SECOND-ORDER MOTION REPRESENTATIONS: GRASSMANNIAN DIFFERENTIAL KINEMATICS FOR UNSUPERVISED HAND-OBJECT INTERACTION SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Representations of human motion are typically built from appearance, spatial configuration, or first-order displacement. We show that what they omit is second-order structure: the rate at which motion direction and speed change. This is not recoverable from zero- or first-order descriptors, yet it is what distinguishes execution modes sharing objects, locations, and appearance - tightening versus loosening, moving versus pouring. We propose a representation that makes this structure explicit. Short windows of 3D scene flow are lifted to points on the Grassmann manifold, yielding a subspace trajectory whose covariant derivatives expose manifold acceleration and direction change as independent, coordinate-free descriptors of how motion evolves. Recovering them from noisy observations poses a representational trade-off: differentiation amplifies noise, while temporal averaging suppresses the transitions that constitute the signal. We resolve this by combining localized Fréchet means with a preemptive jump-diffusion process that halts integration across physical discontinuities.We analyse the resulting representation formally, deriving a discrete curvature error bound under stated assumptions and establishing the necessity of second-order features for a constructed class of kinematically ambiguous boundaries. The representation enables causal and training-free boundary detection. Detected segments are subsequently represented by fixed-dimensional spatial path-signature descriptors and grouped by histogram-intersection similarity with complete-linkage clustering, allowing recurring action instances to receive common discovered labels. We test the representation on unsupervised temporal segmentation. When all methods are given the same 3D scene flow, ours performs best, showing that the gains come from the representation and not from the use of 3D input, and it improves over both zero-order models and recent segmentation methods. We validate our error bounds empirically under severe noise and in realistic settings with estimated scene flow, evaluating on OakInk2, HOT3D, and EPIC-KITCHENS. Accuracy degrades predictably as the kinematic signal-to-noise ratio of the input flow falls, showing that our method operates close to the information limit of the input rather than being limited by the representation itself.

1 INTRODUCTION

Understanding hand-object interactions (HOI) — how humans manipulate objects through continuous, often non-monotone motion — is a fundamental problem in computer vision, robotics, and embodied perception. Large-scale datasets such as OakInk2 Zhan et al. (2024) and HOT3D Banerjee et al. (2025) have enabled significant progress in modeling such interactions. Across a wide range of real-world applications, the meaning of an interaction depends not only on what objects are involved or where the interaction occurs, but critically on how the motion is executed.

For example, in *procedural training in extended reality (XR)*, tasks such as assembly or maintenance require distinguishing correct from incorrect execution of the same step. Actions like tightening vs. loosening or correct vs. incorrect alignment often involve identical objects and configurations, differing only in motion direction and temporal dynamics. These scenarios highlight a key property of HOI: interactions are driven by *fine-grained kinematic variations*, where appearance remains

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.

constant while motion dynamics change. As a result, segmentation from continuous, untrimmed observations is challenging; boundaries often arise from subtle shifts in speed, direction, or temporal evolution rather than visible changes. This is especially pronounced in egocentric settings, where meaningful transitions are primarily reflected in motion evolution. Existing methods largely rely on signals capturing *what* is present and *where*, using appearance, spatial structure, and first-order motion cues, which are effective only when boundaries align with observable visual or semantic changes.

What + Where as the dominant paradigm. Modern approaches to video understanding include supervised temporal localization Zhang et al. (2022); Yi et al. (2021), self-supervised alignment methods Ali et al. (2025), and vision-language models Radford et al. (2021) learn increasingly powerful representations of objects, scenes, and semantic actions. In the context of HOI, recent works further incorporate object-centric and egocentric representations Xu et al. (2023); Hasson et al. (2019) to capture interaction structure. Across these directions, the combination of appearance, spatial configuration, and first-order motion provides strong signals for identifying *what* is happening and *where* it occurs. Unsupervised alternatives detect change in these same signals. The Kernel Temporal Segmentation (KTS) (Potapov et al., 2014) locates boundaries as shifts in the mean of a kernel feature space, and Temporal Subspace Clustering (TSC) (Li et al., 2015) groups frames via iterative subspace optimisation. Both are zero-order, responding to changes in the level of the signal rather than in how it evolves.

From static subspaces to dynamic motion geometry. Alternatively, geometric approaches map linear dynamical systems to Grassmann manifolds Turaga et al. (2011); Slama et al. (2015). However, modeling motion as a *single static subspace* implicitly assumes global linearity, failing to capture the time-varying nature of HOI. Recent unsupervised approaches attempt to flatten complex manifolds into linear subspaces Ding et al. (2023). While effective for static clustering, flattening destroys intrinsic temporal evolution, making fine-grained execution indistinguishable.

The necessity of modeling *how*. In fine-grained HOI, semantic labels and execution speed may remain constant even as kinematic intent abruptly shifts (e.g., reaching to twisting). Zero-order spatial distances and static models are mathematically blind to these transitions. Consider tightening a fastener rather than loosening it, or moving a vessel rather than pouring from it. The objects are the same, the location is the same, and the appearance is the same. The displacement magnitude may also be the same. What differs is how the motion evolves — where it reverses, where it slows before contact, how its direction turns. We show in Section 3 that a representation built from displacement alone is degenerate on this family. We construct trajectories that are indistinguishable under step-wise distance, yet separated once the rate of change of motion is taken into account. The distinction is not difficult to learn from such representations. It is absent from them.

Our approach: Second-order structure, and why it needs a manifold.

Recovering what is missing means describing not just how far the motion moved, but how its evolution changed. Taking differences of raw flow vectors in Euclidean space does not achieve this. Raw flow carries a coordinate frame, so differences confound genuine changes in motion with changes in viewpoint, and they inherit the full noise of every individual point. We therefore represent each short window of 3D scene flow by the subspace its motion spans, giving a point on the Grassmann manifold. This discards the coordinate frame and the per-point noise while keeping the dominant structure of the motion, and it makes differentiation well posed as successive motion states lie in different tangent spaces, and are compared only after parallel transport along the manifold.

The resulting trajectory has a natural notion of unforced motion. A trajectory that follows a geodesic corresponds to motion whose structure is evolving uniformly, for example, a steady sweep or a sustained pour, which may be fast, but does not change in character. Departure from a geodesic is what marks a transition. We measure this departure in two complementary ways. Manifold acceleration responds to any change in the motion state, whether in speed or in direction. The direction-change angle responds only to a turn in the motion, and is provably unaffected by how fast the action is executed. Neither can be recovered from the other: the first is ambiguous between a fast reversal and a sudden slowing, and the second is undefined when the hand is nearly still. Together they describe how the motion is changing, independently of any external frame of reference.

A representational trade-off. Estimating this structure from discrete and noisy observations forces a choice. Differentiating amplifies observation noise, while averaging over time suppresses that noise but also blurs the sharp transitions that carry the signal. We show that the two effects are governed by an explicit separation of temporal scales, and resolve the trade-off by combining localized Fréchet means with a preemptive jump-diffusion process that halts integration across physical discontinuities instead of differentiating through them. This is what allows second-order structure to survive realistic noise, and it is the technical core of the representation.

Our approach enables **causal boundary detection** via a rolling tangent space representation, allowing interaction boundaries to be detected online from local changes in kinematic structure without access to future frames. This provides a principled mechanism for real-time segmentation, in contrast to methods that rely on global temporal context or offline processing. Additionally, the detected segments are subsequently grouped offline using a spatial path signature , which summarises the ordered motion-mode structure of each variable-length segment as a fixed-dimensional descriptor. Histogram-intersection similarity and complete-linkage clustering are then used to assign recurring action instances a common discovered label. Collectively, we provide direct, coordinate-free access to motion evolution, developing a **fully unsupervised, training-free pipeline** that discovers HOI boundaries directly from intrinsic kinematic structure. Our contributions are summarized as follows:

- **Dynamic Geometric Representation:** We lift 3D scene flow into continuous Grassmannian trajectories, capturing intrinsic second-order kinematics.
- **Resolving the Derivative Paradox:** A novel algorithmic framework utilizing localized Fréchet smoothing and preemptive jump-diffusion to extract kinematics without blurring action boundaries.
- **Theoretical Guarantees:** Formal proofs bounding discrete curvature error (Theorem 3.1), proving the necessity of second-order features (Theorem 3.2).
- **Unsupervised Discovery:** A causal segmentation pipeline that improves over self-supervised alignment methods and VLMs in discovering fine-grained execution dynamics.
- **Unsupervised Segment Grouping:** A spatial path-signature descriptor converts variable-length segments into fixed-dimensional representations, while histogram-intersection similarity and complete-linkage clustering group recurring action instances without supervision.

2 METHODOLOGY

We present a geometric framework for segmenting hand-object interactions (HOI) by modeling how motion evolves over time. In egocentric settings, where appearance and spatial configuration remain largely unchanged, interaction boundaries are primarily reflected through changes in motion execution. This motivates a representation that focuses on motion dynamics rather than appearance. The overall pipeline is summarized in Algorithm 1.

2.1 MOTION REPRESENTATION

Given a sequence of point clouds $\{P_t\}_{t=1}^{T}$, with $P_t \in \mathbb{R}^{N\times 3}$, we compute scene flow to isolate the dynamic component of motion and removes static spatial bias, which is particularly important in egocentric interaction scenarios. $V_t = P_{t+1} - P_t$.

2.2 LOCAL SUBSPACE MODELING AND RIEMANNIAN SMOOTHING

We partition the sequence $\{V_t\}$ into non-overlapping temporal windows, where each window is represented by a matrix $F_\tau \in \mathbb{R}^{W\times 3N}$. For each window, we compute a rank-k approximation using singular value decomposition Eckart & Young (1936), yielding a sequence of subspaces $\{U_\tau\} \subset \mathcal{G}(k, 3N)$.

