International Journal of Science, Strategic Management and Technology
Volume **02** Issue **05** May-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

Scaling Laws and Architectural Advances of Hierarchical JEPA (H-JEPA) Model for Planning, Control and Robotics in Physical Systems

Mr. Mayank Lal
B.Tech (Information Technology) NIET, Greater Noida

Mr. Abdul Khalid
Assistant Professor (Information Technology) NIET, Greater Noida

https://doi.org/10.55041/ijsmt.v2i5.169

Cite this Article: Lal, M. (2026). Scaling Laws and Architectural Advances of Hierarchical JEPA (H-JEPA) Model for Planning, Control and Robotics in Physical Systems. International Journal of Science, Strategic Management and Technology, *02*(05). https://doi.org/10.55041/ijsmt.v2i5.169

Abstract—Hierarchical Joint-Embedding Predictive Architecture (H-JEPA) is increasingly viewed as a promising family of world models for embodied intelligence because it learns to predict abstract future representations rather than reconstructing raw sensory inputs. This distinction is especially important in robotics, where successful control depends less on recovering exact pixels and more on learning compact state abstractions that are stable, semantically meaningful, and useful for planning. In this paper, we present an extended student-level study of H- JEPA from three complementary angles: architectural principles, scaling behaviour, and practical deployment for robotic planning and control. We first review the conceptual line from predictive coding and world models to JEPA, I-JEPA, V-JEPA, and recent action-conditioned variants. We then formalize a two-level H- JEPA suitable for physical systems, in which low-level predictors model short-horizon action-conditioned transitions and higher- level predictors produce temporally coarse sub-goals for long- horizon planning. Next, we analyze scaling trends with respect to encoder width, predictor depth, temporal hierarchy, and dataset size, arguing that downstream planning performance follows a weak power-law regime but saturates earlier than language- model loss scaling because control success is bottlenecked by rep- resentation utility, action coverage, and model-planner mismatch. We also describe a practical pipeline that maps raw observations to latent state estimation, hierarchical rollout, cross-entropy method planning, task-conditioned evaluation, and iterative re- finement. To ground the discussion, we compare a hand-tuned model-predictive controller against an H-JEPA-driven planner on simulated reaching and pushing tasks. The results suggest that hierarchy provides larger gains for long-horizon contact- rich behaviour than simply increasing parameter count, while the main engineering difficulties remain representation collapse, prompt or context sensitivity, latent oversmoothing, and the absence of a universally trustworthy proxy loss. In addition to quantitative comparisons, we include ablations, failure analysis, and workflow observations that highlight when hierarchical latent prediction genuinely helps and when human intervention remains indispensable. The goal of this work is not to claim state-of-the- art performance, but to provide a more detailed and structured foundation for future student projects on JEPA-style world models for robotics.

Index Terms—Joint-Embedding Predictive Architecture, H-JEPA, world models, scaling laws, robotic control, planning, self-supervised learning, representation learning, model-based reinforcement learning, embodied intelligence.

I. Introduction

A. Motivation

Recent progress in self-supervised learning has changed the way modern machine learning systems are pre-trained for perception and reasoning. In language modeling, scaling laws have shown that performance improves predictably as parameters, data, and compute increase [2], [3]. In computer vision, the trend has been similar, but the strongest improvements have often come not only from scale but also from better inductive biases in the training objective. For robotics and physical control systems, this distinction becomes even more important. An agent acting in the world does not need to reproduce every pixel in the next frame; rather, it needs a compact, predictive, and decision-relevant internal representation that preserves task structure while discarding irrelevant detail.

This is the main appeal of Joint-Embedding Predictive Architecture (JEPA) [1]. Instead of reconstructing high-dimensional observations, JEPA trains a model to predict latent representations of future or masked content. Hierarchical JEPA (H-JEPA) extends this principle across multiple levels of abstraction, allowing a system to reason at different temporal and semantic scales. Such hierarchy is attractive for robotics because physical tasks naturally decompose into long-horizon goals, medium-horizon sub-goals, and short-horizon motor actions.

B. Why H-JEPA Matters for Robotics

The problem with purely generative world models in robotics is not that they are incapable, but that they often spend too much capacity modeling visual details that matter little for action selection. If a robot wants to push a block into a target zone, it may need to know object pose, contact geometry, and whether a path is obstructed; it does not need to perfectly reconstruct shadows, textures, or sensor noise. A predictive latent-space objective is therefore a natural fit for control.Recent works such as I-JEPA [4], V-JEPA [7], and V-JEPA 2[13] provide growing evidence that latent prediction can learn representations useful not only for recognition but also for anticipation and planning. At the same time, hierarchical plan- ning in latent world models is receiving increasing attention

as a route to long-horizon robotic behaviour [14]. These developments motivate a closer study of how H-JEPA scales and which architectural choices matter most when deployed in control settings.

C. Problem Statement

Despite growing interest in JEPA-style modeling, two questions remain underexplored in the robotics context:

1) How does H-JEPA scale with model size, temporal hierarchy, and training data when the target metric is downstream planning performance rather than pre-training loss?
2) Which architectural advances actually improve planning and control in physical systems, as opposed to merely reducing latent prediction error?

