REASONMOTION: CONSEQUENCE-AWARE MOTION GENERATION VIA REWARD-DRIVEN COUNTERFACTUAL OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Counterfactual motion generation optimizes continuous interventions based on predicted downstream consequences rather than imitating reference trajectories. While essential for actionable sports coaching, existing diffusion models fail here because they require paired data of flawed and perfect executions. To address this, we propose ReasonMotion, a consequence-aware framework that formulates motion generation as reward-driven counterfactual optimization. Our two-stage pipeline couples a bounded stochastic remediation stage that refines an athlete's preparation without paired targets, with a prediction stage that generates improved continuations conditioned on this adjustment. To optimize this continuous multi-step process, we introduce visual Group Relative Policy Optimization (vis-GRPO) tailored for diffusion models to improve motion quality, alongside Sub-Trajectory Step Subsampling (STSS) to improve training efficiency while maintaining comparable motion quality. Empirical evaluations across multiple sports domains demonstrate that ReasonMotion improves both geometric fidelity and movement quality. Compared with the supervised backbone, it reduces average joint-position error from 152.53 to 142.67 mm and raises execution scores from 0.15 to 1.63 in figure skating, while increasing proximity-based hit rates from 60% to 80% and improving striking kinematics quality metrics in boxing. ReasonMotion enables a novel paradigm of consequence-aware AI coaching, bridging the gap between kinematic generation and actionable athletic guidance.

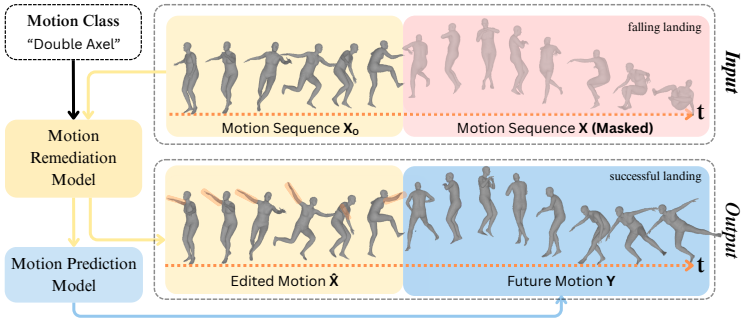


Figure 1: **Overview of ReasonMotion.** Given a motion class (e.g., "Double Axel") and an observed motion that leads to a failed outcome, the Motion Remediation Model edits the preparatory phase, adjusting the kinetic chain without a target execution. The Motion Prediction Model then forecasts the future motion this correction admits, converting the falling landing into a successful one.

1 INTRODUCTION

Human motion control has progressed rapidly across motion generation (Zhu et al., 2023; Liu et al., 2023a; Dai et al., 2024; Zhang et al., 2025) and motion editing (Tevet et al., 2022; Athanasiou et al., 2024; Fang et al., 2024; Li et al., 2025), with applications spanning animation, robotics, and

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.

sports (Zhu et al., 2023; Liu et al., 2023a; Dai et al., 2024; Zhao et al., 2025). While these models effectively support tasks like player tracking (Cioppa et al., 2022) and action quality assessment (Zhou et al., 2026), existing paradigms remain fundamentally tied to supervised imitation. Models can synthesize motions from text or edit sequences using paired source-target data, but they struggle with continuous counterfactual optimization where no ground-truth correction exists.

Actionable sports coaching (Ashutosh et al., 2025) serves as a premier domain to investigate this challenge. When a coach prescribes a kinematic adjustment, its value lies entirely in the consequence it produces. Automating this process requires a consequence-aware framework capable of modifying a flawed motion instance to maximize a future predicted outcome. However, acquiring paired corrective data in such domains is fundamentally unattainable. We cannot capture an athlete executing the exact same element identically except for a single fine-grained corrective adjustment at competition-level intensity (Ranganathan et al., 2020). Without paired targets, standard diffusion models merely regress toward the average training distribution and reproduce the very flaws they should eliminate. Because coaching demands the *best* outcome still reachable from an athlete's unique initiation, the task falls outside the reach of supervised matching and directly into the domain of reward-driven optimization.

To address this, we propose ReasonMotion, a consequence-aware framework that formulates continuous motion generation as reward-driven counterfactual optimization. The pipeline couples a bounded stochastic remediation stage (Meng et al., 2022) that refines a flawed initiation segment without paired targets, and a prediction stage that forecasts the improved consequence segment this adjustment admits. Crucially, the diffusion noise level t_0 acts as a continuous intervention dial, dynamically scaling modification strength while bounding the drift from the original kinematic state.

To optimize this generative process, we introduce visual Group Relative Policy Optimization (*vis*-GRPO). Continuous instance-conditioned generation creates a baseline estimation problem absent in prompt-level tasks. Because achievable outcomes are bounded by their initiations, absolute rewards are incomparable across instances. Standard reinforcement learning baselines fail to resolve this heterogeneity. Learned value networks in PPO (Schulman et al., 2017) often collapse to predicting a global constant, while running statistics in DDPO (Black et al., 2024) can conflate the intrinsic physical difficulty of an initiation with the actual quality of the policy. *vis*-GRPO resolves this structural flaw by estimating advantages exclusively among sibling trajectories generated from the identical condition, isolating the policy's relative improvement. Furthermore, to mitigate the computational bottleneck of long-horizon rollouts, we introduce Sub-Trajectory Step Subsampling (STSS), restricting policy updates to a subset of steps while preserving the full denoising trajectory. Our main contributions are summarized as follows:

- We propose ReasonMotion, a consequence-aware framework that formulates motion generation as reward-driven counterfactual optimization without paired targets, coupling bounded kinematic remediation with downstream consequence prediction.
- We introduce *vis*-GRPO to align diffusion trajectories with target outcome objectives through group-relative comparisons, alongside STSS to significantly reduce training computation while maintaining motion quality.
- Validated across general-purpose continuous generation and athletic domains, ReasonMotion achieves state-of-the-art performance on standard general motion benchmarks, while figure skating and boxing applications break the trade-off between geometric fidelity and execution quality, corroborated by national-level expert assessments.

2 Related Work

Human motion prediction forecasts future movement from observed motion, with compact MLP-based models providing a lightweight approach (Guo et al., 2023). Beyond forecasting continuations, motion generation synthesizes new sequences from conditions such as textual descriptions (Guo et al., 2022; Zhang et al., 2023; Cen et al., 2024), using architectures such as Mamba (Zeng et al., 2025) and text-conditioned diffusion (Zhang et al., 2024). Motion editing extends these capabilities to modifying existing sequences, using paired text-guided editing in MotionFix (Athanasiou et al., 2024) or a unified motion-condition-motion formulation in MotionLab (Guo et al., 2025). Within athletic motion modeling, ViMo (Qiu et al., 2024) uses video-guided generation, SMGDiff

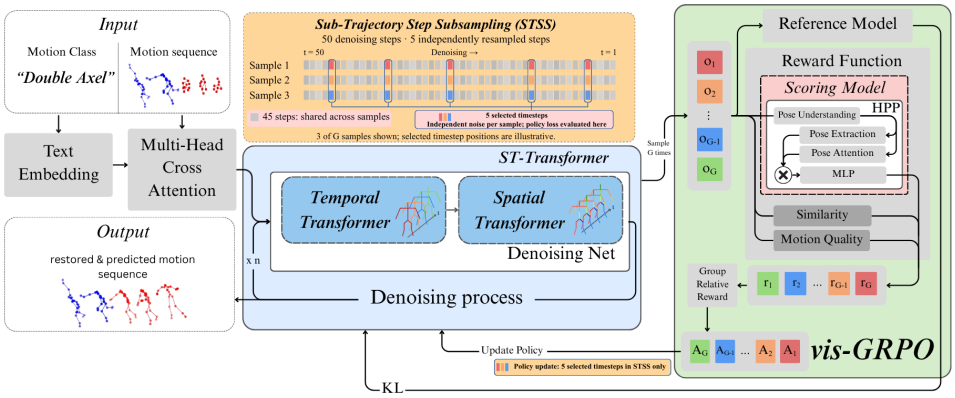


Figure 2: The ReasonMotion architecture consists of a ST-Transformer diffusion model and a vis-GRPO module for reward-guided refinement.

(Yang et al., 2025) develops soccer-specific diffusion, and Sports-Traj (Xu & Fu, 2025) models multi-agent movement trajectories.

Applying motion modeling to sports coaching further requires identifying useful adjustments to an athlete's existing execution. CoachMe (Yeh et al., 2025) generates sport-specific instructions by comparing learner and reference motions. ExpertAF (Ashutosh et al., 2025) produces actionable commentary and corrective visual demonstrations. These methods provide ways to communicate and visualize corrections. Our complementary focus optimizes adjustments based on predicted outcomes, rewarding execution quality over matching observed targets.

Reward-guided diffusion methods provide a basis for such outcome-oriented optimization. DDPO (Black et al., 2024) and DPOK (Fan et al., 2023) formulate reverse denoising as a multi-step decision process and apply policy-gradient optimization. GRPO (Shao et al., 2024) estimates advantages within groups sharing the same condition, avoiding a learned value network; DanceGRPO (Xue et al., 2025) and Flow-GRPO (Liu et al., 2026) extend this approach to visual diffusion and flow-based generation. AWM (Xue et al., 2026) instead connects reward optimization with the pretraining objective through advantage-weighted matching. Building on these approaches, our *vis*-GRPO applies condition-matched group comparisons to motion remediation and consequence prediction. STSS complements this optimization by reducing policy-update computation through timestep sampling without truncating the denoising rollout.

3 Method

3.1 Problem Formulation

We consider skills executed as discrete attempts that decompose into an actionable initiation phase and a downstream consequence phase. An attempt is a skeletal sequence $x \in \mathbb{R}^{K\times L\times C}$ over K joints, L frames, and C feature channels (with $C = 3$ for 3D coordinates), partitioned at frame L_i into sub-sequences handled by two specialized modules:

$$x = \Big(\underbrace{x^{\text{init}}_{1:L_i}}_{\text{Initiation segment}} , \underbrace{x^{\text{cons}}_{L_i+1:L}}_{\text{Consequence segment}} \Big), \tag{1}$$

paired with a semantic token S naming the intended element (e.g., triple lutz in figure skating).

The core task is continuous counterfactual generation, which we demonstrate here through actionable athletic coaching. Given a flawed attempt executed by a specific athlete, synthesize a corrected attempt that (i) remains recognisably that athlete's own motion and (ii) scores higher under domain-specific execution criteria. We split the generation into a remediator $\mathcal{R}_\phi$ over initiation segments and a predictor $\mathcal{F}_\psi$ over consequence segments:

$$\hat{x}^{\text{init}} \sim \mathcal{R}_\phi\big(\cdot \mid x^{\text{init}}_{\text{orig}}, S\big), \quad \hat{x}^{\text{cons}} \sim \mathcal{F}_\psi\big(\cdot \mid \hat{x}^{\text{init}}, S\big), \quad \hat{x} = \big(\hat{x}^{\text{init}}, \hat{x}^{\text{cons}}\big). \tag{2}$$

Both $\mathcal{R}_\phi$ and $\mathcal{F}_\psi$ are conditional diffusion policies sharing the same backbone architecture (Section 3.2) and optimized via *vis*-GRPO (Section 3.3). Structurally identical, they are differentiated by their operational roles: $\mathcal{R}_\phi$ acts as an intervention policy that minimally perturbs and re-denoises the flawed initiation (Section 3.6), whereas $\mathcal{F}_\psi$ is a forward model that predicts the resulting consequence segment conditioned on the remediated initiation (Section 3.7).

3.2 Spatio-Temporal Factorized Diffusion Backbone

Both stages use a spatio-temporally factorized Transformer as their backbone architecture, following CSDI (Tashiro et al., 2021) and its motion instantiation in TCD (Saadatnejad et al., 2023). The backbone consists of stacked residual blocks, each applying temporal attention across frames followed by spatial attention across joints to model motion dynamics and inter-joint relationships. To condition each block on the intended skill, we add cross-attention with semantic-token embeddings as keys and values and diffusion-time-aware motion features as queries. The backbone also retains TCD's spatio-temporal inpainting capability, supporting both partial re-denoising of an existing initiation (Section 3.6) and consequence generation conditioned on a fixed initiation (Section 3.7).

3.3 vis-GRPO

We optimize the conditional diffusion policies with *vis*-GRPO, adapting GRPO (Shao et al., 2024) to motion remediation and prediction. We use $\pi_\theta \in \{\mathcal{R}_\phi, \mathcal{F}_\psi\}$ to denote either the remediator or the predictor when describing their shared optimization procedure. Following prior reward-guided diffusion methods, we treat reverse denoising as a multi-step decision process and optimize a PPO-style clipped surrogate using a scalar reward evaluated on the final generated motion.