Riemannian Smoothing for Noise Robustness. To address the inherent sparsity and noise in 3D scene flow, computing discrete second-order derivatives directly on raw subspaces is mathematically unstable. We introduce a temporal smoothing operator natively on the Grassmann manifold Pennec (2006). Let $\mathcal{N}(\tau)$ be a local causal temporal neighborhood around window τ. We compute the

Algorithm 1 HOI Motion Segmentation and Correspondence Pipeline

Require: Point cloud sequence $\{P_t\}_{t=1}^{T}$, rank k, window size W, smoothing window W_s, jump threshold $L_{\max}$, temporal step Δt, score threshold $s_{\min}$, grouping threshold τ_{B}
Ensure: Action boundaries $\mathcal{F}$, cluster labels $\ell_{1:M}$

Phase 1 - Representation and Riemannian Smoothing

$V_t \leftarrow P_{t+1} - P_t \quad \forall t \in [1, T-1]$ ▷ Compute scene flow
$\{U_\tau\} \leftarrow \textsc{TruncatedSVD}(\{V_t\}, W, k)$ ▷ Extract local subspaces $\in \mathcal{G}(k, 3N)$
for each window τ **do**
  $\bar{U}_\tau \leftarrow \arg\min_{U \in \mathcal{G}} \sum_{t \in \mathcal{N}(\tau)} d_g^2(U, U_t)$ ▷ Fréchet mean over W_s
end for
$\bar{\mathcal{S}} \leftarrow \{\bar{U}_\tau\}$ ▷ Smoothed Grassmann trajectory

Phase 2 - Tangent Features & Jump-Diffusion

$\{\kappa_\tau\}, \{\theta_\tau\} \leftarrow \textsc{ComputeRobustRollingTangent}(\bar{\mathcal{S}}, W, L_{\max}, \Delta t)$ ▷ See Algorithm 2

Phase 3 - Boundary Detection

$\bar{\kappa}_\tau, \bar{\theta}_\tau \leftarrow \textsc{Normalize}(\{\kappa_\tau\}, \{\theta_\tau\})$
$s_\tau \leftarrow \alpha \bar{\theta}_\tau + (1-\alpha)\bar{\kappa}_\tau$ ▷ Combined kinematic score
$\mathcal{B} \leftarrow \textsc{FindPeaks}(s_\tau, s_{\min})$ ▷ Extract boundary window indices
$\mathcal{F} \leftarrow \textsc{MapToFrames}(\mathcal{B}, W)$ ▷ Convert to frame indices

Phase 4 - Offline Segment Grouping

$\mathcal{S} \leftarrow \{U_\tau\}$ ▷ Ordered bases before smoothing
$\mathcal{S}^{(1)}, \ldots, \mathcal{S}^{(M)}, \{c_{i\tau}\} \leftarrow \textsc{CollectSegmentWindows}(\mathcal{S}, \mathcal{F})$
for each segment i **do**
  $\Phi^{(i)} \leftarrow \textsc{ComputeSpatialSignature}(\mathcal{S}^{(i)}, \{c_{i\tau}\}, k)$
end for
$K_{ij} \leftarrow 1 - \frac{1}{2}\|\Phi^{(i)} - \Phi^{(j)}\|_1, \quad \forall i, j \in \{1, \ldots, M\}$ ▷ Segment similarity matrix
$\ell_{1:M} \leftarrow \textsc{CompleteLinkage}(K, \tau_{\mathrm{B}})$
return $\mathcal{F}$, $\ell_{1:M}$

Fréchet (Karcher) mean subspace:

$$\bar{U}_\tau = \arg\min_{U \in \mathcal{G}} \sum_{t \in \mathcal{N}(\tau)} d_g^2(U, U_t), \tag{1}$$

where d_g is the geodesic distance. The smoothed sequence $\{\bar{U}_\tau\}$ forms the basis for subsequent kinematic analysis, effectively acting as a low-pass filter for high-frequency point cloud noise.

2.3 Resolving the Smoothing-Derivative Trade-off

Calculating discrete curvature presents a paradox: differentiation amplifies sensor noise, yet aggressive smoothing obliterates the sharp boundary impulses. We resolve this via **Temporal Scale Separation**. By explicitly separating high-frequency noise ($\mathcal{O}(\epsilon)$) from mid-frequency physical transitions (Δt) and low-frequency sustained actions ($\gg \Delta t$), we construct a targeted geometric pipeline. Utilizing localized Fréchet smoothing combined with preemptive jump-diffusion , we successfully attenuate stochastic noise without blurring critical kinematic boundaries.

2.4 Tangent Features & Jump-Diffusion

The sequence of smoothed subspaces $\{\bar{U}_\tau\}$ defines a trajectory on the Grassmann manifold $\mathcal{G}(k, n)$. Prior works have demonstrated the efficacy of static Grassmann representations for action recognition Turaga et al. (2011); Slama et al. (2015). However, instead of comparing subspaces using pairwise distances, we analyze their evolution over time. The logarithmic map at $\bar{U}_\tau$ maps a nearby point $\bar{U}_{\tau+1} \in \mathcal{G}(k, n)$ to a tangent vector $\xi \in \mathcal{T}_{\bar{U}_\tau}\mathcal{G}$, representing the velocity of the motion state. After computing the tangent velocity $\xi_{\tau-1}$ at step $\tau-1$, we compare it with ξ_τ at step τ. Because these two vectors live in different tangent spaces, we use parallel transport $\mathcal{P}_{\bar{U}_{\tau-1} \to \bar{U}_\tau}(\xi_{\tau-1})$ to move the vector along the geodesic, keeping it consistent with the manifold curvature. The transported vector $\tilde{\xi} \in \mathcal{T}_{\bar{U}_\tau}\mathcal{G}$ can then be legitimately compared with ξ_τ.

As detailed in Algorithm 2, κ_τ measures the manifold acceleration (the rate of change of the Grassmann velocity vector). Our preemptive jump condition (Line 5) explicitly models abrupt physical state transitions. For computational efficiency during the online rolling window, we evaluate this condition using the chordal distance d_c, which tightly bounds the geodesic distance. When this chordal distance exceeds the expected diffusion scale, we record a maximal boundary impulse κ_{jump} and reset the transport chain, avoiding the mathematical instability of differentiating across a physical discontinuity.

Algorithm 2 ComputeRobustRollingTangent

Require: $\mathcal{S} = \{\bar{U}_0, \ldots, \bar{U}_{M_S-1}\}, W, L_{\max}, \Delta t$
Ensure: $\{\kappa_\tau\}, \{\theta_\tau\}$
1: $\xi_1 \leftarrow \mathrm{Log}_{\bar{U}_0}(\bar{U}_1)$ ▷ initial velocity at $\bar{U}_0$
2: $\tilde{\xi} \leftarrow \mathcal{P}_{\bar{U}_0 \to \bar{U}_1}(\xi_1)$ ▷ transport to $\mathcal{T}_{\bar{U}_1}\mathcal{G}$
3: **for** $\tau = 1$ **to** $M_S - 2$ **do**
4: $\quad \xi_\tau \leftarrow \mathrm{Log}_{\bar{U}_\tau}(\bar{U}_{\tau+1})$
5: $\quad$ **if** $d_c(\bar{U}_\tau, \bar{U}_{\tau+1}) > L_{\max}\Delta t / W$ **then**
6: $\quad\quad \kappa_\tau \leftarrow \kappa_{\text{jump}}$ ▷ Preemptive jump-diffusion impulse
7: $\quad\quad \theta_\tau \leftarrow 0$
8: $\quad\quad \tilde{\xi} \leftarrow \xi_\tau$ ▷ Reset transport chain
9: $\quad\quad$ **continue**
10: $\quad$ **end if**
11: $\quad \Delta\xi \leftarrow \xi_\tau - \tilde{\xi}$ ▷ both in $\mathcal{T}_{\bar{U}_\tau}\mathcal{G}$
12: $\quad \kappa_\tau \leftarrow \|\Delta\xi\|_F$ ▷ Manifold acceleration (curvature)
13: $\quad$ **if** $\left\|\tilde{\xi}\right\|_F > \varepsilon$ **and** $\|\xi_\tau\|_F > \varepsilon$ **then**
14: $\quad\quad \theta_\tau \leftarrow \arccos\left(\dfrac{\langle\tilde{\xi},\, \xi_\tau\rangle_F}{\left\|\tilde{\xi}\right\|_F \|\xi_\tau\|_F}\right)$
15: $\quad$ **else**
16: $\quad\quad \theta_\tau \leftarrow 0$
17: $\quad$ **end if**
18: $\quad \tilde{\xi} \leftarrow \mathcal{P}_{\bar{U}_\tau \to \bar{U}_{\tau+1}}(\xi_\tau)$
19: **end for**

2.5 Boundary Detection

The outcome κ_τ and θ_τ computed by Algorithm 2, are normalised to $[0, 1]$:

$$\begin{cases} \bar{\theta}_\tau & = \theta_\tau / \pi \\ \bar{\kappa}_\tau & = \min(\kappa_\tau / \kappa_{p95},\ 1) \end{cases} \tag{2}$$

where κ_{p95} is the 95th percentile of $\{\kappa_\tau\}$ (robust to outlier spikes that would collapse all other values if the maximum were used).

The combined score is:

$$s_\tau = \alpha\, \bar{\theta}_\tau + (1 - \alpha)\, \bar{\kappa}_\tau, \quad \alpha \in [0, 1]. \tag{3}$$

2.6 Offline Segment Grouping: Spatial signature based

Boundary detection produces unlabeled intervals; we group temporally separated intervals with similar motion content to discover recurring action labels. A segment descriptor must (i) summarise an interval of arbitrary length as a fixed-dimensional vector, (ii) be invariant to execution speed, and (iii) be estimable from short intervals, which rules out quantities obtained by differentiation. We further require the descriptor to be computed from quantities the pipeline already produces. This excludes κ_τ and θ_τ: as covariant derivatives, they respond to *change* in motion state rather than the state itself, so $\kappa_\tau \approx 0$ throughout a sustained action regardless of which action it is. Grouping instead requires the sustained spatial structure of the motion, extracted directly from the local bases.

Alternatives. We compare three descriptors and a singleton control in Appendix C.3. The projection kernel discards within-segment evolution and merges dissimilar intervals. Dynamic Time Warping (DTW) on principal-angle sequences retains temporal order but remains unreliable for short intervals, while its fusion with the projection kernel provides limited improvement. Mean per-point displacement is the strongest alternative but captures only motion magnitude. Our spatial descriptor is the only method that outperforms the singleton control in both ARI and NMI.

Choice of bases. We use the ordered bases $U_\tau \in \mathbb{R}^{3N\times k}$ produced by the truncated SVD of Section 2.2, before Fréchet smoothing. The Fréchet mean $\bar{U}_\tau$ is a point on $\mathcal{G}(k, 3N)$ and therefore determines a basis only up to an orthogonal transformation, whereas the descriptor below depends on the singular-value ordering of the columns. The unsmoothed bases retain that ordering.