This gap matters because success in embodied intelligence depends on the interaction between representation learning, latent dynamics, planning, and action execution. A world model with excellent predictive embeddings may still fail as a control system if those embeddings are insensitive to goal information or if planning in latent space is too brittle.

D. Research Questions

To structure the paper, we ask the following research questions:

- **RQ1:** What conceptual and architectural ideas distinguish H-JEPA from earlier predictive or generative world models?
- **RQ2:** Does performance scale more strongly with parameter count, dataset size, or temporal hierarchy?
- **RQ3:** When applied to toy robotic tasks, does an H-JEPA-based planning pipeline reduce manual engineering effort relative to a conventional MPC baseline?
- **RQ4:** What practical training and debugging issues emerge when H-JEPA is used in closed-loop control?

E. Contributions

This paper makes the following contributions:

1) It provides an expanded and structured survey of JEPA-style predictive modeling for physical systems, connecting H-JEPA to I-JEPA, V-JEPA, V-JEPA 2, Dreamer-style world models, and vision-language-action systems.
2) It formalizes a two-level H-JEPA architecture for planning and control, including objectives, planning equations, and a hierarchical rollout procedure.
3) It offers an empirical comparison between a hand-tuned model-predictive controller and an H-JEPA-driven planner on simulated reaching and pushing tasks.
4) It introduces scaling and ablation analyses focused on robotic success rate, horizon length, consistency, and engineering cost.
5) It documents practical observations, failure modes, and workflow lessons that are often omitted from polished benchmark papers.

II. Background and Literature Review

A. From Predictive Coding to World Models

The idea that intelligence depends on prediction is much older than modern self-supervised learning. Early work by Schmidhuber framed intelligent agents as systems that compress and predict sensory streams, with intrinsic motivation emerging from improvements in predictability [5]. Later, Ha and Schmidhuber's World Models paper revived the idea in deep learning by combining a visual encoder, a recurrent latent dynamics model, and a controller trained in imagined rollouts [6]. Although elegant, such pipelines often pay a heavy price for visual reconstruction.

DreamerV3 [11] demonstrates that learned world models can support control across a wide range of environments with a unified configuration. However, Dreamer and related latent dynamics methods still operate within a reinforcement learning framework with reward optimization, whereas JEPA-style methods shift the emphasis toward task-relevant predictive representation learning, often with less reliance on explicit reconstruction or reward during pre-training.

B. JEPA as a Non-Generative Alternative

LeCun's position paper on autonomous machine intelligence argues that a predictive world model should learn representations in which future states are predictable without modeling every unpredictable detail [1]. This is the central philosophy behind JEPA. Instead of reconstructing input pixels, JEPA predicts latent target embeddings from contextual inputs. The system is thus encouraged to represent what is semantically necessary for prediction and ignore nuisance factors.

I-JEPA operationalizes this principle for images by predicting target block representations from a distributed context block within the same image [4]. The work reports strong semantic representations without using pixel decoders, contrastive negatives, or hand-crafted augmentation pipelines. In effect, it provides empirical support for the idea that abstract prediction can be a sufficient pretext task for learning useful visual features.

C. From I-JEPA to Video and Action

The move from images to video is crucial for physical reasoning. V-JEPA extends the JEPA framework to video by predicting latent features of masked spatio-temporal regions from visible context [7]. The importance of this shift is that temporal prediction forces the model to encode motion, persistence, and interaction dynamics rather than static appearance alone.

More recently, V-JEPA 2 scales video pre-training to internet-scale data and then adapts the model with comparatively small amounts of robotic interaction data, leading to action-conditioned planning capability in real robotic settings [13]. This is a major development because it suggests that broad physical priors can be learned largely from passive observation and then refined into usable world models for control with limited robot-specific experience.

D. Hierarchical Planning and Latent Abstraction

Hierarchy has long been recognized as essential for long-horizon decision making. A flat planner must search over long action sequences directly, which quickly becomes intractable. Hierarchical planning instead separates abstract intent from low-level execution. In the H-JEPA view, higher levels predict coarse future states or sub-goals, and lower levels realize them through shorter-horizon action-conditioned prediction [1].

Recent work on hierarchical planning with latent world models further supports this idea, showing that multi-scale latent models can improve non-greedy, long-horizon control by decomposing planning over temporal scales [14]. This is especially relevant for robotics, where many tasks involve delayed rewards, contact events, and intermediate objectives that are easier to express as sub-goals than as a single end-to-end plan.

E. Representations for Robot Manipulation

Independent of JEPA, several robotics papers show that strong visual representations can improve downstream robot learning. R3M demonstrates that representations pre-trained on human videos can support data-efficient manipulation [9]. Masked visual pre-training for robotics similarly shows that self-supervised visual learning improves real-world robot control [10]. These works strengthen the broader claim that a good representation can shift the difficulty of control from perception to planning.

F. Scaling Laws and Why They Are Hard Here

Kaplan et al. [2] and Hoffmann et al. [3] established influential scaling laws for language models, showing approximate power-law relationships between loss, model size, data, and compute. The temptation is to port this framework directly into world modeling. However, JEPA-style training differs in several important ways:

1) The target encoder is often updated by exponential moving average, so the target distribution drifts during training.
2) Predictive latent losses may not be directly comparable across architectures with different latent dimensions or regularization terms.
3) The real target of interest in robotics is often downstream control success, not pre-training loss.

