REGULARIZING GEOMETRY AND ERROR PROPAGATION IN LATENT WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

A latent world model can have a well-regularized representation distribution yet make unreliable multi-step predictions. We study this mismatch through two complementary regularization targets: marginal representation geometry and temporal error cancellation. We combine an anchored ellipsoidal extension of SIGReg with hierarchical rollout gaps (HGap). The ellipsoidal term allows directional target variances while retaining an isotropic anchor. HGap decomposes rollout error into differences between hybrid trajectories of the current model, then penalizes cancellation between groups of these contributions. We characterize the resulting hierarchy of upper bounds in a shared positive-definite metric. Across four visual control tasks, the combined method achieves a mean success rate of 85.50%, compared with 79.25% for our LeWM reproduction, an improvement of 6.25 percentage points. These rates average six evaluation seeds for one trained model per task. Separate component studies examine geometry regularization and frozen-geometry adaptation under their respective protocols. Anchored ellipsoidal regularization achieves a mean success rate of 84.17%. With visual representations frozen, HGap adaptation improves mean success from 43.75% to 47.19% at long horizons and from 81.13% to 82.56% at short horizons. The framework combines directional representation regularization with an explicit measure of temporal error cancellation in a common prediction objective.

1 INTRODUCTION

Latent world models enable planning by predicting the consequences of actions in a compact representation space. For image-goal control, the same space must preserve information about the scene, support action-conditioned prediction, and provide a useful distance to the goal. Joint-embedding approaches learn this space without reconstructing pixels. LeJEPA regularizes embeddings toward an isotropic Gaussian through SIGReg (Balestriero & LeCun, 2025), and LeWorldModel combines Gaussian regularization with latent prediction for end-to-end learning from visual experience (Maes et al., 2026). This formulation addresses representation collapse, but leaves a further question: what should be regularized to make the learned space useful over a sequence of predictions?

Two properties matter. The first is *marginal geometry*: how variation is distributed across latent directions. An isotropic target treats these directions uniformly, even though the data may exhibit directional structure. An anchored ellipsoidal target relaxes this restriction while retaining an isotropic distributional constraint. The second is *temporal error composition*: how errors introduced at different steps combine along a rollout. Large propagated contributions can oppose one another, producing a small final error. The final prediction loss measures their sum, but does not distinguish small contributions from large, mutually cancelling ones. A constraint on the marginal representation distribution does not directly resolve this distinction.

We study these properties through a formulation combining anchored ellipsoidal regularization and hierarchical rollout gaps (Figure 1). The geometry term interpolates between isotropic SIGReg and a data-dependent ellipsoidal target. The isotropic component acts as an anchor while the ellipsoidal component introduces directional structure. The temporal term constructs hybrid trajectories by progressively replacing predicted history with observed embeddings. Differences between consecutive hybrids telescope exactly to the rollout error. Grouping these differences produces a hierarchy of

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

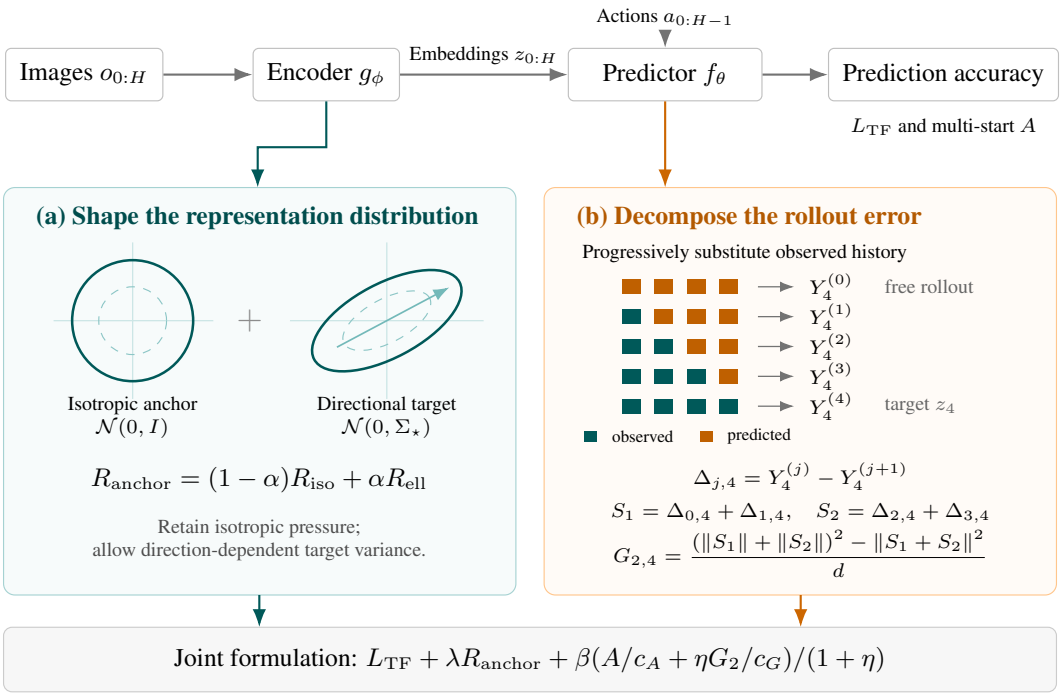

Figure 1: **Regularizing geometry and error propagation.** A shared visual encoder and action-conditioned predictor supply two regularization branches. **(a)** The geometry loss combines isotropic and ellipsoidal target discrepancies; contours illustrate targets, not measured embeddings. **(b)** A four-step example shows hybrid histories, their terminal predictions, and two groups of consecutive prediction differences. HGap penalizes cancellation between group sums while leaving within-group cancellation intact. The diagram uses Euclidean norms and depicts the joint formulation; the experiments report the combined method and separate component studies.

norm bounds; the gap between a bound and the actual error measures cancellation across groups. This construction uses the current nonlinear predictor without a Jacobian approximation.

The two terms act on different aspects of the learning problem. The geometry term shapes the embedding distribution in which predictions are learned. HGap distinguishes accurate rollouts with small error contributions from those with large contributions that cancel. The joint objective combines these signals with prediction accuracy, and the component studies examine their respective effects on control performance.

Across four visual control tasks, the combined method achieves a mean success rate of 85.50%, exceeding our LeWM reproduction by 6.25 percentage points (Table 1). Separate component studies examine the two regularizers under their respective protocols. Anchored ellipsoidal regularization improves mean success by 4.92 percentage points, with gains on three tasks and a 0.66-point decrease on PushT. HGap adaptation with frozen visual geometry improves the two-seed average on all four tasks at both horizons, with mean gains of 3.44 and 1.44 percentage points for long and short horizons, respectively.

This paper makes three contributions. First, it formulates anchored ellipsoidal regularization and HGap as two components targeting marginal geometry and temporal error composition. Second, it characterizes the hierarchical bounds and shows that their gap is a magnitude-weighted measure of angular disagreement among grouped error contributions. Third, it evaluates the combined method across four visual control tasks and examines the roles of geometry regularization and frozen-geometry adaptation in separate component studies.

2 Related Work

World models for visual control. World Models learns spatial and temporal representations for a compact controller (Ha & Schmidhuber, 2018). PlaNet plans actions through latent dynamics (Hafner et al., 2019b), Dreamer learns behaviors through latent imagination (Hafner et al., 2019a), and TD-MPC2 scales control with an implicit, decoder-free world model (Hansen et al., 2023). These methods motivate predicting in a compact space, but differ in how supervision makes that space useful for control. Our setting is reward-free image-goal planning, where a latent prediction and its distance to an encoded goal guide action selection.

Joint-embedding prediction and planning. I-JEPA predicts masked image representations (Assran et al., 2023), and V-JEPA learns visual representations by predicting video features (Bardes et al., 2024). DINO-WM uses pretrained DINOv2 features for action-conditioned world modeling (Zhou et al., 2024); PLDM learns JEPA latent dynamics from reward-free offline data (Sobal et al., 2025); and V-JEPA 2 combines video pretraining with action-conditioned post-training for robotic planning (Assran et al., 2025). LeWorldModel learns representations and dynamics end-to-end with prediction and Gaussian regularization (Maes et al., 2026). Our approach builds on this last formulation. Our frozen-geometry experiments additionally examine the separate setting in which only dynamics are adapted.

Regularization against representation collapse. BYOL and SimSiam use architectural or optimization asymmetries, including a moving-average target and stop-gradient operations (Grill et al., 2020; Chen & He, 2020). Barlow Twins regularizes cross-view correlations (Zbontar et al., 2021), while VICReg separates invariance, variance preservation, and covariance regularization (Bardes et al., 2021). LeJEPA's SIGReg matches random projections to Gaussian targets (Balestriero & LeCun, 2025); VISReg separates variance control from distributional shape matching using a sliced-Wasserstein sketching objective (Wu et al., 2026). Our geometry term retains the SIGReg discrepancy and modifies part of its target. Thus, its intended role is to adjust the shape of the constraint within an existing anti-collapse formulation.

Structured representation geometry. Sub-JEPA constrains random subspaces (Zhao et al., 2026), whereas No Gaussian Required uses action-contrastive inverse dynamics as a transition-dependent alternative to a prescribed Gaussian distribution (Boylan & Hokamp, 2026). Beyond Isotropy derives structured Gaussian covariances under a specified quadratic geometry and distinguishes marginal constraints from predictive coupling (Alvarez, 2026). Our construction uses data-dependent directional target variances together with an isotropic anchor. Temporal Straightening instead regularizes the curvature of latent trajectories for planning (Wang et al., 2026). HGap operates on propagated *prediction-error contributions*, rather than state-trajectory curvature. These distinctions separate three choices: shaping the embedding distribution, shaping state trajectories, and shaping the composition of prediction errors.

Multi-step prediction and error propagation. Scheduled Sampling mixes observed and generated inputs (Bengio et al., 2015); Professor Forcing matches teacher-forced and free-running recurrent behavior (Lamb et al., 2016); and PlaNet uses latent overshooting for multi-step consistency (Hafner et al., 2019b). Multi-Horizon Consistency as Geometry relates multi-step agreement to empirical dynamical expansion (Bhand & Joshi, 2026). HGap also evaluates generated histories, but uses explicit hybrid-trajectory differences to expose cancellation beyond the final prediction error. Related analyses connect model smoothness to multi-step prediction error (Asadi et al., 2018) and show the importance of recovery from erroneous model states (Talvitie, 2016). Our bound applies directly to generated branches without a contraction assumption. It characterizes cancellation, not closed-loop performance; useful self-correction is one reason a smaller gap need not improve control.

3 Representation Geometry and Temporal Error

Our formulation has two components: a distributional constraint on embeddings and a temporal constraint on rollout errors. We first describe each component and then state the joint objective.

Let $z_t = g_\phi(o_t) \in \mathbb{R}^d$ denote an observation embedding. An action-conditioned predictor f_θ maps an available latent history and actions to a future embedding. Teacher-forced prediction uses encoded observations in that history; a free rollout recursively uses predictions. Denote the one-step teacher-forced loss by L_{TF}.

For a clip $(o_0, a_0, \ldots, a_{H-1}, o_H)$, write $\mathcal{H}_t$ for the observed history available to the predictor. A generic dimension-normalized prediction objective is

$$L_{\mathrm{TF}} = \mathbb{E}_{\mathcal{D}}\left[\frac{1}{Hd}\sum_{t=0}^{H-1} \|f_\theta(\mathcal{H}_t, a_t) - z_{t+1}\|_2^2\right]. \tag{1}$$

This notation abstracts the implementation's context window and minibatch reduction. The target representation and predictor share a latent space, so changes to the encoder affect both the marginal regularizer and the prediction residuals.

The geometry regularizer acts on the marginal distribution of z_t; HGap acts on predictions made by f_θ. We keep these roles distinct from the goal cost used by the planner. In particular, representation covariance alone specifies neither prediction uncertainty nor task importance.

3.1 Anchored Ellipsoidal SIGReg

The ellipsoidal component allows the Gaussian target to assign different variances to different directions. For a batch of embeddings Z, let $\widehat{\varphi}_{u^\top Z}(\omega)$ denote the empirical characteristic function along a unit direction u. A Gaussian target with covariance $\Sigma_\star$ has directional characteristic function

$$\varphi_\star(\omega; u) = \exp\left(-\tfrac{1}{2}\omega^2 u^\top \Sigma_\star u\right). \tag{2}$$

Using the same type of projection and frequency discrepancy as SIGReg, define

$$R_{\mathrm{ell}}(Z; \Sigma_\star) = \mathbb{E}_{u,\omega}\left|\widehat{\varphi}_{u^\top Z}(\omega) - \varphi_\star(\omega; u)\right|^2. \tag{3}$$

The expectation is evaluated with a finite projection and frequency scheme. Setting $\Sigma_\star = I$ gives the corresponding isotropic objective; finite matching approximates the target rather than certifying joint Gaussianity.

