PROCEDURAL PLANNING WITH JEPA WORLD MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

In this work, we introduce JEPA-Pro, a Joint-Embedding Predictive Architecture (JEPA)-based world model trained end-to-end for procedural planning, whose unit of prediction is a procedural step and in which text is used only to define actions and goals. World models predict how the state of the world changes under an action, which makes them a natural fit for model-based planning. Current world models, however, predict at a fixed temporal granularity which suits low-level control but is misaligned with the unit of decision in procedural tasks such as assembling furniture or cleaning a house. The challenge in procedural planning is twofold: (1) steps have variable duration ("wash the floor" may take seconds or minutes), and (2) they are rarely contiguous, since people often performs unrelated actions in between. To address this, our model is composed of two core components: an action-conditioned causal predictor trained jointly with the state encoder, and a learned cost function that estimates how far a predicted state is from the goal. At inference time, JEPA-Pro produces plans by test-time search over candidate action sequences in latent space, selecting the sequence whose predicted trajectory minimizes the cost, without generating pixels nor text. We evaluate JEPA-Pro on world modeling, procedural understanding and procedural planning benchmarks. We establish a new state-of-the-art on WorldPrediction-WM (71.8%) and on EgoPlan-Bench (58.7%). On ENACT, the world model in zero-shot outperforms 23 of 30 VLM baseline, and all caption-free baselines on WorldPrediction-PP (41.6%) using up to two order of magnitude less parameters. We show that finetuning the cost with few trajectories enables the world model to adapt to unseen tasks while naturally enabling test-time scaling during planning, outperforming zero-shot VLM policies on ALFRED and BEHAVIOR-1K.

1 INTRODUCTION

Autonomous agents that act in the physical world, from household robots to embodied assistants (Fung et al., 2025), must carry out long-horizon tasks such as assembling furniture or cleaning a house. Such tasks decompose across two timescales: a low-frequency decision layer that selects discrete, semantically meaningful steps toward a language-specified goal (Fikes & Nilsson, 1971; McDermott et al., 1998), and a high-frequency control layer that realizes each step as continuous motor commands (Latombe, 1996; Mayne et al., 2000). *Procedural planning* (Chang et al., 2020) is the decision layer: given a text goal, it must produce the sequence of text actions that fulfills it. It determines what the agent attempts, and its quality bounds every lower-level controller below it.

Procedural planning operates over temporally abstract actions. Its steps (*e.g.,* "pick up the mug") have no canonical duration: the same step may take seconds in one execution and minutes in another. For this reason, transitions are indexed by what happened rather than by how long it took, following a semi-MDP formulation (Sutton et al., 1999) rather than a fixed-stride process. Sequential decision problems of this kind are classically addressed with model-based planning (Hafner et al., 2020): a transition model, built with approaches such as *world models*, predicts how the state changes under a candidate action (Ha & Schmidhuber, 2018), a cost measures the remaining distance to the goal (Ma et al., 2023; Wang et al., 2026b), and the plan is obtained by optimization: simulating and comparing the outcomes of different candidate action sequences before committing to one (Zhou et al., 2025b).

World models (Ha & Schmidhuber, 2018; LeCun et al., 2022) have largely been developed for low-level control (Zhou et al., 2025b; Ye et al., 2026) to do Model Predictive Control (MPC)-based planning in close-loop: candidate action sequences are rolled out and the first action of the best

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.

sequence is executed before the plan is recomputed. Evaluation is therefore external to proposal, and additional test-time search converts into better plans (Silver et al., 2017). At the procedural level, however, the default approach is not MPC-based. Because goals and actions are specified in text, the planner is a LLM prompted with the goal (Chen et al., 2025b), generating the action sequence token by token from priors over canonical procedures acquired during pretraining (Liu et al., 2023; Yang et al., 2025). Such a "planner" is in fact a reactive policy: it maps the goal and the context directly to an action, and the same prior both proposes candidates and scores them. Additional test-time search therefore only re-samples that prior. Closing this gap requires a model-based decomposition at the procedural level: a higher-level world model that predicts how the task state changes under a textual action, and a cost that scores those predicted states against the language goal.

We introduce **JEPA-Pro**, a high-level world model for procedural planning, trained to predict world state dynamics conditioned on semantically and temporally abstract actions described in text. This architecture follows the paradigm of Causal Temporal Prediction (CTP)-JEPA, where autoregressive prediction happens in a state space produced by an end-to-end learned causal encoder with no explicit masking (see §2). The encoder and predictor of JEPA-Pro is optimized at the same time with anti-collapse regularization. After world model pretraining, we train a lightweight cost module on top of the learned state space to allow goal specification with language. We train JEPA-Pro on 4.68M annotated steps spanning 2.51M procedural trajectories, mixing egocentric human activity with robot recordings, and evaluate it on four benchmarks: ENACT Wang et al. (2026a), EgoPlan-Bench (Chen et al., 2025d), WorldPrediction (Chen et al., 2025a), and our adaptation of the procedural planning task (Chang et al., 2020) to ALFRED (Shridhar et al., 2020) and BEHAVIOR-1K. The world model pretrained end-to-end outperforms most prompted VLM baselines on ENACT and all caption-free baselines on WorldPrediction-PP, while establishing new state-of-the-art on both WorldPrediction-WM (71.8%) and EgoPlan-Bench (58.7%). On the simulated procedural trajectories, our model outperforms zero-shot VLM policies while enabling test-time search. Our contributions are twofold:

- **Model-based procedural planning.** We show that our world model approach enables MPC-based planning with test-time search on procedural tasks, with JEPA-Pro outperforming LLM-based off-the-shelf policies.
- **A Causal Temporal Prediction (CTP)-JEPA.** We train a sequence-level action-conditioned causal predictor jointly with a single state encoder to extract high-level, predictive features for procedural planning.

2 Related Work

Procedural Planning. Real-world decision making spans a hierarchy of abstraction and timescales. Low-level controllers execute movements for locomotion and manipulation, while higher-level planners compose these behaviors into extended tasks. Temporally extended behaviors can be formalized as *options* (Sutton et al., 1999), which abstract over sequences of primitive actions. At the procedural level, actions are also semantically meaningful and often expressible in language, such as "fetch a mug" or "attach a table leg." These steps specify what to accomplish while abstracting away execution details and variable durations. *Procedural planning* composes such steps toward a goal, enabling long-horizon embodied behavior and assistance with activities such as cooking, household maintenance, and assembly (Chang et al., 2020; Patel et al., 2023).

