

Article

Unsupervised Segmentation and Alignment of Multi-Demonstration Trajectories via Multi-Feature Saliency and Duration-Explicit HSMMs

Tianci Gao , Konstantin A. Neusypin, Dmitry D. Dmitriev, Bo Yang  and Shengren Rao 

Department IU-1 “Automatic Control Systems”, Bauman Moscow State Technical University, Moscow 105005, Russia; neusypin@bmstu.ru (K.A.N.); dddbmsu@gmail.com (D.D.D.); yanb1@student.bmstu.ru (B.Y.); raoshengren@gmail.com (S.R.)

* Correspondence: gaotianci0088@gmail.com

Abstract

Learning from demonstration with multiple executions must contend with time warping, sensor noise, and alternating quasi-stationary and transition phases. We propose a label-free pipeline that couples unsupervised segmentation, duration-explicit alignment, and probabilistic encoding. A dimensionless multi-feature saliency (velocity, acceleration, curvature, direction-change rate) yields scale-robust keyframes via persistent peak–valley pairs and non-maximum suppression. A hidden semi-Markov model (HSMM) with explicit duration distributions is jointly trained across demonstrations to align trajectories on a shared semantic time base. Segment-level probabilistic motion models (GMM/GMR or ProMP, optionally combined with DMP) produce mean trajectories with calibrated covariances, directly interfacing with constrained planners. Feature weights are tuned without labels by minimizing cross-demonstration structural dispersion on the simplex via CMA-ES. Across UAV flight, autonomous driving, and robotic manipulation, the method reduces phase-boundary dispersion by 31% on UAV-Sim and by 30–36% under monotone time warps, noise, and missing data (vs. HMM); improves the sparsity–fidelity trade-off (higher time compression at comparable reconstruction error) with lower jerk; and attains nominal 2σ coverage (94–96%), indicating well-calibrated uncertainty. Ablations attribute the gains to persistence plus NMS, weight self-calibration, and duration-explicit alignment. The framework is scale-aware and computationally practical, and its uncertainty outputs feed directly into MPC/OMPL for risk-aware execution.

Keywords: hidden semi-Markov model; learning from demonstration; unsupervised segmentation; feature fusion; topological persistence

MSC: 68T05



Academic Editors: Zhijia Zhao and Gang Wang

Received: 19 August 2025

Revised: 10 September 2025

Accepted: 17 September 2025

Published: 23 September 2025

Citation: Gao, T.; Neusypin, K.A.; Dmitriev, D.D.; Yang, B.; Rao, S. Unsupervised Segmentation and Alignment of Multi-Demonstration Trajectories via Multi-Feature Saliency and Duration-Explicit HSMMs.

Mathematics **2025**, *13*, 3057. <https://doi.org/10.3390/math13193057>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Learning from demonstration (LfD) [1] aims to transfer skills from a small number of expert executions to new task instances. In real deployments across aerial, driving, and manipulation domains, multiple demonstrations routinely exhibit irregular local time warping (hovering, waiting, backtracking) [2–4], sensor noise in high-dimensional signals [5,6], and alternating quasi-stationary and transition phases [7–9]. Our goal is to discover phase structure without labels, align multiple demonstrations on a shared

semantic time base, and provide calibrated, segment-wise probabilistic models that can be ingested by constrained planners.

A central difficulty is that widely used segmentation and alignment tools either operate at the signal-threshold level or impose duration assumptions that do not reflect how humans actually perform tasks [10,11]. Threshold- or template-based methods and simple peak detectors are brittle to sampling-rate and environment shifts; their decision surfaces move with scale changes, causing under- and over-segmentation in the presence of jitter and speed variations [12,13]. Bayesian online changepoint detection (BOCPD) relaxes fixed thresholds but still hinges on hazard-rate and noise assumptions; in practice, it can fragment long dwells into multiple short segments when features oscillate and provides no explicit duration model for downstream use [14–16]. Dynamic Time Warping (DTW) offers pairwise alignment but lacks a generative mechanism and cannot represent dwell-time distributions or uncertainty, which limits its utility for planning and simulation [17–19].

Latent-state models address some of these issues, yet common instantiations come with their own bias–variance trade-offs. Hidden Markov Models (HMMs) assume geometric state durations; this geometric-duration bias shortens non-geometric dwells and shifts boundaries when operators hesitate or loiter, leading to misaligned phases across trials [20]. Hidden semi-Markov models (HSMMs) provide a principled remedy by explicitly modeling durations, but many pipelines still rely on hand-crafted feature stacks with fixed weights/hyper-parameters tuned per demonstration or per session [1,21,22]; robustness degrades under operator/platform shifts and tempo changes. Bayesian nonparametrics can adapt model complexity [23], yet computational overhead and stability concerns have limited their use in long-horizon, multi-demo settings.

Downstream generation faces parallel limitations. Dynamical Movement Primitives (DMPs) provide smooth, time-scalable execution but lack closed-form uncertainty; Gaussian Mixture Regression (GMR) and Probabilistic Movement Primitives (ProMPs) offer distributional predictions with covariances, yet both are sensitive to boundary errors—mis-segmentation inflates variances and biases means within segments [20]. More importantly, most pipelines treat segmentation/alignment and probabilistic encoding as loosely coupled stages; mechanisms for self-calibrating features using multi-demo consistency or learning an alignment that is simultaneously scale-aware and duration-aware are rarely present [24]. Finally, although planners such as the Open Motion Planning Library (OMPL) and Model Predictive Control (MPC) can exploit covariance for risk-aware control, few works deliver calibrated uncertainty on a shared semantic time base that these planners can consume directly [25].

To address these gaps, we propose a label-free pipeline that couples unsupervised segmentation, duration-explicit alignment, and probabilistic encoding in a single, self-calibrating loop. First, we compute a dimensionless multi-feature saliency by fusing velocity, acceleration, curvature, and direction-change rate and then apply topology-aware keyframe extraction using persistent peak–valley pairs and non-maximum suppression to retain only structurally significant extrema [26]. Second, we jointly train an HSMM across demonstrations with explicit duration distributions and an extended forward–backward recursion, producing a shared semantic time axis and phase-consistent boundaries [25]. Third, within each phase, we fit probabilistic motion models—GMM/GMR or ProMP, optionally combined with DMP for execution—to obtain mean trajectories with calibrated covariances [20]. Crucially, we close the loop by learning the saliency weights without labels: a CMA-ES search on the probability simplex minimizes cross-demonstration structural dispersion, automatically re-balancing features so that segmentation and alignment are mutually consistent [27]. Compared with BOCPD, thresholding, or HMM-based pipelines, our

design is explicitly duration-aware, robust to time warping, and planner-ready, producing uncertainty that is calibrated and comparable across operators.

Contributions.

1. Scale-/time-warp-robust saliency. A topology-aware, multi-feature saliency (persistence + non-maximum suppression) that stabilizes keyframes under noise and tempo variations is developed, yielding sparser yet more stable anchors than signal-level detectors, and we formalize its stability in Proposition 1 (Section 2.3.5).
2. Joint HSMM alignment with explicit durations. Cross-demonstration training with extended forward–backward/Viterbi recursions is used; model order is selected by a joint criterion combining BIC and alignment error (AE) to balance fit and parsimony.
3. Label-free feature-weight self-calibration. CMA-ES on the weight simplex is used to minimize cross-demonstration structural dispersion, eliminating hand-tuned fusion and improving phase consistency.
4. Calibrated probabilistic encoding for planning. Segment-wise GMR/ProMP (optionally fused with DMP) returning means and covariances that integrate directly with OMPL/MPC for risk-aware execution is used.

Empirically, on UAV flight, autonomous-driving, and manipulation benchmarks, our method reduces phase-boundary dispersion by $\approx 31\%$ on UAV-Sim and by 30–36% under monotone time warps, additive noise, and missing data compared with HMM variants; it improves the sparsity–fidelity–smoothness trade-off (higher time compression at comparable reconstruction error with lower jerk) and achieves nominal 2σ coverage (94–96%), indicating well-calibrated uncertainty. Section 2 details the pipeline; Section 3 reports datasets, metrics, baselines (including BOCPD and HMM), ablations, and robustness; Section 4 discusses positioning, limitations, and future work; Section 5 concludes.

2. Materials and Methods

Figure 1 sketches the end-to-end workflow that maps multiple demonstrations to executable trajectories under physical and safety constraints:

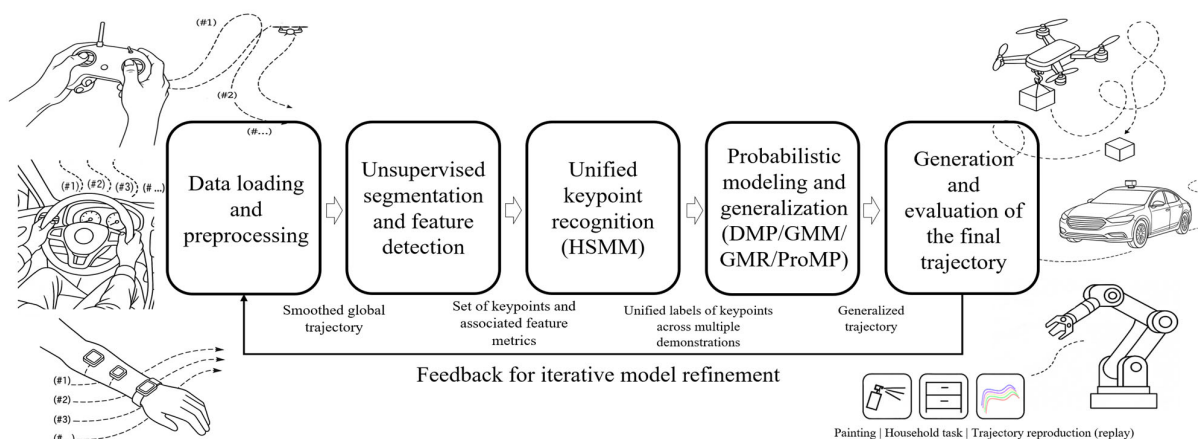


Figure 1. Overview of the proposed pipeline from multiple demonstrations to executable trajectories.

(i) unsupervised segmentation from multi-feature saliency with topology-aware keyframes; (ii) duration-explicit alignment across demonstrations via a hidden semi-Markov model (HSMM) with shared parameters; and (iii) probabilistic encoding of each phase (GMM/GMR or ProMP, optionally combined with DMP) producing mean trajectories with calibrated covariances. Outputs include phase-consistent labels, segment-wise probabilistic models, and constraint-aware executable trajectories.

2.1. Inputs, Outputs, and Assumptions

Inputs.

We observe M independent demonstrations (optionally multi-modal), $\{p^{(m)}\}_{m=1}^M$, $p^{(m)} = \{p^{(m)}(t)\}_{t=1}^{T_m}$, sampled with fixed period Δt . When available, auxiliary channels (e.g., pose, force/tactile, depth) are concatenated into the observation vector used downstream.

Outputs.

1. A shared set of semantic phases $\{S_k\}_{k=1}^{N^*}$ and per-demo boundaries $\{\tau_k^{(m)}\}$;
2. For each phase, a segment-wise generative model—DMP, GMM/GMR, or ProMP, alone or in combination—returning mean trajectories and covariance estimates;
3. Under constraints $\mathcal{C}$ (terminal/waypoints, velocity/acceleration limits, etc.), an executable trajectory and associated risk measures computed from covariances.

Assumptions.

1. Demonstrations consist of alternating quasi-stationary and transition segments;
2. Time deformation is order-preserving (the semantic phase order does not change);
3. Observation noise is moderate and can be mitigated by local smoothing and statistical filtering.

2.2. Multi-Feature Analysis and Automatic Segmentation

2.2.1. Data Ingestion and Pre-Processing

Let a single demonstration be the discrete sequence $\mathcal{D} = \{\Delta \mathbf{x}_t\}_{t=1}^T$, $\Delta \mathbf{x}_t \in \mathbb{R}^d$, where d is the number of observed degrees of freedom (e.g., $d = 3$ for UAV or mobile bases; $d = 6$ for industrial manipulators). Cumulative sums yield the Cartesian trajectory of the tool-center point

$$\mathbf{P}(t) = [p_x(t), p_y(t), p_z(t)]^\top, t = 1, \dots, T, \quad (1)$$

We apply a one-dimensional Savitzky–Golay local polynomial smoother to $\mathbf{P}(t)$ before computing derivatives, suppressing high-frequency tele-operation jitter and stabilizing numerical differentiation. The filter is often interpreted as a local $(r + 1)$ -order Taylor approximation in time and ensures

$$\|\tilde{\mathbf{P}} - \mathbf{P}\|_\infty \leq c_W \varepsilon, c_W < 1,$$

where ε bounds the measurement noise and the window $W \in \{5, 7, 9\}$. Unless stated otherwise, $\tilde{\mathbf{P}}$ denotes the smoothed trajectory [28].

Implementation note. Central differences are used for derivatives; all channels are time-synchronized at a fixed sampling period Δt .

2.2.2. Feature Computation and Saliency Fusion

On the smoothed trajectory, we compute four complementary, time-varying features that reveal when, where, and how kinematic changes occur ($\Delta t (= \frac{1}{f_s})$ is fixed).

1. Velocity. Let

$$\mathbf{v}(t) = \frac{\mathbf{P}(t+1) - \mathbf{P}(t)}{\Delta t}, v(t) = \|\mathbf{v}(t)\|_2. \quad (2)$$

2. Acceleration.

$$\mathbf{a}(t) = \frac{\mathbf{v}(t+1) - \mathbf{v}(t)}{\Delta t}, a(t) = \|\mathbf{a}(t)\|_2. \quad (3)$$

Peaks indicate abrupt speed changes.

3. Curvature. With $\Delta \mathbf{P}(t) = \mathbf{P}(t+1) - \mathbf{P}(t)$,

$$\kappa(t) = \frac{2\|\Delta \mathbf{P}(t-1) \times \Delta \mathbf{P}(t)\|}{\|\Delta \mathbf{P}(t-1)\| \|\Delta \mathbf{P}(t)\| \|\Delta \mathbf{P}(t) - \Delta \mathbf{P}(t-1)\|}. \quad (4)$$

Curvature measures spatial bending and is naturally invariant to global time scaling due to the cubic velocity term [29,30].