Unlike prompt-level generation, continuous instance-conditioned generation bounds achievable rewards by the intrinsic physical difficulty of the initiation $q = (x^{\text{init}}, S)$, causing absolute rewards to conflate policy quality with heterogeneous initial conditions. To cleanly isolate policy performance without a value network, we employ condition-matched group-relative advantage estimation. For each update, we sample G candidate motions $\{o_i\}_{i=1}^G$ from the old policy $\pi_{\theta_{\text{old}}}$ under the exact same condition q (representing $(x^{\text{init}}_{\text{orig}}, S)$ for $\mathcal{R}_\phi$, and $(\hat{x}^{\text{init}}, S)$ for $\mathcal{F}_\psi$). Letting $\mathbf{r} = (r_1, \ldots, r_G)$ denote their terminal rewards, we compute group-normalized advantages $A_i = (r_i - \text{mean}(\mathbf{r}))/(\text{std}(\mathbf{r}) + \delta)$. Because all candidates share the identical initialization, this shared-context normalization strips away instance-specific difficulty. The output-level form of the *vis*-GRPO objective is as follows:

$$J_{\text{vis-GRPO}}(\theta) = \mathbb{E}_{\substack{q\sim P(Q) \\ \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|q)}} \Bigg[\frac{1}{G} \sum_{i=1}^{G} \min \Bigg(\frac{\pi_\theta(o_i \mid q)}{\pi_{\theta_{\text{old}}}(o_i \mid q)} A_i, \\ \text{clip}\left(\frac{\pi_\theta(o_i \mid q)}{\pi_{\theta_{\text{old}}}(o_i \mid q)}, 1-\epsilon, 1+\epsilon \right) A_i \Bigg) \Bigg] - \beta D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}}). \tag{3}$$

Here, ϵ is the clipping parameter, and β controls the KL penalty relative to the reference policy π_{ref}.

For diffusion policies, we implement the policy-gradient update using reverse-transition likelihood ratios along sampled denoising trajectories rather than directly evaluating the marginal motion likelihood $\pi_\theta(o_i \mid q)$. Each candidate motion $o_i = x_0^i$ is generated by a reverse trajectory $\tau_i = (x_T^i, \ldots, x_0^i)$. Following DDPO (Black et al., 2024), we use a per-step clipped surrogate, sharing the terminal advantage A_i across transitions rather than clipping a single ratio for the full trajectory. The transition ratio is $\rho_{i,t}(\theta) = p_\theta(x_{t-1}^i \mid x_t^i, q)/p_{\theta_{\text{old}}}(x_{t-1}^i \mid x_t^i, q)$. With Gaussian transitions $\mathcal{N}(\mu_\theta(x_t^i, t, q), \sigma_t^2 \mathbf{I})$ whose variance is fixed across policies, the log-ratio becomes

$$\log \rho_{i,t}(\theta) = \frac{1}{2\sigma_t^2} \left(\left\| x_{t-1}^i - \mu_{\theta_{\text{old}}}(x_t^i, t, q) \right\|^2 - \left\| x_{t-1}^i - \mu_\theta(x_t^i, t, q) \right\|^2 \right). \tag{4}$$

Here, μ_θ is the reverse-transition mean. We sum the clipped surrogate terms over denoising steps, using rewards evaluated on the final generated motions.

3.4 SUB-TRAJECTORY STEP SUBSAMPLING (STSS)

Standard reward-guided diffusion models inject independent Gaussian noise at every step $t \in \{1, \ldots, T\}$ during the rollout of G candidates. Evaluating policy ratios and gradients across all T transitions incurs substantial training compute and memory overhead. To reduce this cost without truncating the reverse chain, we introduce STSS, which restricts policy updates to a subset of timesteps. For each training iteration, we uniformly sample an active subset of timesteps $\mathcal{S} \subset \{1, \ldots, T\}$ with $|\mathcal{S}| = K$.

During active steps ($t \in \mathcal{S}$), we inject independent noise $z_i \sim \mathcal{N}(0, I)$ for candidate i to promote trajectory divergence. During inactive steps ($t \notin \mathcal{S}$), we apply a *shared* noise $z_{\text{shared}} \sim \mathcal{N}(0, I)$ across all G candidates. Using shared noise as common random numbers (Glasserman, 2004), we avoid introducing additional independent noise differences across candidates during inactive steps. This concentrates candidate-specific noise injection at the active steps $\mathcal{S}$, while inactive transitions continue to propagate differences in candidate states.

All T reverse transitions are executed, but we evaluate the policy log-ratio and accumulate clipped surrogate terms only for the steps $t \in \mathcal{S}$. We scale the sampled sum by T/K, giving the subsampled policy surrogate

$$\hat{J}_{\text{STSS}}(\theta) = \frac{1}{G} \sum_{i=1}^{G} \frac{T}{K} \sum_{t \in \mathcal{S}} \min\Big(\rho_{i,t}(\theta) A_i,\ \text{clip}\big(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon\big) A_i \Big). \tag{5}$$

3.5 REWARD MODEL

Rewards for the individual stages combine three domain-specific terms:

$$r = \lambda_{\text{sim}}\, r_{\text{sim}} + \lambda_{\text{score}}\, r_{\text{score}} + \lambda_{\text{quality}}\, r_{\text{quality}}. \tag{6}$$

r_{score} is a learned execution-quality estimate. We use a Human Pose Perception (HPP) encoder (Yeh et al., 2025) that applies spatio-temporal graph convolution to joint and bone channels, followed by an MLP regression head over spatio-temporally pooled features. The scorer is trained with MSE against sport-specific score labels on the training split alone. The raw score is mapped onto $[0, 1]$ by $r_{\text{score}} = (s - s_{\min})/(s_{\max} - s_{\min})$, where $s_{\min}$ and $s_{\max}$ denote the sport-specific score bounds. The scorer is frozen throughout policy optimisation and evaluation. No test-set annotation is used for scorer training or policy optimisation.

r_{sim} penalizes L_2 deviation from ground-truth joint coordinates, and r_{quality} penalizes second-order finite differences, encouraging kinematic plausibility. Full formulations and sport-specific details are in Appendix C.

3.6 REMEDIATION STAGE: POLICY-GUIDED KINEMATIC REMEDIATION

We implement remediation using the partial diffusion mechanism of SDEdit (Meng et al., 2022), perturbing the original initiation and re-denoising it under the learned policy. Given the original initiation $x_{\text{orig}}^{\text{init}}$ and a designated noise level t_0, we construct

$$x_{t_0} = \sqrt{\bar{\alpha}_{t_0}}\, x_{\text{orig}}^{\text{init}} + \sqrt{1 - \bar{\alpha}_{t_0}}\, \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \mathbf{I}), \tag{7}$$

followed by reverse denoising $x_{t_0} \to x_{t_0-1} \to \cdots \to \hat{x}^{\text{init}}$ under $\mathcal{R}_\phi(\cdot \mid x_{\text{orig}}^{\text{init}}, S)$. This partial-diffusion formulation serves two design goals. First, initializing from the original motion rather than pure noise encourages preservation of the athlete's idiosyncratic kinematic characteristics, which is reinforced by the similarity reward. Second, the noise level t_0 acts as a continuous pedagogical dial controlling correction strength: smaller values confine modifications to fine-grained local adjustments, whereas larger values permit broader structural revisions to the technique.

The policy $\mathcal{R}_\phi$ is optimized with *vis*-GRPO (Section 3.3), sampling G refined candidate initiations $\{\hat{x}^{\text{init},i}\}_{i=1}^{G}$ from the perturbed state x_{t_0} and evaluating their relative advantages within the group. To evaluate the downstream effect of each candidate adjustment, each initiation is completed by the frozen supervised predictor, $\hat{x}^{\text{cons}} = \mathcal{F}_\psi^{\text{sft}}(\hat{x}^{\text{init}}, S)$, yielding a complete synthesized attempt $\hat{x} = (\hat{x}^{\text{init}}, \hat{x}^{\text{cons}})$. We then evaluate each candidate using the closed-loop objective:

$$R(\hat{x}^{\text{init}}) = \lambda_{\text{sim}} r_{\text{sim}}(\hat{x}^{\text{init}}) + \lambda_{\text{score}} r_{\text{score}}(\hat{x}) + \lambda_{\text{quality}} r_{\text{quality}}(\hat{x}^{\text{init}}). \tag{8}$$

The similarity and quality terms are computed exclusively on $\hat{x}^{\text{init}}$ to anchor kinematic identity against $x^{\text{init}}_{\text{orig}}$ and enforce kinematic plausibility, whereas the domain execution score r_{score} evaluates the entire synthesized attempt $\hat{x}$. We do not anchor the consequence phase to ground truth, preventing penalization of valid alternative executions.

3.7 Prediction Stage: Policy-Guided Kinematic Prediction

The predictor generates the consequence segment from the initiation segment. Given $x^{\text{init}}_{1:L_i}$ and the semantic token S, it produces the consequence segment $x^{\text{cons}}_{L_i+1:L}$. Because this kinematic mapping is inherently one-to-many, the semantic token S specifies the intended skill while accommodating valid execution variations. This forward consequence model adopts the spatio-temporal factorized backbone (Section 3.2). Unlike the remediation policy $\mathcal{R}_\phi$, which re-denoises a perturbed input, the forward model treats the initiation as a fixed prefix and predicts the remaining consequence frames.

We train two variants of the predictor with distinct objectives and operational roles:

Supervised training ($\mathcal{F}^{\text{sft}}_\psi$). We optimize the standard diffusion L_2 objective to reconstruct recorded consequence segments given $q = (x^{\text{init}}, S)$. This objective models naturally observed consequence segments, including failed or suboptimal executions, rather than explicitly favoring higher scores. The resulting model is kept frozen during training to supply realistic consequence rollouts for the remediation policy in Eq. 8.

Reward tuning ($\mathcal{F}^{\star}_\psi$). Starting from $\mathcal{F}^{\text{sft}}_\psi$, we apply *vis*-GRPO (Section 3.3) by treating the predictor as the policy π_θ. For each query $q = (x^{\text{init}}, S)$, we sample a group of G candidate consequence segments $\{\hat{x}^{\text{cons},i}\}_{i=1}^{G}$ to form synthesized attempts $\hat{x}^i = (x^{\text{init}}, \hat{x}^{\text{cons},i})$. We optimize these rollouts using the objective in Eq. 6, where r_{sim} penalizes kinematic deviation from the recorded consequence segment $x^{\text{cons}}_{\text{orig}}$, r_{score} evaluates the completed attempt $\hat{x}^i$, and r_{quality} enforces kinematic plausibility on the synthesized consequence segment $\hat{x}^{\text{cons},i}$. Reward tuning thus drives the predictor toward high-scoring consequence segments conditioned on the given initiation, rather than merely replicating the empirical distribution. The tuned model $\mathcal{F}^{\star}_\psi$ is reserved exclusively for generating idealized exemplars at inference and never supplies rollouts during the optimization of $\mathcal{R}_\phi$.

4 Experiments

4.1 Datasets

We evaluate on two sport-specific datasets and one general motion benchmark. **FineFS** (Yanli Ji & Gao, 2023) is a figure-skating dataset annotated with Grade of Execution (GOE) scores (-5 to $+5$). We extract and filter 24-joint jump skeletons via HybrIK (Li et al., 2021), splitting the data 8:1:1. **Boxing** is our custom multi-view sparring dataset. We reconstruct 25-joint skeletons and align them into opponent-centric 60-frame windows, yielding 3,317 training and 1,925 held-out punch segments split entirely by bout to prevent data leakage. **Human3.6M** (Ionescu et al., 2013) contains 3.6M poses across 15 actions, utilizing the standard 17-joint protocol (Mao et al., 2021).

4.2 Experimental Setup

On FineFS, an attempt is 70 frames: a 30-frame take-off prefix and a 40-frame predicted consequence, with the semantic token S carrying the rotation count (*double*, *triple*). On Human3.6M, we predict 100 consequence frames from a 25-frame prefix using standard action tokens. For boxing, the unit is the full 60-frame window; it is remediated in one piece rather than split at a specific L_i, and the outcome scorer evaluates that same continuous window (Appendix C.4).

We instantiate r_{score} according to each domain: a learned GOE estimator for FineFS, a reaction-based judge for boxing, and zero ($\lambda_{\text{score}} = 0$) for Human3.6M. All scorers are trained exclusively on their respective training splits and strictly frozen during *vis*-GRPO optimization to prevent test-set leakage (Appendices B, C).

4.3 Baselines and Metrics

On Human3.6M, we compare against VAE-based (Salzmann et al., 2022; Yuan & Kitani, 2020b;a; Mao et al., 2021; Dang et al., 2022) and diffusion-based (Wei et al., 2023; Chen et al., 2023; Barquero et al., 2023; Tian et al., 2024; Sun & Chowdhary, 2024; Saadatnejad et al., 2023) predictors. On FineFS, we carry forward the strongest of these — TransFusion (Tian et al., 2024), CoMusion (Sun & Chowdhary, 2024) and TCD (Saadatnejad et al., 2023)(reporting both the original model and our supervised backbone augmented with text conditioning). Against these we compare optimisers on a fixed backbone, holding reward, data and step budget constant: DDPO (Black et al., 2024), a PPO variant with a learned critic (Fan et al., 2023), and AWM (Xue et al., 2026).

We report A-MPJPE and F-MPJPE (mm), ADE, FDE, MMADE and MMFDE, GOE, and mean acceleration as a smoothness proxy. Unlike skating, which relies on official GOE rubrics, boxing lacks a standardized execution score. We instead evaluate kinematic technique measures: reach and peak wrist speed (↑), contact distance (↓), and hit rate within 30 cm (↑). Additionally, we report PFLOPs to measure the total training compute required to reach the evaluated checkpoint for each optimizer. Detailed formulations are in Appendix D. To assess execution quality in multiple modalities and architectures, we employ three GOE judges: Trans-R (a Transformer directly regressing GOE), Trans-D (a Transformer predicting categorical score distributions), and 2D CTR-GCN (trained purely on 2D keypoints from an external MMFS dataset (Liu et al., 2023b)). These judges are used only for evaluation, not to compute rewards during model training.

5 Results

5.1 Motion Prediction and Optimizer Efficacy

Table 1: Motion prediction results on Human3.6M. Comparison against selected baselines with the scoring reward disabled ($\lambda_{\text{score}} = 0$). The full evaluation is provided in Appendix F.

