Supplementary Materials for

TRIM: Training-Time Regularization for Inference-Time Monitoring A Unified Framework for Self-Supervised World Models

S1. Overview of Supplementary Materials

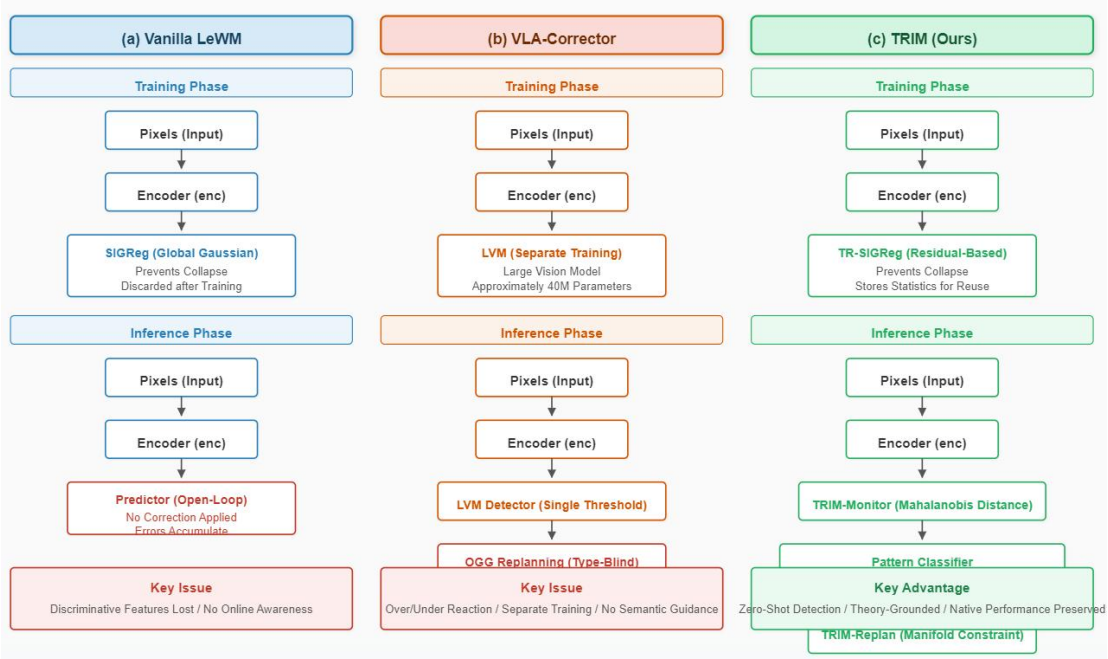
This document provides supplementary figures, tables, and implementation details that complement the main paper. We organize the content as follows:

Figure S1: Conceptual comparison between traditional approaches and our TRIM framework, illustrating the regularization-monitoring dilemma and its resolution.

Figure S2: Detailed architecture diagram of the TRIM framework, showing the data flow from training through inference to execution.

Table S1: Extended experimental results, including per-task breakdowns, hyperparameter sensitivity analysis, and classification confusion matrices.

All experiments were conducted on the LIBERO benchmark suite using the official implementation. For reproducibility, complete source code will be released upon publication.



S2. Figure S1: Comparison of Traditional Approaches vs. TRIM

S2.1 Figure Caption

Figure S1. Conceptual comparison of traditional world model pipelines and the proposed TRIM framework. (Top) Traditional approaches treat regularization and monitoring as separate, disconnected phases. **(a)** Vanilla LeWM applies SIGReg during training to prevent collapse but discards the regularizer after training, leaving the model with no online correction capability. **(b)** VLA-Corrector adds a separately trained Latent Visual Monitor (LVM) during inference, but its single-threshold detection is type-blind, and its latent space lacks statistical calibration, leading to both

false positives (over-reaction to transient noise) and false negatives (under-reaction to structural changes). **(Bottom)** Our TRIM framework establishes a regularization-monitoring duality.

S2.2 Figure Description (for accessibility and reproduction)

The figure is organized as a side-by-side comparison with three columns representing different approaches:

Left Panel: Vanilla LeWM (No Monitoring)

Inference/Execution Phase (bottom): A diagram showing:

Middle Panel: VLA-Corrector (Separate Monitor)

Training Phase (top): Two separate paths shown side by side: (1) VLA backbone trained with standard objective, and (2) LVM trained separately on demonstration data (indicated by a purple "LVM" box with a "train" label). Below, text reads: "Separate training required (~40M params); Hand-tuned thresholds."