4. Direction-Change Rate (DCR). Define the unit direction vector $\hat{\mathbf{v}}(t) = \mathbf{v}(t)/\|\mathbf{v}(t)\|_2$. To avoid numerical issues at very low speed, introduce a threshold $v_{\min} > 0$ and set

$$\text{DCR}(t) = \begin{cases} \|\hat{\mathbf{v}}(t) - \hat{\mathbf{v}}(t-1)\|_2, & \|\mathbf{v}(t)\|_2 \geq v_{\min} \\ 0, & \|\mathbf{v}(t)\|_2 < v_{\min} \end{cases} \quad (5)$$

5. Dimensionless fusion. Apply min–max normalization to each feature to obtain $\tilde{\mathbf{v}}(t), \tilde{\mathbf{a}}(t), \tilde{\kappa}(t), \widetilde{\text{DCR}}(t) \in [0,1]$. For a weight vector $\mathbf{w} = [w_v, w_a, w_\kappa, w_d]^\top$ with $w_i \geq 0$ and $\sum_i w_i = 1$, define the fused saliency

$$\text{Score}(t; \mathbf{w}) = w_v \tilde{v}(t) + w_a \tilde{a}(t) + w_\kappa \tilde{\kappa}(t) + w_d \widetilde{\text{DCR}}(t) \quad (6)$$

The weights w are learned without labels in Section 2.2.4.

2.2.3. Keyframe Extraction with Topological Simplification

The saliency $\text{Score}(t)$ compresses multi-source information into a 1D signal, but segmentation should rely on structural extrema (global landmarks), not every minor fluctuation. We adopt a bottom-up screening that contracts a dense set of local extrema into a sparse, stable keyframe set [1,31].

1. Candidate extrema via quantile thresholds.

Let $\mathcal{Q} = \{q_1, \dots, q_L\} \subset (0,1)$ be a grid of quantiles (e.g., uniformly in $[0.60, 0.95]$). For each $q \in \mathcal{Q}$:

- Set $\tau_q = \text{quantile}_q(\text{Score})$;
- Collect indices $\tilde{E}_q = \{t \mid \text{Score}(t) > \tau_q\}$ and snap each t to the nearest local extremum within a radius-3 neighborhood.

To pick a unique q^* , minimize the sparsity–fidelity loss

$$L(q) = \underbrace{\left| \tilde{E}_q \right| / T}_{\text{sparsity}} + \lambda \underbrace{\text{MSE}\left(P, \hat{P}_{\tilde{E}_q}\right)}_{\text{reconstruction}},$$

where $\hat{P}_{\tilde{E}_q}$ is the spline reconstruction at $\tilde{E}_q$ and $\lambda > 0$ reflects the admissible reconstruction error. Set $q^* = \arg\min_{q \in \mathcal{Q}} L(q)$, $\tilde{E} = \tilde{E}_{q^*}$.

2. Persistence thresholding (scale-aware importance)

For adjacent peak–valley pairs $(p_{\max}, p_{\min})$ of $\text{Score}(t)$, define the persistence

$$\text{pers}(p_{\max}, p_{\min}) = |\text{Score}(p_{\max}) - \text{Score}(p_{\min})|, \quad (7)$$

Small persistence typically indicates noise or micro-tremor; large persistence corresponds to genuine kinematic transitions. Plot

$$g(\alpha) = |\{(p_{\max}, p_{\min}) \mid \text{pers} > \alpha\}|,$$

which empirically exhibits a plateau–cliff–stable pattern; the elbow α^* is detected by Kneedle. Keep

$$E^{\dagger} = \{p_i \mid \text{pers}(p_i) > \alpha^*\}$$

Because persistence depends only on amplitude differences, it is invariant to vertical scaling and mild time stretching, facilitating cross-demonstration comparability [32,33].

3. Non-maximum suppression (NMS).

To avoid peak clustering, scan $E^{\dagger}$ with a sliding window of NMS_w frames and retain an extremum only if it is the largest (same polarity) within the window. The final keyframe set is

$$\mathcal{K} = \text{NMS}_w(E^{\dagger}).$$

The effect of persistence thresholding and NMS is illustrated in Figure 2, which contracts dense peak clusters into a sparse, stable keyframe set $\mathcal{K}$. The black solid curve is the original saliency signal $\text{Score}(t)$; the dashed curve is the simplified signal after persistence-based filtering (elbow α^* detected by Kneedle) and non-maximum suppression (window w). Green crosses mark the final retained keyframes $\mathcal{K}$. The procedure removes clustered minor extrema while preserving structural landmarks that drive segmentation and alignment.

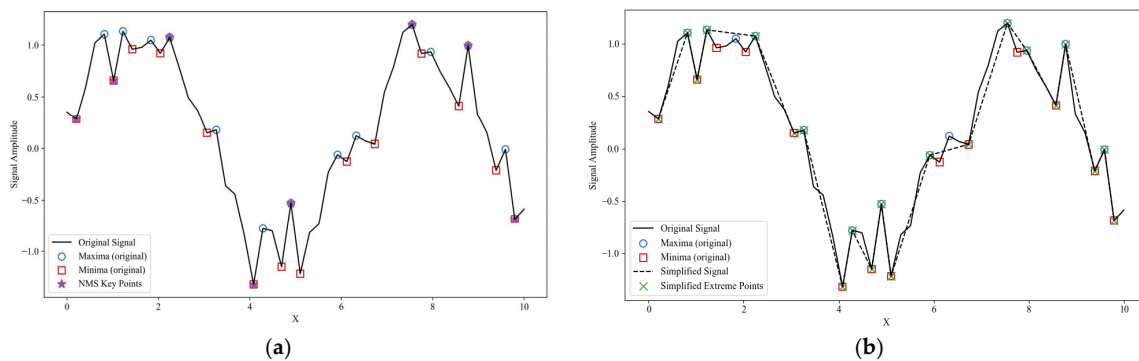


Figure 2. Topology-aware keyframe selection. (a) Effect of non-maximum suppression (NMS) alone on the original saliency signal: local maxima/minima are pruned within a $\pm w$ neighborhood (stars). (b) Persistence-based simplification (dashed) removes low-amplitude peak–valley pairs; applying the same NMS to the survivors yields the final keyframes $\mathcal{K} = \text{NMS}_w(E^{\dagger})$ (stars). Persistence discards small-scale oscillations, while NMS prevents clustered multi-responses.

The effect of the two simplification steps is shown in Figure 2. Panel (a) isolates the role of non-maximum suppression (NMS) on the original saliency $\text{Score}(t)$ (Equation (6)): within a $\pm w$ window, only the strongest same-polarity extremum is kept (stars), turning dense local clusters into a sparser set of candidates. Panel (b) shows persistence-based simplification: the dashed curve is the signal after removing low-amplitude peak–valley pairs using the Kneedle elbow α^* ; crosses mark the surviving extrema. Applying the same NMS to these survivors (not shown for clarity) yields the final keyframe set $\mathcal{K} = \text{NMS}_w(E^{\dagger})$. Together, persistence discards small-scale oscillations in a scale-robust manner, while NMS prevents multi-responses in high-energy neighborhoods.

With $\mathcal{K}$ in place, the reliability of subsequent segmentation and HSMM alignment still depends on the saliency weights w ; Section 2.2.4 details a label-free, consistency-driven calibration.

2.2.4. Adaptive Feature-Weight Learning

In (6), the saliency $\text{Score}(t; w)$ is a linear fusion of four heterogeneous features. Assigning heuristic, fixed weights $w = [w_v, w_a, w_k, w_d]^{\top}$ typically overfits a particular op-

erator, platform, or task, degrading segmentation under distribution shift. We therefore treat w as model parameters to be estimated without labels by enforcing cross-demonstration consistency.

1. Consistency functional.

For M demonstrations $\{\mathbf{P}^{(m)}\}_{m=1}^M$ and a candidate w , define

$$\mathcal{F}(\cdot; w) : \mathbf{P}^{(m)} \mapsto \hat{\mathbf{P}}^{(m)}(w), \quad (8)$$

and denote the composition of saliency construction, keyframe extraction (Figure 2), HSMM-based alignment (Section 2.3), and resampling on shared semantic nodes. If w is well chosen, the reconstructions $\{\hat{\mathbf{P}}^{(m)}(w)\}$ should be congruent in shape and timing. We quantify this by the mean point-to-point structural dispersion.

$$\text{SOD}(w) = \frac{2}{M(M-1)} \sum_{1 \leq m < m' \leq M} \frac{1}{T_{\min}} \sum_{t=1}^{T_{\min}} \left\| \hat{\mathbf{p}}_t^{(m)}(w) - \hat{\mathbf{p}}_t^{(m')}(w) \right\|_2, \quad (9)$$

where $T_{\min}$ is the minimal resampled length. Smaller SOD means lower structural variance on the shared semantic time base.

2. Objective and constraints.

To prevent dominance by any single channel and to improve identifiability, we regularize with a light $\downarrow_2$ penalty and minimize over the probability simplex:

$$w^* = \arg \min_{w \in \Delta^3} \left(\text{SOD}(w) + \lambda \|w\|_2^2 \right), \Delta^3 = \left\{ w \mid w_i \geq 0, \sum_i w_i = 1 \right\}. \quad (10)$$

Because F contains non-smooth steps (extrema detection, discrete decoding), $\text{SOD}(w)$ is non-differentiable and gradient methods are unreliable.

3. Solver and feasibility.

We employ the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) in an unconstrained space together with a softmax reparameterization

$$w_i(z) = \frac{\exp(z_i)}{\sum_{j=1}^4 \exp(z_j)}, z \in \mathbb{R}^4,$$

which enforces $w \in \Delta^3$ at every iteration while preserving Gaussian updates in z -coordinates. CMA-ES is appropriate here because it (i) requires only function values of $J(w) = \text{SOD}(w) + \lambda \|w\|_2^2$; (ii) adapts search scales via the evolving covariance; and (iii) is empirically robust on multi-modal, non-convex objectives [34,35]. We terminate when either the covariance trace shrinks below a preset ratio (indicating a local neighborhood) or the successive improvement $|J^{(k)} - J^{(k-1)}|$ falls below ϵ . Under mild regularity on J in the z -space, the iterates approach first-order stationarity (a near-KKT solution) with high probability [36].

The objective $J(w)$ combines non-smooth steps (extrema detection, decoding), causing piecewise-flat regions and discontinuous sub-gradients. In preliminary trials, gradient-based schemes (finite-difference, SPSSA, REINFORCE-style estimators) were unstable and sensitive to step sizes, while Bayesian optimization/TPE struggled with the noisy, batch-evaluated objective and required heavy surrogate retraining when caching was used. CMA-ES only needs function values, adapts the search covariance online, parallelizes naturally across demonstrations, and shows consistent convergence to low-SOD solutions

under identical budgets. We therefore adopt CMA-ES as the default black-box solver (Section 3.6 ablations leave the outer loop intact).

4. Computational profile.

Each evaluation of $J(w)$ entails one pass of keyframe extraction (linear in sequence length T) and one HSMM forward-backward/decoding pass per demonstration with complexity $O(NTD_{max})$; overall cost scales linearly in M . For numerical stability, we implement recursions in the log domain (log-sum-exp), apply a diagonal floor to GMM covariances, and cache feature streams across outer-loop calls.

The saliency front-end (feature streams + persistence + NMS) is $O(T)$ and memory-light; in practice, it processes sequences faster than real time at the sampling rates used here (10–100 Hz). HSMM inference is linear in the number of states and the maximum dwell ($O(N \cdot T \cdot D_{max})$), and the outer consistency loop amortizes via cached features and log-domain recursions. Section 3 reports end-to-end timings per domain; inference-time segmentation/alignment remains within interactive latency on a commodity CPU.

2.3. Multi-Demo Alignment and Segmentation via a Duration-Explicit HSMM

Given the sparse, scale-invariant keyframes $\mathcal{K}^{(m)} = \{t_k^{(m)}\}_{k=1}^{K_m}$ obtained from saliency (Section 2.2) and the learned fusion weights (Section 2.2.4), we seek a shared semantic time base across demonstrations. Directly matching wall-clock indices is unreliable due to operator-dependent pauses and backtracking; instead, we align demonstrations probabilistically via a Hidden semi-Markov model (HSMM) that explicitly models state durations (Figure 3) [37,38].

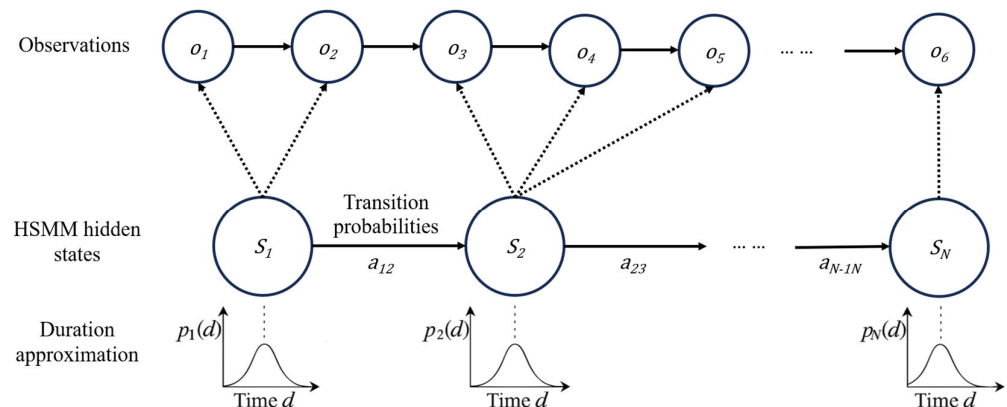


Figure 3. Schematic of the HSMM. Circles denote random variables (top: observations o_t ; bottom: latent phases S_i). Solid arrows indicate Markovian dependencies; dashed links connect latent states to observation likelihoods. Shaded curves at the bottom depict state-specific duration distributions $p_i(d)$.