Model	ADE ↓	FDE ↓	MMADE ↓	MMFDE ↓
BeLFusion	0.372	0.474	0.473	0.507
CoMusion	0.350	0.458	0.494	0.506
TCD (supervised base)	0.331	0.431	0.482	0.503
ReasonMotion	**0.325**	**0.423**	**0.466**	**0.492**

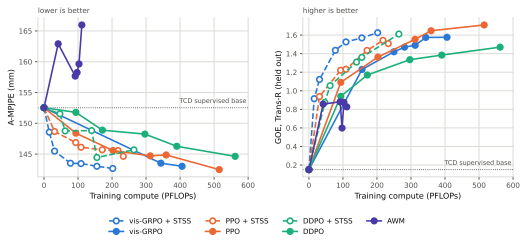


Figure 3: Cost-quality curves across policy-gradient estimators. The plots display A-MPJPE and Trans-R GOE against cumulative PFLOPs. All runs initialize from the identical TCD supervised backbone (dotted line).

ReasonMotion achieves state-of-the-art on the general-purpose Human3.6M benchmark (Table 1), outperforming every evaluated VAE and diffusion baseline while improving upon its supervised backbone across all metrics to confirm robust general prediction. Notably, these gains require no sport-specific execution reward, showing that the optimization framework extends beyond coaching to general motion.

On the athletic FineFS benchmark (Table 2, top), it significantly reduces endpoint kinematic error while driving GOE scores to 1.63 under Trans-R and 1.29 under Trans-D. Although mean acceleration is not the lowest compared to heavily smoothed baselines, ReasonMotion strikes the best balance between geometric fidelity and athletic execution quality.

Isolating the impact of reinforcement strategies on the identical TCD backbone reveals distinct structural trade-offs. As shown in Table 2, the supervised backbone merely imitates recorded attempts. While PPO optimizes the reward, it demands 3.9× our training compute to reach competitive performance. This inefficiency stems from its critic struggling to learn a timestep-dependent value structure. DDPO conflates initiation difficulty with policy advantage, and AWM suffers severe reward

Table 2: Comparison of prediction baselines and reinforcement strategies. All policy-gradient methods are evaluated under the identical TCD backbone and reward formulation to isolate the impact of the optimiser on computational efficiency, geometric error, and motion quality.

Method	PFLOPs ↓	A-MPJPE ↓	F-MPJPE ↓	GOE (Trans-R) ↑	GOE (Trans-D) ↑	Mean acc. ↓
TransFusion	–	163.79	165.84	-0.01 ± 1.95	0.37 ± 1.83	0.0146
CoMusion	–	161.15	186.50	0.51 ± 1.68	0.89 ± 1.36	0.0188
TCD	–	165.66	160.93	-0.11 ± 2.23	-0.14 ± 2.02	0.0465
TCD (ours, supervised)	–	152.53	159.68	0.15 ± 2.30	-0.06 ± 2.11	0.0445
+ PPO	796.1 (3.9×)	143.59	154.67	$\mathbf{1.85} \pm 1.56$	$\mathbf{1.38} \pm 1.41$	**0.0220**
+ PPO + STSS	217.7 (1.1×)	145.57	155.89	1.54 ± 1.64	1.15 ± 1.44	0.0241
+ DDPO	889.8 (4.4×)	144.59	156.15	1.57 ± 1.63	1.19 ± 1.43	0.0357
+ DDPO + STSS	264.3 (1.3×)	145.70	**154.41**	1.61 ± 1.60	1.19 ± 1.42	0.0416
+ AWM	91.7 (0.5×)	157.66	175.71	0.88 ± 1.61	1.06 ± 1.37	0.0257
+ *vis*-GRPO	343.4 (1.7×)	143.53	155.01	1.57 ± 1.61	1.22 ± 1.44	0.0261
+ *vis*-GRPO + STSS (ours)	**202.1 (1.0×)**	**142.67**	156.65	1.63 ± 1.66	1.29 ± 1.48	0.0227

Table 3: Comparisons of generated motions under different noise levels (t_0) and pipeline setups (Supervised/SFT vs. *vis*-GRPO for remediation and prediction stages). Metrics include GOE, mean acceleration, and A-MPJPE for the initiation (Init.) and consequence (Cons.) segments.

			GOE ↑			A-MPJPE ↓		
t_0	Rem.	Pred.	Trans-R	Trans-D	2D CTR-GCN	Init.	Cons.	Acc. ↓
5	SFT	SFT	$0.191 \pm .003$	$-0.052 \pm .016$	$-0.230 \pm .017$	54.06	160.74	0.0590
5	*vis*-GRPO	SFT	$0.572 \pm .009$	$0.307 \pm .029$	$-0.089 \pm .011$	53.44	160.68	0.0460
10	SFT	SFT	$0.195 \pm .063$	$-0.045 \pm .062$	$-0.269 \pm .033$	73.44	169.20	0.0519
10	*vis*-GRPO	SFT	$0.752 \pm .043$	$0.517 \pm .044$	$-0.031 \pm .026$	72.42	170.81	0.0417
15	SFT	SFT	$0.146 \pm .050$	$-0.081 \pm .051$	$-0.393 \pm .009$	91.27	178.25	0.0500
15	*vis*-GRPO	SFT	$0.866 \pm .038$	$0.643 \pm .032$	$-0.048 \pm .011$	89.02	180.01	0.0409
15	*vis*-GRPO	*vis*-GRPO	$\mathbf{2.302} \pm .029$	$\mathbf{1.923} \pm .027$	$\mathbf{0.457} \pm .011$	89.02	**169.39**	**0.0409**

hacking that destroys kinematic plausibility, yielding an A-MPJPE worse than even the unoptimized baseline (see Appendix G.3 for detailed analysis).

In contrast, *vis*-GRPO structurally resolves these credit assignment issues. By comparing sibling trajectories generated from the identical noise state, it completely bypasses the need for a separate critic, isolating the policy's action advantage from the inherent difficulty of the timestep. This mechanistic superiority is evident in the cost-quality curves (Figure 3). Across the evaluated compute range, *vis*-GRPO achieves a more favorable cost-quality trade-off than PPO, maintaining lower geometric errors (A-MPJPE) while reaching comparable execution quality (Trans-R GOE) at a substantially lower computational cost. Furthermore, STSS creates a unique synergy with this group-relative formulation. Because shared noise at inactive steps ensures candidates differ only at the sampled steps, STSS sharpens the group baseline. As Figure 3 illustrates, STSS shifts our optimization trajectory sharply to the left, enhancing *vis*-GRPO's efficiency without sacrificing stability, whereas it degrades PPO's maximum execution quality. Ultimately, this combination offers a favorable cost-quality trade-off in Table 2: competitive GOE (+1.63 Trans-R) with uncompromised geometric fidelity (142.67 mm A-MPJPE) at a fraction of standard computational costs.

5.2 Kinematic Remediation and the Full Pipeline

Evaluating the remediation stage requires isolating the reinforcement objective from stochastic diffusion. Under a purely supervised policy, re-denoising merely returns the take-off to its flawed distribution (Trans-R ≤ 0.195). Conversely, our policy pinpoints the essential kinematic cues while remaining closer to the athlete's original initiation than SFT across all t_0 (yielding lower Initiation A-MPJPE), while consistently elevating execution quality under all t_0. Crucially, in skating, GOE is dominated by landing and flight; yet here, the downstream predictor ($\mathcal{F}_{\psi}^{\text{sft}}$) is entirely frozen and supervised. Refining the initiation alone improves predicted execution quality under the frozen predictor, supporting our consequence-aware remediation approach.

Table 3 demonstrates the benefits of activating *vis*-GRPO across both pipeline stages. As shown in the final row ($t_0 = 15$), applying *vis*-GRPO to both stages raises execution scores to 2.302 Trans-R and 1.923 Trans-D (surpassing both single-stage remediation and standalone forward prediction of 1.63 and 1.29 in Table 2), while improving consequence geometry (reducing Consequence A-MPJPE from 180.01 to 169.39 mm). Consistent improvements across judges support these gains (Appendix C.5), while preserved outcome variance reflects generative diversity (Appendix I).

5.3 HUMAN EVALUATION

Table 4: Human Evaluation Results. Full protocol, evaluator credentials, and per-dimension agreement are reported in Appendix J.

Model	Realism	Expertise	Kinematic Plausibility			Utility
	(Select %)	(Top-1 %)	Rot. Err.↓	Flight(1–5)↑	Land.(1–5)↑	(Helpful %)
Reference Reconstructed Motion	53.3%	16.2%	$0.35 \pm .71$	$3.40 \pm .97$	3.00 ± 1.18	-
TransFusion	3.3%	1.2%	$1.80 \pm .90$	$1.75 \pm .70$	$1.85 \pm .73$	-
CoMusion	3.3%	0.0%	$2.05 \pm .63$	$1.75 \pm .77$	$1.65 \pm .85$	-
ReasonMotion	**94.7%**	**82.5%**	$\mathbf{0.33 \pm .43}$	$\mathbf{3.85 \pm .85}$	$\mathbf{3.85 \pm .91}$	**82.5%**

* '-' denotes not applicable; utility is only evaluated for ReasonMotion outputs.

We evaluate the full pipeline with figure skating experts, blind to model identity. As shown in Table 4, ReasonMotion achieves the strongest results across the comparative evaluation dimensions. It was selected as real in 94.7% of presentations and ranked highest in execution quality in 82.5%, against 3.3% and at most 1.2% for the diffusion baselines, whose rotation-count errors of 1.80 and 2.05 indicate substantial deviations in rotation count. Across eight corrective cases, 82.5% of evaluator responses rated the proposed corrections as helpful, providing an assessment of coaching utility beyond geometric metrics. Comparison videos are provided in the supplementary material.

5.4 CROSS-SPORT TRANSFER

Table 5: Experimental results on the boxing dataset demonstrate ReasonMotion's generalization beyond skating. Metric definitions and reward design are detailed in Appendix C.4.

Setting	Reach (m) ↑	Wrist speed (m/s) ↑	Hit dist. (m) ↓	Hit rate $\leq$30 cm ↑	Mean acc.
Supervised baseline	0.533	6.91	0.287	0.60	0.098
ReasonMotion	**0.603**	**7.77**	**0.207**	**0.80**	0.094

Boxing demonstrates that ReasonMotion generalizes beyond skating. We adapt the outcome scorer and reward design, and the same remediation stage improves the punch across the reported technique measures. As shown in Table 5, reach rises from 0.533 to 0.603 m, peak wrist speed from 6.91 to 7.77 m/s, contact distance falls from 0.287 to 0.207 m, and the rate of landing within 30 cm rises from 0.60 to 0.80, on held-out bouts containing no boxer seen in training. What a new sport requires is not a new architecture but a domain-appropriate scorer with the sport-specific reward design.

6 CONCLUSION

We presented ReasonMotion, a framework formulating continuous motion generation as reward-driven counterfactual optimization without paired targets. By coupling bounded kinematic interventions with downstream consequence prediction, the pipeline refines observed motions based on anticipated outcomes. To optimize this process, *vis*-GRPO cleanly isolates policy advantage from intrinsic instance difficulty without unstable value networks, while STSS significantly reduces training computation. Validated across standard benchmarks and athletic domains, ReasonMotion achieves state-of-the-art performance and breaks the trade-off between geometric fidelity and execution quality. ReasonMotion not only demonstrates profound utility in actionable sports coaching but further bridges the gap between kinematic generation and downstream consequence prediction, extending reward-guided diffusion to continuous counterfactual optimization.

AI USE STATEMENT

We used generative AI tools to aid and polish the writing of the manuscript, to assist with information retrieval and discovery, including identifying related work, and to support research ideation and execution. The authors reviewed AI-assisted text for accuracy and appropriateness and verified the relevant sources before incorporating them into the manuscript. We take responsibility for the final content of this work, including all text, claims, results, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our expert evaluation protocol was reviewed and approved by the Institutional Review Board (IRB) of our home institution through an expedited review process. The study involved minimal-risk questionnaire-based assessments of generated motions. All participants were informed of the study's purpose and tasks and provided consent before participating. Participation was voluntary, with no financial compensation, as evaluators participated as domain expert collaborators. Evaluator names and roles were collected solely for duplicate prevention and session resumption; participants were informed that their identities would not be linked to their responses, as described in Appendix J.1.

REPRODUCIBILITY STATEMENT

We submit the source code and generated motion videos as supplementary materials to support reproducibility and qualitative inspection. Section 3 describes the two-stage framework, *vis*-GRPO, and STSS, with training pseudocode in Appendix A. Sections 4.1–4.2 describe the datasets, preprocessing, data splits, and experimental setup. Appendix B reports model and training hyperparameters; Appendices C and D detail reward construction, scoring models, and evaluation metrics. Appendix G.1 specifies checkpoint selection and computational cost accounting. Appendix J documents the expert evaluation protocol, including evaluator qualifications, stimulus selection, blinding, randomization, and assessment procedures.

REFERENCES

Kumar Ashutosh, Tushar Nagarajan, Georgios Pavlakos, Kris Kitani, and Kristen Grauman. Expertaf: Expert actionable feedback from video. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13582–13594, 2025. doi: 10.1109/CVPR52734.2025.01268.

Nikos Athanasiou, Alpár Cseke, Markos Diomataris, Michael J Black, and Gül Varol. Motionfix: Text-driven 3d human motion editing. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

German Barquero, Sergio Escalera, and Cristina Palmero. Belfusion: Latent diffusion for behavior-driven human motion prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2317–2327, 2023.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024.

Zhi Cen, Huaijin Pi, Sida Peng, Zehong Shen, Minghui Yang, Shuai Zhu, Hujun Bao, and Xiaowei Zhou. Generating human motion in 3d scenes from text descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1855–1866, June 2024.

