Object-Centric Image to Video Generation with Language Guidance

Angel Villar-Corrales [*1] **Gjergj Plepi** [*1] **Sven Behnke** [1]

Abstract

Accurate and flexible world models are crucial for autonomous systems to understand their environment and predict future events. Object-centric models, with structured latent spaces, have shown promise in modeling object dynamics and interactions, but often face challenges in scaling to complex datasets and incorporating external guidance, limiting their applicability in robotics. To address these limitations, we propose TextOCVP, an object-centric model for image-to-video generation guided by textual descriptions. TextOCVP parses an observed scene into object representations, called slots, and utilizes a text-conditioned transformer predictor to forecast future object states and video frames. Our approach jointly models object dynamics and interactions while incorporating textual guidance, thus leading to accurate and controllable predictions. Our method's structured latent space offers enhanced control over the prediction process, outperforming several image-to-video generative baselines. Additionally, we demonstrate that structured object-centric representations provide superior controllability and interpretability, facilitating the modeling of object dynamics and enabling more precise and understandable predictions. Videos and code are available at https://play-slot.github.io/TextOCVP/.

1. Introduction

Understanding and reasoning about the real world is essential for enabling autonomous systems to better comprehend their surroundings, predict future events, and adapt their actions accordingly. Humans achieve these capabilities by perceiving the environment as a structured composition of individual objects that interact and evolve dynamically over time (Kahneman et al., 1992). Neural networks equipped with such compositional inductive biases have shown the ability to learn structured object-centric representations of the world, which enable desirable properties, such as out-of-distribution generalization (Dittadi et al., 2022), compositionality (Greff et al., 2020), or sample efficiency (Mosbach et al., 2024).

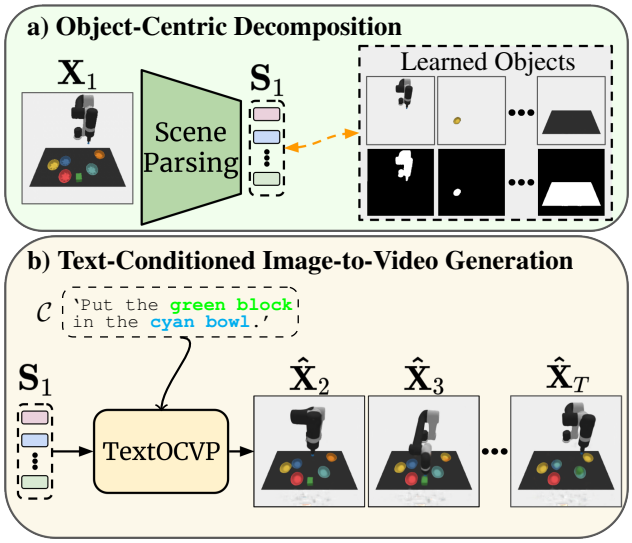

Figure 1: Overview of TextOCVP. **(a)** Our model first parses a reference frame $\mathbf{X}_1$ into its object components $\mathbf{S}_1$. **(b)** Our TextOCVP predictor jointly models object dynamics and interactions while integrating textual guidance, generating future object states and video frames that align with the provided textual instructions.

Recent advances in unsupervised object-centric representation learning have progressed from extracting object representations in synthetic images (Locatello et al., 2020; Lin et al., 2020b) to modeling objects in video (Kipf et al., 2022; Singh et al., 2022) and scaling to real-world scenes (Seitzer et al., 2023; Zadaianchuk et al., 2024). These developments have enabled object-level dynamics modeling for future prediction and planning. Notably, models like SlotFormer (Wu et al., 2023) or OCVP (Villar-Corrales et al., 2023) introduced object-centric world models that explicitly model spatio-temporal relationships between objects, shifting away from image-level approaches that ignore scene compositionality. Despite these advance-

[*]Equal contribution [1]Autonomous Intelligent Systems, Computer Science Institute VI – Intelligent Systems and Robotics, Center for Robotics and the Lamarr Institute for Machine Learning and Artificial Intelligence. Correspondence to: Angel Villar-Corrales <villar@ais.uni-bonn.de>.

ments, current approaches struggle with complex object appearances and dynamics, and lack the ability to incorporate external guidance, thus limiting their effectiveness as world models in robotic applications.

To address these challenges, we propose *TextOCVP*, a novel object-centric model for image-to-video generation with language guidance, illustrated in Fig. 1. Given a reference image and text instruction, TextOCVP extracts object representations and predicts their evolution using a novel text-conditioned object-centric transformer. This module forecasts future object representations by modeling object dynamics and interactions, while also incorporating textual information through a text-to-slot attention mechanism. By jointly modeling spatio-temporal object relationships and integrating textual guidance, TextOCVP generates future object states and video frames that align with the provided instructions.

We evaluate our approach through extensive experiments focusing on image-to-video generation tasks with varying levels of complexity. Our results show that TextOCVP outperforms existing text-conditioned methods by effectively leveraging object-centric representations. This demonstrates the significant advantage of incorporating textual guidance into structured object-centric representations, particularly for scenes featuring multiple moving objects.

Through an in-depth model analysis, we demonstrate the importance of our proposed object-centric latent space, which allows TextOCVP to generate video sequences that closely align with language instructions. We further demonstrate how TextOCVP adapts its generated video sequences based on different captions by routing the text information to the relevant object representations.

In summary, our contributions are as follows:

- We propose TextOCVP, a novel text-driven image-to-video generation model, featuring a text-conditioned object-centric predictor that integrates textual guidance via a text-to-slot attention mechanism.
- Through extensive evaluation, we show that TextOCVP outperforms existing text-conditioned models by leveraging object-centric representations.
- We demonstrate that TextOCVP is controllable, seamlessly adapting to diverse textual instructions.

2. Related Work

2.1. Object-Centric Learning

Representation learning, the ability to extract meaningful features from data, often improves model performance by enhancing its understanding of the input space (Bengio et al., 2013). Object-centric representation learning focuses on simultaneously learning representations that characterize individual objects within an input image or video.

Recently, object-centric models have progressed from learning object representations from synthetic images (Greff et al., 2019; Lin et al., 2020a; Villar-Corrales & Behnke, 2022; Locatello et al., 2020) to videos (Kipf et al., 2022; Singh et al., 2022; Elsayed et al., 2022), and real-world scenes (Seitzer et al., 2023; Aydemir et al., 2023; Zadaianchuk et al., 2024; Kakogeorgiou et al., 2024). The learned object representations benefit downstream tasks, such as reinforcement learning (Mosbach et al., 2024) or visual-question answering (Wu et al., 2023), among others.

2.2. Video Prediction and Generation

Future frame video prediction (VP) is the task of forecasting the upcoming T video frames conditioned on the preceding C seed frames (Oprea et al., 2020). When the number of seed frames is $C = 1$, this task is often referred to as image-to-video generation. Several methods have been proposed to address this challenge, leveraging 2D convolutions (Gao et al., 2022; Chiu et al., 2020), 3D convolutions (Vondrick et al., 2016; Tulyakov et al., 2018), recurrent neural networks (RNNs) (Denton & Fergus, 2018; Villar-Corrales et al., 2022; Wang et al., 2022; Guen & Thome, 2020; Franceschi et al., 2020), Transformers (Rakhimov et al., 2021; Ye & Bilodeau, 2022; 2023), or diffusion models (Höppe et al., 2022; Ho et al., 2022).

2.2.1. Object-Centric Video Prediction

Object-centric VP presents a structured approach that explicitly models the dynamics and interactions of individual objects to forecast future video frames. These methods typically involve three main steps: decomposing seed frames into object representations, forecasting future object states using a dynamics model, and rendering video frames from the predicted object representations. Various approaches have been proposed for this task, using different architectural designs such as RNNs (Creswell et al., 2021; Zoran et al., 2021; Nguyen et al., 2024) or transformers (Wu et al., 2021; 2023; Villar-Corrales et al., 2023; Song et al., 2023; Daniel & Tamar, 2024). Despite promising results, these models are currently limited to simple deterministic datasets or rely on action-conditioning (Mosbach et al., 2024) or inferred latent vectors (Villar-Corrales & Behnke, 2025). In contrast, our proposed model forecasts future video frames conditioned on past object slots and textual descriptions.

2.2.2. Text-Conditioned Video Prediction

This category of VP models leverages text descriptions to provide appearance, motion and action cues that guide the generation of future frames. This task was first pro-

posed by Hu et al. (2022), who utilized a VQ-VAE to encode the images into visual token representations, and modeled the image dynamics with an axial transformer that jointly processes such tokens along with text descriptions. TVP (Song et al., 2024) leverages RNNs to learn text representations, which condition a GAN-based framework in order to generate videos from a single frame. MMVG (Fu et al., 2023) combines a VQ-GAN (Esser et al., 2021) with a masked transformer predictor. More recently, several methods leverage diffusion models with text conditioning (Gu et al., 2024; Ni et al., 2023; Chen et al., 2023).

Concurrently with our work, Wang et al. (2024) combine object-centric learning with text-conditional image-to-video generation. The authors propose an autoregressive diffusion model conditioned on object slots and text descriptions and evaluate on simple datasets. In contrast, we explicitly model the object dynamics with an autoregressive transformer, and evaluate our approach on more complex robotics simulations.

3. Method

We propose TextOCVP – a novel object-centric model for text-conditioned image-to-video generation. Given an initial reference image $\mathbf{X}_1$ and a text caption $\mathcal{C}$, TextOCVP generates the subsequent T video frames $\hat{\mathbf{X}}_{2:T+1}$, which maintain a similar appearance and structural composition as the reference image, and follow the motion described in the text caption.

TextOCVP, which is illustrated in Fig. 2, implements an object-centric approach, in which the reference frame $\mathbf{X}_1$ is first decomposed with a scene parsing module (Sec. 3.1) into a set of $N_\mathbf{S}$ D-dimensional object representations called slots $\mathbf{S}_1 \in \mathbb{R}^{N_\mathbf{S} \times D}$, where each slot represents a single object in the image. The object slots are fed to a text-conditioned transformer predictor (Sec. 3.2), which jointly models their spatio-temporal relations, and incorporates the textual information from the caption $\mathcal{C}$ as a guidance for predicting the future object slots $\hat{\mathbf{S}}_{2:T+1}$. Finally, the predicted object slots are decoded in order to render object representations and video frames (Sec. 3.3).

We propose two different TextOCVP variants, which differ in the underlying object-centric decomposition modules. Specifically, TextOCVP$_{\text{SAVi}}$ leverages SAVi (Kipf et al., 2022), whereas TextOCVP$_{\text{DINO}}$ extends the DINOSAUR (Seitzer et al., 2023) framework for recursive object-centric video decomposition and video rendering.

3.1. Scene Parsing

The scene parsing module decomposes a video sequence $\mathbf{X}_{1:\tau}$ into a set of permutation-invariant object representations called slots $\mathbf{S}_{1:\tau} = (\mathbf{S}_1, ..., \mathbf{S}_\tau)$, with $\mathbf{S}_t = (\mathbf{s}_t^1, ..., \mathbf{s}_t^{N_\mathbf{S}})$, where each slot $\mathbf{s}$ represents a single object.

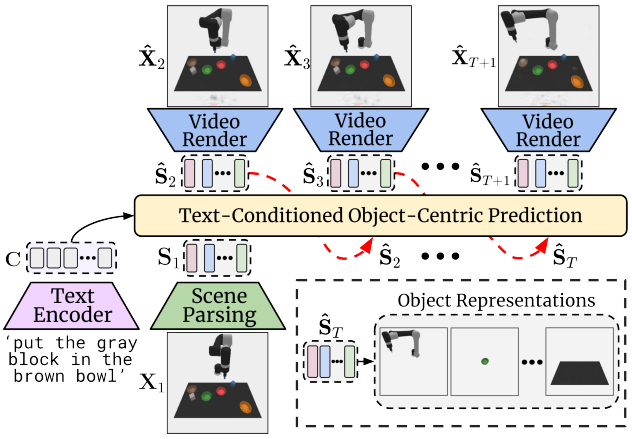

Figure 2: Overview of TextOCVP. TextOCVP parses the reference frame $\mathbf{X}_1$ into object representations $\mathbf{S}_1$. The text-conditioned object-centric predictor models object dynamics and interactions, incorporating information from the description $\mathbf{C}$ to predict future object states and frames.

