TRIP: SELF-SUPERVISED TEMPORAL REPRESENTATIONS FOR IMAGINATION AND PLANNING

Anonymous authors
Paper under double-blind review

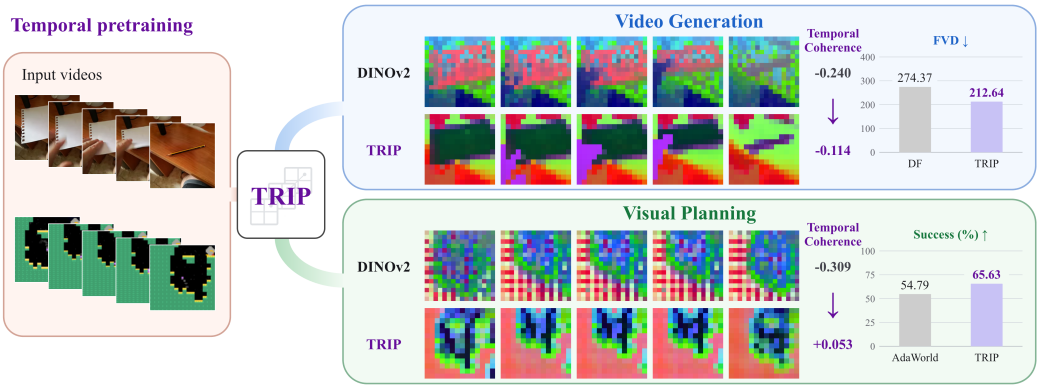


Figure 1: **TRIP: temporal representations for imagination and planning.** TRIP adapts an image-pretrained encoder on downstream videos for both tasks. Colored maps compare frozen DINOv2 and TRIP backbone features from the same frames, with PCA fitted separately per model and clip and fixed across time. Mean temporal coherence (TC) (Wang et al., 2026) is the mean cosine between consecutive displacements in the original high-dimensional features, where higher TC indicates smoother temporal feature trajectories. TRIP features exhibit higher mean TC than frozen DINOv2. TRIP achieves lower Fréchet Video Distance (FVD) than Diffusion Forcing (DF) (Chen et al., 2024) and higher average planning success than AdaWorld (Gao et al., 2025) on Procgen.

ABSTRACT

Visual world models often use frame-wise image representations as prediction targets and to construct planning costs, without explicitly incorporating temporal context into the representations themselves. We introduce TRIP (Temporal Representations for Imagination and Planning), a framework that adapts an image-pretrained visual encoder through causal video self-supervised learning and uses the resulting representations for both future prediction and visual planning. For prediction, a compact bottleneck adapter learns to match the temporal encoder's prototype distributions, providing low-dimensional targets for a diffusion-based feature predictor whose outputs condition RGB video generation. For planning, we reuse the frozen temporal encoder to score imagined rollouts by comparing the goal's prototype distributions with and without rollout context. The resulting goal-representation consistency cost enables model predictive control without additional cost-specific training. TRIP outperforms Diffusion Forcing in video generation on SSv2 and LIBERO and achieves higher average closed-loop planning success than AdaWorld on Procgen. Its scoring encoder also transfers from Procgen to VP^2 without further adaptation, improving planning when paired with an existing world model.

1 INTRODUCTION

Visual agents derive control actions from visual observations in tasks such as robotic manipulation, autonomous navigation, and game control (Bar et al., 2025; Wu et al., 2024; Hafner et al., 2025). To select actions that advance these tasks, agents must consider not only the currently observed

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

scene but also how their actions will change it, particularly when those changes may be irreversible. World models (Bar et al., 2025; Ha & Schmidhuber, 2018; Hafner et al., 2019b;a;b) make this anticipation explicit by predicting the consequences of candidate action sequences before execution. Agents can use these predictions to guide action selection by comparing the predicted states of candidate action sequences and choosing the sequence expected to best achieve the goal, as in model predictive control (MPC) (Zhou et al., 2024; Gao et al., 2025). One promising direction in world modeling (Zhang et al., 2026a; Boduljak et al., 2026) is to predict future states in the feature space of visual foundation models (VFMs) (Assran et al., 2025; Mur-Labadia et al., 2026; Oquab et al., 2023; Siméoni et al., 2025). These models provide semantically rich representations, allowing world models to forecast high-level features rather than reconstruct every detail of raw pixels.

Image-pretrained encoders have already shown strong performance in visual planning (Zhou et al., 2024; Terver et al., 2025). However, some VFM-based world models use prediction targets extracted independently from each frame by an image encoder trained on static images (Zhou et al., 2024; Baldassarre et al., 2025; Zhang et al., 2026a; Karypidis et al., 2026). These targets capture the visual content of individual frames without explicitly encoding temporal context, leaving the predictor to learn dynamics across frame-wise representations. Building on the strengths of image-pretrained encoders, we investigate how adapting their representations to the temporal dynamics of downstream environments can better support both future prediction and planning.

For planning, MPC commonly evaluates an imagined rollout either by comparing only the predicted endpoint with the goal in feature space (Zhou et al., 2024; Assran et al., 2025) or by aggregating frame-wise visual costs over the predicted trajectory (Gao et al., 2025; Wu et al., 2024; Tian et al., 2023). Both formulations compare predicted frame features with goal-image features, using an image encoder that processes each frame independently. A temporal encoder offers another design: the goal representation can be conditioned on the candidate rollout. We investigate whether an encoder adapted for temporal representation learning can also provide a useful planning cost without additional cost-specific training.

