A LIGHTWEIGHT LIBRARY FOR ENERGY-BASED JOINT-EMBEDDING PREDICTIVE ARCHITECTURES

Basile Terver[1,2], **Randall Balestriero**[1], **Megi Dervishi**[1], **David Fan**[1], **Quentin Garrido**[1], **Tushar Nagarajan**[1], **Koustuv Sinha**[1], **Wancong Zhang**[1], **Mike Rabbat**[1], **Yann LeCun**[1,3,†], **Amir Bar**[1,†]
[1]Meta FAIR [2]INRIA [3]New York University
[†]Equal Contribution

ABSTRACT

We present **EB-JEPA**, an open-source library for learning representations and world models using Joint-Embedding Predictive Architectures (JEPAs). JEPAs learn to predict in representation space rather than pixel space, avoiding the pitfalls of generative modeling while capturing semantically meaningful features suitable for downstream tasks. Our library provides modular, self-contained implementations that illustrate how representation learning techniques developed for image-level self-supervised learning can transfer to video, where temporal dynamics add complexity, and ultimately to action-conditioned world models, where the model must additionally learn to predict the effects of control inputs. Each example is designed for single-GPU training within a few hours, making energy-based self-supervised learning accessible for research and education. We provide ablations of JEA components on CIFAR-10. Probing these representations yields 91% accuracy, indicating that the model learns useful features. Extending to video, we include a multi-step prediction example on Moving MNIST that demonstrates how the same principles scale to temporal modeling. Finally, we show how these representations can drive action-conditioned world models, achieving a 97% planning success rate on the Two Rooms navigation task. Comprehensive ablations reveal the critical importance of each regularization component for preventing representation collapse. [1]

1 INTRODUCTION

The idea that intelligent systems should learn internal models of their environment has deep roots in cognitive science, from early theories of mental models (Craik, 1967) to predictive coding accounts of perception (Rao & Ballard, 1999) and learned world models for planning (Sutton, 1991; Schmidhuber, 1990). Recent advances in video generation (Brooks et al., 2024; Blattmann et al., 2023) and interactive world simulators (Bruce et al., 2024; Parker-Holder et al., 2024) have shown impressive results, but those face fundamental challenges: they must model all pixels, *including task-irrelevant details*, thereby requiring substantial computational resources (Balestriero & Lecun, 2024). Joint-Embedding Predictive Architectures (JEPAs) (LeCun, 2022; Assran et al., 2023; Bardes et al., 2024) offer an alternative paradigm. Rather than reconstructing observations in pixel space, JEPAs learn to predict in a learned representation space, focusing computational effort on semantically meaningful features.

JEPA builds on a rich history of self-supervised representation learning (Chen et al., 2020; He et al., 2020; Grill et al., 2020; Zbontar et al., 2021; Chen & He, 2021). JEPAs have demonstrated strong performance for visual representation learning (Assran et al., 2023) and have been extended to video understanding (Bardes et al., 2024) and world modeling for planning (Assran et al., 2025; Sobal et al., 2025; Zhou et al., 2024a; Terver et al., 2026). Despite this growing body of work, accessible implementations that bridge theoretical principles and practical application remain scarce. Production-scale implementations are designed for large-scale training and are challenging to navigate. World model implementations like DINO-WM (Zhou et al., 2024a) and JEPA-WMs (Terver et al.,

[1]Code is available at `https://github.com/facebookresearch/eb_jepa`.

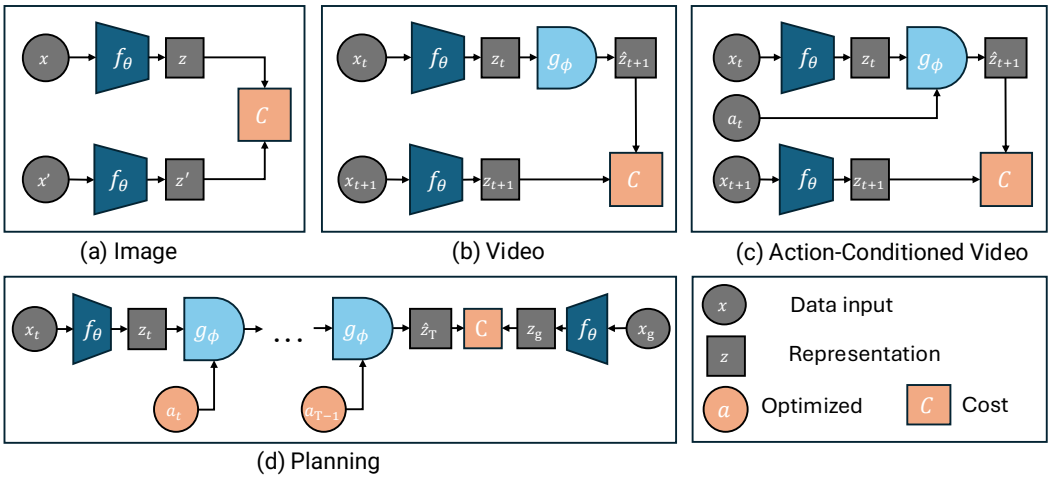


Figure 1: **EB-JEPA** is a modular code base and tutorial, providing self-contained implementations of Joint-Embedding Predictive Architecture for (a) self-supervised image representation learning, (b) video prediction in latent space, and (c) action-conditioned world models that enable goal-directed planning (d).