For learning the object representations, we leverage the object-centric video decomposition framework proposed by Kipf et al. (2022). At time step t, the corresponding input frame $\mathbf{X}_t$ is encoded with a feature extractor module into a set of D_h-dimensional feature maps $\mathbf{h}_t \in \mathbb{R}^{L \times D_h}$ representing L spatial locations. The feature extractor is convolutional neural network in our TextOCVP$_{\text{SAVi}}$ variant and a DINO-pretrained vision transformer (Caron et al., 2021) in TextOCVP$_{\text{DINO}}$. These feature maps are processed with a Slot Attention (Locatello et al., 2020) corrector, which updates the previous slots $\mathbf{S}_{t-1}$ based on visual features from the current frame following an iterative attention mechanism. Namely, Slot Attention performs cross-attention, with the attention weights normalized over the slot dimension, thus encouraging the slots to compete to represent parts of the input. It then updates the slots using a Gated Recurrent Unit (Cho et al., 2014) (GRU). Formally, Slot Attention updates the previous slots $\mathbf{S}_{t-1}$ by:

$$\mathbf{A} = \underset{N_\mathbf{S}}{\text{softmax}}\left(\frac{q(\mathbf{S}_{t-1})k(\mathbf{h}_t)^T}{\sqrt{D}}\right) \in \mathbb{R}^{N_\mathbf{S} \times L}, \quad (1)$$

$$\mathbf{S}_t = \text{GRU}(W_t v(\mathbf{h}), \mathbf{S}_{t-1}) \text{ with } W_{i,j} = \frac{\mathbf{A}_{i,j}}{\sum_{l=1}^{L} \mathbf{A}_{i,l}}, \quad (2)$$

where k, q and v are learned linear projections that map input features and slots into a common dimension. The final output of this module is the set of slots $\mathbf{S}_t$, representing the objects of the input frame.

3.2. Text-Conditioned Object-Centric Predictor

Our proposed text-conditioned predictor module, depicted in Fig. 3, autoregressively forecasts future object states

conditioned on the object slots from the reference frame $\mathbf{S}_1$ and a textual description $\mathcal{C}$.

To condition the prediction process, the text description $\mathcal{C}$ is encoded into text token embeddings $\mathbf{C}$ employing an encoder-only transformer module. We experiment with different variants of this module, including a vanilla transformer encoder (Vaswani et al., 2017) or a pretrained T5 (Raffel et al., 2020) module.

At time step t, the predictor receives as input the corresponding text embeddings $\mathbf{C}$, as well as the previous object slots $\mathbf{S}_{1:t}$, which are initially mapped via an MLP to the predictor token dimensionality D_{Pred}. Additionally, these tokens are augmented with a temporal positional encoding, which applies the same sinusoidal positional embedding to all tokens from the same time-step, thus preserving the inherent permutation-equivariance of the objects.

Each layer of our predictor module mirrors the transformer decoder architecture (Vaswani et al., 2017). First, a self-attention layer enables every slot to attend to all other object representations in the sequence, modeling the spatio-temporal relations between objects. Subsequently, a text-to-slot cross-attention layer enhances the slot representations by incorporating important features from the text embeddings, such as motion or appearance information. Finally, an MLP is independently applied for each token. This process is repeated in every predictor layer, resulting in the predicted object slots of the subsequent time step $\hat{\mathbf{S}}_{t+1}$. Furthermore, we apply a residual connection from $\mathbf{S}_t$ to $\hat{\mathbf{S}}_{t+1}$, which improves the temporal consistency of the predictions. This process is repeated autoregressively to obtain slot predictions for T subsequent time steps.

3.3. Video Rendering

The video rendering module decodes the predicted slots $\hat{\mathbf{S}}_t$ to render the corresponding video frame $\hat{\mathbf{X}}_t$. We leverage two variants of the video rendering module, for our TextOCVP$_{\text{SAVi}}$ and TextOCVP$_{\text{DINO}}$ variants, respectively.

TextOCVP$_{\text{SAVi}}$ Decoder This variant independently decodes each slot in $\hat{\mathbf{S}}_t$ with a CNN-based Spatial Broadcast Decoder (Watters et al., 2019), rendering an object image $\mathbf{o}_t^n$ and mask $\mathbf{m}_t^n$ for each slot $\mathbf{s}_t^n$. The object masks are normalized across the slot dimension, and the representations are combined via a weighted sum to render video frames:

$$\hat{\mathbf{X}}_t = \sum_{n=1}^{N_\mathbf{S}} \mathbf{o}_t^n \cdot \tilde{\mathbf{m}}_t^n \quad \text{with} \quad \tilde{\mathbf{m}}_t^n = \underset{N_\mathbf{S}}{\text{softmax}}(\mathbf{m}_t^n). \tag{3}$$

TextOCVP$_{\text{DINO}}$ Decoder This decoder variant decodes the object slots in two distinct stages. First, following DINOSAUR (Seitzer et al., 2023), an MLP-based Spatial Broadcast Decoder (Watters et al., 2019) is used to generate object features along with their corresponding masks. Similar to the TextOCVP$_{\text{SAVi}}$ decoder, the object masks are normalized and combined with the object features in order to reconstruct the encoded features $\hat{\mathbf{h}}_t \in \mathbb{R}^{L \times D_h}$. In the second stage, the reconstructed features $\hat{\mathbf{h}}_t$ are arranged into a grid format and processed with a CNN decoder to generate the corresponding video frame $\hat{\mathbf{X}}_t$.

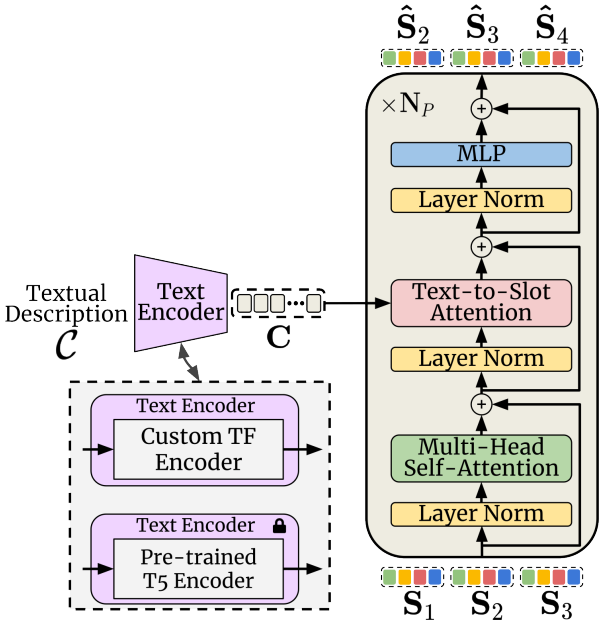

Figure 3: Text-conditioned object-centric predictor.

3.4. Training and Inference

Our proposed TextOCVP is trained in two different stages.

Object-Centric Learning We first train the scene parsing and video rendering modules for decomposing video frames into object-centric representations by minimizing a reconstruction loss. In the TextOCVP$_{\text{SAVi}}$ variant, these modules are trained simply by reconstructing the input images, whereas in TextOCVP$_{\text{DINO}}$ they are trained by jointly minimizing an image and a feature reconstruction loss:

$$\mathcal{L}_{\text{SAVi}} = \frac{1}{T} \sum_{t=1}^{T} ||\hat{\mathbf{X}}_t - \mathbf{X}_t||, \tag{4}$$

$$\mathcal{L}_{\text{DINO}} = \frac{1}{T} \sum_{t=1}^{T} ||\hat{\mathbf{X}}_t - \mathbf{X}_t|| + ||\hat{\mathbf{h}}_t - \mathbf{h}_t||. \tag{5}$$

Predictor Training Given the pretrained scene parsing and rendering modules, we train our TextOCVP predictor for text-conditioned video generation using a dataset containing paired videos and text descriptions. Namely, given the object representations from a reference frame $\mathbf{S}_1$ and the textual description $\mathcal{C}$, our TextOCVP predictor forecasts the subsequent object slots $\hat{\mathbf{S}}_2$, which are decoded into a

predicted video frame $\hat{\mathbf{X}}_2$. This process is repeated autoregressively, i.e. the predicted slots are appended to the input in the next time step, in order to generate the set of slots for the subsequent T time steps. This autoregressive training, in contrast to teacher forcing, enforces our predictor to operate with imperfect inputs, leading to better modeling of long-term dynamics at inference time.

Our predictor is trained by minimizing the following combined loss:

$$\mathcal{L}_{\text{TextOCVP}} = \frac{1}{T}\sum_{t=2}^{T+1} \lambda_{\text{Img}}\mathcal{L}_{\text{Img}} + \lambda_{\text{Slot}}\mathcal{L}_{\text{Slot}}, \tag{6}$$

$$\text{with } \mathcal{L}_{\text{Img}} = ||\hat{\mathbf{X}}_t - \mathbf{X}_t||_2^2 \text{ and } \mathcal{L}_{\text{Slot}} = ||\hat{\mathbf{S}}_t - \mathbf{S}_t||_2^2, \tag{7}$$

where $\mathcal{L}_{\text{Img}}$ measures the future frame prediction error, and $\mathcal{L}_{\text{Slot}}$ enforces the alignment of the predicted object slots with the actual inferred object-centric representations.

Inference At inference time, TextOCVP receives as input a single reference frame and a language instruction. Our model parses the seed frame into object slots and autoregressively predicts future object states and video frames conditioned on the given textual description. By modifying the language instruction, TextOCVP can generate a new sequence continuation that performs the specified task while preserving a consistent scene composition.

4. Experiments

4.1. Experimental Setup

4.1.1. Datasets

We evaluate the performance of TextOCVP for text-driven image-to-video generation on two different datasets, namely CATER and CLIPort. Further dataset details are provided in Appendix D.

CATER CATER (Girdhar & Ramanan, 2020) is a dataset that consists of long video sequences, featuring two simultaneously moving 3D objects, with the motion described by a textual caption. We used the CATER-Hard variant introduced by Hu et al. (2022). This dataset contains 30,000 sequences featuring between three and eight objects, two of which follow a predefined action pattern. In our experiments, we resize the images to 64×64.

CLIPort CLIPort (Shridhar et al., 2022) is a robot manipulation dataset consisting of video-caption pairs. We employ 21,000 336×336 sequences of the Put-Block-In-Bowl task. Each sequence features a table with multiple bowls and blocks of different color, placed at random positions on the table. The corresponding caption describes the action of the robot arm picking a block of a specific color and putting it into a specific bowl.

Table 1: Quantitative evaluation on CATER for prediction horizons of $T = 9$ and $T = 19$. TextOCVP outperforms all baselines. Best two results are highlighted in boldface and underlined, respectively.

	$\text{CATER}_{1\to9}$		$\text{CATER}_{1\to19}$	
Method	SSIM↑	LPIPS↓	SSIM↑	LPIPS↓
OCVP	0.874	<u>0.078</u>	0.854	<u>0.101</u>
Non-OC TF	0.874	0.092	0.849	0.112
SEER	0.723	0.245	0.535	0.299
MAGE	<u>0.877</u>	0.108	<u>0.871</u>	0.111
TextOCVP	**0.924**	**0.035**	**0.904**	**0.042**

4.1.2. Baselines

We select several baselines to benchmark TextOCVP against state-of-the-art image-to-video generation methods and to analyze different design choices within our architecture. To assess the impact of text conditioning, we compare TextOCVP with OCVP-Seq (Villar-Corrales et al., 2023), an unconditional object-centric video prediction model. To evaluate the role of object-centric representations, we introduce a TextOCVP variant (*Non-OC TF*) that replaces the structured slot representation with a single high-dimensional embedding. Finally, we benchmark our approach against three text-conditioned image-to-video generation models: MAGE (Hu et al., 2022), $\text{MAGE}_{\text{DINO}}$, and SEER (Gu et al., 2024). Additional details for the baselines can be found in Appendix C.