We introduce TRIP (Temporal Representations for Imagination and Planning), a framework that adapts an image-pretrained visual encoder to downstream video clips through self-supervised learning with causal temporal attention. For future prediction, we distill the learned features into compact targets for a diffusion predictor whose outputs condition RGB generation. For planning, we reuse the frozen temporal encoder to compare the goal's prototype distributions with and without rollout context. This yields a goal-representation consistency cost without additional cost-specific training. Figure 1 summarizes the resulting improvements in temporal feature coherence and performance on both video prediction and visual planning.

We evaluate TRIP on video prediction and closed-loop visual planning. It improves prediction quality over the evaluated world-model baselines on human-object interaction and robot-manipulation videos, with slower degradation over longer sequences on SSv2 (Goyal et al., 2017). TRIP improves closed-loop planning success on Procgen games (Cobbe et al., 2019). After validating the proposed cost on Procgen, we directly transfer the scoring encoder to VP^2 (Tian et al., 2023) without further adaptation and surprisingly find that it also improves planning with an existing world model.

Our contributions are summarized as follows:

- **Temporal representations for prediction.** We fine-tune an image-pretrained encoder on downstream video clips with causal temporal attention and self-supervised objectives, yielding temporally contextualized targets for future prediction.
- **Goal-representation consistency scoring for planning.** We compare the goal's prototype distributions with and without rollout context, reusing the frozen temporal encoder without additional cost-specific training.
- **Experimental validation.** Experiments on video prediction and visual planning demonstrate the effectiveness of TRIP across diverse environments.

2 Related Work

2.1 Visual World Models

A line of work on visual world modeling learns dynamics through generative modeling of future visual observations (Wu et al., 2024; Bar et al., 2025; Alonso et al., 2024; Hafner et al., 2023).

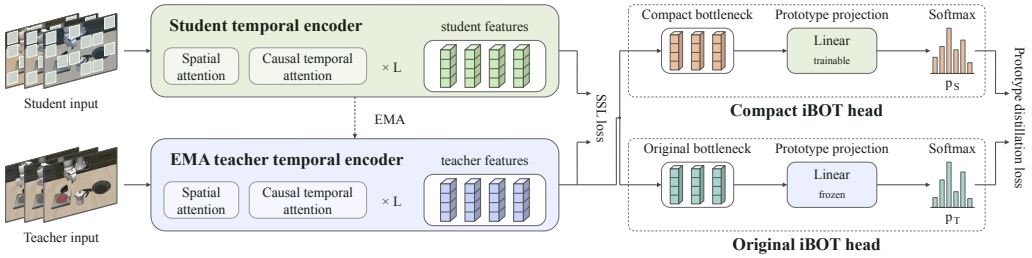


Figure 2: **Temporal and prototype distillation.** Left: temporal self-supervised learning. Right: iBOT prototype distillation. LIBERO frames illustrate input augmentations. The compact branch learns to match the frozen teacher's prototype distribution to provide prediction targets.

iVideoGPT (Wu et al., 2024) autoregressively predicts discrete video tokens, whereas NWM (Bar et al., 2025), IRASim (Zhu et al., 2025), and Vid2World (Huang et al., 2025) employ diffusion-based video prediction conditioned on actions. To leverage videos without action annotations, AdaWorld (Gao et al., 2025), CoLA-World (Wang et al., 2025), and DreamDojo (Gao et al., 2026) infer latent actions from videos and use them to condition future observation prediction during training.

Other methods predict features from pretrained visual encoders. DINO-WM (Zhou et al., 2024) and DINO-Foresight (Karypidis et al., 2026) forecast DINO features, while VFMF (Boduljak et al., 2026) predicts compressed DINOv2 latents. RAE-NWM (Zhang et al., 2026a) predicts dense DINOv2 features for navigation and decodes them into images. These methods use frame-wise image encoders to extract prediction targets.

2.2 Pretrained Visual Representations

Self-supervised image encoders such as iBOT (Zhou et al., 2021), DINOv2 (Oquab et al., 2023), and DINOv3 (Siméoni et al., 2025) learn dense features but lack temporal context when applied frame by frame. LingBot-Vision (Fu et al., 2026) introduces masked boundary modeling to learn spatially structured visual representations for dense perception. R3M (Nair et al., 2022) learns transferable visual representations from human videos for downstream robot manipulation. Video methods capture temporal structure through masked reconstruction (VideoMAE (Tong et al., 2022)), latent prediction (V-JEPA 2 (Assran et al., 2025), V-JEPA 2.1 (Mur-Labadia et al., 2026)), temporal self-distillation (Temporal-DINO (Teeti et al., 2023)), representation restoration (T-CoRe (Liu et al., 2025)), or DINO patch prediction (FRAME (TV et al., 2025)).

2.3 Visual Model Predictive Control

Visual model predictive control (MPC) evaluates candidate action sequences with a world model and replans after executing an initial action segment (Schwenzer et al., 2021; Zhou et al., 2024; Hansen et al., 2024; 2022; Finn & Levine, 2017; Ebert et al., 2018). Existing visual goal costs compare predicted endpoints (Bar et al., 2025; Zhou et al., 2024) or individual predicted frames (Gao et al., 2025; Wu et al., 2024) with goal images using pixel or feature distances (Ma et al., 2022; 2023). PLDM learns latent dynamics from reward-free offline data and plans using cumulative latent distances to the goal (Sobal et al., 2026). Temporal Straightening (Wang et al., 2026) improves feature-distance objectives by regularizing latent trajectory curvature, while TRIP conditions the goal representation on each candidate rollout to construct a goal-representation consistency cost.