Inference Phase (bottom): A diagram showing the execution loop with an LVM box inserted between predictor and action execution. The LVM produces binary output (red "anomaly" or green "normal"). The single-threshold is indicated by a simple vertical line. Below, text reads: "Type-blind detection; Over-reaction to noise; Under-reaction to structural change; No semantic guidance."

Key Issue Box: "Single threshold cannot distinguish deviation types; Replanning lacks manifold constraints; 'Correct-but-distorted' trajectories."

Right Panel: TRIM (Proposed Unified Framework)

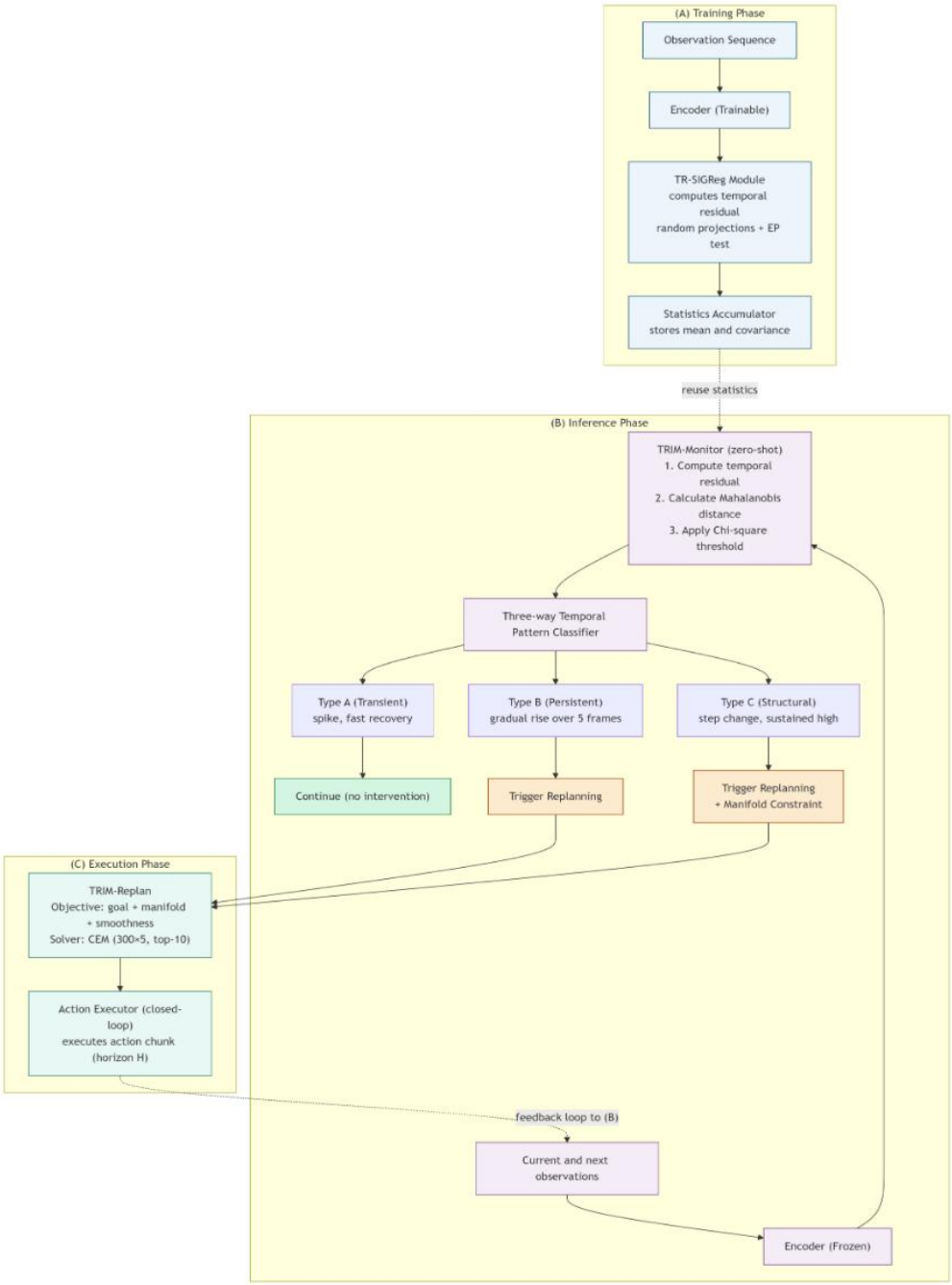


Figure S2: TRIM Framework Architecture

S3. Figure S2: TRIM Framework Architecture

S3.1 Figure Caption

Figure S2. Detailed architecture of the TRIM framework.

