KineJing: resource-efficient adaptation and evaluation of tri-view robot predictors

YiCheng Pan | KineJing Research Team

Author review draft | 25 October 2026 | Not submitted

Abstract

Efficient robot prediction requires distinguishing adaptation gains from useful dynamics. We study KineJing Pro Local, which adapts the released BWM predictor using 1.47 million low-rank parameters. In a matched local configuration, 96 updates reduce future RGB mean squared error by 71.06% over ten previously explored simulation trajectories, while remaining worse than first-frame persistence. Compressed weight storage permits the complete RGB pathway on a 12 GB laptop GPU. To investigate the remaining gap, we train a separate 11,285-parameter identity-anchored predictor and three controls across three seeds. Mean error improves over persistence by 6.54%, but removing EEF inputs produces similar performance. The earlier KineJing-Pro submission separately ranks first in perspective among 60 TriWorldBench entries. These results demonstrate executable adaptation and a small predictive improvement while exposing the need for stronger action-sensitive validation.

Robot video prediction has two distinct experimental demands: a model must run under an explicit resource budget, and its predicted transitions must improve under a meaningful evaluation. A successful execution alone establishes neither accurate dynamics nor useful robot control. This distinction becomes particularly important for three-camera manipulation, where an apparently stable head view can coexist with incorrect local interactions or disagreement at the wrists [1].

Our research question is whether a small, explicitly trained adaptation can improve a released robot video predictor while retaining an executable RGB pathway on a consumer laptop. We study this question through two evidence streams: the earlier KineJing-Pro benchmark submission, and a subsequent, separately versioned local adaptation experiment. They share upstream model lineage but are different systems. The benchmark result is not assigned to the local adapter, and local pixel improvements are not converted into a predicted benchmark score.

The KineJing contribution is an implemented adaptation and execution workflow. It includes checked conversion of bimanual trajectory inputs, a validation-selected query/value adapter, a loading and compressed-storage path, and a strict initial-image-to-RGB interface. The released BWM model supplies the pretrained action-conditioned core; BWM itself initializes from Wan2.2-TI2V-5B [2]. We do not claim to have trained that core, invented low-rank adaptation, or established a new world-model architecture.

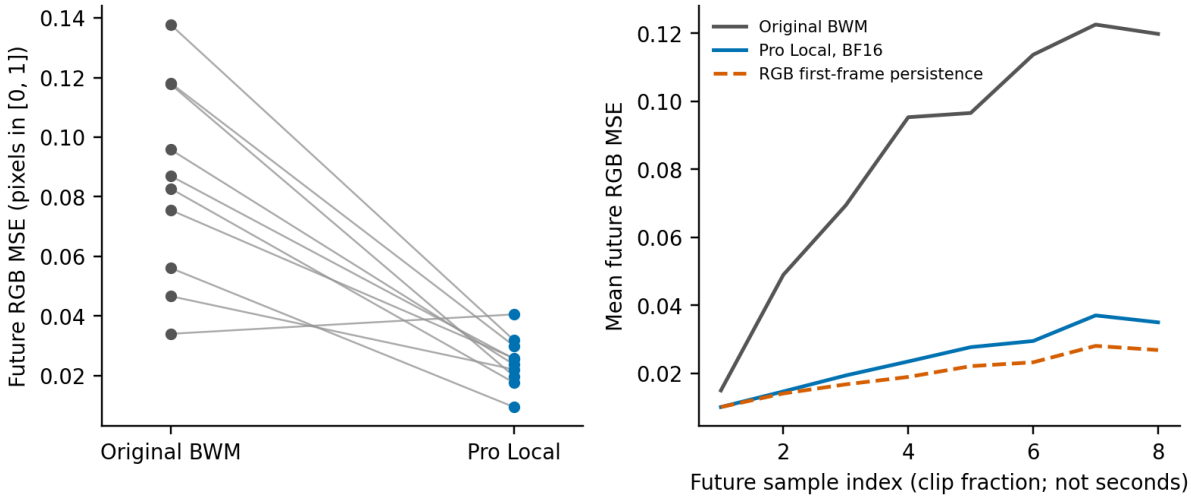
Our evidence supports three specific contributions. First, the local adapter improves decoded future RGB prediction relative to the unchanged BWM checkpoint under matched inputs, noise and integration steps. Second, the complete compressed core and VAE execute together on the tested 12 GB GPU, with clearly scoped memory and timing measurements. Third, a full benchmark profile documents where the earlier submission performs well and where it fails. These contributions are deliberately scoped to measured adaptation and implementation, rather than general robot competence.