2026) enable planning on simple environments but rely on particular setups, e.g., frozen pre-trained encoders. As a result, while JEPAs have shown promise, they still have a high barrier to entry, which we hope to address in this study. This paper introduces **EB-JEPA** (Figure 1), an open-source library that addresses this gap through modular, well-documented implementations of JEPA-based models trainable at small scale with simple, concise code designed for educational purposes and rapid experimentation. Our contributions are:

1. **Accessible implementations**: Three progressively complex examples (image representation learning, video prediction, and action-conditioned planning), each trainable on a single GPU in a few hours.
2. **Modular architecture**: Reusable components (encoders, predictors, regularizers, planners) that can be easily recombined for new applications.
3. **Comprehensive evaluation**: Systematic experiments and ablations demonstrating the importance of each component, with practical guidance on hyperparameter selection.
4. **Educational resource**: Clear documentation and code structure designed to help researchers understand JEPA principles.

2 Related Work

Joint-Embedding methods. EB-JEPA builds on the JEPA framework (Assran et al., 2023; Bardes et al., 2024), focusing on their subclass using regularization-based collapse prevention (Bardes et al., 2022; Zbontar et al., 2021; Balestriero & LeCun, 2025) rather than stop-gradient techniques (Grill et al., 2020; Chen & He, 2021; Oquab et al., 2024). Recent theoretical work has provided deeper understanding of these methods: Shwartz-Ziv et al. (2023) analyze VICReg from an information-theoretic perspective, while Balestriero & LeCun (2022) show connections between contrastive and non-contrastive methods and spectral embedding. While I-JEPA and V-JEPA focus on masked prediction within single images or videos, our action-conditioned example extends this to interactive settings where actions determine future states. Recent work has shown that JEPA-style pretraining leads to emergent understanding of intuitive physics (Garrido et al., 2025), motivating the use of such architectures for world modeling. Importantly, JEPAs differ fundamentally from reconstruction-based methods such as MAE (He et al., 2021) and VideoMAE (Tong et al., 2022; Wang et al., 2023), which predict in pixel space rather than representation space. Balestriero & Lecun (2024) provide theoretical analysis showing that reconstruction-based learning can produce uninformative features for perception, further motivating the joint-embedding paradigm that our library focuses on.

World models for planning. Latent world models have been extensively studied for model-based reinforcement learning (RL) (Hafner et al., 2019; 2024; Hansen et al., 2024). Our work is most

closely related to PLDM (Sobal et al., 2025), IWM (Garrido et al., 2024), DINO-WM (Zhou et al., 2024a), Navigation World Models (Bar et al., 2025), and JEPA-WMs (Terver et al., 2026), which use joint-embedding objectives for planning. Unlike these works, we focus on providing accessible, educational implementations rather than state-of-the-art performance on complex benchmarks.

3 Preliminaries: A Unified JEPA Framework

Our goal is to train models that map inputs to latent semantic representations useful for perception, planning, and action. We view this through the lens of *Energy-Based Models* (EBMs) (LeCun et al., 2006; Hopfield, 1982). An EBM defines a scalar energy function $E(x, y)$ measuring compatibility between inputs x and outputs y, where low energy indicates high compatibility. Learning consists of shaping the energy landscape so that correct input-output pairs have lower energy than incorrect ones.

The key challenge in training EBMs is preventing *collapse*: a degenerate solution where the energy is uniformly low for all inputs. Different strategies address this challenge: contrastive methods push up energy on negative samples (Hinton, 2002; Chen et al., 2020; He et al., 2020); stop-gradient and exponential moving average (EMA) techniques break symmetry (Grill et al., 2020; Chen & He, 2021; Assran et al., 2023; Bardes et al., 2024); and regularization-based approaches maintain representation diversity without negative samples (Bardes et al., 2022; Zbontar et al., 2021; Balestriero & LeCun, 2025). Our library focuses on the regularization approach: we instantiate JEPAs with explicit regularization losses to prevent collapse, defining energy as prediction error in representation space. With the regularizer $\mathcal{R}$ and a given prediction loss $\mathcal{L}_{\text{pred}}$, the JEPA general training objective takes the form

$$\mathcal{L} = \mathcal{L}_{\text{pred}}(g_\phi(z, u), z') + \lambda \mathcal{R}(z), \tag{1}$$

where $z = f_\theta(x)$ is the representation of input x, $u = q_\omega(a)$ is optional conditioning information (e.g., robotic controls), z' is the target representation, and λ balances prediction and regularization. This unified framework encompasses three instantiations of increasing complexity, detailed below.

(i) Image-JEPA: view invariance. Given an image x, we create two views x and x' (random crops, color jittering, etc.). The encoder produces representations $z = f_\theta(x)$ and $z' = f_\theta(x')$. The objective learns representations invariant to different views, with the energy function

$$\mathcal{L}_{\text{image}} = \|z - z'\|_2^2 + \lambda \mathcal{R}(z, z'). \tag{2}$$

Here, the energy directly measures how similar the representations of two views of the same image are. Low energy means the model has learned view-invariant features.

(ii) Video-JEPA: temporal prediction. We denote a video sequence as $x_{1:T} := (x_1, \ldots, x_T)$. The encoder produces per-frame representations $z_t = f_\theta(x_{t-w:t})$, where w is the encoder temporal receptive field. A predictor takes a context of $v + 1$ frame representations, where v is the predictor temporal receptive field (see hyperparameter values in Tab. 6), and predicts the next representation, yielding the energy

$$\mathcal{L}_{\text{video}} = \sum_{t=1}^{T-1} \|g_\phi(z_{t-v:t}) - z_{t+1}\|_2^2 + \lambda \mathcal{R}(z_{1:T}). \tag{3}$$

The model learns to capture temporal dynamics without access to future frames during prediction.