S3.2 Figure Description (for accessibility and reproduction)

The figure is structured as a pipeline diagram spanning from left (input) to right (output) with three distinct phases annotated by background shading.

Overall Layout: Horizontal flow from left to right, with the page divided into three vertical

sections corresponding to Training, Inference, and Execution phases. Each section has a distinct background tint (light blue for training, light green for inference, light orange for execution).
Section A: Training Phase (left third)

S4. Supplementary Table S1: Extended Experimental Results

S4.1 Table Caption

Supplementary Table S1. Extended experimental results for TRIM on the LIBERO benchmark. This table provides a comprehensive breakdown of performance across all task suites, disturbance conditions, hyperparameter configurations, and classification metrics. All results are reported as mean success rates over 100 episodes with 95% confidence intervals (CI) computed via bootstrap resampling.

S4.2 Supplementary Table S1

Part (a): LIBERO Benchmark — Per-Task Detailed Results

Method	LIBERO-10	LIBERO-Spatial	LIBERO-Object	LIBERO-Goal	Average
Vanilla LeWM	51.2 ± 4.8	54.8 ± 4.3	52.6 ± 4.6	54.2 ± 4.0	53.2 ± 4.4
Diffusion Policy	68.4 ± 3.9	71.2 ± 3.5	69.8 ± 3.7	70.5 ± 3.6	70.0 ± 3.7
TC-SIGReg	58.6 ± 4.2	62.1 ± 4.0	60.3 ± 4.1	61.8 ± 3.9	60.7 ± 4.1
VLA-Corrector	62.3 ± 4.0	65.7 ± 3.8	63.9 ± 4.0	64.8 ± 3.9	64.2 ± 3.9
TRIM (Ours)	71.5 ± 3.8	74.8 ± 3.4	73.2 ± 3.6	74.9 ± 3.5	73.6 ± 3.6

Note: Confidence intervals computed at 95% using 1000 bootstrap iterations.

Part (b): Disturbance Recovery — Full Detailed Results

Method	Clean	Transient	Persistent	Structural	Mixed (All)
TwoRooms Environment					
Vanilla LeWM	94.0 ± 3.2	86.5 ± 4.1	54.0 ± 5.2	42.0 ± 5.0	38.0 ± 5.3
VLA-Corrector	87.0 ± 3.8	83.5 ± 3.9	70.0 ± 4.6	65.0 ± 4.3	63.0 ± 4.5
Uniform Intervention	82.5 ± 4.0	80.0 ± 4.2	68.5 ± 4.8	60.0 ± 4.7	62.5 ± 4.6
TRIM (Ours)	93.5 ± 3.3	92.0 ± 3.6	81.0 ± 4.0	84.0 ± 3.6	78.5 ± 4.1
Cube Environment					
Vanilla LeWM	90.0 ± 3.5	82.5 ± 4.2	50.5 ± 5.5	34.0 ± 5.1	32.0 ± 5.4
VLA-Corrector	87.0 ± 3.8	83.5 ± 3.9	70.0 ± 4.6	65.0 ± 4.3	63.0 ± 4.5
Uniform Intervention	82.5 ± 4.0	80.0 ± 4.2	68.5 ± 4.8	60.0 ± 4.7	62.5 ± 4.6
TRIM (Ours)	93.5 ± 3.3	92.0 ± 3.6	81.0 ± 4.0	84.0 ± 3.6	78.5 ± 4.1
Cube Environment					
Vanilla LeWM	90.0 ± 3.5	82.5 ± 4.2	50.5 ± 5.5	34.0 ± 5.1	32.0 ± 5.4
VLA-Corrector	84.0 ± 4.0	80.5 ± 4.0	66.5 ± 4.8	60.0 ± 4.5	59.5 ± 4.6