A common approach uses language or vision–language models to generate plans from a goal and observed context, drawing on procedural knowledge acquired during pretraining (Patel et al., 2023; Islam et al., 2024). Used this way, the model acts as a policy: it proposes actions directly, without an explicit transition model for evaluating their consequences. Sampling more plans broadens the proposals but does not itself provide an independent measure of whether they will succeed. Search-based methods such as VidAssist (Islam et al., 2024) therefore add candidate assessment and search around the language model. When pretrained knowledge is insufficient for a particular environment, prompt refinement alone offers limited control over the model's understanding of its dynamics, motivating learning predictive models from grounded trajectories.

Vision-Language World Model (VLWM) (Chen et al., 2025b) pursues this direction by training on structured action–state trajectories extracted from videos. It learns both an action policy and a

language-based dynamics model, enabling a learned critic to evaluate imagined outcomes during search. However, each imagined transition requires autoregressive decoding of a textual state description, adding computation that competes with search breadth under a fixed inference budget. JEPA-Pro instead learns action-conditioned dynamics in a continuous visual representation space through joint-embedding prediction. Language specifies actions and goals, but intermediate states are predicted and scored as embeddings. This retains the separation between action proposal and outcome evaluation while removing intermediate text decoding, enabling more efficient search.

JEPA World Models. Joint-embedding predictive architectures learn by predicting representations rather than reconstructing observations. Dual-encoder distillation-based methods such as I-JEPA (Assran et al., 2023) and V-JEPA (Bardes et al., 2024) learns strong representation, but the predictor is limited to masked patch predition rather than causal world modeling. Frozen-encoder world models such as DINO-WM (Zhou et al., 2025a), V-JEPA 2-AC (Assran et al., 2025), and JEPA-WMs (Terver et al., 2026) learn action-conditioned dynamics, but the encoder remain frozen thus limiting learning abstract predicted features best suited for world modeling.

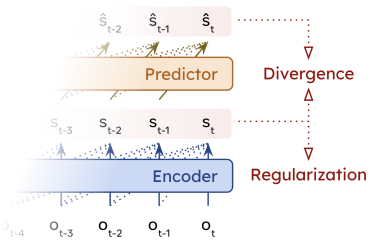

Figure 1: CTP-JEPA.

We refer to *Causal Temporal Prediction (CTP)-JEPA* (Fig. 1) as a complementary paradigm: sequence-level autoregressive prediction with a single shared encoder trained jointly with a causal predictor, and explicit anti-collapse regularization (Bardes et al., 2022)). The causal predictor can parallelize next state prediction, and at inference it can be rolled out autoregressively. Joint training allows the representation to adapt to the dynamics objective, while regularization preserves information that prediction matching alone could discard.

PLDM (Sobal et al., 2026) and LeWorldModel (Maes et al., 2026) instantiate key elements of this paradigm such as autoregressive prediction and end-to-end training with explicit regularization. These methods demonstrate representation-space planning in navigation and robotic motion control of high-frequency low-level actions. JEPA-Pro applies this paradigm to high-level procedural dynamics. Its actions are semantically meaningful steps described in language, with variable durations and potentially substantial changes between observations. Additionally, JEPA-Pro use a causal state estimator aggregates history to address the partial observation challenge.

3 METHODOLOGY

JEPA-Pro (Fig. 2) is a video world model using the joint-embedding predictive architecture to learn representations and predictions of semantically and temporally abstract actions and states. A **state estimation** module aggregates current and historical observations into state embeddings. An action-conditioned **state prediction** module predict the embedding of the next abstract world state estimated by the model in an autoregressive manner. Both state estimation and prediction modules are optimized jointly with anti-collapse regularization.

After training, we use JEPA-Pro for **model-based procedural planning**. Given observation and a candidate action sequence, we autoregressively roll out state prediction to obtain a predicted resulting state. A lightweight cost module evaluates this state relative to a goal state, allowing candidate sequences to be ranked by their predicted outcomes.

Notation. We train JEPA-Pro on procedural videos $\{[o_0, o_1, ..., o_n]\}$ where each o_t is a video clip and n is the total number of steps. We have corresponding textual action descriptions, and use a_t to represent the encoded action embedding for the t-th step. We use s_t to represent the state estimated by the model, and use $\hat{s}_{t+1}$ to represent the prediction of the next step.

3.1 WORLD MODELING WITH JEPA-PRO

State Estimation. Procedural planning requires accurate estimation of the state of the world to allow searching for an action plan to achieve the goal. Real world procedural video often involves *partial observation*–each video segment may reveal only a narrow view of the world, so estimating the current state requires aggregating information from earlier observations. We instantiate $s_t = \texttt{state estimation}(o_{\leq t})$ with three components:

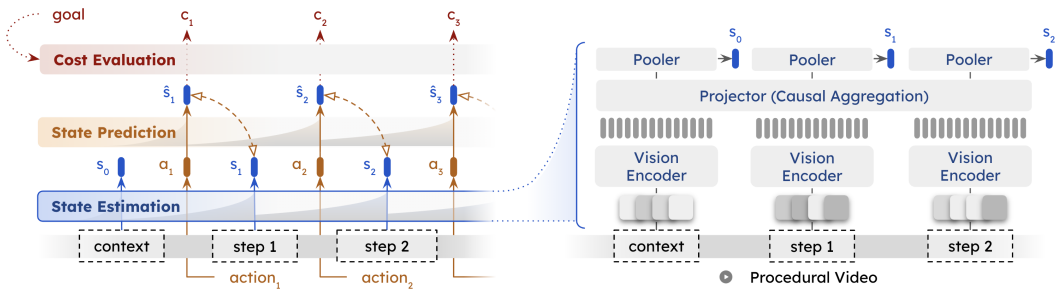

Figure 2: JEPA-Pro architecture. A state estimator aggregates visual observations into state embeddings, an action-conditioned predictor models their evolution, and a goal-conditioned cost module evaluates candidate plans.

- **Vision Encoder.** A Vision Transformer (ViT) encodes the frames into dense patch tokens. It operates locally and independently for individual steps.
- **Projector.** Transformer layers contextualize these tokens using block-causal attention across steps, allowing the current step attend to itself and the past.
- **Pooler.** Attentive pooling layer compresses the contextualized tokens of each step into single-vector state embeddings.