(iii) Action-conditioned video-JEPA (AC-video-JEPA): world modeling. Given observation-action sequences $(x_t, a_t)_{t=1}^T$, an action encoder q_ω maps actions to control representations $u_t = q_\omega(a_{t-w:t})$, and the predictor is conditioned on these representations, yielding the energy

$$\mathcal{L}_{\text{world}} = \sum_{t=1}^{T-1} \|g_\phi(z_{t-v:t}, u_{t-v:t}) - z_{t+1}\|_2^2 + \lambda \mathcal{R}(z_{1:T}, u_{1:T}). \tag{4}$$

This learns a latent dynamics model suitable for planning: given a current state and control representation, predict the next state representation.

A unified energy formulation. The three settings above share a common structure. Given an encoder f_θ, a predictor g_ϕ, and optional conditioning a with conditioning encoder q_ω, we can write a general energy function

$$E(x, x', a) = \mathcal{L}_{\text{pred}}(g_\phi(f_\theta(x), q_\omega(a)), f_\theta(x')). \tag{5}$$

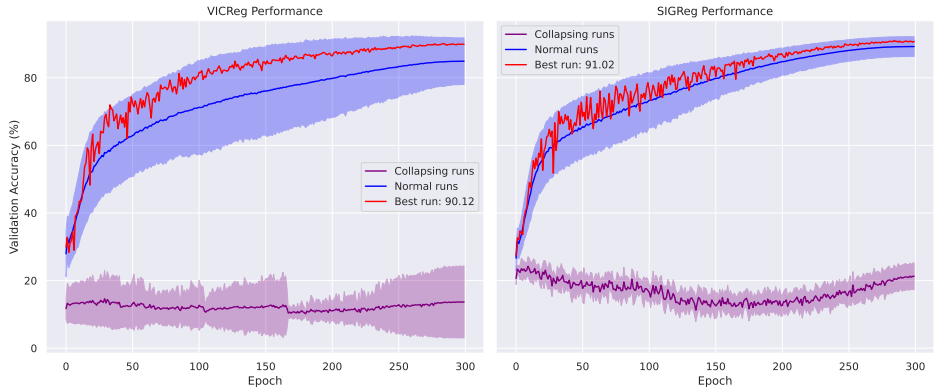


Figure 2: Hyperparameter sensitivity comparison between SIGReg and VICReg on CIFAR-10. SIGReg demonstrates greater stability across different hyperparameter configurations, while VICReg achieves similar peak performance but requires more careful tuning.

Image-JEPA corresponds to $g_\phi = \text{Id}$ (identity) and no conditioning; video-JEPA uses a temporal predictor without conditioning; AC-video-JEPA includes the full formulation with action conditioning. This unified view highlights how the same energy-based principle, minimizing prediction error in representation space, underlies all three settings, with complexity increasing as we move from static images to video to action-conditioned dynamics.

Regularization: Preventing Collapse. The key challenge in training JEPAs is preventing *representation collapse*, where the encoder learns trivial constant representations. EB-JEPA implements two regularization families. *VICReg* (Bardes et al., 2022) prevents collapse through two complementary terms. The *variance* term ensures each feature dimension has sufficient spread across the batch and reads

$$\mathcal{L}_{\text{var}}(Z) = \frac{1}{D}\sum_{j=1}^{D} \max\left(0, \gamma - \sqrt{\text{Var}(Z_{:,j}) + \epsilon}\right), \tag{6}$$

where $Z \in \mathbb{R}^{N\times D}$ is the batch of embeddings, D is the feature dimension, and γ is the target standard deviation (typically 1). The *covariance* term decorrelates feature dimensions to encourage the model to use all available capacity and reads

$$\mathcal{L}_{\text{cov}}(Z) = \frac{1}{D(D-1)}\sum_{i\neq j}[C(Z)]_{i,j}^2, \quad C(Z) = \frac{1}{N-1}(Z-\bar{Z})^\top(Z-\bar{Z}). \tag{7}$$

The full VICReg regularizer is $\mathcal{R}_{\text{VICReg}} = \alpha\mathcal{L}_{\text{var}} + \beta\mathcal{L}_{\text{cov}}$.

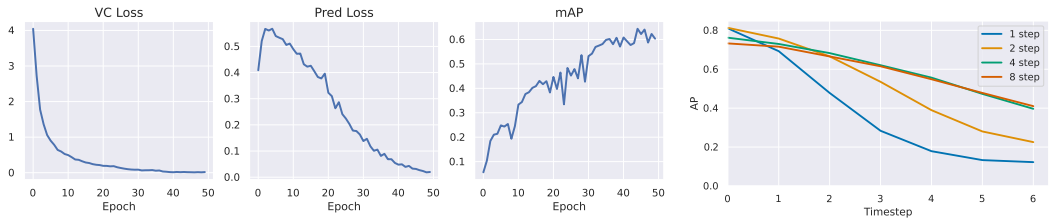
For image-JEPA and video-JEPA, the regularization losses are computed in a projected space rather than directly on the encoder outputs. A learned projector h_ψ maps *representations* to *embeddings* $r = h_\psi(z)$ on which the regularizer is computed. *LeJEPA* (Balestriero & LeCun, 2025) introduces SIGReg, a theoretically grounded alternative regularizer. It identifies the isotropic Gaussian $\mathcal{N}(0, I)$ as the optimal embedding distribution for minimizing downstream prediction risk. The SIGReg objective enforces this by testing Gaussianity along random 1D projections $\xi_p \sim \mathcal{N}(0, I)$ and reads

$$\mathcal{R}_{\text{SIGReg}}(Z) = \frac{1}{P}\sum_{p=1}^{P}\mathcal{G}(Z\xi_p), \tag{8}$$

where $\mathcal{G}$ is the Epps-Pulley Gaussianity test statistic. This approach offers a single hyperparameter λ, linear time/memory complexity, and stability across architectures.