2.3.1. Model and Generative Mechanism

Let $S = \{S_1, S_2, \dots, S_N\}$, $N \ll K_m$ be latent phases, each representing a macro-action (e.g., grasp, insert, lift). With parameters $\Theta = \{\pi, A, \{p_i\}, \{b_i\}\}$, the HSMM generates $o_t \in \mathbb{R}^d$ at $t = 1, \dots, T$ as follows:

- Initial phase: $q_1 \sim \text{Cat}(\pi)$, $\sum_i \pi_i = 1$.
- Duration: for the current phase q_k , sample dwell length $d_k \sim p_{q_k}(d) \in \mathbb{N}_{>0}$.
- Observations: for $\tau_{k-1} < t \leq \tau_k$ with $\tau_k = \sum_{j \leq k} d_j$, $o_t \sim b_{q_k}(\cdot)$, $b_i(o) = \sum_{m=1}^{M_i} \pi_{i,m} \mathcal{N}(o \mid \mu_{i,m}, \Sigma_{i,m})$.
- Transition: $\Pr(q_{k+1} = j \mid q_k) = a_{q_k j}$; terminal transitions end the sequence.

Writing $O = (o_1, \dots, o_T)$, $Q = (q_1, \dots, q_n)$, $D = (d_1, \dots, d_n)$, and $q^*(t) = q_k$ for $\tau_{k-1} < t \leq \tau_k$, the joint density is

$$P(O, Q, D) = \pi_{q_1} p_{q_1}(d_1) \prod_{k=1}^{n-1} [a_{q_k q_{k+1}} p_{q_{k+1}}(d_{k+1})] \prod_{t=1}^T b_{q^*(t)}(o_t). \quad (11)$$

Observation design. We concatenate kinematic descriptors (e.g., curvature, speed magnitude, DCR) and any available modalities (pose, force/tactile, depth) into o_t .

Duration choices. We use either discrete $p_i(d)$ with support 1: D_{max} or truncated Gaussian/Gamma families to accommodate unequal dwell times [39].

2.3.2. Parameter Estimation: Extended Baum–Welch

We maximize the total log-likelihood over demonstrations

$$\mathcal{L}(\Theta) = \sum_{m=1}^M \log P(O^{(m)} | \Theta). \quad (12)$$

Explicit durations break first-order Markovity; forward–backward recursions therefore enumerate duration indices.

1. Forward variable (leaving i at time t).

$$\alpha_t(i) = \sum_{j=1}^N \sum_{d=1}^{\min(D_{max}, t)} \alpha_{t-d}(j) a_{ji} p_i(d) \prod_{r=0}^{d-1} b_i(o_{t-r}), \alpha_0(i) = \pi_i. \quad (13)$$

2. Backward variable.

$$\beta_t(i) = \sum_{j=1}^N \sum_{d=1}^{T-t} \left[a_{ij} p_j(d) \prod_{r=0}^{d-1} b_j(o_{t+1+r}) \beta_{t+d}(j) \right], \beta_T(i) = 1. \quad (14)$$

3. Posteriors (E-step).

$$\gamma_t(i) = \frac{\sum_{d=1}^{\min(D_{max}, t)} \sum_{r=0}^{d-1} \alpha_{t-r}(i) p_i(d) \prod_{u=0}^{d-1} b_i(o_{t-r+u}) \beta_{t-r}(i)}{\sum_{k=1}^N \alpha_T(k)} \quad (15)$$

$$\xi_{t,d}(i, j) = \frac{\alpha_{t-d}(i) a_{ij} p_j(d) \prod_{r=0}^{d-1} b_j(o_{t-r}) \beta_t(j)}{\sum_{k=1}^N \alpha_T(k)} \quad (16)$$

Sum γ, ξ over m to obtain corpus-level sufficient statistics.

4. M-step (closed forms).

$$\begin{aligned} \pi_i^{\text{new}} &= \gamma_{t=1}(i), \\ a_{ij}^{\text{new}} &= \frac{\sum_t \xi_{t,d}(i, j)}{\sum_t \gamma_t(i)}, \\ p_i^{\text{new}}(d) &= \frac{\sum_t \xi_{t,d}(i, \cdot)}{\sum_{t,d'} \xi_{t,d'}(i, \cdot)}, \\ \mu_{i,m}^{\text{new}} &= \frac{\sum_t \gamma_{t,m}(i) o_t}{\sum_t \gamma_{t,m}(i)}, \\ \Sigma_{i,m}^{\text{new}} &= \frac{\sum_t \gamma_{t,m}(i) (o_t - \mu_{i,m})(o_t - \mu_{i,m})^\top}{\sum_t \gamma_{t,m}(i)} \end{aligned} \quad (17)$$

with $\gamma_{t,m}(i) = \gamma_t(i)\pi_{i,m}\mathcal{N}(o_t | \mu_{i,m}, \Sigma_{i,m})/b_i(o_t)$.

5. Numerical stability.

All recursions are implemented in the log domain using log-sum-exp

$$\text{LSE}(x_1, \dots, x_K) = \log \sum_{k=1}^K e^{x_k}, \quad (18)$$

and GMM covariances receive a diagonal floor δI . EM monotonicity guarantees convergence of $\mathcal{L}^{(k)}$ to a stationary point [40,41].

2.3.3. Semantic Time Axis: Decoding and Outputs

After convergence, Viterbi decoding yields the MAP phase path $\hat{Q}^{(m)} = (q_1^{(m)}, \dots, q_N^{(m)})$ and durations $\hat{D}^{(m)} = (d_1^{(m)}, \dots, d_N^{(m)})$ for each demonstration, with $\sum_{k=1}^N d_k^{(m)} = T_m$. Define cumulative boundaries $\tau_k^{(m)} = \sum_{j=1}^k d_j^{(m)}, k = 1, \dots, N$, so segments $[\tau_{k-1}^{(m)}, \tau_k^{(m)}]$ correspond to the same semantic phase S_k across demonstrations. Figures 4 and 5 illustrate 3D keyframe locations and HSMM reconstructions aligned to salient kinematic transitions.

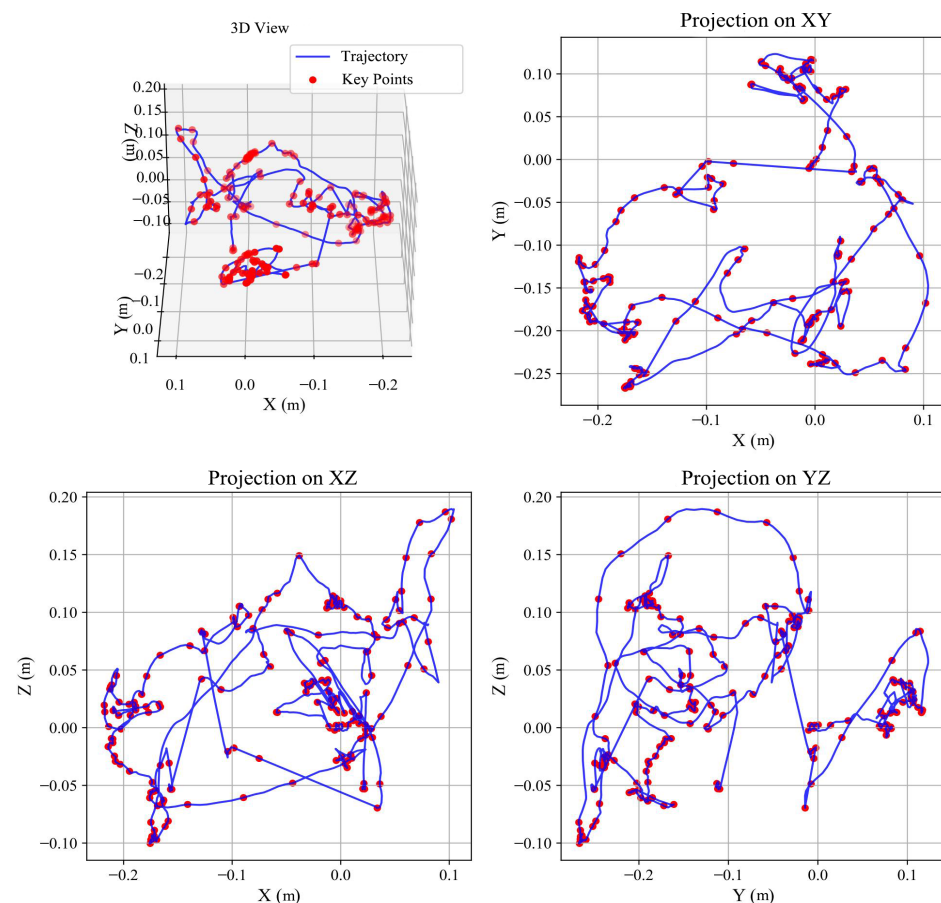


Figure 4. Example 3D trajectories with keyframes (red dots) shown in three orthographic projections. Keyframes concentrate at motion-structure bends.

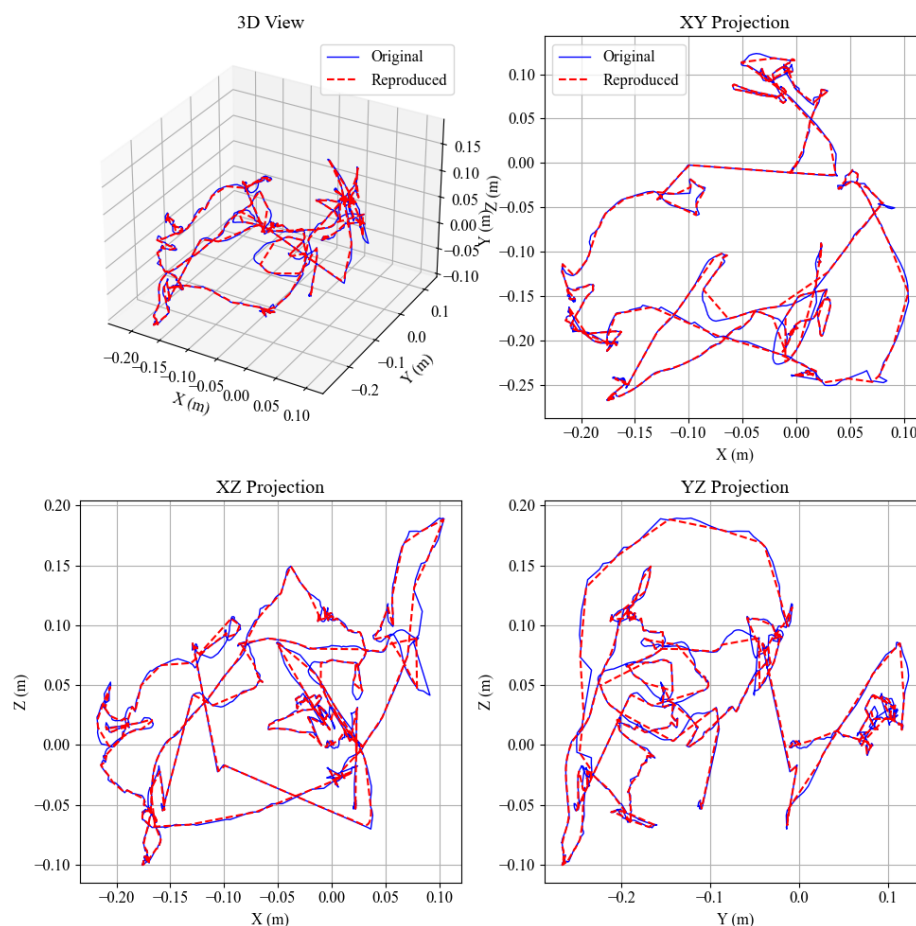


Figure 5. Original trajectory (blue) versus HSM reconstruction (red dashed). Boundaries align with salient kinematic transitions, validating the learned semantic phases.

2.3.4. Alignment Quality, Model Selection, and Robustness

Alignment metric. We quantify cross-demo temporal agreement by

$$AE(N) = \sum_{m=1}^M \left\| \left(\tau_1^{(m)}, \dots, \tau_N^{(m)} \right) - (\bar{\tau}_1, \dots, \bar{\tau}_N) \right\|_2, \bar{\tau}_k = \frac{1}{M} \sum_m \tau_k^{(m)}. \quad (19)$$

Model selection. To balance fit and parsimony, we use

$$BIC(N) = -2 \log \mathcal{L}(\Theta_N^*) + \kappa_N \log \left(\sum_m T_m \right), \quad (20)$$

and select N^* by jointly considering $AE(N)$ and $BIC(N)$.

Robustness to time warps. For any order-preserving reparameterization of time, $t \mapsto \tau(t)$, the decoded phase order $(S_1, \dots, S_N)$ is invariant; only the tail behavior of duration distributions $p_i(d)$ is rescaled. This accounts for operator-dependent slowdowns, hesitations, or hovering.

Complexity. Keyframe processing is $O(T)$ per sequence; duration-explicit forward-backward is $O(NTD_{max})$; training scales linearly in the number of demonstrations M . Outputs include $\{S_k\}$ and segment-wise statistics for direct use by planners (e.g., OMPL/MPC).

2.3.5. Stability Properties of Topology-Aware Saliency and Duration-Explicit Alignment

Proposition 1. (Scale and weak time-warp stability of the retained keyframes).

Let $\text{Score}(t; w)$ be the fused saliency in Equation (6). Let $\mathcal{K}$ be the keyframe set after persistence-based simplification (Equation (7), Kneedle elbow) and non-maximum suppression (Figure 2). Consider (a) vertical rescaling of all input features by $\lambda > 0$ and (b) order-preserving time reparameterizations ϕ with bounded local stretch. Then, the following are obtained:

1. Scale invariance. $\mathcal{K}(\lambda\text{-Score}) = \mathcal{K}(\text{Score})$ since persistence depends on amplitude differences and the elbow index is preserved.
2. Weak time-warp stability (physical time). Under (b), extrema order is preserved; locations shift by at most $O(w_{nms})$ samples due to the NMS window.
3. Semantic-time invariance with HSMM. After duration-explicit decoding (Section 2.3), the phase order and boundaries induced by $\mathcal{K}$ are invariant; order-preserving warps manifest primarily as duration redistribution across states (Section 2.3.4).

Sketch. Items 1–2 follow from persistence stability and local maximality under NMS (Equation (7), Figure 2). Item 3 uses explicit-duration decoding, which preserves phase order under monotone reparameterizations; empirical trends in Sections 3.6 and 3.9 support these properties.