State Prediction. It learns to predict future state embeddings conditioned on an action and the past trajectory, *i.e.*, $\hat{s}_{t+1} = \texttt{state prediction}(s_{\leq t}, a_{\leq t})$. Each action description is encoded with the text encoder initialized through VL-JEPA (Chen et al., 2026a) pretraining, interleave the action embeddings with the estimated states, and feed them into a causal Transformer. The output at each action embedding position predicts the next state. We train the predictor backbone to output a residual relative to the action embedding with zero-initialized linear head to inherent from contextless VL-JEPA pretraining.

The predictor can attend to past states and actions, allowing it to use information beyond the most recent state–action pair. We supervise $\hat{s}_{t+1}$ with s_{t+1}, computed in a separate forward pass through the same state estimation module. We train the predictor autoregressively with teacher forcing: its ℓ_2-normalized input states are estimated from observed video. Causal attention allows predictions at all steps to be computed in parallel without accessing future observations or actions.

Training Objectives. We jointly train state estimation and state prediction by matching $\hat{s}_{t+1}$ to s_{t+1}. Since s_{t+1} is also learned, prediction matching alone admits a trivial solution in which all states collapse to the same embedding. We therefore combine three objectives to learn predictive states while preserving their information:

1. **Contrastive prediction** ($\mathcal{L}_{\text{InfoNCE}}$) (van den Oord et al., 2019). We use a bidirectional InfoNCE loss to align each prediction $\hat{s}_{t+1}$ with its corresponding state s_{t+1} through cosine similarity, using other pairs in the batch as negatives.
2. **Variance-covariance regularization** ($\mathcal{L}_{\text{VCReg}}$) (Bardes et al., 2022). We further regularize the estimated states s_{t+1} with two terms: a variance term maintains variation in each feature dimension across samples, and a covariance term reduces redundancy between dimensions.
3. **Inverse dynamics** ($\mathcal{L}_{\text{IDM}}$). We apply an MLP head to s_t to recover the embedding of action a_t. We align the output with detached action embeddings with $\mathcal{L}_{\text{IDM}} = -\cos(\hat{a}_t, \texttt{stop-grad}(a_t))$.

The overall objective is $\mathcal{L} = \lambda_{\text{InfoNCE}}\mathcal{L}_{\text{InfoNCE}} + \lambda_{\text{VCReg}}\mathcal{L}_{\text{VCReg}} + \lambda_{\text{IDM}}\mathcal{L}_{\text{IDM}}$.

3.2 Planning with JEPA-Pro

Learning a Cost Module. The world model describes how states transition under text actions, but not whether a given step makes progress towards a goal. The cost module supplies that judgement,

i.e., $c_t = \texttt{cost}(\hat{s}_{t+1}, a_t, g)$ for a language goal g (Fig. 3). The cost module simply consists of projection trained on top of the frozen world model, which enables efficient downstream task adaptation and is what makes planning towards a language goal possible. We train two projectors: one on top of the state embedding, and one on top of the goal embedding, encoded by the same text encoder used for actions. Both outputs are ℓ_2-normalized into a shared space, and the cost is one minus their cosine similarity.

We train both projectors with two objectives while the world model stays frozen: a contrastive term pulls each state towards the projection of its own goal and away from the other goals in the batch, which establishes the goal a state belongs to. Additionally, a progress term supplies the ordering: a reward shaping loss requiring the cost of the next predicted state to fall below the cost of the current one, so that the cost decreases as the procedure advances. We apply both objectives to predicted states and to states estimated from observations in equal proportion, so that the cost scores an observed state as readily as an imagined one, which is what a plan needs when it starts from ground-truth visual context.

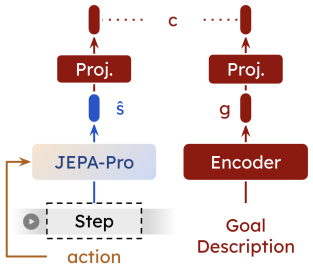

Figure 3: Learned cost: text goal alignment and temporal progress in latent space.

Model-based Planning. Planning is search over a discrete set of text actions, scored by predicted outcome rather than generated token by token. From the estimated state and observed history, we roll a candidate sequence forward by feeding each predicted state back into the predictor, without observing intermediate future clips: the simulated history keeps the observed prefix and appends the generated states and actions. The cost evaluates each predicted successor against the goal, and the planner returns the sequence minimizing cumulative cost over the horizon. This is value-based planning: the cost plays the role of a learned value function over imagined states, and the world model supplies the transitions it is evaluated on. Because the parameters are fixed at inference, the search budget is a free parameter, and plan quality can be traded against inference cost after training.

4 Implementation of JEPA-Pro

4.1 Data

Procedural Trajectories. We assemble a mixture of procedural videos with step-level textual action descriptions from public data sources of egocentric human activities and robotic action recordings. The procedural sources are Ego4D (Grauman et al., 2022), Ego-Exo4D (Grauman et al., 2024), Assembly101 (Sener et al., 2022), COIN (Tang et al., 2019), EgoExoLearn (Huang et al., 2024), EgoVerse (Punamiya et al., 2026), HD-EPIC (Perrett et al., 2025), HoloAssist (Wang et al., 2023), Nymeria (Ma et al., 2024), OpenEgo (Jawaid & Xiang, 2025), Stera-10M (Palanisamy et al., 2026), Xperience10M (Ropedia, 2026), and EPIC-KITCHENS-100 (Damen et al., 2022). We remove actions that do not substantially change physical world states, such as looking, watching, and reading.

Single-Step Actions. We also incorporate video-action pairs from EgoDex (Hoque et al., 2026), SSv2 (Goyal et al., 2017), TASTE-Rob (Zhao et al., 2025a), DROID (Khazatsky et al., 2024), Galaxea (Jiang et al., 2025), GR1 (Wu et al., 2024a), Humanoid Everyday (Zhao et al., 2025b), InternM1 (Chen et al., 2025c), OXE (Ji et al., 2025), RH20T (Fang et al., 2024), RoboCoin (Wu et al., 2025), RoboMind (Wu et al., 2024b), and RoboMind2 (Hou et al., 2025) as single-step trajectories. The combined mixture contains 4.68M annotated action steps across 2.51M contiguous trajectories, with human activities accounting for 62.5% of the steps. Single-step trajectories constitute 93.4% of the trajectories, whereas the remaining 6.6% are multi-step and contribute approximately half of all action steps.