Spatial signature of a local basis. Let

$$u_{\tau,r}^{(n)} = \begin{bmatrix} U_\tau(3n-2, r) \\ U_\tau(3n-1, r) \\ U_\tau(3n, r) \end{bmatrix} \in \mathbb{R}^3$$

denote the three-dimensional block of the rth basis vector associated with point n. We define its nonnegative participation magnitude as

$$w_{\tau rn} = \|u_{\tau,r}^{(n)}\|_2, \qquad p_{\tau rn} = \frac{w_{\tau rn}}{Z_\tau}, \qquad Z_\tau = \sum_{r=1}^{k}\sum_{n=1}^{N} w_{\tau rn}.$$

Since every column of U_τ has unit norm, $\sum_n w_{\tau rn} \geq 1$, and hence $Z_\tau \geq k > 0$. Therefore, p_τ defines a probability distribution over mode–point pairs. To preserve the ordering of the singular modes, let $q_r = (r - \frac{1}{2})/k$ and define

$$\phi_{\tau n}^{\text{lead}} = \sum_{r=1}^{k}(1-q_r)p_{\tau rn}, \qquad \phi_{\tau n}^{\text{trail}} = \sum_{r=1}^{k} q_r p_{\tau rn}.$$

The resulting descriptor

$$\Phi_\tau = \left[(\phi_\tau^{\text{lead}})^\top, (\phi_\tau^{\text{trail}})^\top\right]^\top \in \mathbb{R}^{2N}$$

summarises both the total participation of each point and its first moment along the ordered mode axis. In particular,

$$\sum_{n=1}^{N}\left(\phi_{\tau n}^{\text{lead}} + \phi_{\tau n}^{\text{trail}}\right) = 1,$$

so Φ_τ is a normalized nonnegative descriptor. This construction is inspired by ordered path representations (Chen, 1958; Lyons, 1994), but the ordering is over singular modes rather than time.

Segment-level pooling. Let $I_i = [a_i, b_i)$ be an interval delimited by the detected boundaries $\mathcal{F}$, and let $c_{i\tau}$ count the frames of I_i falling in window τ. Pooling with these overlap weights,

$$\Phi^{(i)} = \sum_\tau \omega_{i\tau}\Phi_\tau, \qquad \omega_{i\tau} = \frac{c_{i\tau}}{\sum_\nu c_{i\nu}}, \tag{4}$$

gives a fixed-dimensional descriptor independent of interval length. Because the weights sum to one and each Φ_τ has unit mass, $\Phi^{(i)}$ is itself a distribution. Normalising by total support is what makes the descriptor invariant to execution speed, satisfying requirement (ii)

Similarity and clustering. Since descriptors are distributions, we compare them by histogram intersection,

$$K_{ij} = 1 - \tfrac{1}{2}\left\|\Phi^{(i)} - \Phi^{(j)}\right\|_1 = \sum_{r=1}^{2N}\min\left(\Phi_r^{(i)}, \Phi_r^{(j)}\right), \tag{5}$$

and apply complete-linkage clustering to the distances $1 - K_{ij}$, cutting the hierarchy at $1 - \tau_B$. Complete linkage is chosen over single-linkage or threshold-graph alternatives because it bounds cluster *diameter* rather than nearest-neighbour distance: every pair of segments in a returned cluster satisfies $K_{ij} \geq \tau_B$ so dissimilar segments cannot be merged transitively through a chain of intermediates. The cut also fixes the number of clusters without a silhouette sweep. Sensitivity to τ_B is reported in Appendix C.2. Clustering helper algorithms for Spatial Signature and Complete Linkage are provided in Appendix B.4.

3 THEORETICAL ANALYSIS

We provide a principled justification for our geometric framework, focusing on the stability of second-order kinematic features under observational noise. Detailed formalisms and full proofs are deferred to the Appendix.

3.1 MODELING NOISE AND SCALE SEPARATION

Raw 3D scene flow is inherently noisy. Let the true subspace at window τ be U_τ^* and the observed subspace be $\hat{U}_\tau$, with spatial perturbation variance σ^2. To mitigate this, we employ the Fréchet mean over a temporal neighborhood $\mathcal{N}(\tau)$ of size W_s.

This smoothing introduces a bias-variance trade-off. While the variance of the empirical mean decreases as $\mathcal{O}(\sigma^2/W_s)$, the temporal bias (phase blur) scales quadratically with W_s relative to the true trajectory's maximum curvature $\kappa_{\max}^*$. To resolve the smoothing-derivative paradox, the window size W_s must satisfy the theoretical scale separation constraint:

$$1 < W_s \ll \frac{1}{\Delta t}\sqrt{\frac{\sigma}{\kappa_{\max}^*}}. \tag{6}$$

Our jump-diffusion formulation guarantees this by adaptively halting the integration when local variation exceeds the expected diffusion scale, effectively resetting the parallel transport chain to avoid differentiating across a physical discontinuity.

3.2 ERROR BOUNDS FOR DISCRETE CURVATURE

The core of our pipeline relies on computing the discrete curvature $\kappa_\tau = \|\xi_\tau - \tilde{\xi}_{\tau-1}\|_F$ via parallel transport. We bound the approximation error caused by the residual smoothed noise and the temporal discretization.

Theorem 3.1 (Curvature Error Bound)**.** *Assume the residual noise after Fréchet smoothing satisfies $d_g(\bar{U}_\tau, U_\tau^*) \le \epsilon$. Let κ_τ^* be the true continuous manifold curvature at τ, and $\hat{\kappa}_\tau$ be the empirical discrete curvature computed with step size Δt. The error is bounded by:*

$$|\hat{\kappa}_\tau - \kappa_\tau^*| \le C_1 \frac{\epsilon}{\Delta t} + C_2(\Delta t)^2, \tag{7}$$

where $C_1, C_2 > 0$ depend on the maximum sectional curvature of $\mathcal{G}(k, n)$.

Proof Sketch. The error decomposes into a discretization term bounded by $\mathcal{O}((\Delta t)^2)$ and a noise perturbation term. Because the logarithmic map and parallel transport are locally Lipschitz, the spatial error ϵ translates to a velocity error bounded by $C_1\epsilon$, which is divided by Δt when computing the curvature rate. (Full proof in Appendix. A.2).

3.3 THE INSUFFICIENCY OF ZERO-ORDER DISTANCES

We formally establish why second-order curvature is mathematically necessary over baseline first-order distances.

Theorem 3.2 (Curvature Sensitivity)**.** *Let $\gamma_1(\tau)$ (constant motion) and $\gamma_2(\tau)$ (accelerating motion) be distinct subspace trajectories on $\mathcal{G}(k, n)$. There exists a parameterization where their step-wise zero-order distances are identical ($d_g(\gamma_1(\tau), \gamma_1(\tau+1)) = d_g(\gamma_2(\tau), \gamma_2(\tau+1))$), but their discrete covariant derivatives strictly distinguish them ($\kappa_\tau^{(1)} = 0$ while $\kappa_\tau^{(2)} > 0$).*

Proof Sketch. Constant volumetric displacement masks interaction boundaries in zero-order metric space. Curvature (κ_τ), acting as the covariant derivative, isolates non-geodesic deviations independent of magnitude, breaking the degeneracy. (Full proof in Appendix. A.3).

4 EXPERIMENTS

Our experiments validate: (i) the ability of second-order kinematics to resolve boundaries missed by zero-order distances, (ii) robustness of discrete curvature under scene-flow noise, (iii) dependency of

performance on upstream flow quality via kinematic SNR (kSNR), and (iv) the effectiveness of the spatial signature for grouping recurring actions.

4.1 Datasets and Evaluation Protocol

We evaluate on OakInk2 Zhan et al. (2024) and HOT3D Banerjee et al. (2025), which provide high-resolution geometric reconstructions of hand-object interactions. OakInk2 emphasizes fine-grained manipulation dynamics, so we reannotate sequences longer than two minutes to distinguish subtle motions such as pick, pour, and scrape ($\approx$26 actions per sequence, 110+ distinct actions). We similarly annotate a subset of HOT3D, though it is not a procedural dataset. Annotation and preprocessing details are in Appendix G.

Metrics. For boundary detection and segmentation, we report Segmental F1 at varying overlap thresholds (F1@10, F1@50), Mean-of-frames accuracy (MoF), Mean IoU (mIoU), and Edit distance, following the Hungarian-matching protocol of Ali et al. (2025). For interval clustering, we report Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) with arithmetic-mean entropy normalization

4.2 Comparison with State-of-the-Art Baselines

We compare against five paradigms: self-supervised alignment (VASOT) Ali et al. (2025), supervised segmentation (ASFormer) Yi et al. (2021), a vision-language model (Qwen3.6-27B) Team (2026) (selected for its strong multimodal performance on complex reasoning benchmarks), Kernel Temporal Segmentation Potapov et al. (2014), and Temporal Subspace Clustering Li et al. (2015). For VASOT, we use the official implementation, performing self-supervised alignment on the evaluation sequences to obtain the reported performance; this alignment must be re-run whenever new sequences are evaluated. Since HOT3D is not a procedural dataset, this per-sequence alignment gives VASOT a visual-similarity-matching advantage that our training-free method does not leverage, which marginally raises its scores on metrics such as MoF. ASFormer is trained with a 70-30 split using ground-truth annotations (VLM prompt in Appendix H; TSC/KTS parameters in Appendix E.1). As shown in Table 1, our method achieves higher segmentation F1 than all OakInk2 baselines. Temporal networks and alignment methods degrade in fine-grained scenarios where appearance is static but motion dynamics change (e.g., tightening vs. loosening), and VLMs lack native access to continuous kinematic signals needed for such micro-transitions. KTS and TSC rely on within-segment similarity or temporally regularized representations that can smooth brief transitions, whereas our explicit curvature and direction-change signals better align with fine hand-object boundaries.

The baselines (VASOT, ASFormer) differ from our method in both input modality and algorithm, so a win could reflect richer input rather than the pipeline itself. We isolate this by holding input fixed (2D via RAFT Teed & Deng (2020); 3D via point clouds/MotionCrafter) and varying only the algorithm (Euclidean $\rightarrow$ PCA $\rightarrow$ Grassmannian), refer appendix D.2.

For real world applicability we evaluation on a large egocentric procedural dataset (EPIC-Kitchens) Damen et al. (2020) in the appendix D.4.