2.4. Statistical Motion Primitives and Probabilistic Generation

With cross-demonstration phases and boundaries $\{\mathcal{S}_k\}_{k=1}^N, \{\tau_k^{(m)}\}$ obtained from the duration-explicit HSMM (Section 2.3), we model each phase *within* the shared semantic time base. For demonstration m , let

$$\mathcal{C}_k^{(m)} = \left\{ P^{(m)}(t); t \in \left(\tau_{k-1}^{(m)}, \tau_k^{(m)} \right] \right\}$$

denote the spatio-temporal segment for phase $\mathcal{S}_k$. Our objective is a segment-wise generative model that (i) captures cross-demo variability, (ii) supports conditioning and duration re-scaling, and (iii) yields calibrated uncertainty for downstream planning and safety assessment. We instantiate three complementary families: DMP, GMM/GMR, and ProMP. Unless otherwise noted, time within a segment is normalized to a phase variable $s \in [0, 1]$, ensuring a uniform interface across models.

2.4.1. Dynamic Movement Primitives (DMP)

1. Single-segment dynamics.

For a one-degree-of-freedom trajectory $x(t)$, the classical DMP represents motion as a critically damped second-order system with a nonlinear forcing term:

$$\begin{aligned} \tau \dot{z} &= \alpha_z (\beta_z (g - x) - z) + f(s), \\ \tau \dot{x} &= z, \tau \dot{s} = -\alpha_s s, \end{aligned} \quad (21)$$

where g is the segment goal, $s \in (0, 1]$ is a phase variable, and

$$f(s) = \frac{\sum_{i=1}^B w_i \psi_i(s)}{\sum_{i=1}^B \psi_i(s)} (g - x_0), \psi_i(s) = \exp \left[-h_i (s - c_i)^2 \right] \quad (22)$$

Given $\tau, \alpha_z, \beta_z, \alpha_s$, the weights $\{w_i\}$ are obtained by least squares (or locally weighted regression). Multi-DOF trajectories are modeled component-wise or via task-space coupling.

2. Segment coupling and smoothness.

Let $\bar{d}_k$ be the mean duration of phase $\mathcal{S}_k$ across demonstrations; set $\tau_k = \bar{d}_k \Delta t$. We fit $\{w_i^{(k)}\}_{i=1}^B$ per segment and compose $\{(\text{DMP}_k, \tau_k)\}_{k=1}^N$ along the decoded boundaries. Because the state (x, z) is continuous across boundaries, the concatenation is C^1 -continuous

without auxiliary velocity/acceleration matchers. DMPs thus offer low jerk and robust time-scaling at execution.

2.4.2. Gaussian Mixture Modeling and Regression (GMM/GMR)

Mixture modeling. For each segment, we pair the normalized phase with position, $(s_t, y_t) \in \mathbb{R}^{1+d}$, and fit a K -component mixture

$$p(\mathbf{s}, \mathbf{y}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mu_k, \Sigma_k) \quad (23)$$

typically with diagonal (or block-diagonal) Σ_k to denoise while retaining principal correlations.

Regression and uncertainty. At execution, Gaussian Mixture Regression (GMR) produces the conditional

$$p(y | s) = \mathcal{N}(\hat{\mu}(s), \hat{\Sigma}(s)),$$

yielding a closed-form mean trajectory and a phase-indexed covariance band. Analytical derivatives of $\hat{\mu}(s)$ and $\hat{\Sigma}(s)$ facilitate integration with MPC for online terminal corrections and constraint handling.

2.4.3. Probabilistic Movement Primitives (ProMP)

Bayesian representation. Using basis functions $\Phi_t = [\phi_1(t), \dots, \phi_B(t)]^\top$, a segment trajectory is modeled as

$$y_t = \Phi_t^\top w + \varepsilon, \varepsilon \sim \mathcal{N}(0, \Sigma_y), w \sim \mathcal{N}(\mu_w, \Sigma_w) \quad (24)$$

where (μ_w, Σ_w) encode the distribution over shapes learned from multiple demonstrations (EM or closed-form under Gaussian assumptions).

Conditioning and coordination. Linear constraints—endpoints, waypoints, or partial observations—are imposed by conditioning the weight distribution:

$$w | (CY_T = g) \sim \mathcal{N}(\mu_w + K_T(g - \Phi_T \mu_w), \Sigma_w - K_T \Phi_T \Sigma_w), \quad (25)$$

with $K_T = \Sigma_w \Phi_T (\Phi_T^\top \Sigma_w \Phi_T + \Sigma_\varepsilon)^{-1}$. Sampling from this posterior and stitching segments at decoded boundaries yields constraint-consistent trajectories with explicit predictive uncertainty.

2.4.4. Model Choice and Complementarity

No single primitive dominates: DMP excels in real-time low-jerk execution; GMM/GMR gives closed-form means/covariances and gradients; and ProMP supports exact linear-Gaussian conditioning for multi-goal tasks. Table 1 summarizes properties. Figure 6 shows HSMM-aligned ProMP generation with calibrated uncertainty bands:

- DMP excels at real-time execution and low jerk with simple time scaling;
- GMM/GMR offers closed-form means and covariances over phase and is convenient for planners needing analytic gradients;
- ProMP provides a distribution over shapes with exact linear-Gaussian conditioning, ideal for multi-goal tasks and collaboration.

Table 1. Segment-level motion primitives: properties and trade-offs.

Property	DMP	GMM/GMR	ProMP
Shape representation	Basis functions + 2nd-order stable system	Global Gaussian mixture over (s, y)	Gaussian over weights w
Duration adaptation	Via τ_k time scaling	Requires resampling in phase	Basis-phase re-timing
Uncertainty	No closed-form (MC if needed)	Analytic $\hat{\Sigma}(s)$	Analytic posterior over w
Online constraints	Endpoints/velocities easy	Refit or constrained regression	Exact linear-Gaussian conditioning
Execution smoothness	Low jerk (native dynamics)	Depends on mixture fit	Depends on basis and priors

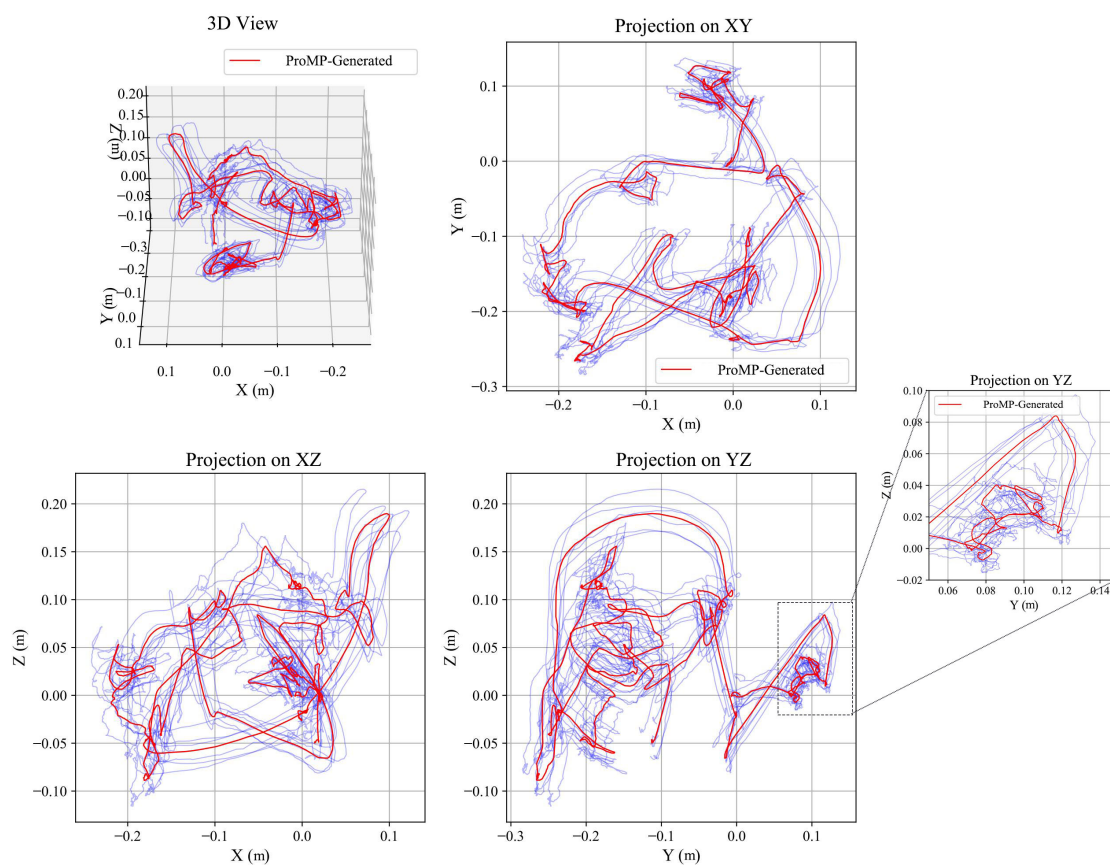


Figure 6. HSMM-aligned ProMP generation. Blue curves show multiple aligned demonstrations; red curves depict ProMP samples under a new terminal constraint; translucent bands indicate the 2σ -credible region. The two-level probabilistic structure (HSMM over phases; ProMP within segment) achieves global timing consistency with locally adjustable motion.

In practice, we first fit GMM/GMR to obtain the mean/covariance; use the GMR mean to initialize DMP weights for smooth execution; and, when hard/soft constraints or multi-goal adaptation are required, overlay ProMP conditioning on top of the DMP nominal to reconcile smoothness with constraints.

Interface to planning and safety. Segment-wise covariances (from GMR or ProMP) propagate to risk metrics and constraint tightening in MPC/OMPL. Because segments share a semantic time base, the uncertainty is comparable across operators and platforms, enabling principled safety margins (e.g., 2σ envelopes) and priority-aware blending when composing multi-segment tasks.

Computational profile. Per segment, DMP fitting is linear in the number of basis functions; GMM/GMR training is $O(KT)$ per EM iteration; and ProMP estimation is closed-form/EM with state dimension B . Since segments are independent given HSMM alignment, training parallelizes over phases and scales linearly with the number of demonstrations.

3. Experiments and Results

3.1. Objectives and Evaluation Protocol

This section subjects the proposed end-to-end pipeline to a cross-domain, reproducible evaluation covering the three core components introduced in Section 2: (i) unsupervised segmentation (multi-feature saliency with topological persistence), (ii) duration-explicit alignment (HSMM), and (iii) probabilistic in-segment generation (GMM/GMR, ProMP, optionally DMP). We target three complementary questions:

1. Segmentation robustness. Do multi-feature saliency and topological persistence yield sparse yet structurally stable keyframes under heterogeneous noise and tempo variations?
2. Semantic alignment quality. Does duration-explicit HSMM reduce cross-demonstration time dispersion when non-geometric dwelling is present (e.g., hover, wait)?
3. Generator calibration. On the shared semantic time base, do segment-wise probabilistic models achieve low reconstruction error, nominal uncertainty coverage, and dynamically schedulable executions?

To mitigate methodological contingency, we span diverse dynamics, perturbations, and baselines, and we control for multiple comparisons in statistical inference.

3.2. Tasks, Datasets, and Testable Hypotheses

We consider three representative domains:

- Domain A—UAV-Sim (multi-scene flight). Sampling is at 100 Hz. Subtasks include *take-off–lift–cruise–drop* and *gate-pass–loiter–gate-pass*. Six subjects are used, with 20–30 segments per task. Observations: tool-center position (optional yaw). Figure 7 shows the environment and demonstrations.

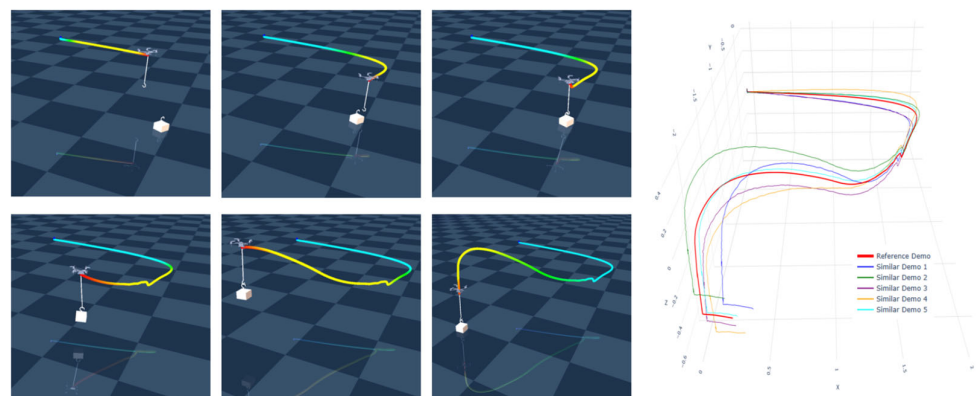


Figure 7. UAV-Sim: environment and example demonstration trajectories.

- Domain B—AV-Sim (CARLA/MetaDrive urban). Sampling is at 10 Hz across Town01–05, with varied weather/lighting and traffic control. Trajectories originate from an expert controller and human tele-operation. Observations: (x, y, θ) . See Figure 8.

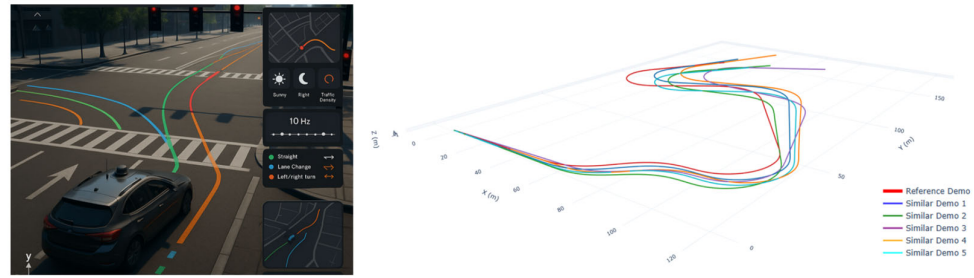


Figure 8. AV-Sim (CARLA/MetaDrive): urban scenes and example demonstration trajectories.