4.1.3. Implementation Details

All our models are implemented in PyTorch and trained on a single NVIDIA A6000 (48Gb) GPU. $\text{TextOCVP}_{\text{SAVi}}$ closely follows Kipf et al. (2022) for the design of the scene parsing and video rendering modules. $\text{TextOCVP}_{\text{DINO}}$ leverages DINOv2 (Oquab et al., 2023) as image encoder, uses a four-layer MLP-based Spatial-Broadcast Decoder (Watters et al., 2019), which is shared among all slots, to decode the object slots into object features and masks. Additionally, it uses a CNN decoder to map the reconstructed scene features back to images. On the CATER dataset, we use the $\text{TextOCVP}_{\text{SAVi}}$ variant with eight 128-dimensional object slots; whereas on CLIPort we employ the $\text{TextOCVP}_{\text{DINO}}$ variant with ten 128-dimensional slots. Our predictor module is an eight-layer transformer with 512-dimensional tokens, eight attention heads, and a hidden dimension of 1024. Further implementation details are provided in Appendix B.

4.2. Results

4.2.1. CATER Results

On the CATER dataset, we train the models to generate nine future frames given a single reference frame and a text

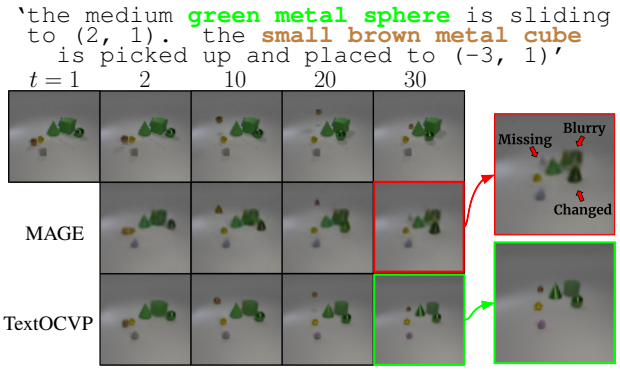

Figure 4: Qualitative evaluation for text-driven image-to-video generation on CATER. Top row shows ground truth frames. TextOCVP generates sharp future frames, whereas MAGE blurs and misses objects.

Table 2: Quantitative evaluation on CLIPort for prediction horizons of $T = 9$ and $T = 19$. TextOCVP and MAGE$_{\text{DINO}}$ outperform all methods. Best two results are highlighted in boldface and underlined, respectively.

	CLIPort$_{1\to9}$		**CLIPort**$_{1\to19}$	
Method	SSIM↑	LPIPS↓	SSIM↑	LPIPS↓
Non-OC TF	0.901	0.184	0.872	0.210
SEER	0.887	0.141	0.622	0.331
MAGE$_{\text{DINO}}$	<u>0.940</u>	<u>0.064</u>	**0.931**	**0.075**
TextOCVP	**0.950**	**0.062**	**0.931**	<u>0.078</u>

caption. In Tab. 1 we report quantitative evaluations on CATER using the same setting as in training, i.e. predicting $T = 9$ frames, as well as when predicting $T = 19$ future frames. In both settings, TextOCVP outperforms all other models, demonstrating superior perceptual quality.

Fig. 4 shows a qualitative comparison between TextOCVP and MAGE. Our proposed method generates a sequence that is closely aligned to the ground-truth, whereas MAGE predictions feature multiple errors and artifacts, including missing objects, blurry contours, and significant changes on object shapes. This highlights the effectiveness of object-centric representations to accurately represent and model object dynamics. Additional qualitative results are provided in Appendix E.

4.2.2. CLIPort Results

Tab. 2 shows a detailed quantitative evaluation for image-to-video generation on the CLIPort dataset. TextOCVP outperforms all baselines when evaluated on the same setting as in training (i.e. $1 \to 9$). When generating for longer horizons, our model shows competitive performance, closely following MAGE$_{\text{DINO}}$.

In our qualitative evaluations, we observe that TextOCVP often generates the most accurate generations given the reference frame and text description. Fig. 5 shows two qualitative examples comparing TextOCVP with the MAGE$_{\text{DINO}}$ baseline. MAGE$_{\text{DINO}}$ fails to complete the task outlined in the textual description, as it stops generating consistent robot motion after 30 frames (Fig. 5a), or misses the target block after several prediction time steps (Fig. 5b). In contrast, TextOCVP correctly generates the sequences following instructions provided in the given textual description.

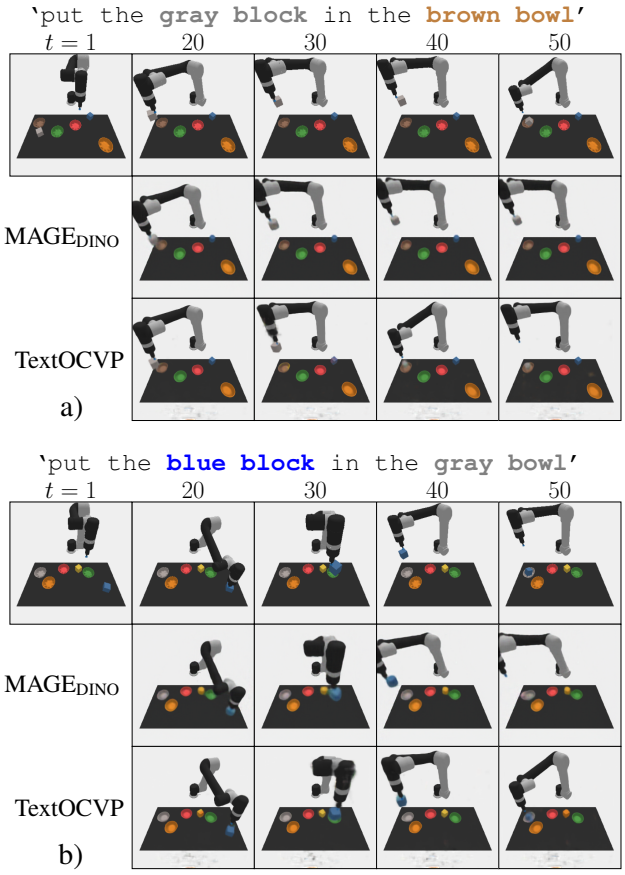

Figure 5: Qualitative evaluation on CLIPort. Top rows depict the ground truth frames. TextOCVP successfully completes the pick-and-place task, whereas MAGE fails to predict robot motion (a) or the block disappears (b).

However, we identify that TextOCVP often generates artifacts in the background and lacks textured details in certain objects, thus degrading its quantitative performance despite generating accurate future frames. Additional qualitative results are provided in Appendix E.

4.3. Model Analysis

4.3.1. Ablation Studies

We perform several ablation studies to support and validate the architectural choices of our model components and their impact on TextOCVP's image-to-video generation performance. The results are presented in Tab. 3.

Table 3: Ablation studies. We ablate different architectural design choices in our proposed TextOCVP model.

(a) Effect of number of layers in predictor.

	$\textbf{CATER}_{1\rightarrow9}$		
# Layers	PSNR↑	SSIM↑	LPIPS↓
2	31.83	0.912	0.041
4	32.58	0.921	0.036
8	**32.82**	**0.924**	**0.035**

(b) Evaluation of different text encoders on CATER and CLIPort.

	$\textbf{CATER}_{1\rightarrow9}$			$\textbf{CLIPort}_{1\rightarrow9}$		
Text Encoder	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Custom TF	31.62	0.909	0.045	-	-	-
Frozen T5	**31.83**	**0.912**	**0.041**	**26.99**	**0.950**	**0.062**
FT T5	-	-	-	26.85	0.947	0.065

(c) Effect of residual connection in predictor.

	$\textbf{CLIPort}_{1\rightarrow9}$		
Residual	PSNR↑	SSIM↑	LPIPS↓
No	26.55	0.946	0.066
Yes	**26.99**	**0.950**	**0.062**

(d) Effect of the number of slots $N_{\mathbf{S}}$.

	$\textbf{CLIPort}_{1\rightarrow9}$		
# Slots	PSNR↑	SSIM↑	LPIPS↓
8	25.95	0.890	0.079
10	**26.99**	**0.950**	**0.062**

Table 4: Quantitative evaluation on a CLIPort test set with unseen colors. We report the video generation performance and its drop relative to the known-color dataset.

	$\textbf{CLIPort}_{1\rightarrow9}$		$\textbf{CLIPort}_{1\rightarrow19}$	
Method	SSIM↑	LPIPS↓	SSIM↑	LPIPS↓
$\text{MAGE}_{\text{DINO}}$	0.935-0.5%	0.076-19%	0.924-0.7%	0.087-16%
TextOCVP	**0.946-0.4%**	**0.066-6.4%**	**0.927-0.4%**	**0.083-6.4%**

Number of Layers (Tab. 3a) TextOCVP performs best with $N_{\text{Pred}} = 8$ predictor layers. Scaling beyond eight layers was not explored, as the performance improvements when increasing from four to eight layers were marginal.

Residual Connection (Tab. 3c) Applying a residual connection to the predictor output, $\hat{\mathbf{S}}_{t+1} := \hat{\mathbf{S}}_{t+1} + \mathbf{S}_t$, enhances the model's performance by improving the temporal consistency of the predicted slots.

Text Encoder (Tab. 3b) We evaluate the performance of different text encoders, including a vanilla transformer, a frozen T5 encoder, and a fine-tuned (FT) T5. A frozen T5 module led to the best TextOCVP performance on both CATER and CLIPort datasets. We argue that given the relatively small vocabulary size on CLIPort, fine-tuning T5 did not prove to be beneficial.

Number of Slots (Tab. 3d) Using $N_{\mathbf{S}} = 10$ slots resulted in better performance for TextOCVP on CLIPort, even though each scene can be described with eight slots (six objects, robot arm and background). This observation suggests that additional slots can be used as registers in order to help with internal computations (Darcet et al., 2024).

4.3.2. Model Robustness

We evaluate the performance of TextOCVP and $\text{MAGE}_{\text{DINO}}$ on a CLIPort evaluation set featuring color variations in the text instructions that were not encountered during training. This evaluation measures how well the models generalize to unseen scene features, highlighting their ability to handle novel configurations.

Tab. 4 presents the results on this evaluation set. We report both the video generation performance as well as the performance drop relative to the evaluation set with known colors. Our model outperforms $\text{MAGE}_{\text{DINO}}$ for both prediction horizons. Most notably, TextOCVP demonstrates significantly higher robustness to objects with previously unseen colors compared to the baseline, showing a much smaller drop in performance for the perceptual LPIPS metric. This highlights the effectiveness of leveraging object-centric representations for image-to-video generation and planning, in contrast to relying on holistic scene representations that struggle with novel scene compositions.

4.3.3. Controllability

A key objective of text-driven image-to-video generation is to provide control over the model generations. This is achieved through language instructions that describe the objects in the scene and their expected motion. We qualitatively assess the controllability of our model on both the CATER and CLIPort datasets.

CATER In Fig. 6, we showcase the control that TextOCVP provides over its generations. We qualitatively evaluate this ability by generating multiple videos conditioned on the same reference frame while varying the language instruction. We experiment with altering both the moving and target objects and their actions in the language instruction, as well as providing instructions that include a number of tasks beyond what the model was trained on.