3 Method

As illustrated in Figure 2, TRIP adapts a DINOv2 (Oquab et al., 2023) backbone to downstream videos through self-supervised learning (Section 3.1). For prediction, a distilled bottleneck adapter provides compact targets for predicting future states and generating RGB videos (Section 3.2). For planning, we reuse the frozen temporal encoder to jointly encode imagined RGB rollouts and visual goals for model predictive control (MPC) (Section 3.3).

3.1 TEMPORAL DINO ENCODER

DINOv2 preliminaries. DINOv2 (Oquab et al., 2023) trains a student encoder f_θ with a teacher encoder $f_{\bar{\theta}}$, whose parameters are an exponential moving average of the student's. The student receives two global and multiple local views per image, and the teacher receives only unmasked global views. DINOv2 supervises the student primarily through two losses: the DINO loss (Caron et al., 2021) matches global-token prototype distributions across different views of the same image, while the iBOT loss (Zhou et al., 2021) matches prototype distributions at masked student patches to those at corresponding unmasked teacher patches. These prototype distributions are obtained from scores produced by projection heads. In particular, the iBOT head maps each patch feature to a low-dimensional bottleneck representation and then projects it to prototype scores (logits):

$$v = h_{\text{bottleneck}}(u), \qquad s = h_{\text{proto}}(v). \tag{1}$$

Here u is an encoded patch feature, $h_{\text{bottleneck}}$ maps it to the compact representation v, and h_{proto} maps v to the vector of prototype scores s. The prototype scores are converted into student probability distributions using temperature-scaled softmax and teacher target distributions using sharpening and Sinkhorn–Knopp normalization (Caron et al., 2020). We denote the resulting teacher and student distributions by p_t, p_s for global tokens and $q^T_{b,i}, q^S_{b,i}$ for patches at spatial location i in global view b. The objectives are

$$\mathcal{L}_{\text{DINO}} = \frac{1}{|\mathcal{P}|} \sum_{(p_t,p_s)\in\mathcal{P}} H(p_t, p_s), \qquad \mathcal{L}_{\text{iBOT}} = \frac{1}{B_g} \sum_{b\in\mathcal{I}} \frac{1}{|\mathcal{M}_b|} \sum_{i\in\mathcal{M}_b} H(q^T_{b,i}, q^S_{b,i}), \tag{2}$$

where H denotes cross-entropy. For the DINO loss, $\mathcal{P}$ contains all same-image teacher–student view pairs in the batch, excluding pairs from the same view. For the iBOT loss, B_g counts all student global views, and $\mathcal{I}$ indexes those containing masked patches. Within each such view b, $\mathcal{M}_b$ contains the masked patch indices, with $|\mathcal{M}_b|$ denoting their number.

Temporal adaptation. For future prediction, we construct representations that incorporate information from past frames. We extend the DINOv2 backbone with causal temporal attention, allowing each frame representation to incorporate information from past frames. Each spatial self-attention layer is followed by a temporal attention layer.

To prepare this temporal encoder for video training, we initialize its patch-level prediction head and adjust its input resolution and patch size. For head initialization, we first fine-tune DINOv2 on ImageNet (Deng et al., 2009) and use the resulting iBOT head weights. We use an input resolution of 256×256 and adapt the backbone to a patch size of 16×16 pixels.

With these initialization weights, we fine-tune the temporal encoder on video clips from the downstream environment. Each clip is augmented into two global views and multiple local views. All frames within a view share the same crop, resize operation, and appearance augmentation parameters, including color jitter. We train on these augmented views using video adaptations of the DINO and iBOT objectives. The DINO objective matches predictions for the same frame across different views. The set $\mathcal{P}$ contains these teacher–student prediction pairs across all clips and frames in a mini-batch. For the iBOT objective, we apply tube masking (Tong et al., 2022) to these views, using the same spatial mask across all frames within each view. This shared mask prevents the student from directly observing the masked locations in adjacent frames.

Extending patch-level supervision to unmasked patches can improve feature map smoothness during fine-tuning. Following iBOT++ from TIPSv2 (Cao et al., 2026), the patch-level prediction loss therefore covers both masked and unmasked patches.

3.2 GENERATION PIPELINE

Compact feature distillation. The trained temporal encoder provides the features used by our generation pipeline. Given the observed frames $x_{\leq t}$, the frozen encoder extracts a spatial feature map $u_t = f_{\bar{\theta}}(x_{\leq t})$ for frame t that incorporates information from past frames.

Although DINO patch features could be used directly as prediction targets, prior work on image generation and feature forecasting recommends using compact representations for modeling high-dimensional visual features (Zhang et al., 2025; Boduljak et al., 2026). To obtain compact prediction

targets, we build on the bottleneck structure of the teacher's iBOT projection head (Zhou et al., 2021; Oquab et al., 2023). However, we find that the original bottleneck features are suboptimal as prediction targets. We therefore use a narrower bottleneck, which improves video generation from predicted features compared with the original bottleneck (Table 5).

We obtain compact prediction targets by training a bottleneck adapter to match the frozen teacher's prototype distributions, without an additional VAE. Only the adapter is optimized, using the cross-entropy loss:

$$\mathcal{L}_{\text{adapter}} = -\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \sum_{k=1}^{K} p_{\text{T}}^{(i,k)} \log p_{\text{A}}^{(i,k)}, \tag{3}$$

where $\mathcal{I}$ indexes all supervised patch tokens in the mini-batch, K is the number of prototypes, and $p_{\text{T}}^{(i,k)}$ and $p_{\text{A}}^{(i,k)}$ are the teacher and adapter probabilities for prototype k at token i.

Semantic feature generator. Using the compact representations obtained through feature distillation, we train a semantic feature generator with Diffusion Forcing (Chen et al., 2024) to predict future compact feature maps conditioned on the observed feature history and conditioning inputs.