We also provide a mechanism level visualization of segmentation in Fig. 3 in the appendix. Finally we conduct real world tests using a 3D scene flow estimator SOA that eliminates the need for tracked point cloud and record our observations in the Appendix D.1. **For Qualitative results refer to the supplementary video**

4.3 Ablations

Zero-Order vs. Second-Order Kinematics. We establish a baseline using direct zero-order distances (geodesic distance without parallel transport) for boundary detection. The zero-order baseline consistently fails to detect boundaries where the magnitude of motion is constant but the execution direction changes (e.g., transitioning from "wiping" to "placing"). Our combined second-order kinematic features ($\kappa_\tau + \theta_\tau$) capture both acceleration and directional transitions natively, improving overall segmentation F1@50 by **14.7%** and downstream clustering ARI by **4.00%** shown in Table 1, A qualitative example of this is visible in Figure 1. Quantitatively the performance of

Dataset	Method	F1@10 ↑	F1@50 ↑	mIoU ↑	MoF ↑	Edit Distance ↓	ARI ↑	NMI ↑
HOT3D	VASOT	0.46	0.237	0.264	**0.503**	17.2	0.389	0.666
HOT3D	Qwen3.6-27B	0.289	0.122	0.132	0.316	25.4	0.11	0.338
HOT3D	Temporal Subspace Clustering	0.173	0.041	0.247	0.342	166.4	0.175	0.487
HOT3D	Kernel Temporal Segmentation*	0.340	0.138	0.206	0.419	23.4	NA	NA
HOT3D	Zero-order (Distance)	0.624	0.318	0.338	0.495	12.5	0.414	0.676
HOT3D	**Ours** ($\kappa_\tau + \theta_\tau$)	**0.640**	**0.338**	**0.349**	0.501	**12.2**	**0.431**	**0.691**
OakInk2	VASOT	0.378	0.299	0.147	0.416	24.0	0.346	0.638
OakInk2	ASFormer	0.352	0.269	0.173	0.430	26.3	0.312	0.623
OakInk2	Qwen3.6-27B	0.467	0.324	0.203	0.329	25.6	0.196	0.538
OakInk2	Temporal Subspace Clustering	0.304	0.113	0.374	0.376	92.4	0.283	0.744
OakInk2	Kernel Temporal Segmentation*	0.529	0.213	0.278	0.459	22.1	NA	NA
OakInk2	Zero-order (Distance)	0.733	0.453	0.433	0.557	12.4	0.524	0.779
OakInk2	**Ours** ($\kappa_\tau + \theta_\tau$)	**0.749**	**0.520**	**0.479**	**0.600**	**10.7**	**0.545**	**0.787**

Table 1: Comparison with visual baselines and matched-input temporal segmentation baselines.Temporal Subspace Clustering and Kernel Temporal Segmentation use the same input as our method. *Since Kernel Temporal Segmentation does not have a clustering algorithm, we do not report clustering metrics for it.

Frechet Smoothing over different noise levels and efficacy of Jump Diffusion is shown in Appendix D.3 (Figure 2b).

Clustering. Refer the appendix C.3 for the ablation study.

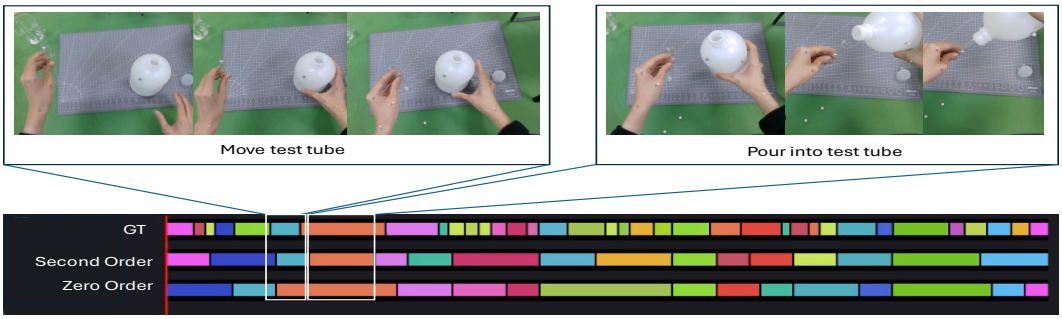


Figure 1: The figure shows zero order kinematics tending to under-segment fine motion such as move and pour into the same segment while second order kinematics is able to separate them.

5 Conclusion and Limitations

In this work, we introduced a fundamentally new geometric perspective for understanding fine-grained hand-object interactions. By lifting 3D scene flow into continuous trajectories on the Grassmann manifold, we demonstrated that capturing the intrinsic evolution of motion-specifically second-order kinematics is strictly necessary for resolving subtle execution boundaries where appearance remains static. To extract these features from noisy discrete manifolds, we proposed a novel preemptive jump-diffusion model that successfully resolves the smoothing-derivative paradox. This theoretical foundation enabled a causal, fully unsupervised pipeline that improves segmentation F1 and clustering over zero-order spatial models and Vision-Language baselines, while remaining **training-free and computationally lightweight (CPU-deployable)**.

Limitations. Our pipeline relies on the kinematic fidelity of upstream 3D scene flow, which egocentric estimators can degrade through noise and occlusion, and second-order kinematics are less sensitive to semantic changes with minimal motion. Segment grouping further presupposes persistent point identity and is thus confined to a single sequence (Section 2.6). However, our boundary detection, which requires no correspondence, is unaffected. Our training-free framework remains well-positioned to benefit from advances in 3D flow estimation.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Ali Shah Ali, Syed Ahmed Mahmood, Mubin Saeed, Andrey Konin, M. Zeeshan Zia, and Quoc-Huy Tran. Joint self-supervised video alignment and action segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10807–10818, October 2025.

Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Shangchen Han, Fan Zhang, Linguang Zhang, Jade Fountain, Edward Miller, Selen Basol, et al. Hot3d: Hand and object tracking in 3d from egocentric multi-view videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7061–7071. IEEE, 2025.

Kuo-Tsai Chen. Integration of paths–a faithful representation of paths by noncommutative formal power series. *Transactions of the American Mathematical Society*, 89(2):395–407, 1958.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4125–4141, 2020.

Tianjiao Ding, Shengbang Tong, Kwan Ho Ryan Chan, Xili Dai, Yi Ma, and Benjamin D Haeffele. Unsupervised manifold linearizing and clustering. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5427–5438. IEEE, 2023.

Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022.

Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11807–11816, 2019.

Sheng Li, Kang Li, and Yun Fu. Temporal subspace clustering for human motion segmentation. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4453–4461, 2015. doi: 10.1109/ICCV.2015.506.

Xiaolong Liu, Qimeng Wang, Yao Hu, Xu Tang, Shiwei Zhang, Song Bai, and Xiang Bai. End-to-end temporal action detection with transformer. *IEEE Transactions on Image Processing*, 31: 5427–5441, 2022a.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20981–20990. IEEE, 2022b.

Terry Lyons. Differential equations driven by rough signals (I): an extension of an inequality of L. C. Young. *Mathematical Research Letters*, 1(4):451 – 464, 1994. doi: 10.4310/MRL.1994.v1.n4.a5. URL https://doi.org/10.4310/MRL.1994.v1.n4.a5.

Xavier Pennec. Intrinsic statistics on riemannian manifolds: Basic tools for geometric measurements. *J. Math. Imaging Vis.*, 25(1):127–154, July 2006. ISSN 0924-9907. doi: 10.1007/s10851-006-6228-4. URL https://doi.org/10.1007/s10851-006-6228-4.

Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 540–555, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10599-4.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Rim Slama, Hazem Wannous, Mohamed Daoudi, and Anuj Srivastava. Accurate 3d action recognition using learning on the grassmann manifold. *Pattern Recognition*, 48(2):556–567, 2015. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2014.08.011. URL https://www.sciencedirect.com/science/article/pii/S0031320314003252.

Qwen Team. Qwen3. 6-27b: Flagship-level coding in a 27b dense model, 2026.

Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pp. 402–419. Springer, 2020.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Pavan Turaga, Ashok Veeraraghavan, Anuj Srivastava, and Rama Chellappa. Statistical computations on grassmann and stiefel manifolds for image and video-based recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(11):2273–2286, 2011.

David A Winter et al. *Biomechanics and motor control of human movement*, volume 4. wiley Hoboken, NJ, 2009.

Yue Xu, Yong-Lu Li, Zhemin Huang, Michael Xu Liu, Cewu Lu, Yu-Wing Tai, and Chi-Keung Tang. Egopca: A new framework for egocentric hand-object interaction understanding. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5250–5261. IEEE, 2023.

Fangqiu Yi, Hongyu Wen, and Tingting Jiang. Asformer: Transformer for action segmentation. In *The British Machine Vision Conference (BMVC)*, 2021.

Mihai Zanfir, Marius Leordeanu, and Cristian Sminchisescu. The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection. In *2013 IEEE International Conference on Computer Vision*, pp. 2752–2759, 2013. doi: 10.1109/ICCV.2013.342.

Xinyu Zhan, Lixin Yang, Yifei Zhao, Kangrui Mao, Hanlin Xu, Zenan Lin, Kailin Li, and Cewu Lu. OAKINK2: A dataset of bimanual hands-object manipulation in complex task completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 445–456, June 2024.

Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *European conference on computer vision*, pp. 492–510. Springer, 2022.

Hang Zhang, Xin Li, and Lidong Bing. Video-LLaMA: An instruction-tuned audio-visual language model for video understanding. In Yansong Feng and Els Lefever (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 543–553, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-demo.49. URL https://aclanthology.org/2023.emnlp-demo.49/.

Ruijie Zhu, Jiahao Lu, Wenbo Hu, Xiaoguang Han, Jianfei Cai, Ying Shan, and Chuanxia Zheng. Motioncrafter: Dense geometry and motion reconstruction with a 4d vae. *arXiv preprint arXiv:2602.08961*, 2026.

A Appendix: Extended Theoretical Analysis and Proofs

A.1 Extended Modeling: Noise and Bias-Variance

In Section 2.3, we introduced the Fréchet mean and the scale separation constraint. Here we provide the detailed formalism.

Bias-Variance Trade-off. Let the true underlying subspace at window τ be U_τ^*, and the observed subspace be $\hat{U}_\tau = \exp_{U_\tau^*}(\eta_\tau)$, where $\eta_\tau \in T_{U_\tau^*}\mathcal{G}$ is a random noise vector with bounded variance $\mathbb{E}[\|\eta_\tau\|^2] \leq \sigma^2$. We compute the Fréchet mean $\bar{U}_\tau$ over a window of size W_s.