Results

A small trained adapter improves the matched local predictor

The adapter contains 1,474,560 trainable parameters, approximately 0.0293% of the 5,025,700,032-parameter upstream core. Rank-four updates are installed on the query and value projections of 30 self-attention blocks. We train for four epochs over 24 complete trajectories, select the checkpoint using four separate validation trajectories, and evaluate ten further trajectories. All three views remain in the same trajectory partition. The public development data had been explored previously; this is a local holdout comparison rather than a new blind benchmark.

Under identical evaluation inputs and noise, future RGB MSE decreases by 71.06%, from 0.08517601 to 0.02464687. Mean episode PSNR increases from 11.050 to 16.355 dB, and SSIM from 0.54946 to 0.63926. The recorded 10,000-resample paired whole-trajectory bootstrap gives a 95% interval of 59.08-77.61% for relative MSE reduction. This interval describes variation over the evaluated trajectories; it does not include training-seed variability, unseen tasks or population-wide robot performance.



Matched local prediction and horizon errors. Left, lines join the original and adapted errors for each of ten complete evaluation trajectories; points are trajectories, not independent frames. Right, mean errors across the eight future sample indices. Inputs contain three 96 x 96 views and supplied future EEF states. Samples are clip fractions, not elapsed seconds. Persistence is included to expose the remaining dynamic-prediction gap. Source: retained PRO-REAL-0001 RGB evaluation.

Compressed storage retains the observed local quality

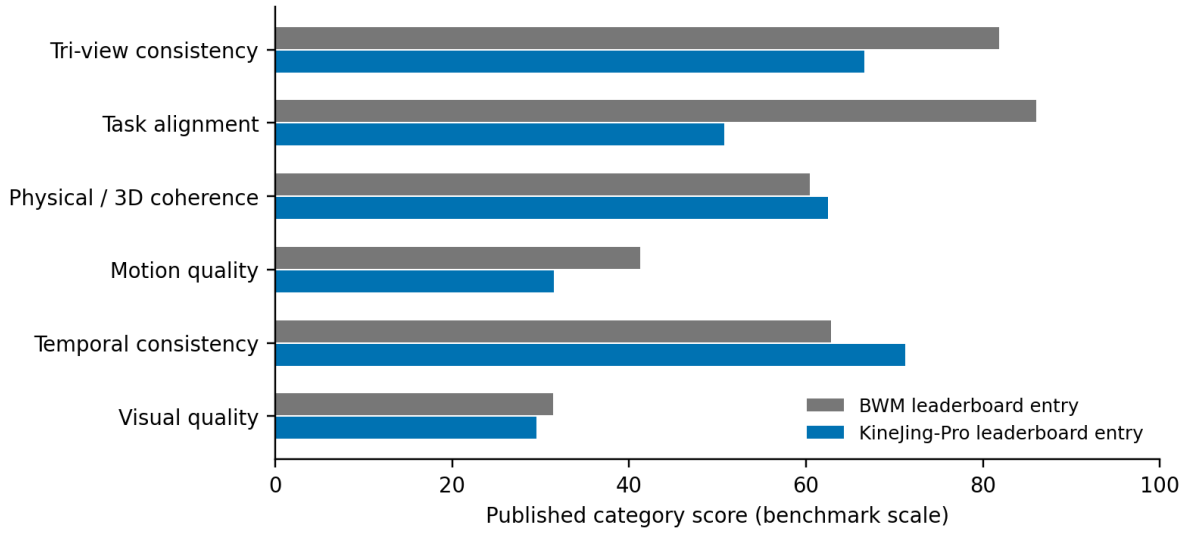
The compressed path stores linear weights in row-scaled FP8 E4M3FN representation and dequantizes them for BF16 matrix multiplication. It is a storage implementation, not native FP8 GEMM. Its mean future RGB MSE is 0.02439503, compared with 0.02464687 for the adapted BF16 model. The recorded paired interval for this small relative difference spans -0.93% to 2.86%; the data do not establish an additional quality gain from quantization. The supported result is that the compressed representation meets the experiment's quality-retention criterion on these inputs.