4.2 Model

State Estimator. We instantiate three variants of JEPA-Pro using ViT-B (86M), ViT-L (0.3B), and ViT-G (2B) backbones. Their ViT projectors contain 28M, 0.1B, and 0.6B parameters, respectively. We refer to the resulting models as JEPA-Pro-{base, large, giant}, with respectively {1.1B, 1.5B and

3.5B} total parameters. An attentive pooler with a learned query aggregates the projector outputs into a state embedding.

State Predictor. All three models use a 24-layer causal Transformer with 0.5B parameters, including the output head and token-type embeddings. A zero-initialized linear output head predicts a residual added to the conditioning action embedding. Input states are ℓ_2-normalized. We encode action descriptions using the text tower of Perception Encoder.

4.3 TRAINING

VL-JEPA Pretrain. State estimation requires strong semantic abstraction. We initialize the vision encoder with V-JEPA 2.1 (Mur-Labadia et al., 2026), which provides motion and appearance features. We initialize the projector and pooler through query-free VL-JEPA pretraining on paired video clips and action descriptions (Fig. 4). This stage can be viewed as context-free, action-conditioned JEPA world-model pretraining. Together, these initializations provide a starting point for end-to-end world modeling.

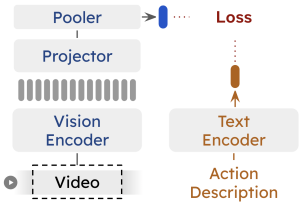

Figure 4: VL-JEPA pretrain.

Sequence Packing. To train JEPA-Pro, we sample windows of up to 16 action steps from longer trajectories, and pack shorter trajectories without allowing them attend to each other. Each trajectory starts with its first frame repeated four times. For each action segment, we use either four copies of the final frame or three random frames followed by the final frame.

Counterfactual Actions. We also replace the conditioning action with alternative actions sampled from the batch and add the resulting predictions to the negative pool. These counterfactual negatives encourage the predictor to distinguish the effects of different actions under the same context.

Context Dropout. State estimation uses causal attention with randomized context dropout. For each segment, we sample the number of visible preceding segments and randomly select a subset of that size, while retaining the current segment. The context and supervision states use shared parameters but separate forward passes with independent context sampling. Gradients flow through the supervision states; there is no stop-gradient target branch. Predictor context dropout similarly samples the accessible history while preserving the most recent state–action pair.

5 EXPERIMENTS

To validate our approach, we first evaluate our pretrained world model in zero-shot on ENACT (Section 5.1) and on WorldPrediction (Section 5.2), after which we evaluate jointly the cost and the world model on EgoPlan-Bench (Section 5.3). Then we show that finetuning a simple cost enables efficient downstream task-adaptation and test-time scaling on Procedural Planning (Section 5.4)

5.1 ENACT

Task Definition. ENACT (Wang et al., 2026a) tests whether a model can recover the temporal structure of extended egocentric interaction trajectories in BEHAVIOR-1K (Li et al., 2023). Each test sample contains an initial observation o_0, $L-1$ subsequent observations, and $L-1$ action descriptions, with $L \in \{3, \ldots, 10\}$. The "forward world modeling" task requires a model to order shuffled observations given the ordered actions, while "inverse world modeling" orders shuffled actions given the ordered observations. We measure task accuracy which measures the fraction of complete orderings accepted by the official verifier, and report average accuracy over 4,486 questions per direction.

Evaluation Setup. We parse the action descriptions from VQA prompts and encode them as a_t, then use JEPA-Pro to search for the best trajectory by evaluating compatibility score $\cos(\hat{s}_{t+1}, s_{t+1}) + \cos(\hat{a}_t, a_t)$: the prediction score measures agreement with the observed successor state, while the IDM score measures agreement with provided actions. We maximize the summed score using exact subset dynamic programming for forward ordering and exact linear assignment for inverse ordering.

Results and Discussion. Figure 5 compares task accuracy against prompted VLM baselines that leverage language-based reasoning to solve the task. JEPA-Pro's performance is competitive, it

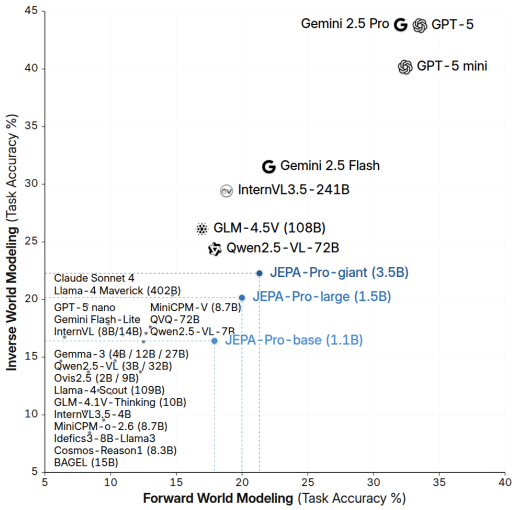

Figure 5: **ENACT benchmark results.** Dashed lines project JEPA-Pro scores, baselines in each region have no higher accuracies on either tasks.

Model	World Modeling	Procedural Planning
Caption-based Models		
Llama-3.1 (8 / 70B)	48.7 / 49.8	26.7 / 31.2
Llama-3.3 (70B)	52.2	35.1
Llama-4 (Scout / Maverick)	52.7 / 53.6	32.8 / 34.7
Qwen2.5 (3 / 72B)	44.0 / 48.5	25.6 / 30.7
DeepSeek-R1 (distilled)	50.8	28.4
Gemini-2.0	55.6	33.5
GPT-4o	52.0	33.7
Claude-3.5-Sonnet	53.3	38.1
VLWM-critic-1B	-	45.4
Caption-free Models		
InternVL2.5 (2 / 38B)	20.0 / 50.3	21.1 / 31.1
Qwen2.5-VL (3 / 72B)	21.6 / 57.0	29.1 / 36.7
I2VGenXL (RGB / DINOv2)	26.1 / 26.7	-
CogVideoX (RGB / DINOv2)	30.1 / 30.5	-
OEPP-MLP	-	36.8
VL-JEPA (BASE / SFT)	63.9 / 65.7	-
JEPA-Pro (ViT B / L / G)	57.1 / 62.9 / **71.8**	30.7 / 38.2 / **41.6**

Table 1: **WorldPrediction accuracies (%).** World modeling and procedural planning tasks evaluate four-way action and plan selection given observations of initial and final states, with candidates represented with action-equivalent clips.

outperforms 23 out of 30 VLM baselines, including the ones with 100+ times more parameters. Importantly instead of relying on text prior, JEPA-Pro performs latent reasoning and test-time search over possible trajectories. The task is challenging especially in long-horizon scenarios–at $L = 10$, there are $9! = 362{,}880$ possible ordering that the model needs to pick from. The chance performance of forward world modeling at $L = 10$ is 0.004453‰ and our models achieved nonzero accuracies at 4 / 6 / 8‰, along with only four of the 30 VLM baselines: GPT-5 (50‰), GPT-5 mini (28‰), Gemini 2.5 Pro (36‰), and Gemini 2.5 Flash (2‰). Note that we test JEPA-Pro in zero-shot without training on BEHAVIOR-1K.