4 Training and Planning with World Models

Multistep Rollout Training. In practice, for both video JEPA and Action-Conditioned JEPA, we augment single-step prediction with multistep rollout losses, following Terver et al. (2026); Assran



(a) (b)

Figure 3: Video-JEPA training dynamics and multistep rollout ablation. (a) Training dynamics over 50 epochs: variance-covariance regularization loss $\mathcal{R}$ (left), prediction loss $\mathcal{L}_{\text{pred}}$ (center), and mean Average Precision (right). (b) Training with k-step recursive predictions achieves significantly better Average Precision compared to single-step predictions, demonstrating improved temporal understanding, with a Pareto optimum around $k = 4$ rollout steps.

Figure 4: Video JEPA visualization on Moving MNIST. From left to right: input frames, 1-step prediction visualization, and full autoregressive rollout. The model maintains coherent predictions of digit motion over extended horizons, correctly capturing trajectory and dynamics.

et al. (2025). At each training iteration, we compute k-step rollout losses $\mathcal{L}_k$ for $k = 1, \ldots, K$, where $\mathcal{L}_1$ recovers the single-step loss of Eqs. (3)–(4). Let us define the order of a prediction as the number of calls to the predictor function required to obtain it from a groundtruth representation. For a predicted representation $z_t^{(k)}$, we denote the timestep it corresponds to as t and its prediction order as k, with $z^{(0)} = z = f_\theta(x)$. For $k \geq 1$, $\mathcal{L}_k$ is defined as

$$\mathcal{L}_k = \sum_{t=1}^{T-k} \|g_\phi(z_{t-v:t}^{(k-1)}, u_{t-v:t}) - z_{t+1}\|_2^2, \tag{9}$$

where $z_t^{(k)}$ is obtained by recursively unrolling the predictor for all $t \leq T$, as

$$z_{t+1}^{(k)} = g_\phi(z_{t-v:t}^{(k-1)}, u_{t-v:t}), \quad z_t^{(0)} = f_\theta(x_{t-w:t}). \tag{10}$$

The total energy function losses of Eqs. (3)–(4) then read

$$\mathcal{L}_{\text{video}} = \mathcal{L}_{\text{pred}} + \lambda \mathcal{R}(z_{1:T}), \quad \mathcal{L}_{\text{world}} = \mathcal{L}_{\text{pred}} + \lambda \mathcal{R}(z_{1:T}, u_{1:T}), \quad \mathcal{L}_{\text{pred}} = \sum_{k=1}^{K} \mathcal{L}_k. \tag{11}$$

Note that we could perform truncated backpropagation through time (TBPTT) (Jaeger, 2002), detaching the gradient after each call to the predictor. Training with k-step rollouts aligns the training procedure with autoregressive inference, reducing exposure bias and improving long-horizon prediction quality (see Figure 3).

Additional Regularizers for World Models. Training action-conditioned JEPAs in randomized environments requires additional regularization beyond VICReg or SIGReg terms. The temporal similarity loss $\mathcal{L}_{\text{sim}}$ encourages smooth representation trajectories along action sequences, and the inverse dynamics model (IDM) loss (Pathak et al., 2017) $\mathcal{L}_{\text{IDM}}$ predicts actions from consecutive representations. These losses read

$$\mathcal{L}_{\text{sim}} = \sum_t \|z_t - z_{t+1}\|_2^2, \quad \mathcal{L}_{\text{IDM}} = \sum_t \|a_t - \text{MLP}(z_t, z_{t+1})\|_2^2. \tag{12}$$

Table 1: Image-JEPA Linear probing accuracy on CIFAR-10 with ResNet-18 backbone trained for 300 epochs, comparing regularizers (SIGReg and VICReg) and the impact of using a projector.

	Best acc. ↑	Average acc. ↑	w/o Projector	Hyperparams	Best projector
SIGReg	91.02%	89.22%	-3.3 points	1	2048×128
VICReg	90.12%	84.90%	-2.9 points	2	2048×1024

This term is critical for preventing collapse from spurious correlations in randomized environments (Sobal et al., 2022). The full training objective for action-conditioned video-JEPA combines prediction with all regularization terms and reads

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \alpha\mathcal{L}_{\text{var}} + \beta\mathcal{L}_{\text{cov}} + \delta\mathcal{L}_{\text{sim}} + \omega\mathcal{L}_{\text{IDM}}. \tag{13}$$

Goal-Conditioned Planning. We perform goal-conditioned planning by optimizing action sequences to reach a goal observation x_g. This extends the energy function from Eq. (5) to trajectories: rather than measuring prediction error for a single step, we accumulate the energy over an imagined rollout towards the goal as

$$E_{\text{plan}}(a_{0:H}; x_0, x_g) = \sum_{t=1}^{H} \|f_\theta(x_g) - \hat{z}_t\|_2, \quad \text{where } \hat{z}_{t+1} = g_\phi(\hat{z}_{t-v:t}, u_{t-v:t}), \quad \hat{z}_0 = f_\theta(x_0). \tag{14}$$

Low energy corresponds to action sequences that successfully reach the goal; planning thus reduces to finding the minimum-energy trajectory. We use Model Predictive Path Integral (MPPI) (Williams et al., 2015), a population-based optimizer that samples action trajectories, weights them by exponentiated negative energy (i.e., a Boltzmann distribution over trajectories), and iteratively refines the proposal distribution toward lower-energy solutions. Summing over intermediate states (rather than only the final state) encourages efficient paths and provides robustness to prediction compounding errors.

5 Experiments

Experimental Setup. We evaluate the JEPA framework on three tasks of increasing complexity: image representation learning on CIFAR-10, video prediction on Moving MNIST (Srivastava et al., 2015), and goal-conditioned planning on the Two Rooms environment (Sobal et al., 2025). Our implementation uses modular building blocks: **Encoders** (ResNet-18 (He et al., 2016), Vision Transformers (Dosovitskiy et al., 2021), IMPALA (Espeholt et al., 2018)), **Predictors** (UNet-based spatial predictors, GRU-based temporal predictors), **Regularizers** (VICReg, SIGReg, temporal similarity, inverse dynamics losses), and **Planners** (MPPI (Williams et al., 2015) and Cross-Entropy Method (CEM) optimizers). We provide comprehensive hyperparameter tables in Appendix A: Tables 5 and 6 summarize the best training hyperparameters for each example, and Table 7 details the planning configuration.