Ling-Hao Chen, Jiawei Zhang, Yewen Li, Yiren Pang, Xiaobo Xia, and Tongliang Liu. Humanmac: Masked motion completion for human motion prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9544–9555, 2023.

Anthony Cioppa, Silvio Giancola, Adrien Deliège, Le Kang, Xin Zhou, Zhiyu Cheng, Bernard Ghanem, and Marc Van Droogenbroeck. Soccernet-tracking: Multiple object tracking dataset

and benchmark in soccer videos. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 3490–3501, 2022. doi: 10.1109/CVPRW56347.2022.00393.

Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Motionlcm: Real-time controllable motion generation via latent consistency model. In *European Conference on Computer Vision*, pp. 390–408. Springer, 2024.

Lingwei Dang, Yongwei Nie, Chengjiang Long, Qing Zhang, and Guiqing Li. Diverse human motion prediction via gumbel-softmax sampling from an auxiliary space. In *Proceedings of the 30th ACM international conference on multimedia*, pp. 5162–5171, 2022.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. DPOK: Reinforcement learning for fine-tuning text-to-image diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Qihang Fang, Chengcheng Tang, Bugra Tekin, and Yanchao Yang. CigTime: Corrective instruction generation through inverse motion editing. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3237. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/b9079234614422c036c5a92bb8ec04c4-Abstract-Conference.html.

Paul Glasserman. *Monte Carlo Methods in Financial Engineering*, volume 53. Springer Science & Business Media, 2004.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5152–5161, June 2022.

Wen Guo, Yuming Du, Xi Shen, Vincent Lepetit, Xavier Alameda-Pineda, and Francesc Moreno-Noguer. Back to mlp: A simple baseline for human motion prediction. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 4809–4819, 2023.

Ziyan Guo, Zeyu Hu, Na Zhao, and De Wen Soh. Motionlab: Unified human motion generation and editing via the motion-condition-motion paradigm. *International Conference on Computer Vision (ICCV)*, 2025.

Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013.

Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3383–3393, 2021.

Zhengyuan Li et al. SimMotionEdit: Text-based human motion editing with motion similarity prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *Advances in neural information processing systems*, 38:40783–40818, 2026.

Shaowei Liu, Yang Zhou, Jimei Yang, Saurabh Gupta, and Shenlong Wang. Contactgen: Generative contact modeling for grasp generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20609–20620, 2023a.

Sheng-Lan Liu, Yu-Ning Ding, Gang Yan, Si-Fan Zhang, Jin-Rong Zhang, Wen-Yue Chen, and Xue-Hai Xu. Fine-grained action analysis: A multi-modality and multi-task dataset of figure skating. *arXiv preprint arXiv:2307.02730*, 2023b.

Wei Mao, Miaomiao Liu, and Mathieu Salzmann. Generating smooth pose sequences for diverse human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13309–13318, 2021.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022.

Liangdong Qiu, Chengxing Yu, Yanran Li, Zhao Wang, Haibin Huang, Chongyang Ma, Di Zhang, Pengfei Wan, and Xiaoguang Han. Vimo: Generating motions from casual videos. *arXiv preprint arXiv:2408.06614*, 2024.

Rajiv Ranganathan, Mei-Hua Lee, and Karl M. Newell. Repetition without repetition: Challenges in understanding behavioral flexibility in motor skill. *Frontiers in Psychology*, 11:2018, 2020.

Saeed Saadatnejad, Ali Rasekh, Mohammadreza Mofayezi, Yasamin Medghalchi, Sara Rajabzadeh, Taylor Mordan, and Alexandre Alahi. A generic diffusion-based approach for 3d human pose prediction in the wild. In *International Conference on Robotics and Automation (ICRA)*, 2023.

Tim Salzmann, Marco Pavone, and Markus Ryll. Motron: Multimodal probabilistic human motion forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6457–6466, 2022.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Jiarui Sun and Girish Chowdhary. Comusion: Towards consistent stochastic human motion prediction via motion diffusion. In *European conference on computer vision*, pp. 18–36. Springer, 2024.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in neural information processing systems*, 34:24804–24816, 2021.

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model, 2022. URL https://arxiv.org/abs/2209.14916.

Sibo Tian, Minghui Zheng, and Xiao Liang. Transfusion: A practical and effective transformer-based diffusion model for 3d human motion prediction. *IEEE Robotics and Automation Letters*, 9(7):6232–6239, 2024. doi: 10.1109/LRA.2024.3401116.

Dong Wei, Huaijiang Sun, Bin Li, Jianfeng Lu, Weiqing Li, Xiaoning Sun, and Shengxiang Hu. Human joint kinematics diffusion-refinement for stochastic motion prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6110–6118, 2023.

Yi Xu and Yun Fu. Sports-traj: A unified trajectory generation model for multi-agent movement in sports. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=9aTZf71uiD.

Shuchen Xue, Chongjian Ge, Shilong Zhang, Yichen Li, and Zhi-Ming Ma. Advantage weighted matching: Aligning rl with pretraining in diffusion models, 2026. URL https://arxiv.org/abs/2509.25050.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrpo: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

Hongdi Yang, Chengyang Li, Zhenxuan Wu, Gaozheng Li, Jingya Wang, Jingyi Yu, Zhuo Su, and Lan Xu. Smgdiff: Soccer motion generation using diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11807–11817, 2025.

Huili Huang Lijing Mao Yang Zhou Yanli Ji, Lingfeng Ye and Lingling Gao. Localization-assisted uncertainty score disentanglement network for action quality assessment. In *ACM MM*, pp. 1–10, 2023.

Wei-Hsin Yeh, Yu-An Su, Chih-Ning Chen, Yi-Hsueh Lin, Calvin Ku, Wenhsin Chiu, Min-Chun Hu, and Lun-Wei Ku. Coachme: Decoding sport elements with a reference-based coaching instruction generation model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29126–29151, 2025.

Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, pp. 346–364. Springer, 2020a.

Ye Yuan and Kris M Kitani. Diverse trajectory forecasting with determinantal point processes. In *International Conference on Learning Representations*, 2020b.

Ling-An Zeng, Guohong Huang, Gaojie Wu, and Wei-Shi Zheng. Light-t2m: A lightweight and fast model for text-to-motion generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 9797–9805, 2025.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14730–14740, 2023.

Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *IEEE transactions on pattern analysis and machine intelligence*, 46(6):4115–4128, 2024.

Zeyu Zhang, Yiran Wang, Wei Mao, Danning Li, Rui Zhao, Biao Wu, Zirui Song, Bohan Zhuang, Ian Reid, and Richard Hartley. Motion anything: Any to motion generation. *arXiv preprint arXiv:2503.06955*, 2025.

Zhonghan Zhao, Wenhao Chai, Shengyu Hao, Wenhao Hu, Guanhong Wang, Shidong Cao, Mingli Song, Jenq-Neng Hwang, and Gaoang Wang. A survey of deep learning in sports applications: Perception, comprehension, and decision. *IEEE Transactions on Visualization and Computer Graphics*, 31(10):9368–9386, 2025.

Kanglei Zhou, Ruizhi Cai, Liyuan Wang, Hubert PH Shum, and Xiaohui Liang. A comprehensive survey of action quality assessment: Method and benchmark. *Pattern Recognition*, pp. 113933, 2026.

Wentao Zhu, Xiaoxuan Ma, Dongwoo Ro, Hai Ci, Jinlu Zhang, Jiaxin Shi, Feng Gao, Qi Tian, and Yizhou Wang. Human motion generation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(4):2430–2449, 2023.

A VIS-GRPO ALGORITHM

In this appendix, we provide the complete training procedure for policy-guided motion prediction using *vis*-GRPO with Sub-Trajectory Step Subsampling (STSS). Algorithm 1 details the rollout generation with Common Random Numbers (CRN), group-relative advantage normalization, and policy updates using the STSS surrogate with reference-policy KL regularization.

Algorithm 1 Training Motion Predictor with *vis*-GRPO and STSS

Require: SFT-initialized policy π_θ, frozen reference policy π_{ref}, diffusion schedule $\{\alpha_t, \beta_t, \bar{\alpha}_t\}_{t=1}^{T}$
Require: Group size G, active subsampled step count K, clipping threshold ϵ, KL weight β, learning rate η

Initialize rollout policy: $\pi_{\theta_{\text{old}}} \leftarrow \pi_\theta$
for each training batch $(x_{\text{past}}, S, x_{\text{future}})$ **do**
  Synchronize rollout policy: $\pi_{\theta_{\text{old}}} \leftarrow \pi_\theta$
  Sample active timesteps $\mathcal{S} \subset \{1, \dots, T\}$ with $|\mathcal{S}| = K$ uniformly at random ▷ STSS
  // Step 1: Candidate Rollout with Dual-Noise Injection
  Sample initial latent $x_T^{(i)} \sim \mathcal{N}(0, \mathbf{I})$ for each candidate $i \in \{1, \dots, G\}$
  for $t = T$ **down to** 1 **do**
    Compute transition mean $\mu_{\theta_{\text{old}}}(x_t^{(i)}, t, q)$ with conditioning $q = (x_{\text{past}}, S)$
    if $t \in \mathcal{S}$ **then** ▷ Active step: independent exploration
      Sample candidate-specific noise $z_t^{(i)} \sim \mathcal{N}(0, \mathbf{I})$ for each $i \in \{1, \dots, G\}$
      $x_{t-1}^{(i)} \leftarrow \mu_{\theta_{\text{old}}}(x_t^{(i)}, t, q) + \sigma_t z_t^{(i)}$
      Record rollout log-probability $\log p_{\theta_{\text{old}}}(x_{t-1}^{(i)} \mid x_t^{(i)}, q)$
    else ▷ Inactive step: shared common random numbers
      Sample common noise $z_t^{\text{shared}} \sim \mathcal{N}(0, \mathbf{I})$ shared across all G candidates
      $x_{t-1}^{(i)} \leftarrow \mu_{\theta_{\text{old}}}(x_t^{(i)}, t, q) + \sigma_t z_t^{\text{shared}}$
    end if
    Cache latent state $x_t^{(i)}$
  end for
  Obtain generated motions $\hat{x}_{\text{future}}^{(i)} \leftarrow x_0^{(i)}$ for $i \in \{1, \dots, G\}$
  // Step 2: Reward Evaluation and Advantage Estimation
  for $i = 1$ to G **do**
    $r_i \leftarrow R(\hat{x}_{\text{future}}^{(i)}, x_{\text{future}}, S)$ ▷ Eq. 9
  end for
  Compute group advantage: $A_i \leftarrow \big(r_i - \text{mean}(\mathbf{r})\big) / \big(\text{std}(\mathbf{r}) + \delta\big)$
  // Step 3: Policy Update
  Compute the STSS surrogate $\hat{J}_{\text{STSS}}(\theta)$ using the cached transitions at $t \in \mathcal{S}$ ▷ Eq. 5
  Compute the clipped output-level KL penalty $\mathcal{D}_{\text{KL}}^{\text{clip}}(\theta) \leftarrow \min\big(D_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}}),\, 0.2\big)$
  Form the regularized policy objective $\mathcal{L}(\theta) \leftarrow -\hat{J}_{\text{STSS}}(\theta) + \beta \mathcal{D}_{\text{KL}}^{\text{clip}}(\theta)$
  Update θ using optimizer steps on $\nabla_\theta \mathcal{L}(\theta)$
end for

B IMPLEMENTATION DETAILS

B.1 HYPERPARAMETERS CONFIGURATION

This section provides the hyperparameters and configuration settings used to train ReasonMotion.

C REWARD MODEL DETAILS

The reward is a terminal, trajectory-level scalar:

$$R(\hat{x}) = \lambda_{\text{sim}}\, r_{\text{sim}}(\hat{x}, x) + \lambda_{\text{score}}\, r_{\text{score}}(\hat{x}) + \lambda_{\text{quality}}\, r_{\text{quality}}(\hat{x}), \tag{9}$$

with $\lambda_{\text{sim}} + \lambda_{\text{score}} + \lambda_{\text{quality}} = 1$. Each term takes values in $[0, 1]$, so the weights are directly comparable and no per-term rescaling is applied; advantages are standardised group-wise inside *vis*-GRPO.

Table 6: **Implementation hyperparameters.** Both sports share one backbone and one optimiser; they differ in the skeleton, the window, and the outcome scorer. Boxing is temporally downsampled from 60 to 30 fps.