Because the Grassmannian is a manifold of non-negative sectional curvature, the Fréchet mean variance decreases proportionally to the window size: $\mathbb{E}[d_g^2(\bar{U}_\tau, U_\tau^*)] \approx \mathcal{O}(\sigma^2/W_s)$. However, if the true motion accelerates with maximum curvature $\kappa_{\max}^*$, averaging introduces a temporal bias scaling quadratically: $\text{Bias}(\bar{U}_\tau) \approx \mathcal{O}(\kappa_{\max}^*(W_s\Delta t)^2)$. Balancing these yields the constraint $1 < W_s \ll \frac{1}{\Delta t}\sqrt{\frac{\sigma}{\kappa_{\max}^*}}$.

A.2 Proof of Theorem 3.1: Curvature Error Bound

Theorem 3.1 holds under the following assumptions:

(A1) **Bounded residual noise.** After Fréchet smoothing, the smoothed subspace lies within a bounded geodesic distance of the true subspace: $d_g(\bar{U}_\tau, U_\tau^*) \leq \epsilon$.

(A2) **Bounded sectional curvature.** The true continuous trajectory has bounded curvature, $\kappa_{\max}^* < \infty$; the constant C_2 depends on the maximum sectional curvature of $\mathcal{G}(k, n)$.

(A3) **No cut-locus crossing.** Consecutive subspaces satisfy $d_g(U_\tau, U_{\tau+1}) < \pi/2$, so the logarithmic map is uniquely defined and locally Lipschitz.

Remark A.1. *Assumption (A3) is guaranteed in practice by our pipeline: the high temporal density of the input stream keeps consecutive subspaces close ($\theta_i \ll \pi/2$), and the jump-diffusion gate (Algorithm 2, line 5) rejects any inter-window transition that would approach the cut locus, resetting the transport chain instead of differentiating across it.*

Proof. The empirical discrete curvature is given by $\hat{\kappa}_\tau = \frac{1}{\Delta t}\left\|\hat{\xi}_\tau - \mathcal{P}_{\bar{U}_{\tau-1}\to\bar{U}_\tau}\left(\hat{\xi}_{\tau-1}\right)\right\|_F$, where $\hat{\xi}_\tau = \log_{\bar{U}_\tau}(\bar{U}_{\tau+1})$. The total error can be decomposed into two parts using the triangle inequality:

$$|\hat{\kappa}_\tau - \kappa_\tau^*| \leq \underbrace{|\hat{\kappa}_\tau - \kappa_\tau^{\text{clean}}|}_{\text{Noise Perturbation Error}} + \underbrace{|\kappa_\tau^{\text{clean}} - \kappa_\tau^*|}_{\text{Discretization Error}} \tag{8}$$

where $\kappa_\tau^{\text{clean}}$ is the discrete curvature computed on the true, noise-free subspaces $\{U_\tau^*\}$.

Step 1: Discretization Error. The finite-difference approximation of the covariant derivative via parallel transport over a step size Δt incurs an error proportional to the squared step size: $|\kappa_\tau^{\text{clean}} - \kappa_\tau^*| \leq C_2(\Delta t)^2$.

Step 2: Noise Perturbation Error. Given $d_g(\bar{U}_\tau, U_\tau^*) \leq \epsilon$, the logarithmic map (a smooth diffeomorphism) is locally Lipschitz. The perturbation in the velocity vector $\hat{\xi}_\tau$ is proportional to the spatial perturbation: $\|\hat{\xi}_\tau - \xi_\tau^*\|_F \leq L_1\epsilon$. Furthermore, the parallel transport operator $\mathcal{P}_\to$ is an isometry. Applying the triangle inequality to the velocity difference yields a total velocity error bounded by $C_1\epsilon$. Since curvature is the rate of change of velocity, we divide by the temporal step Δt: $|\hat{\kappa}_\tau - \kappa_\tau^{\text{clean}}| \leq C_1\frac{\epsilon}{\Delta t}$.

Combining both bounds yields the final result. □

A.3 Proof of Theorem 3.2: Curvature Sensitivity vs. Zero-Order Distance

Proof. Consider two sequences of subspaces on $\mathcal{G}(k, n)$:

$$\gamma_1(\tau) = \exp_{U_0}(\tau V_1) \quad \text{(Constant motion)}$$
$$\gamma_2(\tau) = \exp_{U_0}(\tau V_2(\tau)) \quad \text{(Accelerating/turning motion)}$$

It is trivial to construct $V_2(\tau)$ such that the step-wise distance is identical: $d_g(\gamma_1(\tau), \gamma_1(\tau+1)) = d_g(\gamma_2(\tau), \gamma_2(\tau+1)) = C$. In an egocentric video, this means the hand is displacing the same volume of point clouds per frame, masking the boundary.

However, applying the discrete covariant derivative (Theorem 3.1):

$$\kappa_\tau^{(1)} = \|\nabla_{\dot\gamma_1}\dot\gamma_1\|_F = 0 \quad \text{(Geodesic flow)}$$
$$\kappa_\tau^{(2)} = \|\nabla_{\dot\gamma_2}\dot\gamma_2\|_F > 0 \quad \text{(Non-geodesic flow)}$$

The curvature operator κ_τ naturally breaks the degeneracy, reliably signaling the interaction state change. □

B Detailed Explanation of individual block of main algorithm

B.1 Feature κ_τ - Manifold Acceleration

Definition.

$$\kappa_\tau = \|\Delta\xi_\tau\|_F = \left\|\xi_\tau - \tilde{\xi}_{\tau-1}\right\|_F = \left\|\underbrace{\mathrm{Log}_{U_\tau}(U_{\tau+1})}_{\text{current velocity}} - \underbrace{\mathcal{P}_{U_{\tau-1}\to U_\tau}(\xi_{\tau-1})}_{\text{transported prev. velocity}}\right\|_F . \tag{9}$$

κ_τ is the **Frobenius norm of the tangent-space acceleration**: the rate of change of the Grassmann velocity vector, measured after aligning both vectors to the same tangent space via parallel transport.

Physical meaning. Think of the Grassmann trajectory as a curve on a curved surface. At each window step the curve has a "velocity" (the tangent vector ξ_τ pointing toward the next subspace). κ_τ measures how much that velocity changed from one step to the next analogous to acceleration in classical mechanics.

Why the Frobenius norm is sufficient for acceleration. The tangent vector $\xi_\tau \in \mathcal{T}_{U_\tau}\mathcal{G}$ is a matrix in $\mathbb{R}^{3N\times k}$. There are many ways to measure the size of a matrix; the question is whether $\|\cdot\|_F$ captures everything relevant about the acceleration signal.

Velocity alone ($\|\xi_\tau\|_F$) measures the *speed* at window τ - how far the subspace moves in one step. It does *not* detect direction changes: a hand moving at constant speed along a curve has $\|\xi_\tau\|_F = \text{const}$ even at a sharp corner.

Acceleration as $\Delta\xi$ fixes this by measuring how much the velocity *changed*, in both magnitude and direction, from the previous step. The change vector $\Delta\xi = \xi_\tau - \tilde{\xi}$ lies in $\mathcal{T}_{U_\tau}\mathcal{G}$ (both terms are in the same tangent space after transport).

What makes κ_τ large.

- **Abrupt directional change:** the hand suddenly changes which dimension of the point cloud it is sweeping (e.g. a grasp reversal). The angle between $\tilde{\xi}$ and ξ_τ is large, making $\left\|\xi_\tau - \tilde{\xi}\right\|_F$ large even if both norms are equal.
- **Abrupt speed change:** the hand decelerates sharply before contact or accelerates on release. Even if direction is constant, $\|\xi_\tau\|_F \neq \left\|\tilde{\xi}\right\|_F$ and $\kappa_\tau > 0$.
- **Both simultaneously:** maximum at a hard action transition.

Decomposition of κ_τ. Using $\Delta\xi = \xi_\tau - \tilde{\xi}$, we can decompose:

$$\begin{aligned}\kappa_\tau^2 = \|\Delta\xi\|_F^2 &= \|\xi_\tau\|_F^2 + \left\|\tilde{\xi}\right\|_F^2 - 2\langle\xi_\tau, \tilde{\xi}\rangle_F \\ &= \underbrace{\|\xi_\tau\|_F^2 + \left\|\tilde{\xi}\right\|_F^2}_{\text{speed terms}} - 2\,\|\xi_\tau\|_F \left\|\tilde{\xi}\right\|_F \cos\theta_\tau. \end{aligned} \tag{10}$$

This shows κ_τ encodes *both* θ_τ (direction) and $\|\xi\|_F$ (speed) simultaneously - it cannot separate them.

Limitation. Two scenarios produce the same κ_τ:

1. Constant speed, large direction change ($\theta \approx \pi$, $\|\xi_\tau\|_F = \left\|\tilde{\xi}\right\|_F$).
2. No direction change ($\theta = 0$), large speed change ($\|\xi_\tau\|_F \gg \left\|\tilde{\xi}\right\|_F$).

θ_τ breaks this ambiguity by isolating the directional component.

B.2 Feature θ_τ - Direction Change Angle

Definition.

$$\boxed{\theta_\tau = \arccos\left(\frac{\langle\tilde{\xi}_{\tau-1},\ \xi_\tau\rangle_F}{\left\|\tilde{\xi}_{\tau-1}\right\|_F \|\xi_\tau\|_F}\right) \in [0, \pi].} \tag{11}$$

θ_τ is the angle between the *transported previous velocity* and the *current velocity*, measured in the Frobenius inner product on the tangent space $\mathcal{T}_{U_\tau}\mathcal{G}$. Both vectors are in the same tangent space (transport ensures this), so the inner product is geometrically valid.

Physical meaning. θ_τ answers: **"by how many degrees did the motion direction rotate on the Grassmann manifold from the previous window to this one?"** It is a *pure geometric angle*, completely independent of how fast the hand is moving.

- If $\xi_\tau = \tilde{\xi}$: same direction, $\theta = 0$.
- If $\xi_\tau \perp \tilde{\xi}$: orthogonal direction, $\theta = \pi/2$.
- If $\xi_\tau = -\tilde{\xi}$: exact reversal, $\theta = \pi$.

Speed invariance - proof. Scale $\xi_\tau \to \lambda\xi_\tau$ for any $\lambda > 0$:

$$\theta_\tau = \arccos\left(\frac{\langle\tilde{\xi}, \lambda\xi_\tau\rangle_F}{\left\|\tilde{\xi}\right\|_F \|\lambda\xi_\tau\|_F}\right) = \arccos\left(\frac{\lambda\langle\tilde{\xi}, \xi_\tau\rangle_F}{\lambda\left\|\tilde{\xi}\right\|_F \|\xi_\tau\|_F}\right) = \arccos\left(\frac{\langle\tilde{\xi}, \xi_\tau\rangle_F}{\left\|\tilde{\xi}\right\|_F \|\xi_\tau\|_F}\right). \tag{12}$$

The λ cancels. θ_τ is invariant to the magnitude of both velocity vectors.