The complete pipeline accepts CPU initial RGB images and a supplied trajectory, encodes the initial images with the Wan VAE, predicts with the compressed core, decodes future RGB and returns the result to CPU. On the tested RTX 5070 Ti Laptop GPU, peak CUDA allocated memory during prediction is 6.291 GiB after resetting the counter following model loading. The first prediction takes 3.901 s; three subsequent repetitions on one fixed input have a median of 1.501 s. The full process, including loading, quantization and four prediction repetitions, takes 86.640 s. These measurements establish feasibility for the specified small configuration, not real-time, production-resolution or hardware-independent performance.

The earlier benchmark submission has specific geometric strengths

In the 25 October 2026 TriWorldBench snapshot, KineJing-Pro scores 54.96 and is listed 44th among 60 published model entries. Perspective is 91.43, ranked first; physical/3D coherence is 62.50 and temporal consistency 71.22, both ranked sixth. These are strengths of the submitted system in that dated benchmark, rather than evidence of a newly trained foundation model. The official table and its embedded JSON are two representations of one response; agreement between them checks extraction consistency, not independent evaluation replication.

The same snapshot lists BWM at 65.54 overall. KineJing-Pro has higher published perspective, physical/3D and temporal scores, but lower overall, tri-view, task-alignment and motion scores. We report the six-category profile in full rather than selecting only winning metrics. Because checkpoint identity, sampling settings and per-episode scores are not jointly available for these entries, their difference is a descriptive leaderboard comparison. It is not an ablation proving that KineJing post-processing caused the geometric or temporal advantages.



Complete six-category benchmark comparison. Scores are copied from the same dated public leaderboard snapshot. Blue denotes the earlier KineJing-Pro submission and grey the published BWM entry. The local LoRA adapter is not evaluated in this figure. Category values have benchmark-specific meanings; bar lengths do not measure physical error or robot success. No uncertainty intervals are available for these aggregate entries. Source: TriWorldBench, accessed 25 October 2026.

Persistence remains a stronger local pixel baseline

A useful positive comparison must survive an inexpensive reference. Exact repetition of the permitted initial RGB frame yields future MSE 0.02005718, PSNR 17.281 dB and SSIM 0.69815. The BF16 adapter therefore remains 22.88% worse in MSE than RGB persistence. Repeating the initial latent and decoding it also outperforms the predictor, with MSE 0.01720288. Access to future reference latents yields a VAE-reconstruction MSE of 0.00059897; this last result is an oracle reconstruction diagnostic and is never treated as a predictor.

This comparison narrows the finding: the adapter repairs a substantial part of the original predictor's error in the local configuration, but has not established a useful dynamic advantage over a static baseline. Nor does a low pixel error alone prove correct contact, action response or task completion. Related evaluation work separates perceptual dimensions from task and functional utility [3-5]. The decisive validation target is consequently an action-sensitive, persistence-beating result on untouched trajectories. The separate small-predictor experiment below tests these requirements individually.

Changed-region diagnostics do not remove the adaptation gap

A fresh CPU reanalysis tests whether the adapted predictor loses only because of errors in static regions. We construct evaluation-only masks from the RGB difference between each target and the initial image; these masks never enter inference. At RGB root-mean-square change greater than 0.05, the FP8-stored adapter has changed-region MSE 0.05051663 versus persistence 0.04613697. The ordering is unchanged at thresholds 0.02 and 0.10. Because image change also includes camera motion, illumination and occlusion, this diagnostic measures change-conditioned reconstruction rather than verified object motion. It nevertheless rules out the narrow explanation that all excess error lies in low-change pixels.