(a) Diffusion backbone (TCD)

Parameter	FineFS	Boxing	H36M
Layers	12	12	12
Joints	24	25	17
Channels	64	64	64
Attention heads	8	8	8
Diffusion emb. dim	128	128	128
Time emb. dim	128	128	128
Feature emb. dim	16	16	16
Dense FFN dim	64	64	64
Axis order	T→S	T→S	T→S
Text encoder	SBERT 384-d, Cross-Attention		
β start / end	1×10^{-4} / 0.5		
Diffusion steps (T)	50, cosine schedule		

(b) Supervised stage

Parameter	FineFS	Boxing	H36M
Epochs	500	150	50
Batch size	32	32	16
Learning rate	1×10^{-3}	1×10^{-3}	2×10^{-4}
Optimiser	Adam	Adam	Adam
Weight decay	—	—	1×10^{-6}
Gradient clipping	1.0	1.0	1.0
LR schedule	const.	const.	warmup+cos.
CFG dropout	0.05	0.05	0.05
Input frames	30	0	25
Output frames	40	60	100
Window stride	5	10	1
Temporal downsample	1	2	1

(c) vis-GRPO stage

Parameter	FineFS	Boxing	H36M
Policy LR	5×10^{-6}	5×10^{-6}	3×10^{-6}
LR decay	×0.5/5k		—
Prompts per batch	2	2	2
Group size (G)	16	16	8
Inner epochs	2	2	2
Clip ratio (ϵ)	0.2	0.2	0.1
KL coef. (β)	0.05	0.05	1×10^{-2}
KL clip	0.2	0.2	0.2
Reference update	fixed	fixed	fixed
Sampling σ scale	1.0	1.15	1.0
STSS steps (K)	5	5	5
Gradient clipping	1.0	1.0	1.0

(d) Reward

Parameter	FineFS	Boxing	H36M
λ_{sim}	0.65	0.40	0.70
λ_{score}	0.25	0.50	0
λ_{quality}	0.10	0.10	0.30
r_{sim}	$\exp(-5\, d_{\text{GT}})$		
r_{quality}	$\exp(-10\, \lVert\ddot{x}\rVert)$		
Scorer	ST-GCN (GOE)	ST-GCN (off.)	—
Score mapping	$(s+5)/10$	$\exp(-s^2/2\sigma^2)$	—
Score clamp	$[0,1]$	$[0,1]$	—
Arm-extension gate	—	0.45 m	—
Remediation t_0	15	$[10,25]$	—

Let $\hat{x}, x \in \mathbb{R}^{L\times J\times 3}$ be the generated and recorded motions, and $m \in \{0,1\}^{L\times J}$ select the joint-frames the policy generates: all entries where the task is unconditional (boxing), the consequence phase alone where a prefix is observed (FineFS, Human3.6M).

C.1 Motion Similarity Reward

The motion similarity reward penalizes deviations from the ground-truth motion x over the policy-generated frames selected by the binary mask $m \in \{0,1\}^{L\times J}$. Let $d_{\text{GT}}(\hat{x}, x)$ denote the mean Euclidean distance per joint-frame:

$$d_{\text{GT}}(\hat{x}, x) = \frac{\sum_{t=1}^{L}\sum_{j=1}^{J} m_{t,j}\, \lVert \hat{x}_{t,j} - x_{t,j} \rVert_2}{\sum_{t=1}^{L}\sum_{j=1}^{J} m_{t,j}}. \tag{10}$$

The similarity reward maps this distance into $(0,1]$ via exponential decay:

$$r_{\text{sim}}(\hat{x}, x) = \exp\big(-5.0 \cdot d_{\text{GT}}(\hat{x}, x)\big). \tag{11}$$

C.2 Motion Quality Reward

The motion quality reward encourages physical plausibility by penalizing discrete joint acceleration. Let $\Delta^2 \hat{x}_{t,j} = \hat{x}_{t+1,j} - 2\hat{x}_{t,j} + \hat{x}_{t-1,j}$ denote the second-order finite difference at frame t. The mean acceleration magnitude over evaluated frames is:

$$\bar{a}(\hat{x}) = \frac{\sum_{t=2}^{L-1}\sum_{j=1}^{J} m_{t,j}\, \big\lVert \Delta^2 \hat{x}_{t,j} \big\rVert_2}{\sum_{t=2}^{L-1}\sum_{j=1}^{J} m_{t,j}}. \tag{12}$$

The quality reward is mapped into $(0, 1]$ via:

$$r_{\text{quality}}(\hat{x}) = \exp\big(-10.0 \cdot \bar{a}(\hat{x})\big). \tag{13}$$

C.3 Scoring Reward: FineFS

For FineFS, the HPP encoder operates on the fixed SMPL-24 skeleton with joint and bone channels ($6 \times 70 \times 24$). The scorer is trained against annotated Grade of Execution (GOE) labels. GOE is a figure-skating judging scale that measures how well a technical element is performed, rather than its intrinsic difficulty. Scores range from -5 to $+5$, with negative values indicating execution deficiencies and positive values indicating higher-quality execution. The predicted raw score s is linearly normalized onto $[0, 1]$:

$$r_{\text{score}} = \text{clip}\left(\frac{s - s_{\min}}{s_{\max} - s_{\min}},\, 0,\, 1\right), \qquad [s_{\min}, s_{\max}] = [-5, +5]. \tag{14}$$

It reaches **82.13%** pairwise ranking accuracy over held-out attempt pairs, rising to **90.22%** on pairs whose annotated scores differ by at least 2 points.

C.4 Scoring Reward: Boxing

Reaction-based judge. The judge shares the HPP backbone used on FineFS, retopologised to the Body25 skeleton and taking joint and bone channels over the full window ($6 \times 60 \times 25$), with an MLP regression head over spatio-temporally pooled features. Its supervision is the physical consequence of the punch: in the opponent-centric frame, the minimum distance attained over the window between either striking wrist (Body25 joints 4 and 7) and the opponent's head (joint 0),

$$D = \text{clip}\Big(\min_t \min\big(\|\mathbf{p}_{\text{rw}}(t) - \mathbf{p}^{\text{opp}}_{\text{head}}(t)\|,\ \|\mathbf{p}_{\text{lw}}(t) - \mathbf{p}^{\text{opp}}_{\text{head}}(t)\|\big),\ 0.05,\ 2.00\Big)\ [\text{m}]. \tag{15}$$

The opponent defines the label but is never an input: the judge regresses D in metres from the athlete's own motion alone, which is what makes it usable as a reward on generated motion for which no opponent exists. It is trained with MSE on the eight training bouts and evaluated on the four held-out bouts, over which it attains a Pearson correlation of 0.806 and a Spearman correlation of 0.810 against the measured distance, with a median absolute error of $11.8\,\text{cm}$ (MAE $= 17.0\,\text{cm}$) over $4{,}605$ windows (this validation set retains neutral windows and is therefore larger than the punch-only policy split). The discrimination that matters for reward is between landing and not landing: on windows whose measured distance is within $30\,\text{cm}$ (mean $19.2\,\text{cm}$) it predicts $22.4\,\text{cm}$, and on windows beyond $60\,\text{cm}$ (mean $96.0\,\text{cm}$) it predicts $77.1\,\text{cm}$, separating the two regimes by $54.7\,\text{cm}$. The predicted distance $\hat{D}$ becomes a reward through a Gaussian,

$$r_{\text{score}} = \exp\left(-\frac{\hat{D}^2}{2\sigma^2}\right), \qquad \sigma = 0.25\,\text{m}, \tag{16}$$

clamped to $[0, 1]$: $\hat{D} \le 5\,\text{cm}$ scores 0.98, $15\,\text{cm}$ scores 0.84, $25\,\text{cm}$ scores 0.61, $40\,\text{cm}$ scores 0.28, and anything beyond $75\,\text{cm}$ scores below 0.01. A linear map over the label range would spend most of its gradient distinguishing one miss from another; the Gaussian concentrates it inside a glove's reach, where contact is decided, and flattens outside it. The judge is frozen throughout *vis*-GRPO and evaluation, and no held-out bout enters its training.

Physics gate. A reaction-based judge that observes one fighter scores where the fist ends up, not how it got there. Because the boxing window is generated unconditionally, the predicted distance can also be reduced by carrying a folded guard forward, a motion that shortens the gap without throwing a punch. The gate makes the scoring term conditional on the arm actually extending. With shoulders at Body25 joints 2 and 5 and wrists at 4 and 7, we take the largest shoulder-to-wrist extension reached anywhere in the generated window,

$$\text{Reach}_{\max} = \max_t \max\big(\|\mathbf{p}_{\text{rw}}(t) - \mathbf{p}_{\text{rs}}(t)\|,\ \|\mathbf{p}_{\text{lw}}(t) - \mathbf{p}_{\text{ls}}(t)\|\big), \tag{17}$$

and multiply the scoring reward by $\mathbb{1}[\text{Reach}_{\max} \ge \tau]$ with $\tau = 0.45\,\text{m}$. On these reconstructed skeletons a folded guard measures 0.30–$0.35\,\text{m}$ and a committed punch 0.65–$0.73\,\text{m}$, so τ falls

between the two regimes rather than inside either. The gate multiplies λ_{score} alone; similarity and quality are untouched, so a gated candidate is not pushed away from the recorded motion, it simply earns nothing for the outcome. Extension is necessary for a strike without certifying a good one, which is why the technique measures in Table 5 report reach and peak wrist speed alongside contact distance. The operating point used throughout is $\lambda_{\text{sim}} = 0.40$, $\lambda_{\text{score}} = 0.50$, $\lambda_{\text{quality}} = 0.10$, with the scoring term clamped to $[0, 1]$ before gating.

C.5 Held-Out Scorers

Trans-R and Trans-D use the same joint-and-bone inputs as HPP, with factorised spatial and temporal attention, a hidden width of $d = 96$, and mean pooling over joints and frames. Neither incorporates an explicit skeletal graph or adjacency prior. Trans-R has 463k parameters, while Trans-D has 464k and predicts logits over 21 bins spanning $[-5, +5]$. Trans-D computes the expected score under the resulting softmax distribution, keeping its output within the label range. Both are trained on the FineFS training split using the same annotations as HPP, providing architectural diversity but not independence from HPP's training labels. Neither is used to select checkpoints from *vis*-GRPO.

The 2D CTR-GCN judge processes 17-joint COCO keypoint sequences ($3 \times 70 \times 17$) through 10 spatio-temporal graph convolution blocks, followed by global mean pooling and a linear regression head. Its training on MMFS excludes FineFS annotations, while its 2D inputs avoid reliance on 3D pose lifting. Table 7 reports the agreement of all scorers with ground-truth annotations on the FineFS test set.

Table 7: Agreement of HPP and the held-out judges with ground-truth annotations on the FineFS test set (n=181).

Model	Modality	Training Data	MAE ↓	MSE ↓	SRCC ↑	Pearson ↑
HPP (Reward)	3D (SMPL-24)	FineFS	**1.192**	**3.032**	**0.732**	0.810
Trans-R (Audit)	3D (SMPL-24)	FineFS	1.434	3.876	0.630	0.749
Trans-D (Audit)	3D (SMPL-24)	FineFS	1.533	4.110	0.605	0.738
CTR-GCN 3ch	2D (COCO-17)	MMFS	1.796	4.510	0.694	**0.819**

D Evaluation Metrics Definitions

MPJPE (Mean Per Joint Position Error). MPJPE is our primary evaluation metric for position accuracy. For a given frame t, MPJPE is defined as the average Euclidean distance between the predicted joint positions $\hat{x}_{j,t}$ and the ground-truth positions $x_{j,t}$:

$$\text{MPJPE}(t) = \frac{1}{J} \sum_{j=1}^{J} \left\| \hat{x}_{j,t} - x_{j,t} \right\|_2, \tag{18}$$

where J is the number of joints. We report two variants:

- **A-MPJPE:** Mean MPJPE averaged over all predicted future frames in the generated sequence.
- **F-MPJPE:** MPJPE computed only in the final predicted frame.

ADE (Average Displacement Error). For human motion prediction, we treat each predicted future pose as a vector $\hat{\mathbf{y}}_t \in \mathbb{R}^{3J}$. Given M generated hypotheses, ADE is computed using the best matching hypothesis:

$$\text{ADE} = \min_{m=1,\dots,M} \frac{1}{T} \sum_{t=1}^{T} \left\| \hat{\mathbf{y}}_t^{(m)} - \mathbf{y}_t \right\|_2. \tag{19}$$

FDE (Final Displacement Error). FDE measures the final-frame pose error using the best hypothesis:

$$\text{FDE} = \min_{m=1,\dots,M} \left\| \hat{\mathbf{y}}_T^{(m)} - \mathbf{y}_T \right\|_2. \tag{20}$$

MMADE. For multi-modal evaluation, each test sample may have a set of plausible ground-truth futures $\mathcal{Y}_i$. MMADE averages the best-hypothesis ADE over all valid ground-truth futures:

$$\text{MMADE}_i = \frac{1}{|\mathcal{Y}_i|} \sum_{\mathbf{y} \in \mathcal{Y}_i} \min_{m=1,\dots,M} \frac{1}{T} \sum_{t=1}^{T} \left\| \hat{\mathbf{y}}_t^{(m)} - \mathbf{y}_t \right\|_2 . \tag{21}$$

MMFDE. MMFDE averages the best-hypothesis final-frame error over all plausible ground-truth futures:

$$\text{MMFDE}_i = \frac{1}{|\mathcal{Y}_i|} \sum_{\mathbf{y} \in \mathcal{Y}_i} \min_{m=1,\dots,M} \left\| \hat{\mathbf{y}}_T^{(m)} - \mathbf{y}_T \right\|_2 . \tag{22}$$

where T is the prediction horizon, J is the number of joints, M is the number of hypotheses generated, $\hat{\mathbf{y}}_t^{(m)}$ denotes the m-th predicted pose vector at time t, $\mathbf{y}_t$ is the ground-truth pose vector, and $\|\cdot\|_2$ is the Euclidean norm. For multi-modal evaluation, $\mathcal{Y}_i$ denotes the set of plausible ground-truth futures for test sample i.

GOE. Grade of Execution measures execution quality in figure skating on a scale from -5 to $+5$. We report GOE scores predicted by the frozen held-out judges Trans-R, Trans-D, and 2D CTR-GCN described in Appendix C.

Mean acceleration. Mean ℓ_2 norm of the second temporal difference of joint positions, used as a smoothness proxy. Lower is smoother; note that values substantially below the ground-truth figure indicate suppression of motion content, not improvement.

Boxing technique measures. *Reach*: maximum wrist displacement from the shoulder along the punch axis. *Peak wrist speed*: maximum instantaneous wrist velocity within the window. *Contact distance*: minimum distance between the striking wrist and the target region on the opponent. *Hit rate*: fraction of windows in which contact distance falls within 30 cm.