Image Representation Learning. Tables 1, 2, and 3 compare VICReg and SIGReg on CIFAR-10, using a naive hyperparameter search. Both methods achieve approximately 90-91% linear probing accuracy, competitive with prior self-supervised methods on this benchmark. We find that using a learned projector provides around a 3 point improvement over directly regularizing encoder outputs. Projector architecture matters: a bottleneck design (large hidden → small output) works best for SIGReg, while VICReg prefers larger output dimensions. Having only one hyperparameter, SIGReg can be easier to tune in this naive setting (Figure 2).

Video Prediction. Multi-step autoregressive rollouts on Moving MNIST maintain prediction quality over extended horizons (Figure 4). Training with k-step prediction (rather than single-step) significantly improves Average Precision on downstream detection tasks by reducing exposure bias, i.e., the discrepancy between teacher-forced training and autoregressive inference. Figure 3 shows that models trained with longer prediction horizons achieve better downstream performance, as recursive prediction during training aligns with the autoregressive inference procedure.

Table 2: Ablation of Image-JEPA on loss hyperparameters when training on CIFAR-10 with ResNet-18 backbone trained for 300 epochs.

	SIGReg		VICReg	
Rank	Hyperparameters	Accuracy ↑	Hyperparameters	Accuracy ↑
1	$\lambda = 10$	90.88%	std = 1, cov = 100	90.12%
2	$\lambda = 1$	86.94%	std = 1, cov = 10	89.93%
3	$\lambda = 100$	80.86%	std = 10, cov = 10	89.20%
-1	$\lambda = 0.1$	27.20%	std = 100, cov = 100	10.00%

Table 3: Image-JEPA ablation of projector design when training on CIFAR-10 with ResNet-18 backbone trained for 300 epochs.

	SIGReg		VICReg	
Rank	Dimensions	Accuracy ↑	Dimensions	Accuracy ↑
1	2048×128	91.02%	2048×1024	90.12%
2	4096×1024	91.00%	4096×512	90.10%
3	2048×64	90.99%	1024×1024	90.05%
4	512×256	90.99%	2048×512	90.03%
5	4096×64	90.96%	4096×1024	90.02%
N/A	None	87.75%	None	87.27%

Action-Conditioned Video-JEPA. We display three successful planning evaluation episodes in Figure 5, showing the ability of the model to plan given randomized initial and goal states. This navigation task is non-monotonic, meaning that the optimal trajectory requires first getting farther from the goal, in order to reach it ultimately. Table 4 shows planning results on the challenging random-wall setup. Our best model achieves 97% success rate using MPPI with cumulative cost over the planning horizon.

We perform an **ablation of the regularization components** of the action-conditioned video-JEPA models. Table 4 reveals the importance of each regularization component: IDM is critical (without it, the model collapses to 1% success due to spurious correlations (Sobal et al., 2022)); variance and covariance terms each contribute ∼50% absolute improvement; temporal similarity adds ∼35%.

We ablate the importance of **planning cost design**. Using cumulative cost over all timesteps ($\sum_t \|z_g - \hat{z}_t\|$) outperforms final-state-only cost by 8% (Table 4). This formulation encourages efficient paths and provides gradient signal throughout the trajectory.

6 Future Directions

EB-JEPA is designed for fast iteration on algorithmic innovations at small scale: single-GPU training, simple datasets, and controlled simulated environments. This enables rapid prototyping and fundamental research on JEPA architectures before scaling to more complex settings. We identify three promising algorithmic directions that EB-JEPA's modular design enables researchers to explore.

Advancing Regularization Theory. Our experiments highlight the critical role of regularization in preventing representation collapse, yet the theoretical understanding of why certain regularization combinations work remains incomplete. EB-JEPA provides a testbed for systematically studying regularization dynamics: investigating the interplay between variance, covariance, temporal similarity, and inverse dynamics terms (Bardes et al., 2022; Balestriero & LeCun, 2025; Sobal et al., 2022); understanding when each becomes necessary; and developing principled methods for automatic hyperparameter selection. The controlled, single-GPU setting enables rapid iteration on these fundamental questions without the confounding factors introduced by large-scale distributed training.

Table 4: AC-video-JEPA planning ablations on Two Rooms with randomized wall positions. Each result averages over 3 seeds $\times$ 3 checkpoints $\times$ 20 episodes. **Left:** Planner configuration comparison. **Right:** Regularization term ablation; removing IDM causes collapse.

Configuration	Success ↑	Time
MPPI (full cost)	$97 \pm 2\%$	37s
CEM (full cost)	$96 \pm 2\%$	37s
MPPI (last state)	$89 \pm 2\%$	37s

Ablated Term	Success ↑
None (full model)	$97 \pm 2\%$
Variance ($\alpha = 0$)	$47 \pm 3\%$
Covariance ($\beta = 0$)	$46 \pm 3\%$
Temporal Sim. ($\delta = 0$)	$61 \pm 2\%$
IDM ($\omega = 0$)	$1 \pm 1\%$