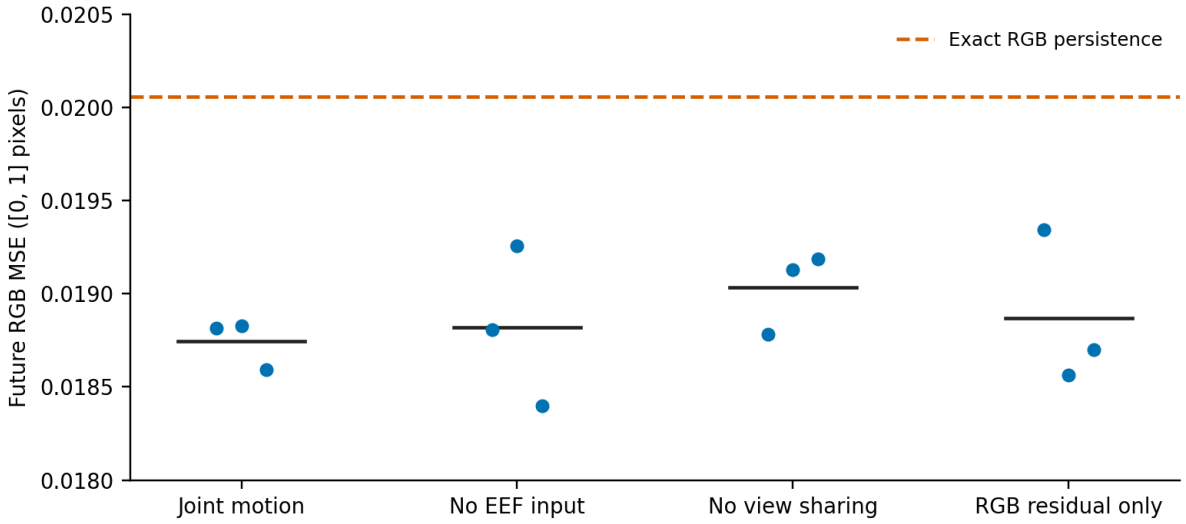
We also freeze a residual-shrinkage grid and select its threshold on the four validation trajectories before applying it to the ten evaluation trajectories. Selection returns zero shrinkage, preserving the adapted prediction. This baseline therefore does not repair the persistence gap. Both outcomes are retained, including the unsuccessful alternative, rather than selecting a favorable region or threshold after seeing evaluation errors.

A small identity-anchored predictor improves pixels without establishing action dependence

We next train a separate, randomly initialized 11,285-parameter predictor that warps the initial views and adds a bounded RGB residual. It is a research prototype, distinct from both the BWM adapter and the earlier scored submission. Three seeds are trained for each of four variants: joint prediction, EEF-input removal, removal of cross-view sharing, and residual-only prediction. All variants use the same trajectory partitions and validation selection; all twelve checkpoint choices are sealed before their evaluation predictions are generated.

The joint variant has mean future RGB MSE 0.01874503, averaged over the errors of three separately fitted models; it is not a three-model prediction ensemble. The across-seed standard deviation of mean MSE is 0.00013303. Relative to exact RGB persistence, error falls by 6.54% (whole-trajectory bootstrap interval 1.64-10.45%, conditional on these three fitted seeds). The result establishes a small improvement on the reused evaluation set, rather than performance on an untouched test distribution.

The EEF-input-free control achieves MSE 0.01882163, giving only a 0.41% advantage for the joint model, with interval -0.93% to 1.63%. The residual-only control reaches 0.01886998; the joint advantage is 0.66%, with interval -0.78% to 2.05%. Removing shared-view features gives MSE 0.01903336, a 1.51% difference, with an exploratory unadjusted interval of 0.04-2.96%. None of these intervals addresses task generalization or multiplicity. The frozen promotion criterion requires an advantage over both persistence and the actionless control; it is not met. Thus pixel improvement does not establish that the supplied EEF values or the deformation component are responsible.



Exploratory controlled training of a separate small predictor. Each dot is the mean error of one fitted seed on the same ten complete simulation trajectories; horizontal ticks are arithmetic means over the three fitted seeds. The dashed line is exact initial-frame persistence. Checkpoints are selected using four other trajectories. Lower error is better. Dots are training seeds, not independent test episodes, and the four variants are not four versions of the officially scored model. All twelve runs are retained.

Discussion

The study separates three experimentally different achievements: repairing a fixed pretrained predictor, executing its complete RGB pathway under a laptop memory budget, and improving over an inexpensive static reference. The adapter establishes the first two; the separate small predictor provides a modest instance of the third. Its actionless control performs similarly, showing why a persistence-beating RGB number alone is insufficient to demonstrate an action-sensitive world model.

Two distinctions govern interpretation. The first is lineage versus contribution: using BWM provides a strong pretrained starting point, while KineJing's adapter, input checks and runtime are separate engineering and experimental additions. The second is benchmark profile versus causal mechanism: favorable perspective and temporal scores describe the earlier submission, whereas the local experiment directly compares trained and unchanged weights. Neither evidence stream can replace the other.