To specify directional structure, the ellipsoidal design estimates a principal subspace from centered embeddings and constructs

$$\Sigma_\star = \sigma_\parallel^2 P_k + \sigma_\perp^2 (I - P_k), \qquad P_k = V_{1:k} V_{1:k}^\top. \tag{4}$$

Here C denotes the empirical covariance, and $V_{1:k}$ contains its leading eigenvectors. The dimension statistic $\mathrm{tr}(C)/\|C\|_{\mathrm{op}}$ is the stable rank of $C^{1/2}$. The statistic measures spectral concentration; the integer subspace dimension k determines how many leading directions receive the parallel target variance. We use target standard deviations $\sigma_\parallel = 1.0$ and $\sigma_\perp = 0.20$, giving target variances 1.0 and 0.04. Appendix R summarizes the numerical loss coefficients; Appendix S specifies rank selection, target updates, and projection sampling. The characteristic-function loss acts on raw embeddings to retain the zero-mean target; centering is used for covariance estimation.

Rather than fully replacing the isotropic target, we interpolate the two discrepancies:

$$R_{\mathrm{anchor}} = (1-\alpha)R_{\mathrm{iso}} + \alpha R_{\mathrm{ell}}, \qquad L_{\mathrm{ell}} = L_{\mathrm{TF}} + \lambda R_{\mathrm{anchor}}. \tag{5}$$

An interpolation of discrepancies is different from simply interpolating Gaussian covariance matrices. For a fixed ellipsoidal target and a common projection measure, let $a = \widehat{\varphi}$, $b = \varphi_I$, and $c = \varphi_{\Sigma_\star}$. Then

$$(1-\alpha)|a-b|^2 + \alpha|a-c|^2 = |a - [(1-\alpha)b + \alpha c]|^2 + \alpha(1-\alpha)|b-c|^2. \tag{6}$$

Thus, conditional on a fixed target, the anchor matches a mixture characteristic function up to an embedding-independent term. The corresponding idealized distribution is a mixture of two centered Gaussians, not generally a single Gaussian. Appendix F derives the distinction and explains why a target estimated from the current embeddings requires additional care.

The evaluated anchored ellipsoidal regularizer uses $\alpha = 0.1$, retaining most of the isotropic regularization weight. This anchor limits the departure from the baseline objective, but does not guarantee preservation of task-relevant information. The outer coefficient λ controls the overall strength of distributional regularization, while α controls the relative contribution of the two target discrepancies. Appendices E–G derive the target spectrum, anchor interpretation, and finite-estimator gradients.

Directional flexibility without removing the scale constraint. For a unit vector u, the target variance in Equation (4) is

$$u^\top \Sigma_\star u = \sigma_\perp^2 + (\sigma_\parallel^2 - \sigma_\perp^2)\,\|P_k u\|_2^2\,. \tag{7}$$

Consequently, positive choices of both variances bound every directional target variance between their minimum and maximum. The target permits anisotropy while retaining a nonzero scale in the complementary subspace. This is a property of the target, not a guarantee that a finite empirical discrepancy prevents collapse. The isotropic anchor additionally retains the baseline's distributional preference. Together, these choices relax equal directional variances without making the target unconstrained; they do not assume that high-variance directions are necessarily the most useful for control.

3.2 Hierarchical Rollout Gaps

HGap distinguishes a small rollout error from large error contributions that cancel. To obtain these contributions without linearizing the dynamics, fix a target time t and an observed action sequence. Construct hybrid predictions $Y_t^{(j)}$ for $j = 0, \dots, t$: $Y_t^{(0)}$ is the free prediction from the initial observation; increasing j replaces more of the early predicted history with observed embeddings; $Y_t^{(t)} = z_t$. The predictor and available-history rules are the same across branches. Define

$$\Delta_{j,t} = Y_t^{(j)} - Y_t^{(j+1)}, \qquad j = 0, \dots, t-1. \tag{8}$$

Then, without a linearization,

$$e_t := Y_t^{(0)} - z_t = \sum_{j=0}^{t-1} \Delta_{j,t}. \tag{9}$$

Each $\Delta_{j,t}$ measures the change in the terminal prediction after one additional history substitution. The identity is exact for the generated branches. A deterministic intervention interpretation also requires coupled stochastic layers; the frozen-geometry experiments disable dropout.

From contributions to a hierarchy. For a partition $\mathcal{P}_t$ of the intervention indices, define $S_{B,t} = \sum_{j \in B} \Delta_{j,t}$. With a positive-definite matrix Q shared by all branches at time t, let $\|x\|_Q = \sqrt{x^\top Q x}$ and

$$E_t^Q = \frac{\|e_t\|_Q^2}{d}, \qquad U_{\mathcal{P},t}^Q = \frac{1}{d}\left(\sum_{B \in \mathcal{P}_t} \|S_{B,t}\|_Q\right)^2. \tag{10}$$

Proposition 1 (Hierarchy and cancellation). *For the same branch arrays and shared $Q \succ 0$, if $\mathcal{P}_f$ refines $\mathcal{P}_c$, then*

$$E_t^Q \le U_{\mathcal{P}_c,t}^Q \le U_{\mathcal{P}_f,t}^Q. \tag{11}$$

For any partition with blocks ordered arbitrarily,

$$U_{\mathcal{P},t}^Q - E_t^Q = \frac{2}{d} \sum_{B<C} \left(\|S_{B,t}\|_Q \, \|S_{C,t}\|_Q - S_{B,t}^\top Q S_{C,t}\right) \ge 0. \tag{12}$$

The proof follows by telescoping, the triangle inequality, expansion of the squared norm, and Cauchy–Schwarz; see Appendix A. A single block gives exact rollout error; singleton blocks give the finest sum-of-norms bound. Equation (12) shows that the penalty measures cross-block angular disagreement weighted by contribution magnitudes. It does not penalize cancellation inside each block. Appendices I and J give equality conditions, the exact change under refinement, and a normalized angular interpretation. Cancellation can also be physically appropriate, so reducing this gap is not universally desirable.

What final-error supervision cannot distinguish. Consider two block sums with a fixed terminal residual e:

$$S_1 = \tfrac{1}{2}e + v, \qquad S_2 = \tfrac{1}{2}e - v. \tag{13}$$

Every choice of v gives the same rollout loss $\|e\|_Q^2/d$, although the contributions can differ substantially. At $v = 0$, the gap is zero; when $e = 0$, the gap is $4\,\|v\|_Q^2/d$. Thus, endpoint accuracy alone

cannot distinguish vanishing contributions from large mutually cancelling ones. HGap supplies this additional training signal through the generated hybrid branches. Grouping makes the preference selective: HGap-2 preserves cancellation within a pair and acts only on disagreement between pair sums. This example explains the objective's distinct role without implying that every cancellation is undesirable or that arbitrary block configurations are realizable by the predictor.

The HGap-2 objective. Grouping controls which cancellations are penalized. HGap-2 uses contiguous blocks of at most two intervention differences, retaining cancellation within each pair while penalizing it across pairs. For seven prediction times, we use temporal weights $q_t = t/28$. Define

$$W^Q = \sum_{t=1}^{7} q_t E_t^Q, \qquad G_2^Q = \sum_{t=1}^{7} q_t \left(U_{2,t}^Q - E_t^Q \right). \tag{14}$$

Let $\widehat{z}_t^{[s]}$ be the prediction at t obtained by rolling out from observed history ending at s. Multi-start supervision averages all start–target pairs:

$$A = \frac{2}{H(H+1)d} \sum_{s=0}^{H-1} \sum_{t=s+1}^{H} \left\| \widehat{z}_t^{[s]} - z_t \right\|_2^2, \qquad H = 7. \tag{15}$$

There are 28 pairs in an eight-frame clip. This objective weights individual start–target pairs uniformly, unlike W^Q, which uses $q_t = t/28$ for errors from a fixed initial history. Appendix M derives the induced horizon weights. The reported frozen-geometry HGap adaptation uses $Q = I$ and

$$L_{\text{adapt}} = L_{\text{TF}} + 0.25 \frac{A/c_A + 0.25 G_2^I / c_G}{1.25}, \tag{16}$$

where positive scales are calibrated on a fixed set of 512 training clips and then held constant. Appendix L derives the optimization signals and distinguishes this calibration from adaptive detached normalization.

3.3 Combining Geometry and Temporal Regularization

Our joint objective uses the Euclidean HGap metric:

$$\boxed{L_{\text{joint}} = L_{\text{TF}} + \lambda R_{\text{anchor}} + \beta \frac{A/c_A + \eta G_2^I / c_G}{1+\eta}.} \tag{17}$$

Equation (17) combines the geometry and temporal regularizers. Setting $\beta = 0$ removes the entire temporal auxiliary term, including multi-start accuracy and HGap. Setting $\eta = 0$ removes the gap penalty and retains multi-start accuracy, with its coefficient changing from $\beta/(1+\eta)$ to β. Setting $\alpha = 0$ restores isotropic geometry regularization. The coefficients λ and β weight the geometry and temporal terms, respectively, while η controls the relative contribution of the normalized gap within the temporal term.

We use $\lambda = 0.09$, $\beta = 0.25$, and $\eta = 0.25$, with $\alpha = 0.1$ for the geometry mixture. Appendix R gives the resulting explicit loss coefficients.

The joint objective couples two regularization signals through the learned embeddings: R_{anchor} shapes their marginal distribution, while G_2^I penalizes cross-group cancellation in the predictor. Whether this coupling is beneficial depends on the task and on how temporal gradients affect representation learning.

The analysis above allows any common positive-definite metric to make the assumptions of the bound explicit. The reported HGap adaptation uses the Euclidean case $Q = I$. Appendix P explains why marginal regularization, prediction accuracy, and grouped cancellation supply different optimization signals.

4 Experiments

We organize the evaluation around the combined objective and the roles of its two components. Section 4.2 reports the main results of joint regularization. Sections 4.3 and 4.4 present component analyses for ellipsoidal geometry and HGap adaptation, respectively. The component studies use their respective training and evaluation protocols.

4.1 Experimental Setup

The experiments concern TwoRoom, PushT, Reacher, and Cube, using image observations and actions. The LeWM-based model uses a 192-dimensional latent space and a ViT-tiny visual encoder. We preserve the baseline and evaluation protocol of each experiment family. Training seeds, adaptation seeds, planner seeds, and sampled starting states measure different sources of variation and are reported separately.

The ellipsoidal study uses one training seed and six evaluation seeds. HGap adaptation uses two continuation seeds from the same task-specific checkpoint, with 200 evaluation starts per task and horizon. The encoder, projector, and BatchNorm buffers remain frozen during adaptation. Appendix C gives the horizon and planning settings.

Joint training and evaluation. The combined model uses one training seed per task and is trained for 10 epochs with AdamW, learning rate 10^{-4}, and batch size 64. Each model is evaluated with six evaluation seeds and 50 initial states per seed and task. Task scores are averaged over evaluation seeds, and the overall score weights tasks equally. Appendix R summarizes the configuration.

4.2 Joint Regularization: Main Results

This section reports the combined effect of anchored ellipsoidal regularization and HGap using the objective in Equation (17).

Table 1 reports success rates of 98.00%, 69.00%, 92.67%, and 82.33% on TwoRoom, Reacher, PushT, and Cube, respectively. The equally weighted four-task mean is 85.50%, compared with 79.25% for our LeWM reproduction. The reported differences are positive on all four tasks, with the largest gains on TwoRoom and Cube. The mean difference is 6.25 percentage points.

Table 1: Main results for joint regularization: success rates (%). LeWM results are from our own reproduction, not copied from the original paper, and are averaged over six evaluation seeds for one trained model. Ours denotes the combined method, trained with one seed per task and evaluated with six seeds and 50 initial states per evaluation seed. Means weight tasks equally; differences are computed from the reported task rates.

Method	TwoRoom	Reacher	PushT	Cube	Mean
LeWM (our reproduction)	87.67	67.33	91.33	70.67	79.25
Ours	98.00	69.00	92.67	82.33	85.50
Difference (pp)	+10.33	+1.67	+1.34	+11.66	+6.25

4.3 Component Analysis: Ellipsoidal Geometry

We first examine the geometry component by comparing anchored ellipsoidal regularization with the isotropic baseline, without introducing HGap in this experiment series.