As shown in Fig. 6, TextOCVP successfully identifies the objects described in the text and executes the instructions accordingly. Notably, our model can distinguish between two nearly identical purple cones in the scene, despite their

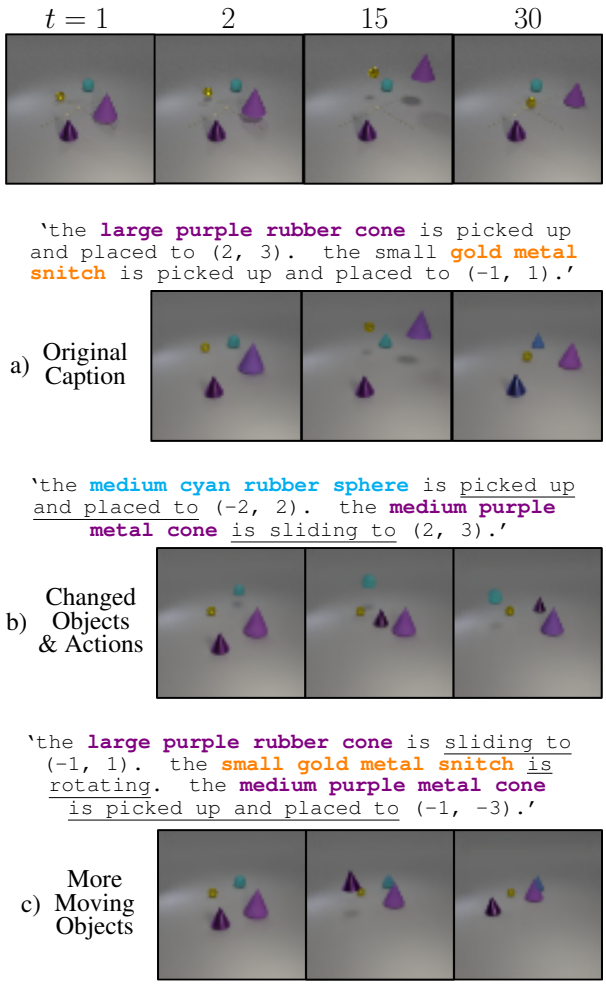

Figure 6: Qualitative evaluation of TextOCVP controllability on CATER. Top row shows the ground truth sequence. We underline the changed actions with respect to the original caption. We demonstrate TextOCVP's controllability by generating multiple sequences from the same reference frame, each conditioned on a different text instruction.

identical shapes and color, and correctly generates a sequence with the specified motion.

This demonstrates that key object attributes, such as size or color, are effectively captured through the text-to-slot attention mechanism. Leveraging this information, the text-conditioned predictor accurately predicts the future motion of each individual object as specified in the instruction. These results highlight the benefits of combining object-centric representations with text-based conditioning.

CLIPort We perform a similar experiment on the CLIPort dataset, as shown in Fig. 7. A single frame containing multiple colored blocks and bowls is provided to TextOCVP. By modifying the textual instruction, we can control which block the robot arm picks up and the bowl

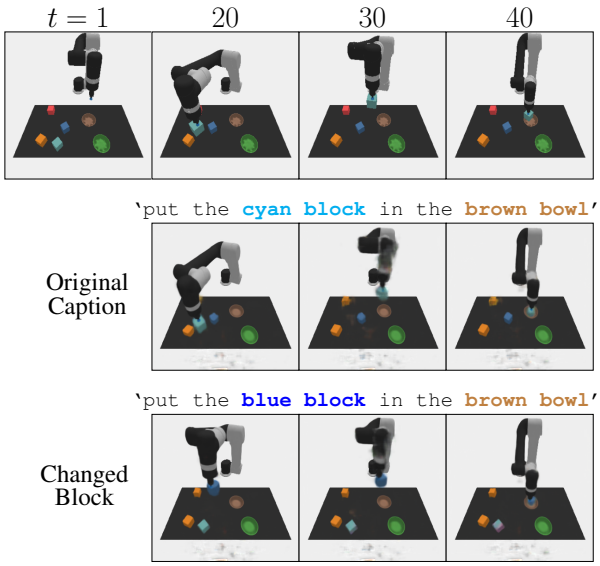

Figure 7: Qualitative evaluation of TextOCVP controllability on CLIPort. Top row depicts ground-truth frames. TextOCVP correctly generates a sequence where the robot picks up the correct block and places it into the bowl as specified in the textual instruction.

where it is placed. In both examples, TextOCVP successfully selects the correct block and places it in the specified bowl. Furthermore, the movement of the robot arm adapts to the motion described in the instruction, demonstrating accurate and responsive control.

5. Conclusion

We presented TextOCVP, a novel object-centric model for text-driven image-to-video generation. Given a single input image and a text description, TextOCVP generates a sequence that matches the description by parsing the environment into object representations and modeling their dynamics conditioned on the textual instruction. This is achieved with a novel text-conditioned object-centric transformer, which predicts future object states by modeling the spatio-temporal relationships between objects while incorporating the guidance from text. Through extensive evaluations, we demonstrated that TextOCVP outperforms existing text-conditioned models for the image-to-video generation, highlighting our model's ability to generate consistent long sequences and adapt its predictions based on language input. Moreover, we validated our architectural choices through ablation studies, highlighting the importance of combining textual and object-centric information. With its structured latent space and superior controllability, TextOCVP offers potential for robotics applications, where it can serve as a controllable world model, enabling efficient planning, reasoning, and decision-making.

Acknowledgment

This work was funded by grant BE 2556/16-2 (Research Unit FOR 2535 Anticipating Human Behavior) of the German Research Foundation (DFG)

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Aydemir, G., Xie, W., and Guney, F. Self-supervised object-centric learning for videos. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2013.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Chen, X., Liu, Z., Chen, M., Feng, Y., Liu, Y., Shen, Y., and Zhao, H. Livephoto: Real image animation with text-guided motion control. *arXiv preprint arXiv:2312.02928*, 2023.

Chiu, H.-k., Adeli, E., and Niebles, J. C. Segmenting the future. *IEEE Robotics and Automation Letters (RAL)*, 5 (3):4202–4209, 2020.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

Clark, A., Donahue, J., and Simonyan, K. Adversarial video generation on complex datasets. In *International Conference on Learning Representations (ICLR)*, 2020.

Creswell, A., Kabra, R., Burgess, C., and Shanahan, M. Unsupervised object-based transition models for 3d partially observable environments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Daniel, T. and Tamar, A. DDLP: Unsupervised Object-centric Video Prediction with Deep Dynamic Latent Particles. *Transactions on Machine Learning Research (TMLR)*, 2024.

Darcet, T., Oquab, M., Mairal, J., and Bojanowski, P. Vision transformers need registers. In *International Conference on Learning Representations (ICLR)*, 2024.

Denton, E. and Fergus, R. Stochastic video generation with a learned prior. In *International Conference on Machine Learning (ICML)*, 2018.

Dittadi, A., Papa, S., De Vita, M., Schölkopf, B., Winther, O., and Locatello, F. Generalization and robustness implications in object-centric learning. In *International Conference on Machine Learning (ICML)*, 2022.

Elsayed, G., Mahendran, A., Van Steenkiste, S., Greff, K., Mozer, M. C., and Kipf, T. Savi++: Towards end-to-end object-centric learning from real-world videos. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Esser, P., Rombach, R., and Ommer, B. Taming transformers for high-resolution image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12873–12883, 2021.

Franceschi, J.-Y., Delasalles, E., Chen, M., Lamprier, S., and Gallinari, P. Stochastic latent residual video prediction. In *International Conference on Machine Learning (ICML)*, 2020.

Fu, T.-J., Yu, L., Zhang, N., Fu, C.-Y., Su, J.-C., Wang, W. Y., and Bell, S. Tell me what happened: Unifying text-guided video completion via multimodal masked video generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Gao, Z., Tan, C., Wu, L., and Li, S. Z. SimVP: Simpler yet better video prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Girdhar, R. and Ramanan, D. CATER: A diagnostic dataset for Compositional Actions and TEmporal Reasoning. In *International Conference on Learning Representations (ICLR)*, 2020.

Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al. The" something something" video database for learning and evaluating visual common sense. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

Greff, K., Kaufman, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M. M., and Lerchner, A. Multi-object representation learning with iterative variational inference. In *International Conference on Machine Learning (ICML)*, 2019.

Greff, K., Van Steenkiste, S., and Schmidhuber, J. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

Gu, X., Wen, C., Ye, W., Song, J., and Gao, Y. Seer: Language instructed video prediction with latent diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

Guen, V. L. and Thome, N. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models. *Advances in Neural Information Processing Systems (NeurIPS)*, 35: 8633–8646, 2022.

Höppe, T., Mehrjou, A., Bauer, S., Nielsen, D., and Dittadi, A. Diffusion models for video prediction and infilling. *arXiv preprint arXiv:2206.07696*, 2022.

Hu, Y., Luo, C., and Chen, Z. Make it move: Controllable image-to-video generation with text descriptions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Jiang, J., Deng, F., Singh, G., and Ahn, S. Object-centric slot diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Kahneman, D., Treisman, A., and Gibbs, B. J. The reviewing of object files: Object-specific integration of information. *Cognitive psychology*, 24(2):175–219, 1992.

Kakogeorgiou, I., Gidaris, S., Karantzalos, K., and Komodakis, N. Spot: Self-training with patch-order permutation for object-centric learning with autoregressive transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Kipf, T., Elsayed, G. F., Mahendran, A., Stone, A., Sabour, S., Heigold, G., Jonschkowski, R., Dosovitskiy, A., and Greff, K. Conditional Object-Centric Learning from Video. In *International Conference on Learning Representations (ICLR)*, 2022.

Lin, Z., Wu, Y., Peri, S. V., Sun, W., Singh, G., Deng, F., Jiang, J., and Ahn, S. SPACE: unsupervised object-oriented scene representation via spatial attention and decomposition. In *International Conference on Learning Representations (ICLR)*, 2020a.

Lin, Z., Wu, Y.-F., Peri, S. V., Sun, W., Singh, G., Deng, F., Jiang, J., and Ahn, S. Space: Unsupervised object-oriented scene representation via spatial attention and decomposition. In *International Conference on Learning Representations (ICLR)*, 2020b.

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., and Kipf, T. Object-centric learning with slot attention. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

Mosbach, M., Niklas Ewertz, J., Villar-Corrales, A., and Behnke, S. SOLD: Reinforcement Learning with Slot Object-Centric Latent Dynamics. In *arXiv preprint arXiv:2410.08822*, 2024.

Nguyen, T., Mansouri, A., Madan, K., Nguyen, K. D., Ahuja, K., Liu, D., and Bengio, Y. Reusable slotwise mechanisms. *Advances in Neural Information Processing Systems (NeurIPS)*, 36, 2024.

Ni, H., Shi, C., Li, K., Huang, S. X., and Min, M. R. Conditional image-to-video generation with latent flow diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Oprea, S., Martinez-Gonzalez, P., Garcia-Garcia, A., Castro-Vargas, J. A., Orts-Escolano, S., Garcia-Rodriguez, J., and Argyros, A. A review on deep learning techniques for video prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research (JMLR)*, 21(140):1–67, 2020.

Rakhimov, R., Volkhonskiy, D., Artemov, A., Zorin, D., and Burnaev, E. Latent video transformer. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, 2021.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252, 2015.

Seitzer, M., Horn, M., Zadaianchuk, A., Zietlow, D., Xiao, T., Simon-Gabriel, C.-J., He, T., Zhang, Z., Schölkopf, B., Brox, T., et al. Bridging the gap to real-world object-centric learning. In *International Conference on Learning Representations (ICLR)*, 2023.

Shridhar, M., Manuelli, L., and Fox, D. CLIPort: What and where pathways for robotic manipulation. In *Conference on Robot Learning (CoRL)*, 2022.

Singh, G., Wu, Y.-F., and Ahn, S. Simple unsupervised object-centric learning for complex and naturalistic videos. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:18181–18196, 2022.

Song, X., Chen, J., Zhu, B., and Jiang, Y.-G. Text-driven video prediction. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2024.

Song, Y.-J., Kim, H., Choi, S., Kim, J.-H., and Zhang, B.-T. Learning object motion and appearance dynamics with object-centric representations. In *Advances in Neural Information Processing Systems Workshops (NeurIPSW)*, 2023.

Tulyakov, S., Liu, M.-Y., Yang, X., and Kautz, J. MoCoGAN: Decomposing motion and content for video generation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Villar-Corrales, A. and Behnke, S. Unsupervised image decomposition with phase-correlation networks. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, 2022.

Villar-Corrales, A. and Behnke, S. Playslot: Learning inverse latent dynamics for controllable object-centric video prediction and planning. *arXiv preprint arXiv:2502.07600*, 2025.