The observed profile motivates, but does not prove, a stability-dynamics trade-off. A video that changes too little may retain appearance and geometric regularity without following the desired manipulation. Alternatively, score differences could reflect input normalization, rollout settings, image resizing, post-processing or evaluator sensitivity. The present artifacts do not distinguish these explanations. A controlled study should fix the upstream checkpoint, trajectory conversion, random draws, resolution and codec, then change one KineJing component at a time.

The limitations are material. BWM-adapter training has one seed; evaluation uses ten previously explored simulated trajectories, square low-resolution images and nine sparse sample positions. The supplied future EEF values are recorded states, not verified intervention commands. Complete-pipeline consistency was checked on one trajectory, and latency on three hot repetitions of one input. The scored competition package predates the local adapter. No new official score, unseen-task generalization, closed-loop benefit, real-robot validation or safety outcome is established here.

Methods

Trajectory and view contracts

The retained TWB input adapter reads permitted initial head, left-wrist and right-wrist images and the trajectory HDF5 data. Each arm contributes Cartesian position, XYZ Euler rotation and gripper opening, giving fourteen columns. Quaternion inputs use the recorded wxyz convention; the adapter checks shape, finiteness, unit quaternion norm and temporal length before conversion. These checks avoid silently treating joint coordinates, quaternions and absolute EEF states as interchangeable. They validate representation, not full coordinate-frame or hardware-command equivalence.

For the local experiment, each trajectory provides nine sampled positions and three 96 x 96 RGB views, stacked along image height before VAE processing. Input normalization uses fixed upstream first and ninety-ninth percentile bounds. Future ground-truth images are targets only; the prediction path uses the initial images and supplied EEF sequence. Later sparse indices have no calibrated mapping to seconds. A separate packaging adapter resizes generated submission videos and permits at most three repeated trailing frames to meet expected counts; this formatting operation is not claimed as a learned improvement or proof that the exact adapter version produced the scored package.

Adapter training and checkpoint selection

For each selected frozen projection W, low-rank adaptation adds a residual delta W = (alpha/r) BA, with rank r = 4 and alpha = 4. We follow the low-rank adaptation construction rather than claiming the decomposition itself as novel [7]. Adapter factors are FP32; their residual is cast for the BF16 core. The base core and action encoder remain frozen. The retained implementation initializes B to zero and A with Kaiming initialization, so the initial residual is zero.

The recorded recipe uses AdamW, learning rate 0.0001, zero weight decay, batch size one and L2 gradient clipping at one. Four passes over 24 training trajectories yield 96 updates. Ninety-five updates record positive finite gradients; one samples the upstream scheduler's zero-loss-weight endpoint. Validation uses two fixed noise draws per trajectory. Checkpoint choice minimizes mean validation weighted future flow loss, breaking ties by the earliest epoch. Epoch four is selected and its hash sealed before the evaluation stage. Evaluation uses the same inputs, noise and eight Euler integration steps for the original and adapted core.

Loading, storage and execution checks

The implementation validates expected checkpoint keys and shapes, loads checkpoint-backed tensors with bounded CPU buffers and rejects missing-weight or random fallback. Unsupported legacy modules are explicitly disabled. The compressed path stores 304 linear matrices in row-scaled FP8 representation while retaining biases, normalization and action parameters in BF16; weights are dequantized for the matrix multiplication. Core attention uses the recorded math backend and the VAE retains its native default backend. A prior incorrect VAE backend scope and a failed output-consistency audit are retained in the development history; the corrected complete entry matches saved staged predictions on the checked trajectory.

Identity-anchored prototype and matched training controls

The prototype uses two stride-two convolutional layers (3 to 12 to 16 channels), with SiLU activation, applied with shared weights to all views. A conditioner maps initial EEF values, their supplied current displacement and clip time (29 inputs) through widths 24 and 16. The head receives local features, their mean across the three views, conditioned features and a three-component view-role indicator. This mean is a simple feature-sharing operation, not calibrated 3D correspondence. A 51-to-16-to-5 convolutional head predicts two displacement channels and three RGB residual channels. Its final layer is initialized to zero, so training starts at initial-image persistence.