Anchored ellipsoidal regularization improves the four-task mean from 79.25% to 84.17% relative to the baseline trained in the same series (Table 2). The reported differences are +8.66 points on TwoRoom, +3.34 on Reacher, -0.66 on PushT, and +8.33 on Cube. The largest improvements occur on TwoRoom and Cube. The six-seed averages characterize evaluation variation for fixed models, not robustness across independent training runs.

4.4 Component Analysis: Frozen-Geometry HGap Adaptation

We next examine the temporal component with visual geometry frozen. This study evaluates the full HGap adaptation objective, including multi-start accuracy and the gap penalty, relative to its base checkpoint.

With visual geometry fixed, seed-averaged HGap adaptation improves long-horizon success on all four tasks (Table 3). The equally weighted four-task mean rises from 43.75% to 47.19%; the corresponding short-horizon means are 81.13% and 82.56%. The short-horizon point estimates also

Table 2: Ellipsoidal geometry component analysis. The LeWM baseline is our own reproduction, not the result reported in the original paper. Success rates (%) averaged over six evaluation seeds, with one training seed. Each entry is an arithmetic mean over evaluation seeds. Appendix B specifies the aggregation and its interpretation.

Method	TwoRoom	Reacher	PushT	Cube	Mean
LeWM (our reproduction)	87.67	67.33	91.33	70.67	79.25
Ellipsoid	96.33	70.67	90.67	79.00	84.17
Difference (pp)	+8.66	+3.34	-0.66	+8.33	+4.92

improve on all four tasks, with Cube increasing from 76.50% to 78.25%. These results characterize the complete adaptation objective under a fixed-representation protocol.

Table 3: HGap adaptation: success rates (%), averaged over two adaptation seeds with 200 starts per seed. The two adaptation runs share a base checkpoint and the same 200 evaluation starts. A dagger identifies documented reuse of evaluation cohorts from earlier experiments; it does not distinguish held-out from non-held-out cohorts.

Task	Horizon	Base	HGap mean	Difference (pp)
TwoRoom	Long	37.50	41.75	+4.25
TwoRoom	Short†	93.50	94.00	+0.50
PushT	Long†	21.50	24.50	+3.00
PushT	Short	91.00	92.50	+1.50
Reacher	Long†	57.50	62.75	+5.25
Reacher	Short	63.50	65.50	+2.00
Cube	Long	58.50	59.75	+1.25
Cube	Short†	76.50	78.25	+1.75
Four-task mean	Long	43.75	47.19	+3.44
Four-task mean	Short	81.13	82.56	+1.44

Dependence on evaluation horizon. The aggregate gain is 3.44 percentage points for long horizons and 1.44 points for short horizons. At the task level, the long-horizon gain exceeds the short-horizon gain for TwoRoom, PushT, and Reacher, whereas Cube shows the reverse ordering (1.25 versus 1.75 points). The temporal recipe therefore has a larger average effect in the more difficult long-horizon setting.

5 Discussion and Limitations

Distinct roles of the regularizers. The geometry term retains distributional pressure against collapse while introducing directional target variances. The temporal term distinguishes small propagated contributions from large ones that cancel across groups. These roles explain why the terms are not redundant, as formalized in Appendix P. Table 1 reports the combined method, while the component studies examine the regularizers separately. Their different protocols prevent interpreting the differences between these studies as matched removal ablations.

Performance across evaluation settings. The combined method and Ellipsoid reach mean success rates of 85.50% and 84.17%, respectively, compared with 79.25% for our LeWM reproduction (Tables 1 and 2). The combined method exceeds Ellipsoid by 1.33 percentage points on average; the task-level differences are +1.67 on TwoRoom, -1.67 on Reacher, +2.00 on PushT, and +3.33 on Cube. These are descriptive comparisons across the reported protocols. In the separate frozen-geometry setting, HGap adaptation raises mean success by 3.44 points at long horizons and 1.44 points at short horizons. These evaluations characterize the combined method, directional geometry regularization, and temporal adaptation under their respective protocols.

What the gap measures. HGap describes cancellation among generated error contributions. The bound does not certify counterfactual predictions for planner-selected actions, task sufficiency of

the representation, or closed-loop success. Cancellation may reflect useful self-correction, so its reduction is an inductive bias whose value must be measured. Likewise, a fixed positive-definite metric amounts to a change of coordinates; using inverse representation covariance does not by itself supply an optimal planning metric or a new robustness guarantee.

Strength of the empirical evidence. Evaluation averages for a fixed trained model characterize evaluation variability rather than variation across independently trained models. HGap adaptation shares pretrained checkpoints across seeds and partially reuses evaluation cohorts. We interpret the combined scores in Table 1 as descriptive success rates; the aggregate tables do not quantify training variability or establish statistical significance. The component studies use distinct protocols and do not isolate component interactions through matched removal experiments. The HGap comparison evaluates the complete adaptation recipe and does not separate the effects of the gap penalty, multi-start supervision, and additional optimization.

6 Conclusion

We combine two regularization targets for latent world models: the shape of the embedding distribution and the composition of multi-step prediction errors. Anchored ellipsoidal SIGReg adjusts directional target variances, while HGap exposes cross-group cancellation through exact hybrid-trajectory differences. The combined objective achieves a mean success rate of 85.50% across four visual control tasks, improving on our LeWM reproduction by 6.25 percentage points. Separate component studies show a 4.92-percentage-point mean gain from anchored geometry and gains of 3.44 and 1.44 points for long- and short-horizon frozen-geometry HGap adaptation, respectively. The formulation integrates distributional geometry and temporal error composition into a common prediction objective.

AI Use Statement

Generative AI assisted with organization of research records and literature, interpretation of experimental results, conceptual development and experimental comparisons, manuscript drafting, mathematical exposition and proof writing, and LaTeX preparation. It was not used to execute model training or evaluation during manuscript preparation.

Reproducibility Statement

The mathematical assumptions and proof of the hierarchical bound are given in Appendix A, with the explicit hybrid-trajectory construction in Appendix H. Appendix B explains result aggregation, Appendix C describes the evaluation protocols, and Appendix R specifies the joint training settings and regularization coefficients. Equation (16) specifies the frozen-geometry adaptation loss, including its weights and fixed calibration procedure.

References

Robert Jenkinson Alvarez. Beyond Isotropy in JEPAs: Hamiltonian Geometry and Symplectic Prediction. *arXiv preprint arXiv:2605.20107*, 2026. URL https://arxiv.org/abs/2605.20107.

Kavosh Asadi, Dipendra Misra, and Michael L. Littman. Lipschitz Continuity in Model-based Reinforcement Learning. *arXiv preprint arXiv:1804.07193*, 2018. URL https://arxiv.org/abs/1804.07193.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture. *arXiv preprint arXiv:2301.08243*, 2023. URL https://arxiv.org/abs/2301.08243.

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning. *arXiv preprint arXiv:2506.09985*, 2025. URL https://arxiv.org/abs/2506.09985.

Randall Balestriero and Yann LeCun. LeJEPA: Provable and Scalable Self-Supervised Learning Without the Heuristics. *arXiv preprint arXiv:2511.08544*, 2025. URL https://arxiv.org/abs/2511.08544.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning. *arXiv preprint arXiv:2105.04906*, 2021. URL https://arxiv.org/abs/2105.04906.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting Feature Prediction for Learning Visual Representations from Video. *arXiv preprint arXiv:2404.08471*, 2024. URL https://arxiv.org/abs/2404.08471.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks. *arXiv preprint arXiv:1506.03099*, 2015. URL https://arxiv.org/abs/1506.03099.

Kavya Bhand and Aadi Joshi. Multi-Horizon Consistency as Geometry: When Latent Dynamics Contract, and When They Do Not. *arXiv preprint arXiv:2607.21645*, 2026. URL https://arxiv.org/abs/2607.21645.

Jack Boylan and Chris Hokamp. No Gaussian Required: Contrastive Inverse Dynamics for JEPA World Models. *arXiv preprint arXiv:2608.17542*, 2026. URL https://arxiv.org/abs/2608.17542.

Xinlei Chen and Kaiming He. Exploring Simple Siamese Representation Learning. *arXiv preprint arXiv:2011.10566*, 2020. URL https://arxiv.org/abs/2011.10566.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised Learning. *arXiv preprint arXiv:2006.07733*, 2020. URL https://arxiv.org/abs/2006.07733.

David Ha and Jürgen Schmidhuber. World Models. *arXiv preprint arXiv:1803.10122*, 2018. URL https://arxiv.org/abs/1803.10122.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to Control: Learning Behaviors by Latent Imagination. *arXiv preprint arXiv:1912.01603*, 2019a. URL https://arxiv.org/abs/1912.01603.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning Latent Dynamics for Planning from Pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565, 2019b. URL https://proceedings.mlr.press/v97/hafner19a.html.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, Robust World Models for Continuous Control. *arXiv preprint arXiv:2310.16828*, 2023. URL https://arxiv.org/abs/2310.16828.

Alex Lamb, Anirudh Goyal, Ying Zhang, Saizheng Zhang, Aaron Courville, and Yoshua Bengio. Professor Forcing: A New Algorithm for Training Recurrent Networks. *arXiv preprint arXiv:1610.09038*, 2016. URL https://arxiv.org/abs/1610.09038.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorld-Model: Stable End-to-End Joint-Embedding Predictive Architecture from Pixels. *arXiv preprint arXiv:2603.19312*, 2026. URL https://arxiv.org/abs/2603.19312.

Vlad Sobal, Wancong Zhang, Kyunghyun Cho, Randall Balestriero, Tim G. J. Rudner, and Yann LeCun. Learning from Reward-Free Offline Data: A Case for Planning with Latent Dynamics Models. *arXiv preprint arXiv:2502.14819*, 2025. URL https://arxiv.org/abs/2502.14819.

Erik Talvitie. Self-Correcting Models for Model-Based Reinforcement Learning. *arXiv preprint arXiv:1612.06018*, 2016. URL https://arxiv.org/abs/1612.06018.

Ying Wang, Oumayma Bounou, Gaoyue Zhou, Randall Balestriero, Tim G. J. Rudner, Yann LeCun, and Mengye Ren. Temporal Straightening for Latent Planning. *arXiv preprint arXiv:2603.12231*, 2026. URL https://arxiv.org/abs/2603.12231.

Haiyu Wu, Randall Balestriero, and Morgan Levine. VISReg: Variance-Invariance-Sketching Regularization for JEPA training. *arXiv preprint arXiv:2606.02572*, 2026. URL https://arxiv.org/abs/2606.02572.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow Twins: Self-Supervised Learning via Redundancy Reduction. *arXiv preprint arXiv:2103.03230*, 2021. URL https://arxiv.org/abs/2103.03230.

Kai Zhao, Dongliang Nie, Yuchen Lin, Zhehan Luo, Yixiao Gu, Deng-Ping Fan, and Dan Zeng. Sub-JEPA: Subspace Gaussian Regularization for Stable End-to-End World Models. *arXiv preprint arXiv:2605.09241*, 2026. URL https://arxiv.org/abs/2605.09241.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World Models on Pre-trained Visual Features enable Zero-shot Planning. *arXiv preprint arXiv:2411.04983*, 2024. URL https://arxiv.org/abs/2411.04983.

A Proof and Interpretation of the Hierarchical Bound

By Equation (9), $e_t = \sum_{B \in \mathcal{P}_t} S_{B,t}$. Applying the triangle inequality in the norm induced by $Q \succ 0$ gives $\|e_t\|_Q \le \sum_B \|S_{B,t}\|_Q$. Squaring and dividing by d proves $E_t^Q \le U_{\mathcal{P},t}^Q$. Each coarse block is the union of fine blocks; applying the same inequality inside every coarse block gives the refinement ordering. Expanding the squared norm of $\sum_B S_{B,t}$ and subtracting it from the squared sum of block norms yields Equation (12). Each summand is nonnegative by Cauchy–Schwarz.

For nonzero blocks, each summand equals $\|S_B\|_Q \|S_C\|_Q (1 - \cos\theta_{BC}^Q)$. Thus the gap vanishes when all nonzero block sums lie along a common nonnegative ray. Conversely, opposed block sums can have zero total error and a positive gap. For example, with two blocks $S_1 = v$ and $S_2 = -v$, $E = 0$ and $U = 4\|v\|_Q^2/d$. This example explains why reducing the gap is not equivalent to reducing the observed rollout error.