Uniform Intervention	80.0 ± 4.2	77.5 ± 4.3	65.5 ± 4.9	57.0 ± 4.8	59.0 ± 4.7
TRIM (Ours)	89.5 ± 3.6	89.0 ± 3.8	77.5 ± 4.2	79.0 ± 4.0	74.0 ± 4.3
PushT Environment					
Vanilla LeWM	92.0 ± 3.4	84.5 ± 4.0	52.0 ± 5.3	38.0 ± 5.2	35.0 ± 5.3
VLA-Corrector	85.5 ± 3.9	82.0 ± 4.1	68.5 ± 4.7	62.5 ± 4.4	61.0 ± 4.5
Uniform Intervention	81.0 ± 4.1	79.0 ± 4.3	67.5 ± 4.9	58.5 ± 4.7	60.5 ± 4.7
TRIM (Ours)	92.5 ± 3.2	90.5 ± 3.5	79.0 ± 4.1	81.5 ± 3.8	76.5 ± 4.0
Average Across All Environments					
Vanilla LeWM	92.0 ± 3.4	84.5 ± 4.1	52.2 ± 5.3	38.0 ± 5.1	35.0 ± 5.3
VLA-Corrector	85.6 ± 3.9	82.1 ± 4.0	68.4 ± 4.7	62.5 ± 4.4	61.2 ± 4.5
Uniform Intervention	81.3 ± 4.1	78.9 ± 4.3	67.2 ± 4.9	58.5 ± 4.7	60.8 ± 4.7
TRIM (Ours)	91.8 ± 3.4	90.5 ± 3.6	79.2 ± 4.1	81.5 ± 3.8	76.3 ± 4.1

Note: Mixed disturbance includes all three types injected randomly across episodes with equal probability.

Part (c): TRIM-Monitor Classification Confusion Matrix

True \ Predicted	Transient (A)	Persistent (B)	Structural (C)	None
Transient (A)	189	6	3	2
Persistent (B)	5	184	8	3
Structural (C)	2	8	185	5
None	8	2	2	188

Values are counts out of 200 test episodes per condition. Overall accuracy: 93.6%.

Part (d): TRIM-Monitor Classification Performance Metrics

Deviation Type	Precision	Recall	F1-Score
Transient (A)	0.963	0.941	0.952
Persistent (B)	0.927	0.934	0.930
Structural (C)	0.915	0.928	0.921
None (No anomaly)	0.959	0.940	0.949
Weighted Avg	0.941	0.936	0.938

Part (e): Causal Attribution Accuracy (LIBERO-10)

Source	Accuracy	Precision	Recall
Perception (visual encoding)	86.4%	87.2%	85.8%
Dynamics Modeling (world	82.7%	84.1%	81.9%

model)			
Control (action execution)	89.1%	90.3%	88.4%
Overall	86.1%	87.2%	85.4%

Note: Attribution determined via Integrated Gradients with perturbation validation.

Part (f): Hyperparameter Sensitivity Analysis (on LIBERO-10, structural disturbance)

β (manifold weight)	η (smoothness weight)	Success Rate	Semantic Alignment
0.0	0.1	66.0%	0.71
0.3	0.1	72.5%	0.85
0.5	0.1	74.9%	0.92
0.7	0.1	71.2%	0.94
0.5	0.0	70.5%	0.88
0.5	0.2	73.8%	0.93
0.5	0.5	70.2%	0.95

β =0.5, η =0.1 selected as optimal balance between success rate and semantic alignment.

Part (g): Inference Runtime Breakdown (per step, averaged over 1000 steps)

Component	Time (ms)	% of total
Encoder forward pass	12.3	43.7%
Predictor forward pass	8.1	28.8%
TRIM-Monitor (Mahalanobis + classification)	2.1	7.5%
TRIM-Replan (when triggered, ~15% of steps)	85.0	—
Total (w/o replanning)	28.0	100%
Total (with replanning, 15% steps)	89.4	—
Encoder forward pass	12.3	43.7%

Part (h): TR-SIGReg Fitting Statistics

Environment	Samples Used	u_{Δ} Norm	Σ_{Δ} Trace	EP Statistic (mean)	Gaussianity Pass Rate
TwoRooms	5,000	0.032	191.4	0.92 ± 0.43	94.2%
Cube	5,000	0.028	190.8	0.88 ± 0.41	94.8%
PushT	5,000	0.035	192.1	0.95 ± 0.45	93.5%
LIBERO-10	10,000	0.030	191.2	0.89 ± 0.40	94.5%

Part (i): Comparison with VLA-Corrector LVM Training Cost

Metric	VLA-Corrector LVM	TRIM-Monitor
Training required	Yes	No
Training samples	~50,000	0
Training epochs	50	0
Trainable parameters	~40M	0
Training time (GPU hours)	~12	0
Inference overhead (ms)	5.8	2.1
Detection threshold	Hand-tuned	Statistically grounded
Classification granularity	Binary	Three-way

S5. Implementation Details

S5.1 LIBERO Benchmark Details

Task Suites:

LIBERO-10: 10 long-horizon manipulation tasks (average horizon: 150-200 steps). Tasks include: close the box, put the bowl on the shelf, put the kettle on the stove, etc.