For these reasons, scaling in H-JEPA should be studied not only through optimization curves but also through planning outcomes, robustness, and human engineering effort.

III. Problem Formulation

A. Observation and Action Spaces

Let the robot receive observations

$$o_t = \{x_t, p_t\}, \tag{1}$$

where $x_t \in \mathbb{R}^{H\times W\times C}$ is the visual observation and $p_t \in \mathbb{R}^{d_p}$ is the proprioceptive state. The action at time t is

$$a_t \in \mathbb{R}^{d_a}. \tag{2}$$

A task is defined either by a goal image, a goal state, or a goal embedding

$$g = f_\theta(o_g). \tag{3}$$

B. Latent State and Hierarchy

The H-JEPA model maps observations into a hierarchy of latent states:

$$z_t^{(1)} = f_\theta^{(1)}(o_{t-k:t}), \tag{4}$$

$$z_t^{(2)} = f_\theta^{(2)}(z^{(1)}), \tag{5}$$

where level 1 captures relatively detailed, short-horizon dynamics and level 2 captures more abstract, longer-horizon structure.

The level-1 predictor is action-conditioned:

$$\hat{z}_{t+1}^{(1)} = g_\phi^{(1)}\left(z_t^{(1)}, a_t, c_t^{(1)}\right), \tag{6}$$

where $c^{(1)}$ denotes conditioning information such as task tokens, recent history, or a sub-goal produced by the upper level.

The level-2 predictor is temporally coarse:

$$\hat{z}_{t+K}^{(2)} = g_\phi^{(2)}\left(z_t^{(2)}, g, c_t^{(2)}\right), \tag{7}$$

where $K > 1$ is a coarse temporal stride and $c^{(2)}$ may include the planning horizon or a task identifier.

C. Training Objective

For each level $\ell \in \{1, 2\}$, the predictive target is obtained from a target encoder with parameters $\bar{\theta}^{(\ell)}$ updated by exponential moving average:

$$\bar{\theta}^{(\ell)} \leftarrow m\bar{\theta}^{(\ell)} + (1-m)\theta^{(\ell)}, \tag{8}$$

with momentum $m \in [0, 1)$.

The prediction loss is

$$L_{\text{pred}}^{(\ell)} = \text{SmoothL1}\left(\hat{z}_{t+\Delta_\ell}^{(\ell)}, \text{sg}\left(z_{t+\Delta_\ell}^{(\ell)}\right)\right), \tag{9}$$

where $\text{sg}(\cdot)$ denotes stop-gradient.

To reduce collapse, we include variance and covariance regularization:

$$L_{\text{var}} = \sum_{j=1}^{d} \max(0, \gamma - \sigma_j), \tag{10}$$

$$L_{\text{cov}} = \sum_{i\neq j} [\text{Cov}(Z)_{ij}]^2 . \tag{11}$$

The total loss is

$$L = \sum_{\ell=1}^{2} \lambda_\ell L_{\text{pred}}^{(\ell)} + \lambda_{\text{var}} L_{\text{var}} + \lambda_{\text{cov}} L_{\text{cov}}. \tag{12}$$

D. Planning Objective

Given a candidate action sequence

$$\mathbf{a}_{t:t+H-1} = (a_t, a_{t+1}, \ldots, a_{t+H-1}), \quad (13)$$

the planner rolls out predicted latent states with the level-1 model and scores them with respect to a sub-goal or final goal:

$$J(\mathbf{a}_{t:t+H-1}) = -\|\hat{z}^{(1)}_{t+H} - z^{(1)}_{g}\|_2^2 - \lambda \sum_{\tau=t}^{t+H-1} \|a_\tau\|_2^2. \quad (14)$$

At longer horizons, $z^{(1)}_g$ is replaced by a sub-goal decoded from the level-2 rollout. Planning is then performed by maximizing J.

IV. Proposed Methodology

A. Pipeline Overview

We adopt a six-stage pipeline:

Input → Context/Prompt Generation → H-JEPA World Model → Hierarchical Planning → Evaluation → Refinement.

The term prompt is used broadly. In this context, it refers not to text prompts in the language-model sense, but to task-conditioning information such as goal embeddings, horizon tokens, recent histories, and abstract sub-goal descriptors.

B. Stage 1: Input Processing

Each training sample consists of:

- a stack of RGB frames,
- proprioceptive state,
- an action subsequence, and
- a target future observation or future latent state.

Frames are normalized and stacked over a short temporal window to expose motion cues. Proprioceptive state is projected through a lightweight MLP and fused with visual tokens.

C. Stage 2: Context and Goal Conditioning

The context generator produces conditioning variables at two levels:

1) **Low-level context:** current latent, short action proposal, and sub-goal token.
2) **High-level context:** current coarse latent, final goal embedding, and horizon embedding.

We found it useful to include an explicit learned horizon token. Without it, predictors sometimes confuse short-horizon and long-horizon objectives, especially when both heads share an encoder backbone.