The finest bound can be interpreted as worst-case alignment of contributions under abstract norm-preserving transformations in whitened coordinates. Such transformations are not necessarily realizable environment perturbations. No physical robustness guarantee follows without an explicit uncertainty model connecting these transformations to the environment.

B Averaging and Interpretation of Reported Results

The combined-method and ellipsoidal results average six evaluation seeds for a single trained model per task. HGap adaptation results average two continuation seeds sharing a pretrained checkpoint. The former summarize evaluation variability, whereas the latter also reflect variation between adaptation runs from a common initialization.

Let x_{mks} be the success rate in percentage points for method m, task k, and evaluation or adaptation seed s. For S_{mk} seeds and K tasks, we report

$$\bar{x}_{mk} = \frac{1}{S_{mk}} \sum_{s=1}^{S_{mk}} x_{mks}, \qquad \bar{x}_m = \frac{1}{K} \sum_{k=1}^{K} \bar{x}_{mk}. \tag{18}$$

Task means receive equal weight. For Tables 1 and 2, we compute differences from the reported task rates. For Table 3, we average the reported task-level rates before rounding the displayed macro-averages and their differences. For example, the long-horizon HGap average is 47.1875%, while its baseline is 43.75%, giving a 3.4375-point difference. The short-horizon values are 82.5625% and 81.125%, respectively, giving 1.4375 points. Table 3 rounds these quantities to two decimal places.

For adaptation, let c_{ks} denote the number of successful episodes among $n = 200$ starts. The reported rate is

$$\bar{x}_k = \frac{100}{S} \sum_{s=1}^{S} \frac{c_{ks}}{n}, \qquad S = 2. \tag{19}$$

Equal denominators make this arithmetically identical to dividing the sum of counts by Sn. Nevertheless, repeated use of the same starting states and pretrained checkpoint prevents interpreting Sn as the number of independent environmental or training replications. The experimental unit is unchanged by this arithmetic equivalence.

We report mean success rates without standard deviations, confidence intervals, or significance tests. Appendix Q describes the experimental units and interpretation of these averages.

C Evaluation Protocols and Attribution

The ellipsoidal and HGap experiments use their respective baselines. Table 3 evaluates both the base checkpoint and the adapted predictors on 200 starts per task and horizon. Long-horizon evaluation uses goal offset 75 for PushT and 100 for the other tasks, with execution budget 150. Short-horizon evaluation uses goal offset 25 and execution budget 50. PushT uses 30 CEM iterations and the other tasks use 10.

HGap continuation starts from an existing trained model, freezes the encoder, projector, and BatchNorm buffers, and optimizes dynamics-related parameters. Its two adaptation seeds share the initial checkpoint. These continuation results therefore do not measure changes to the learned visual representation.

Frozen-geometry adaptation settings. The adaptation objective is $L_{\mathrm{TF}} + 0.20A/c_A + 0.05G_2^I/c_G$, which is algebraically identical to Equation (16). Training clips contain eight observations ($H = 7$), giving 28 start–target pairs for A. The gap uses contiguous pairs of intervention differences, the Euclidean metric $Q = I$, and temporal weights $q_t = t/28$. Both positive calibration scales are computed on a fixed set of 512 training clips and held constant during adaptation. The encoder, projector, and BatchNorm buffers are frozen, and dropout is disabled. The two continuation seeds share a pretrained checkpoint; each task–horizon setting uses 200 evaluation starts per seed. These settings describe the frozen-geometry component experiment, not the jointly trained model.

D Notation and Mathematical Scope

This supplement separates exact identities from additional modeling assumptions. The hierarchical inequalities concern a finite set of predictions produced by a fixed model on a fixed observation–action sequence. They do not require that these predictions be correct. Distributional interpretations of the ellipsoidal objective instead concern a population of embeddings and, where stated, a fixed target distribution. Derivatives are conditional on the quantities explicitly held fixed. These distinctions matter when the encoder, covariance estimate, and predictor are optimized together.

D.1 Notation

We use H for the maximum prediction horizon and d for the embedding dimension. A training clip contains $H + 1$ observations. The symbols N, M, and L respectively denote the number of embeddings used by a distributional estimate, the number of sampled directions, and the number of frequency nodes. The analysis is stated for arbitrary sample sizes, projection counts, and frequency grids.

Table 4: Notation for the mathematical supplement.

Symbol	Meaning
g_ϕ, f_θ	Visual representation and action-conditioned predictor
$z_t, \widehat{z}_t^{[s]}$	Encoded observation and prediction rolled out from history ending at s
$Y_t^{(j)}$	Terminal prediction after j observed-history substitutions
$\Delta_{j,t}$	Difference $Y_t^{(j)} - Y_t^{(j+1)}$
$\mathcal{P}_t, S_{B,t}$	Partition of intervention indices and sum in block B
$E_t^Q, U_{\mathcal{P},t}^Q$	Actual error energy and grouped norm bound
$G_{\mathcal{P},t}^Q$	Gap $U_{\mathcal{P},t}^Q - E_t^Q$
$C, P_k, \Sigma_\star$	Empirical covariance, principal projector, target covariance
Q	Common positive-definite error metric
$\alpha, \lambda, \beta, \eta$	Target interpolation and loss coefficients

D.2 Finite-sample and population statements

Let Z denote a random embedding and $z_1, \ldots, z_N$ a finite collection. Its population characteristic function and empirical estimate are

$$\varphi_Z(v) = \mathbb{E}[\exp(\mathrm{i}v^\top Z)], \qquad \widehat{\varphi}_Z(v) = \frac{1}{N}\sum_{n=1}^{N} \exp(\mathrm{i}v^\top z_n). \tag{20}$$

Equality of population characteristic functions for every $v \in \mathbb{R}^d$ identifies a distribution. Matching finitely many (u, ω) pairs, with $v = \omega u$, imposes only finitely many constraints. Accordingly,

statements about an ideal target below explain the objective; they do not assert that a trained finite-dimensional network realizes that distribution exactly.

For the temporal analysis, fix t and abbreviate $S_b = S_{B_b,t}$ for $b = 1, \dots, m$. Define $r_b = \|S_b\|_Q$, $R = \sum_b r_b$, and $e = \sum_b S_b$. Then

$$E = \frac{\|e\|_Q^2}{d}, \qquad U = \frac{R^2}{d}, \qquad G = U - E. \tag{21}$$

The same Q is used for every block and for the total. It may differ between clips or target times without affecting a pointwise inequality, provided it remains positive definite and common within each comparison. A partition may also depend on the clip; comparing two bounds still requires nested partitions of the same contributions.

D.3 What is being held fixed

The distributional analysis distinguishes a target selected before an update from one differentiated through the current embeddings. The temporal analysis distinguishes a fixed metric from a trainable metric. Neither distinction changes the forward algebraic identities. Both change the optimization dynamics, because a parameter can influence the target or metric as well as the predicted embeddings. These dependencies are written explicitly in the derivative sections instead of being absorbed into the notation.

E Ellipsoidal Targets: Spectrum, Scale, and Directional Matching

E.1 Empirical geometry

For a batch of N embeddings, define

$$\bar{z} = \frac{1}{N}\sum_{n=1}^{N} z_n, \qquad C = \frac{1}{N}\sum_{n=1}^{N}(z_n - \bar{z})(z_n - \bar{z})^\top. \tag{22}$$

The denominator specifies a convention for this derivation. Replacing N by $N - 1$ rescales C but does not change its eigenvectors or the dimension statistic below. Write $C = V\,\mathrm{diag}(\lambda_1, \dots, \lambda_d)V^\top$, with $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$. For $C \neq 0$,

$$d_{\mathrm{eff}} = \frac{\mathrm{tr}(C)}{\|C\|_{\mathrm{op}}} = \frac{\sum_i \lambda_i}{\lambda_1}, \qquad 1 \leq d_{\mathrm{eff}} \leq \mathrm{rank}(C) \leq d. \tag{23}$$

The first bound follows from retaining λ_1 in the sum, and the second from $\lambda_i \leq \lambda_1$ for each nonzero eigenvalue. Equal nonzero eigenvalues attain the upper bound. The statistic is unchanged by a global rescaling $C \mapsto cC$ for $c > 0$. The statistic is undefined at $C = 0$; the spectral identities below assume $C \neq 0$.

Let $P = P_k$ be an orthogonal projector: $P^2 = P = P^\top$ and $\mathrm{tr}(P) = k$. Put $a = \sigma_\parallel^2$ and $b = \sigma_\perp^2$. Then

$$\Sigma_\star = aP + b(I - P), \qquad \mathrm{tr}(\Sigma_\star) = ka + (d - k)b. \tag{24}$$

For $0 < k < d$ and $a, b > 0$, the eigenvalues are a with multiplicity k and b with multiplicity $d - k$. Consequently,

$$\Sigma_\star^{-1} = a^{-1}P + b^{-1}(I - P), \qquad \kappa(\Sigma_\star) = \frac{\max(a, b)}{\min(a, b)}. \tag{25}$$

This gives a direct separation between total target variance and anisotropy. A change in both quantities cannot be attributed solely to directional structure.

E.2 Directional variance

For a unit vector u, orthogonality gives

$$u^\top \Sigma_\star u = a\,\|Pu\|_2^2 + b\,\|(I - P)u\|_2^2 = b + (a - b)\,\|Pu\|_2^2\,. \tag{26}$$

Thus the target variance lies between $\min(a, b)$ and $\max(a, b)$. If directions are uniform on the unit sphere, symmetry implies $\mathbb{E}[uu^\top] = I/d$, and hence

$$\mathbb{E}_u[u^\top \Sigma_\star u] = \frac{\mathrm{tr}(\Sigma_\star)}{d}, \qquad \mathbb{E}_u[\|Pu\|_2^2] = \frac{k}{d}. \tag{27}$$

Targets with the same trace have the same directional average but can distribute variance differently across directions. This explains why controlling only the average projection variance does not determine anisotropy. Conversely, selecting directions non-uniformly changes which parts of the covariance are emphasized by a finite estimator.

E.3 Covariance estimation and mean control

Centering in Equation (22) makes the principal subspace insensitive to a global translation of embeddings. The characteristic-function discrepancy need not be translation-invariant. If $Z' = Z + c$, then

$$\varphi_{Z'}(\omega u) = \exp(\mathrm{i}\omega u^\top c)\varphi_Z(\omega u). \tag{28}$$

Matching to a zero-mean Gaussian penalizes this phase change. Centering the inputs to the discrepancy would remove the same mean-control mechanism. Estimating geometry from centered embeddings and applying the target discrepancy to raw embeddings therefore serve different purposes. The former measures spread, while the latter also responds to location.

F Distributional Interpretation of the Isotropic Anchor

F.1 Completion of the square in characteristic-function space

Assume both discrepancies use the same nonnegative measure μ over projection–frequency pairs, and hold $\Sigma_\star$ fixed. For complex numbers a, b, c and $\alpha \in [0, 1]$, expansion of the two squared moduli gives

$$\begin{aligned}&(1-\alpha)|a-b|^2 + \alpha|a-c|^2 \\ &\quad = |a|^2 - 2\,\mathrm{Re}\Big(a\overline{(1-\alpha)b + \alpha c}\Big) + (1-\alpha)|b|^2 + \alpha|c|^2.\end{aligned} \tag{29}$$

The identity

$$(1-\alpha)|b|^2 + \alpha|c|^2 - |(1-\alpha)b + \alpha c|^2 = \alpha(1-\alpha)|b-c|^2 \tag{30}$$

then proves Equation (6). Integrating pointwise with $a = \varphi_Z$, $b = \varphi_I$, and $c = \varphi_{\Sigma_\star}$ yields

$$R_{\mathrm{anchor}} = \int |\varphi_Z - \varphi_{\mathrm{mix}}|^2 \, d\mu + C_\alpha, \quad \varphi_{\mathrm{mix}} = (1-\alpha)\varphi_I + \alpha\varphi_{\Sigma_\star}, \tag{31}$$

where $C_\alpha = \alpha(1-\alpha)\int |\varphi_I - \varphi_{\Sigma_\star}|^2 d\mu$. The same calculation applies to an empirical characteristic function. Under the fixed-target assumption, C_α is independent of the optimized embeddings, though it is generally positive.

Characteristic functions are linear under mixtures. Therefore φ_{mix} is the characteristic function of

$$P_{\mathrm{mix}} = (1-\alpha)\mathcal{N}(0, I) + \alpha\mathcal{N}(0, \Sigma_\star). \tag{32}$$

An ideal unrestricted population matcher would target this mixture. A finite empirical matcher with limited directions and frequencies can have additional minimizers. Finite matching therefore does not uniquely identify the population distribution.