Displacements are bounded to twelve pixels times clip fraction; RGB residuals to 0.1 times clip fraction. Bilinear sampling uses border padding and aligned corners, followed by a [0,1] clamp. Zero clip time returns the initial image irrespective of EEF input. Predicting image motion conditioned on actions has established prior work [10]; the present architecture is an exploratory baseline implementation, not a claim to invent motion-based video prediction.

Each of twelve fits uses Adam with learning rate 0.001, gradient clipping at one, twenty epochs and twenty-four trajectory updates per epoch. One future sample position is drawn per trajectory update; all three views remain together. The objective is RGB MSE plus 0.0001 times mean squared displacement and 0.01 times mean squared RGB residual. Every second epoch is evaluated over all eight future positions in the four validation trajectories, with the smallest error and earliest tie selecting the checkpoint; the identity initialization is eligible. Seeds are 17, 29 and 43. Inference uses initial RGB, initial/current supplied EEF values and time, with no future RGB input. All operations run on CPU; a 1,200-second time guard and 0.8 GiB available-memory floor bound the job. Twelve fits and evaluation complete in 496.94 seconds.

For the new controls, uncertainty resamples ten complete paired trajectories 10,000 times with seed 20261025, using the arithmetic mean of the three per-seed errors for each trajectory. Seed means and their sample standard deviation are separately reported. This bootstrap conditions on the three trained models and does not combine training-seed and population uncertainty. The comparison family is exploratory and the intervals are not corrected for multiple comparisons. Historical local data had already been inspected; a frozen current recipe prevents within-run evaluation-based selection but cannot make this a blind study.

As an additional exploratory check, exact two-sided sign tests compare the joint model with persistence and each of the three controls; Holm adjustment covers these four tests. Joint prediction has eight trajectory wins and two losses against persistence (adjusted $p = 0.4375$), and seven wins and three losses against EEF removal (adjusted $p = 0.6875$). These tests concern win probability rather than mean error reduction, assume independent trajectory signs, and provide no adjusted evidence of systematic superiority. The task-level independence assumption remains unverified.

Quality metrics and uncertainty

For each trajectory we average squared RGB error over all three views, eight future sample positions and pixels in [0,1], excluding the conditioning frame. The primary estimate is the equally weighted mean over trajectories. PSNR is computed per trajectory as -10 log10(MSE) and then averaged; it is not obtained by transforming the overall mean. SSIM uses an eleven-pixel Gaussian window, sigma 1.5, population covariance and unit data range. The appendix retains all model and persistence means, including the unsuccessful baselines.

For a candidate with per-trajectory errors c and reference errors b, relative improvement is 100 times (1 - mean(c)/mean(b)). The recorded bootstrap resamples ten complete paired trajectories with replacement 10,000 times using seed 20261007. All views and future positions remain together. The reported 95% interval is a trajectory-resampling interval for one fitted model; it is not a confidence statement over random initializations. Multiple secondary comparisons are exploratory, and no confirmatory family-wise significance claim is made.

Benchmark extraction and version separation

We preserve the public leaderboard HTML and extract its rendered table and embedded records without a hidden API. All 60 entry names and order are compared, and 1,560 score/rank cells are checked with tolerance for two-decimal display rounding. We retain published overall ordinal positions, including tie ordering; metric ranks use one plus the count of strictly greater scores. We do not reconstruct TWB-Score by averaging displayed category scores because that does not reproduce the published aggregate. No official package hash is present in the public target record.

Data and code availability

The draft includes aggregate local metrics, the published leaderboard extraction, bibliographic records and checksums. The benchmark dataset and upstream model weights are obtained from their original providers under their respective terms. Raw videos, reference images, upstream weights, personal correspondence and private company records are not redistributed here. The KineJing inference implementation and adapter weights are not publicly released with this draft. Access arrangements sufficient for review and independent reproduction remain to be resolved before a journal submission; no such access is claimed to have been granted.

Acknowledgements and research assistance

We acknowledge the BWM and Wan contributors for the pretrained model components and the TriWorldBench authors for the evaluation framework. AI assistance was used for literature retrieval, artifact inspection, manuscript drafting and consistency checks, including the Scientific Agent Skills procedures [6]. The manuscript does not treat assistant-generated prose or adapters as evidence of model performance. Author verification and responsibility for the final content remain necessary.

Author contributions and competing interests

YiCheng Pan is the named author, affiliated with KineJing Research Team. A detailed contribution statement and disclosure of relevant commercial interests require author confirmation before submission. No institutional endorsement or external independent replication is implied by this draft.