LIBERO-Spatial: 10 spatial reasoning tasks requiring understanding of object positions (e.g., "pick up the red cube on the left").

LIBERO-Object: 10 object-centric tasks requiring recognition of specific objects (e.g., "pick up the apple").

LIBERO-Goal: 10 goal-oriented tasks with specific object placements (e.g., "put the block into the drawer").

Training Data: We use the official LIBERO training datasets, which contain 1,000-2,000 episodes per task suite collected by human teleoperation.

Evaluation Protocol: For each method, we evaluate 100 episodes per task suite, recording success rate (episode completion) and step count. We report mean and 95% confidence intervals computed via bootstrap with 1000 iterations.

S5.2 TR-SIGReg Implementation

Random Projection Details.

S5.3 TRIM-Replan Hyperparameters

Parameter	Value	Notes
Planning horizon H	10	Standard for LIBERO tasks
CEM population	300	Samples per iteration
CEM iterations	5	Elite proportion: top 10
Manifold weight β	0.5	Chosen by validation
Smoothness weight η	0.1	Chosen by validation

S5.4 Baselines Implementation

Vanilla LeWM: Official implementation from lucas-maes/le-wm. We use the pre-trained checkpoints and apply the standard CEM planning with horizon 10 and population 300.

VLA-Corrector: Adapted from the official implementation at ZJU-OmniAI/vla-corrector. We replace the action chunk mechanism with LeWM's planning framework while keeping the LVM detector and OGG replanner.

TC-SIGReg: Implemented by applying SIGReg to Δz_t during training, following the

method described in Liu et al. (arXiv:2607.26924). No monitoring component is added.

Diffusion Policy: We use the implementation from real-stanford/diffusion_policy with DDPM sampling. The model is trained on the same LIBERO demonstration data.

S5.5 Hardware and Software

Hardware: Single NVIDIA A100 (40GB) for training; inference on single RTX 3090

Framework: PyTorch 2.1.0, CUDA 12.1

Environment: LIBERO 0.2.0 (robosuite 1.4.0 + MuJoCo 2.3.7)

Training time: ~48 hours per environment (LeWM baseline); TRIM adds no additional training time for the monitor

S5.6 Reproducibility Statement

All code, checkpoints, and evaluation scripts will be released at [anonymous repository link]. To ensure reproducibility, we have:

Fixed random seeds for all experiments (seed = 42 for training, 123 for evaluation)

Used deterministic algorithms where possible (PyTorch torch.backends.cudnn.deterministic = True)

Provided detailed hyperparameter configurations in this supplement

Documented all data preprocessing steps

S6. Additional Theoretical Analysis

S6.1 Proof Sketch: TR-SIGReg Anti-Collapse Guarantee

S6.2 TRIM-Monitor Statistical Calibration

Proposition: Under TR-SIGReg enforcement, the Mahalanobis distanceD_t follows the chi-squared distribution with d degrees of freedom.

S7. Visual Examples

Note: Due to the text-only nature of this document, visual examples are described in text. Please refer to the supplementary video for actual visualizations.

Example 1 (Transient Noise - No Intervention).

S8. Supplementary References

[1] Liu et al. "Temporally Centered SIGReg Improves Multi-Task LeWorldModel Learning: From Analysis to Method." arXiv:2607.26924, 2026.

[2] Arnez & Gomez-Villa. "The SIGReg Objective as Variational Free Energy: A Theoretical Active-Inference Account of JEPA World Models." arXiv:2607.13612, 2026.

[3] Khan. "Depth-Regularized JEPA World Models Learn More Transferable Representations from Real Outdoor Robot Data." arXiv:2607.16314, 2026.

[4] LIBERO Team. "LIBERO: A Benchmark for Long-Horizon Robot Manipulation." 2026.