Given the observed semantic feature history $z_{\leq t}$ and action condition c, the semantic feature generator samples future features as $\hat{z}_{t+1:t+H} \sim p_\phi(z_{t+1:t+H} \mid z_{\leq t}, c)$, where H is the prediction horizon. The action condition c is provided as an action sequence $a_{t:t+H-1}$.

Appearance generator. To translate the predicted semantic features into RGB video, we train an appearance generator following PiD (Lu et al., 2026) and Latent Forcing (Baade et al., 2026). The generator operates in the latent space of a frozen Wan VAE (Wan et al., 2025). Given the VAE latents $y_{\leq t}$ of the observed prefix, it samples future latents:

$$\hat{y}_{t+1:t+H} \sim p_\psi(y_{t+1:t+H} \mid y_{\leq t}, \hat{z}_{t+1:t+H}) \,. \tag{4}$$

To improve robustness to semantic prediction errors, we follow PiD (Lu et al., 2026) and train the appearance generator with noisy features $\tilde{z}_\sigma = (1-\sigma)z + \sigma\epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$ and $\sigma \sim \mathcal{U}(0, 0.8)$ in all experiments. At inference, we condition on predicted features $\hat{z}$. At the same number of function evaluations (NFE), our pipeline's FVD grows more slowly with sequence length than that of RGB Diffusion Forcing (Figure 4).

3.3 Model Predictive Control

MPC preliminaries. Visual model predictive control (MPC) (Zhou et al., 2024; Maes et al., 2026; Gao et al., 2025) selects actions by evaluating future trajectories predicted by a world model. Given the observation history $o_{\leq t}$ and a candidate action sequence $\mathbf{a} = a_{t:t+H-1}$ over a planning horizon H, a world model F predicts a rollout:

$$\hat{o}_{t+1:t+H} = F(o_{\leq t}, \mathbf{a}). \tag{5}$$

This predicted trajectory is an imagined rollout: it simulates the candidate actions' consequences without executing them in the environment. Each predicted rollout is assigned a cost $J(\mathbf{a})$ measuring its compatibility with a visual goal o_g. Common costs compare the predicted endpoint with the goal in a pretrained visual feature space (Zhou et al., 2024), and a rollout can be scored by averaging the cosine similarity between each predicted frame's features and the goal's features (Gao et al., 2025), with one minus this average used as the cost. MPC uses these costs to optimize an action sequence and executes its initial segment. After execution, it incorporates the new observations and repeats the planning process.

Goal-representation consistency scoring. We reuse the frozen temporal encoder to construct a planning cost without learning an additional reward model. We append the goal image o_g to each predicted RGB rollout and read out the prototype distributions at the goal position. Through causal temporal attention, this readout can access the preceding predicted frames. Let $\mathcal{E}$ denote the mapping from image sequences to patch-wise prototype distributions, combining the temporal DINO encoder, its original iBOT head, and softmax normalization. The contextualized goal distributions are

$$q_g^{\text{ctx}}(\mathbf{a}) = \left[\mathcal{E}(\hat{o}_{t+1}, \ldots, \hat{o}_{t+H}, o_g)\right]_{H+1} , \tag{6}$$

Table 1: SSv2 and LIBERO generation comparison. FVD measures distributional similarity to real videos (lower is better), and ECS measures frame-level cosine similarity using I3D (Carreira & Zisserman, 2017) (higher is better).

Method	SSv2 FVD (↓)	SSv2 ECS (↑)	LIBERO FVD (↓)	LIBERO ECS (↑)
Diffusion Forcing	274.37	0.820	101.48	0.924
AdaWorld	594.58	0.616	476.25	0.625
DiLA	459.24	0.751	902.37	0.519
TRIP	**212.64**	**0.875**	**88.89**	**0.926**

Table 2: Visual planning results in games. For all selected scenes from four Procgen environments, we report the success rate (%) and standard error over four random seeds.

Method	**Heist**	**Jumper**	**Maze**	**CaveFlyer**	**Average**
Random	23.33 ± 4.08	18.33 ± 3.97	43.33 ± 5.61	19.17 ± 2.85	26.04
DINO-WM	45.00 ± 4.41	20.00 ± 2.72	30.00 ± 4.51	44.17 ± 1.60	34.79
LeWorldModel	64.17 ± 2.10	27.50 ± 2.85	57.50 ± 2.50	46.67 ± 1.36	48.96
AdaWorld	**70.00** ± 1.36	60.00 ± 4.51	57.50 ± 2.85	31.67 ± 0.96	54.79
Diffusion Forcing	66.67 ± 4.08	55.83 ± 4.38	56.67 ± 2.72	60.00 ± 1.36	59.79
TRIP	**70.00** ± 2.36	**60.83** ± 3.70	**66.67** ± 1.36	**65.00** ± 2.89	**65.63**

where $[\cdot]_j$ selects the patch-wise prototype distributions at sequence position j. We encode the goal image alone as $q_g^{\text{clean}} = \mathcal{E}(o_g)$ and use it as the same reference for all candidate rollouts. Using this reference as the target, we define the planning cost as the average patch-wise cross-entropy:

$$J(\mathbf{a}) = \frac{1}{P} \sum_{i=1}^{P} H\left(q_{g,i}^{\text{clean}}, q_{g,i}^{\text{ctx}}(\mathbf{a})\right), \tag{7}$$

where P is the number of spatial patches, i indexes corresponding patch locations, and H denotes cross-entropy. This defines a goal-representation consistency cost in the encoder's existing prototype space. The comparison is between two encodings of the same goal image, obtained with and without the predicted rollout as context. We choose cross-entropy to reuse the prototype-distribution matching form used in self-supervised training. Both the temporal encoder and its iBOT head remain frozen during scoring, so no additional cost-specific training is required. MPC uses this cost to optimize the action sequence.