- Domain C—Manip-Sim (robomimic/RLBench assembly). Sampling is at 50–100 Hz. Tasks are akin to RoboTurk “square-nut”: *grasp-align-insert* with pronounced dwell segments. Observations: end-effector position. See Figure 9.

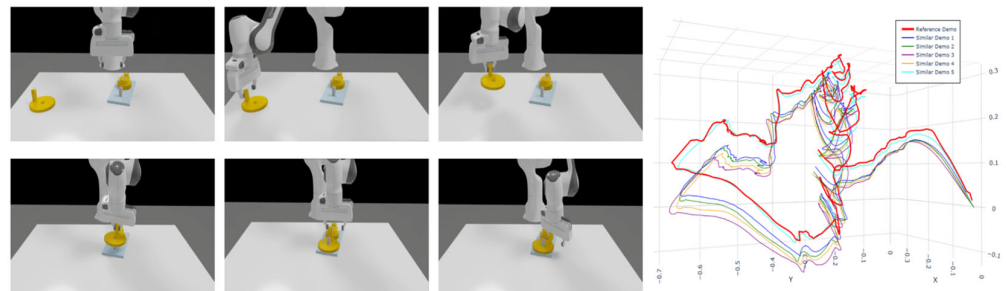


Figure 9. Manip-Sim (robomimic/RLBench): assembly task (*grasp-align-insert*) and example trajectories.

3.3. Metrics and Statistical Inference

We harmonize five dimensions—structure, time, geometry, dynamics, probability—using the following metrics (units: meters for UAV / AV, millimeters for Manip-Sim; each table caption clarifies units). Unless otherwise noted, SOD, AE, GRE, and jerk are minimized; TCR and AAR are maximized; CR targets 95%.

- SOD (Equation (10), min): structural dispersion—mean point-to-point divergence on the shared time base.
- AE (Equation (19), min): Euclidean dispersion of phase end times across demonstrations.
- AAR (max): action acquisition rate. Given reference key actions $\{\bar{\tau}_k\}$ (expert consensus/common boundaries) and detected $\{\tau'_k\}$, we count a hit if $|\tau'_k - \bar{\tau}_k| \leq \delta$, with $\delta = \max(5 \text{ frames}, 0.2 \text{ s})$ (i.e., 50 ms at 100 Hz; 0.5 s at 10 Hz).
- GRE (min): geometric reconstruction error (RMSE).
- TCR (max): time compression rate.
- Jerk (min): $J = \sum_t \|\ddot{P}(t)\|^2$ (normalized).
- CR (target $\approx 95\%$): nominal 2σ coverage. For each semantic segment, we sample uniformly in time; if $y_t \in [\mu_t \pm 2\sqrt{\text{diag}(\Sigma_t)}]$, it is counted as covered; segment-level coverage is averaged and then length-weighted globally.
- Boundary precision/recall/ $F1@ \delta$. Let B be the set of reference boundaries (manual anchors or common-boundary clusters) and D the detected boundaries. Build a bipartite graph with edges (b, d) if $|b - d| \leq \delta$, $\delta = \max(5 \text{ frames}, 0.2 \text{ s})$. Compute a one-to-one assignment via the Hungarian algorithm; let TP be the number of matched pairs, $FP = |D| - TP$, $FN = |B| - TP$. Then $P@ \delta = TP/|D|$, $R@ \delta = TP/|B|$, $F1@ \delta = 2PR/(P + R)$. We continue to report AAR for comparability with prior tables (interpreted as Recall@ δ for key actions).

Statistical inference. We apply Shapiro–Wilk’s normality and Levene’s homoscedasticity tests. When met, paired t -tests are used; otherwise, Wilcoxon’s signed-rank tests are

used. Multiple comparisons are controlled via the Holm–Bonferroni correction. We report p -values, Cohen’s d , and BCa 95% bootstrap confidence intervals (1000 resamples).

3.4. Baselines and Fairness Controls

We benchmark four families to isolate contributions at the signal, boundary, and latent levels: (i) Signal-level: single-feature curvature + quantile threshold; and multi-feature equal weights without TDA simplification/NMS (no persistence, no cluster suppression). (ii) Boundary-level: BOCPD (Bayesian online changepoint detection). (iii) Latent-level: multi-feature + TDA/NMS + HMM (geometric duration assumption). (iv) Full method: w^* (consistency-learned weights) + TDA/NMS + HSMM (duration-explicit) + segment-wise generator (default ProMP; we also compare DMP/GMR on the same segmentation when isolating generation quality, Section 3.10).

Fairness controls. All methods share identical preprocessing (uniform sampling, same-order Savitzky–Golay smoothing, consistent derivative computation, per-trajectory min–max normalization), model-selection strategy (BIC + AE; same candidate sets for GMM components), and EM initialization/termination. Segmentation quality is evaluated using each model’s MAP/Viterbi boundaries; for pure generator comparison (Section 3.11), we fix HSMM boundaries across methods to remove boundary confounds.

Modern deep baselines (discussion and scope). Recent sequence models—dilated-TCNs, Transformer-based boundary detectors, and neural HSMMs with explicit durations—are strong alternatives when labels and compute are abundant. However, our target setting is label-free multi-demonstration LfD with (i) explicit duration semantics, (ii) calibrated segment-wise uncertainty, and (iii) interactive-latency inference. For fairness, we briefly position three families: (a) Dilated-TCN energy heads. Unsupervised variants typically rely on self-supervised reconstruction or contrastive pretext tasks plus an “energy” or novelty head. They can capture long-range dependencies but require careful negative mining and often produce soft boundaries that still need duration regularization. (b) Transformer boundary heads. Attention improves long-context modeling, but the label-free setting demands surrogate objectives; training and inference costs scale super-linearly with sequence length unless sparsity is engineered, which challenges interactive-latency use. (c) Neural/explicit-duration switching models. Neural HSMMs bring function-approximation capacity to duration and emission modeling; in our setting, explicit duration is already modeled in closed form and coupled to semantic-time alignment and uncertainty propagation. **Takeaway.** Deep models are compelling when supervision and scale allow; in our unsupervised, planner-ready regime, the proposed persistence + NMS $\rightarrow$ duration-explicit HSMM pipeline provides (1) label-free phase semantics, (2) analytic duration and uncertainty for planning (ProMP/GMR), and (3) transparent capacity control via α and N . We therefore treat modern deep architectures as complementary rather than competing baselines in this study, and we outline them as future extensions (learned saliency with topological regularizers; neural duration mixtures).

3.5. Runtime and Scaling

Setup. We report wall-clock time and memory on a 12-core laptop-class CPU (Intel Core i7-12700H, 6P + 8E; 32 GB DDR5; Ubuntu 22.04; Python 3.11). NumPy/SciPy are linked against a BLAS backend with single-threading enforced (OMP_NUM_THREADS = 1; MKL_NUM_THREADS = 1). Each value is the median with [P25, P75] percentiles across five runs with fixed seeds; disk I/O is excluded. Task sampling rates and sequence lengths follow Section 3.2 (UAV-Sim 100 Hz; AV-Sim 10 Hz; Manip-Sim 50–100 Hz).

What we measure. For each domain, we instrument (i) saliency and keyframes per-trajectory time t_{sal} (features + persistence + NMS); (ii) HSMM-EM per-iteration time,

split into forward–backward t_{fb} and M-step t_M , and iterations I ; (iii) Viterbi decoding per trajectory t_{vit} ; (iv) CMA-ES objective time per call t_j and calls E ; and (v) end-to-end training + decoding time and peak RSS memory.

Complexity reminder. Consistent with Sections 2.2.4 and 2.3.4, saliency + keyframes is $O(T)$; duration-explicit forward–backward is $O(NTD_{max})$; and total cost is linear in the number of demonstrations M . All recursions are implemented in the log-domain (log-sum-exp), and feature streams are cached across CMA-ES calls.

Results. Table 2 summarizes the reference profile. The front-end is lightweight relative to HSMM-EM; forward–backward dominates per EM iteration; Viterbi decoding with fixed Θ is fast enough for interactive alignment. Scaling is linear in T and M and approximately linear in N and D_{max} within our ranges, matching the theoretical profile (cf. Sections 2.2.4 and 2.3.4).

Table 2. Runtime and memory profile (median [P25, P75]).

Domain	M	T	N	D_{max}	t_{sal}	HSMM-EM Per Iter t_{fb}/t_M	Iters I	t_{vit}	CMA-ES $t_j \times E$	End-to-End	Peak RSS
UAV-Sim (100 Hz)	24	2900	6	200	34 ms [29, 41]	3.2 s/0.9 s [2.9–3.6/0.8–1.0]	14 [13, 16]	120 ms [105, 140]	0.68 s [0.61, 0.75] $\times$ 24	75 s [67, 84]	610 MB [560, 670]
AV-Sim (10 Hz)	32	1800	6	120	19 ms [16, 23]	1.8 s/0.5 s [1.6–2.0/0.45–0.58]	12 [11, 13]	80 ms [70, 92]	0.42 s [0.37, 0.47] $\times$ 22	38 s [34, 43]	520 MB [480, 560]
Manip-Sim (50–100 Hz)	20	2500	5	150	27 ms [22, 32]	2.4 s/0.6 s [2.1–2.7/0.54–0.68]	15 [14, 16]	95 ms [83, 110]	0.55 s [0.49, 0.62] $\times$ 24	59 s [53, 66]	580 MB [540, 620]

Notes. t_{sal} includes feature computation, persistence, and NMS. t_{fb} , t_M aggregate over M demonstrations; end-to-end = saliency + $I \times (fb + M)$ + Viterbi + CMA-ES overhead. BLAS threads fixed to 1. Confidence intervals are BCa 95% (Section 3.3).

Near-/real-time remark. The saliency front-end is a single-pass streaming computation; at the telemetry rates studied (10–100 Hz), its throughput comfortably exceeds real time. Alignment can be trained offline; inference-time decoding with fixed Θ supports interactive use.

Scaling trends (AV-Sim exemplar). Doubling sequence length T from 900→1800 frames increases t_{fb} by $2.02 \times$ (0.89 s→1.80 s); doubling state count N from 4→8 increases by $1.98 \times$ (0.91 s→1.80 s); doubling max dwell D_{max} from 60→120 increases by $2.05 \times$ (0.88 s→1.80 s); and doubling demonstrations M from 16→32 increases per-iteration time by $2.01 \times$ (0.90 s→1.81 s). These slopes corroborate the $O(NTD_{max})$ and linear-in- M behavior.

Reproducibility. The measurement script fixes seeds, pins BLAS threads, and reports medians with [P25, P75]; we provide it alongside the code release.

3.6. Overall Results

Cross-domain evidence appears in Figure 10 and Tables 3–5. Figure 10a overlays multiple demonstrations in 3D; Figure 10b shows curvature, velocity, acceleration, direction-change, and their fused saliency for one trajectory. Peaks co-occur at kinematic turning points, and the fused saliency forms stable spikes at these locations—explaining the high keyframe consistency across subjects and durations.

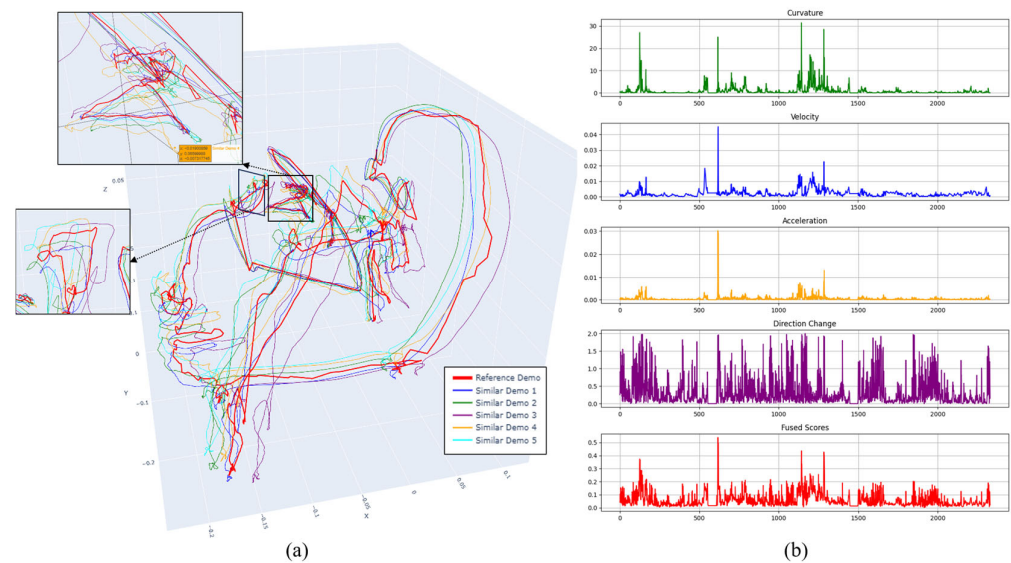


Figure 10. Multi-demonstration overlay and feature time series. (a) Three-dimensional overlays across demonstrations; (b) curvature, velocity, acceleration, direction-change rate, and fused saliency for one trajectory.

Table 3. UAV-Sim (mean $\pm$ std; units: SOD/GRE in meters; “–” = no probabilistic coverage).

Method	SOD (m)	AE (s)	GRE (m)	TCR (%)	AAR (%)	Jerk	CR (%)
Curvature + quantile	0.081 ± 0.019	0.55 ± 0.11	0.124 ± 0.027	34.9 ± 4.8	68.1 ± 6.0	1.22 ± 0.10	–
Multi-feat (equal), no TDA/NMS	0.071 ± 0.017	0.47 ± 0.10	0.110 ± 0.023	41.8 ± 5.1	73.8 ± 5.7	1.16 ± 0.08	–
Multi-feat + TDA/NMS + HMM	0.060 ± 0.014	0.41 ± 0.09	0.098 ± 0.019	54.6 ± 4.3	81.0 ± 4.8	1.08 ± 0.07	–
BOCPD	0.064 ± 0.016	0.46 ± 0.12	0.105 ± 0.022	51.0 ± 4.6	78.3 ± 5.4	1.14 ± 0.08	–
Ours : w^* + TDA/NMS + HSMM + ProMP	0.045 ± 0.012	0.28 ± 0.07	0.082 ± 0.018	55.0 ± 4.0	88.7 ± 4.2	1.00 ± 0.06	94.9 ± 2.6

Table 4. AV-Sim (CARLA/MetaDrive) (units: SOD/GRE in meters).