E QUALITATIVE ANALYSIS

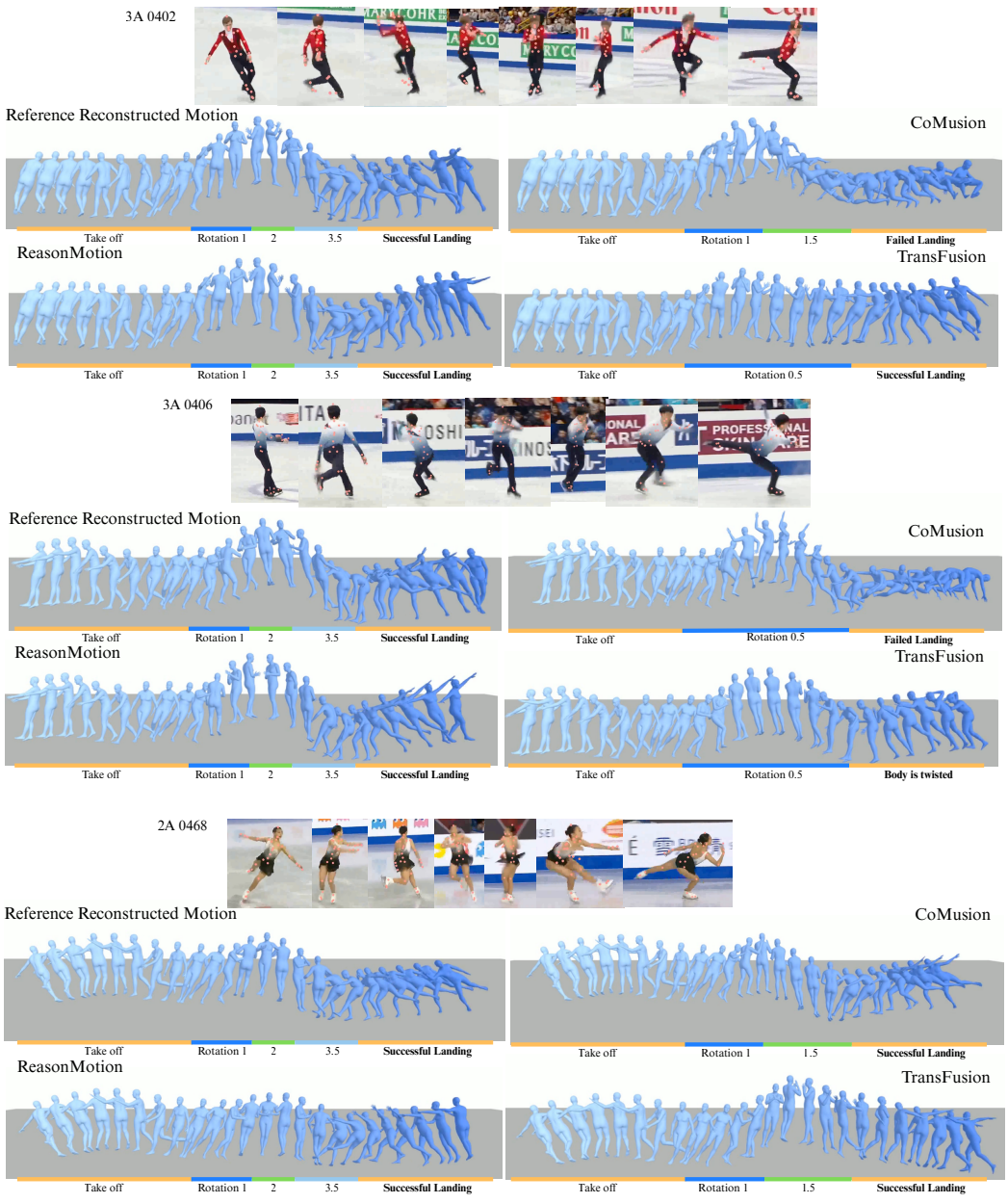


Figure 4: Qualitative analysis of successful jumps (GOE ≥ 0). ReasonMotion maintains high fidelity to the elite reference while preserving fluid rotational momentum, whereas baselines often collapse or remain static during the peak flight phase.

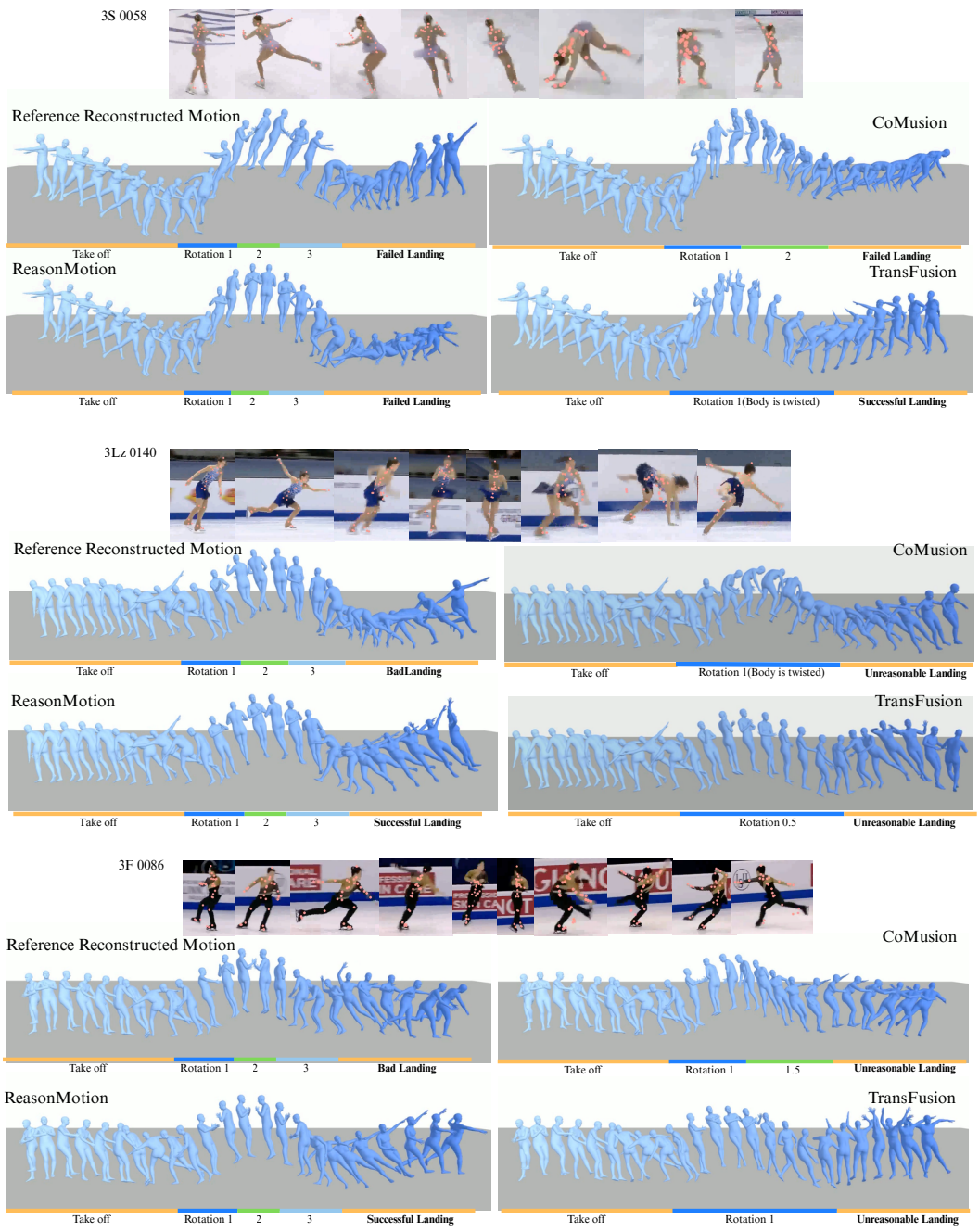


Figure 5: Qualitative analysis of suboptimal jumps (GOE < 0). In cases of failed references (e.g., 3S_0058), ReasonMotion attempts to optimize the kinematic trajectory. However, it respects physical constraints, producing a failed continuation when the initial takeoff is insufficient for recovery.

E.1 Emergent Learning Characteristics

Evolution of Optimality (Fig. 6B). We analyze how the model solves predicted failures through *vis*-GRPO, shifting from probabilistic realism to kinematic optimality. *Realism (Early Stages):* Initially, the model reflects the athlete's likely "Personal Style," predicting failure from a flawed takeoff. This serves as a risk assessment, showing the probable outcome if no corrections are made. *Optimality (Late Stages):* Through RL refinement, the model discovers extreme compensatory strategies ("Textbook Form"). It learns to save failing jumps via elite mid-air positional adjustments, providing a theoretical upper bound for successful landings despite suboptimal initial conditions.

Semantic Controllability (Fig. 6C). Using a fixed takeoff (Batch 2500), we vary the prompt from "Single" to "Quadruple." The model predicts successful outcomes for 1–3 rotations but faithfully identifies that the takeoff velocity cannot support a Quadruple, resulting in a fall. This demonstrates the model does not function as a "physics-defying" animator. Despite attempting to correct the rotational axis and optimize the vertical kinetic chain, the model still simulates a fall if the takeoff conditions (e.g., initial angular momentum or center-of-mass trajectory) are beyond the physical threshold for recovery. This highlights the model's utility as a consequence-aware motion adaptation framework, reflecting the limits of athletic recovery under suboptimal conditions.

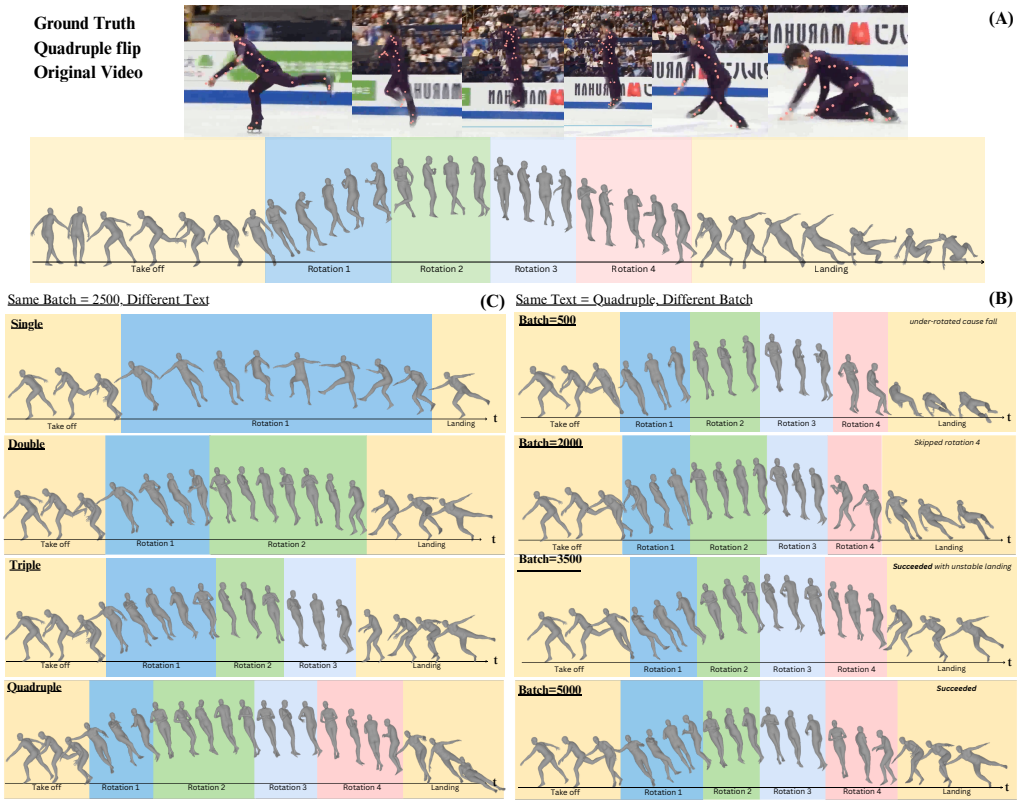


Figure 6: *Learning Evolution.* (A) Inference ground truth (a failed jump). (B) Optimal Recovery: By Batch 5000, the model discovers compensatory adjustments to "save" the landing. (C) Physical Honesty: At Batch 2500, the model correctly predicts failure (fall) for an insufficient Quadruple takeoff.

E.2 Visual Fidelity and Kinematic Realism

To evaluate technical adjustments, we utilize an instruction-grounded module for localized edits as interventions on the takeoff (Fig. 7). This "coaching sandbox" reveals how perturbations propagate through the kinetic chain. Unlike baselines (TransFusion) that yield "hunched" postures due to supervised dataset averaging, *vis*-GRPO prioritizes performance rewards to discover a stable, vertical

kinetic chain. This kinematic realism confirms that reward-guided reasoning produces elite-level motions, providing a reliable tool for risk assessment and strategy discovery.