4 Experiments

4.1 Experimental Setup

Benchmarks. Following the evaluation setting of AdaWorld (Gao et al., 2025), we evaluate video generation on Something-Something V2 (SSv2) (Goyal et al., 2017) and LIBERO (Liu et al., 2023), and visual planning on four Procgen games (Cobbe et al., 2019) and six tasks from the VP^2 (Tian et al., 2023) benchmark.

Metrics. For video prediction, we report Fréchet Video Distance (FVD) (Unterthiner et al., 2018) and embedding cosine similarity (ECS) (Sun et al., 2024). For visual planning, we report closed-loop task success rate.

Baselines. To evaluate video generation quality, we compare against autoregressive world models AdaWorld (Gao et al., 2025) and DiLA (Zhang et al., 2026b), as well as Diffusion Forcing (Chen et al., 2024), a diffusion model that supports autoregressive video generation.

For visual planning, we follow the evaluation settings of AdaWorld (Gao et al., 2025) and iVideoGPT (Wu et al., 2024), comparing against a random policy and AdaWorld on Procgen, and iVideoGPT, AdaWorld, and DiLA (Zhang et al., 2026b) on VP^2. We additionally include Diffusion Forcing (Chen et al., 2024) as a planning baseline on Procgen. To assess DINO-WM (Zhou et al., 2024) and LeWorldModel (Maes et al., 2026) in complex dynamic environments, we train both models on Procgen using their official training scripts.

Implementation details. For video generation, we follow the evaluation settings of AdaWorld (Gao et al., 2025) and evaluate 17-frame video clips on both Something-Something V2 (SSv2)

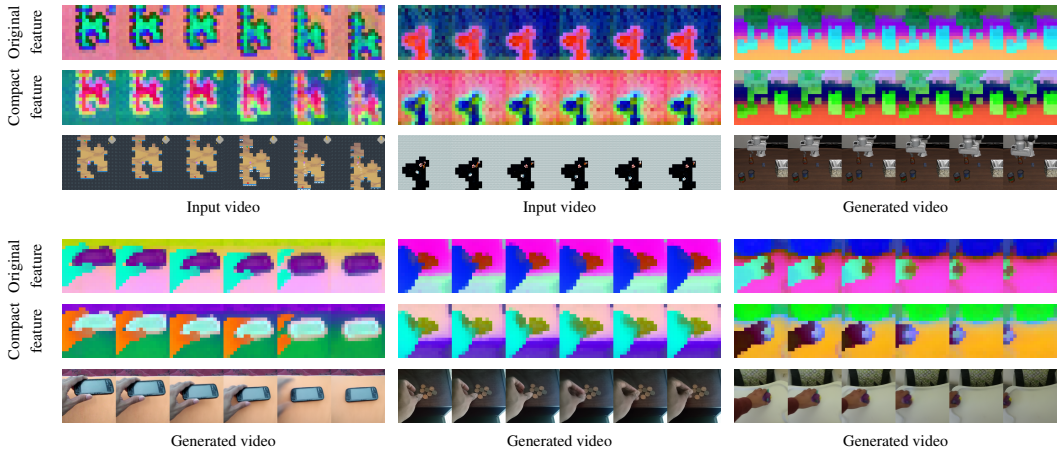


Figure 3: **Original and compact representations of input and generated videos.** Each example shows six RGB frames alongside PCA visualizations of our temporal encoder's original 1024-dimensional features and the compact features obtained through distillation. Top: Procgen input videos (Jumper and CaveFlyer) and a generated LIBERO video. Bottom: generated SSv2 videos.

and LIBERO. Diffusion Forcing uses 20 function evaluations (NFE). Our pipeline also uses 20 NFE, allocating 4 to the semantic feature generator and 16 to the appearance generator. AdaWorld uses 20 sampling steps, while DiLA follows the inference settings of its original paper (Zhang et al., 2026b). Both our model and the Diffusion Forcing (Chen et al., 2024) baseline are initialized from Wan2.1 (Wan et al., 2025) weights, except for the patch embedding and output layers, which are randomly initialized. We train a 128-channel bottleneck adapter to match the frozen temporal encoder's original iBOT prototype distributions and use its compact features as semantic prediction targets.

For visual planning on Procgen (Cobbe et al., 2019), since AdaWorld (Gao et al., 2025) does not release its evaluation scenes, we generate scenes using its official scripts. All methods are evaluated on the same scenes with matched random seeds. For each Procgen game, we use a separately trained RGB Diffusion Forcing world model to generate candidate future RGB rollouts. We append the goal image to each rollout and encode the resulting sequence with our frozen temporal DINO encoder and its original iBOT head to compute the TRIP goal-representation consistency cost (Section 3.3). On VP^2, we follow iVideoGPT's visual-planning protocol (Wu et al., 2024), using its world model and MPC settings with our goal-representation consistency cost. The temporal DINO encoder transfers from Procgen without further training.

4.2 Experimental Results

Video prediction. Table 1 compares TRIP with Diffusion Forcing, AdaWorld, and DiLA on SSv2 and LIBERO. TRIP achieves the lowest FVD and highest ECS on both benchmarks. Compared with Diffusion Forcing, the strongest baseline, TRIP reduces FVD from 274.37 to 212.64 and increases ECS from 0.820 to 0.875 on SSv2. On LIBERO, it reduces FVD from 101.48 to 88.89 and increases ECS from 0.924 to 0.926.