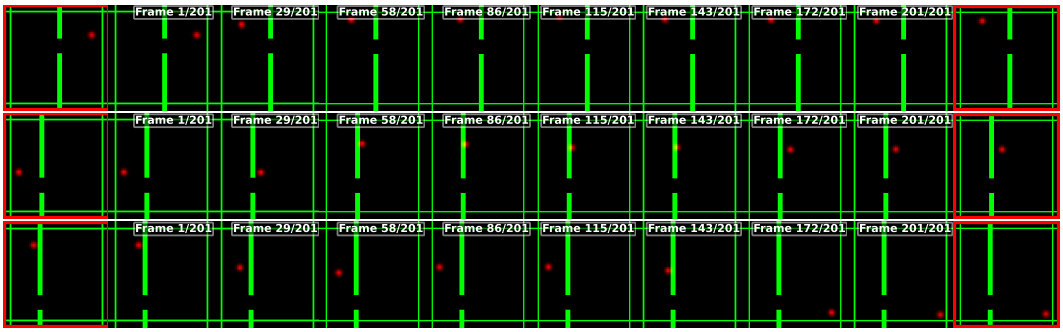


Figure 5: Visualization of three successful planning evaluation episodes of our AC-video-JEPA on the Two Rooms environment with random wall. From left to right: initial frame (red), full episode outputted by the planning optimization procedure, goal frame used to define planning cost (red). Each episode allows a maximum of 200 steps in the environment.

Hierarchical World Models. Current JEPA models predict at a single temporal resolution, but intelligent planning often requires reasoning at multiple timescales (Schmidhuber, 2015; Hafner et al., 2022). Hierarchical world models could learn to predict both fine-grained dynamics for local control and coarse-grained abstractions for long-horizon planning. Prior work in hierarchical reinforcement learning (Nachum et al., 2018; Levy et al., 2019) has demonstrated the benefits of learning at multiple levels of abstraction. EB-JEPA's separation of encoder, predictor, and regularizer components provides a natural starting point for implementing such multi-scale architectures, and future releases may include basic hierarchical prediction examples.

Learned Cost and Value Functions. Our current planning approach uses simple distance-based costs in representation space, but this may be suboptimal for complex tasks. Learning task-specific cost functions or value functions from demonstrations or sparse rewards could enable more sophisticated goal-directed behavior. Combining JEPA world models with learned value functions (Hansen et al., 2022; 2024) offers a promising avenue for making better use of the predictive models trained with this codebase, potentially bridging the gap between pure world modeling and reward-driven reinforcement learning. EB-JEPA's simple planning interface makes it straightforward to experiment with alternative cost formulations.

Complementary to Large-Scale Codebases. EB-JEPA is intended for algorithmic exploration and fundamental research. Once promising approaches are validated at small scale, researchers can transition to codebases supporting distributed training, pre-trained visual backbones, and more complex environments, such as JEPA-WMs (Terver et al., 2026) for planning with frozen encoders on diverse benchmarks, and `stable-pretraining` (Balestriero et al., 2025) for general self-supervised learning (SSL) pretraining. EB-JEPA complements these by prioritizing educational simplicity and self-contained, single-GPU examples. This two-stage workflow enables efficient research: rapid prototyping with EB-JEPA followed by rigorous evaluation at scale.

7 CONCLUSION

We have presented EB-JEPA, an open-source library for learning representations and world models using Joint-Embedding Predictive Architectures. Our implementations span image representation learning, video prediction, and action-conditioned planning, each designed to be trainable on a single GPU within a few hours. Comprehensive experiments demonstrate that our implementations achieve strong results on established benchmarks while providing insights into the importance of each component. The ablation studies reveal that all regularization terms (variance, covariance, temporal similarity, and inverse dynamics) play important roles in preventing collapse and enabling effective planning. We hope EB-JEPA serves as both a practical toolkit for researchers exploring JEPA-based methods and an educational resource for understanding energy-based self-supervised learning.

ETHICS STATEMENT

EB-JEPA is an educational library for self-supervised learning research. All experiments use standard public benchmarks (CIFAR-10, Moving MNIST) or procedurally generated environments (Two Rooms). None of these datasets contain personally identifiable information. We see no direct ethical concerns with this work.

REPRODUCIBILITY STATEMENT

Reproducibility is the central goal of this work. Our full codebase is included in the supplementary material, with all training scripts, model implementations, and evaluation code. Each example is self-contained and trains on a single GPU in a few hours, removing the need for large compute clusters. Hyperparameters for all experiments are listed in Appendix A. The Two Rooms environment is procedurally generated with documented seeds.

ACKNOWLEDGMENTS

We thank Adrien Bardes and Gaoyue Zhou for participating in the discussions and conceptualization of the project.

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 *CVPR*, 2023.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba, Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning, 2025.

Randall Balestriero and Yann LeCun. Contrastive and non-contrastive self-supervised learning recover global and local spectral embedding methods. In *NeurIPS*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Randall Balestriero and Yann Lecun. How learning by reconstruction produces uninformative features for perception. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *ICML*, volume 235 of *Proceedings of Machine Learning Research*, pp. 2566–2585. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/balestriero24b.html.

Randall Balestriero and Yann LeCun. Lejepa: Provable and scalable self-supervised learning without the heuristics, 2025. URL https://arxiv.org/abs/2511.08544.

Randall Balestriero, Hugues Van Assel, Sami BuGhanem, and Lucas Maes. stable-pretraining-v1: Foundation model research made simple, 2025. URL https://arxiv.org/abs/2511.19484.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *CVPR*, pp. 15791–15801, June 2025.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets, 2023. URL https://arxiv.org/abs/2311.15127.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators, 2024. URL https://openai.com/research/video-generation-modelsas-world-simulators.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *ICML*, 2024.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, pp. 02783649241273668, 2023.

Kenneth James Williams Craik. *The nature of explanation*, volume 445. CUP Archive, 1967.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. IMPALA: Scalable distributed deep-RL with importance weighted actor-learner architectures. In Jennifer Dy and Andreas Krause (eds.), *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1407–1416. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/espeholt18a.html.

Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning, 2024.

Quentin Garrido, Nicolas Ballas, Mahmoud Assran, Adrien Bardes, Laurent Najman, Michael Rabbat, Emmanuel Dupoux, and Yann LeCun. Intuitive physics understanding emerges from self-supervised pretraining on natural videos, 2025. URL https://arxiv.org/abs/2502.11831.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *ICML*, volume 97, pp. 2555–2565. PMLR, 2019.

Danijar Hafner, Kuang-Huei Lee, Ian Fischer, and Pieter Abbeel. Deep hierarchical planning from pixels. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *NeurIPS*, 2022.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models, 2024.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *ICLR*, 2024.

Nicklas A Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8387–8406. PMLR, 17–23 Jul 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2021.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. volume 14, pp. 1771–1800, Cambridge, MA, USA, August 2002. MIT Press. doi: 10.1162/089976602760128018. URL https://doi.org/10.1162/089976602760128018.

J J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982. doi: 10.1073/pnas.79.8.2554. URL https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554.

Herbert Jaeger. Tutorial on training recurrent neural networks, covering bppt, rtrl, ekf and the echo state network approach. *GMD-Forschungszentrum Informationstechnik, 2002.*, 5, 01 2002.

Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *ICML*, 2022.

Yann LeCun. A path towards autonomous machine intelligence. *Open Review*, Jun 2022.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. A tutorial on energy-based learning. 2006.

Andrew Levy, Robert Platt, and Kate Saenko. Hierarchical reinforcement learning with hindsight. In *ICLR*, 2019.

Ofir Nachum, Shixiang (Shane) Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *NeurIPS*, volume 31. Curran Associates, Inc., 2018.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Jack Parker-Holder, Philip Ball, Jake Bruce, Vibhavari Dasagi, Kristian Holsheimer, Christos Kaplanis, Alexandre Moufarek, Guy Scully, Jeremy Shar, Jimmy Shi, Stephen Spencer, Jessica Yung, Michael Dennis, Sultan Kenjeyev, Shangbang Long, Vlad Mnih, Harris Chan, Maxime Gazeau, Bonnie Li, Fabio Pardo, Luyu Wang, Lei Zhang, Frederic Besse, Tim Harley, Anna Mitenkova, Jane Wang, Jeff Clune, Demis Hassabis, Raia Hadsell, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 2: A large-scale foundation world model. 2024. URL https://deepmind.google/discover/blog/genie-2-a-large-scale-foundation-world-model/.

Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *ICML*, ICML'17, pp. 2778–2787. JMLR.org, 2017.

R. P. Rao and D. H. Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature neuroscience*, 2(1):79–87, January 1999. ISSN 1097-6256. doi: 10.1038/4580. URL http://dx.doi.org/10.1038/4580.

Juergen Schmidhuber. On learning to think: Algorithmic information theory for novel combinations of reinforcement learning controllers and recurrent neural world models, 2015. URL https://arxiv.org/abs/1511.09249.

Jurgen Schmidhuber. Making the world differentiable: on using self supervised fully recurrent neural networks for dynamic reinforcement learning and planning in non-stationary environments. *Forschungsberichte, TU Munich*, FKI 126 90:1–26, 1990. URL https://api.semanticscholar.org/CorpusID:28490120.

Ravid Shwartz-Ziv, Randall Balestriero, Kenji Kawaguchi, Tim G. J. Rudner, and Yann LeCun. An information theory perspective on variance-invariance-covariance regularization. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *NeurIPS*, volume 36, pp. 33965–33998. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6b1d4c03391b0aa6ddde0b807a78c950-Paper-Conference.pdf.

Vlad Sobal, Jyothir S V, Siddhartha Jalagam, Nicolas Carion, Kyunghyun Cho, and Yann LeCun. Joint embedding predictive architectures focus on slow features, 2022. URL https://arxiv.org/abs/2211.10831.

Vlad Sobal, Wancong Zhang, Kyunghyun Cho, Randall Balestriero, Tim Rudner, and Yann LeCun. Learning from reward-free offline data: A case for planning with latent dynamics models, 02 2025.

Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using lstms. In *ICML*, ICML'15, pp. 843–852. JMLR.org, 2015.

Richard S. Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *SIGART Bull.*, 2(4):160–163, July 1991. ISSN 0163-5719. doi: 10.1145/122344.122377. URL https://doi.org/10.1145/122344.122377.

Basile Terver, Tsung-Yen Yang, Jean Ponce, Adrien Bardes, and Yann LeCun. What drives success in physical planning with joint-embedding predictive world models?, 2026. URL https://arxiv.org/abs/2512.24497.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *NeurIPS*, volume 35, pp. 10078–10093. Curran Associates, Inc., 2022.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *CVPR*, 2023.

Grady Williams, Andrew Aldrich, and Evangelos Theodorou. Model predictive path integral control using covariance variable importance sampling, 2015.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning, 2024a. URL https://arxiv.org/abs/2411.04983.

Guangyao Zhou, Sivaramakrishnan Swaminathan, Rajkumar Vasudeva Raju, J. Swaroop Guntupalli, Wolfgang Lehrach, Joseph Ortiz, Antoine Dedieu, Miguel Lázaro-Gredilla, and Kevin Murphy. Diffusion model predictive control. *arXiv preprint arXiv:2410.05364*, 2024b.

A HYPERPARAMETERS

This section provides the hyperparameters used for training and evaluation across our examples. Tables 5 and 6 summarize the key training hyperparameters, including the number of rollout steps K used for multistep prediction training (Eq. (9)) and the trajectory slice length T for temporal examples. Table 7 details the MPPI planning configuration used for goal-conditioned navigation in the action-conditioned video-JEPA example.