Villar-Corrales, A., Karapetyan, A., Boltres, A., and Behnke, S. MSPred: Video prediction at multiple spatio-temporal scales with hierarchical recurrent networks. *British Machine Vision Conference (BMVC)*, 2022.

Villar-Corrales, A., Wahdan, I., and Behnke, S. Object-centric video prediction via decoupling of object dynamics and interactions. In *IEEE International Conference on Image Processing (ICIP)*, 2023.

Vondrick, C., Pirsiavash, H., and Torralba, A. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

Wang, X., Li, X., Hu, Y., Zhu, H., Hou, C., Lan, C., and Chen, Z. TIV-Diffusion: Towards Object-Centric Movement for Text-driven Image to Video Generation. *arXiv preprint arXiv:2412.10275*, 2024.

Wang, Y., Wu, H., Zhang, J., Gao, Z., Wang, J., Philip, S. Y., and Long, M. PredRNN: A recurrent neural network for spatiotemporal predictive learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(2):2208–2225, 2022.

Watters, N., Matthey, L., Burgess, C. P., and Lerchner, A. Spatial broadcast decoder: A simple architecture for learning disentangled representations in vaes. *arXiv preprint arXiv:1901.07017*, 2019.

Wu, Y.-F., Yoon, J., and Ahn, S. Generative video transformer: Can objects be the words? In *International Conference on Machine Learning (ICML)*, 2021.

Wu, Z., Dvornik, N., Greff, K., Kipf, T., and Garg, A. SlotFormer: Unsupervised visual dynamics simulation with object-centric models. In *International Conference on Learning Representations (ICLR)*, 2023.

Ye, X. and Bilodeau, G.-A. VPTR: Efficient transformers for video prediction. In *International Conference on Pattern Recognition (ICPR)*. IEEE, 2022.

Ye, X. and Bilodeau, G.-A. Video prediction by efficient transformers. *Image and Vision Computing*, 130: 104612, 2023.

Zadaianchuk, A., Seitzer, M., and Martius, G. Object-centric learning for real-world videos by predicting temporal feature similarities. *Advances in Neural Information Processing Systems (NeurIPS)*, 36, 2024.

Zoran, D., Kabra, R., Lerchner, A., and Rezende, D. J. Parts: Unsupervised segmentation with slots, attention and independence maximization. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

A. Limitations and Future Work

While TextOCVP demonstrates promising results for text-guided object-centric video prediction, it presents some limitations that we plan to address in future work:

Prediction Artifacts TextOCVP occasionally generates artifacts in the predicted frames, such as blurriness, inconsistent object appearances, lack of textured details in the objects, or visual artifacts in the background. We believe that these limitations stem from the video rendering module, which might lack the representational power to reconstruct precise image details from the latent space representation.

Poor Temporal Consistency We observe that TextOCVP's predictions often lack temporal consistency when forecasting for long prediction horizons ($T > 30$). We attribute this limitation to the fact that TextOCVP is trained by optimizing reconstruction losses, which do not penalize temporal inconsistencies.

Future Work To address these limitations, we plan to extend our TextOCVP framework with more powerful decoder modules, such as autoregressive transformers (Singh et al., 2022) or diffusion models (Jiang et al., 2023), as well as scale our predictor module. Furthermore, we plan to incorporate temporal discriminators (Clark et al., 2020) to improve the temporal consistency of the predicted video frames. We believe that exploring these architectural modifications will enable us to utilize TextOCVP as a world model in complex real-world robotic environments.

B. Implementation Details

We employ TextOCVP$_{\text{SAVi}}$ for the experiments on CATER and TextOCVP$_{\text{DINO}}$ for experiments on CLIPort. Below we discuss the implementation details for each of these variants.

B.1. TextOCVP$_{\text{DINO}}$

TextOCVP$_{\text{DINO}}$ variant consists of our proposed text-conditioned predictor module and an object-centric decomposition module that extends the DINOSAUR (Seitzer et al., 2023) framework for recursive object-centric video decomposition and video rendering.

Text-Conditioned Predictor The predictor is composed of $N_{\text{Pred}} = 8$ identical layers, each containing 8-head attention mechanisms and an MLP with a single hidden layer of dimension 1024 and a ReLU activation function. Furthermore, the predictor uses an embedding dimensionality of 512, context window size of ten frames, and applies a residual connection from the predictor input to its output.

Text Encoder TextOCVP$_{\text{DINO}}$ leverages a pretrained and frozen small version of T5 encoder (Raffel et al., 2020), which consists of 6 T5 blocks. This text encoder uses a vocabulary with size 32,128.

Scene Parsing The scene parsing module generates $N_{\mathbf{S}} = 10$ slots of dimension 128. As feature extractor, we use DINOv2 ViT-Base (Oquab et al., 2023), featuring 12 layers, using a patch size of 14, and producing patch features with dimension $D_h = 768$. The Slot Attention corrector module processes the first video frame with three iterations in order to obtain a good initial object-centric decomposition, and a single iteration for subsequent frames, which suffices to recursively update the slot representation. The initial object slots $\mathbf{S}_0$ are randomly sampled from a Gaussian distribution with learned mean and covariance. We use a single Transformer encoder block as the transition function, which consists of a four attention heads and an MLP with hidden dimension 512.

Video Rendering The video rendering module consists of two distinct decoders. First, a four-layer MLP-based Spatial Broadcast Decoder (Watters et al., 2019) with hidden dimension 1024 reconstructs the patch features from the slots. Then, to reconstruct the images from the features, we implement a CNN-based decoder with four convolutional layers, where each layer uses 3×3 kernels. Every layer is followed by bilinear upsampling and a ReLU activation function. A final convolutional layer is applied to map to the RGB channels of the image.

Training We train our model for object-centric decomposition using videos sequences of length five frames for 1000 epochs. We use batch size of 16, the Adam optimizer (Kingma & Ba, 2015), and a base learning rate of 4×10^{-4}, which is linearly warmed-up for the first 10000 steps, followed by cosine annealing for the remaining of the training process. Moreover, we clip the gradients to a maximum norm of 0.05. The predictor module is trained given a frozen and pretrained object-centric decomposition model for 700 epochs. The predictor is trained using the same hyper-parameters as for object-centric decomposition. In the predictor loss function $\mathcal{L}_{\text{TextOCVP}}$, we set $\lambda_{\text{Img}} = 1$ and $\lambda_{\text{Slot}} = 1$.

B.2. TextOCVP$_{\text{SAVi}}$

TextOCVP$_{\text{SAVi}}$ uses the same text-conditioned predictor and text-encoder architectures as TextOCVP$_{\text{DINO}}$, but employs SAVi (Kipf et al., 2022) as the object-centric decomposition module.

Scene Parsing The scene parsing module generates $N_{\mathbf{S}} = 8$ slots of dimension 128. Following Kipf et al. (2022), we use as feature extractor a four-layer CNN with ReLU activation function, where each convolutional layer features 32 5×5 kernels, stride $= 1$, and padding $= 2$. The Slot Attention corrector follows the same structure as in TextOCVP$_{\text{DINO}}$.

Video Rendering Following Kipf et al. (2022), we utilize a CNN-based Spatial Broadcast Decoder (Watters et al., 2019) with four convolutional layers with 32 kernels of size 5×5 , stride $= 1$, and padding $= 2$, and a final convolutional layer which maps to four channels (RGB + alpha mask).

Training We train our model for object-centric decomposition using video sequences of length ten frames for 1000 epochs, using batch size of 64, and an initial learning rate of 10^{-4}, which is warmed up for 2500 steps, followed by cosine annealing for the remaining of the training process. Moreover, we clip the gradients to a maximum norm of 0.05. The predictor module is trained given a frozen and pretrained object-centric decomposition model for 1400 epochs. The predictor is trained using the same hyper-parameters as for object-centric decomposition. In the predictor loss function $\mathcal{L}_{\text{TextOCVP}}$, we set $\lambda_{\text{Img}} = 1$ and $\lambda_{\text{Slot}} = 1$.

C. Baselines

We employ four different baselines to compare againts our TextOCVP model on the image-to-video generation task on CATER and CLIPort datasets. To emphasize the importance of incorporating textual information, we include a comparison with OCVP-Seq (Villar-Corrales et al., 2023), a recent object-centric video prediction model that does not utilize text conditioning. Additionally, we evaluate a non-object-centric TextOCVP variant (*Non-OC TF*) that processes the input image into a single high-dimensional slot representation, instead of multiple object-centric slots, thus allowing us to evaluate the effect of object-centric representations. Moreover, we compare TextOCVP with two popular text-conditioned image-to-video generation baselines that do not incorporate object-centricity: MAGE (Hu et al., 2022) and SEER (Gu et al., 2024). We train both models on CATER and CLIPort closely following the original implementation details[1][2].

C.1. MAGE

MAGE is an autoregressive text-guided image-to-video generation framework that utilizes a VQ-VAE (Russakovsky et al., 2015) encoder-decoder architecture to learn efficient visual token representations. A cross-attention module aligns textual and visual embeddings to produce a spatially-aligned motion representation termed Motion Anchor (MA), which is fused with visual tokens via an axial transformer for video generation. For experiments on CATER, we use a codebook size of 512×256 with a downsampling ratio of four, whereas on CLIPort we use a codebook size of 512×1024. Moreover, on CLIPort we replace MAGE's standard CNN encoder and decoder with the DINOv2 ViT encoder and CNN decoder used in our TextOCVP$_{\text{DINO}}$ approach. This adjustment ensures a fair comparison and significantly enhances MAGE's performance on CLIPort. We refer to this modified version as MAGE$_{\text{DINO}}$. Additionally, to align with our experimental setup, we omit the speed parameter.

C.2. SEER

SEER is a diffusion-based model for language-guided video prediction. It employs an Inflated 3D U-Net derived from a pretrained text-to-image 2D latent diffusion model (Rombach et al., 2022), extending it along the temporal axis and integrating temporal attention layers to simultaneously model spatial and temporal dynamics. For the language conditioning module, SEER introduces a novel Frame Sequential Text (FSText) Decomposer, which decomposes global instructions generated by the CLIP text encoder (Radford et al., 2021) into frame-specific sub-instructions. These are aligned with frames using a transformer-based temporal network and injected into the diffusion process via cross-attention layers. We initialize SEER from a checkpoint pretrained on the Something-Something V2 dataset (Goyal et al., 2017), and further fine-tune it for a few epochs. We observed that incorporating a text loss enhanced SEER's performance, while other hyper parameters were kept consistent with its original implementation.

[1] https://github.com/Youncy-Hu/MAGE

[2] https://github.com/seervideodiffusion/SeerVideoLDM/tree/main

D. Datasets

CATER CATER (Girdhar & Ramanan, 2020) is a dataset that consists of long video sequences, each described by a textual caption. The video scenes consist of multiple 3D geometric objects in a 2D table plane, which is split into a 6×6 grid with fixed axis, allowing the exact description of object's positions using coordinates. The text instruction describes the movement of specific objects through four atomic actions: 'rotate', 'pick-place', 'slide', and 'contain'. The caption follows a template consisting of the subject, action, and an optional object or end-point coordinate, depending on the action. The movement of the objects starts at the same time step. Furthermore, the initial positions are randomly selected from the plane grid, and the camera position is fixed for every sequence.

In our work, we employ CATER-hard, which is a complete version of the CATER dataset, containing 30000 video-caption pairs. It includes 5 possible objects: cone, cube, sphere, cylinder, or snitch, which is a special small object in metallic gold color, shaped like three intertwined toruses. Furthermore, every object is described by its size (small, medium, or large), material (metal or rubber), and color (red, blue, green, yellow, gray, brown, purple, cyan, or gold if the object is the snitch), and this description is included in the textual caption. Every atomic action is available. The 'rotate' action is afforded by cubes, cylinders and the snitch, the 'contain' action is only afforded by the cones, while the other two actions are afforded by every object. Every video has between 3 and 8 objects, and two actions happen to different objects at the same time. The vocabulary size is 50.