Degradation with video length. Surprisingly, TRIP also exhibits slower degradation as the rollout length increases. Figure 4 shows that, from 10 to 17 frames, its fitted FVD growth rate is lower than that of RGB Diffusion Forcing (7.75 versus 11.95 FVD/frame). One possible explanation is that predicting dynamics in a compact semantic space provides structured guidance for the subsequent RGB generation stage, reducing its burden of modeling motion and appearance jointly.

Procgen planning. Table 2 reports planning success rates on four Procgen games across four evaluation seeds. TRIP achieves the highest average success rate of 65.63%, outperforming Diffusion Forcing (59.79%), AdaWorld (54.79%), LeWorldModel (48.96%), DINO-WM (34.79%), and Random (26.04%). This represents a gain of 5.84 percentage points over Diffusion Forcing. It leads on Jumper, Maze, and CaveFlyer and ties with AdaWorld on Heist. The lower performance of DINO-WM and LeWorldModel may partly stem from deterministic prediction. These models learn

Table 3: Visual planning results on the VP^2 benchmark. iVideoGPT, AdaWorld, and DiLA are published results; TRIP is evaluated in our own control runs. Task success rates (%) are reported as mean ± standard error. The final column averages success rates across the six tasks after normalization by the corresponding simulator success rates.

Method	RoboSuite Push	Open Slide	Blue Button	Green Button	Red Button	Upright Block	Norm. aggregate
iVideoGPT	78.33 ±0.60	16.11 ±1.95	**95.56** ±1.47	82.50 ±2.41	**92.22** ±1.11	44.72 ±1.21	73.22
AdaWorld	63.50 ±1.71	5.83 ±2.85	29.17 ±2.50	10.83 ±2.50	10.00 ±2.36	5.00 ±0.96	22.18
DiLA	68.00 ±1.41	15.00 ±5.00	78.33 ±3.73	35.83 ±4.93	20.83 ±5.95	3.33 ±2.72	39.32
TRIP	**85.33** ±2.40	**21.11** ±2.94	94.44 ±1.11	**95.56** ±1.11	88.89 ±1.11	**66.67** ±5.09	**81.14**

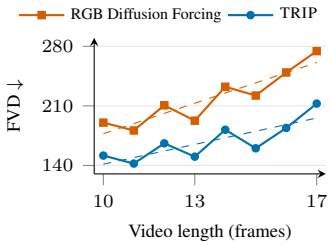
Table 4: Video prediction on SSv2 with different DINOv2 encoder variants.

Variant	FVD ↓	ECS ↑
Image-only	262.05	0.828
Frame-wise	255.49	0.841
Shuffled temporal	220.76	0.864
Ordered temporal	**212.64**	**0.875**

Table 5: Feature compression for video generation on SSv2.

Feature	Dim.	FVD_{GT} ↓	$FVD_{Pred.}$ ↓
Teacher	256	**74.77**	245.70
PCA	128	97.39	223.07
VAE	128	90.57	224.43
Prototype	128	84.23	**212.64**

Figure 4: FVD versus video length for RGB Diffusion Forcing and TRIP on SSv2 videos.



next-state feature prediction with squared-error regression under teacher forcing, conditioning on features from observed histories. When multiple futures are plausible, regression may average their features rather than represent a realizable outcome (Boduljak et al., 2026). During planning, the models instead condition on their own predictions, which can shift subsequent contexts away from the training distribution and compound errors over the rollout.

VP^2 planning. Using the iVideoGPT world model and planning protocol, TRIP achieves a simulator-normalized aggregate of 81.14 across six tasks, compared with 73.22 for the published iVideoGPT results (Table 3). The scoring encoder transfers from Procgen without adaptation.

4.3 Ablation Studies

Temporal representation. Table 4 compares video prediction with four DINOv2 encoder variants on SSv2. Frame-wise adaptation treats each frame independently. In the shuffled temporal variant, we shuffle the frame order when encoding videos with the temporal DINO encoder, then restore the encoded features to their original temporal order for video generation training. Frame-wise adaptation obtains an FVD of 255.49, compared with 262.05 for the image-only encoder. The shuffled temporal variant obtains 220.76, while preserving frame order further lowers FVD to 212.64 and increases ECS from 0.864 to 0.875. Both temporal variants outperform frame-wise adaptation, with ordered encoding performing best in this comparison.

Compact prediction features. Table 5 compares three ways to obtain compact prediction features: PCA, a VAE using the VFMF autoencoder (Boduljak et al., 2026), and prototype-based distillation. The GT and Pred. columns report FVD for generation conditioned on ground-truth and predicted features, respectively. All three 128-dimensional representations improve generation from predicted features over the teacher's original 256-dimensional bottleneck, with relatively similar FVD scores. We choose the prototype bottleneck because it is the simplest to implement within our existing prototype-matching framework and does not require training an additional VAE.