B.3 Why θ_τ and κ_τ Are Complementary

κ_τ alone is ambiguous. From the decomposition $\kappa_\tau^2 = \|\xi_\tau\|_F^2 + \left\|\tilde{\xi}\right\|_F^2 - 2\,\|\xi_\tau\|_F \left\|\tilde{\xi}\right\|_F \cos\theta_\tau$, the same κ value can arise from:

- Fast scrubbing (large $\|\xi\|_F$, $\theta \approx \pi$): $\kappa^2 \approx 4\,\|\xi\|_F^2$.
- Slow sudden reversal (small $\|\xi\|_F$, $\theta = \pi$): $\kappa^2 \approx 4\,\|\xi\|_F^2$.

These are physically different actions but produce identical κ.

θ_τ alone is undefined under near-zero velocity. When $\|\xi_\tau\|_F < \varepsilon$ (the hand is stationary), the cosine similarity is numerically unstable. The code sets $\theta = 0$ in this case, but κ correctly captures the near-zero velocity as a boundary signal when motion resumes.

Combined score $s_\tau = \alpha\theta_\tau/\pi + (1-\alpha)\kappa_\tau/\kappa_{p95}$. Using both:

1. θ fires on directional reversals regardless of speed.
2. κ fires on speed changes regardless of direction.
3. Together they fire on all action boundaries with high score, while interior frames (smooth motion) have low scores in both.

B.4 Additional Clustering algorithms

The main method of the paper is defined in 2.6, for additional helper algorithms, please refer to Algorithms 3 and 4.

Algorithm 3 ComputeSpatialSignature

Require: Segment bases $\mathcal{S}^{(i)}$, overlap counts $\{c_{i\tau}\}$, rank k
Ensure: Segment descriptor $\Phi^{(i)} \in \mathbb{R}^{2N}$

$\Phi^{(i)} \leftarrow \mathbf{0}$; $Z_i \leftarrow \sum_\tau c_{i\tau}$
for each window τ in $\mathcal{S}^{(i)}$ **do**
  $b_{\tau rn} \leftarrow \|u^{(n)}_{\tau,r}\|_2$ for all r, n ▷ Spatial participation
  $p_{\tau rn} \leftarrow b_{\tau rn} / \sum_{h,m} b_{\tau hm}$
  $q_r \leftarrow (r - \frac{1}{2})/k$ for $r = 1, \ldots, k$ ▷ Singular-mode order
  $\mathbf{a}_\tau \leftarrow \sum_r (1 - q_r) p_{\tau r,:}$
  $\mathbf{b}_\tau \leftarrow \sum_r q_r p_{\tau r,:}$
  $\Phi_\tau \leftarrow (\mathbf{a}_\tau, \mathbf{b}_\tau)$
  $\Phi^{(i)} \leftarrow \Phi^{(i)} + (c_{i\tau}/Z_i)\Phi_\tau$
end for
return $\Phi^{(i)}$

Algorithm 4 CompleteLinkage

Require: Segment similarity matrix $K \in \mathbb{R}^{M \times M}$, threshold τ_{B}
Ensure: Cluster labels $\ell_{1:M}$

$\mathcal{C} \leftarrow \{\{1\}, \ldots, \{M\}\}$ ▷ One cluster per segment
while $|\mathcal{C}| > 1$ **do**
  $(C, C') \leftarrow \underset{C \neq C' \in \mathcal{C}}{\operatorname{argmin}} \ \max_{i \in C, j \in C'} (1 - K_{ij})$
  if $\max_{i \in C, j \in C'} (1 - K_{ij}) > 1 - \tau_{\mathrm{B}}$ **then**
    break
  end if
  $\mathcal{C} \leftarrow (\mathcal{C} \setminus \{C, C'\}) \cup \{C \cup C'\}$
end while
Assign one label ℓ_i to every segment in each cluster
return $\ell_{1:M}$

C Additional Ablations

C.1 Alpha

We compare the α values shown in Table 2. Our method works well with $\alpha = 0.2$ across datasets.

α	**F1@50**
0.0	0.486
0.2	0.520
0.4	0.506
0.6	0.491
0.8	0.468
1.0	0.443

Table 2: Effect of α on F1@50

C.2 Clustering threshold (τ) sweep

τ_{B}	ARI ↑	NMI ↑
0.600	0.285	0.576
0.650	0.348	0.627
0.700	0.377	0.654
0.750	0.449	0.707
0.800	0.511	0.748
0.850	0.532	0.767
0.900	**0.545**	**0.787**
0.950	0.524	0.781
1.000	0.478	0.783

Table 3: Clustering-threshold sweep on OakInk2 using Tracked PointCloud input. The proposed method achieves its highest ARI and NMI at $\tau_{\mathrm{B}} = 0.9$ and remains robust to nearby threshold selections.

C.3 Clustering Alternates

To answer "Why this clustering method?" we have tried the following alternates to understand how the current method is better and other options that exist.

1. *Singleton or identity*: Each segment returned after segmentation is given their own label. As the method does not train with Ground Truth labels, we obtain the accuracy of Singleton label assignment with Hungarian Matching per sequence, following the protocol used to measure clustering efficacy.
2. *Projection Kernel + DTW clustering*: Each segment $\mathcal{S}^{(i)}$ is represented by a sequence of local Grassmannian subspaces. A structural similarity is first computed using the projection kernel
$$K_{ij}^{\mathrm{proj}} = \left\| \bar{U}_i^\top \bar{U}_j \right\|_F^2, \tag{13}$$
where $\bar{U}_i$ and $\bar{U}_j$ denote the dominant segment-level motion subspaces. Temporal evolution is captured by representing each segment as a sequence of principal-angle vectors $A^{(i)} \in \mathbb{R}^{L_i \times k}$ and computing the DTW distance
$$d_{\mathrm{DTW}}(i,j) = \min_{\pi} \sum_{(p,q)\in\pi} \|A_p^{(i)} - A_q^{(j)}\|_2, \tag{14}$$
which is converted into a temporal kernel
$$K_{ij}^{\mathrm{DTW}} = \exp\left(-\frac{d_{\mathrm{DTW}}(i,j)^2}{2\sigma^2} \right). \tag{15}$$
The final affinity matrix combines structural and temporal similarity,
$$K = \beta_K K^{\mathrm{proj}} + (1-\beta_K) K^{\mathrm{DTW}}, \tag{16}$$
and spectral clustering is applied to K to obtain segment groups. The resulting affinity matrix is clustered using spectral clustering, with the number of clusters selected via silhouette analysis.

3. *Participation based clustering*: The boundary detector first partitions the point-cloud sequence $P_t \in \mathbb{R}^{N\times 3}$ into intervals $I_i = [a_i, b_i)$. Unlike the proposed method, the subsequent grouping stage does not use Grassmannian coordinates, singular vectors, or spatial signatures. Instead, up to $2{,}048$ uniformly sampled point indices are retained throughout the sequence and represented by their average motion participation. For point n, let $v_{tn} = \|P_{t+1,n} - P_{t,n}\|_2$ denote the displacement magnitude and let $\mathcal{T}_i$ be the displacement rows within interval I_i. The participation descriptor is computed as

$$\bar{v}_{in} = \frac{1}{|\mathcal{T}_i|}\sum_{t\in\mathcal{T}_i} v_{tn}, \qquad e_{in} = \frac{\bar{v}_{in}}{\sum_{m=1}^{N}\bar{v}_{im} + 10^{-8}}, \tag{17}$$

where e_i encodes the fraction of total motion contributed by each indexed point. Similarity between intervals is measured using histogram intersection,

$$A_{ij} = \sum_{n=1}^{N} \min(e_{in}, e_{jn}), \tag{18}$$

and intervals are connected whenever $A_{ij} \geq \tau_{\mathrm{HI}}$ with $\tau_{\mathrm{HI}} = 0.90$. The final cluster assignments are obtained as connected components of the resulting affinity graph. This representation captures which points participate in a motion but discards temporal ordering, motion direction, and higher-order geometric structure.

4. *Spatial signature based clustering (Ours)*: This is the method described in our paper, in section 2.6

Table 4: Comparison of alternative segment clustering strategies on OakInk2.

Clustering method	**ARI** ↑	**NMI** ↑
Singleton (identity) clustering	0.478	0.783
Projection Kernel ($\beta = 1$)	0.314	0.625
DTW clustering ($\beta = 0$)	0.408	0.697
Participation-based clustering	0.512	0.763
Spatial signature based clustering (Ours)	0.545	0.787

D Additional Results

D.1 Real-World Applicability and Upstream Flow Estimation.

Segmentation performance is governed by the fidelity of the input scene flow: degradation from point clouds to estimated flow (especially egocentric) arises from noise, temporal inconsistency, and occlusion. We quantify this via kinematic SNR (kSNR), defined as the ratio of low-frequency motion energy to residual noise. We evaluate OakInk2 in both egocentric and allocentric views using ground-truth and estimated (MotionCrafter) scene flow to assess real-world performance. Table 5 shows across paired sequences, drops in kSNR strongly correlate with decreases in F1, indicating that performance is limited by input signal quality rather than the algorithm.

Kinematic SNR. To formally diagnose the performance discrepancy between exocentric and egocentric views on identical video sequences, we introduce a Kinematic Signal-to-Noise Ratio (kSNR) metric for 3D scene flow. We isolate the true motion "signal" using a temporal low-pass filter to extract the coherent trajectory $\bar{V}_t$, and define the "noise" as the high-frequency spatial residual. The kSNR is computed as the logarithmic ratio of the smoothed signal energy to the residual noise variance:

$$\mathrm{kSNR} = 10\log_{10}\left(\frac{\sum_t \|\bar{V}_t\|_F^2}{\sum_t \|V_t - \bar{V}_t\|_F^2}\right) \tag{19}$$

Table 6 reports the HOT3D comparison across ground-truth and MotionCrafter-estimated flow.