F.2 Why covariance interpolation is insufficient

Both components have zero mean, so the mixture covariance is

$$\mathrm{Cov}(P_{\mathrm{mix}}) = (1-\alpha)I + \alpha\Sigma_\star. \tag{33}$$

Nevertheless, the mixture usually differs from a Gaussian with this covariance. For a fixed unit direction u, let $v = u^\top \Sigma_\star u$. The mixture has second moment $(1-\alpha) + \alpha v$ and fourth moment $3[(1-\alpha) + \alpha v^2]$. The difference from the fourth moment of the covariance-matched Gaussian is

$$3[(1-\alpha) + \alpha v^2] - 3[(1-\alpha) + \alpha v]^2 = 3\alpha(1-\alpha)(v-1)^2. \tag{34}$$

For $0 < \alpha < 1$ and $v \neq 1$, this difference is strictly positive. Matching the covariance therefore does not reproduce the characteristic-function target. The anchored population target consequently retains higher-order structure beyond covariance interpolation.

F.3 What the anchor controls

Because each discrepancy is nonnegative, for $\alpha < 1$,

$$R_{\mathrm{iso}} \le \frac{R_{\mathrm{anchor}}}{1-\alpha}. \tag{35}$$

This is a direct bound on the measured isotropic discrepancy. It is not a lower bound on every empirical covariance eigenvalue, because a finite set of projections and frequencies need not detect every degeneracy. At $\alpha = 0$, the objective is exactly the isotropic discrepancy; at $\alpha = 1$, Equation (35) no longer gives control of it. Intermediate values retain a quantitative penalty against departures visible to the isotropic estimator.

F.4 Data-dependent targets

If the target is recomputed as $\Sigma_\star(Z)$, Equation (31) remains a pointwise identity but C_α also depends on Z. Differentiating through the target gives

$$\frac{dR_{\mathrm{ell}}}{dZ} = \left.\frac{\partial R_{\mathrm{ell}}}{\partial Z}\right|_{\Sigma_\star} + \frac{\partial R_{\mathrm{ell}}}{\partial \Sigma_\star}\frac{\partial \Sigma_\star}{\partial Z}. \tag{36}$$

Stopping gradients through the target removes the second term for that update. It does not make the target constant across successive updates. Principal eigenvectors can also become non-unique at eigenvalue ties, and changing a discrete rank k introduces another source of non-smoothness. The forward covariance formula alone does not determine these gradient contributions.

G Finite Projection Estimators and Their Gradients

G.1 A real-valued implementation

Consider unit directions $u_1, \dots, u_M$, frequency nodes $\omega_1, \dots, \omega_L$, and nonnegative weights w_ℓ summing to one. This defines a generic finite estimator compatible with Equation (3); the analysis treats the numbers of directions and frequency nodes as arbitrary finite parameters. Write

$$c_{m\ell} = \frac{1}{N}\sum_{n=1}^{N} \cos(\omega_\ell u_m^\top z_n), \qquad s_{m\ell} = \frac{1}{N}\sum_{n=1}^{N} \sin(\omega_\ell u_m^\top z_n), \tag{37}$$

$$t_{m\ell} = \exp(-\tfrac{1}{2}\omega_\ell^2 u_m^\top \Sigma_\star u_m). \tag{38}$$

The Gaussian target is real, so complex arithmetic is unnecessary:

$$\widehat{R}_{\mathrm{ell}} = \frac{1}{M}\sum_{m=1}^{M}\sum_{\ell=1}^{L} w_\ell[(c_{m\ell} - t_{m\ell})^2 + s_{m\ell}^2]. \tag{39}$$

The real component responds to a combination of variance and higher even moments, while the imaginary component responds to odd asymmetries, including a nonzero mean. The two components are evaluated together; interpreting the loss as only a covariance penalty discards information it actually measures.

G.2 Embedding gradients for a fixed target

Holding directions, nodes, and targets fixed, elementary differentiation gives

$$\nabla_{z_n}\widehat{R}_{\mathrm{ell}} = \frac{2}{MN}\sum_{m,\ell} w_\ell \omega_\ell u_m \, h_{nm\ell},$$

$$h_{nm\ell} = -(c_{m\ell} - t_{m\ell})\sin(\omega_\ell u_m^\top z_n) + s_{m\ell}\cos(\omega_\ell u_m^\top z_n). \tag{40}$$

Each embedding affects the empirical mean for every projection, coupling gradients across samples. Even when the target is fixed, the loss is not a sum of independent per-example distances to fixed

target samples. If the target is differentiated, Equation (36) adds its dependence on all embeddings through the estimated covariance.

For a Gaussian target, its differential with respect to a symmetric perturbation $d\Sigma$ is

$$dt_{m\ell} = -\tfrac{1}{2}\omega_\ell^2 t_{m\ell}\,\mathrm{tr}(u_m u_m^\top d\Sigma). \tag{41}$$

This expression makes the frequency dependence explicit: higher frequencies can induce stronger sensitivity to covariance changes, although the exponential target also decreases with frequency. The actual sensitivity depends on the entire quadrature and feature distribution.

G.3 Low-frequency interpretation

Suppose the projected variable has sufficiently many finite moments. Near $\omega = 0$,

$$\varphi_{u^\top Z}(\omega) = 1 + \mathrm{i}\omega\mathbb{E}[u^\top Z] - \tfrac{1}{2}\omega^2\mathbb{E}[(u^\top Z)^2] + O(|\omega|^3). \tag{42}$$

For a centered distribution, the leading real discrepancy from the Gaussian target is $-\frac{1}{2}\omega^2(u^\top \mathrm{Cov}(Z)u - u^\top\Sigma_\star u)$. Squaring gives a leading covariance-mismatch term of order ω^4. A nonzero projected mean instead produces an imaginary discrepancy at order ω. Frequencies away from zero probe information not captured by these leading moments. This expansion explains the relation to moment control without equating the full objective to a second-order penalty.

G.4 Finite-sample bias

For independent samples and a fixed target t, let $X_n = \exp(\mathrm{i}\omega u^\top Z_n)$, with $\mathbb{E}X_n = \varphi$. Since $|X_n| = 1$,

$$\mathbb{E}|\widehat{\varphi} - t|^2 = |\varphi - t|^2 + \frac{1 - |\varphi|^2}{N}. \tag{43}$$

To verify this, write $\widehat{\varphi} - t = (\varphi - t) + (\widehat{\varphi} - \varphi)$, expand the squared modulus, and use the zero expectation of the centered term. Independence makes its variance $(1 - |\varphi|^2)/N$. The bias term depends on the representation distribution. Thus minimizing the expected finite-batch discrepancy is not identical to minimizing the population discrepancy.

If embeddings come from correlated positions in a clip, the variance instead contains

$$\mathbb{E}|\widehat{\varphi} - \varphi|^2 = \frac{1}{N^2}\sum_{n,r}\mathbb{E}[(X_n - \varphi)\overline{(X_r - \varphi)}], \tag{44}$$

assuming a common marginal mean φ. Covariance estimation from the same batch further couples the target to this estimator. These observations explain why the geometry loss depends on batch construction. They do not imply that finite-sample bias explains the observed task differences.

H Hybrid Trajectories as Exact Nonlinear Interventions

H.1 Explicit construction for Markov dynamics

For clarity, first consider a deterministic predictor $F_t(z) = f_\theta(z, a_t)$ and observed embeddings $z_0, \ldots, z_H$. Define the downstream composition

$$\Phi_{t:j} = F_{t-1} \circ F_{t-2} \circ \cdots \circ F_j, \qquad \Phi_{t:t} = \mathrm{Id}\,. \tag{45}$$

A concrete hybrid family is $Y_t^{(j)} = \Phi_{t:j}(z_j)$. It follows that

$$Y_t^{(0)} = \Phi_{t:0}(z_0), \qquad Y_t^{(t)} = z_t, \tag{46}$$

$$\Delta_{j,t} = \Phi_{t:j+1}(F_j(z_j)) - \Phi_{t:j+1}(z_{j+1}). \tag{47}$$

The local teacher-forced residual $\delta_j = F_j(z_j) - z_{j+1}$ is injected at time $j+1$ and propagated by the same downstream nonlinear map. The terminal difference is an exact finite difference of that map, not a product of Jacobians evaluated on a reference trajectory.

Summing adjacent differences cancels all interior hybrids:

$$\sum_{j=0}^{t-1} \Delta_{j,t} = (Y_t^{(0)} - Y_t^{(1)}) + (Y_t^{(1)} - Y_t^{(2)}) + \cdots + (Y_t^{(t-1)} - Y_t^{(t)}) \\ = Y_t^{(0)} - z_t. \tag{48}$$

This algebra remains valid for any generated arrays with the stated endpoints. The intervention interpretation is stronger: it requires the branches to differ in the intended history substitution, with all other prediction rules held consistent.

H.2 History-conditioned predictors

For a predictor using a finite context, the state of the rollout includes a history buffer. One can write an augmented transition $\widetilde{F}_t$ that appends the predicted embedding and discards entries outside the context window. A hybrid branch initializes the buffer with the appropriate observed prefix and then applies this transition. Consecutive branches may differ in several internal buffer entries after their start times, even though their initialization differs by one additional observed prefix element.

The telescoping identity does not require the augmented state to have dimension d: only the terminal embeddings being subtracted must lie in the common output space. However, identifying $\Delta_{j,t}$ with a propagated one-step residual requires a precisely specified buffer and reset convention. This is why the simple Markov construction is an explanatory special case, not a substitute for the architecture's history rules.

H.3 Connection to differential propagation

Assume the downstream map in Equation (47) is continuously differentiable on the line segment between its two arguments. Applying the fundamental theorem of calculus along that segment gives

$$\Delta_{j,t} = \left[\int_0^1 J\Phi_{t:j+1}(z_{j+1} + s\delta_j)\, ds \right] \delta_j. \tag{49}$$

The bracketed matrix is an integrated Jacobian along a segment. A fixed-Jacobian approximation replaces it by a matrix evaluated at one point; HGap instead computes the two terminal predictions directly. Equation (49) clarifies what is retained by the finite difference: the downstream response over the whole displacement, including changes in local derivatives.

If each F_r is L_r-Lipschitz in a common norm, repeated application yields the conditional estimate

$$\|\Delta_{j,t}\| \le \left(\prod_{r=j+1}^{t-1} L_r \right) \|\delta_j\|. \tag{50}$$

The product for $j = t-1$ is one. Summing these estimates recovers a conventional propagation upper bound. HGap does not need known L_r values because it measures the branch differences, but the absence of such constants also means it provides no uniform bound over unobserved trajectories.

H.4 Stochastic branches

If the predictor includes randomness, let $Y_t^{(j)}(\xi)$ denote a branch with specified random inputs. A family generated with any chosen coupling still satisfies telescoping pathwise. With independent dropout masks, however, $\Delta_{j,t}$ can include mask-induced variation in addition to the effect of the history substitution. Sharing random inputs across comparable operations or disabling stochastic layers makes the interpretation more direct. The reported frozen-geometry adaptation disables dropout. This evaluates a deterministic predictor rather than an expectation over stochastic predictions.

I Hierarchy, Equality Conditions, and Refinement

I.1 Proof in whitened coordinates

Let $Q \succ 0$ and choose its symmetric square root. Setting $x_b = Q^{1/2} S_b$ gives $\|S_b\|_Q = \|x_b\|_2$ and $Q^{1/2} e = \sum_b x_b$. The triangle inequality therefore yields

$$dE = \left\| \sum_b x_b \right\|_2^2 \le \left(\sum_b \|x_b\|_2 \right)^2 = dU. \tag{51}$$

Suppose a coarse block B is partitioned into fine blocks $C_1, \dots, C_r$. Since $S_B = \sum_i S_{C_i}$,

$$\sum_{B \in \mathcal{P}_c} \|S_B\|_Q \le \sum_{B \in \mathcal{P}_c} \sum_{C \subseteq B,\ C \in \mathcal{P}_f} \|S_C\|_Q = \sum_{C \in \mathcal{P}_f} \|S_C\|_Q \,. \tag{52}$$

Both sides are nonnegative; squaring proves the refinement ordering. Because the same E is subtracted at every level, the gaps inherit the same ordering. This proof applies to irregular partitions as well as equal-size contiguous groups. Contiguity supplies a temporal interpretation but is not needed for the inequality.