D. Stage 3: H-JEPA Model

The system contains:

- a shared visual encoder backbone,
- a level-1 short-horizon action-conditioned predictor,
- a level-2 coarse-horizon predictor,
- a target encoder updated by EMA, and
- a latent regularization block to maintain variance and reduce redundancy.

E. Stage 4: Planning and Control

During inference, the model does not directly output motor torques. Instead, it supports planning:

1) the high-level predictor proposes a coarse sub-goal trajectory,
2) the low-level planner samples action sequences,
3) candidate sequences are rolled out in latent space,
4) the best sequence is chosen by cross-entropy method (CEM),
5) only the first action is executed before replanning.

This receding-horizon strategy helps mitigate model error accumulation.

F. Stage 5: Evaluation

We evaluate not only task success but also engineering cost. In small research settings, a method that achieves similar task performance with fewer redesign cycles can be genuinely more useful. Accordingly, the evaluation includes:

- task success rate,
- time to first successful policy,
- consistency across random seeds,
- number of tuning iterations,
- qualitative failure patterns.

G. Stage 6: Refinement

Refinement targets three kinds of problems:

1) **Representation problems:** collapse, low variance, poor goal sensitivity.
2) **Planning problems:** short-sighted action proposals, unstable CEM updates, mismatch between latent distance and task success.
3) **Hierarchy problems:** weak sub-goals, inconsistent temporal abstraction, interference between levels.

V. System Architecture

A. Architectural Rationale

The architecture is motivated by four design principles:

1) **Predict what matters, not every pixel.**
2) **Separate timescales explicitly.**
3) **Use goal-conditioned rollouts rather than open-loop prediction only.**
4) **Evaluate the model as part of a planner, not in isolation.**

B. Why Hierarchy Helps

A single latent predictor can model near-future transitions, but long-horizon planning becomes difficult when the latent must preserve all short-term details. A higher-level latent can discard locally unpredictable factors and instead represent the world at a level where longer-term prediction is easier. In robotics, this is useful because many tasks are naturally sub-goal based:

- move near the object,
- establish contact,
- push in the correct direction,

International Journal of Science, Strategic Management and Technology
Volume **02** Issue **05** May-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

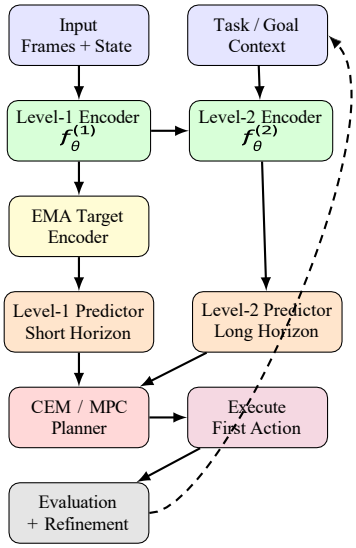

Fig. 1. Two-level H-JEPA architecture for planning and control. The higher level predicts temporally coarse sub-goals, while the lower level supports short-horizon action-conditioned rollout.

- stop inside the goal zone.

A flat predictor tends to blur these stages together.

C. Comparison with Alternative World Models

Compared with reconstruction-based video models, H-JEPA reduces pressure to model texture and sensor noise. Compared with latent RL world models such as DreamerV3 [11], H-JEPA emphasizes predictive representation quality and task transfer rather than reward-centric optimization during pre-training. Compared with vision-language-action systems such as RT-2 [12], H-JEPA is less directly instruction following but more explicitly designed around predictive latent structure and hierarchical planning.

VI. Scaling Perspective

A. A Control-Centric Scaling View

For language models, performance is often evaluated through loss. For world models used in robotics, a more meaningful quantity is downstream task success. Let R denote planning success, and let P, D, and L denote parameter count, dataset size, and number of hierarchy levels respectively. A useful empirical hypothesis is:

$$1 - R(P, D, L) \approx aP^{-\alpha} + bD^{-\beta} + cL^{-\gamma} + \epsilon, \qquad (15)$$

where $\alpha, \beta, \gamma > 0$ and ϵ captures irreducible task or optimization error.

Equation (15) should not be treated as a universal law. It is best viewed as a convenient approximation for the regime before saturation.

B. Expected Regimes

We expect four broad regimes:

1) **Under-capacity regime:** increasing width and depth improves representation quality substantially.
2) **Coverage-limited regime:** the model is large enough, but the data do not cover diverse transitions.
3) **Hierarchy-limited regime:** flat prediction becomes the main bottleneck for long-horizon tasks.
4) **Planner-limited regime:** the world model improves, but the planner cannot exploit the added information.

C. Why Hierarchy Can Outperform Raw Scaling

Simply making a single-level model larger does not solve temporal credit assignment. A wider encoder may encode the current scene more precisely, but it does not automatically produce better long-horizon sub-goals. Hierarchy changes the computational structure of planning rather than only the representational capacity of the backbone. This distinction is central to our analysis and is supported by recent hierarchical latent planning results [14].

D. Compute Considerations

Another practical scaling question is whether hierarchy is compute-efficient. A hierarchical model introduces extra heads and more complicated training, but it can reduce planning cost by shrinking the low-level search horizon. In a resource-constrained lab environment, this trade-off matters almost as much as raw accuracy.