Extended results

Method	Future MSE	PSNR (dB)	SSIM
Original BWM, local BF16	0.08517601	11.050	0.54946
KineJing Pro Local, BF16	0.02464687	16.355	0.63926
Pro Local, FP8 storage	0.02439503	16.377	0.64220
RGB first-frame persistence	0.02005718	17.281	0.69815
Latent persistence	0.01720288	17.899	0.67850
VAE reconstruction (oracle)	0.00059897	32.609	0.94000

All local raw-RGB comparisons. Lower MSE and higher PSNR/SSIM are better. Ten whole trajectories, one training seed, identical local inputs and sampling configuration. The reconstruction reference consumes future reference latents and is not a predictor.

Category	KineJing-Pro	BWM	Pro rank / 60
Tri-view consistency	66.60	81.87	51
Task alignment	50.79	86.05	56
Physical / 3D coherence	62.50	60.40	6
Motion quality	31.53	41.29	47
Temporal consistency	71.22	62.81	6
Visual quality	29.54	31.42	29

Complete six-category leaderboard profile, 25 October 2026 snapshot. These systems were not run under a matched local ablation.

Supplementary company research context

The following projects have separate local records and do not constitute a jointly evaluated architecture. They document additional KineJing implementation and training work without assigning their metrics to the scored KineJing-Pro system. Source document hashes and the complete relevant limitations are retained in the accompanying company-technical-evidence record.

Learned latent dynamics and a fast residual branch

KineJing Dynamics learns a 1,341,824-parameter three-camera future-feature predictor above a frozen DINOv2 ViT-S/14 encoder [11]. An 80/10/10 episode partition and one training seed give normalized feature MSE 0.835667 versus first-frame persistence 0.991879 and shuffled actions 0.895649. Normalization uses training data only. This is a learned KineJing component, while DINOv2 supplies the pretrained visual representation. It has no RGB decoder, uses previously explored public development data and has not been integrated into the scored Pro package.

The separate Pro Fast branch fits train-only PCA and a recurrent low-rank residual, with 391,056 trainable parameters per predictor and three training seeds. Its ensemble MSE is 0.831492 versus 0.835667 for the frozen predictor; the 0.500% gain has an interval of -0.073% to 0.986%, so improvement is unconfirmed. Masking future action inputs yields 0.871077, but both branches share an action-trained frozen backbone. RGB-to-feature inference has median 25.500 ms and p95 58.707 ms over 250 fixed-input timings after 30 warmups; acquisition, file decoding and RGB generation are excluded.

State dynamics trained without pretrained weights

A separate state-input PushT experiment uses three randomly initialized networks, 139,014 parameters each (417,042 total), with no pretrained model or external encoder. Simulation data comprise 256 training, 32 validation and 32 same-distribution test episodes in the PushT environment associated with Diffusion Policy [13]. Two history states contain agent XY, block XY and block angle; controls are absolute agent target coordinates, sampled at 0.1 s intervals. Block XY RMSE at one, three and sixteen steps is 5.907, 9.543 and 31.739 pixels, compared with ridge at 8.402, 16.100 and 43.001 pixels and persistence at 12.839, 24.640 and 71.231 pixels. These are recorded offline results, not rerun in this study.

The recorded primary three-step encoded block-error reduction against ridge is 71.17%; its paired interval for candidate-minus-reference error is -0.0174103 to -0.0114703. Oracle state input and recorded actions simplify the prediction problem. The ridge baseline is better on directly actuated agent XY. These outcomes support domain-specific learned dynamics, without demonstrating video prediction, unknown-task transfer, real-robot performance or closed-loop planning.

Visual inference and simulation runtime work

KineJing Mini General loads the released V-JEPA 2 encoder and predictor [12], totaling 325,971,328 pretrained parameters; those weights were not trained by KineJing. The implementation work covers checked loading, sparse execution, graph consistency and streaming. On three selected public clips, equal-clip dense feature MSE is 0.56618249 versus persistence 1.08851823. Eight windows per profile, unknown pretraining overlap and the absence of action input or an RGB decoder sharply restrict interpretation. Dense and sparse execution use different scored patch spaces, so their accuracy values are not a controlled comparison.