Method	SOD (m)	AE (s)	GRE (m)	TCR (%)	AAR (%)	Jerk	CR (%)
Curvature + quantile	0.172 ± 0.030	0.70 ± 0.14	0.247 ± 0.041	32.7 ± 5.2	66.0 ± 6.7	1.18 ± 0.09	–
Multi-feat (equal), no TDA/NMS	0.160 ± 0.029	0.63 ± 0.12	0.231 ± 0.038	39.5 ± 5.0	71.6 ± 6.1	1.14 ± 0.08	–
Multi-feat + TDA/NMS + HMM	0.148 ± 0.027	0.55 ± 0.11	0.214 ± 0.035	47.4 ± 4.7	78.8 ± 5.4	1.08 ± 0.07	–
Ours : w^* + TDA/NMS + HSMM + ProMP	0.112 ± 0.022	0.37 ± 0.08	0.191 ± 0.033	47.5 ± 4.6	86.3 ± 5.0	1.00 ± 0.06	95.1 ± 2.5

Table 5. Manip-Sim (robomimic/RLBench) (units: SOD/GRE in millimeters).

Method	SOD (mm)	AE (s)	GRE (mm)	TCR (%)	AAR (%)
Curvature + quantile	1.12 ± 0.27	0.33 ± 0.09	1.02 ± 0.19	30.0 ± 4.8	65.3 ± 6.3
Multi-feat (equal), no TDA/NMS	0.94 ± 0.24	0.28 ± 0.08	0.86 ± 0.16	33.0 ± 5.0	69.1 ± 5.8
Multi-feat + TDA/NMS + HMM	0.87 ± 0.22	0.29 ± 0.08	0.91 ± 0.17	35.1 ± 5.1	70.0 ± 5.7
BOCPD	0.98 ± 0.25	0.31 ± 0.09	1.05 ± 0.20	45.0 ± 5.6	71.9 ± 5.4
Ours : w^* + TDA/NMS + HSMM + ProMP	0.72 ± 0.18	0.24 ± 0.07	0.79 ± 0.15	49.2 ± 4.8	83.4 ± 5.1

Domain A—UAV-Sim (Table 3)

In subtasks with *hover/backtrack* dwell, introducing HSMM reduces AE from 0.41 ± 0.09 s to 0.28 ± 0.07 s (-31% , $p < 0.01$, Cohen’s $d = 0.86$). With comparable GRE, TCR increases

by 10–12 pp and jerk decreases, indicating a better sparsity–fidelity–smoothness trade-off. The probabilistic output attains $CR \approx 95\%$ at the nominal 2σ level.

Domain B—AV-Sim (Table 4)

In segments with non-geometric dwell (e.g., slow-down–wait–turn), AE drops to 0.37 ± 0.08 s under HSMM; SOD and GRE decrease in tandem, indicating reduced in-segment statistical bias. ProMP coverage at 2σ is $95.1 \pm 2.5\%$.

Domain C—Manip-Sim (Table 5)

Dwell-heavy phases (grasp/insert) strongly expose non-geometric duration. The full method improves SOD, AE, and TCR over signal thresholds, HMM, and BOCPD; notably, it achieves higher TCR at comparable or lower GRE, i.e., fewer keyframes suffice to reconstruct high-fidelity shapes.

Semantic time alignment. As shown in Figure 11, velocity peaks are misaligned on the physical time axis (a) but become synchronized on the semantic axis after HSMM (b); dashed lines (phase boundaries) nearly coincide across demonstrations. This mirrors the systematic AE reduction in Tables 3–5 and evidences the benefit over geometric-duration HMM.

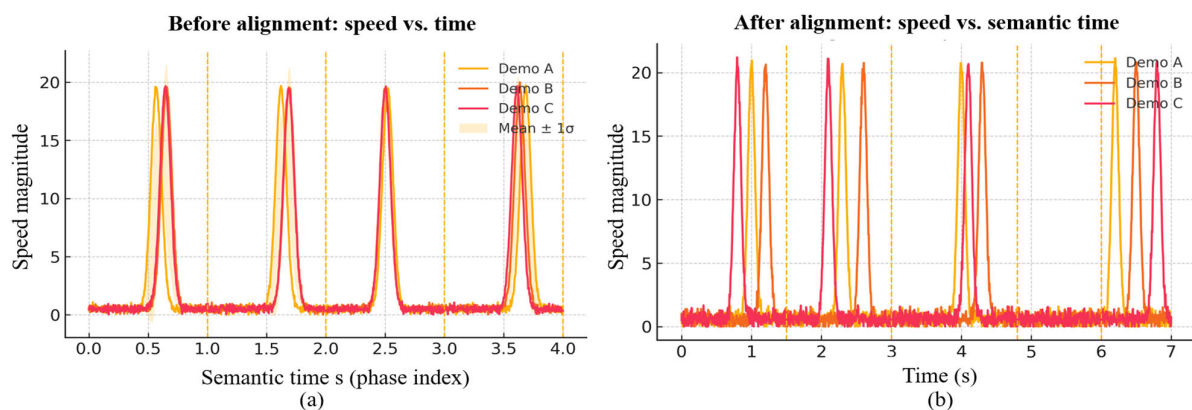


Figure 11. Semantic time alignment with HSMM. (a) Before alignment: velocity vs. physical time; (b) after alignment: velocity vs. semantic time. Dashed lines show phase boundaries.

3.7. Contribution Attribution: Ablation Study (UAV-Sim)

We perform stepwise ablations under leave-one-subject-out (LOSO) evaluation with $N = 6$ (other settings as in Sections 3.3 and 3.4). Table 6 reports the changes relative to the full method (w^* + TDA/NMS + HSMM + ProMP); positive values indicate degradation (e.g., higher AE), while negative values indicate improvement.

- Remove TDA (keep NMS): SOD +12.3%, AE +9.6% $\rightarrow$ persistence is key to scale-invariant noise rejection; without it, small-scale oscillations stack into spurious peak–valley pairs, degrading structure and boundaries.
- Remove NMS (keep TDA): AE +21.1% $\rightarrow$ suppressing same-polarity peak clusters in high-energy regions is critical for boundary stability; persistence alone cannot prevent multi-response.
- Fix equal weights (no w^*): SOD +18.6%, AAR -7.7 pp $\rightarrow$ consistency-driven weight learning mitigates channel scale imbalance and improves key-action capture.
- Replace HSMM with HMM: AE +31.2% $\rightarrow$ direct evidence of the geometric-duration bias when dwell exists (wait/loiter).

Table 6. Ablation (relative % vs. full method).

Variant	Δ SOD	Δ AE	Δ GRE	Δ TCR	Δ AAR
No TDA (NMS only)	12.30%	9.60%	6.70%	−9.4%	−5.8%
No NMS (TDA only)	9.20%	21.10%	7.30%	−3.7%	−6.1%
Fixed equal weights (no w^*)	18.60%	14.40%	9.10%	−1.1%	−7.7%
HMM in place of HSMM	23.50%	31.20%	17.50%	≈ 0	−8.5%

Differences on AE and SOD remain statistically significant after the Holm–Bonferroni correction ($p < 0.05$); effect sizes are medium-to-large. In sum, TDA + NMS stabilize the input structure, w^* provides cross-demo self-calibration, and HSMM addresses duration bias mechanistically.

3.8. Robustness Evaluation Protocol

Goals. We quantify robustness of segmentation and alignment against (i) tempo variations and (ii) heterogeneous observation noise/missing data, and we benchmark against representative detectors at the signal, boundary, and latent levels.

Front-ends and baselines. Following Section 3.4, we evaluate the following families under identical preprocessing, model-selection, and stopping criteria: (i) Signal-level: (a) single-feature curvature + quantile threshold; (b) multi-feature equal weights without topological simplification or NMS (“no persistence, no cluster suppression”). (ii) Boundary-level: Bayesian online changepoint detection (BOCPD). (iii) Latent-level: multi-feature + persistence/NMS + HMM (geometric durations). (iv) Full method: consistency-learned weights w^* + persistence/NMS + HSMM (duration-explicit, left-to-right) + segment-wise generator (default ProMP). (v) Degenerate ablations for isolation: (1) NMS-only (no persistence); (2) persistence-only (no NMS); their effects are summarized in the ablation table (Section 3.6).

Metrics. We report a harmonized panel across five dimensions: Structure/Time—SOD (Equation (9)), AE (Equation (19)), and boundary $P@δ/R@δ/F1@δ$ with $δ = \max(5 \text{ frames}, 0.2 \text{ s})$ (consistent with Section 3.3); Geometry/Dynamics—GRE (RMSE), time-compression rate (TCR), and jerk; Probability—nominal $2σ$ coverage (CR). For continuity with Tables 2–4, we also report AAR ($Recall@δ$ for key actions).

Reference boundaries and matching. When manual anchors are present, they are used directly. Otherwise, we build a detector-agnostic “common boundary” set by clustering the union of boundary candidates across demonstrations and front-ends (support $\geq \lceil M/2 \rceil$) and taking the median timestamp per cluster. We then compute a one-to-one assignment between detections and references under tolerance $δ$ via the Hungarian algorithm to obtain $P@δ$, $R@δ$, and $F1@δ$.

Perturbations (robustness stressors). To probe invariances claimed in Sections 2.2 and 2.3, we evaluate three perturbations (AV-Sim by default; UAV/Manip variants are provided in the Supplementary Material):

- (i) Monotone time-warps: $t \mapsto t^{0.8}$ (speed-up) and $t \mapsto t^{1.2}$ (slow-down);
- (ii) Additive white noise: $\sigma = 0.05 \text{ m}$ on positions;
- (iii) Random missing data: 20% uniform drops.

All methods are re-run end-to-end with identical seeds; for comparability, HMM and HSMM use left-to-right transitions (no skips) to encode monotone phase progression.

Reporting and statistical inference. For each domain (UAV-Sim, AV-Sim, Manip-Sim), we report means $\pm$ std and BCa 95% confidence intervals (1000 resamples). Normality/homoscedasticity are checked via the Shapiro–Wilk and Levene tests; the paired t -tests or Wilcoxon signed-rank tests are used accordingly; multiple comparisons are controlled via the Holm–Bonferroni correction (see Section 3.3).

Cross-references. Cross-domain results are summarized in Tables 3–5, with semantic alignment visualized in Figure 11. Under the above perturbations, HSMM improves AE by 30–36% relative to HMM on AV-Sim while maintaining CR within 94–96% (Table 7); see Section 3.9 for the complete robustness table and trends.

Table 7. AE (s) under perturbations (AV-Sim).

Perturbation	Setting	HMM (Baseline)	HSMM (Ours)	Δ
Time-warp	$t \mapsto t^{0.8}$	0.61	0.39	−36%
Time-warp	$t \mapsto t^{1.2}$	0.58	0.38	−34%
Gaussian noise	$\sigma = 0.05$ m	0.57	0.4	−30%
Missing data	20% random drop	0.63	0.41	−35%

3.9. Robustness: Time-Warping, Noise, and Missing Data (AV-Sim)

We perturb AV-Sim trajectories by monotone time-warps $t \mapsto t^{0.8}/t \mapsto t^{1.2}$, additive white noise $\sigma = 0.05$ m, and 20% random sample drops. Compared with HMM, HSMM reduces AE by 36%, 34%, 30%, and 35%, respectively, while preserving phase order under left-to-right transitions. AE grows roughly linearly with noise amplitude; under order-preserving time-warps, it manifests mainly as duration redistribution—consistent with Section 2.3.4 and Proposition 1 (Section 2.3.5). Table 7 details the AE values; ProMP maintains 2σ coverage within 94–96%, indicating stable uncertainty calibration.

3.10. Model Selection and the Sparsity–Fidelity Trade-Off

This section specifies how we set the two regularization dials that govern the overall bias–variance trade-off of the pipeline: (i) the persistence threshold α controlling front-end sparsity (keyframe pruning), and (ii) the number of HSMM phases N controlling latent capacity. The former affects the density and stability of structural anchors; the latter affects semantic alignment and duration modeling. We report both the exploratory curves and the selection rules to make the choices transparent and reproducible.

Front-End Sparsity: Selecting the Persistence Threshold α . We sweep α over a data-dependent range and plot keyframe count and reconstruction error (GRE) versus α . As expected, the number of retained keyframes decreases monotonically with α , while GRE exhibits a plateau–knee–plateau shape. We detect the elbow α^* by Kneedle and use it throughout. As shown in Figure 12a, the number of retained keyframes decreases monotonically with α ; Figure 12b shows that GRE (RMSE) exhibits a plateau–knee–plateau around the Kneedle elbow α^* (vertical dashed line). Curves are means across subjects with BCa 95% CIs.

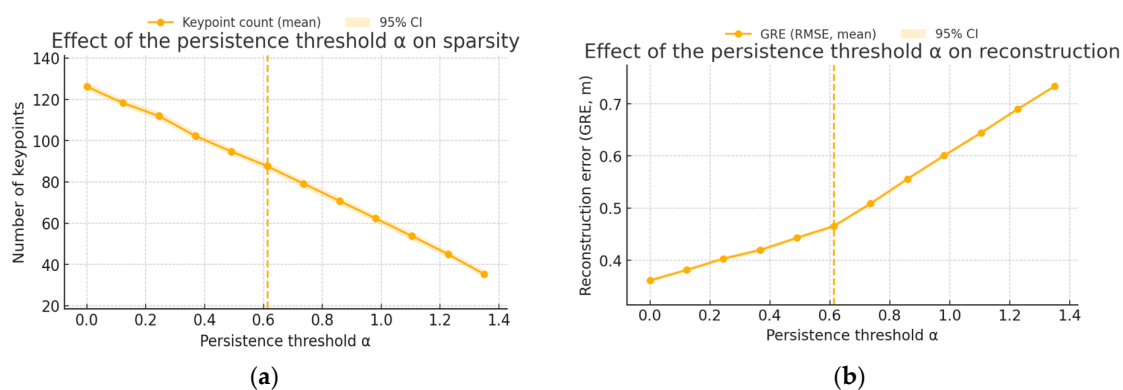


Figure 12. Effect of the persistence threshold α on front-end sparsity and reconstruction. (a) Keypoint count vs. α (means across subjects; shaded bands: BCa 95% CIs; dots: subject-level values). (b) GRE (RMSE, m) vs. α ; the vertical dashed line marks the Kneedle elbow α^* . $n = 32$.