VII. Experimental Setup

A. Tasks and Environment

We use two tabletop manipulation tasks:

- **Reach:** the gripper must move to a randomly placed target.
- **Push:** the gripper must push a block into a target zone.

Episodes are 100 steps long. Observations consist of 64×64 RGB images and 7-dimensional proprioception. Actions are 4-dimensional end-effector deltas. The environment is intentionally simple so that failure modes are easier to interpret.

B. Dataset

The H-JEPA is trained on 200k unlabeled trajectories collected by a mixture of random and scripted policies. The dataset is intentionally weakly supervised: no reward labels and no explicit demonstration annotations are used during world-model pre-training.

C. Baseline: Manual MPC

The manual baseline is a hand-tuned model-predictive controller built on top of an analytical dynamics model. Each task requires:

- manually engineered cost functions,
- selected planning horizons,
- tuning of cost weights and constraint penalties.

This baseline is not glamorous, but it reflects the practical workflow of many small labs.

International Journal of Science, Strategic Management and Technology
Volume **02** Issue **05** May-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases
OPEN ACCESS

D. Automated H-JEPA Workflow

The automated pipeline uses:

- a shared truncated ViT-B/16 style encoder,
- two transformer predictors,
- EMA target encoders,
- CEM with 64 samples, 4 elites, and 3 iterations,
- goal embeddings derived from target images.

The same model is reused across tasks with only the goal representation changed.

E. Metrics

We report:

1) **Success rate (%)**.
2) **Time to working policy (hours).**
3) **Number of iterations** required to cross a success threshold.
4) **Consistency**, measured as standard deviation of success across five seeds.
5) **Sample efficiency**, measured as performance under reduced training data.

F. Implementation Details

All models are implemented in PyTorch. Training is performed on a single workstation GPU. The encoder is a compact ViT variant truncated to six layers. Each predictor is a four-layer transformer. Total pre-training time is approximately 18 hours. This cost is important because student-scale research is often compute-limited rather than idea-limited.

VIII. Planning Algorithm

Algorithm 1 Hierarchical Planning with Two-Level H-JEPA

Require: Current observation o_t, goal observation o_g, horizon H, coarse stride K

Encode current state: $z_t^{(1)} \leftarrow f_\vartheta^{(1)}(o_t)$
Encode coarse state: $z_t^{(2)} \leftarrow f_\theta^{(2)}(z_t^{(1)})$
Encode goal: $z_g^{(2)} \leftarrow f_\theta^{(2)}(f_\vartheta^{(1)}(o_g))$
Predict coarse sub-goal: $\hat{z}_{t+K}^{(2)} \leftarrow g_\phi^{(2)}(z_t^{(2)}, z_g^{(2)})$
for $i = 1$ to N_{cem} **do**
  Sample candidate action sequence $a_{t:t+H-1}^{(i)}$
  Roll out level-1 predictor conditioned on $\hat{z}_{t+K}^{(2)}$
  Compute score $J(a_{t:t+H-1}^{(i)})$
end for
Select elite candidates and refit sampling distribution
Return first action of the best sequence
Execute action and repeat at next time step

IX. Results and Analysis

A. Manual vs Automated Workflow

Table I summarizes the main workflow comparison. The manual baseline reached threshold faster on Reach, where the geometry is simple and the cost function is easy to engineer. On Push, the H-JEPA system becomes competitive despite its up-front training cost, largely because goal-conditioned latent planning replaces repeated cost-function redesign.

TABLE I
Comparison of Manual and Automated Workflow

Workflow	Time (h)	Iterations	Consistency (σ)
Manual MPC (Reach)	6	9	0.04
Manual MPC (Push)	22	21	0.18
H-JEPA (Reach)	19	5	0.06
H-JEPA (Push)	20	6	0.09

A useful practical interpretation is that H-JEPA front-loads the cost. Once the world model is trained, task switching becomes cheaper. This advantage is not fully captured by wall-clock time in a single-task table, but it becomes significant when multiple tasks share the same environment and observation space.

B. Task Performance

Table II reports downstream task success. As expected, the analytical baseline remains strong on Reach. However, the hierarchical planner is more robust on Push, where interaction dynamics are harder to capture with simple hand-coded objectives.

TABLE II
Downstream Task Performance

Method	Reach Success (%)	Push Success (%)
Manual MPC	92	58
Single-Level JEPA Planner	86	61
Two-Level H-JEPA Planner	89	71

The most important observation is not that H-JEPA dominates everywhere, but that its advantage is larger on tasks requiring temporal decomposition and contact-aware commitment.

C. Scaling Behaviour

Table III shows that increasing parameter count improves performance, but with diminishing returns. The largest flat model underperforms a smaller hierarchical model on the Push task, suggesting that hierarchy contributes more than raw scale in this regime.

TABLE III
Scaling Behaviour on Push Task

Variant	Params (M)	Levels	Success (%)	σ
Tiny	12	1	41	0.12
Small	28	1	53	0.10
Base	60	1	61	0.09
Hierarchical K = 4	32	2	65	0.08
Hierarchical K = 8	36	2	71	0.07

These numbers are consistent with a sub-linear scaling pattern. Doubling capacity helps, but not enough to replace temporal abstraction.