5.2 WorldPrediction

Task Definition. The WorldPrediction (Chen et al., 2025a) benchmark evaluates action–effect understanding in real-world procedural videos. Given initial and final visual states, World Modeling (WM) consists of selecting the correct action clip representing that transition from four candidates, while Procedural Planning (PP) selects the correctly ordered sequence of action clips from four candidate sequences. Each action clip is an action equivalent, a clip depict the same actions but in different contexts or viewpoints to eliminate shortcuts from visual continuity.

Evaluation Setup. We encode each candidate clip and apply the IDM head to obtain its action embedding a. For WM, we predict $\hat{s}_1$ from the initial state and candidate action, and recover $\hat{a}$ with the IDM head. We select the candidate maximizing $\cos(\hat{s}_1, s_1) + \cos(\hat{a}, a)$. For PP, we roll out each candidate sequence and select the one maximizing terminal-state similarity $\cos(\hat{s}_t, s_t)$.

Results and Discussion. As shown in Table 1, JEPA-Pro-giant achieves 71.8% WM accuracy, exceeding the previous state-of-the-art VL-JEPA (SFT) by 6.1%. Our model also reaches 41.6% on PP, outperforming all caption-free baselines. Caption based-model uses text captions of each candidate to solve the task, while caption-free models, including ours, score the clips directly as video, as intended in the benchmark. Ordering a full plan still benefits from text description, as shown by the PP ceiling from VLWM-critic at 45.4%.

5.3 EgoPlan-Bench

Task Definition. EgoPlan-Bench (Chen et al., 2025d) evaluates next-action selection from a partially observed egocentric procedure. Each question supplies an observed video history as context, along with the current state as a static image, a language goal g, and four candidate action descriptions $a^{(1)}, \ldots, a^{(4)}$, one of which optimally continues the procedure to reach the goal. The

benchmark contains three levels probe, each increasing distribution shift: L1 is in-distribution and drawn from EPIC-KITCHENS-100 (Damen et al., 2022), L2 is cross-environment and drawn from Ego4D (Grauman et al., 2022), and L3 is cross-environment and cross-task.

Evaluation Setup. We select the candidate action by next-state prediction. The state estimator encodes the observed prefix into s_t; for each candidate the predictor produces one successor state $\hat{s}_{t+1}^{(i)} = \texttt{state prediction}(s_{\leq t}, a^{(i)})$, and the cost module scores it against the goal, $c^{(i)} = \texttt{cost}(\hat{s}_{t+1}^{(i)}, a^{(i)}, g)$. We select the candidate that minimizes this cost: $\arg\min_i c^{(i)}$. We jointly fine-tune the world model and the cost on the benchmark's training split with a four-way cross-entropy objective over the negative candidate costs. Its setup combines a video history as context with the current state, given as a single static image, from which the plan must continue. We sample that history into two eight-frame clips and repeat the static image to form a third, so the current state enters the model in the same form as an observed step. We compare against the original references reported with the benchmark (Chen et al., 2025d), as well as more recent supervised VLMs (Park et al., 2026) and reinforcement-learning approaches (Chen et al., 2026b).

Model	L1	L2	L3	Macro
Random chance	25.0	25.0	25.0	25.0
Qwen2-VL-7B zero-shot*	34.8	34.0	31.6	33.5
Qwen2-VL-7B SFT*	43.8	44.1	38.2	42.0
Qwen3-VL-8B zero-shot	43.8	48.9	44.0	45.5
Qwen2-VL-7B GRPO*	46.0	50.2	44.9	47.0
DeepVideo-R1*	52.0	55.7	47.7	51.8
GRPO-CARE*	57.0	57.0	53.4	54.6
JEPA-Pro-base	55.3	57.3	43.3	52.0
JEPA-Pro-large	56.3	62.6	51.6	56.8
JEPA-Pro-giant	**59.5**	**62.1**	**54.7**	**58.7**

Table 2: **EgoPlan-Bench accuracies (%).** Rows marked * are taken directly from their original papers.

Results and Discussions. Our world model approach establishes a new state of the art on EgoPlan-Bench (Table 2). The largest model reaches 58.7%, 4.1 points above GRPO-CARE, the strongest fine-tuned vision-language baseline, and leads at every level. Accuracy rises consistently with encoder size, from 52.0% to 56.8% to 58.7%, and that ordering holds at each level individually. The margin is widest on L2, 62.1% against 57.0%, and narrowest on L3, where both the environment and the task change. We show that, instead of relying on language priors like most LLM-based baseline, predicting the outcome of each candidate action with a world model is a stronger basis for plan continuation towards a language goal.

5.4 Procedural Planning and Test-Time Search

Task Definition. We adopt the existing task of Procedural Planning (Chang et al., 2020; Patel et al., 2023) in two widely used simulated environments, ALFRED (Shridhar et al., 2020) and BEHAVIOR-1K. Given a language goal g, the current visual observation o_t and the sequence of actions already executed $a_{1:t}$, the planner selects the next $T = 4$ actions to reach the language goal. We split both datasets into *seen* and *unseen* by their task, where a task is defined by its language goal: 189 tasks and 1,106 test samples in ALFRED, 50 tasks and 8,075 test samples in BEHAVIOR-1K. The action space is discrete and built from the entire *unseen* split vocabulary with 179 unique text actions in ALFRED and 491 unique text actions in BEHAVIOR-1K. Following the original definition of the task, we report success rate (SR) as the fraction of windows where all four actions exactly match the reference, and mean accuracy (mAcc) as the fraction of individual positions matched. Every number is reported on the unseen test split.

