbol{r}(h) = \boldsymbol{o} + h\boldsymbol{d}$ is the camera ray with its origin $\boldsymbol{o} \in \mathbb{R}^3$ and unit direction vector $\boldsymbol{d} \in \mathbb{R}^3$, and $T(h) = \exp(-\int_{h_{\text{near}}}^{h} \sigma(s)ds)$ is the accumulated transparency between the pre-defined near depth h_{near} and far depth h_{far} along that camera ray. The mean squared error between the reconstructed color $\hat{\boldsymbol{C}}$ and the ground truth $\boldsymbol{C}$ is:

$$\mathcal{L}_{\text{rec}} = \sum_{\boldsymbol{r}} \|\hat{\boldsymbol{C}}(\boldsymbol{r}) - \boldsymbol{C}(\boldsymbol{r})\|_2^2. \tag{1}$$

Neural Radiance Field for Dynamic Scenes. One key limitation of NeRF is that it assumes the scene is static. For a dynamic scene, it must learn a separate radiance field f_{NeRF} for each time step. This severely limits the ability of NeRF to model environments that change over time, as it is both time-consuming and unclear how to transfer knowledge across time or when a new scene is similar to an old one. While other models have shown generalization across scenes [3, 4], these representations do not capture fine details. To enable f_{NeRF} to model dynamic scenes, we learn an encoding function f_{enc} that maps the visual observations to a feature representation $\boldsymbol{s}_t$ for each time step and learn the volumetric radiance field decoding function based on $\boldsymbol{s}_t$.

Specifically, given a 3D point $\boldsymbol{x}$, a viewing direction unit vector $\boldsymbol{d}$, and a scene representation $\boldsymbol{s}_t$, we learn a function $f_{\text{dec}}(\boldsymbol{x}, \boldsymbol{d}, \boldsymbol{s}_t) = (\sigma_t, \boldsymbol{c}_t)$ to predict the radiance field represented by the density σ_t and RGB color $\boldsymbol{c}_t$. Similar to NeRF, we use the integrated information along the camera ray to render the color of image pixels from an input viewpoint and then compute the image reconstruction loss using Equation 1. During each training iteration, we render two images from different viewpoints to calculate more accurate gradient updates. f_{dec} depends on the scene representation $\boldsymbol{s}_t$, forcing it to encode the 3D contents of the scene to support rendering from different camera poses.

Time Contrastive Learning. To enable the image encoder to be viewpoint invariant, we regularize the feature representation of each image $\boldsymbol{v}_t^i$ using multi-view time contrastive loss (TCN) [5] (see Figure 2a). The TCN loss encourages features of images from different viewpoints at the same time step to be similar, while repulsing features of images from different time steps to be dissimilar.

2.2 Learning the Predictive Model

After we have obtained the latent state representation $\boldsymbol{s}$, we use supervised learning to estimate the forward dynamics model, $\hat{\boldsymbol{s}}_{t+1} = f_{\text{dyn}}(\boldsymbol{s}_t, \boldsymbol{a}_t)$. Given $\boldsymbol{s}_t$ and a sequence of actions $\{\boldsymbol{a}_t, \boldsymbol{a}_{t+1}, \dots\}$, we predict H steps in the future by iteratively feeding in actions into the one-step forward model (Figure 3a). We implement f_{dyn} as an MLP network which is trained by optimizing the following loss function:

$$\mathcal{L}_{\text{dyn}} = \sum_{h=1}^{H} \|\hat{\boldsymbol{s}}_{t+h} - \boldsymbol{s}_{t+h}\|_2^2, \quad \text{where} \quad \hat{\boldsymbol{s}}_{t+h} = f_{\text{dyn}}(\hat{\boldsymbol{s}}_{t+h-1}, \boldsymbol{a}_{t+h-1}), \quad \hat{\boldsymbol{s}}_t = \boldsymbol{s}_t. \tag{2}$$

We define the final loss as a combination of the image reconstruction loss, the time contrastive loss, and the dynamics prediction loss: $\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{tc}} + \mathcal{L}_{\text{dyn}}$. We first train the encoder f_{enc} and decoder f_{dec} together using stochastic gradient descent (SGD) by minimizing $\mathcal{L}_{\text{rec}}$ and $\mathcal{L}_{\text{tc}}$, which makes sure that the learned scene representation $\boldsymbol{s}$ encodes the 3D contents and is viewpoint-invariant. We then fix the encoder parameters, and train the dynamics model f_{dyn} by minimizing $\mathcal{L}_{\text{dyn}}$ using SGD.

Please see the supplementary materials for the (1) network architecture, (2) training details, (3) how we enable viewpoint extrapolation using the auto-encoding framework, and (4) how we use the predictive model for closed-loop control.

3 Experiments

We consider four environments that involve both fluid and rigid objects for evaluating the proposed model and baseline approaches. Figure 4c shows the goal configuration, and we ask the learned model to perform three control trials where the goal is specified from different types of viewpoints.

Figure 5 shows the qualitative comparison between our model (Ours), a variant of our model that does not perform the auto-decoding test-time optimization (Ours w/o AD), and the best-performing baseline: time contrastive learning + autoencoder (TC+AE). We find that when the target view is outside the training distribution and vastly different from the agent view (bottom left image in each trial block), our full method shows a much better performance in achieving the target configuration. The variant without auto-decoding optimization and TC+AE fail to accomplish the task and exhibit an apparent deviation from the ground truth in the 3D points space.