Feature visualization. Figure 3 compares PCA visualizations of our temporal encoder's original 1024-dimensional features and the learned compact features obtained through distillation, with the corresponding RGB frames shown at matched times. The compact features exhibit spatial structures similar to those of the original features in these visualizations. Figure 5 compares PCA projections

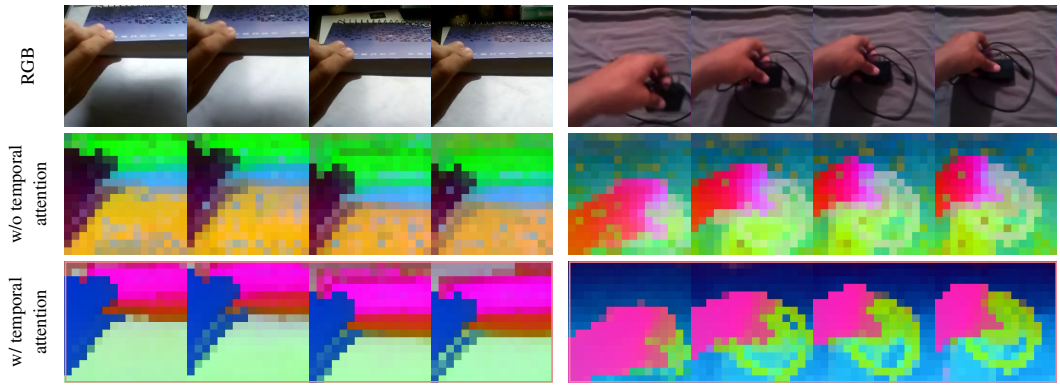


Figure 5: **Feature visualization with and without temporal attention.** Rows show RGB frames and PCA features without and with temporal attention; columns show matched times.

of DINOv2 features with and without temporal attention at four matched times from two video sequences.

Planning score comparison. To isolate scoring design from a world model's ability to predict future states, we use the ground-truth Procgen simulator to generate future trajectories for candidate action sequences and compare scoring functions under fixed MPC settings. Figure 6 compares six scoring variants across four environments, with four seeds. DINOv2 uses the pretrained image encoder, while T-DINO uses our temporally trained encoder in a frame-wise mode without cross-frame context. Both encode each rollout frame and the goal independently, average cosine similarities over corresponding patches, and use the maximum frame score for each trajectory. Our goal-representation consistency cost (Section 3.3) with the temporal DINO encoder achieves the highest success rate of 84.17%, outperforming all frame-wise scoring baselines and RGB cosine. V-JEPA 2.1-C measures consistency using the L1 distance between predicted and target goal features, yielding 39.79% success, below its frame-wise baseline of 50.21%. This comparison suggests that the effectiveness of the proposed cost depends on the underlying representation.

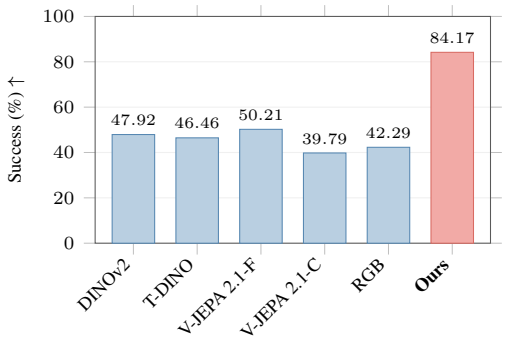


Figure 6: Planning success using ground-truth future trajectories. DINOv2, T-DINO (our temporal DINO encoder), and V-JEPA 2.1-F use frame-wise scoring. V-JEPA 2.1-C uses L1 distance between predicted and target goal features; Ours uses prototype cross-entropy with our temporal DINO encoder.

5 Conclusion

We presented TRIP, a framework that adapts an image-pretrained encoder to video clips through self-supervised learning with causal temporal attention. We distill the encoder's prototype distributions into a compact bottleneck representation for video prediction and reuse the frozen encoder to score imagined rollouts using a goal-representation consistency cost, without additional cost-specific training. Results on SSv2 and LIBERO support the utility of the learned representations for video prediction. Closed-loop experiments on Procgen demonstrate improved planning with learned dynamics, while simulator-based ablations support the utility of the proposed cost under ground-truth dynamics. On VP^2, the scoring encoder transfers from Procgen without further adaptation and improves planning with a fixed iVideoGPT world model.

AI Use Statement

We used large language models to assist with drafting and polishing the manuscript, retrieving and identifying related literature, and research ideation or execution. We did not use these tools to

generate synthetic datasets or prove mathematical claims. The authors take full responsibility for the final content of this work, including all text, references, methods, and results produced with AI assistance.

REFERENCES

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Alan Baade, Eric Ryan Chan, Kyle Sargent, Changan Chen, Justin Johnson, Ehsan Adeli, and Li Fei-Fei. Latent forcing: Reordering the diffusion trajectory for pixel-space image generation. *arXiv preprint arXiv:2602.11401*, 2026.

Federico Baldassarre, Marc Szafraniec, Basile Terver, Vasil Khalidov, Francisco Massa, Yann LeCun, Patrick Labatut, Maximilian Seitzer, and Piotr Bojanowski. Back to the features: Dino as a foundation for video world models. *arXiv preprint arXiv:2507.19468*, 2025.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15791–15801. IEEE, 2025.

Gabrijel Boduljak, Yushi Lan, Christian Rupprecht, and Andrea Vedaldi. Vfmf: Dense forecasting by generating foundation model features. In *Forty-third International Conference on Machine Learning*, 2026.

Bingyi Cao, Koert Chen, Kevis-Kokitsi Maninis, Kaifeng Chen, Arjun Karpur, Ye Xia, Sahil Dua, Tanmaya Dabral, Guangxing Han, Bohyung Han, et al. Tipsv2: Advancing vision-language pretraining with enhanced patch-text alignment. *arXiv preprint arXiv:2604.12012*, 2026.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9630–9640. IEEE, 2021.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 4724–4733. IEEE, 2017.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Karl Cobbe, Christopher Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. *arXiv preprint arXiv:1912.01588*, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Frederik Ebert, Chelsea Finn, Sudeep Dasari, Annie Xie, Alex Lee, and Sergey Levine. Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. *arXiv preprint arXiv:1812.00568*, 2018.

Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 2786–2793. IEEE, 2017.