Evaluation Setup. We first fine-tune the world model on the *seen* split of both datasets to learn the general transition dynamics of these synthetic environments, then freeze the world model to train only the cost and a lightweight action prior on the *unseen* training split, such that the cost serves only as a task-specific adapter. The action prior here is another cost term with the same architecture but taking as input the action history instead of the predicted state, so that a candidate is scored against the same goal from both ends of the trajectory: by the state it is predicted to reach, and by the actions already taken towards that goal. At inference time, JEPA-Pro rolls out $T = 4$ steps in latent space from the current state s_t, and at each step the costs scores the predicted state $\hat{s}_{t+k}$ against the goal g for every action in the pool. We vary two axes independently: the search width, from greedy to beam-4, which changes only the computation spent at inference time, and the number of unseen training trajectories the cost and the prior are finetuned on. As baselines, we evaluate various vision-language models in zero-shot, giving each the current visual context, the language goal, the executed actions and the full action pool, and prompting it to generate the next four actions directly.

Method	ALFRED SR	ALFRED mAcc	Behavior-1K SR	Behavior-1K mAcc
Qwen3-VL-2B	2.8	9.0	0.5	4.7
Qwen3-8B (text-only)	13.7	26.1	3.2	11.1
Qwen3-VL-8B	22.2	36.6	4.9	16.2
Qwen3-VL-30B-A3B	27.1	43.1	6.4	20.0
JEPA-Pro-giant				
$Cost_{10}$	4.6	11.2	3.4	5.9
$Cost_{100}$	32.2	46.4	9.0	21.3
$Cost_{full}$	**64.6**	**77.9**	**10.0**	**24.6**

Table 3: **Procedural Planning results** with beam-4 and different number of training trajectories. Success Rate (SR) and Mean Accuracy (mAcc) in %.

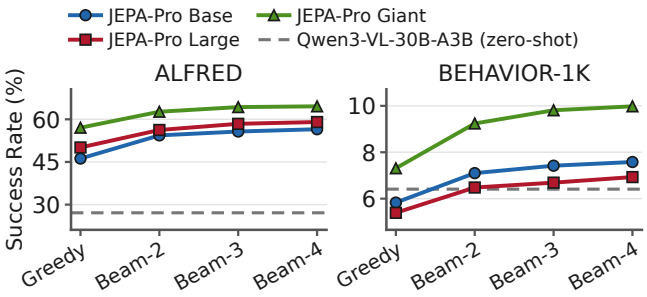

Figure 6: **Test-time scaling.** Success rate against beam width for all three JEPA-Pro sizes on ALFRED and BEHAVIOR-1K. Wider search improves performance for every model, showing that more test-time compute improves planning performance. Dashed line is the strongest zero-shot baseline.

Results and Discussions. JEPA-Pro outperforms every zero-shot baseline on both environments with as few as one hundred training trajectories for the cost as shown in Table 3, reaching 32.2% success rate and 46.4% mean accuracy against respectively 27.1% and 43.1% on ALFRED; and 9.0% and 21.3% against respectively 6.4% and 20.0% on BEHAVIOR-1K. Ten trajectories cover too few of the unseen tasks for the planner to adapt, while finetuning the cost on the full unseen split lifts success rate substantially further, to 64.6% on ALFRED and 10.0% on BEHAVIOR-1K. Although the world model has not been trained on these tasks specifically, we show that leveraging it for new procedural tasks only requires a few training trajectories for the cost. BEHAVIOR-1K remains the harder of the two throughout, with an action pool nearly three times larger, 491 actions against 179, and more complex tasks. Additionally, success rate improves monotonically with search width for all three JEPA-Pro sizes on both environments, as shown in Figure 6), from 57.1% to 64.6% for JEPA-Pro-giant on ALFRED and from 7.3% to 10.0% on BEHAVIOR-1K. This performance gain from scaling test-time compute naturally comes from the world model-based planning approach

6 Conclusion

We introduce JEPA-Pro, the first end-to-end JEPA world model for procedural planning, composed of two components: (1) a single state encoder trained jointly with an action-conditioned causal predictor trained on 4.68M action steps of egocentric human and robot video, and (2) a cost module learned on top of the resulting state space to ground planning towards a language goal. This allows the model to perform MPC-based procedural planning by rolling out different actions in latent space to evaluate which predicted state brings the high-level plan closer to the specified goal, without generating text nor pixels. We show that a world model trained on procedural trajectories, from both human activities and robot recordings, can achieve strong zero-shot performance in understanding the temporal structure of high-level procedures, outperforming 23 out of 30 prompted VLM baselines on ENACT and every caption-free baseline on WorldPrediction-PP. JEPA-Pro also established a new state-of-the-art on both WorldPrediction-WM at 71.8% and EgoPlan-Bench at 58.7%. We further show that this design enables test-time scaling for procedural planning in ALFRED and BEHAVIOR-1K with very lightweight finetuning of the cost module, even on held-out tasks unseen by the world model.

Limitations point to the next steps. Our simulated evaluation scores plans against annotated references rather than executing them, so success rate measures agreement with a reference plan and not task completion in closed loop. Absolute performance on BEHAVIOR-1K remain relatively low. Executing plans in the synthetic environment, and learning a cost that improves with the search it guides, are the natural directions from here. Coupling this MPC-based high-level planner with an off-the-shelf low-level policy is the most direct way to close the two-timescale decomposition.

ETHICS STATEMENT

This work uses existing, publicly released datasets of egocentric human activity and robot recordings, and involves no new data collection and no human subjects. These sources record ordinary household, kitchen and manipulation activity, and none of them targets harmful or dangerous behaviour. Egocentric footage captures homes, workplaces and bystanders, and we rely on the consent and anonymisation procedures of the original dataset providers rather than introducing our own. Our planner emits high-level action sequences and is evaluated entirely offline, so no real-world execution is involved. Because our metrics measure agreement with a reference plan and not the safety of carrying it out, deploying such a planner on hardware would require a lower-level controller with its own safety guarantees and an evaluation we do not provide here.

AI USE STATEMENT

In this work, we used generative AI tools across the research and writing process. In developing the work, we used them to refine hypotheses, to give feedback on our methodology and experiments, to identify relevant literature, and for brainstorming. In running the experiments, we used them to write code for model implementation, training and evaluation, and to clean and reformat data, including constructing the seen and unseen task splits of ALFRED and BEHAVIOR-1K. In analysing and reporting, we used them to interpret results, to support qualitative and thematic data analysis, to produce and refine plots, to format tables, to assist with translation, and to polish writing. We did not use generative AI tools to generate data nor to develop theoretical frameworks, and every diagram and image in this paper other than the plots was created by the authors. Mathematical claims and proofs are not applicable to this work.