A Panda simulation runtime study preserves bitwise-identical action and observation histories across sixteen paired seeds. On 176 warm paired measurements, median perception-plus-planning latency changes from 33.0751 to 20.3386 ms. The retained acceptance gate is false. This is a scoped inference-path optimization, not evidence that a new learned model improves control or outperforms AWM.

Tracked-state forecasting prototype

KineEdge combines an upstream YOLOX detector [14] with explicit temporal state and short-horizon forecasting. On oracle MOT17 trajectories, twelve history frames predict thirty future frames; one-second constant-velocity ADE is 7.25 pixels versus persistence 15.11 and fitted ridge 9.23. The 7,944 overlapping windows cover only 61 tracks and are not independent samples. On a separate 799-instance pedestrian check, recall is 38.55% for Tiny and 64.33% for S. File-replay pipeline p95 values of 31.79 and 51.41 ms therefore do not establish adequate detection quality, live-camera operation or industrial safety.

Relation to AWM and action-faithful video prediction

AWM introduces abductive state inference through an Entity-Dynamic-Relation hierarchy above a frozen predictive representation, evaluated for physical prediction, causal reasoning, action understanding and targeted interventions [8]. Our RGB adaptation, feature predictors and small image controls address different tasks. Their percentage error reductions cannot be compared with AWM task metrics to infer superiority. DreamTrue combines geometric action rendering, offline calibration and counterfactual reward-based post-training [9]. These precedents preclude claiming that action-conditioned image motion or generic counterfactual evaluation alone constitutes a new KineJing research principle.

Closed implementation and evaluation access

KineJing core implementations, model checkpoints and internal training assets remain private. This draft releases manuscript material, descriptive methods, aggregate and per-trajectory evaluation metrics, protocol receipts, public-source bibliography and checksums. It does not redistribute third-party datasets or model weights. Access sufficient for editorial and independent verification has not been arranged or approved; restrictions and any access mechanism must be resolved for the selected venue before submission.

References

1. Liu, Xuanyi et al.. TriWorldBench: A Tri-View Consistency Perspective on Embodied World Models. arXiv:2609.26314 (2026). https://arxiv.org/abs/2609.26314

2. BWM Team. BWM: A Low-Cost High-Fidelity World Simulator for Robot Learning. arXiv:2607.29302 (2026). https://arxiv.org/abs/2607.29302

3. Huang, Ziqi et al.. VBench: Comprehensive Benchmark Suite for Video Generative Models. arXiv:2311.17982 (2023). https://arxiv.org/abs/2311.17982

4. Yue, Hu et al.. EWMBench: Evaluating Scene, Motion, and Semantic Quality in Embodied World Models. arXiv:2505.09694 (2025). https://arxiv.org/abs/2505.09694

5. Shang, Yu et al.. WorldArena: A Unified Benchmark for Evaluating Perception and Functional Utility of Embodied World Models. arXiv:2602.08971 (2026). https://arxiv.org/abs/2602.08971

6. Kassis, Timothy et al.. Scientific Agent Skills: A Library of Procedural Knowledge for Research Agents. arXiv:2609.00065 (2026). https://arxiv.org/abs/2609.00065

7. Hu, Edward J. et al. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685 (2021). https://arxiv.org/abs/2106.09685

8. Liu, Ziqi et al.. Abductive World Modeling via Causal Representation Learning. arXiv:2609.36985 (2026). https://arxiv.org/abs/2609.36985

9. Li, Junyan et al.. DreamTrue: Action-Faithful Robot World Model with Counterfactual Post-Training. arXiv:2610.12468 (2026). https://arxiv.org/abs/2610.12468

10. Finn, Chelsea et al.. Unsupervised Learning for Physical Interaction through Video Prediction. arXiv:1605.07157 (2016). https://arxiv.org/abs/1605.07157

11. Oquab, Maxime et al.. DINOv2: Learning Robust Visual Features without Supervision. arXiv:2304.07193 (2023). https://arxiv.org/abs/2304.07193

12. Assran, Mido et al.. V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning. arXiv:2506.09985 (2025). https://arxiv.org/abs/2506.09985

13. Chi, Cheng et al.. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. arXiv:2303.04137 (2023). https://arxiv.org/abs/2303.04137

14. Ge, Zheng et al.. YOLOX: Exceeding YOLO Series in 2021. arXiv:2107.08430 (2021). https://arxiv.org/abs/2107.08430