CLIPort CLIPort (Shridhar et al., 2022) is a robot manipulation dataset, consisting of video-caption pairs, i.e. long videos whose motion is described by a textual video caption. There are many variants of the CLIPort dataset, but we focus on the *Put-Block-In-Bowl* variant. We generate 21000 video-caption pairs with resolution 336×336. Every video contains 6 objects on a 2D table plane, and a robot arm. Objects can be either a block or a bowl, and there is at least one of them in every sequence. The starting position of each object is random, with the only constrain to be placed on the table. Each video describes the action of the robot arm picking a block, and putting it in a specific bowl. The video caption follows the template 'put the [color] block in the [color] bowl'. Each individual object in the scene has a different color. In the train and validation set, the block and the bowl that are part of the caption can have one of the following colors: blue, green, red, brown, cyan, gray, or yellow, while in the test set they can have blue, green, red, pink, purple, white, or orange color. The other 4 objects, called distractors, can have any color. During a video sequence, it can be possible that the robot arm goes out of frame, and comes back in later frames, thus requiring the model to leverage long range dependencies. The vocabulary size is 15.

E. Additional Results

E.1. SAVi vs. DINO

Current object-centric approaches for video generation are limited to relatively simple synthetic datasets. We attribute this limitation primarily to the object-centric modules used for learning object representations. Motivated by this observation, we extend DINOSAUR (Seitzer et al., 2023) to handle video data and reconstruct images effectively.

To demonstrate the significance of the object-centric module in scaling to more complex datasets, we input the same video sequence to both SAVi and our extended DINOSAUR trained on CLIPort. As illustrated in Fig. 8, SAVi struggles to accurately model objects on the 2D plane, whereas our proposed object-centric model successfully reconstructs the scene, closely resembling the input. The visual features extracted by the DINOv2 (Oquab et al., 2023) encoder contain high-level semantic information, and during training, the slots are specifically optimized to efficiently encode this information. This design enables the model to scale and handle more complex object-centric video data effectively.

E.2. Robustness to Number of Objects

We evaluate the performance of TextOCVP and $\text{MAGE}_{\text{DINO}}$ on a CLIPort evaluation set consisting of scenes with a larger number of objects than encountered during training, i.e. the 2D table plane contains 8 instead of 6 objects. This evaluation demonstrates the robustness of models when generalizing to scenes with more objects. The quantitative results are presented in Tab. 5. We report both the video generation performance and the performance drop relative to the original evaluation set.

We observe that our model outperforms $\text{MAGE}_{\text{DINO}}$ on every metric, for both prediction horizons. Notably, TextOCVP demonstrates significantly higher robustness to scenes with more objects, as reflected by the smaller decline in performance

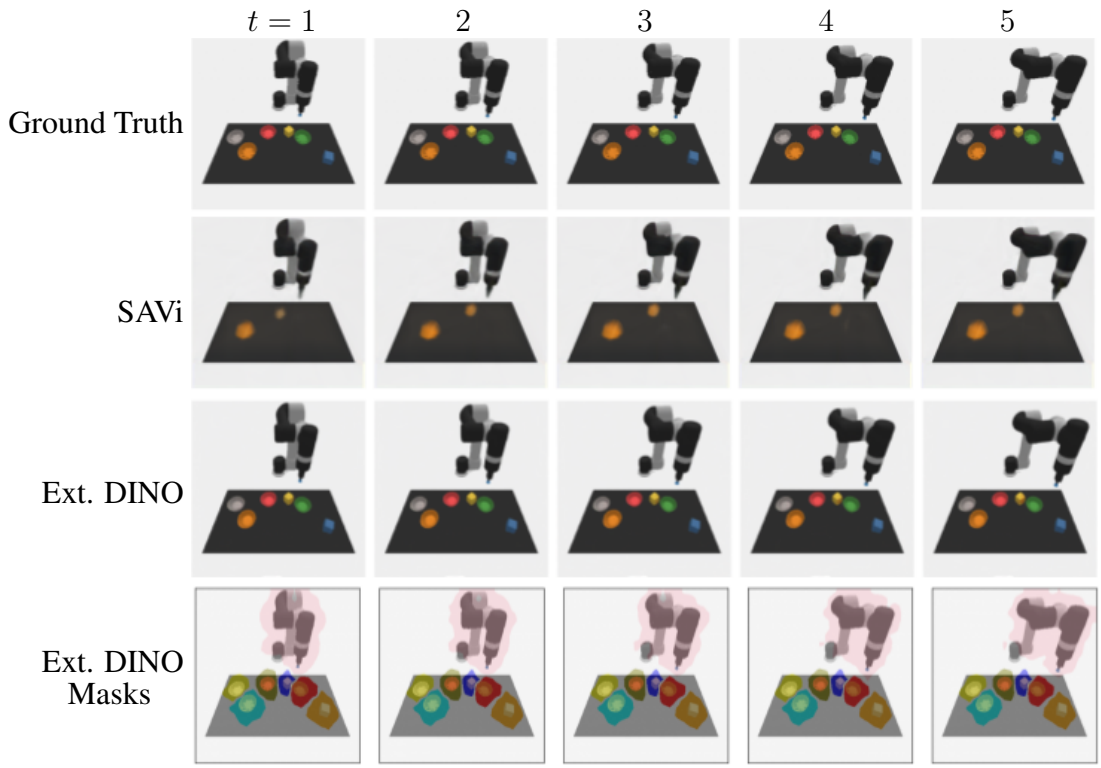

Figure 8: Comparison between SAVi and our Extended DINOSAUR (Ext. DINO) decomposition modules in reconstructing an CLIPort video sequence.

Table 5: Quantitative evaluation on a CLIPort test set with more objects. We report the video generation performance and its drop relative to the original dataset.

	CLIPort$_{1\to9}$		**CLIPort**$_{1\to19}$	
Method	SSIM↑	LPIPS↓	SSIM↑	LPIPS↓
MAGE$_{\text{DINO}}$	0.929 **(-1.2%)**	0.088 (-37.5%)	0.920 (-1.2%)	0.094 (-25.3%)
TextOCVP	**0.936** (-1.5%)	**0.076 (-22.6%)**	**0.921 (-1.1%)**	**0.090 (-15.4%)**

for the perceptual LPIPS metric. This result is further illustrated in Fig. 9, where we show a qualitative comparison of video generations for scenes with 8 objects. As observed, our model correctly illustrates the motion described in the text instructions, while MAGE$_{\text{DINO}}$ fails to generate accurate sequences, missing the target block, a behaviour observed in various examples.

These results highlight once again the effectiveness of object-centric representations in image-to-video generation, as TextOCVP is able to generalize to scenes with more objects by simply increasing the number of slots.

E.3. Text-to-Slot Attention Visualizations

An additional advantage of using object-centric representations is the improved interpretability. This can be shown in the visualizations of text-to slot attention weights, which help us understand how the textual information influences and guides the model predictions. First, in Fig. 10, we visualize the text-to-slot attention weights for different cross-attention heads for a single slot that represents the rotating red cube. We observe that the slot attends to relevant text tokens from the input, such as the object shape, size and the action taking place.

In Fig. 11, we additionally visualize the text-to-slot attention weights, averaged across attention heads, for different slots in a CATER sequence. We observe that slots that represent objects in the textual description attend to relevant text tokens, such as their target coordinate locations. These results demonstrate that the text-to-slot attention mechanism effectively aligns

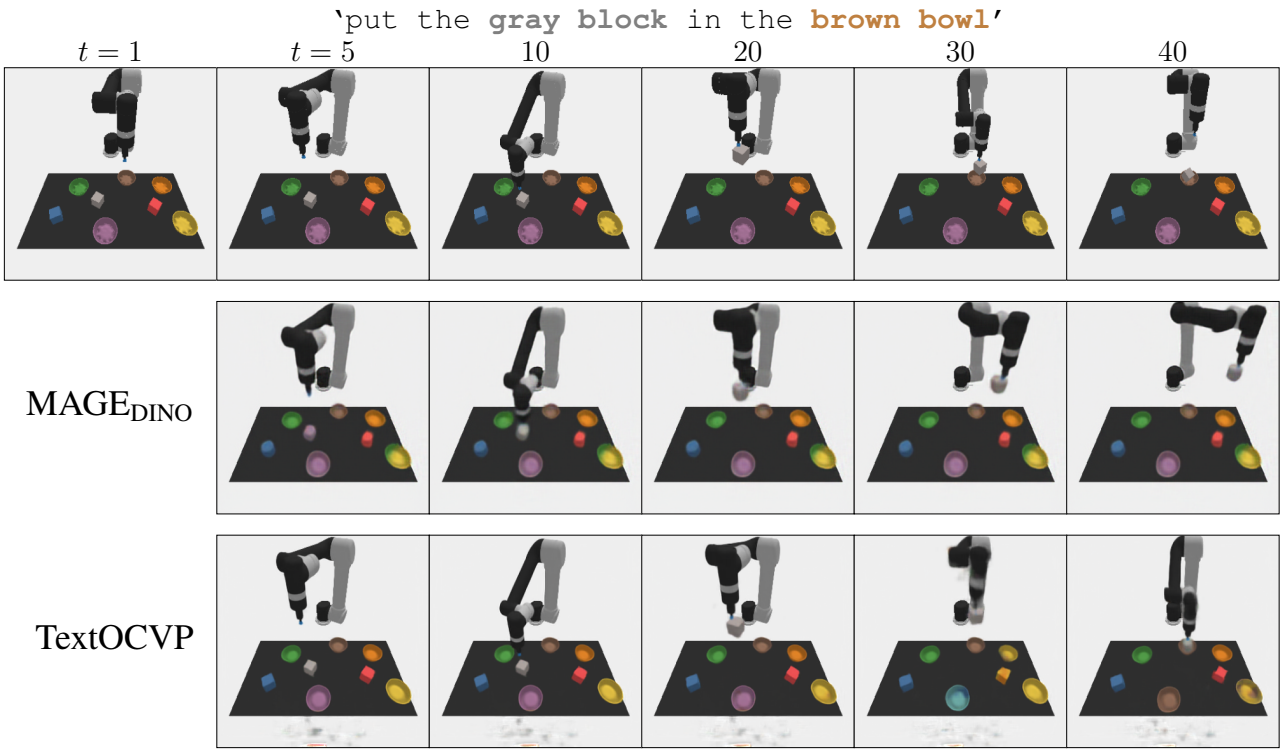

Figure 9: Qualitative evaluation of MAGE$_{\text{DINO}}$ and TextOCVP on a CLIPort sequence with more objects than those seen during training. TextOCVP correctly generates a sequence where the robot picks up the gray block and places it in the brown bowl, while MAGE$_{\text{DINO}}$ fails by missing the target bowl.

the textual information with the object-centric representations, enabling the model to generate accurate video sequences based on the given text instructions, while providing superior interpretability.

E.4. Qualitative Evaluations

Figs. 12 and 13 show qualitative evaluations on CATER in which both MAGE and TextOCVP successfully generate a sequence following the instructions from the textual description. Fig. 14 illustrates an example where MAGE fails to generate a correct sequence, while TextOCVP successfully completes the task described by the text.

Figs. 15 and 16 show examples of TextOCVP's control over the predictions. In both sequences, TextOCVP generates a correct sequence given the text instructions, and seamlessly adapts its generations to a modified version of the textual instructions.