I.2 Exact change induced by one split

Consider splitting a coarse block $S = a + b$. Let T be the sum of norms of all other blocks. The coarse and fine radii are $T + \|a + b\|_Q$ and $T + \|a\|_Q + \|b\|_Q$. Their squared difference gives

$$\begin{aligned} d(U_f - U_c) = &\big(\|a\|_Q + \|b\|_Q - \|a + b\|_Q \big) \\ &\cdot \big(2T + \|a\|_Q + \|b\|_Q + \|a + b\|_Q \big). \end{aligned} \tag{53}$$

The first factor is the triangle-inequality slack inside the split block. The second is nonnegative and depends on both local magnitudes and the remaining groups. Refinement therefore exposes cancellation previously hidden inside a block, with an effect that also depends on the rest of the trajectory. It does not introduce a new prediction or alter the total error.

If a and b are nonzero, the first factor vanishes precisely when their whitened vectors lie on the same nonnegative ray. If either is zero, splitting also leaves the bound unchanged. Applying this argument repeatedly identifies equality conditions for any nested refinement: every coarse block must have no triangle slack among its constituent fine blocks.

I.3 Nested partitions and monotonicity

Comparing group sizes alone does not establish an ordering. For indices $0, \dots, 5$, pair blocks $\{0,1\}, \{2,3\}, \{4,5\}$ and triple blocks $\{0,1,2\}, \{3,4,5\}$ are not refinements of one another. Either grouped radius can be larger depending on the contributions. For example, in one dimension with contributions $(1, 1, -2, 0, 0, 0)$, the pair radius is 4 while the triple radius is 0. With contributions $(0, 0, 1, -1, 0, 0)$, the pair radius is 0 while the triple radius is 2.

These examples have the same total error, zero, yet reverse the bound ordering. Monotonicity holds for explicitly nested partitions, such as successive pairwise merges; increasing the nominal block length alone is insufficient. The HGap-2 objective itself is well-defined without a full hierarchy.

I.4 Equality of the bound and actual error

Expanding in whitened coordinates gives

$$dG = 2 \sum_{b<c} \left(\|x_b\|_2 \, \|x_c\|_2 - x_b^\top x_c \right). \tag{54}$$

Every summand is nonnegative. Consequently, $G = 0$ if and only if every pair of nonzero block vectors is positively collinear. Equivalently, there is a vector v and nonnegative coefficients a_b such that $S_b = a_b v$ for all nonzero blocks. The case of all-zero blocks also has $G = 0$.

A zero gap is therefore compatible with a large prediction error: choose any large v and align all contributions with it. Conversely, zero prediction error is compatible with a large gap, as shown by $S_1 = v, S_2 = -v$. Neither loss alone determines the other. Keeping prediction accuracy in the objective is essential to the intended interpretation of the joint loss.

J Angular Interpretation and Scale Dependence of HGap

J.1 Magnitude-weighted angular dispersion

For nonzero blocks let $n_b = Q^{1/2} S_b / r_b$, so $\|n_b\|_2 = 1$. With $R > 0$, set $p_b = r_b / R$ and omit zero blocks. Then $p_b \geq 0$ and $\sum_b p_b = 1$. The normalized error and gap satisfy

$$\frac{E}{U} = \left\| \sum_b p_b n_b \right\|_2^2, \qquad \frac{G}{U} = 1 - \left\| \sum_b p_b n_b \right\|_2^2. \tag{55}$$

The norm of a convex combination of unit vectors is at most one, so $0 \leq G/U \leq 1$. The ratio measures dispersion of the whitened directions with weights proportional to contribution magnitude. It is not an unweighted average of angles: a tiny block contributes little even if it points in the opposite direction.

Using $\|n_b - n_c\|_2^2 = 2(1 - n_b^\top n_c)$, the same expression can be written as

$$\frac{G}{U} = \frac{1}{2} \sum_{b,c} p_b p_c \, \|n_b - n_c\|_2^2 . \tag{56}$$

To prove this, expand the squared distances, sum the two unit-norm terms to two, and collect the cross term as $2 \left\| \sum_b p_b n_b \right\|_2^2$. This representation links the cancellation gap to the weighted dispersion of directions. It does not introduce a new training term; the unnormalized G remains the objective used in the stated formulation.

J.2 Magnitude and angle cannot be separated by the raw gap

If every block is rescaled by a common c, then

$$E(cS_1, \ldots, cS_m) = c^2 E, \quad U(cS_1, \ldots, cS_m) = c^2 U, \quad G(cS_1, \ldots, cS_m) = c^2 G. \tag{57}$$

For $c \neq 0$ and $U > 0$, the ratio G/U is invariant. Thus a reduction in raw gap can come from smaller contributions, better directional alignment, or both. Examining E, U, and G together shows whether a decrease in the raw gap is accompanied by a decrease in the total contribution magnitude. The representation regularizer also matters here: shrinking embeddings can reduce Euclidean temporal losses without necessarily improving the information available for control.

For m blocks, Cauchy–Schwarz applied to their nonnegative norms gives

$$U \leq \frac{m}{d} \sum_{b=1}^{m} \|S_b\|_Q^2 , \qquad 0 \leq G \leq U. \tag{58}$$

This separates the sum-of-norms bound from a sum of per-block energies. They are related but different objectives. The former contains all pairwise magnitude products; the latter does not explicitly encode interactions between blocks. A fixed block count is needed when comparing the scale of these expressions across horizons.

J.3 A two-block example

For blocks of equal norm r with whitened angle θ, direct calculation gives

$$E = \frac{2r^2}{d}(1 + \cos\theta), \qquad U = \frac{4r^2}{d}, \qquad G = \frac{2r^2}{d}(1 - \cos\theta). \tag{59}$$

At $\theta = 0$, the gap vanishes and the error is maximal for these fixed magnitudes. At $\theta = \pi$, the error vanishes while the gap is maximal. Penalizing G alone can therefore favor alignment even when alignment increases error. The prediction terms in the combined objective balance this preference for alignment. This example explains why the gap is optimized together with prediction accuracy.

J.4 Random directions as an analytical reference

If the whitened directions are independent and have zero mean, conditional on fixed magnitudes r_b, their expected cross inner products are zero. Therefore

$$\mathbb{E}[E \mid r_1, \dots, r_m] = \frac{1}{d} \sum_b r_b^2, \qquad \mathbb{E}[G \mid r_1, \dots, r_m] = \frac{2}{d} \sum_{b<c} r_b r_c. \tag{60}$$

For equal magnitudes this yields expected normalized gap $1 - 1/m$. A large normalized gap need not require near-perfect opposition; it can also arise from dispersed directions. Real propagated contributions are neither independent nor generally isotropic. This calculation is an analytical reference for interpretation, not a distributional assumption imposed on the experiments.

K Metric Consistency, Whitening, and Identifiability

K.1 Coordinate changes

For an invertible map T, change coordinates by $z' = Tz$ and transform the metric by $Q' = T^{-\top} Q T^{-1}$. Then for every contribution,

$$(TS_b)^\top Q' (TS_c) = S_b^\top Q S_c. \tag{61}$$

All norms, errors, bounds, and gaps are consequently unchanged. Choosing $T = Q^{1/2}$ gives $Q' = I$, proving the Euclidean equivalence for a fixed positive-definite metric. This invariance is an algebraic property of the error construction, not a claim that the entire training system is invariant. The representation target, direction sampler, optimizer, and planner can change under the same coordinate transformation unless they are transformed consistently.

If the metric is held at I while embeddings are transformed by a non-orthogonal T, temporal losses generally change. Orthogonal transformations preserve Euclidean errors, whereas anisotropic rescaling changes the relative costs of directions. This distinction is particularly relevant when interpreting apparent improvements from changing latent scale or covariance.

K.2 Why the same metric is required

The triangle inequality compares all vectors in one normed space. A counterexample shows why independently chosen metrics cannot be inserted into the bound. In one dimension, take two contributions $S_1 = S_2 = 1$, evaluate the total error with $Q_0 = 1$, and each block norm with $Q_1 = Q_2 = \varepsilon$. Then

$$E^{Q_0} = 4, \qquad (\|S_1\|_{Q_1} + \|S_2\|_{Q_2})^2 = 4\varepsilon. \tag{62}$$

For $0 < \varepsilon < 1$, the supposed bound is smaller than the actual error. Positive definiteness of each metric separately is insufficient. A varying metric is admissible across clips or target times, but not arbitrarily across the terms entering one triangle-inequality argument.

K.3 Spectral bounds

Suppose $mI \preceq Q \preceq MI$ for constants $0 < m \le M$. Then

$$m \|x\|_2^2 \le \|x\|_Q^2 \le M \|x\|_2^2, \tag{63}$$

and, for the same partition,

$$mE^I \le E^Q \le ME^I, \qquad mU^I \le U^Q \le MU^I. \tag{64}$$

These inequalities follow by applying the norm bounds before summing nonnegative block norms. They give scale control for error and upper-bound energies. Subtracting two lower bounds does not yield a bound on their difference G^Q. The always-valid estimates include $0 \le G^Q \le U^Q \le MU^I$.

For the inverse ellipsoidal metric in Equation (25), the largest inverse weight is $1/\min(a, b)$. A small target variance can therefore amplify residuals in its associated directions. If those directions reflect numerical noise rather than task information, inverse weighting can be counterproductive. A spectral floor controls amplification but does not identify the correct task metric.

K.4 A TRAINABLE METRIC CAN CHANGE THE QUESTION

If Q is freely optimized, replacing it by cQ for $c > 0$ multiplies E, U, and G by c. Letting $c \to 0$ reduces every metric-weighted temporal loss without changing any prediction. A scale constraint is necessary to rule out this trivial route, but scale alone does not fix orientation. For example, even under $\mathrm{tr}(Q) = d$, residual-aligned eigenvalues can approach zero while orthogonal eigenvalues increase.

Scale normalization and lower spectral bounds constrain these degeneracies; the hierarchical inequality alone imposes neither constraint. In the reported adaptation experiments $Q = I$, so metric rescaling is not an optimization degree of freedom. The fixed Euclidean metric keeps the meaning of the error measure unchanged during adaptation.

L GRADIENTS AND OPTIMIZATION EFFECTS

L.1 DIFFERENTIATION WITH RESPECT TO BLOCK CONTRIBUTIONS

Hold Q fixed and suppose each $r_b = \|S_b\|_Q$ is nonzero. Differentiating Equation (21) gives

$$\nabla_{S_b} E = \frac{2}{d} Qe, \tag{65}$$

$$\nabla_{S_b} U = \frac{2R}{d} \frac{QS_b}{r_b}, \tag{66}$$

$$\nabla_{S_b} G = \frac{2}{d} \left(R \frac{QS_b}{r_b} - Qe \right). \tag{67}$$

The error gradient is the same for every block because all blocks contribute additively to e. The gap gradient additionally depends on the direction of the individual block. For a parameter ϑ that affects multiple contributions, the chain rule is

$$\nabla_{\vartheta} G = \sum_b \left(\frac{\partial S_b}{\partial \vartheta} \right)^\top \nabla_{S_b} G. \tag{68}$$

The terms share network parameters and can interact through their Jacobians. Interpreting the loss gradient as an independent rotation of each error vector is therefore only a geometric intuition, not the actual neural-network update.

L.2 ACCURACY PLUS GAP

For the simplified objective $J = E + \eta G$ with $\eta \geq 0$,

$$J = (1 - \eta) E + \eta U, \qquad \nabla_{S_b} J = \frac{2}{d} \left((1 - \eta) Qe + \eta R \frac{QS_b}{r_b} \right). \tag{69}$$

At $\eta = 1$, this simplified objective equals U. For $0 \leq \eta \leq 1$, it interpolates error and upper-bound energies. For $\eta > 1$, the algebraic coefficient of E is negative, although the original expression $E + \eta G$ remains nonnegative because $E, G \geq 0$. The joint objective in Equation (17) contains teacher-forced and multi-start losses with different reductions, so this simplified interpolation is not an equivalent expression for the joint objective. The calculation isolates the role of a gap penalty when the underlying error term matches exactly.

L.3 BEHAVIOR AT ZERO CONTRIBUTIONS

The block norm is non-differentiable at zero. Its subdifferential is the dual-norm unit ball,

$$\partial \|S\|_Q \big|_{S=0} = \{ v \in \mathbb{R}^d : \|v\|_{Q^{-1}} \leq 1 \}. \tag{70}$$

To see this, the subgradient inequality at zero is $v^\top x \leq \|x\|_Q$ for every x. Writing $x = Q^{-1/2} y$ identifies the supremum over unit y as $\|v\|_{Q^{-1}}$. Numerical autodifferentiation uses a particular convention at zero, which must be distinguished from the nonzero formula in Equation (67). The forward bound and cancellation identity remain valid even when some blocks vanish.