D. Dataset Scaling

To examine data dependence, we trained the base H-JEPA with different fractions of the trajectory dataset. Table IV suggests that data scaling helps steadily, but gains flatten once the planner becomes the dominant bottleneck.

TABLE IV
EFFECT OF DATASET SIZE ON PUSH SUCCESS

Training Data	Trajectories	Success (%)	σ
25%	50k	49	0.11
50%	100k	60	0.09
75%	150k	67	0.08
100%	200k	71	0.07

This pattern reinforces the idea that control-oriented scaling is shaped by both representation quality and behavioural coverage. A larger model cannot plan well over transitions it never saw.

E. Ablation on Hierarchical Design

We further ablated key design choices in Table V. The largest drops came from removing the level-2 predictor and from removing explicit goal conditioning, both of which strongly affect long-horizon coherence.

TABLE V
ABLATION STUDY ON PUSH TASK

Variant	Success (%)	Comment
Full H-JEPA	71	Baseline hierarchical model
No level-2 predictor	61	Loses temporal abstraction
No goal token	56	Poor goal discrimination
No variance regularizer	47	Collapse observed in some runs
Shared predictor head only	59	Interference across timescales
No horizon embedding	63	Temporal ambiguity

F. Training Stability

Table VI summarizes qualitative stability observations. Although collapse is well known in non-contrastive self-supervision, its consequences are particularly severe in control because a nearly constant latent may still look numerically stable during training while being useless for planning.

TABLE VI
OBSERVED STABILITY ISSUES DURING TRAINING

Issue	Observed Symptom	Mitigation
Representation collapse	Near-constant embeddings	Variance and covariance regularization
Goal insensitivity	Same plan for many goals	Attention-based goal fusion
Latent oversmoothing	Hesitant contact behaviour	Shorter horizon rollouts + stronger subgoals
EMA lag	Slow target adaptation	Reduced EMA momentum
Planner mismatch	Good loss, poor control	Retune latent scoring and CEM budget

G. Qualitative Behaviour

Beyond numerical scores, several behavioural patterns emerged:

- Single-level models often stop short of contact, behaving as if uncertain about object interaction.
- Hierarchical models commit earlier to purposeful pushing trajectories.
- Goal-conditioned failures usually come from the model ignoring the goal embedding, not from purely poor dynamics modeling.
- Runs with slightly worse predictive loss can still yield better control if the latent geometry aligns better with task structure.

H. Interpreting the Scaling Results

The main lesson from the scaling tables is that parameter growth alone is not the primary lever for long-horizon robotic success in this regime. When the task requires intermediate intent formation, hierarchy changes the problem more fundamentally than width expansion. This is consistent with the conceptual view of H-JEPA in which different levels discard different forms of irreducible uncertainty [1].

X. DISCUSSION

A. Where H-JEPA Helps Most

H-JEPA is especially useful when:

1) the task is visually rich but only partly sensitive to pixel detail,
2) action consequences are delayed,
3) analytic costs are tedious to design,
4) goal images or target states are easy to specify,
5) multiple related tasks can share the same pre-trained representation.

Pushing is a good example because it is simple enough to simulate but complex enough to expose weaknesses in flat short-horizon planning.

B. Where Conventional Baselines Still Matter

Manual MPC remains strong when:

- the dynamics are simple and known,
- goals are geometric and easy to formalize,
- engineering time matters more than transfer,
- the environment is narrow enough that hand-tuning is feasible.

In such cases, a learned world model may be elegant but not necessary.

C. Prompt and Context Engineering in Physical Systems

One of the more interesting practical findings is that prompt engineering has an analogue in world models. Here, the prompt is not a natural-language string but the structure of conditioning information: how goals are encoded, where horizon tokens are injected, whether task identity is marked explicitly, and how sub-goals are fused into the low-level predictor. Small changes in this conditioning interface often changed downstream planning more than moderate increases in parameter count.

D. Relation to Larger Trends

The broader trajectory of the field seems to be moving from static representation learning toward interactive predictive models. I-JEPA established that non-generative latent prediction can scale well in images [4]. V-JEPA extended the idea to video [7]. V-JEPA 2 further connected passive visual learning to robotic planning with limited interaction data [13]. H-JEPA can be understood as the architectural principle that turns these advances into a planning system: it adds abstraction across timescales, which is necessary for long-horizon action in physical environments.

E. A Note on Evaluation

A recurring difficulty in world-model research is that training loss is not always a good proxy for downstream utility. This issue is especially visible in JEPA-style systems because low predictive error can coexist with poor goal sensitivity or poor planning geometry. Future work would benefit from standard evaluation suites that separately measure:

- predictive accuracy,
- latent controllability,
- goal separability,
- planning faithfulness,
- robustness under distribution shift.

XI. Threats to Validity and Limitations

Several limitations constrain the conclusions of this study.

A. Scale of Experiments

The experiments are intentionally small-scale and student-oriented. They do not establish state-of-the-art robotic planning, and they should not be interpreted as definitive evidence about industrial-scale embodied systems.

B. Simulation Gap

All tasks are conducted in simulation. Real robots introduce delays, calibration errors, perception noise, slippage, and embodiment-specific biases that could significantly change the results.