We have reviewed all AI-assisted work. Every AI-drafted passage was read and revised by the authors before inclusion. All numerical claims in the text, tables and figures were checked against the recorded experiment outputs by at least one author, and the figures were regenerated from those same recorded outputs rather than from numbers reproduced by a model. Interpretations proposed with AI assistance were retained only where the authors could verify them directly against the reported results. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15619–15629. IEEE, 2023.

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.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=xm6YD62D1Ub.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024.

Chien-Yi Chang, De-An Huang, Danfei Xu, Ehsan Adeli, Li Fei-Fei, and Juan Carlos Niebles. Procedure planning in instructional videos. In *European Conference on Computer Vision*, pp. 334–350. Springer, 2020.

Delong Chen, Willy Chung, Yejin Bang, Ziwei Ji, and Pascale Fung. Worldprediction: A benchmark for high-level world modeling and long-horizon procedural planning. In *ICML 2025 Workshop on Assessing World Models*, 2025a. URL https://openreview.net/forum?id=3GuGN0bacr.

Delong Chen, Theo Moutakanni, Willy Chung, Yejin Bang, Ziwei Ji, Allen Bolourchi, and Pascale Fung. Planning with reasoning using vision language world model. *arXiv preprint arXiv:2509.02722*, 2025b.

Delong Chen, Mustafa Shukor, Theo Moutakanni, Willy Chung, Lei Yu, Tejaswi Kasarla, Allen Bolourchi, Yann LeCun, and Pascale Fung. Vl-jepa: Joint embedding predictive architecture for vision-language. In *International Conference on Learning Representations*, volume 2026, pp. 89791–89807, 2026a.

Xinyi Chen, Yilun Chen, Yanwei Fu, Ning Gao, Jiaya Jia, Weiyang Jin, Hao Li, Yao Mu, Jiangmiao Pang, Yu Qiao, et al. Internvla-m1: A spatially guided vision-language-action framework for generalist robot policy. *arXiv preprint arXiv:2510.13778*, 2025c.

Yi Chen, Yuying Ge, Rui Wang, Yixiao Ge, Lu Qiu, Ying Shan, and Xihui Liu. Exploring the effect of reinforcement learning on video understanding: Insights from seed-bench-r1. *arXiv preprint arXiv:2503.24376*, 2025d.

Yi Chen, Yuying Ge, Rui Wang, Yixiao Ge, Junhao Cheng, Ying Shan, and Xihui Liu. Grpocare: Consistency-aware reinforcement learning for multimodal reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 4305–4320, 2026b.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *International Journal of Computer Vision*, 2022.

Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Chenxi Wang, Junbo Wang, Haoyi Zhu, and Cewu Lu. Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 653–660. IEEE, 2024.

Richard E Fikes and Nils J Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial intelligence*, 2(3-4):189–208, 1971.

Pascale Fung, Yoram Bachrach, Asli Celikyilmaz, Kamalika Chaudhuri, Delong Chen, Willy Chung, Emmanuel Dupoux, Hongyu Gong, Hervé Jégou, Alessandro Lazaric, et al. Embodied ai agents: Modeling the world. *arXiv preprint arXiv:2506.22355*, 2025.

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 *2017 IEEE international conference on computer vision (ICCV)*, pp. 5843–5851. IEEE, 2017.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022.

Kristen Grauman et al. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives. In *Computer Vision and Pattern Recognition (CVPR)*, 2024.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations (ICLR)*, 2020.

Ryan Hoque, Peide Huang, David Yoon, Jian Zhang, et al. Egodex: Learning dexterous manipulation from large-scale egocentric video. In *International Conference on Learning Representations*, volume 2026, pp. 4218–4237, 2026.

Chengkai Hou, Kun Wu, Jiaming Liu, Zhengping Che, Di Wu, Fei Liao, Guangrun Li, Jingyang He, Qiuxuan Feng, Zhao Jin, et al. Robomind 2.0: A multimodal, bimanual mobile manipulation dataset for generalizable embodied intelligence. *arXiv preprint arXiv:2512.24653*, 2025.

Yifei Huang, Guo Chen, Jilan Xu, Mingfang Zhang, Lijin Yang, Baoqi Pei, Hongjie Zhang, Lu Dong, Yali Wang, Limin Wang, et al. Egoexolearn: A dataset for bridging asynchronous ego-and exo-centric view of procedural activities in real world. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22072–22086. IEEE, 2024.

Md Mohaiminul Islam, Tushar Nagarajan, Huiyu Wang, Fu-Jen Chu, Kris Kitani, Gedas Bertasius, and Xitong Yang. Propose, assess, search: Harnessing llms for goal-oriented planning in instructional videos. In *European Conference on Computer Vision*, pp. 436–452. Springer, 2024.

Ahad Jawaid and Yu Xiang. Openego: A large-scale multimodal egocentric dataset for dexterous manipulation. *arXiv preprint arXiv:2509.05513*, 2025.

Guanhua Ji, Harsha Polavaram, Lawrence Yunliang Chen, Sandeep Bajamahal, Zehan Ma, Simeon Adebola, Chenfeng Xu, and Ken Goldberg. Oxe-auge: A large-scale robot augmentation of oxe for scaling cross-embodiment policy learning. *arXiv preprint arXiv:2512.13100*, 2025.

Tao Jiang, Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Jianning Cui, Xiao Liu, Shuiqi Cheng, Jiyang Gao, Huazhe Xu, and Hang Zhao. Galaxea open-world dataset and g0 dual-system vla model. *arXiv preprint arXiv:2509.00576*, 2025.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

J-C Latombe. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE transactions on Robotics and Automation*, 12(4):566–580, 1996.

Yann LeCun et al. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. 2022.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning*, pp. 80–93. PMLR, 2023.

Jiateng Liu, Sha Li, Zhenhailong Wang, Manling Li, and Heng Ji. A language-first approach for procedure planning. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1941–1954, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.122. URL https://aclanthology.org/2023.findings-acl.122/.

Lingni Ma, Yuting Ye, Fangzhou Hong, Vladimir Guzov, Yifeng Jiang, Rowan Postyeni, Luis Pesqueira, Alexander Gamino, Vijay Baiyya, Hyo Jin Kim, et al. Nymeria: A massive collection of multimodal egocentric daily motion in the wild. In *European Conference on Computer Vision*, pp. 445–465. Springer, 2024.