View	**Flow Type**	**F1@10↑**	**F1@50↑**	**Edit Dist↓**	**ARI↑**	**NMI↑**	**kSNR (dB)↑**
Allocentric	GT Flow	0.755	0.486	11.1	0.461	0.779	-
Allocentric	MotionCrafter	0.715	0.454	12.4	0.457	0.774	23.7
Egocentric	GT Flow	0.725	0.456	12.0	0.475	0.779	-
Egocentric	MotionCrafter	0.680	0.414	16.3	0.471	0.775	21.4
NA	GT PointCloud	0.749	0.520	10.7	0.545	0.787	-

Table 5: Performance comparison on second order kinematics across views, flow types, and distance metrics on the OakInk2 dataset. GT Flow refers to scene flow calculated only from observed pixels. MotionCrafter flow is generated via Zhu et al. (2026).

Flow Type	F1@10 ↑	F1@50 ↑	Edit Distance ↓	ARI ↑	NMI ↑	kSNR(dB) ↑
GT Scene Flow	0.588	0.305	15.2	0.339	0.695	-
MotionCrafter	0.517	0.273	19.7	0.283	0.652	20.1
GT PointCloud	0.64	0.338	12.2	0.431	0.691	-

Table 6: Comparison across flow types in egocentric view for HOT3D Dataset. All results are for second order kinematics.

D.2 Matched-Input Baselines

We evaluate Euclidean acceleration and PCA-based change-point detection on RAFT optical flow and 3D scene flow obtained from the method described in G.2 Ground Truth PointCloud Calculation.

We implement the below baselines to test against the modalities mentioned above -

- **Eucledian** - Compute frame-to-frame L2 distance between raw vectorized 2D/3D flow windows, then take the first finite difference as acceleration. This is the Euclidean analogue of our κ^τ, operating in $\mathbb{R}^{3N}$ rather than $\mathrm{G}(k, n)$. Expected to suffer from the catastrophic noise amplification characterized in Theorem 3.1.
- **PCA** - Run PELT change-point detection on the first k PCA components of the 2D/3D flow window. Tests whether Grassmannian manifold structure adds value over flat Euclidean PCA.

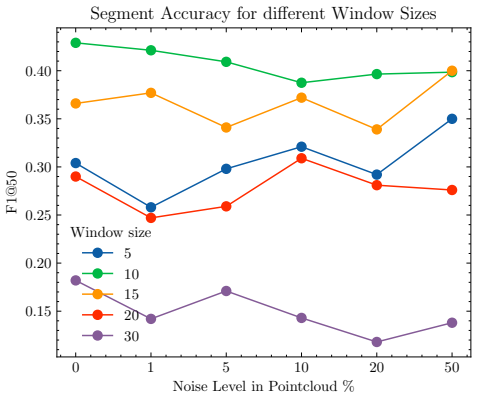
Method	F1@10 ↑	F1@50 ↑	mIoU ↑	MoF ↑	Edit Distance ↓	ARI ↑	NMI ↑
RAFT Euclidean	0.673	0.304	0.310	0.499	14.6	0.284	0.606
RAFT PCA	0.674	0.425	0.391	0.532	15.1	0.337	0.670
3D scene flow Euclidean	0.634	0.348	0.294	0.516	15.5	0.445	0.724
3D scene flow PCA	0.692	0.387	0.352	0.533	12.8	0.431	0.721
Ours (3D scene flow)	**0.749**	**0.520**	**0.479**	**0.600**	**10.7**	**0.545**	**0.787**

Table 7: Matched-input comparisons on OakInk2.

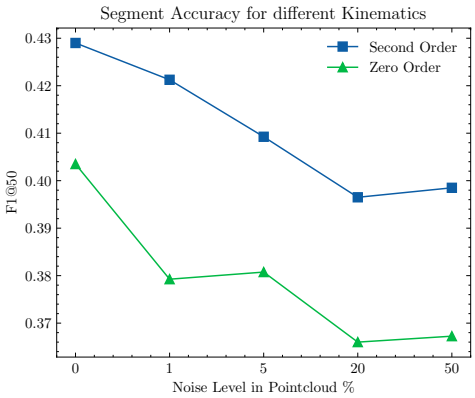
D.3 Jump-Diffusion and Fréchet Smoothing

The Smoothing-Derivative Trade-off: As theoretically established in Section 3.2, the Fréchet smoothing window W_s must satisfy a strict scale separation constraint to balance noise attenuation and boundary preservation. As illustrated in Figure 2a, we ablate the temporal window across $W_s \in \{5, 10, 15, 20, 30\}$ frames under varying levels of injected point-cloud noise. As predicted, segmentation performance follows an inverted-U distribution. At smaller windows ($W_s = 5$), the discrete parallel transport remains vulnerable to stochastic sensor noise, resulting in an F1@50 score of approximately 0.31. Conversely, as W_s increases past the physical micro-transition scale ($W_s \geq 20$), the temporal bias (phase blur) suppresses the curvature impulse (κ_τ), causing the framework to miss fine-grained boundaries entirely and plummeting the F1@50 score to 0.18 (for

$W_s = 30$). The optimal trade-off is consistently achieved at $W_s = 10$ (peaking at an F1@50 of 0.43), empirically validating our temporal scale separation hypothesis across all noise levels.



(a) Impact of the Fréchet smoothing window size (W_s) on segmentation accuracy (F1@50) under varying levels of injected point-cloud noise.



(b) The graceful degradation of Our second-order curvature metric, even at extreme 50% noise, empirically validates the theoretical error bounds established in Theorem 3.1. Also, to observe the trend, the F1 scores are averaged over smoothing radii 0,2,5 and 10.

Figure 2: Comparisons for various noise levels.

Robustness to Point Cloud Noise: Theorem 3.1 bounds the error of our discrete covariant derivative under spatial perturbations. To empirically validate this, we conduct a controlled noise injection study. We systematically apply varying levels of noise (from 0% to 50%) to the raw 3D scene flow vectors, simulating varying degrees of sensor error and spatial occlusion. As illustrated in Figure 2b, standard zero-order spatial comparisons (chordal distance) suffer an immediate performance drop when noise is introduced, falling from an F1@50 of approximately 0.40 to 0.36. In contrast, our second-order curvature framework maintains a distinct and consistent performance margin across all noise levels. Starting at an F1@50 of roughly 0.43, it degrades gracefully and stabilizes near 0.40 even under severe 50% noise injection. This persistent gap explicitly validates our theoretical claim: the combination of local Fréchet smoothing and mathematically stable parallel transport effectively insulates the second-order kinematic features from high-frequency spatial noise, matching the theoretical $\mathcal{O}(\epsilon)$ bound derived in our Appendix. Further results on Fréchet and Jump diffusion are available in Appendix section D.3

To simulate sensor dropouts, we corrupt 10% of frames and compare boundary F1 with and without jump-diffusion. With jump-diffusion, F1 decreases from 0.63 to 0.62, compared with a decrease from 0.62 to 0.58 without it.

Condition	Clean boundary F1	Boundary F1 at 10% corruption
Without jump-diffusion	0.62	0.58
With jump-diffusion	0.63	0.62

Table 8: Jump-diffusion under simulated sensor corruption on Oakink2.

We also perturb 3D point-cloud flow with increasing noise. Fréchet smoothing improves F1@50 at 50% noise, but does not improve it at 20% or 30%. This is consistent with a bias–variance trade-off: smoothing reduces noise variance while potentially attenuating boundary impulses.

Smoothing	20% point-cloud noise	30% point-cloud noise	50% point-cloud noise
Off	0.390	0.409	0.355
On	0.377	0.380	0.384

Table 9: Fréchet smoothing: F1@50 on OakInk2 under point-cloud noise.

D.4 EPIC-KITCHENS Evaluation

We evaluate 62 EPIC-KITCHENS test sequences using their existing action annotations, without re-annotation. RAFT and MotionCrafter provide estimated optical flow and scene flow, respectively. VASOT uses RGB input, TSC and KTS are given MotionCrafter inputs. The Euclidean and PCA controls operate on the corresponding estimated flow. As seen from the Edit distance, TSC generates highly fragmented predictions, hence marginally surpasses our method at F1@10. However, other metrics remain low as we increase the penalty of misalignment with Ground Truth in metrics F1@50 and ARI.

Method	Input	F1@10 ↑	F1@50 ↑	Edit Dist. ↓	mIoU ↑	MoF ↑	ARI ↑	NMI ↑
VASOT	RGB	0.449	0.355	95.1	0.216	0.418	0.223	0.568
TSC	MotionCrafter	**0.532**	0.412	124.5	0.313	0.375	0.187	0.581
KTS*	MotionCrafter	0.496	0.374	52.9	0.264	0.420	NA	NA
RAFT Euclidean	RAFT	0.421	0.380	49.9	0.213	0.442	0.228	0.554
RAFT PCA	RAFT	0.482	0.433	47.2	0.269	0.459	0.246	0.583
MotionCrafter Euclidean	MotionCrafter	0.466	0.414	48.5	0.238	0.438	0.231	0.567
MotionCrafter PCA	MotionCrafter	0.458	0.406	48.4	0.249	0.436	0.236	0.577
Ours	MotionCrafter	0.527	**0.452**	**43.8**	**0.316**	**0.461**	**0.248**	**0.607**

Table 10: EPIC-KITCHENS results under matched input baselines and comparison with VASOT, TSC and KTS. Since TSC and KTS natively do not use RGB, we use MotionCrafter Scene flow as input instead of Tracked PointCloud. *Similar to earlier evaluations, KTS is not evaluated on clustering.

Our F1@50 is 0.452, compared with 0.355 for VASOT, a relative improvement of approximately 27%.

D.5 Qualitative Results

Figure 3 shows κ and θ evolution alongside the cuts generated by the kinematic score.

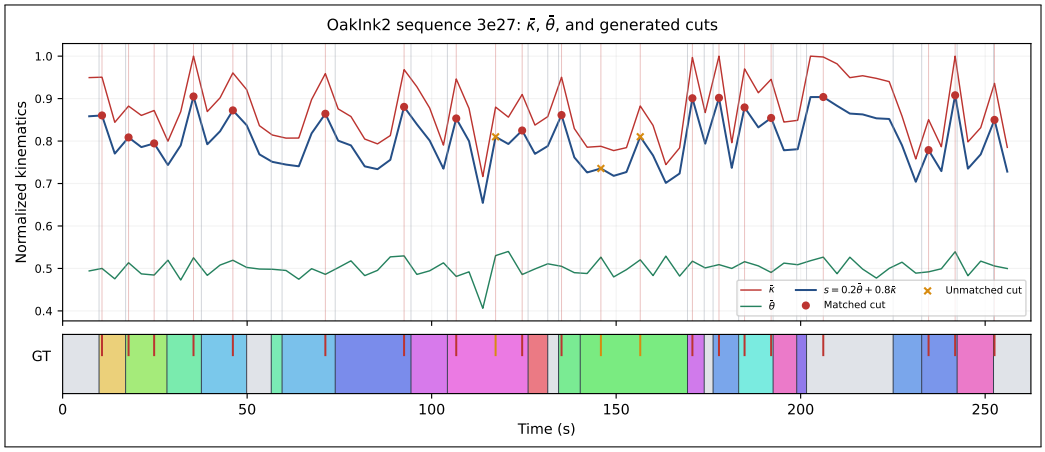


Figure 3: κ and θ evolution for an OakInk2 sequence. Top: normalized $\bar{\kappa}$, $\bar{\theta}$, their combined score, and detected cuts. Bottom: ground-truth action intervals and predicted cuts.