Zelin Fu, Bin Tan, Changjiang Sun, Shaohui Liu, Kecheng Zheng, Yinghao Xu, Xing Zhu, Yujun Shen, and Nan Xue. Vision pretraining for dense spatial perception. *arXiv preprint arXiv:2607.05247*, 2026.

Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. *arXiv preprint arXiv:2503.18938*, 2025.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, et al. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pp. 5842–5850, 2017.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3):440, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019b.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Danijar Hafner, Wilson Yan, and Timothy Lillicrap. Training agents inside of scalable world models. *arXiv preprint arXiv:2509.24527*, 2025.

Nick Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, volume 2024, pp. 47376–47405, 2024.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

Siqiao Huang, Jialong Wu, Qixing Zhou, Shangchen Miao, and Mingsheng Long. Vid2world: Crafting video diffusion models to interactive world models. *arXiv preprint arXiv:2505.14357*, 2025.

Efstathios Karypidis, Ioannis Kakogeorgiou, Spyridon Gidaris, and Nikos Komodakis. Dinoforesight: Looking into the future with dino. *Advances in Neural Information Processing Systems*, 38:163779–163811, 2026.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

Yang Liu, Qianqian Xu, Peisong Wen, Siran Dai, and Qingming Huang. When the future becomes the past: Taming temporal correspondence for self-supervised video representation learning, 2025.

Yifan Lu, Qi Wu, Jay Zhangjie Wu, Zian Wang, Huan Ling, Sanja Fidler, and Xuanchi Ren. Pid: Fast and high-resolution latent decoding with pixel diffusion. *arXiv preprint*, 2026.

Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

Yecheng Jason Ma, Vikash Kumar, Amy Zhang, Osbert Bastani, and Dinesh Jayaraman. Liv: Language-image representations and rewards for robotic control. In *International Conference on Machine Learning*, pp. 23301–23320. PMLR, 2023.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. Leworld-model: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mido Assran, Koustuv Sinha, Mike Rabbat, Yann LeCun, Nicolas Ballas, and Adrien Bardes. V-jepa 2.1: Unlocking dense features in video self-supervised learning. *arXiv preprint arXiv:2603.14482*, 2026.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Max Schwenzer, Muzaffer Ay, Thomas Bergs, and Dirk Abel. Review on model predictive control: An engineering perspective. *The international journal of advanced manufacturing technology*, 117(5):1327–1349, 2021.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Uladzislau Sobal, Wancong Zhang, Kyunghyun Cho, Randall Balestriero, Tim GJ Rudner, and Yann LeCun. Learning from reward-free offline data: A case for planning with latent dynamics models. *Advances in Neural Information Processing Systems*, 38:43905–43941, 2026.

Yihong Sun, Hao Zhou, Liangzhe Yuan, Jennifer J Sun, Yandong Li, Xuhui Jia, Hartwig Adam, Bharath Hariharan, Long Zhao, and Ting Liu. Video creation by demonstration. *arXiv preprint arXiv:2412.09551*, 2024.

Izzeddin Teeti, Rongali Sai Bhargav, Vivek Singh, Andrew Bradley, Biplab Banerjee, and Fabio Cuzzolin. Temporal dino: A self-supervised video strategy to enhance action prediction. In *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 3273–3283. IEEE, 2023.

Basile Terver, Tsung-Yen Yang, Jean Ponce, Adrien Bardes, and Yann LeCun. What drives success in physical planning with joint-embedding predictive world models? *arXiv preprint arXiv:2512.24497*, 2025.

Stephen Tian, Chelsea Finn, and Jiajun Wu. A control-centric benchmark for video prediction. *arXiv preprint arXiv:2304.13723*, 2023.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022.

Sethuraman TV, Savya Khosla, Vignesh Srinivasakumar, Jiahui Huang, Seoung Wug Oh, Simon Jenni, Derek Hoiem, and Joon-Young Lee. Frame: Pre-training video feature representations via anticipation and memory. *arXiv preprint arXiv:2506.05543*, 2025.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Ying Wang, Oumayma Bounou, Gaoyue Zhou, Randall Balestriero, Tim GJ Rudner, Yann LeCun, and Mengye Ren. Temporal straightening for latent planning. *arXiv preprint arXiv:2603.12231*, 2026.

Yucen Wang, Fengming Zhang, De-Chuan Zhan, Li Zhao, Kaixin Wang, and Jiang Bian. Co-evolving latent action world models. *arXiv preprint arXiv:2510.26433*, 2025.

Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. ivideogpt: Interactive videogpts are scalable world models. *Advances in Neural Information Processing Systems*, 37:68082–68119, 2024.

Mingkun Zhang, Wangtian Shen, Fan Zhang, Haijian Qin, Zihao Pei, and Ziyang Meng. Rae-nwm: Navigation world model in dense visual representation space. *arXiv preprint arXiv:2603.09241*, 2026a.

Shilong Zhang, He Zhang, Zhifei Zhang, Chongjian Ge, Shuchen Xue, Shaoteng Liu, Mengwei Ren, Soo Ye Kim, Yuqian Zhou, Qing Liu, et al. Both semantics and reconstruction matter: Making representation encoders ready for text-to-image generation and editing. *arXiv preprint arXiv:2512.17909*, 2025.

Tianqiu Zhang, Muyang Lyu, Yufan Zhang, Fang Fang, and Si Wu. Dila: Disentangled latent action world models. *arXiv preprint arXiv:2605.15725*, 2026b.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

Fangqi Zhu, Hongtao Wu, Song Guo, Yuxiao Liu, Chilam Cheang, and Tao Kong. Irasim: A fine-grained world model for robot manipulation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9834–9844. IEEE, 2025.