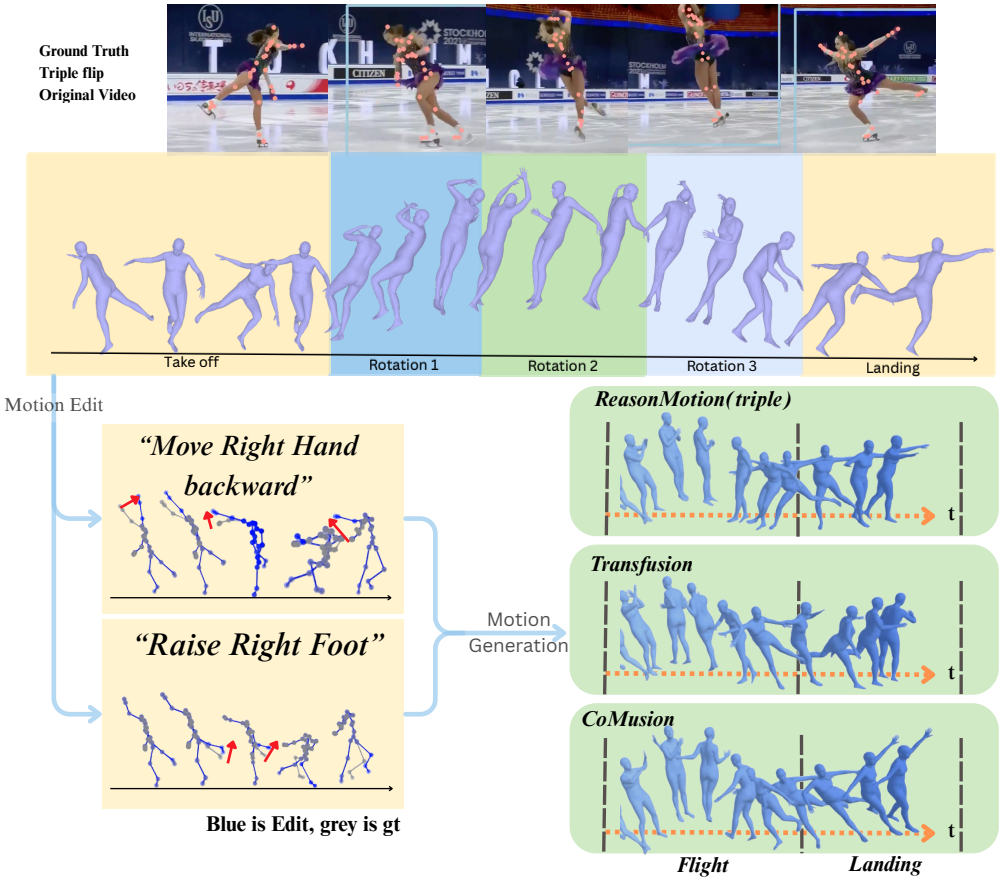


Figure 7: *Visual Comparison.* Top: Inference ground truth. Bottom Left: Localized edits on a takeoff (blue indicates edits, grey is ground truth). Bottom Right: Comparison. ReasonMotion maintains a vertical axis, whereas baselines produce unstable artifacts.

F HUMAN3.6M: FULL COMPARISON

Table 8: **Full comparison for motion prediction on Human3.6M**, 100 frames from a 25-frame prefix. The scoring reward is disabled ($\lambda_{\text{score}} = 0$), so gains reflect the optimiser alone.

Type	Model	ADE ↓	FDE ↓	MMADE ↓	MMFDE ↓
VAE	Motron	0.375	0.488	0.509	0.539
VAE	DSF	0.493	0.592	0.550	0.599
VAE	DLow	0.425	0.518	0.495	0.531
VAE	GSPS	0.389	0.496	0.476	0.525
VAE	DivSamp	0.370	0.485	0.475	0.516
DM	MotionDiff	0.411	0.509	0.508	0.536
DM	HumanMAC	0.369	0.480	0.509	0.545
DM	BeLFusion	0.372	0.474	0.473	0.507
DM	TransFusion	0.358	0.468	0.506	0.539
DM	CoMusion	0.350	0.458	0.494	0.506
DM	TCD (base)	0.331	0.431	0.482	0.503
DM	**ReasonMotion**	**0.325**	**0.423**	**0.466**	**0.492**

G OPTIMISER DIAGNOSTICS

This appendix supports Table 2: how checkpoints are selected and compute is counted, the full cost–quality curves behind the single operating point reported there, and the measurements behind our account of why the alternative estimators fall short.

G.1 CHECKPOINT SELECTION AND COMPUTE ACCOUNTING

Every run optimises the same reward through the same TCD backbone from the same supervised initialisation, and processes 32 sampled motions per training batch (2 conditions $\times$ G=16 candidates; $32 \times G$=1 for DDPO, following its published setting). We report each run at the checkpoint with the highest validation reward and, alongside it, the compute spent to reach that checkpoint. Wall-clock time is not comparable here, as the runs occupy different GPUs; we therefore count floating-point operations.

Per-step cost is measured directly with PyTorch's `FlopCounterMode` on the backbone rather than estimated: one forward pass over a batch of 32 costs 0.397 TFLOPs (12.42 GFLOPs per sample), and a backward pass costs 2$\times$ a forward. Expressing each training batch in forward-equivalents gives 393 for an un-subsampled estimator (T=50 rollout steps, 49 reference-policy evaluations, and forward and backward passes over the updated steps), 85 with STSS (K=5), and 70 for AWM, i.e. 156.1, 33.8 and 27.8 TFLOPs per batch. Cumulative compute is this quantity times the number of batches consumed. Since validation runs at a fixed batch interval, the reported figure is quantised to roughly $\pm$15 PFLOPs.

G.2 COST–QUALITY CURVES

Table 2 reports one checkpoint per run, which invites the question of whether the ordering is an artefact of that choice. Figure 3 removes the choice: it plots every checkpoint we evaluated, under the identical protocol, against the compute spent to reach it. The ordering in the table is the ordering of the curves.

G.3 WHY THE ALTERNATIVES FALL SHORT

The measurements in this subsection are internal to optimisation — a critic's error, the state of a normalisation buffer, the spread of rewards within a group — and are therefore read from the training logs rather than from the evaluation protocol used everywhere else in the paper.

PPO: the critic does not resolve the noise level. The critic is trained to predict the terminal reward from an intermediate latent x_t. We log the absolute advantage $|r - V(x_t, t)|$ separately for early ($t \in [35, 50]$), middle ($t \in [15, 35)$) and late ($t < 15$) denoising steps:

	$t \in [35, 50]$	$t \in [15, 35)$	$t < 15$
First 200 batches	0.0745	0.0732	0.0730
Final 800 batches	0.0742	0.0733	0.0732

The error is flat in t and does not improve with training, while the mean prediction tracks the mean reward closely (0.562 against 0.556 late in training). The critic therefore supplies little beyond a global constant: at $t < 15$, where x_t is nearly the finished motion and the reward should be largely determined, it is no more accurate than at $t = 50$. For scale, the reward spread within a group of candidates drawn from the same condition is 0.053 early in training and 0.028 late (measured on *vis*-GRPO, whose rewards are on the same scale), i.e. the baseline's error is of the order of, or larger than, the differences it must rank.

DDPO: one baseline per jump type. DDPO normalises rewards with per-prompt running statistics, keyed by the conditioning prompt. Our prompt is the rotation-count token, so the tracker holds four buffers for the entire dataset — the logged buffer count is 4 from the first batch to the last — and every take-off of a given jump type, easy or hard, is normalised against the same history. The advantage then carries the difficulty of the take-off as well as the quality of the policy, which is what motivated the group-relative baseline of Section 3.3. With $G{=}1$ there is additionally no within-condition comparison available at all.

AWM: stability and reward optimisation are exclusive. AWM does not differentiate through the sampled trajectory: it re-noises the final motion at a freshly drawn timestep and weights the pretraining denoising loss by the advantage, which regresses the policy onto its own high-reward samples. We ran it under two settings, both from the same supervised initialisation and the same reward:

Setting	lr	β_{KL}	A-MPJPE ↓	GOE (HPP) ↑	Div.
Published	$3{\cdot}10^{-4}$	0.001	165.02	1.09	0.91
Stabilised[†]	$5{\cdot}10^{-6}$	2.5	152.17	0.62	5.89
Supervised backbone	—	—	153.16	0.40	6.28

[†]Evaluated on the same test windows with $N{=}5$, but under an earlier run of the protocol that scored only with HPP and did not fix the generation seed.

Under the published setting the reward is optimised (validation reward $0.55 \rightarrow 0.64$) but diversity collapses within 200 batches and never recovers. Raising the KL coefficient until training is stable removes the collapse and holds diversity at the level of the backbone, but the policy then barely moves: over $11{,}000$ batches its validation reward drifts from 0.546 to 0.533, and its test GOE reaches 0.62 against the backbone's 0.40 while its geometry is indistinguishable from the backbone. We report the published setting in Table 2, which is also what our selection rule (highest validation reward) picks.

The effect of shared noise. STSS shares the injected noise across candidates at inactive steps, so the candidates of a group differ only at the K sampled steps. This is visible in the spread of rewards within a group, which falls from 0.053 to 0.022 early in training when STSS is enabled: the group's variation is concentrated on the transitions whose log-ratios enter the surrogate, which is what the group-relative baseline compares.

H Reward Weighting and the Onset of Reward Hacking

To rigorously validate that our chosen operating point ($\lambda_{\text{score}} = 0.25$) yields genuine physical refinements rather than exploiting the in-loop HPP scorer, we perform a boundary test by sweeping

Table 9: Reward-weight sweep on FineFS. Evaluation of geometric fidelity, execution quality, and out-of-bound predictions across varying scoring reward weights (λ_{score}). Trans-R and Trans-D serve as independent held-out judges. *Exc.* denotes the fraction of generated attempts scored beyond the annotated label range by the in-loop scorer. The deployed operating point is in bold.

	MPJPE (mm) ↓				Score ↑		
λ_{score}	A	F	Acc.	Div.	Trans-R	Trans-D	Exc. %
0.15	**142.63**	157.04	0.0178	2.44	+1.38	+0.88	0.0
0.25	142.67	**156.65**	**0.0227**	**2.63**	**+1.63**	**+1.29**	**0.0**
0.35	179.08	219.87	0.0210	2.62	+2.64	+1.94	10.5
0.65	270.72	285.25	0.0129	3.89	+2.60	+2.00	72.9
0.90	324.48	332.23	0.0139	5.30	+2.35	+2.01	81.7
Recorded	—	—	0.0693	—	+0.26	+0.33	0.0

the scoring reward weight (Table 9). At excessively high weights (e.g., 0.65 and 0.90), the model exhibits classic reward hacking: geometric fidelity severely degrades (A-MPJPE increasing to over 270 mm), and the in-loop scorer produces an unrealistic fraction of out-of-bound labels (Exc. reaching up to 81.7%). Crucially, in this degenerate regime, the independent judges decouple from the optimization objective—Trans-R scores decline from their peak, and Trans-D saturates, providing evidence that the in-loop reward is being exploited rather than reflecting uniformly improved execution quality.

In contrast, at our deployed operating point ($\lambda_{\text{score}} = 0.25$), the generated motions preserve geometric fidelity (142.67 mm) with zero out-of-bound labels. Here, both independent judges demonstrate consistent agreement with the in-loop scorer, granting substantial and reliable scores of 1.63 for Trans-R and 1.29 for Trans-D. Because the independent judges reject the hacked motions but highly rate the motions generated at $\lambda_{\text{score}} = 0.25$, this alignment demonstrates that our reward properly aligns with human-perceptible execution quality within the operating boundary. The recorded gains therefore reflect authentic kinematic improvements rather than numerical exploitation.

I Diversity Retention and Inter-Athlete Structure

Section 5.2 states that neither stage collapses onto a template. This appendix supplies the measurements behind that claim. Both stages are probed on the same population: 39 double-axel instances, giving $\binom{39}{2} = 741$ pairs. The remediation stage is probed by sweeping t_0, since a policy converging on a canonical corrected form would converge harder as t_0 grows and discards more of the athlete's own trajectory. The prediction stage is probed by partitioning the pairs by how dissimilar their takeoffs were, since a predictor mapping every instruction onto one optimal landing would erase that ordering by the final frame.

Table 10: **The remediation stage does not collapse athletes onto a template**, 39 double-axel instances. *div*: mean pairwise ℓ_2 distance between flattened 70-frame position vectors over all 741 pairs of remediated attempts. *ret*: value as a fraction of the recorded population's.

		t_0				
Policy		5	10	15	20	25
Recorded (39 attempts)				11.981		
Supervised	*div*	11.169	11.330	11.448	11.376	11.365
	ret	93.2%	94.6%	95.6%	94.9%	94.9%
vis-GRPO	*div*	9.369	9.327	9.264	9.330	9.259
	ret	78.2%	77.9%	77.3%	77.9%	77.3%

Reward optimisation reduces inter-athlete spread to 77–78% of the recorded population's, and that number is flat: across a five-fold range of t_0 it moves by 0.9 percentage points, with no monotone trend. Template convergence would instead deepen with edit strength. The supervised policy retains

93–96% under the identical perturbation, so the gap is a fixed cost of optimising against a reward rather than a drift that accumulates as more of the motion is rewritten. Note that *div* is computed on flattened position vectors and is not directly comparable in scale to the per-phase distances in Table 11; only the ratio *ret* is interpreted here.

Table 11: **The prediction stage does not collapse athletes onto a template**, same 741 pairs, partitioned into thirds by take-off dissimilarity. Windows: take-off F0–29 (observed prefix), flight F30–49, landing F50–69. Parenthesised values are the change from the flight phase.

Take-off group	Pairs	Take-off	Flight	Landing
High (top 33%)	247	170.83	202.55	184.37 (-18.2)
Medium (mid 33%)	247	222.64	254.98	215.46 (-39.5)
Low (bottom 33%)	247	307.09	337.16	258.01 (-79.2)

All 39 instances carry the same instruction, yet pairwise distance at landing remains 184–258 mm and preserves the take-off ordering, where collapse onto one optimal execution would drive it toward zero. The reduction is also not uniform across the attempt. Every group *expands* through the aerial phase ($1.19\times$, $1.15\times$ and $1.10\times$ from the most to the least similar third), so even near-identical take-offs receive measurably distinct corrective pathways. Each then *contracts* at landing in proportion to its initial dissimilarity (-18.2, -39.5 and -79.2 mm). Athletes entering with different flaws receive different aerial compensations and are brought to a common stable landing. This is the structure coaching should have: the landing of a well-executed jump is biomechanically convergent, and individualised landings would be individualised failure.