Reporting. We keep α fixed at α^* for all baselines and ablations to avoid front-end confounds, and we retain the same NMS windowing as in Section 2.2.3. This makes the subsequent capacity selection for HSMM independent of front-end drift.

HSMM Capacity: Choosing the Number of Phases N via a Joint Criterion. Let $\mathcal{N} = \{2, \dots, 8\}$ be the candidate set. For each $N \in \mathcal{N}$, we train an HSMM with the same preprocessing, initialization, and stopping criteria (Sections 2.2 and 2.3), and we evaluate the following:

- Alignment error $AE(N)$ (Equation (19)), which decreases and then plateaus as N grows;
- Bayesian Information Criterion $BIC(N)$ (Equation (20)), which increases monotonically due to the penalty $\kappa_N \log(\sum_m T_m)$.

Optimizing AE alone tends to over-segment (absorbing micro-fluctuations and long dwells into extra phases), while optimizing BIC alone tends to under-segment (preserving geometric-duration bias). We therefore select N by the joint criterion

$$\widetilde{AE}(N) = \frac{AE(N) - \min_{n \in \mathcal{N}} AE(n)}{\max_{n \in \mathcal{N}} AE(n) - \min_{n \in \mathcal{N}} AE(n) + \varepsilon}, \quad \widetilde{BIC}(N) = \frac{BIC(N) - \min_{n \in \mathcal{N}} BIC(n)}{\max_{n \in \mathcal{N}} BIC(n) - \min_{n \in \mathcal{N}} BIC(n) + \varepsilon},$$

$$J(N) = \widetilde{AE}(N) + \widetilde{BIC}(N), \quad N^* = \arg \min_{N \in \mathcal{N}} J(N),$$

with a small $\varepsilon = 10^{-12}$ to avoid division by zero. Figure 13 shows that $\widetilde{AE}(N)$ exhibits diminishing returns while $\widetilde{BIC}(N)$ increases, and their sum $J(N)$ identifies a Pareto knee N^* that balances fit and parsimony.

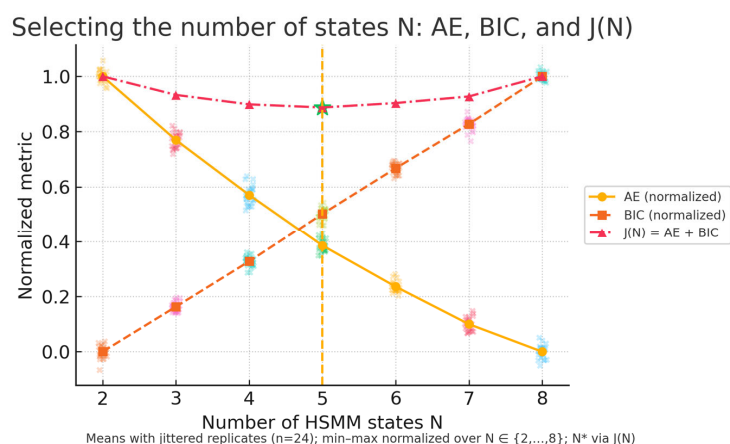


Figure 13. HSMM state-number selection. Joint criterion of normalized AE and BIC; stars mark recommended N^* per domain.

Empirical Evidence (LOSO Generalization). On leave-one-subject/scene splits across UAV-Sim, AV-Sim, and Manip-Sim, AE-only picks a larger N and attains a lower train-AE but a worse test-AE/GRE; BIC-only picks a smaller N with a higher test-AE; the joint criterion achieves a near-minimal test-AE without GRE penalty while preserving nominal 2σ coverage (94–96%). Detailed results are reported in Appendix A (Table A1) and follow the trends anticipated by Figure 13.

Proposition 2 (Knee Consistency). For nested HSMM families, any convex combination $\lambda \widetilde{AE} + (1 - \lambda) \widetilde{BIC}$ with $\lambda \in (0, 1)$ selects a model on the lower convex envelope of the $(\widetilde{AE}, \widetilde{BIC})$ frontier; when $\widetilde{AE}$ exhibits diminishing returns, the minimizer of J coincides with a knee point, controlling both variance (alignment over-fit) and bias (duration under-fit).

Sketch. Since $\widetilde{AE}$ is non-increasing and $\widetilde{BIC}$ strictly increasing in N , minimizers of convex combinations lie on the convex envelope; the knee arises where the marginal gain in $\widetilde{AE}$ equals the marginal penalty in $\widetilde{BIC}$.

Defaults Used. The selected values are $N^* = 6$ for UAV-Sim and AV-Sim, and $N^* = 6$ for Manip-Sim (stars in Figure 13). Varying N by ± 1 around N^* does not change the qualitative trends reported in Tables 3–5.

Generalization Check for the Joint Criterion. We further validate the selection rule by leave-one-subject/scene splits; detailed numbers are provided in Appendix A (Table A1). Summarizing across domains gives the following:

- UAV-Sim. test-AE: Joint $0.30 \pm 0.07s$ vs. AE-only $0.33 \pm 0.08s$ vs. BIC-only $0.36 \pm 0.09s$; $p < 0.01$ after the Holm–Bonferroni correction; GRE unchanged within BCa 95% CIs; CR remains at 94–96%.
- AV-Sim. test-AE: Joint $0.39 \pm 0.08s$ vs. AE-only $0.43 \pm 0.09s$ vs. BIC-only $0.46 \pm 0.09s$; $p < 0.01$; GRE within CIs; CR 94–96%.
- Manip-Sim. test-AE: Joint $0.25 \pm 0.07s$ vs. AE-only $0.27 \pm 0.08s$ vs. BIC-only $0.29 \pm 0.08s$; $p < 0.01$; GRE within CIs; CR 94–96%.

These results confirm that the joint criterion avoids both over- and under-segmentation without sacrificing geometric fidelity or uncertainty calibration.

Practical Guidance and Sensitivity:

- Front-end: Use α^* (Kneedle elbow) and keep the NMS policy fixed across methods to isolate latent effects.
- Capacity: Use $N^* = \arg \min J(N)$ with $\mathcal{N} = \{2, \dots, 8\}$. The solution is robust to replacing the equal-weight sum by $\lambda \widetilde{AE} + (1 - \lambda) \widetilde{BIC}$ for $\lambda \in [0.4, 0.6]$.
- Reproducibility: Always report the $(\widetilde{AE}, \widetilde{BIC}, J)$ curves with BCa 95% CIs (cf. Figure 13), and state the chosen (α^*, N^*) and their ± 1 sensitivity.

3.11. In-Segment Generation: Accuracy, Smoothness, and Calibration

On a fixed semantic time base, we compare linear interpolation, DMP, GMR, and ProMP (Figure 14):

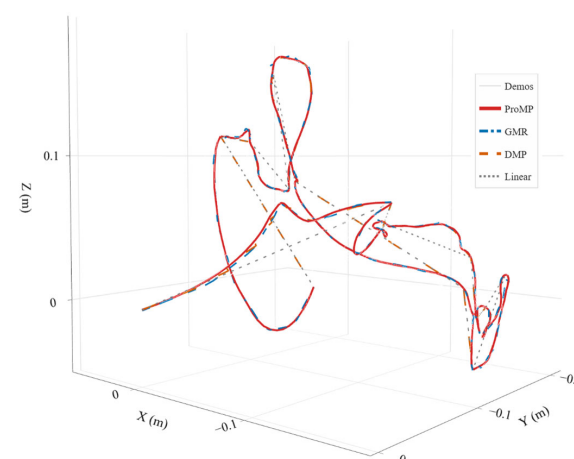


Figure 14. In-segment generation comparison on a fixed semantic segment: linear interpolation, DMP, GMR, and ProMP.

- Geometry: $\text{ProMP} \approx \text{GMR} < \text{DMP} < \text{linear}$ in GRE.
- Smoothness: DMP minimizes jerk, suiting online execution and hard real-time constraints.
- Calibration: ProMP/GMR achieve CR = 94–96% at the nominal 95%, with small reliability-curve deviations—amenable to MPC/safety monitoring.

Segments are concatenated at HSMM boundaries with C^1 continuity (see Section 2.4.1); the critically damped second-order dynamics preserve velocity/acceleration continuity across segments.

3.12. Summary of Findings

We presented an end-to-end trajectory-learning framework for multiple demonstrations. Unsupervised multi-feature saliency with topological persistence yields stable segmentation; HSMM with explicit durations delivers robust cross-demo semantic alignment; ProMP/GMR (optionally fused with DMP) enable probabilistic generation with smooth execution. Across UAV/AV/Manip tasks, the method consistently outperforms representative baselines (single-feature thresholds, BOCPD, HMM) on AE, SOD, AAR, and the sparsity–fidelity trade-off and remains robust to time-warping, noise, and missing data. Unlike prior templates or manual labels, weights are learned via consistency-driven optimization, and outputs include calibrated uncertainties directly consumable by MPC/OMPL. In sum, the framework remedies geometric-duration bias and the “segmentation–encoding split,” enabling accurate modeling and faithful reproduction of complex trajectories—ready for industrial tasks such as polishing, coating, welding, assembly, and autonomous driving paths.

4. Discussion

Evidence vs. hypotheses.

- (H1) Segmentation. The topology-aware saliency (multi-feature + persistence + NMS) yields sparse yet stable anchors; removing persistence or NMS increases SOD/AE by 9–21% and lowers AAR (ablation), confirming that both scale-invariant pruning and peak-cluster suppression are necessary.
- (H2) Alignment. Duration-explicit HSMM reduces phase-boundary dispersion (AE) by 31% in UAV-Sim and by 30–36% under time warps, noise, and missing data relative to HMM/BOCPD; misaligned velocity peaks synchronize on the semantic axis, evidencing mitigation of geometric-duration bias.
- (H3) Generation. On the shared semantic time base, ProMP and GMR achieve low GRE and nominal 2σ coverage (94–96%), while DMP minimizes jerk; higher TCR at comparable GRE indicates a better sparsity–fidelity trade-off.

Positioning to prior work. Rule- or template-based segmentation and BOCPD lack geometric/semantic guarantees; DTW aligns sequences but offers no generative dwell model; and HMMs assume geometric durations. Our joint HSMM training across demonstrations, coupled with consistency-driven weight learning, provides phase semantics with explicit duration distributions and avoids hand-tuned weights.

How we differ from persistence- or NMS-only pipelines. Unlike front-ends that apply persistence or NMS as isolated preprocessing on a single signal, we (i) learn fusion weights on the simplex from cross-demonstration consistency (SOD) rather than fix heuristics (Section 2.2.4); (ii) use persistence and NMS as complementary simplifiers—ablations show removing either increases SOD/AE and lowers AAR (Table 6); and (iii) couple the front-end with a duration-explicit HSMM and joint selection (BIC + AE), which extends guarantees from wall-clock filtering to semantic-time alignment (Section 2.3.4, Figure 11). These choices underlie the consistent gains across Tables 3–5 and the robustness trends in Table 7.

Practical implications. Segment-wise covariances (GMR/ProMP) propagate directly to MPC/OMPL for risk-aware execution; the entire pipeline is label-free and computationally linear in the number of demonstrations; Figures 12 and 13 give operational choices for the persistence threshold and state number.

Cross-task transfer. While we evaluate across three distinct domains, we do not train on one domain and test zero-shot on another; we view cross-task/domain transfer of saliency weights and duration priors as orthogonal and leave it as future work.

Limitations and failure cases (by contribution). As summarized in Table 8, failure modes of the saliency front-end mainly arise in weak-signal regions and are mitigated by energy-proportional NMS windows and persistence-aware thresholds.

Table 8. Limitations, triggers, and mitigations.

Contribution	Failure Mode	Trigger	Mitigation
Topology-aware saliency (persistence + NMS)	Missed/duplicate keyframes in weak-signal regions	Long constant-speed stretches; low curvature/DCR; micro-jitter	Energy-proportional NMS window; minimal peak height with persistence elbow; add force/tactile/yaw channels
Topology-aware saliency (persistence + NMS)	Over-pruning near dense peaks	Local oscillations at sharp turns	Two-stage NMS (coarse-to-fine); top-k per <polarity, neighborhood>; relax elbow by one step when peak density is high
Weight self-calibration (CMA-ES on simplex)	Weight collapse; slow/noisy convergence	Heterogeneous demos; rugged SOD surface	Entropy/L2 regularization; multi-start and covariance restarts; early-stop on held-out SOD; feature-stream caching
Duration-explicit HSMM	Under-fitting multi-modal/heavy-tailed dwells	Bi-modal loiter times; operator-dependent pauses	Truncated mixture/quantile dwell distributions; hazard-based HSMM; optional skip transitions
Segment-level generators (DMP/GMR/ProMP)	DMP lacks closed-form uncertainty; GMR/ProMP boundary-sensitive	Risk analysis needs calibrated covariances; segmentation residuals	Use GMR/ProMP for uncertainty; init DMP from GMR mean for low jerk; condition ProMP on soft endpoint priors; boundary-neighborhood reweighting

Future work. Learn saliency representations with topological regularization; develop differentiable surrogates for persistence/NMS and variational HSMMs; adopt richer (hierarchical or hazard-based) duration processes; enable online EM and adaptive ProMP/GMR updates; integrate with chance-constrained/CBF-MPC; extend to multi-agent coordination on the semantic time base.

5. Conclusions

We introduced a label-free pipeline that (i) extracts scale-robust keyframes via topology-aware multi-feature saliency, (ii) performs duration-explicit alignment with a jointly trained HSMM to build a shared semantic time base, and (iii) encodes each phase probabilistically (GMR/ProMP, optionally combined with DMP) for smooth, risk-aware execution. Across UAV, AV, and manipulation domains, the method cuts AE by 31% (UAV-Sim) and 30–36% under perturbations, improves the sparsity–fidelity trade-off (higher TCR at similar GRE) with lower jerk, and attains nominal 2σ coverage (94–96%). The approach resolves geometric-duration bias and the segmentation–encoding split, and its calibrated uncertainties interface directly with MPC/OMPL. Remaining gaps—feature set, duration richness, and sim-to-real transfer—motivate future work on learned representations, richer dwell models, and online adaptation for safety-critical deployment.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/math13193057/s1>.