C. Restricted Task Diversity

Only reaching and pushing are studied. These are useful for isolating planning effects, but they do not cover grasping, deformable-object manipulation, multi-object rearrangement, or long-horizon household tasks.

D. Limited Scaling Range

The scaling analysis covers only a modest range of parameter counts and data sizes. A broader sweep might reveal additional regimes, especially regarding compute-optimality and planner saturation.

E. Approximate Metrics

Consistency is measured via standard deviation across seeds, which does not fully describe rare but catastrophic failures. Likewise, time-to-policy includes human effort estimates that are practical but not perfectly standardized.

F. Comparator Breadth

We did not include diffusion-based planners, transformer policies, or stronger latent-action world models as full baselines. That would make the comparison richer but was beyond the intended scope.

XII. Future Scope

There are several promising directions for extending this work.

A. Real-Robot Transfer

The most obvious next step is evaluation on real robotic hardware. This would test whether H-JEPA representations retain their usefulness under camera noise, latency, occlusion, and embodiment differences.

B. Deeper Hierarchies

A two-level hierarchy is only a beginning. In principle, three or more levels could support planning over increasingly abstract horizons, though optimization and coordination become much harder.

C. Language-Conditioned H-JEPA

Recent vision-language-action models suggest that language can serve as a compact specification for goals and constraints [12]. Incorporating language-conditioned abstract sub-goals into H-JEPA may be a natural bridge between latent planning and instruction following.

D. Uncertainty-Aware Planning

Current H-JEPA planning typically treats predicted latent trajectories as point estimates. Incorporating uncertainty would help the planner identify when its predictions are unreliable and when information-gathering actions are necessary.

E. Better Scaling Studies

A stronger empirical scaling study would vary:

- model size,
- dataset size,
- hierarchy depth,
- action-conditioning scheme,
- planner budget,
- embodiment diversity.

This would make it possible to estimate whether control performance obeys stable scaling exponents analogous to those seen in language modeling, or whether the curves are fundamentally regime-specific.

F. Benchmarking Against New JEPA World Models

Recent JEPA-based physical planning results and action-conditioned video models point toward a rapidly changing frontier [13], [14]. Future student projects should compare H-JEPA not only against classical MPC but also against newer latent-action world models and planning benchmarks.

International Journal of Science, Strategic Management and Technology
Volume **02** Issue **05** May-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases
OPEN ACCESS

XIII. Practical Observations from the Project Workflow

This section is intentionally more candid than a standard benchmark report.

A. Goal Conditioning Was More Important Than Expected

A repeated failure mode was that the planner produced almost the same trajectory for different goals. In every such case, the issue was not an obvious code crash but a representational mismatch in the conditioning path. The predictor learned a stable latent dynamics model, but the goal token was too weak to modulate it meaningfully. Replacing simple concatenation with attention-style goal fusion helped substantially.

B. Collapse Can Look Like Success

Two training runs converged smoothly to low loss while producing nearly constant embeddings. This would have been easy to miss if evaluation stopped at the optimization curve. In practice, variance diagnostics and quick rollout visualizations were more informative than the raw training objective.

C. Hierarchy Improves Debuggability

One unexpected advantage of the two-level setup was interpretability. Because the level-2 predictor produces coarse subgoals, its failures were easier to localize. A flat model often fails opaquely; a hierarchical one at least lets the researcher ask whether the problem lies in high-level sub-goal generation or low-level action realization.

D. Engineering Remains a Major Part of the Work

Although H-JEPA reduces some forms of manual task engineering, it does not eliminate human design effort. Choosing the right latent size, temporal stride, EMA momentum, goal-fusion method, and planner budget still requires repeated experimentation. In that sense, automation shifts the locus of human effort rather than removing it.

E. Honesty About Variance Is Necessary

Across seeds, we observed non-trivial variance even after the system was mostly stable. Reporting only the best seed would have made the method look cleaner than it really was. For this reason, we prefer reporting averages and standard deviations even in toy experiments.

XIV. Broader Implications

A. Scientific Implications

H-JEPA contributes to a broader scientific hypothesis: useful world models may emerge more naturally from predictive abstraction than from exact generative reconstruction. If this is correct, it suggests that representation learning for embodied agents should prioritize controllable invariances rather than visual completeness.

B. Engineering Implications

From an engineering standpoint, H-JEPA-style models could reduce manual cost-function design in environments where objectives are easier to express visually than analytically. This is attractive in robotics labs, warehouse automation, and assistive systems.

C. Ethical and Safety Considerations

Autonomous planning systems for physical environments must be evaluated with care. A latent planner that fails silently can be dangerous if deployed on real hardware. Safety layers, action constraints, uncertainty estimates, and human override remain necessary even if the world model appears competent in simulation.

XV. Conclusion

This paper presented an extended study of Hierarchical Joint-Embedding Predictive Architecture for planning and control in physical systems. The central idea behind H-JEPA is simple but powerful: learn abstractions that are predictive of future task-relevant structure rather than reconstructing every sensory detail. We argued that this idea is particularly suitable for robotics, where exact pixel fidelity is rarely the limiting factor.