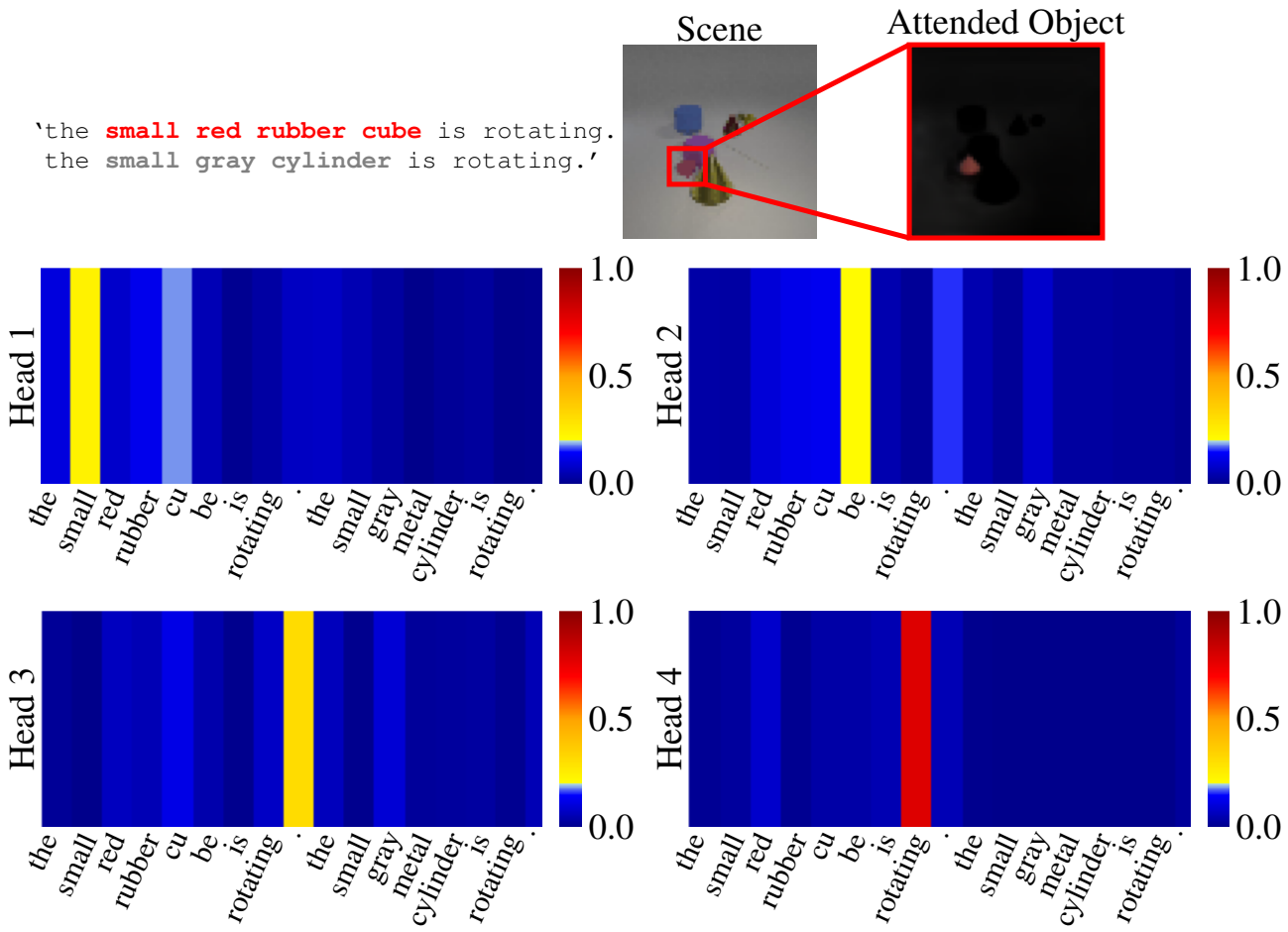

Figure 10: Visualization of the text-to-slot attention between an object and the textual caption. We visualize the text-to-slot attention weights for different heads. Our predictor module focuses on relevant text tokens to predict the scene dynamics, including the object size, shape or the action.

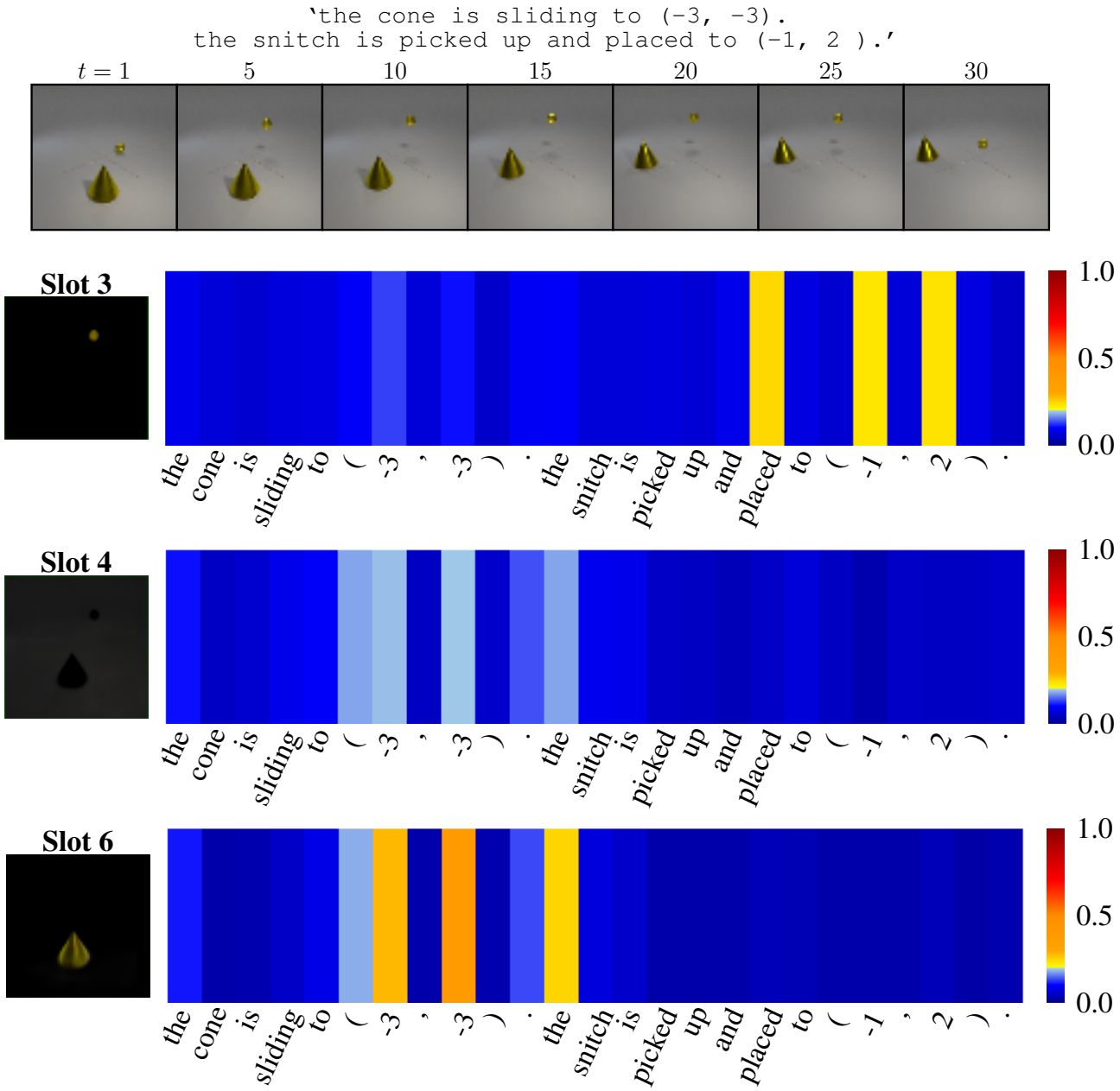

Figure 11: Text-to-slot attention weights, averaged across attention heads, for different objects in the sequence. Slots that represent objects in the textual description attend to relevant text tokens, such as their target locations.

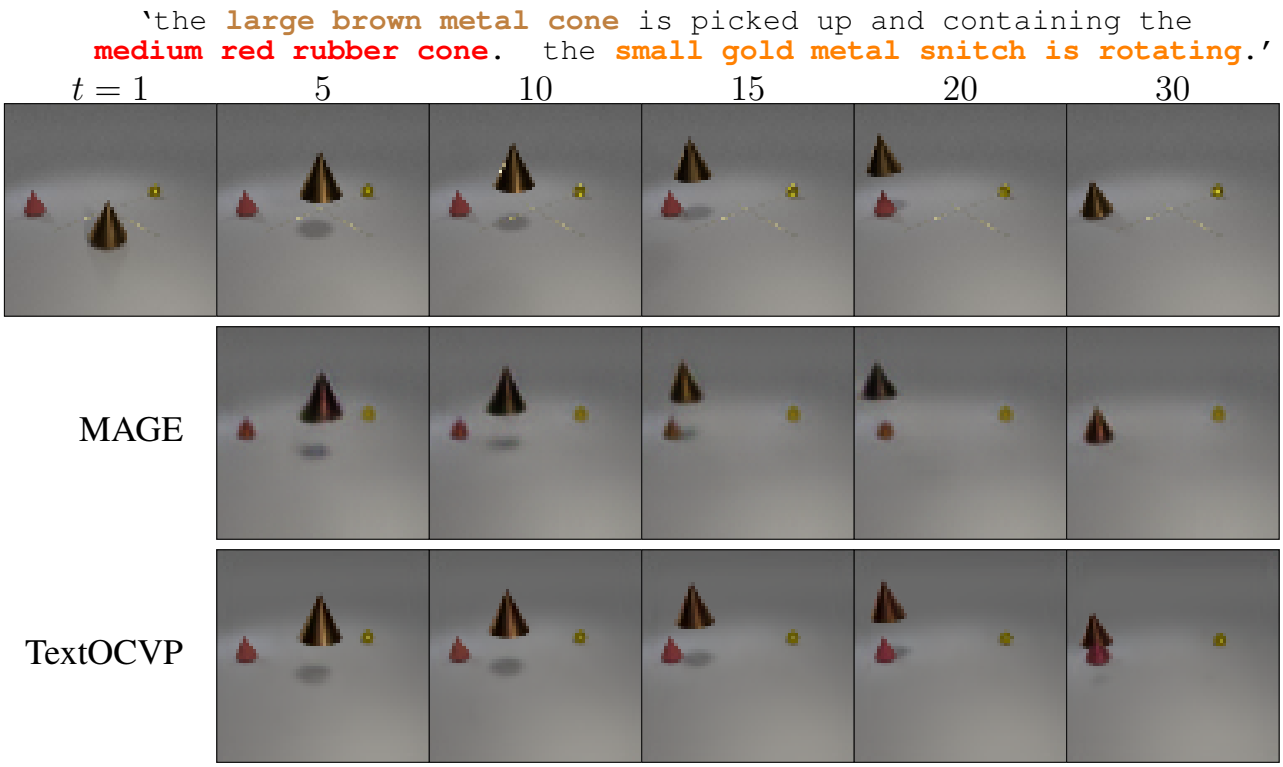

Figure 12: Qualitative evaluation on CATER. Both MAGE and TextOCVP successfully generate a sequence following the instructions from the textual description.