L.4 Fixed calibration versus adaptive normalization

For fixed positive c_A, c_G, normalization only rescales the gradient contributions. If a loss is divided by its own detached per-batch value, the forward ratio can be numerically close to one while its gradient remains nonzero:

$$\ell(\vartheta) = \frac{L(\vartheta)}{\mathrm{stopgrad}(L(\vartheta))} \quad \Longrightarrow \quad \nabla_\vartheta \ell = \frac{\nabla_\vartheta L}{\mathrm{stopgrad}(L)}. \tag{71}$$

This behaves like a batch-dependent gradient weight and differs from a fixed calibrated objective. The fixed calibration used in the adaptation objective preserves the relative weighting across batches; replacing it with detached batch-dependent scales would change the optimization objective.

M Temporal Weighting and Multi-Start Supervision

M.1 Counting start–target pairs

A clip with observations $z_0, \dots, z_H$ contains $H(H+1)/2$ pairs (s,t) satisfying $0 \le s < t \le H$. At a fixed lag $h = t - s$, there are $H + 1 - h$ such pairs. Writing

$$\ell_{s,h} = \frac{1}{d} \left\| \widehat{z}^{[s]}_{s+h} - z_{s+h} \right\|_2^2, \tag{72}$$

Equation (15) becomes

$$A = \sum_{h=1}^{H} \frac{2(H+1-h)}{H(H+1)} \frac{1}{H+1-h} \sum_{s=0}^{H-h} \ell_{s,h}. \tag{73}$$

Thus uniform weighting of prediction pairs gives a decreasing weight to lag: more short predictions fit inside a clip. For $H = 7$, the lag weights are $7/28, 6/28, \dots, 1/28$. All lags are supervised, but the objective does not allocate equal weight to their averages. This distinction is relevant when interpreting the phrase "multi-step supervision."

The HGap weighting has a different purpose. It combines terminal gaps from a fixed initial history with

$$q_t = \frac{2t}{H(H+1)}, \qquad \sum_{t=1}^{H} q_t = 1. \tag{74}$$

For $H = 7$, this gives the weights $t/28$ used in our objective. A larger target time receives a larger coefficient, emphasizing later predictions within the training clip. The combination of A and G_2 therefore supplies two distinct signals: prediction accuracy across many restart points and a cancellation penalty increasingly weighted toward later target times.

M.2 The effective horizon distributions

For a distribution over lags, its mean indicates where its weight is centered. Standard finite-sum identities give

$$\sum_{h=1}^{H} h \frac{2(H+1-h)}{H(H+1)} = \frac{H+2}{3}, \tag{75}$$

$$\sum_{t=1}^{H} t \frac{2t}{H(H+1)} = \frac{2H+1}{3}. \tag{76}$$

For the eight-frame setting these values are 3 and 5. They characterize the reduction weights, not the actual locations of the largest gradients: residual magnitudes and network Jacobians also matter. Nevertheless, the calculation explains why the multi-start term and temporally weighted gap are not redundant even before considering their different mathematical content.

M.3 Combining pointwise inequalities over time

At each time t, $E_t^Q \le U_{2,t}^Q$. Nonnegative temporal weights preserve this relation:

$$W^Q = \sum_t q_t E_t^Q \le \sum_t q_t U_{2,t}^Q = W^Q + G_2^Q. \tag{77}$$

The same holds after averaging clips or minibatches because averaging is linear and preserves inequalities. It does not imply $A \le W^Q + G_2^Q$, since A uses a different set of starts and, unless specified, a different metric. Replacing one by the other in a derivation would change the objective. Keeping their definitions explicit makes the role of the gap interpretable without overstating the bound.

M.4 What HGap-2 groups at each horizon

At target time t, HGap-2 partitions indices into $\{0,1\},\{2,3\},\ldots$, with a singleton when t is odd. The number of groups is $\lceil t/2 \rceil$. At $t=1$ and $t=2$, there is only one group, so

$$U_{2,t}^Q = E_t^Q, \qquad G_{2,t}^Q = 0 \quad (t \le 2). \tag{78}$$

The gap first becomes active at $t=3$. It therefore does not add a cancellation penalty to the earliest single-group predictions; their accuracy is supervised by the prediction terms. At $t=7$, four groups contribute. This gives the pair grouping a concrete temporal meaning: it retains all cancellation within adjacent pairs and exposes only cancellation spanning those pairs.

For each nonempty block, $S_{B,t}$ is a finite displacement between suitable hybrid predictions when B is contiguous. If $B=\{r,\ldots,s\}$, interior terms cancel and

$$S_{B,t} = Y_t^{(r)} - Y_t^{(s+1)}. \tag{79}$$

Thus the grouped loss can be interpreted directly in terms of distances between selected hybrid endpoints. Computing and storing every individual difference is conceptually convenient but not mathematically necessary for a fixed contiguous partition.

N Linear Dynamics: An Exact Worked Example

The following affine example gives an exact decomposition of local residuals and their propagation. It illustrates the objective analytically without assuming linear dynamics in the control environments.

N.1 Residual propagation

Consider an affine action-conditioned predictor

$$F_j(z) = Mz + b_j, \tag{80}$$

where b_j can depend on the observed action. Its teacher-forced residual is $\delta_j = Mz_j + b_j - z_{j+1}$. Differences between two trajectories receiving the same actions satisfy a homogeneous recurrence because the offsets cancel. Consequently,

$$\Delta_{j,t} = M^{t-1-j}\delta_j, \qquad e_t = \sum_{j=0}^{t-1} M^{t-1-j}\delta_j. \tag{81}$$

This equality follows either by direct recurrence or by substituting the affine downstream map into Equation (47). It separates local residuals from their propagation. A small one-step residual can become a large terminal contribution when powers of M amplify its direction; a larger local residual can become negligible when its direction contracts.

The ordinary rollout error expands as

$$dE_t^I = \sum_j \left\| M^{t-1-j}\delta_j \right\|_2^2 + 2\sum_{j<r} \delta_j^\top (M^{t-1-j})^\top M^{t-1-r}\delta_r. \tag{82}$$

Negative cross terms can offset the sum of contribution energies. HGap aggregates selected contributions into blocks and replaces their cross inner products by the product of their norms in U. The difference between these two quantities is precisely what final-error minimization alone leaves unmeasured.

N.2 Cancellation across two temporal blocks

Take $d = 1$, $M = 1$, $t = 4$, and residuals $(v, 0, -v, 0)$ with $v \neq 0$. The propagated contributions are the same sequence. Pair grouping gives $S_1 = v$ and $S_2 = -v$, hence

$$e_4 = 0, \qquad E_4 = 0, \qquad U_{2,4} = 4v^2, \qquad G_{2,4} = 4v^2. \tag{83}$$

The exact terminal prediction is correct despite two nonzero group contributions. The gap records this dependence on cancellation. Now change the residuals to $(v, -v, 0, 0)$. The cancellation occurs inside the first pair, so $S_1 = S_2 = 0$ and the grouped gap vanishes. HGap-2 therefore does not penalize every departure from one-step accuracy. Its grouping deliberately distinguishes local compensation from compensation across larger temporal segments.

These examples also clarify the role of A and L_{TF}. Within-pair cancellation can hide nonzero individual residuals from HGap-2, while one-step and restarted predictions can still expose them. The loss components cover different aspects of the same sequence rather than defining identical penalties at different weights.

N.3 Contraction and amplification

If $\|M\|_Q \leq \rho$ for the operator norm induced by Q, then

$$\|\Delta_{j,t}\|_Q \leq \rho^{t-1-j} \|\delta_j\|_Q . \tag{84}$$

Under a uniform local bound $\|\delta_j\|_Q \leq \epsilon$, the triangle inequality gives

$$\|e_t\|_Q \leq \epsilon \sum_{r=0}^{t-1} \rho^r = \begin{cases} \epsilon(1-\rho^t)/(1-\rho), & \rho \neq 1, \\ t\epsilon, & \rho = 1. \end{cases} \tag{85}$$

For $\rho < 1$ this is bounded as t grows; for $\rho > 1$ it can grow rapidly. HGap does not enforce $\rho < 1$, and a low empirical gap cannot certify contraction. Instead, it measures the actual contributions along evaluated trajectories, preserving nonlinear behavior in the general construction. This is useful when a single global contraction constant would be either unavailable or too coarse to describe the observed rollouts.

N.4 Why a covariance constraint does not fix these dynamics

Even an exactly isotropic marginal can coexist with different temporal couplings. For a simple population illustration, let $Z \sim \mathcal{N}(0, I)$ and compare predictors $F_1(z) = z$ and $F_2(z) = Oz$, where O is any orthogonal matrix. Both outputs have the same marginal distribution as Z, but for a target next state equal to Z their prediction errors differ:

$$\mathbb{E}\,\|F_1(Z) - Z\|_2^2 = 0, \qquad \mathbb{E}\,\|F_2(Z) - Z\|_2^2 = \|O - I\|_F^2 . \tag{86}$$

The second equality follows from $\mathbb{E}[ZZ^\top] = I$. Marginal matching cannot select the correct coupling on its own. This example supplies a direct mathematical rationale for combining representation regularization with conditional and temporal supervision. It does not imply that every anisotropic representation is inferior or that every temporal regularizer improves planning.

O Sensitivity to Perturbations of Error Contributions

O.1 A deterministic perturbation bound

Fix a metric Q and a partition. Perturb each block by h_b, writing $S'_b = S_b + h_b$, and define

$$\rho = \sum_b \|h_b\|_Q , \qquad e' = \sum_b S'_b, \qquad R' = \sum_b \|S'_b\|_Q . \tag{87}$$

The triangle inequality gives $\|e' - e\|_Q \le \rho$. The reverse triangle inequality applied to each block gives $|R' - R| \le \rho$. Since $R' \le R + \rho$,

$$|U' - U| \le \frac{\rho(2R + \rho)}{d}, \tag{88}$$

$$|E' - E| \le \frac{\rho(2\,\|e\|_Q + \rho)}{d} \le \frac{\rho(2R + \rho)}{d}, \tag{89}$$

$$|G' - G| \le \frac{2\rho(2R + \rho)}{d}. \tag{90}$$

For the first inequality use $|R'^2 - R^2| = |R' - R|(R' + R)$. The second follows by the same argument for $\|e'\|_Q$ and $\|e\|_Q$. The third uses $G = U - E$. These bounds are conservative but require no differentiability and remain valid at zero contributions.

The result explains one motivation for controlling contribution magnitudes: it limits how much the bound and gap can change under a given additive perturbation. It does not show that a smaller gap alone lowers sensitivity, because the right-hand side depends on R, not only on G. A method can reduce G by alignment while leaving R large. The sensitivity bound therefore depends on contribution magnitudes as well as cancellation.

O.2 Worst-case alignment in a defined uncertainty set

Let $x_b = Q^{1/2} S_b$ and suppose each whitened contribution can undergo an arbitrary orthogonal transformation O_b. Consider the purely algebraic uncertainty set

$$\mathcal{U} = \{(O_1 x_1, \ldots, O_m x_m) : O_b^\top O_b = I\}. \tag{91}$$

The triangle inequality gives an upper bound R on the norm of every possible sum. It is attained by mapping each nonzero x_b to $r_b v$ for a common unit vector v. Therefore

$$\sup_{(x'_1,\ldots,x'_m)\in\mathcal{U}} \frac{1}{d} \left\| \sum_b x'_b \right\|_2^2 = \frac{R^2}{d} = U. \tag{92}$$

The gap is the difference between this alignment envelope and the observed energy. This characterizes the grouped bound exactly, without replacing nonlinear trajectories by Jacobian products.

The uncertainty set is deliberately stated in contribution space. It allows independent rotations of blocks, which need not correspond to any realizable perturbation of images, actions, or environment states. Hence the equality is an interpretation of the objective, not an adversarial-control or physical-robustness guarantee. Grouping changes the uncertainty set by preserving the within-group sum before permitting between-group rotations.

O.3 Cancellation and relative sensitivity

When e is nearly zero but R is large, small changes in contribution directions can cause a large change relative to the original total error. For the two-block example $S_1 = v$ and $S_2 = -v$, perturb the second block by h. The new error is exactly h, although the original error was zero. This observation alone does not make cancellation undesirable: a system may reliably produce compensating contributions. HGap encodes a preference against relying on such compensation across the chosen blocks. The empirical merit of that preference depends on whether the measured cancellation persists on the trajectories relevant to planning.