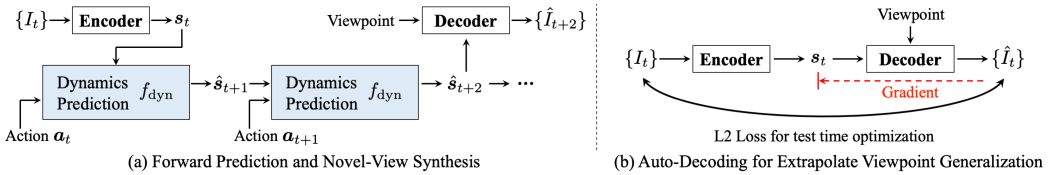


Figure 3: **Forward prediction and viewpoint extrapolation.** (a) We first feed the input image(s) at time t to the encoder to derive the scene representation $\boldsymbol{s}_t$. The dynamics model then takes $\boldsymbol{s}_t$ and the corresponding action sequence as input to iteratively predict the future. The decoder synthesizes the visual observation conditioned on the predicted state representation and an input viewpoint. (b) We propose an auto-decoding inference-via-optimization framework to enable extrapolated viewpoint generalization. Given an input image I_t taken from a viewpoint outside the training distribution, the encoder first predicts the scene representation $\boldsymbol{s}_t$. Then the decoder reconstructs the observation $\hat{I}_t$ from $\boldsymbol{s}_t$ and the camera viewpoint from I_t. We calculate the L2 distance between I_t and $\hat{I}_t$ and backpropagate the gradient through the decoder to update $\boldsymbol{s}_t$. The updating process is repeated for K iterations, resulting in a more accurate $\boldsymbol{s}_t$ of the underlying 3D scene.

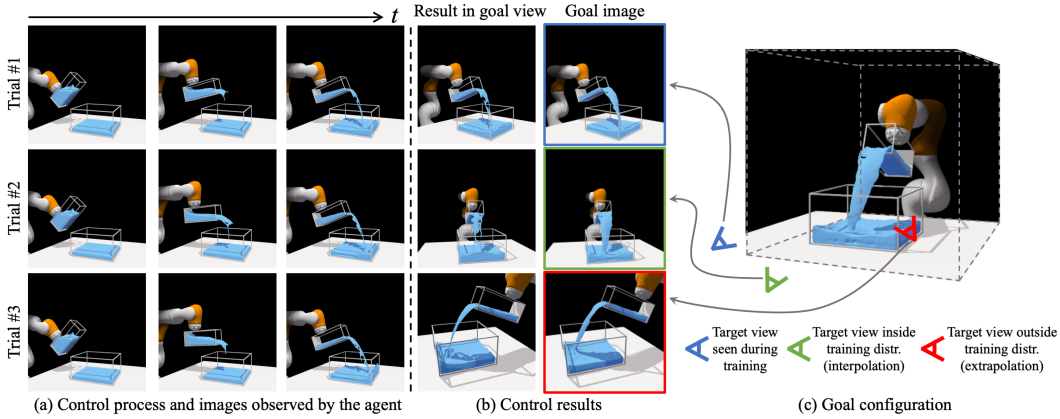


Figure 4: **Qualitative control results of our method on three types of testing scenarios.** The image on the right shows the target configuration we aim to achieve. The left three columns show the control process, which are also the input images to the agent. The fourth column is the control results from the same viewpoint as the goal image. Trial #1 specifies the goal using a different viewpoint from the agent's but has been encountered during training. Trial #2 uses a goal view that is an interpolation of training viewpoints. Trial #3 uses an extrapolated viewpoint that is outside the training distribution. Our method performs well in all settings.

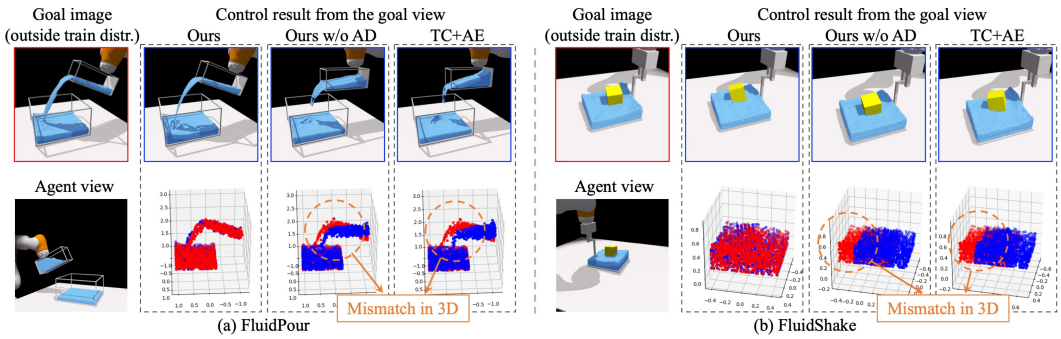


Figure 5: **Qualitative comparisons between our method and baseline approaches on the control tasks.** We show the closed-loop control results on the FluidPour and FluidShake environment. The goal image viewpoint (top-left image of each block) is outside the training distribution and is different from the viewpoint observed by the agent (bottom-left image of each block). Our final control results are much better than a variant that does not perform auto-decoding test-time optimization (Ours w/o AD) and the best-performing baseline (TC+AE), both of which fail to accomplish the task and their control results (blue points) exhibit an apparent deviation from the target configuration (red points) when measured in the 3D points space.

References

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision*, pp. 405–421, Springer, 2020.

[2] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 165–174, 2019.

[3] V. Sitzmann, M. Zollhöfer, and G. Wetzstein, "Scene representation networks: Continuous 3d-structure-aware neural scene representations," *arXiv preprint arXiv:1906.01618*, 2019.

[4] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3504–3515, 2020.

[5] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1134–1141, IEEE, 2018.