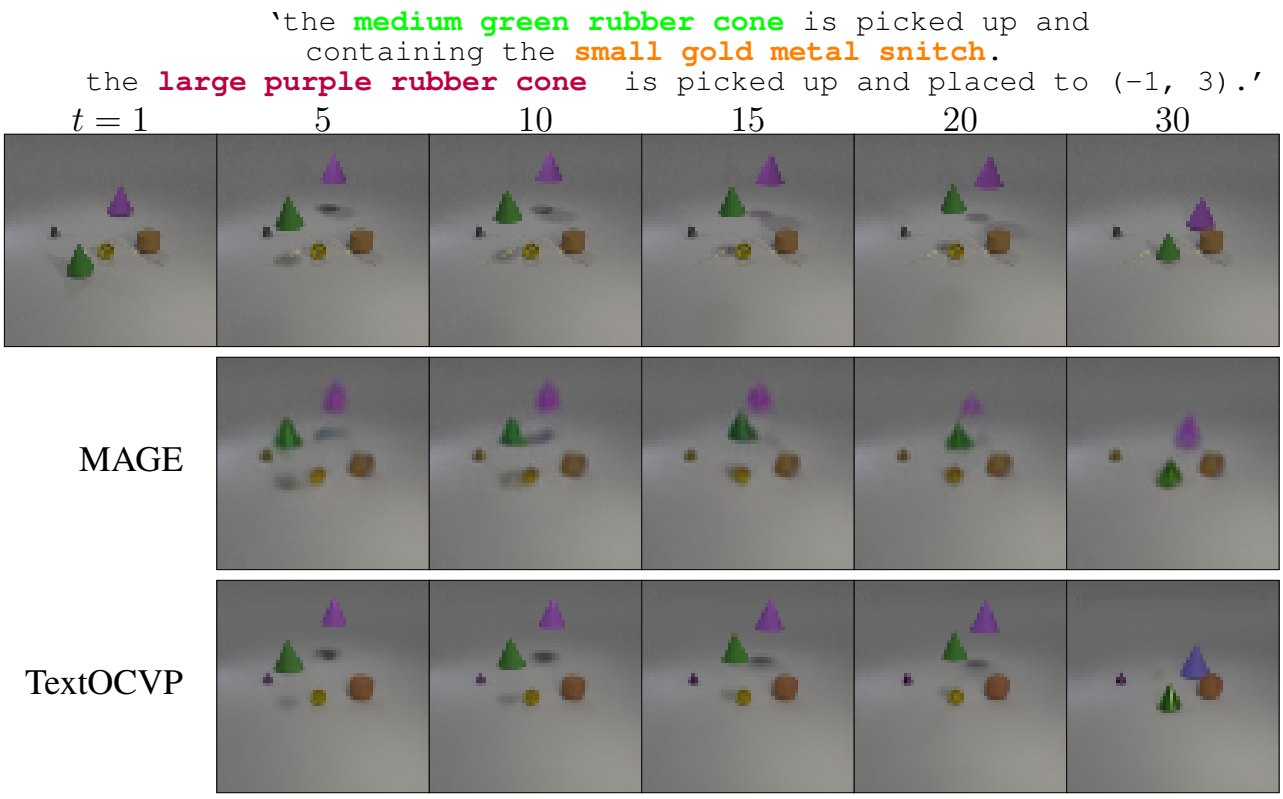

Figure 13: Qualitative evaluation on CATER. Both MAGE and TextOCVP successfully generate a sequence that illustrates the motion described in the text, but MAGE's predictions are of a lower resolution.

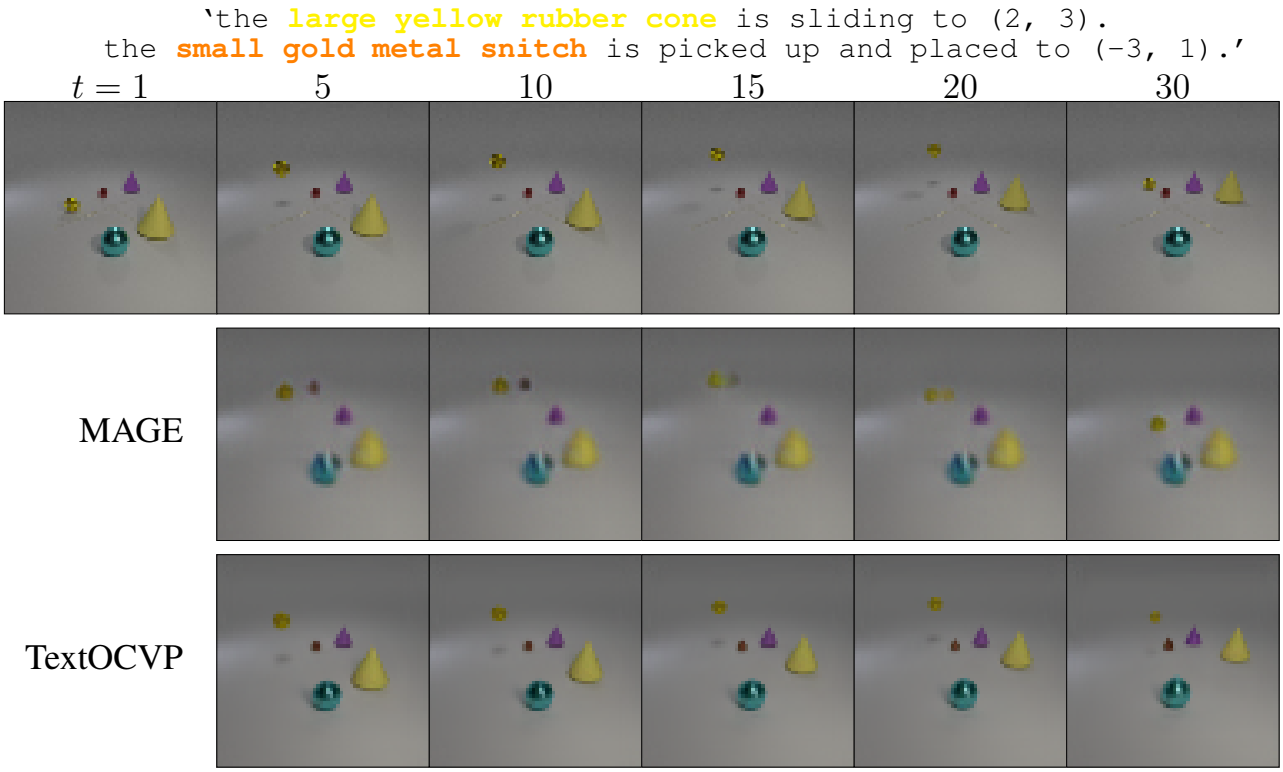

Figure 14: Qualitative evaluation on CATER. MAGE fails to generate a sequence that accurately follows the motion described in the text. Specifically, the yellow cone does not slide as expected, and artifacts such as the merging of two small objects are introduced. On the other hand, the sequence generated by TextOCVP is closely aligned with the ground truth, accurately capturing the motion of the objects.

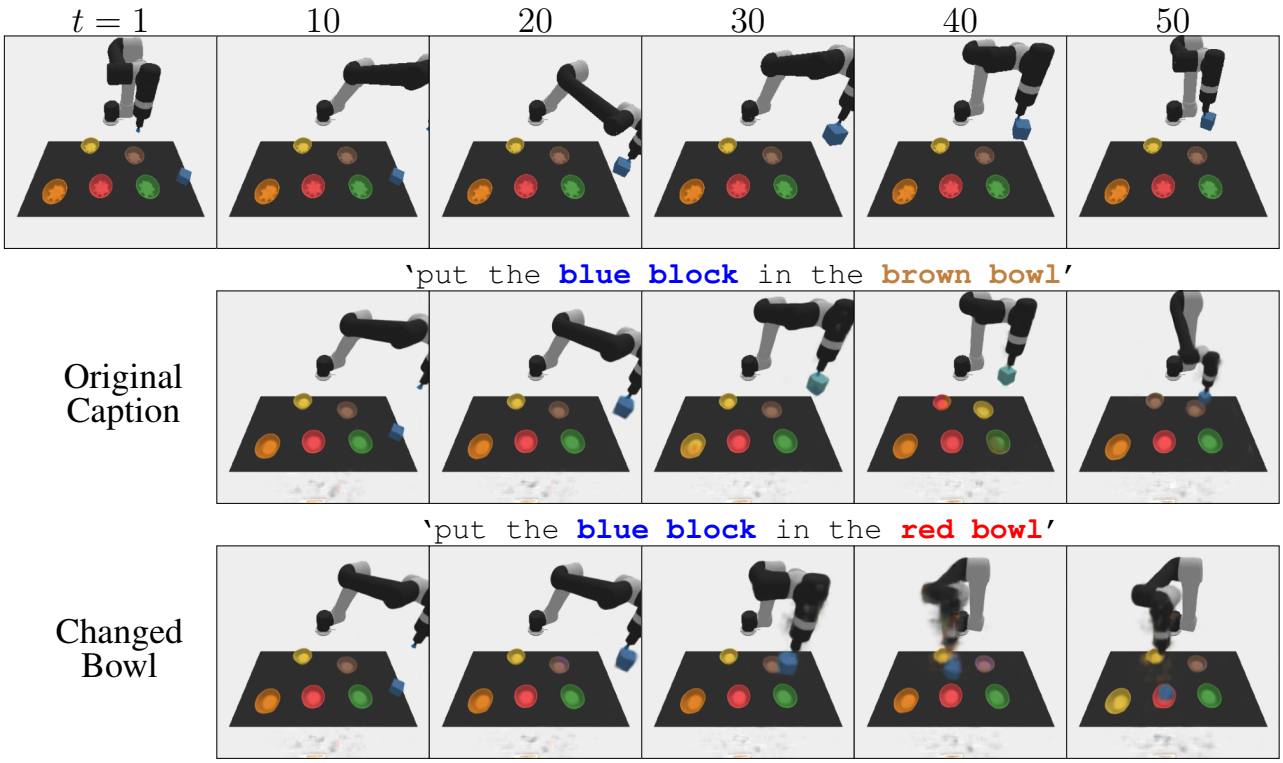

Figure 15: Qualitative evaluation of TextOCVP controllability on CLIPort. TextOCVP correctly generates a sequence where the robot picks up and places the block specified in the textual instruction.

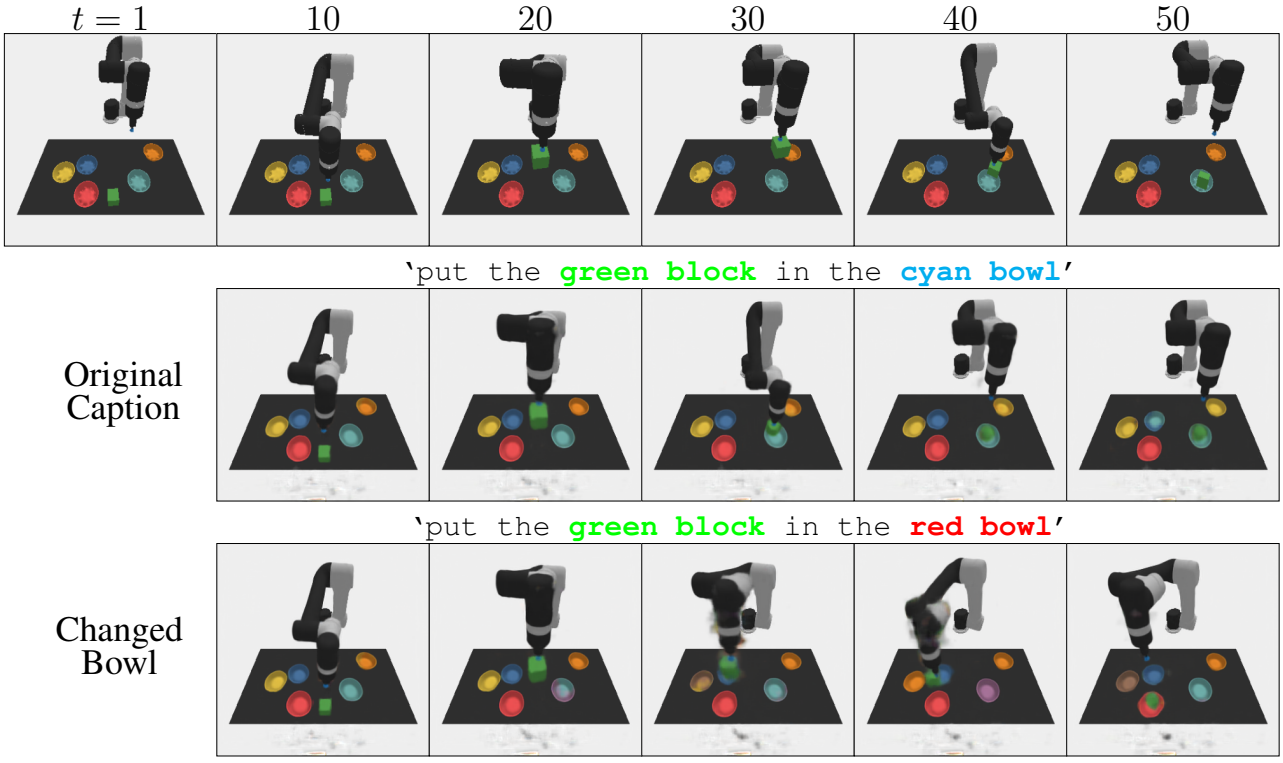

Figure 16: Qualitative evaluation of TextOCVP controllability on CLIPort. TextOCVP correctly generates a sequence where the robot picks up and places the block specified in the textual instruction.