J Expert Evaluation Protocol

In this appendix, we provide the comprehensive details of our human evaluation experiment, including participant information, stimuli selection, interface implementation, specific evaluation phases, and statistical analysis procedures.

J.1 Participant Information

Recruitment Criteria. To ensure high-quality and domain-specific feedback, we established rigorous qualification criteria for our evaluators. Participants are classified as "figure skating experts" if they meet at least one of the following criteria:

- Certified professional figure skating coaches actively training competitive athletes.
- Active competitive/elite figure skating athletes with specialized experience in multi-rotation jumps (double and triple jumps).

Evaluator Cohort. We conducted a human evaluation with 10 expert evaluators: 4 professional coaches and 6 competitive international level figure skating athletes.

Table 12: **Evaluator Demographics.** Our study cohort ($n = 10$) consists of elite figure skating professionals with extensive competitive and instructional experience.

Role	**Level / Experience**	**Count**	**Technical Milestone**
Professional Coach	Int'l/National (Avg. 6.8 yrs)	4	Coached Int'l/National Medalists
Elite Athlete	National Team (Avg. 9.5 yrs)	6	Confirmed Triple-jump proficiency

Individual Evaluator Background. To ensure the highest standard of technical assessment, our cohort comprises elite members of the figure skating community. The coaches possess an average of 6.8 years of professional instructional experience, with several having successfully trained athletes for international ISU competitions. The athlete group consists of active national-level competitors

with an average of 9.5 years of experience, all of whom have confirmed technical proficiency in triple-jump execution (Table 13).

Table 13: **Detailed Expertise Profiles of Evaluators** ($n = 10$)**.**

ID	Role	Experience	Competitive / Coaching Milestone
C-1	Coach	10 Years	Trained athletes for International Competitions
C-2	Coach	8 Years	Trained athletes for National Competitions
C-3	Coach	5 Years	Former National Athlete; National-level Coach
C-4	Coach	4 Years	Former National Athlete; National-level Coach
A-1	Athlete	15 Years	National Team Member; Triple-jump proficient
A-2	Athlete	16 Years	National Team Member; Triple-jump proficient
A-3	Athlete	5 Years	International competitor; Double-jump proficient
A-4	Athlete	5 Years	International competitor; Double-jump proficient
A-5	Athlete	7 Years	National Team Member; Triple-jump proficient
A-6	Athlete	9 Years	National Team Member; Triple-jump proficient

Ethics and Consent. Participation in this study is entirely voluntary and conducted anonymously. The platform collects the names and roles of the evaluators solely to prevent duplicate entries and to enable session-saving (progress can be saved mid-survey and resumed by entering the same identity, which is crucial for busy professional coaches and athletes). Evaluators are explicitly informed that their identity will not be linked to their responses.

J.2 Stimuli and Setup

Case Selection (Pools). Our evaluation stimuli are sampled from the testing set of the proposed dataset. Specifically, we focus on double and triple jumps because:

1. Multi-rotational jumps are highly representative of technical difficulty in figure skating.
2. These sequences exhibit high-fidelity motion capture quality, ensuring that artifacts from data collection do not bias the evaluation of generative performance.

The specific pool consists of 15 jump cases divided into four categories:

- **2-Turns Negative**: Unsuccessful double jumps (e.g., `2A_0276`, `2T_0006`).
- **2-Turns Positive**: Successful double jumps (e.g., `2A_0394`, `2A_0468`).
- **3-Turns Negative**: Unsuccessful triple jumps (e.g., `3A_0057`, `3A_0477`, `3F_0086`, `3Lo_0152`, `3Lz_0140`).
- **3-Turns Positive**: Successful triple jumps (e.g., `3A_0160`, `3A_0402`, `3A_0421`, `3F_0095`, `3Lo_0039`, `3Lz_0239`).

Models Compared. We compare four distinct conditions:

1. **Reference Reconstructed motion**: Reconstructed human motion extracted by HybrIK Li et al. (2021) from raw video.
2. **ReasonMotion_rl** (Our proposed model).
3. **CoMusion** Sun & Chowdhary (2024) (Baseline).
4. **TransFusion** Tian et al. (2024) (Baseline).

Presentation Format. All animations are presented as high-fidelity, loop-rendered 3D human body sequences using the SMPL body model overlay, allowing experts to scrutinize subtle joint placements and takeoff mechanics. All model identifiers are completely blinded and randomized under placeholder labels (e.g., "Video 1" to "Video 4").

Randomization and Counterbalancing. To avoid order bias:

- The section orders (Realism, GOE Ranking, Training Utility, and Kinematic Plausibility) and trial orders are randomized.
- The relative column positions of the videos within each grid are randomized per trial.
- Shuffling is performed deterministically using a user-specific stable hashing seed generated from their Name and Role, ensuring a uniform distribution across conditions while maintaining consistent state across browser refreshes.

J.3 Phase-Specific Instructions and Interfaces

Phase 1: Realism Test(15 Questions, 60 Videos Evaluated).

- **Instruction Prompt**: "Which of these look real? Please select all videos that appear realistic (you can select more than one !!!!). Note: Due to potential errors in motion capture and human shape rendering (e.g., twisting, body shape distortion, hunchback, etc.), please use the realism of the movement posture and jump as the primary basis for judgment."
- **Interface Layout**: A grid of four videos corresponding to the four models.
- **Evaluation Format**: Multiple-choice checkbox. Evaluators can select any number of videos (from 1 to 4) they deem realistic.

Phase 2: Quality (GOE) Ranking (8 Questions, 32 Videos Evaluated).

- **Instruction Prompt**: "GOE Execution Quality Ranking. Please rank the following videos by execution quality (GOE) from high to low."
- **Interface Layout**: A grid of four blinded videos with individual dropdown menus.
- **Evaluation Format**: Forced-unique ranking from 1 (Best) to 4 (Worst). The frontend ensures that evaluators cannot assign the same rank to multiple videos in a single trial.

Phase 3: Training Utility (4–24 Questions, 8 Videos Evaluated). This phase specifically targets corrective generation, using unsuccessful jumps (Negative pools) as input.

- **Instruction Prompt**: "Does watching the corrected motion video help your training? In this stage, we use the take-off phase of the jump on the left as input to generate a video of a jump that theoretically has a higher GOE in the subsequent 'air spin' and 'landing.' If you were the skater on the left and performed a poor jump, could the video on the right serve as a demonstration to help you understand how to improve? The right video is a vertical stack of three videos: a real motion (grey) and two corrected motions, one demonstrating a lighter correction (light blue), and one demonstrating a stronger correction (dark blue)."
- **Interface Layout**: A split-screen showing the original failed jump on the left and the overlay/corrected stack on the right.
- **Questions and Logic**:
  1. *Primary Question*: "Is this corrected video helpful for your training?" (Binary choice: `Yes` / `No`).
  2. *Conditional Sub-Questions* (rendered only if the primary answer is `Yes`):
     - **Executability**: "How high is the 'executability' of the physical adjustments presented in real training?" (5-point Likert scale, anchors: 1: Very Low, 2: Low, 3: Fair, 4: High, 5: Very High).
     - **Blind Spot Detection**: "Does the video point out blind spots?" (`Yes` / `No`).
     - **Problem Understanding**: "Better understanding of issue?" (`Yes` / `No`).
     - **Improvement Visualization**: "Help you visualize improvement?" (`Yes` / `No`).
     - **Inspiration Source**: "Main inspiration?" (Multi-select checkboxes: `Arms`, `Landing`, `Core`).

Phase 4: Kinematic Plausibility (32 Questions, 8 Videos Evaluated). To rigorously assess kinematic violations, each model's output is evaluated independently.

- **Instruction Prompt**: "Is the video physically plausible? Please rate the plausibility of the movement."
- **Questions and Scale Anchors**:
  1. *Air Rotations*: "How many air rotations did you observe in the video?" (Discretized selection: `0`, `0.5`, `1.0`, `1.5`, `2.0`, `2.5`, `3.0`, `3.5`, `4.0`).
  2. *Takeoff Phase*: "Takeoff: Is the movement physically plausible and coherent? (e.g., is the power generation natural? Are there any takeoffs from a complete standstill or unrealistic pulling of the center of gravity?)"
  3. *Flight Phase*: "Flight: Is the movement physically plausible and coherent?"
  4. *Landing Phase*: "Landing: Is the movement physically plausible and coherent?"

  All three sub-phases are rated using a 5-point Likert scale (anchors: 1: Very Poor, 2: Poor, 3: Fair, 4: Good, 5: Excellent).

J.4 Measurement and Aggregation

For each criterion, we aggregate the responses from our professional evaluators as follows.

Realism is reported as the selection rate, i.e., the percentage of trials in which evaluators selected a given model's output as visually realistic.

Expertise (GOE) is reported as the Top-1 rate, i.e., the percentage of trials in which a model's output was ranked first in technical quality by the evaluators.

Kinematic Plausibility is measured along three dimensions: (1) rotation count error, computed as the absolute difference between the evaluator-counted rotations and the ground truth rotation count; (2) flight phase rating, reported as the mean Likert score (1–5) across all evaluators; and (3) landing phase rating, similarly reported as the mean Likert score (1–5).

Training Utility is aggregated as the percentage of evaluators who indicated that the generated motion was helpful as a coaching demonstration. This metric is only applicable to ReasonMotion outputs, as baseline models do not support instruction-based motion generation.

J.5 Results

Across 10 expert evaluators, we collected a total of 600 video-level realism judgments in Phase 1 (15 trials $\times$ 4 videos $\times$ 10 evaluators), 320 video-level GOE rankings in Phase 2 (8 trials $\times$ 4 videos $\times$ 10 evaluators), and 80 case-level responses each in Phase 3 and Phase 4 (8 cases $\times$ 10 evaluators).

Table 14: Realism & Execution Quality (Phase 1 & 2). Asterisks (*) indicate methods statistically significantly lower ($p < 0.05$) than ReasonMotion. ReasonMotion achieved a realism rate of 94.7%, preferred over the reference reconstructed motion, likely due to denoising and idealization.

Model	Realism (%) ↑	GOE Top-1 (%) ↑	Mean Rank ↓
Ref. Reconstructed motion	53.3*	16.2	2.08
ReasonMotion (Ours)	**94.7**	**82.5**	**1.18**
CoMusion	3.3*	0.0	3.55*
TransFusion	3.3*	1.2	3.20*

Table 15: Training Utility (ReasonMotion only) (Phase 3). The data confirms a clear split: athletes find light corrections easier to execute and focus on landing, whereas coaches find strong corrections more helpful for identifying core kinematic issues.

Correction Type	Helpful (%) ↑	Executability (1–5) ↑	Blind Spot (%) ↑	Prob. Und. (%) ↑
Light (Athletes)	79.2%	3.84 ± 0.81	78.9%	78.9%
Strong (Coaches)	**87.5%**	3.29 ± 0.59	**100.0%**	**92.9%**

Table 16: Inspiration Source Distribution (ReasonMotion only) (Phase 3). Athletes primarily focus on landing adjustments under light correction, while coaches shift attention to core biomechanics under strong correction.

Correction Type	Arms (%)	Core (%)	Landing (%)
Light (Athletes)	15.8%	15.8%	**68.4%**
Strong (Coaches)	14.3%	**78.6%**	7.1%

Table 17: Kinematic Plausibility (Phase 4). Likert scores (1–5) reported as mean ± std. ReasonMotion scores higher than reference reconstructed motion in several kinematic categories, likely because our model synthesizes "idealized" motion based on instructions.

Model	Rot. Error ↓	Takeoff ↑	Flight ↑	Landing ↑	Overall Avg ↑
Reference Reconstructed Motion	0.35 ± 0.71	3.35 ± 1.11	3.40 ± 0.97	3.00 ± 1.18	3.25
ReasonMotion (Ours)	$\mathbf{0.33 \pm 0.43}$	$\mathbf{3.90 \pm 0.94}$	$\mathbf{3.85 \pm 0.85}$	$\mathbf{3.85 \pm 0.91}$	**3.87**
CoMusion	2.05 ± 0.63	2.85 ± 1.19	1.75 ± 0.77	1.65 ± 0.85	2.08
TransFusion	1.80 ± 0.90	2.60 ± 1.16	1.75 ± 0.70	1.85 ± 0.73	2.07

J.6 Qualitative Feedback and User Impressions

Beyond numerical ratings, we collected open-ended feedback to assess the model's potential for real-world integration into training workflows. The experts highlighted several key advantages:

- **Novelty and Instructional Value:** One senior coach (C-2) noted that the model's approach to motion adaptation is *"highly novel and promises to be exceptionally helpful for day-to-day technical training."* This underscores the model's role as a practical framework for sports coaching rather than just a visualization tool.
- **Precision in Error Diagnosis:** An elite athlete (A-2) emphasized the system's accuracy, stating it was *"very cool to see a system that can effectively identify specific jumps that need improvement"* and expressed a strong willingness to use such a tool in actual practice.
- **Simulation and Prototyping:** Another athlete (A-6) remarked on the practical utility of the framework, finding it particularly useful for *"simulating potential jump strategies,"* allowing athletes to visualize successful outcomes before attempting them physically.

These comments suggest that ReasonMotion provides a unique "coaching sandbox" that aligns with the professional cognitive models of elite skaters.