Table 5: Training hyperparameters for image-JEPA examples on CIFAR-10. The "ViT Image-JEPA" column documents hyperparameters for a ViT-based example provided in the codebase, which achieves 87% linear probing accuracy; all image results reported in the main paper (Tables 1, 2, and 3) use the ResNet-18 backbone.

Group	**Hyperparameter**	**VICReg**	**ViT Image-JEPA**	**SIGReg**
Optimization	Optimizer	LARS	AdamW	LARS
	LR schedule	10-epoch warmup (from 3×10^{-5}) + cosine to 0		
	Learning rate	0.3	0.001	0.3
	Epochs	300	300	300
	Batch size	256	512	256
	Weight decay	10^{-4}	10^{-4}	10^{-4}
Data	Dataset	CIFAR-10	CIFAR-10	CIFAR-10
	Image size	32×32	32×32	32×32
Architecture	Encoder	ResNet-18	ViT-S	ResNet-18
	Predictor	Identity	Identity	Identity
	Encoder output dim	512	384	512
	Projector hidden dim	2048	2048	2048
	Projector output dim	2048	2048	128
	Projector layers	3	3	3
Loss	Loss type	VICReg	VICReg	SIGReg
	Variance coeff. α	1	1	–
	Covariance coeff. β	80	80	–
	SIGReg coeff. λ	–	–	10

B PLANNING ALGORITHM

We use MPPI control (Williams et al., 2015) for planning, a sampling-based optimization algorithm that uses importance sampling to iteratively refine action sequences. Unlike the CEM which fits a Gaussian to elite samples, MPPI weights all samples by their exponentiated costs, providing smoother gradient information and better exploration.

Given a trained encoder f_θ, predictor g_ϕ, and action encoder q_ω, we minimize the planning energy E_{plan} from Eq. (14) over action sequences as described in Algorithm 1.

The key differences from CEM are: (1) MPPI uses soft weighting via the exponential transform rather than hard elite selection for the update, (2) the temperature parameter τ controls the sharpness of the weight distribution, and (3) MPPI naturally handles multi-modal cost landscapes through its importance sampling formulation. In our implementation, we combine MPPI with elite selection: we first select the top Q trajectories, then apply exponential weighting only among these elites, which provides both the robustness of elite selection and the smooth gradients of importance weighting.

C EXTENDED RELATED WORK

Diffusion-Based Planning. An alternative paradigm for planning uses diffusion models to generate trajectories. Diffuser (Janner et al., 2022) pioneered planning with diffusion by treating trajectory optimization as iterative denoising. Diffusion MPC (Zhou et al., 2024b) extends this to model

Table 6: Training hyperparameters for video-JEPA examples. K denotes the number of training rollout steps (multistep prediction), and T denotes the training trajectory slice length.

Group	Hyperparameter	Video-JEPA	AC-Video-JEPA
Optimization	Learning rate	0.001	0.001
	Epochs	50	12
	Batch size	64	384
	Weight decay	–	10^{-5}
Data	Dataset	Moving MNIST	Two Rooms
	Trajectory length T	10	17
	Image size	64×64	65×65
Architecture	Encoder	ResNet-5	IMPALA
	Predictor	ResUNet	GRU
	Latent dimension d	16	32
	Hidden dimension	32	32
	Encoder receptive field w	1	1
	Predictor receptive field v	2	1
Loss	Rollout steps K	4	8
	Variance coeff. α	10	16
	Covariance coeff. β	100	8
	Time similarity coeff. δ	–	12
	IDM coeff. ω	–	1

Table 7: Planning hyperparameters for the action-conditioned video-JEPA example using MPPI, corresponding to the notations of Algorithm 1. The total number of replanning steps for an evaluation episode is $\frac{M}{m}$.

Hyperparameter	Symbol	Value
Planning horizon	H	90
Number of parallel samples	N	200
Number of iterations	J	20
Number of elites	Q	20
Noise scale	σ	2
Temperature	τ	0.005
Actions stepped per plan	m	1
Max steps per episode	M	200

predictive control settings, while Diffusion Policy (Chi et al., 2023) applies diffusion to visuomotor policy learning. These approaches complement JEPA-based methods: while diffusion models excel at generating diverse, multimodal trajectories, JEPAs provide efficient latent dynamics suitable for fast online planning.

Algorithm 1 Model Predictive Path Integral (MPPI)

Input: Initial observation x_0, goal observation x_g, initial mean $\mu \in \mathbb{R}^{H \times A}$, noise scale σ, temperature τ, number of samples N, number of iterations J, number of elites Q, max steps per episode M
Encode initial and goal: $\hat{z}_0 = f_\theta(x_0)$, $z_g = f_\theta(x_g)$
for $j = 1$ to J **do**
  Sample N noise perturbations: $\epsilon_i \sim \mathcal{N}(0, \sigma^2 \mathrm{I})$ for $i = 1, \ldots, N$
  Compute candidate action sequences: $a^{(i)}_{0:H-1} = \mu + \epsilon_i$
  Unroll predictor for each trajectory: $\hat{z}^{(i)}_{t+1} = g_\phi(\hat{z}^{(i)}_{t-v:t}, u^{(i)}_{t-v:t})$ for $t = 0, \ldots, H-1$
  Compute trajectory costs: $S_i = \sum_{t=1}^{H} \|z_g - \hat{z}^{(i)}_t\|_2$
  Select top Q elite samples with lowest costs
  Compute weights over elites: $w_i = \frac{\exp(-S_i/\tau)}{\sum_{q=1}^{Q} \exp(-S_q/\tau)}$
  Update mean: $\mu \leftarrow \sum_{i=1}^{Q} w_i \cdot a^{(i)}_{0:H-1}$
end for
Return: Execute first m actions of μ, then replan from new observation until M steps reached