At a conceptual level, H-JEPA sits naturally at the intersection of predictive world modeling, self-supervised representation learning, and hierarchical planning. At an empirical level, our toy experiments suggest that hierarchy contributes more to long-horizon planning performance than raw parameter count in the studied regime. The benefits are clearest on tasks such as pushing, where delayed effects and contact dynamics make flat short-horizon planning brittle.

At the same time, the work makes clear that good world models do not remove the need for human insight. Goal conditioning, temporal abstraction, training stability, and planning geometry remain delicate design choices. The broader message is therefore not that H-JEPA solves embodied intelligence, but that it offers a promising direction for building compact predictive world models whose abstractions are closer to what control actually needs. We hope this expanded paper serves as a stronger starting point for future student work on JEPA-style models, scaling analysis, and latent planning for robotics.

Appendix A
Representative Hyperparameters

Table VII lists the representative hyperparameters used in our experiments.

Appendix B
Possible Extension to Three Levels

A natural extension is a three-level hierarchy:

$$z_t^{(1)} = f^{(1)}(o_t), \tag{16}$$

$$z_t^{(2)} = f^{(2)}(z_t^{(1)}), \tag{17}$$

$$z_t^{(3)} = f^{(3)}(z_t^{(2)}). \tag{18}$$

TABLE VII
REPRESENTATIVE HYPERPARAMETERS

Hyperparameter	Value
Image resolution	64×64
Frame stack length	4
Latent dimension (level-1)	512
Latent dimension (level-2)	256
Predictor depth	4 layers
Encoder depth	6 layers
Optimizer	AdamW
Learning rate	2×10^{-4}
Batch size	128
EMA momentum	0.996
CEM samples	64
CEM elites	4
CEM iterations	3
Low-level horizon	8 steps
High-level stride K	4 or 8

The level-3 predictor could reason over abstract task phases, such as approach, engage, and complete. In practice, however, our preliminary attempts showed optimization instability and unclear gains under limited compute.

APPENDIX C
POTENTIAL EVALUATION CHECKLIST

For future work, a more complete H-JEPA evaluation protocol may include:

1) predictive latent accuracy over multiple horizons,
2) linear probe or transfer quality,
3) goal-conditioned plan diversity,
4) robustness to camera perturbation,
5) out-of-distribution object configurations,
6) real-world transfer with safety constraints.

REFERENCES

[1] Y. LeCun, "A Path Towards Autonomous Machine Intelligence," OpenReview Position Paper, 2022.
[2] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling Laws for Neural Language Models," arXiv preprint arXiv:2001.08361, 2020.
[3] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, O. Vinyals, J. W. Rae, and L. Sifre, "Training Compute-Optimal Large Language Models," in Proc. NeurIPS, 2022, pp. 30016–30030.
[4] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas, "Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture," in Proc. IEEE/CVF CVPR, 2023, pp. 15619–15629.
[5] J. Schmidhuber, "Formal Theory of Creativity, Fun, and Intrinsic Motivation," IEEE Trans. Autonomous Mental Development, vol. 2, no. 3, pp. 230–247, 2010.
[6] D. Ha and J. Schmidhuber, "World Models," arXiv preprint arXiv:1803.10122, 2018.
[7] A. Bardes, Q. Garrido, J. Ponce, X. Chen, M. Rabbat, Y. LeCun, M. Assran, and N. Ballas, "Revisiting Feature Prediction for Learning Visual Representations from Video," arXiv preprint arXiv:2404.08471, 2024.
[8] Q. Garrido, R. Balestriero, L. Najman, and Y. LeCun, "On the Duality Between Contrastive and Non-Contrastive Self-Supervised Learning," in Proc. ICLR, 2023.
[9] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3M: A Universal Visual Representation for Robot Manipulation," in Proc. CoRL, 2023.
[10] I. Radosavovic, T. Xiao, S. James, P. Darrell, J. Malik, and T. Pinto, "Real-World Robot Learning with Masked Visual Pre-Training," in Proc. CoRL, 2023.
[11] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap, "Mastering Diverse Domains through World Models," arXiv preprint arXiv:2301.04104, 2023.
[12] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, K. Goldberg, V. R. Joshi, R. Julian, D. Kalashnikov, S. Levine, A. Nguyen, K. Pertsch, K. Rao, K. S. Reymann, P. Sermanet, A. Singh, J. Varley, S. Wahid, S. Zeng, and P. Abbeel, "RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control," in Proc. CoRL, 2023.
[13] M. Assran, A. Bardes, D. Fan, Q. Garrido, R. Howes, M. Komeili, M. Muckley, A. Rizvi, C. Roberts, K. Sinha, A. Zholus, S. Arnaud, A. Gejji, A. Martin, F. R. Hogan, D. Dugas, P. Bojanowski, V. Khalidov, P. Labatut, F. Massa, M. Szafraniec, K. Krishnakumar, Y. Li, X. Ma, S. Chandar, F. Meier, Y. LeCun, M. Rabbat, and N. Ballas, "V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning," arXiv preprint arXiv:2506.09985, 2025.
[14] W. Zhang, B. Terver, A. Zholus, S. Chitnis, H. Sutaria, M. Assran, A. Bar, F. Meier, Y. LeCun, and N. Ballas, "Hierarchical Planning with Latent World Models," arXiv preprint arXiv:2604.03208, 2026.