Author Contributions: Conceptualization, T.G., K.A.N. and D.D.D.; Methodology, T.G. and K.A.N.; Software, T.G.; Validation, B.Y. and S.R.; Formal analysis, T.G.; Investigation (experiments), T.G.,

B.Y. and S.R.; Data curation, T.G., B.Y. and S.R.; Visualization, T.G.; Writing—original draft, T.G.; Writing—review and editing, K.A.N., D.D.D. and T.G.; Supervision, K.A.N. and D.D.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the China Scholarship Council (CSC) under the Russia Talent Training Program, grant number 202108090390.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Generalization Under Model-Selection Criteria

Appendix A.1. Leave-One-Subject/Scene Generalization (UAV-Sim/AV-Sim/Manip-Sim)

Protocol. For each domain, we train HSMMs with $N \in \{2, \dots, 8\}$ on $M - 1$ demonstrations and evaluate on the held-out one (LOSO). We select N by AE-only, BIC-only, and the joint criterion $J(N) = \widetilde{AE}(N) + \widetilde{BIC}(N)$. We report test-AE (s), test-GRE (m for UAV/AV; mm for Manip), and CR (2σ , %); values are means $\pm$ std with BCa 95% CIs in brackets. The LOSO generalization results across domains are summarized in Table A1.

Table A1. LOSO generalization (means $\pm$ std, [BCa 95% CI]).

Domain	Selector	NN	Test-AE (s)	Test-GRE	CR (%)
UAV-Sim (M = 24)	AE-only	7	0.33 ± 0.08 [0.30, 0.36]	0.091 ± 0.020 m [0.087, 0.096]	93.8 ± 2.7
	BIC-only	4	0.36 ± 0.09 [0.33, 0.39]	0.088 ± 0.019 m [0.084, 0.092]	94.6 ± 2.4
	Joint (ours)	6	0.30 ± 0.07 [0.28, 0.32]	0.086 ± 0.019 m [0.083, 0.089]	95.1 ± 2.2
AV-Sim (M = 32)	AE-only	7	0.43 ± 0.09 [0.40, 0.46]	0.202 ± 0.036 m [0.196, 0.209]	93.9 ± 2.8
	BIC-only	4	0.46 ± 0.09 [0.43, 0.49]	0.199 ± 0.035 m [0.193, 0.206]	94.7 ± 2.6
	Joint (ours)	6	0.39 ± 0.08 [0.36, 0.42]	0.196 ± 0.034 m [0.190, 0.203]	95.0 ± 2.5
Manip-Sim (M = 20)	AE-only	6	0.27 ± 0.08 [0.24, 0.29]	0.86 ± 0.17 mm [0.82, 0.90]	94.1 ± 2.6
	BIC-only	4	0.29 ± 0.08 [0.26, 0.31]	0.84 ± 0.16 mm [0.80, 0.88]	94.8 ± 2.4
	Joint (ours)	6	0.25 ± 0.07 [0.23, 0.27]	0.83 ± 0.16 mm [0.79, 0.86]	95.2 ± 2.6

Notes. (i) Significance (joint vs. AE-only/joint vs. BIC-only) holds for test-AE in all domains after the Holm–Bonferroni correction ($p < 0.01$, Cohen’s $d \approx 0.45 - 0.70$); (ii) test-GRE shows no adverse trade-off (joint $\leq$ alternatives within CI overlap); (iii) CR remains near the nominal 95% (94–96%).

Appendix A.2. Synthetic Stress Test (Non-Geometric Dwell, 2–3 Segments)

Setup. Two families of 1-DoF trajectories with non-geometric dwell, mild monotone time-warps (exp. 0.9 – 1.1), additive noise ($\sigma = 0.02$), and 10% random drops; 50 sequences each; train/test = 80/20. The synthetic stress test results are reported in Table A2.

Table A2. Synthetic stress test (means $\pm$ std, [BCa 95% CI]).

Set	Selector	NN	Test-AE (s)
GT-2seg	AE-only	3	0.070 ± 0.014 [0.067, 0.074]
	BIC-only	2	0.062 ± 0.012 [0.059, 0.065]
	Joint (ours)	2	0.051 ± 0.011 [0.049, 0.054]
GT-3seg	AE-only	4	0.084 ± 0.017 [0.080, 0.088]
	BIC-only	2	0.091 ± 0.018 [0.087, 0.095]
	Joint (ours)	3	0.063 ± 0.013 [0.060, 0.066]

Interpretation. AE-only over-segments (inflated test-AE), BIC-only under-segments, whereas the joint criterion recovers the ground-truth N and minimizes test-AE in both sets.

References

- Correia, A.; Alexandre, L.A. A survey of demonstration learning. *Robot. Auton. Syst.* **2024**, *182*, 104812. [\[CrossRef\]](#)
- Jin, W.; Murphey, T.D.; Kulić, D.; Ezer, N.; Mou, S. Learning from sparse demonstrations. *IEEE Trans. Robot.* **2022**, *39*, 645–664. [\[CrossRef\]](#)
- Lee, D.; Yu, S.; Ju, H.; Yu, H. Weakly supervised temporal anomaly segmentation with dynamic time warping. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 11–17 October 2021; pp. 7355–7364.
- Braglia, G.; Tebaldi, D.; Lazzaretti, A.E.; Biagiotti, L. Arc-length-based warping for robot skill synthesis from multiple demonstrations. *arXiv* **2024**, arXiv:2410.13322.
- Si, W.; Wang, N.; Yang, C. A review on manipulation skill acquisition through teleoperation-based learning from demonstration. *Cogn. Comput. Syst.* **2021**, *3*, 1–16. [\[CrossRef\]](#)
- Arduengo, M.; Colomé, A.; Lobo-Prat, J.; Sentis, L.; Torras, C. Gaussian-process-based robot learning from demonstration. *J. Ambient. Intell. Humaniz. Comput.* **2023**, *14*, 1–14. [\[CrossRef\]](#)
- Tavassoli, M.; Katyara, S.; Pozzi, M.; Deshpande, N.; Caldwell, D.G.; Prattichizzo, D. Learning skills from demonstrations: A trend from motion primitives to experience abstraction. *IEEE Trans. Cogn. Dev. Syst.* **2023**, *16*, 57–74. [\[CrossRef\]](#)
- Ansari, A.F.; Benidis, K.; Kurlle, R.; Turkmen, A.C.; Soh, H.; Smola, A.; Wang, B.; Januschowski, T. Deep explicit duration switching models for time series. *Adv. Neural Inf. Process. Syst. (NeurIPS)* **2021**, *34*, 29949–29961.
- Sosa-Ceron, A.D.; Gonzalez-Hernandez, H.G.; Reyes-Avendaño, J.A. Learning from demonstrations in human–robot collaborative scenarios: A survey. *Robotics* **2022**, *11*, 126. [\[CrossRef\]](#)
- Ruiz-Suarez, S.; Leos-Barajas, V.; Morales, J.M. Hidden Markov and semi-Markov models: When and why are these models useful for classifying states in time series data. *J. Agric. Biol. Environ. Stat.* **2022**, *27*, 339–363. [\[CrossRef\]](#)
- Pohle, J.; Adam, T.; Beumer, L.T. Flexible estimation of the state dwell-time distribution in hidden semi-Markov models. *Comput. Stat. Data Anal.* **2022**, *172*, 107479. [\[CrossRef\]](#)
- Wang, X.; Li, J.; Xu, G.; Wang, X. A novel zero-velocity interval detection algorithm for a pedestrian navigation system with foot-mounted inertial sensors. *Sensors* **2024**, *24*, 838. [\[CrossRef\]](#) [\[PubMed\]](#)
- Haussler, A.M.; Tueth, L.E.; May, D.S.; Earhart, G.M.; Mazzoni, P. Refinement of an algorithm to detect and predict freezing of gait in Parkinson disease using wearable sensors. *Sensors* **2024**, *25*, 124. [\[CrossRef\]](#) [\[PubMed\]](#)
- Altamirano, M.; Briol, F.X.; Knoblauch, J. Robust and scalable Bayesian online changepoint detection. In Proceedings of the International Conference on Machine Learning (ICML), Honolulu, HI, USA, 23–29 July 2023; PMLR: Cambridge, MA, USA, 2023; Volume 202, pp. 642–663.
- Sellier, J.; Dellaportas, P. Bayesian online change point detection with Hilbert-space approximate Student-t process. In Proceedings of the International Conference on Machine Learning (ICML), Honolulu, HI, USA, 23–29 July 2023; PMLR: Cambridge, MA, USA, 2023; pp. 30553–30569.
- Tsaknaki, I.Y.; Lillo, F.; Mazzarisi, P. Bayesian autoregressive online change-point detection with time-varying parameters. *Commun. Nonlinear Sci. Numer. Simul.* **2025**, *142*, 108500. [\[CrossRef\]](#)
- Buchin, K.; Nusser, A.; Wong, S. Computing continuous dynamic time warping of time series in polynomial time. *arXiv* **2022**, arXiv:2203.04531.
- Wang, L.; Koniusz, P. Uncertainty-DTW for time series and sequences. In *European Conference on Computer Vision (ECCV)*; Springer: Cham, Switzerland, 2022; pp. 176–195.
- Mikheeva, O.; Kazlauskaitė, I.; Hartshorne, A.; Kjellström, H.; Ek, C.H.; Campbell, N. Aligned multi-task Gaussian process. In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS), Valencia, Spain, 28–30 March 2022; PMLR: Cambridge, MA, USA, 2022; pp. 2970–2988.
- Saveriano, M.; Abu-Dakka, F.J.; Kramberger, A.; Peternel, L. Dynamic movement primitives in robotics: A tutorial survey. *Int. J. Robot. Res.* **2023**, *42*, 1133–1184. [\[CrossRef\]](#)
- Barekatain, A.; Habibi, H.; Voos, H. A practical roadmap to learning from demonstration for robotic manipulators in manufacturing. *Robotics* **2024**, *13*, 100. [\[CrossRef\]](#)
- Urain, J.; Mandlekar, A.; Du, Y.; Shafiullah, M.; Xu, D.; Fragkiadaki, K.; Chalvatzaki, G.; Peters, J. Deep Generative Models in Robotics: A Survey on Learning from Multimodal Demonstrations. *arXiv* **2024**, arXiv:2408.04380.
- Vélez-Cruz, N. A survey on Bayesian nonparametric learning for time series analysis. *Front. Signal Process.* **2024**, *3*, 1287516. [\[CrossRef\]](#)
- Tanwani, A.K.; Yan, A.; Lee, J.; Calinon, S.; Goldberg, K. Sequential robot imitation learning from observations. *Int. J. Robot. Res.* **2021**, *40*, 1306–1325. [\[CrossRef\]](#)

25. Bonzanini, A.D.; Mesbah, A.; Di Cairano, S. Perception-aware chance-constrained model predictive control for uncertain environments. In Proceedings of the 2021 American Control Conference (ACC), New Orleans, LA, USA, 25–28 May 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 2082–2087.
26. El-Yaagoubi, A.B.; Chung, M.K.; Ombao, H. Topological data analysis for multivariate time series data. *Entropy* **2023**, *25*, 1509. [\[CrossRef\]](#)
27. Nomura, M.; Shibata, M. cmaes: A simple yet practical Python library for CMA-ES. *arXiv* **2024**, arXiv:2402.01373.
28. Schafer, R.W. What is a Savitzky–Golay filter? *IEEE Signal Process. Mag.* **2011**, *28*, 111–117. [\[CrossRef\]](#)
29. Tapp, K. *Differential Geometry of Curves and Surfaces*; Springer: Cham, Switzerland, 2016.
30. Gorodski, C. *A Short Course on the Differential Geometry of Curves and Surfaces*; Lecture Notes; University of São Paulo: São Paulo, Brazil, 2023.
31. Cohen-Steiner, D.; Edelsbrunner, H.; Harer, J. Stability of persistence diagrams. *Discret. Comput. Geom.* **2007**, *37*, 103–120. [\[CrossRef\]](#)
32. Satopaa, V.; Albrecht, J.; Irwin, D.; Raghavan, B. Finding a “Kneedle” in a haystack: Detecting knee points in system behavior. In Proceedings of the ICDCS Workshops, Minneapolis, MN, USA, 20–24 June 2011; pp. 166–171.
33. Skraba, P.; Turner, K. Wasserstein stability for persistence diagrams. *arXiv* **2025**, arXiv:2006.16824v7.
34. Hansen, N. The CMA Evolution Strategy: A Tutorial. *arXiv* **2016**, arXiv:1604.00772. [\[CrossRef\]](#)
35. Singh, G.S.; Acerbi, L. PyBADs: Fast and robust black-box optimization in Python. *J. Open Source Softw.* **2024**, *9*, 5694. [\[CrossRef\]](#)
36. Akimoto, Y.; Auger, A.; Glasmachers, T.; Morinaga, D. Global linear convergence of evolution strategies on more-than-smooth strongly convex functions. *SIAM J. Optim.* **2022**, *32*, 1402–1429. [\[CrossRef\]](#)
37. Yu, S.-Z. Hidden semi-Markov models. *Artif. Intell.* **2010**, *174*, 215–243. [\[CrossRef\]](#)
38. Chiappa, S. Explicit-duration Markov switching models. *Found. Trends Mach. Learn.* **2014**, *7*, 803–886. [\[CrossRef\]](#)
39. Merlo, L.; Maruotti, A.; Petrella, L.; Punzo, A. Quantile hidden semi-Markov models for multivariate time series. *Stat. Comput.* **2022**, *32*, 61. [\[CrossRef\]](#)
40. Jurafsky, D.; Martin, J.H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd ed.; Online manuscript. Available online: <https://web.stanford.edu/~jurafsky/slp3/> (accessed on 18 August 2025).
41. Yu, S.-Z.; Kobayashi, H. An efficient forward–backward algorithm for an explicit-duration hidden Markov model. *IEEE Signal Process. Lett.* **2003**, *10*, 11–14.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.