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. In *International Conference on Learning Representations (ICLR)*, 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.

{D. Q.} Mayne, {J. B.} Rawlings, {C. V.} Rao, and {P. O.M.} Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, June 2000. ISSN 0005-1098. doi: 10.1016/S0005-1098(99)00214-9.

Drew McDermott, Malik Ghallab, Adele E. Howe, Craig A. Knoblock, Ashwin Ram, Manuela M. Veloso, Daniel S. Weld, and David E. Wilkins. Pddl-the planning domain definition language. 1998. URL https://api.semanticscholar.org/CorpusID:59656859.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mido Assran, Koustuv Sinha, Mike Rabbat, Yann LeCun, and Nicolas Ballas. V-jepa 2.1: Unlocking dense features in video self-supervised learning. In *European Conference on Computer Vision*, pp. 671–689. Springer, 2026.

Senthil Palanisamy, Abhishek Anand, Satpal Singh Rathore, Pratyush Patnaik, Shubhanshu Khatana, and Ekaksh Janweja. Mobileego anywhere: Open infrastructure for long horizon egocentric data on commodity hardware. *arXiv preprint arXiv:2605.05945*, 2026.

Jinyoung Park, Jeehye Na, Jinyoung Kim, and Hyunwoo J Kim. Deepvideo-r1: Video reinforcement fine-tuning via difficulty-aware regressive grpo. *Advances in Neural Information Processing Systems*, 38:138605–138632, 2026.

Dhruvesh Patel, Hamid Eghbalzadeh, Nitin Kamra, Michael Louis Iuzzolino, Unnat Jain, and Ruta Desai. Pretrained language models as visual planners for human assistance. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15256–15268. IEEE, 2023.

Toby Perrett, Ahmad Darkhalil, Saptarshi Sinha, Omar Emara, Sam Pollard, Kranti Kumar Parida, Kaiting Liu, Prajwal Gatti, Siddhant Bansal, Kevin Flanagan, et al. Hd-epic: A highly-detailed egocentric video dataset. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23901–23913. IEEE, 2025.

Ryan Punamiya, Simar Kareer, Zeyi Liu, Josh Citron, Ri-Zhao Qiu, Xiongyi Cai, Alexey Gavryushin, Jiaqi Chen, Davide Liconti, Lawrence Y Zhu, et al. Egoverse: An egocentric human dataset for robot learning from around the world. *arXiv preprint arXiv:2604.07607*, 2026.

Ropedia. Xperience-10m: A large-scale egocentric multimodal dataset with structured 3d/4d annotations, 2026. Dataset.

Fadime Sener, Dibyadip Chatterjee, Daniel Shelepov, Kun He, Dipika Singhania, Robert Wang, and Angela Yao. Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21064–21074. IEEE, 2022.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. ALFRED: A benchmark for interpreting grounded instructions for everyday tasks. In *Computer Vision and Pattern Recognition (CVPR)*, 2020.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

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.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. COIN: A large-scale dataset for comprehensive instructional video analysis. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Basile Terver, Tsung-Yen Yang, Jean Ponce, Adrien Bardes, and Yann LeCun. What drives success in physical planning with joint-embedding predictive world models? *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=cHZn5Gdh8e. Reproducibility Certification.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019. URL https://arxiv.org/abs/1807.03748.

Qineng Wang, Wenlong Huang, Yu Zhou, Hang Yin, Tianwei Bao, Jianwen Lyu, Weiyu Liu, Ruohan Zhang, Jiajun Wu, Li Fei-Fei, et al. Enact: Evaluating embodied cognition with world modeling of egocentric interaction. In *International Conference on Learning Representations*, volume 2026, pp. 153022–153077, 2026a.

Xin Wang, Taein Kwon, Mahdi Rad, Bowen Pan, Ishani Chakraborty, Sean Andrist, Dan Bohus, Ashley Feniello, Bugra Tekin, Felipe Vieira Frujeri, et al. Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20213–20224. IEEE, 2023.

Ying Wang, Oumayma Bounou, Gaoyue Zhou, Randall Balestriero, Tim G. J. Rudner, Yann LeCun, and Mengye Ren. Temporal straightening for latent planning. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=Ik1mKtUYlZ.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *International Conference on Learning Representations*, volume 2024, pp. 10641–10662, 2024a.

Kun Wu, Chengkai Hou, Jiaming Liu, Zhengping Che, Xiaozhu Ju, Zhuqin Yang, Meng Li, Yinuo Zhao, Zhiyuan Xu, Guang Yang, et al. Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. *arXiv preprint arXiv:2412.13877*, 2024b.

Shihan Wu, Xuecheng Liu, Shaoxuan Xie, Pengwei Wang, Xinghang Li, Bowen Yang, Zhe Li, Kai Zhu, Hongyu Wu, Yiheng Liu, et al. Robocoin: An open-sourced bimanual robotic data collection for integrated manipulation. *arXiv preprint arXiv:2511.17441*, 2025.

Dejie Yang, Zijing Zhao, and Yang Liu. Planllm: Video procedure planning with refinable large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 9166–9174, 2025.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Naveed Malik, Kyungmin Lee, William Liang, Nadun Ranawaka Arachchige, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Danfei Xu, Yilun Du, Ryan Julian, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi Fan, and Joel Jang. World action models are zero-shot policies. In *ICLR 2026 the 2nd Workshop on World Models: Understanding, Modelling and Scaling*, 2026. URL https://openreview.net/forum?id=cd33uUB609.

Hongxiang Zhao, Xingchen Liu, Mutian Xu, Yiming Hao, Weikai Chen, and Xiaoguang Han. Taste-rob: Advancing video generation of task-oriented hand-object interaction for generalizable robotic manipulation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27683–27693. IEEE, 2025a.

Zhenyu Zhao, Hongyi Jing, Xiawei Liu, Jiageng Mao, Abha Jha, Hanwen Yang, Rong Xue, Sergey Zakharov, Vitor Guizilini, and Yue Wang. Humanoid everyday: A comprehensive robotic dataset for open-world humanoid manipulation. *arXiv preprint arXiv:2510.08807*, 2025b.

Gaoyue Zhou, Hengkai Pan, Yann Lecun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79115–79135. PMLR, 13–19 Jul 2025a. URL https://proceedings.mlr.press/v267/zhou25t.html.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: world models on pre-trained visual features enable zero-shot planning. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025b.