P Why the Loss Components Are Used Together

P.1 Representation shape and temporal prediction solve different problems

The geometry term responds to a batch distribution. The prediction terms respond to the relationship between observations and actions at particular times. Their dependence on the encoder differs: geometry can favor distributed variance even when an encoder makes dynamics harder to predict,

while prediction can favor an easy but insufficient representation. The combination expresses a trade-off between these signals rather than assuming either signal uniquely determines a useful latent space.

For the isotropic anchor weight used in our experiments, $1 - \alpha = 0.9$, Equation (35) becomes

$$R_{\mathrm{iso}} \leq \frac{10}{9} R_{\mathrm{anchor}}. \tag{93}$$

This states a precise benefit of retaining the anchor: a low combined discrepancy still bounds the isotropic discrepancy measured by the same estimator. Full replacement lacks this implication. It does not establish that every task-relevant direction survives, but explains how the objective preserves a substantial part of the baseline regularization while admitting a directional target.

P.2 What the hierarchical gap adds

Ordinary rollout loss has the form $\|\sum_b S_b\|_Q^2 / d$ and is insensitive to changes in individual blocks that preserve their sum. HGap removes part of that indeterminacy by also measuring $\sum_b \|S_b\|_Q$. For example, replacing two blocks by $S_1 + h$ and $S_2 - h$ leaves e unchanged, yet generally changes U and G. The temporal objective can therefore distinguish trajectories that receive the same final-error score.

This additional discrimination is the mathematical benefit of the gap construction. It is not equivalent to declaring every cancelled error harmful. Pair grouping makes the discrimination selective: transformations inside a pair that preserve its sum do not change HGap-2. The representation and prediction losses remain responsible for aspects of the sequence that the grouped gap intentionally leaves unconstrained.

P.3 The role of multi-start accuracy

A final prediction from one initial history covers one route through a clip. Multi-start supervision also tests predictions initialized from later observed histories. In the Markov construction, these predictions coincide with the hybrid terminal values for matching start and target indices. As a result, A constrains the accuracy of the family of branch predictions from which the gap is constructed, rather than only the accuracy of their total telescoping sum.

This gives a useful interpretation of the adaptation loss: the model is trained to make its individual restarted rollouts accurate while limiting selected cancellations among their differences. The two goals are related but distinct. A branch family can have small cross-group gap and poor accuracy if all contributions align; it can have accurate endpoints and nonzero gap if intermediate branch predictions differ. Including accuracy terms prevents the cancellation criterion from becoming the sole definition of a good predictor.

P.4 Frozen geometry makes the temporal effect easier to interpret

With the encoder and projector frozen, z_t and the target history are fixed functions of the observations. Updating the predictor changes generated branches but cannot directly shrink or rotate the encoded targets to reduce temporal losses. This removes an important degree of freedom from the optimization, which is consistent with analyzing HGap as a dynamics adaptation objective.

In end-to-end learning the encoder is also updated. The same change can affect the marginal regularizer, teacher-forced errors, all hybrid branches, and the goal representation. The exact telescoping identity survives, but the optimization problem is different. The frozen-geometry adaptation results alone therefore do not establish how temporal regularization behaves when the encoder is updated. Table 1 evaluates joint training, while Table 3 evaluates predictor adaptation with fixed representations.

Q Statistical Meaning of Seed-Averaged Success

Q.1 Averages and experimental units

Let $X_{s,i} \in \{0,1\}$ denote success for adaptation seed s on evaluation start i. For S seeds and n starts, the average rate is

$$\widehat{p} = \frac{1}{Sn} \sum_{s=1}^{S} \sum_{i=1}^{n} X_{s,i}. \tag{94}$$

Its variance, without an independence assumption, is

$$\mathrm{Var}(\widehat{p}) = \frac{1}{S^2 n^2} \sum_{s,r} \sum_{i,j} \mathrm{Cov}(X_{s,i}, X_{r,j}). \tag{95}$$

The familiar independent-Bernoulli expression discards every off-diagonal covariance. Reused starting states, a common pretrained checkpoint, and related training data can make that simplification inappropriate. The shared initialization and evaluation states introduce sources of dependence between the reported outcomes.

Averaging over evaluation seeds for a fixed trained model likewise does not estimate variation over new training runs. If Θ denotes training randomness and V evaluation randomness, the law of total variance separates these sources:

$$\mathrm{Var}_{\Theta,V}(X) = \mathbb{E}_{\Theta}[\mathrm{Var}_V(X \mid \Theta)] + \mathrm{Var}_{\Theta}(\mathbb{E}_V[X \mid \Theta]). \tag{96}$$

Repeating evaluation for one realized Θ informs only a conditional quantity. It cannot recover the between-training component. This is the reason the ellipsoidal table explicitly identifies one training seed despite averaging six evaluations.

Q.2 Paired comparisons

For methods evaluated on the same starts, define paired outcomes X_i, Y_i and differences $D_i = Y_i - X_i$. Let n_{01} count baseline failures converted into successes and n_{10} count baseline successes converted into failures. Then

$$\widehat{p}_Y - \widehat{p}_X = \frac{n_{01} - n_{10}}{n}. \tag{97}$$

The marginal success counts determine their difference but not the total number of disagreements $n_{01} + n_{10}$. This total controls the variability of a paired comparison. Under independent sampling of paired starts, with $q_{01} = \Pr(X = 0, Y = 1)$ and $q_{10} = \Pr(X = 1, Y = 0)$,

$$\mathrm{Var}\left(\frac{1}{n} \sum_i D_i\right) = \frac{q_{01} + q_{10} - (q_{01} - q_{10})^2}{n}. \tag{98}$$

Thus a paired standard error depends on episode-level disagreements, which are not determined by the marginal success rates alone.

Q.3 Task averages

The four-task mean gives every task equal weight:

$$\widehat{p}_{\mathrm{macro}} = \frac{1}{4} \sum_{k=1}^{4} \widehat{p}_k. \tag{99}$$

This is a descriptive average over the chosen tasks, not an expectation over an independently sampled population of all control problems. In general, a macro-average can increase despite a decline on an individual task. In the current tables, Ellipsoid improves the macro-average while decreasing PushT by 0.66 percentage points. All four task-level differences are positive for joint regularization and for HGap adaptation at each horizon. Reporting task-level means alongside the macro-average preserves that distinction. We use percentage-point differences rather than relative percentage gains to avoid conflating a change in success probability with a proportional change against different baseline levels.

R Training Configuration and Result Aggregation

R.1 Optimization

The combined model is trained for 10 epochs using AdamW with learning rate 10^{-4} and batch size 64. Each task uses one training seed. The visual backbone is ViT-tiny, with a 192-dimensional latent representation. The first epoch uses the isotropic prediction objective, followed by nine epochs with the joint objective; Appendix S specifies calibration and implementation details.

R.2 Regularization coefficients

The geometry regularizer uses $\lambda = 0.09$ and $\alpha = 0.1$. The directional target standard deviations are $\sigma_{\|} = 1.0$ and $\sigma_{\perp} = 0.20$, corresponding to variances 1.0 and 0.04, respectively. Thus the target covariance in Equation (4) is

$$\Sigma_\star = P_k + 0.04(I - P_k). \tag{100}$$

The joint temporal coefficients are $\beta = 0.25$ and $\eta = 0.25$. Substituting these coefficients into Equation (17) yields

$$L_{\text{joint}} = L_{\text{TF}} + 0.081 R_{\text{iso}} + 0.009 R_{\text{ell}} + 0.20 \frac{A}{c_A} + 0.05 \frac{G_2^I}{c_G}. \tag{101}$$

The geometry coefficients multiply unnormalized distributional discrepancies, whereas the temporal coefficients multiply scale-normalized losses. Their numerical magnitudes therefore should not be interpreted as directly comparable gradient strengths.

R.3 Evaluation units

For the combined method in Table 1, each trained model is evaluated using six evaluation seeds, with 50 initial states per seed and task. The reported task success rate is the arithmetic mean of the six evaluation rates. The four-task mean assigns equal weight to each task. Repeating evaluation for a fixed model characterizes evaluation variation, not variation across independent training runs. The frozen-geometry HGap study retains its separate protocol with two adaptation seeds and 200 starts per seed, as specified in Table 3.

S Implementation Details

This appendix specifies the subspace selection, target updates, loss calibration, and finite projection estimator used in our implementation. The calibration schedule below applies to joint training; the frozen-geometry adaptation protocol is described separately in Appendix C.

S.1 Integer subspace dimension and spectral safeguards

For a batch of B clips containing $H+1$ observed frames each, we form the covariance from $N = B(H+1)$ embeddings using Equation (22). We compute the covariance and its symmetric eigendecomposition in float32 even when network operations use mixed precision. We order the eigenvalues as $\lambda_1 \geq \cdots \geq \lambda_d$ and set $\epsilon_C = 10^{-6}$. When $\lambda_1 > \epsilon_C$, we choose

$$k = \min\left\{d, \max\left\{8, \left\lceil \frac{\operatorname{tr}(C)}{\lambda_1} \right\rceil\right\}\right\}, \qquad d = 192. \tag{102}$$

Thus the stable-rank statistic is rounded upward and clipped to $[8, 192]$. This lower limit is a numerical design choice, not a consequence of the bound. We use the leading k orthonormal eigenvectors to form P_k. If $\lambda_1 \leq \epsilon_C$ or the eigendecomposition returns nonfinite values, we use $\Sigma_\star = I$ for that batch. If $k = d$, the target is also I because $\sigma_\parallel = 1$. Repeated eigenvalues at the cutoff can make the selected subspace nonunique; stopping gradients avoids differentiating this choice but does not remove its forward sensitivity.

S.2 Target update frequency and gradient routing

We recompute C, k, and $\Sigma_\star$ once per optimizer step using the observed embeddings from that step. We detach the target construction from automatic differentiation:

$$\widetilde{\Sigma}_\star = \operatorname{stopgrad}(P_k + 0.04(I - P_k)). \tag{103}$$

We evaluate $R_{\mathrm{ell}}(Z; \widetilde{\Sigma}_\star)$ on the original, uncentered embeddings. Gradients flow through the empirical characteristic function into the encoder and projector, but not through covariance estimation, eigenvectors, or the discrete rank choice. We hold the same target fixed across all projections and frequency nodes within the step. We do not use a moving-average covariance. This makes the fixed-target derivative in Appendix G applicable within an update while allowing the target to evolve between updates.

S.3 Joint-loss calibration time and reduction

Within the 10-epoch training budget, we optimize $L_{\mathrm{TF}} + 0.09 R_{\mathrm{iso}}$ for the first epoch. At the end of that epoch, we select 512 clips from the training split using a fixed calibration seed of 0. We evaluate each clip once with dropout disabled and without updating model parameters or BatchNorm running statistics. For clip i, we compute A_i as the mean of its 28 dimension-normalized start–target errors, and compute $G^I_{2,i}$ with the same pair grouping and weights $q_t = t/28$ used during subsequent training. We define

$$c_A = \max\left(\frac{1}{512}\sum_{i=1}^{512} A_i, 10^{-6}\right), \qquad c_G = \max\left(\frac{1}{512}\sum_{i=1}^{512} G^I_{2,i}, 10^{-6}\right). \tag{104}$$

The floor prevents division by zero; it is not a bound on the resulting gradient magnitude. We compute these scales separately for each task and training seed, then hold them fixed for the remaining nine epochs, using Equation (101). All calibration clips come from training data; no validation or evaluation success rates enter the scale calculation. The calibration pass determines the scales for each task and training seed.

S.4 Projection directions and frequency quadrature

We use $M = 1024$ directions per optimizer step. We draw $v_m \sim \mathcal{N}(0, I_d)$ and normalize $u_m = v_m / \|v_m\|_2$, resampling a vector if its norm is zero. We share these directions across all frames

in the step. We resample the directions at the next step and share the same directions between the isotropic and ellipsoidal losses. We use $L = 17$ fixed frequency nodes

$$\omega_\ell = \frac{3\ell}{16}, \quad \ell = 0, \dots, 16, \qquad w_\ell = \begin{cases} 1/32, & \ell \in \{0, 16\}, \\ 1/16, & 1 \le \ell \le 15. \end{cases} \tag{105}$$

These are normalized trapezoidal weights on $[0, 3]$, with $\sum_\ell w_\ell = 1$. For a real projected random variable and a centered Gaussian target, the squared characteristic-function discrepancy is even in frequency, so nonnegative nodes suffice for a symmetric frequency objective. The zero-frequency discrepancy vanishes exactly. We compute the real-valued estimator in Equation (39), averaging over directions and applying the frequency weights above. No additional variance normalization is applied to the embeddings. This quadrature instantiates the generic estimator in Appendix G.