D.6 Observed statistical variation for the method

Table 11 gives the mean and Standard Deviation for our method on OakinkV2 to quantify the variation each measurement metric across sequences.

Metric	Value	Standard deviation
F1@10	0.749	0.064
F1@50	0.520	0.076
mIoU	0.479	0.034
MoF	0.600	0.035
Edit Distance	10.7	2.906
ARI	0.545	0.105
NMI	0.787	0.041

Table 11: Observed statistical variation for our method on OakinkV2

E Hyperparameter Summary

Table 12 describe our hyperparameters used in our algorithm.

Symbol	Config key	Default	Guidance
k	`K`	5	Subspace rank. Increase for richer motions; decrease for clean/simple actions.
W	`W`	10	Window used for the participation-clustering experiments.
α	`ALPHA`	0.2	Weight of θ vs. κ in boundary score. 1.0 = direction only; 0.0 = acceleration only.
τ_B	HI threshold	0.90	Complete linkage threshold

Table 12: Hyperparameters and tuning guidance.

E.1 Settings for the Additional Temporal Baselines

The Temporal Subspace Clustering (TSC) and Kernel Temporal Segmentation (KTS) results are included in main-paper Table 1. Both use identical 3D scene-flow features. KTS uses a linear kernel, no gamma, no standardization, $ncp_{\max} = 30$, $v_{\max} = 1.0$, $l_{\min} = 1$, and $l_{\max} = 100000$. TSC uses $r = 100$, $s = 6$, binary weights, $\lambda_1 = 0.01$, $\lambda_2 = 10$, $\alpha = 0.1$, $\eta = 1.0$, a maximum of 1,000 iterations, tolerance 10^{-5}, and 30 clusters.

F Compute Requirements

Our System operates on CPU without requirement of GPU. For experiments we use a system with 60vCPUs, 120GB of RAM

G Additional data generation framework

G.1 Annotations

To facilitate proper evaluation of motion segregation, we devise a re-annotation framework for OakInk2 Dataset. We choose sequences larger than 2 minutes, and sample the 30 fps videos at Stride 8. Each sequence ranges approximately between 1600 to 1000 frames. We then give the annotation instruction to the annotator - "Label segments with the motion type, and choose a different label when the motion or object changes". Using that instruction, the same process is repeated for HOT3D. All labels are a string values containing characters, numbers or underscores.

Annotation protocol and agreement. Two lab-member annotators who are not authors independently annotate motion segments using a custom in-house tool. An adjudicator re-marks disagreements. Labels follow a noun–verb taxonomy, and annotators introduce a new segment whenever the motion or interacting object changes. The reported annotation statistics are summarized below. Full boundary and label annotations are planned for release on acceptance.

Detail	OakInk2	HOT3D
Sequences	53	28 (evaluation split)
Frames per sequence	936–1,636	312–781
Unique labels	118 string labels	65 string labels
Taxonomy	Noun + verb	Noun + verb
Tool	Custom in-house	Custom in-house
Annotators	Two + adjudicator, non-author lab members	Two + adjudicator, non-author lab members
Conflict resolution	Adjudicator re-marks disagreements	Adjudicator re-marks disagreements
Boundary-IoU agreement	0.78 ± 0.09	0.71 ± 0.12
Average actions per sequence	~ 26 (18–34)	~ 23 (19–28)
Total segments	$\sim 1,373$	~ 633
Release	Full boundaries and labels on acceptance	Full boundaries and labels on acceptance

Table 13: Annotation details and inter-annotator agreement.

G.2 Ground Truth PointCloud and Scene Flow

We assume temporally consistent point clouds, as provided by the OakInk2 and HOT3D datasets, enabling direct computation of scene flow. As both datasets provide hand-object meshes at every timestamp. We sample a fixed set of points from the meshes to ensure fixed length of flow points, and point-to-point correspondence.

For Scene flow evaluations in Tables 5 and 6, to match the input modality with MotionCrafter, we map the full scene flow to a view - allocentric/egocentric and consider those points that are visible directly to the view. We eliminate the occluded points such as the ones occurring at the back of the object, invisible from the view via a z-buffer. We then map these flow points to an image grid 640x480 to be at parity with computed scene flow via State-of-art.

H VLM action segment prediction prompt

We use the following prompt to evaluate the vision-language model (VLM) on video understanding and action decomposition.

Role. Analyze a video and decompose it into a precise sequence of actions, producing structured annotations for each segment.

Step 1: Action Segmentation. Divide the input video into distinct temporal segments, where each segment corresponds to a single continuous *primitive action*. Segments must be temporally ordered and non-overlapping.

Step 2: Structured Annotation. Global annotation. Provide a holistic description of the video:

```
{
    "obj_list": ["List of all objects involved in the scene"],
    "description": "Detailed natural language summary of the entire video"
}
```

Local annotation. For each temporal segment, output a structured dictionary with the following schema:

```
{
```

```
    "start": Start time in seconds (float),
    "end": End time in seconds (float),
    "primitive": "High-level action label (or 'None')",
    "mode": "rh_main | lh_main | bh_main (or 'None')",
    "lh": "Left-hand atomic action (or 'None')",
    "rh": "Right-hand atomic action (or 'None')",
    "lh_obj": ["Object interacted with by left hand (or 'None')"],
    "rh_obj": ["Object interacted with by right hand (or 'None')"],
    "desc": "Detailed description of the segment (or 'None')"
}
```

Constraints.

- Do not simplify or omit labels.
- Maintain explicit left-hand (`lh`) and right-hand (`rh`) decomposition.
- Ensure exact key matching: `start`, `end`, `primitive`, `mode`, `lh`, `rh`, `lh_obj`, `rh_obj`, `desc`.

I Related Work

Appearance-driven and VLM temporal modeling. A large body of work models video understanding through representations that capture object identity, scene context, and semantic action labels. Supervised temporal architectures Zhang et al. (2022); Liu et al. (2022a); Yi et al. (2021) and self-supervised alignment method Tong et al. (2022) achieve strong performance when action boundaries align with visible semantic changes. Similarly, recent advances in Vision-Language Models (VLMs) Radford et al. (2021); Alayrac et al. (2022); Zhang et al. (2023) have significantly improved open-vocabulary recognition and semantic alignment. However, these methods are fundamentally driven by *what* is present and *where* it occurs. VLMs, in particular, primarily process videos as sparse sequences of static frames. While they excel at appearance and language alignment, they are fundamentally unequipped to track continuous, second-order kinematic evolution, rendering them largely blind to interaction modes that differ solely in execution dynamics (e.g., constant-speed reaching versus twisting). In a recent work proposed by Ali et al. (2025) (VASOT) focused on temporal consistency and marks zero order visual changes as boundaries contrary to our approach of having motion based segmentation in a self-supervised way. Similarly, supervised algorithms such as Yi et al. (2021) attempt to provide direct frame level prediction in context of vicinity. They struggle with temporal consistency, and fine-grained motion difference.

Unsupervised temporal segmentation and clustering. Kernel Temporal Segmentation (KTS) Potapov et al. (2014) detects change points by minimizing within-segment feature scatter in a kernel space, but does not cluster recurring segments into shared action classes. Temporal Subspace Clustering (TSC) Li et al. (2015) learns a temporally regularized dictionary and representation through iterative optimization before spectral clustering. This optimization is time-consuming, particularly for long sequences.

Egocentric hand-object interaction. Within the domain of HOI, large-scale datasets Liu et al. (2022b); Grauman et al. (2022) and 3D pose-estimation frameworks Xu et al. (2023); Hasson et al. (2019) emphasize object-centric and egocentric representations. While these approaches successfully incorporate 3D geometry and scene flow, they primarily focus on estimating spatial configuration or interaction semantics rather than explicitly addressing the unsupervised temporal segmentation of non-linear interaction dynamics.

Kinematic smoothing and temporal alignment. While some works attempt to extract raw kinematic features (velocity, acceleration) from 3D trajectories Zanfir et al. (2013), they fundamentally struggle with the amplification of high-frequency sensor noise during discrete differentiation Winter et al. (2009). Standard low-pass filters or continuous moving averages effectively suppress this noise but inadvertently blur the sharp kinematic impulses that define action boundaries.

Geometric motion representations and our departure. Geometric approaches provide a principled alternative by representing linear dynamical systems as points on Grassmann manifolds Turaga

et al. (2011); Slama et al. (2015). However, these classic approaches collapse sequences into a *single static subspace*, implicitly assuming global linearity. More recent work aims to flatten complex manifolds into unions of subspaces Ding et al. (2023); while effective for static clustering, this flattening destroys the intrinsic temporal evolution of the trajectory.

Summary of our departure: Across all these domains, existing representations are mathematically blind to the localized, second-order kinematic shifts that define fine-grained HOI. In this work, we depart from appearance-based and static subspace models entirely. By lifting noisy 3D scene flow into continuous Grassmannian trajectories and resolving the smoothing-derivative paradox via jump-diffusion, we extract the intrinsic *how* of motion, enabling causal boundary detection and unsupervised temporal clustering.

AI USE STATEMENT

A closed-source generative AI system was used solely for language editing, including grammar correction, improvements to phrasing, and enhancement of the readability of portions of this manuscript. The AI system was not used for idea generation, methodological design, theorem development, experimentation, data analysis, interpretation of results, figure generation, code generation, or manuscript drafting. All scientific content, algorithms, proofs, experiments, analyses, and conclusions were conceived, implemented, verified, and written by the authors.

REPRODUCIBILITY STATEMENT

The methods, algorithms, datasets, evaluation protocols, mathematical formulations, implementation details, hyperparameter settings, and ablation studies needed to reproduce and independently verify the reported findings are described in the main paper and appendices.